# 32-bit Arm Cortex-M0+ with 5V Support, CAN-FD, PTC, and Advanced Analog

## SAM C20/C21 Family

Product Page Links

## Features

### Operating Conditions

- 2.7V – 5.5V
  - -40°C to +125°C, DC to 48 MHz
  - -40°C to +85°C, DC to 64 MHz

### Core

- Arm® Cortex®-M0+ CPU running at up to 48 MHz or 64 MHz:
  - Single-cycle hardware multiplier
  - Micro Trace Buffer
  - Memory Protection Unit (MPU)

### Memories

- 32/64/128/256 KB in-system self-programmable Flash
- 1/2/4/8 KB independent self-programmable Flash for EEPROM emulation
- 4/8/16/32 KB SRAM main memory

### System

- Power-on Reset (POR) and Brown-out Detection (BOD)
- Internal and external clock options with 48 MHz to 96 MHz Fractional Digital Phase Locked Loop (FDPLL96M)
- External Interrupt Controller (EIC) (Interrupt pin debouncing is only available in SAM C20/C21 N)
- 16 external interrupts
  - Hardware debouncing (only available in SAM C20/C21 N)
- One non-maskable interrupt
- Two-pin Serial Wire Debug (SWD) programming, test, and debugging interface

### Low-Power

- Idle and Standby Sleep modes
- SleepWalking peripherals

### Peripherals

- Hardware Divide and Square Root Accelerator (DIVAS)
- 12-channel Direct Memory Access Controller (DMAC)
- 12-channel Event System
- Up to eight 16-bit Timer/Counters (TC), configurable as either (see Note):
  **Note:** Maximum and minimum capture is only available in the SAM C21N devices.
  - One 16-bit TC with compare/capture channels
  - One 8-bit TC with compare/capture channels


© 2026 Microchip Technology Inc. and its subsidiaries

  - One 32-bit TC with compare/capture channels, by using two TCs
- Two 24-bit and one 16-bit Timer/Counter for Control (TCC), with extended functions:
  - Up to eight PWM channels on each 24-bit TCC
  - Up to two PWM channels on each 16-bit TCC
  - Up to four compare channels with optional complementary output
  - Generation of synchronized pulse width modulation (PWM) pattern across port pins
  - Deterministic fault protection, fast decay and configurable dead-time between complementary output
  - Dithering that increase resolution with up to 5 bit and reduce quantization error
- Frequency Meter (The division reference clock is only available in the SAM C21N)
- 32-bit Real Time Counter (RTC) with clock/calendar function
- Watchdog Timer (WDT)
- CRC-32 generator
- Up to two Controller Area Network (CAN) interfaces in the SAM C21:
  - CAN 2.0A/B and CAN-FD (ISO 11898-1:2015)
    - Each CAN interface have two selectable pin locations to switch between two external CAN transceivers (without the need for an external switch)
- Up to eight Serial Communication Interfaces (SERCOM), each configurable to operate as either:
  - USART with full-duplex and single-wire half-duplex configuration
  - $I^2C$ up to 3.4 MHz (Except SERCOM6 and SERCOM7)
  - SPI
  - LIN host/client
  - RS-485
  - PMBus
- One Configurable Custom Logic (CCL)
- Up to Two 12-bit, 1 Msps Analog-to-Digital Converter (ADC) with up to 12 channels each (20 unique channels)
  - Differential and single-ended input
  - Automatic offset and gain error compensation
  - Oversampling and decimation in hardware to support 13, 14, 15 or 16-bit resolution
- One 16-bit Sigma-Delta Analog-to-Digital Converter (SDADC) with up to 3 differential channels in the SAM C21
- 10-bit, 350 ksps Digital-to-Analog Converter (DAC) in the SAM C21
- Up to four Analog Comparators (AC) with Window Compare function
- Integrated Temperature Sensor in the SAM C21
- Peripheral Touch Controller (PTC)
  - 256-Channel capacitive touch and proximity sensing

**I/O**

- Up to 84 programmable I/O pins

**Qualification**

- AEC - Q100 Grade 1 (-40°C to 125°C)

**Packages**

- 100-pin TQFP


© 2026 Microchip Technology Inc. and its subsidiaries

- 64-pin TQFP, VQFN
- 56-pin WLCSP
- 48-pin TQFP, VQFN
- 32-pin TQFP, VQFN

**General**

- Drop in compatible with SAM D20 and SAM D21 (see Note)

**Note:** Only applicable to 32-pin, 48-pin, and 64-pin TQFP and VQFN packages.


© 2026 Microchip Technology Inc. and its subsidiaries

# Table of Contents

Features ... 1

1. Configuration Summary ... 14
2. Ordering Information ... 18
3. Block Diagram ... 19
4. Pinout ... 21
   - 4.1. SAM C21E / SAM C20E ... 21
   - 4.2. SAM C21G / SAM C20G ... 22
   - 4.3. SAM C21J / SAM C20J ... 23
   - 4.4. SAM C21N / SAM C20N ... 25
5. Signal Descriptions List ... 26
6. I/O Multiplexing and Considerations ... 28
   - 6.1. Multiplexed Signals ... 28
   - 6.2. Other Functions ... 33
7. Power Supply and Start-Up Considerations ... 36
   - 7.1. Power Domain Overview ... 36
   - 7.2. Power Supply Considerations ... 37
   - 7.3. Power-Up ... 38
   - 7.4. Power-on Reset and Brown-out Detector ... 38
8. Product Mapping ... 40
9. Memories ... 44
   - 9.1. Embedded Memories ... 44
   - 9.2. Physical Memory Map ... 44
   - 9.3. NVM User Row Mapping ... 44
   - 9.4. NVM Software Calibration Area Mapping ... 46
   - 9.5. NVM Temperature Calibration Area Mapping, SAM C21 ... 46
   - 9.6. Serial Number ... 47
10. Processor and Architecture ... 48
    - 10.1. Cortex M0+ Processor ... 48
    - 10.2. Nested Vector Interrupt Controller ... 49
    - 10.3. Micro Trace Buffer ... 52
    - 10.4. High-Speed Bus System ... 53
11. PAC - Peripheral Access Controller ... 56
    - 11.1. Overview ... 56
    - 11.2. Features ... 56
    - 11.3. Block Diagram ... 56
    - 11.4. Product Dependencies ... 56
    - 11.5. Functional Description ... 57
    - 11.6. Register Summary ... 61
    - 11.7. Register Description ... 61


© 2026 Microchip Technology Inc. and its subsidiaries

12. Peripherals Configuration Summary ..... 80
12.1. SAM C20/C21 N ..... 80
12.2. SAM C20/C21 E/G/J ..... 84

13. DSU - Device Service Unit ..... 88
13.1. Overview ..... 88
13.2. Features ..... 88
13.3. Block Diagram ..... 89
13.4. Signal Description ..... 89
13.5. Product Dependencies ..... 89
13.6. Debug Operation ..... 90
13.7. Chip Erase ..... 92
13.8. Programming ..... 92
13.9. Intellectual Property Protection ..... 93
13.10. Program and Debug Interface Disable (PDID) ..... 94
13.11. Device Identification ..... 94
13.12. Functional Description ..... 95
13.13. Register Summary ..... 100
13.14. Register Description ..... 101

14. DIVAS – Divide and Square Root Accelerator ..... 124
14.1. Overview ..... 124
14.2. Features ..... 124
14.3. Block Diagram ..... 124
14.4. Signal Description ..... 124
14.5. Product Dependencies ..... 124
14.6. Functional Description ..... 125
14.7. Register Summary ..... 128
14.8. Register Description ..... 128

15. Clock System ..... 136
15.1. Clock Distribution ..... 136
15.2. Synchronous and Asynchronous Clocks ..... 137
15.3. Register Synchronization ..... 137
15.4. Enabling a Peripheral ..... 138
15.5. On-demand, Clock Requests ..... 139
15.6. Power Consumption vs. Speed ..... 139
15.7. Clocks after Reset ..... 139

16. GCLK - Generic Clock Controller ..... 141
16.1. Overview ..... 141
16.2. Features ..... 141
16.3. Block Diagram ..... 141
16.4. Signal Description ..... 142
16.5. Product Dependencies ..... 142
16.6. Functional Description ..... 143
16.7. Register Summary ..... 149
16.8. Register Description ..... 150

17. MCLK – Main Clock ..... 159


© 2026 Microchip Technology Inc. and its subsidiaries

17.1. Overview....159
17.2. Features....159
17.3. Block Diagram....159
17.4. Signal Description....159
17.5. Product Dependencies....159
17.6. Functional Description....161
17.7. Register Summary....164
17.8. Register Description....164

18. RSTC – Reset Controller....180
18.1. Overview....180
18.2. Features....180
18.3. Block Diagram....180
18.4. Signal Description....180
18.5. Product Dependencies....180
18.6. Functional Description....181
18.7. Register Summary....183
18.8. Register Description....183

19. PM - Power Manager....185
19.1. Overview....185
19.2. Features....185
19.3. Block Diagram....185
19.4. Signal Description....185
19.5. Product Dependencies....185
19.6. Functional Description....186
19.7. Register Summary....191
19.8. Register Description....191

20. OSCCTRL – Oscillators Controller....194
20.1. Overview....194
20.2. Features....194
20.3. Block Diagram....195
20.4. Signal Description....195
20.5. Product Dependencies....195
20.6. Functional Description....196
20.7. Register Summary....205
20.8. Register Description....206

21. OSC32KCTRL – 32.768 kHz Oscillators Controller....232
21.1. Overview....232
21.2. Features....232
21.3. Block Diagram....233
21.4. Signal Description....233
21.5. Product Dependencies....233
21.6. Functional Description....235
21.7. Register Summary....241
21.8. Register Description....241

22. SUPC – Supply Controller....256


© 2026 Microchip Technology Inc. and its subsidiaries

22.1. Overview....256
22.2. Features....256
22.3. Block Diagram....256
22.4. Signal Description....257
22.5. Product Dependencies....257
22.6. Functional Description....258
22.7. Register Summary....262
22.8. Register Description....262

23. WDT – Watchdog Timer....274
23.1. Overview....274
23.2. Features....274
23.3. Block Diagram....274
23.4. Signal Description....275
23.5. Product Dependencies....275
23.6. Functional Description....276
23.7. Register Summary....281
23.8. Register Description....281

24. RTC – Real-Time Counter....290
24.1. Overview....290
24.2. Features....290
24.3. Block Diagram....290
24.4. Signal Description....291
24.5. Product Dependencies....291
24.6. Functional Description....293
24.7. Register Summary - Mode 0 - 32-Bit Counter....298
24.8. Register Description - Mode 0 - 32-Bit Counter....298
24.9. Register Summary - Mode 1 - 16-Bit Counter....311
24.10. Register Description - Mode 1 - 16-Bit Counter....311
24.11. Register Summary - Mode 2 - Clock/Calendar....325
24.12. Register Description - Mode 2 - Clock/Calendar....325

25. DMAC – Direct Memory Access Controller....339
25.1. Overview....339
25.2. Features....339
25.3. Block Diagram....341
25.4. Signal Description....341
25.5. Product Dependencies....341
25.6. Functional Description....342
25.7. Register Summary....364
25.8. Register Description....365
25.9. Register Summary - SRAM....395
25.10. Register Description - SRAM....395

26. EIC – External Interrupt Controller....402
26.1. Overview....402
26.2. Features....402
26.3. Block Diagram....402
26.4. Signal Description....402


© 2026 Microchip Technology Inc. and its subsidiaries

26.5. Product Dependencies....403
26.6. Functional Description....404
26.7. Register Summary....411
26.8. Register Description....411

27. NVMCTRL – Nonvolatile Memory Controller....427
27.1. Overview....427
27.2. Features....427
27.3. Block Diagram....427
27.4. Signal Description....428
27.5. Product Dependencies....428
27.6. Functional Description....429
27.7. Register Summary....437
27.8. Register Description....437

28. PORT - I/O Pin Controller....453
28.1. Overview....453
28.2. Features....453
28.3. Block Diagram....454
28.4. Signal Description....454
28.5. Product Dependencies....454
28.6. Functional Description....456
28.7. Register Summary....462
28.8. Register Description....463

29. Event System (EVSYS)....482
29.1. Overview....482
29.2. Features....482
29.3. Block Diagram....482
29.4. Signal Description....482
29.5. Product Dependencies....483
29.6. Functional Description....484
29.7. Register Summary....489
29.8. Register Description....489

30. SERCOM – Serial Communication Interface....503
30.1. Overview....503
30.2. Features....503
30.3. Block Diagram....504
30.4. Signal Description....504
30.5. Product Dependencies....504
30.6. Functional Description....506

31. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter....511
31.1. Overview....511
31.2. USART Features....511
31.3. Block Diagram....512
31.4. Signal Description....512
31.5. Product Dependencies....512
31.6. Functional Description....514


© 2026 Microchip Technology Inc. and its subsidiaries

31.7. Register Summary ..... 528
31.8. Register Description ..... 528

32. SERCOM SPI – SERCOM Serial Peripheral Interface ..... 550
32.1. Overview ..... 550
32.2. Features ..... 550
32.3. Block Diagram ..... 551
32.4. Signal Description ..... 551
32.5. Product Dependencies ..... 551
32.6. Functional Description ..... 553
32.7. Register Summary ..... 562
32.8. Register Description ..... 562

33. SERCOM I²C – Inter-Integrated Circuit ..... 580
33.1. Overview ..... 580
33.2. Features ..... 580
33.3. Block Diagram ..... 581
33.4. Signal Description ..... 581
33.5. Product Dependencies ..... 581
33.6. Functional Description ..... 583
33.7. Register Summary - I2C Client ..... 602
33.8. Register Description - I²C Client ..... 602
33.9. Register Summary - I2C Host ..... 616
33.10. Register Description - I²C Host ..... 616

34. CAN - Control Area Network (SAM C21 Only) ..... 634
34.1. Overview ..... 634
34.2. Features ..... 634
34.3. Block Diagram ..... 634
34.4. Signal Description ..... 635
34.5. Product Dependencies ..... 635
34.6. Functional Description ..... 636
34.7. Register Summary ..... 656
34.8. Register Description ..... 659
34.9. Message RAM ..... 725

35. Timer/Counter (TC) ..... 734
35.1. Overview ..... 734
35.2. Features ..... 734
35.3. Block Diagram ..... 735
35.4. Signal Description ..... 735
35.5. Product Dependencies ..... 736
35.6. Functional Description ..... 737
35.7. Register Description ..... 752

36. Timer/Counter for Control Applications (TCC) ..... 818
36.1. Overview ..... 818
36.2. Features ..... 818
36.3. Block Diagram ..... 819
36.4. Signal Description ..... 819


© 2026 Microchip Technology Inc. and its subsidiaries

36.5. Product Dependencies....820
36.6. Functional Description....821
36.7. Register Summary....853
36.8. Register Description....855

37. Configurable Custom Logic (CCL)....901
37.1. Overview....901
37.2. Features....901
37.3. Block Diagram....901
37.4. Signal Description....902
37.5. Product Dependencies....902
37.6. Functional Description....903
37.7. Register Summary....914
37.8. Register Description....914

38. ADC - Analog-to-Digital Converter....919
38.1. Overview....919
38.2. Features....919
38.3. Block Diagram....920
38.4. Signal Description....920
38.5. Product Dependencies....920
38.6. Functional Description....922
38.7. Register Summary....933
38.8. Register Description....933

39. SDADC – Sigma-Delta Analog-to-Digital Converter (SAM C21 only)....961
39.1. Overview....961
39.2. Features....961
39.3. Block Diagram....962
39.4. Signal Description....962
39.5. Product Dependencies....963
39.6. Functional Description....964
39.7. Register Summary....972
39.8. Register Description....972

40. AC – Analog Comparators....998
40.1. Overview....998
40.2. Features....998
40.3. Block Diagram....999
40.4. Signal Description....1000
40.5. Product Dependencies....1000
40.6. Functional Description....1001
40.7. Register Summary....1011
40.8. Register Description....1011

41. DAC – Digital-to-Analog Converter (SAM C21 only)....1027
41.1. Overview....1027
41.2. Features....1027
41.3. Block Diagram....1027
41.4. Signal Description....1027


© 2026 Microchip Technology Inc. and its subsidiaries

41.5. Product Dependencies....1027
41.6. Functional Description....1029
41.7. Register Summary....1034
41.8. Register Description....1034

42. Peripheral Touch Controller (PTC)....1047
42.1. Overview....1047
42.2. Features....1047
42.3. Block Diagram....1048
42.4. Signal Description....1049
42.5. System Dependencies....1049
42.6. Functional Description....1050

43. TSENS – Temperature Sensor....1051
43.1. Overview....1051
43.2. Features....1051
43.3. Block Diagram....1051
43.4. Signal Description....1051
43.5. Product Dependencies....1051
43.6. Functional Description....1053
43.7. Register Summary....1057
43.8. Register Description....1057

44. FREQM – Frequency Meter....1074
44.1. Overview....1074
44.2. Features....1074
44.3. Block Diagram....1074
44.4. Signal Description....1074
44.5. Product Dependencies....1074
44.6. Functional Description....1076
44.7. Register Summary....1079
44.8. Register Description....1079

45. Electrical Characteristics 85°C (SAM C20/C21 E/G/J)....1089
45.1. Disclaimer....1089
45.2. Absolute Maximum Ratings....1089
45.3. General Operating Ratings....1089
45.4. Injection Current....1090
45.5. Supply Characteristics....1090
45.6. Maximum Clock Frequencies....1091
45.7. Power Consumption....1092
45.8. Wake-Up Time....1094
45.9. I/O Pin Characteristics....1094
45.10. Analog Characteristics....1095
45.11. NVM Characteristics....1108
45.12. Oscillator Characteristics....1109
45.13. Timing Characteristics....1116

46. Electrical Characteristics 105°C (SAM C20/C21 E/G/J)....1119
46.1. Disclaimer....1119


© 2026 Microchip Technology Inc. and its subsidiaries

46.2. General Operating Ratings....1119
46.3. Power Consumption....1119
46.4. Analog Characteristics....1120
46.5. NVM Characteristics....1124
46.6. Oscillator Characteristics....1125

47. Electrical Characteristics 105°C (SAM C20/C21 N)....1128
47.1. Disclaimer....1128
47.2. General Operating Ratings....1128
47.3. Power Consumption....1128
47.4. Analog Characteristics....1129
47.5. NVM Characteristics....1140
47.6. Oscillator Characteristics....1140

48. Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 E/G/J)....1144
48.1. Disclaimer....1144
48.2. General Operating Ratings....1144
48.3. Supply Characteristics....1144
48.4. Power Consumption....1144
48.5. I/O Pin Characteristics....1145
48.6. Analog Characteristics....1146
48.7. NVM Characteristics....1152
48.8. Oscillator Characteristics....1152
48.9. Timing Characteristics....1155

49. Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 N)....1158
49.1. Disclaimer....1158
49.2. Power Consumption....1158
49.3. Analog Characteristics....1159
49.4. Oscillator Characteristics....1165
49.5. Timing Characteristics....1168

50. Appendix A....1170
50.1. ISELED FULL License Enabled Functional Devices....1170
50.2. Ordering Information....1170

51. Appendix B....1171
51.1. SIL 2-Enabled Functional Safety Devices....1171
51.2. Ordering Information....1171

52. Packaging Information....1172
52.1. Package Marking Information....1172
52.2. Thermal Considerations....1176
52.3. Package Drawings....1177
52.4. Soldering Profile....1222

53. Schematic Checklist....1223
53.1. Introduction....1223
53.2. Operation in Noisy Environment....1223
53.3. Power Supply....1223
53.4. External Analog Reference Connections....1225


© 2026 Microchip Technology Inc. and its subsidiaries

53.5. External Reset Circuit....1226
53.6. Unused or Unconnected Pins....1228
53.7. Clocks and Crystal Oscillators....1228
53.8. Programming and Debug Ports....1231

54. Revision History....1235
54.1. Revision History....1235
54.2. Rev KJ - 11/2016....1246
54.3. Rev J - 10/2016....1246
54.4. Rev I - 09/2016....1246
54.5. Rev H - 05/2016....1248
54.6. Rev G - 04/2015....1248
54.7. Rev F - 02/2015....1249
54.8. Rev E - 12/2015....1250
54.9. Rev D - 09/2015....1250
54.10. Rev C - 09/2015....1251
54.11. Rev B - 06/2015....1251
54.12. Rev A - 04/2015....1251

55. Product Identification System....1252

Microchip Information....1253
Trademarks....1253
Legal Notice....1253
Microchip Devices Code Protection Feature....1253

Product Page Links....1254


© 2026 Microchip Technology Inc. and its subsidiaries

# 1. Configuration Summary

**Table 1-1.** SAM C20 Device-Specific Features

| Device | Flash (KB) | SRAM (KB) |
|---|---|---|
| ATSAMC20E15 | 32 | 4 |
| ATSAMC20E16 | 64 | 8 |
| ATSAMC20E17 | 128 | 16 |
| ATSAMC20E18 | 256 | 32 |
| ATSAMC20G15 | 32 | 4 |
| ATSAMC20G16 | 64 | 8 |
| ATSAMC20G17 | 128 | 16 |
| ATSAMC20G18 | 256 | 32 |
| ATSAMC20J15 | 32 | 4 |
| ATSAMC20J16 | 64 | 8 |
| ATSAMC20J17 | 128 | 16 |
| ATSAMC20J18 | 256 | 32 |
| ATSAMC20N17 | 128 | 16 |
| ATSAMC20N18 | 256 | 32 |

**Table 1-2.** SAM C21 Device-specific Features

| Device | Flash (KB) | SRAM (KB) |
|---|---|---|
| ATSAMC21E15 | 32 | 4 |
| ATSAMC21E16 | 64 | 8 |
| ATSAMC21E17 | 128 | 16 |
| ATSAMC21E18 | 256 | 32 |
| ATSAMC21G15 | 32 | 4 |
| ATSAMC21G16 | 64 | 8 |
| ATSAMC21G17 | 128 | 16 |
| ATSAMC21G18 | 256 | 32 |
| ATSAMC21J15 | 32 | 4 |
| ATSAMC21J16 | 64 | 8 |
| ATSAMC21J17 | 128 | 16 |
| ATSAMC21J18 | 256 | 32 |
| ATSAMC21N17 | 128 | 16 |
| ATSAMC21N18 | 256 | 32 |

**Table 1-3.** SAM C21 Family Features

| | SAM C21N | SAM C21J | SAM C21G | SAM C21E |
|---|---|---|---|---|
| Pins | 100 | 64 (56 for WLCSP) | 48 | 32 |
| General Purpose I/O-pins (GPIOs) | 84 | 52 (44 for WLCSP) | 38 | 26 |
| Flash | 256/128 KB | 256/128/64/32 KB | 256/128/64/32 KB | 256/128/64/32 KB |
| Flash RWW section | 8/4 KB | 8/4/2/1 KB | 8/4/2/1 KB | 8/4/2/1 KB |
| System SRAM | 32/16 KB | 32/16/8/4 KB | 32/16/8/4 KB | 32/16/8/4 KB |
| Timer Counter (TC) instances | 8 | 5 | 5 | 5 |
| Waveform/PWM output / Capture Input channels per TC instance | 2 | 2 | 2 | 2 |
| TC Maximum and Minimum Capture | Yes | No | No | No |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 1-3.** SAM C21 Family Features (continued)

| | SAM C21N | SAM C21J | SAM C21G | SAM C21E |
|---|---|---|---|---|
| Timer Counter for Control (TCC) instances | 3 | 3 | 3 | 3 |
| Waveform/PWM output channels per TCC (TCC0/ TCC1/TCC2) | 8/4/2 | 8/4/2 | 8/4/2 | 8/4/2 |
| DMA channels | 12 | 12 | 12 | 12 |
| CAN interface | 2 | 2 | 2 | 1 |
| Configurable Custom Logic (CCL) (LUTs) | 4 | 4 | 4 | 4 |
| Serial Communication Interface (SERCOM) instances | 8 | 6 | 6 | 4 |
| Divide and Square Root Accelerator (DIVAS) | Yes | Yes | Yes | Yes |
| Analog-to-Digital Converter (ADC) channels | 22 | 20 | 14 | 10 |
| Analog-to-Digital Converter (ADC) instances | 2 | 2 | 2 | 2 |
| Sigma-Delta Analog-to-Digital Converter (SDADC) channels | 3 | 3 | 2 | 1 |
| Analog Comparators (AC) | 4 | 4 | 4 | 4 |
| Digital-to-Analog Converter (DAC) channels | 1 | 1 | 1 | 1 |
| Temperature Sensor (TSENS)(1) | 1 | 1 | 1 | 1 |
| Real-Time Counter (RTC) | Yes | Yes | Yes | Yes |
| RTC alarms | 1 | 1 | 1 | 1 |
| RTC compare values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values |
| External Interrupt lines | 16 with HW debouncing | 16 | 16 | 16 |
| Peripheral Touch Controller (PTC) Number of self-capacitance channels (Y-lines) | 32 | 32 | 22 | 16 |
| Peripheral Touch Controller (PTC) Number of mutual-capacitance channels (X x Y lines) | 256 (16x16) | 256 (16x16) | 121 (11x11) | 64 (8x8) |
| Frequency Meter (FREQM) reference clock divider | Yes | Yes | Yes | Yes |
| Maximum CPU frequency | 48 / 64 MHz (2) | | | |
| Packages | TQFP | VQFN<br>TQFP<br>WLCSP | VQFN<br>TQFP | VQFN<br>TQFP |
| Oscillators | 32.768 kHz crystal oscillator (XOSC32K)<br>0.4-32 MHz crystal oscillator (XOSC)<br>32.768 kHz internal oscillator (OSC32K)<br>32.768 kHz ultra-low-power internal oscillator (OSCULP32K)<br>48 MHz high-accuracy internal oscillator (OSC48M)<br>96 MHz Fractional Digital Phased Locked Loop (FDPLL96M) | | | |
| Event System channels | 12 | 12 | 12 | 12 |
| SW Debug Interface | Yes | Yes | Yes | Yes |
| Watchdog Timer (WDT) | Yes | Yes | Yes | Yes |

**Notes:**

1. TSENS is not available in AEC-Q100 qualified device part numbers.
2. See the Ordering Information section to determine the parts running at 48 MHz or 64 MHz.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 1-4.** SAM C20 Family Features

| | SAM C20N | SAM C20J | SAM C20G | SAM C20E |
|---|---|---|---|---|
| Pins | 100 | 64 (56 for WLCSP) | 48 (44 for WLCSP) | 32 |
| General Purpose I/O-pins (GPIOs) | 84 | 52 | 38 | 26 |
| Flash | 256/128 KB | 256/128/64/32 KB | 256/128/64/32 KB | 256/128/64/32 KB |
| Flash RWW section | 8/4 KB | 8/4/2/1 KB | 8/4/2/1 KB | 8/4/2/1 KB |
| System SRAM | 32/16 KB | 32/16/8/4 KB | 32/16/8/4 KB | 32/16/8/4 KB |
| Timer Counter (TC) instances | 8 | 5 | 5 | 5 |
| Waveform/PWM output / Capture Input channels per TC instance | 2 | 2 | 2 | 2 |
| TC Maximum and Minimum Capture | Yes | No | No | No |
| Timer Counter for Control (TCC) instances | 3 | 3 | 3 | 3 |
| Waveform output channels per TCC (TCC0/TCC1/TCC2) | 8/4/2 | 8/4/2 | 8/4/2 | 8/4/2 |
| DMA channels | 12 | 6 | 6 | 6 |
| Configurable Custom Logic (CCL) (LUTs) | 4 | 4 | 4 | 4 |
| Serial Communication Interface (SERCOM) instances | 8 | 4 | 4 | 4 |
| Divide and Square Root Accelerator (DIVAS) | Yes | Yes | Yes | Yes |
| Analog-to-Digital Converter (ADC) channels | 12 | 12 | 12 | 10 |
| Analog-to-Digital Converter (ADC) instances | 1 | 1 | 1 | 1 |
| Analog Comparators (AC) | 4 | 2 | 2 | 2 |
| Real-Time Counter (RTC) | Yes | Yes | Yes | Yes |
| RTC alarms | 1 | 1 | 1 | 1 |
| RTC compare values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values | One 32-bit value or two 16-bit values |
| External Interrupt lines | 16 with HW debouncing | 16 | 16 | 16 |
| Peripheral Touch Controller (PTC) Number of self-capacitance channels (Y-lines) | 32 | 32 | 22 | 16 |
| Peripheral Touch Controller (PTC) Number of mutual-capacitance channels (X x Y lines) | 256 (16x16) | 256 (16x16) | 121 (11x11) | 64 (8x8) |
| Frequency Meter (FREQM) reference clock divider | Yes | Yes | Yes | Yes |
| Maximum CPU frequency | 48/64 MHz (1) | | | |
| Packages | TQFP | VQFN<br>TQFP<br>WLCSP | VQFN<br>TQFP | VQFN<br>TQFP |
| Oscillators | 32.768 kHz crystal oscillator (XOSC32K)<br>0.4-32 MHz crystal oscillator (XOSC)<br>32.768 kHz internal oscillator (OSC32K)<br>32.768 kHz ultra-low-power internal oscillator (OSCULP32K)<br>48 MHz high-accuracy internal oscillator (OSC48M)<br>96 MHz Fractional Digital Phased Locked Loop (FDPLL96M) | | | |
| Event System channels | 12 | 12 | 12 | 12 |
| SW Debug Interface | Yes | Yes | Yes | Yes |
| Watchdog Timer (WDT) | Yes | Yes | Yes | Yes |

**Note:**

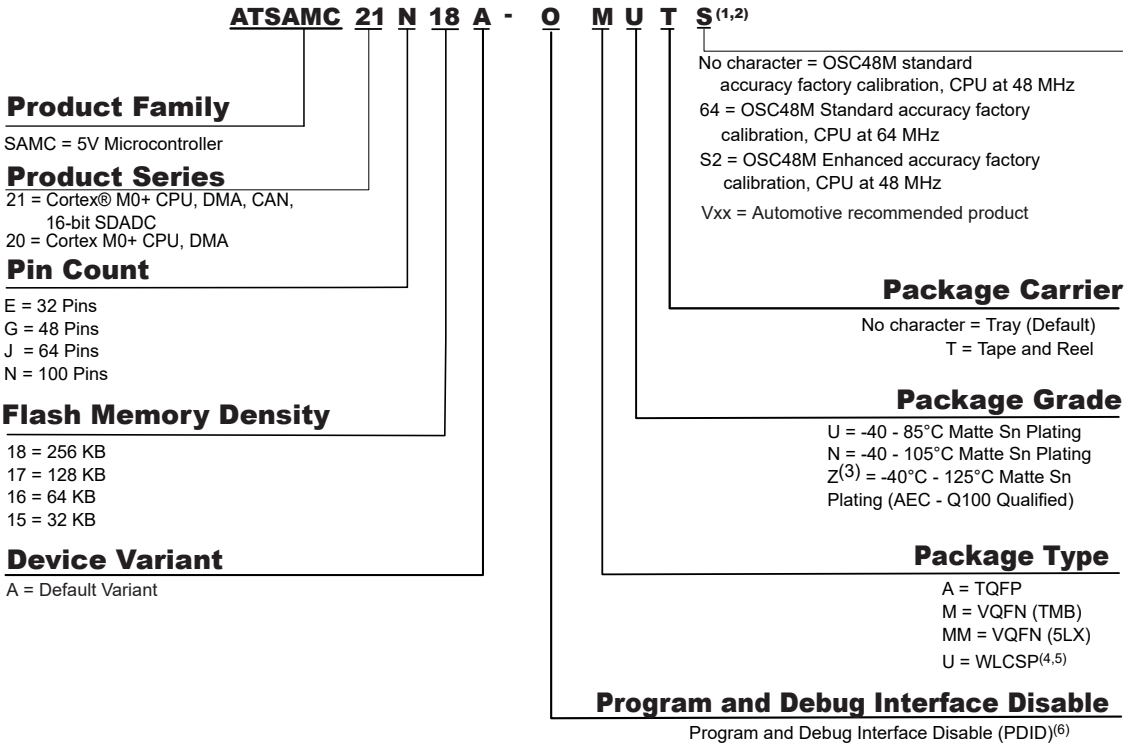
1. See the Ordering Information section to determine the parts running at 48 MHz or 64 MHz.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
I/O Multiplexing and Considerations


© 2026 Microchip Technology Inc. and its subsidiaries

# 2. Ordering Information



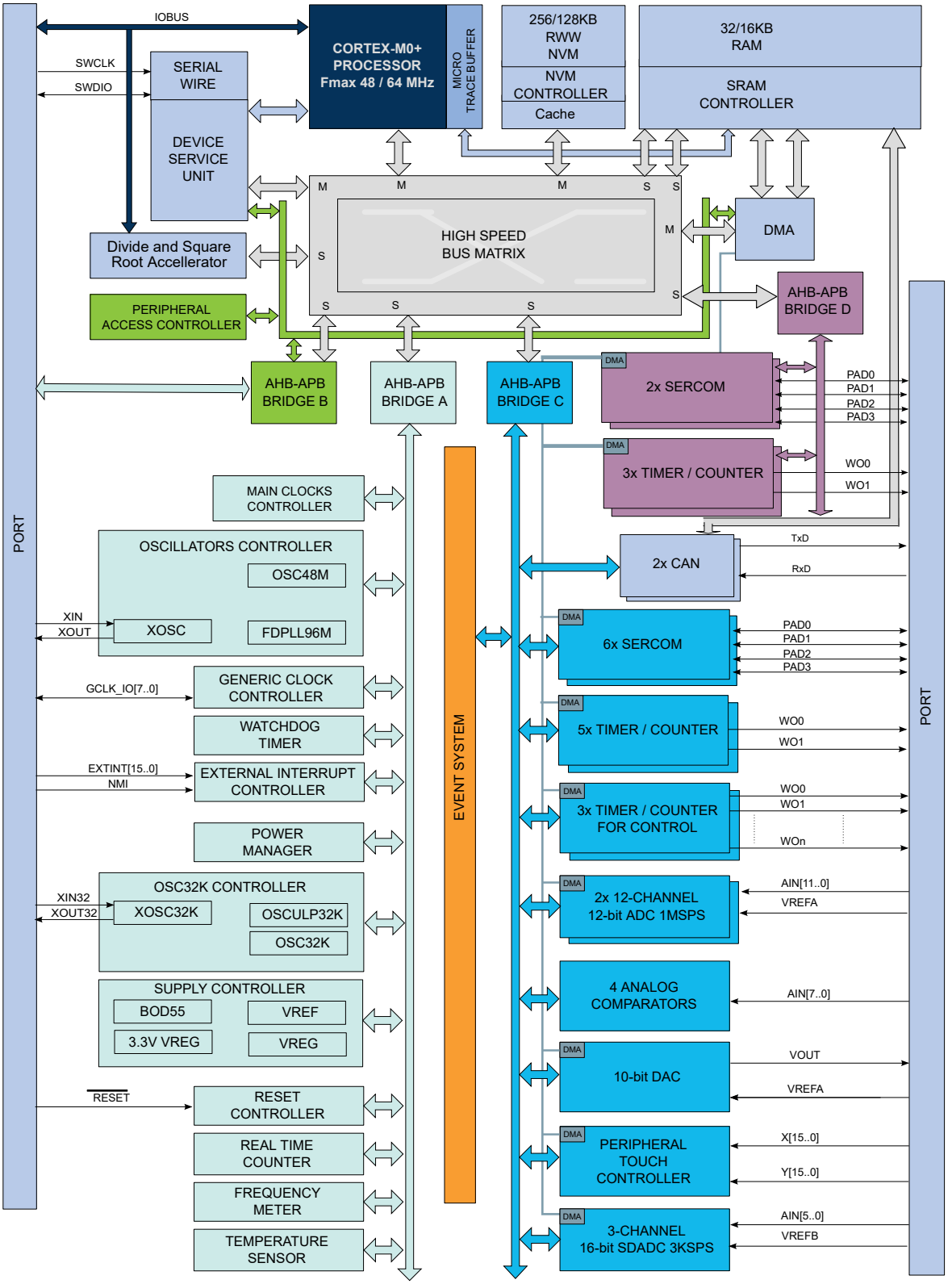

**Notes:**

1. Not all combinations are valid. The available ordering numbers are listed in the Configuration Summary.
2. The SAM C20/C21 N product is available for the 105°C temperature grade and AEC - Q100 Grade 1 (125°C temperature).
3. The AEC - Q100 Grade 1 qualified version is offered for SAM C20/C21 E/G/J in the TQFP and VQFN packages only, and for SAM C20/C21 N in the TQFP package only. The VQFN package will have wettable flanks, and both TQFP and VQFN packages are assembled with gold bond wires. The TSENS is not available in AEC - Q100 qualified device part numbers.
4. Devices in the WLCSP package include a factory programmed Bootloader. Contact your local Microchip sales office for additional information.
5. The WLCSP package type is available only with the package Grade U.
6. Contact a local Microchip sales office for additional information about PDID functionality.


© 2026 Microchip Technology Inc. and its subsidiaries

# 3. Block Diagram

**Figure 3-1.** System Block Diagram for SAM C20/C21





© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** Not all features are available for all devices. Please refer to Table 1-3 and Table 1-4 to determine feature availability for the particular device.

**Related Links**

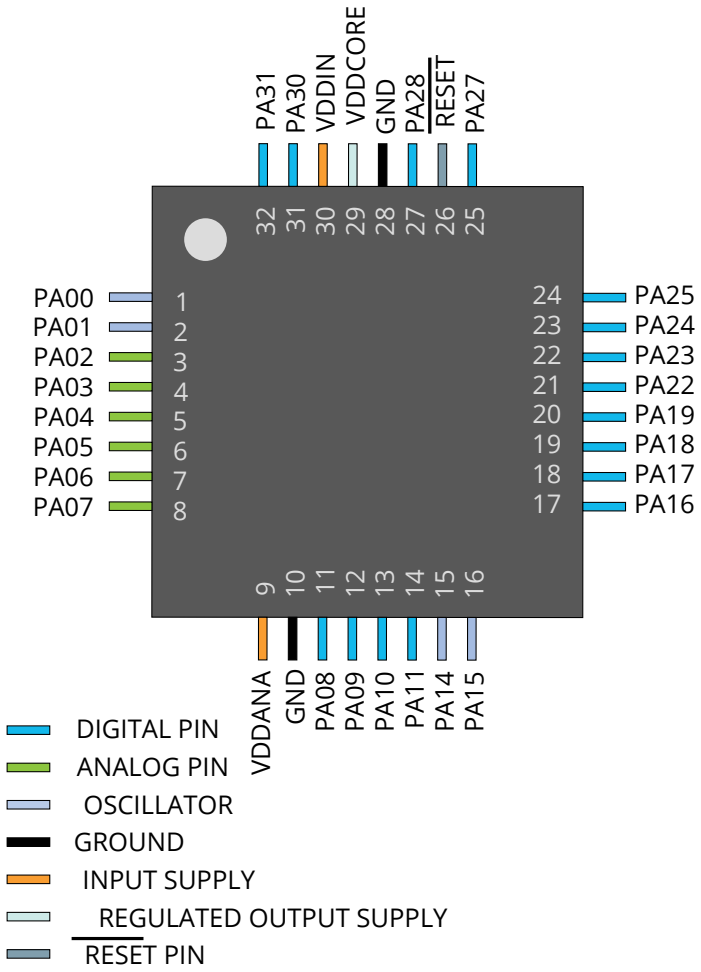
TCC Configurations
Multiplexed Signals


© 2026 Microchip Technology Inc. and its subsidiaries

# 4. Pinout

## 4.1. SAM C21E / SAM C20E

### 4.1.1. VQFN32/TQFP32

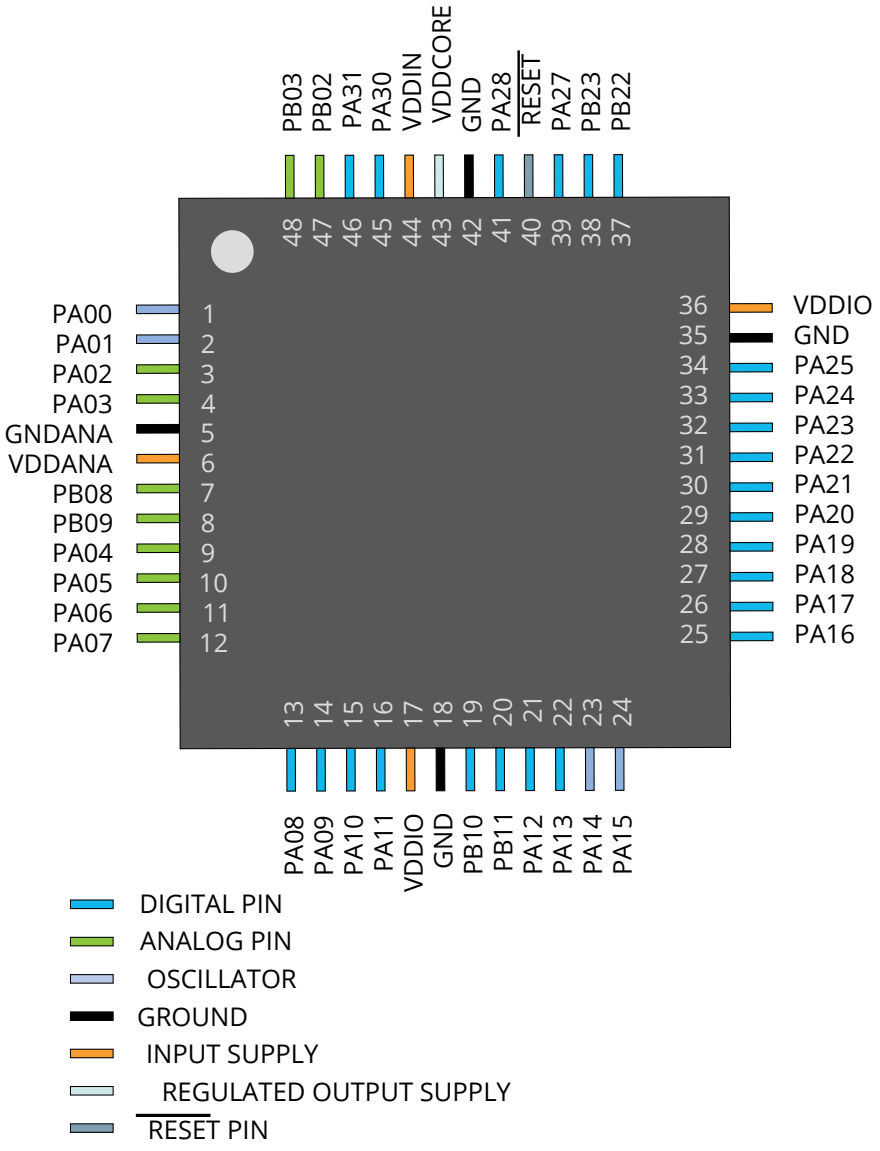




© 2026 Microchip Technology Inc. and its subsidiaries

## 4.2. SAM C21G / SAM C20G

### 4.2.1. VQFN48 / TQFP48

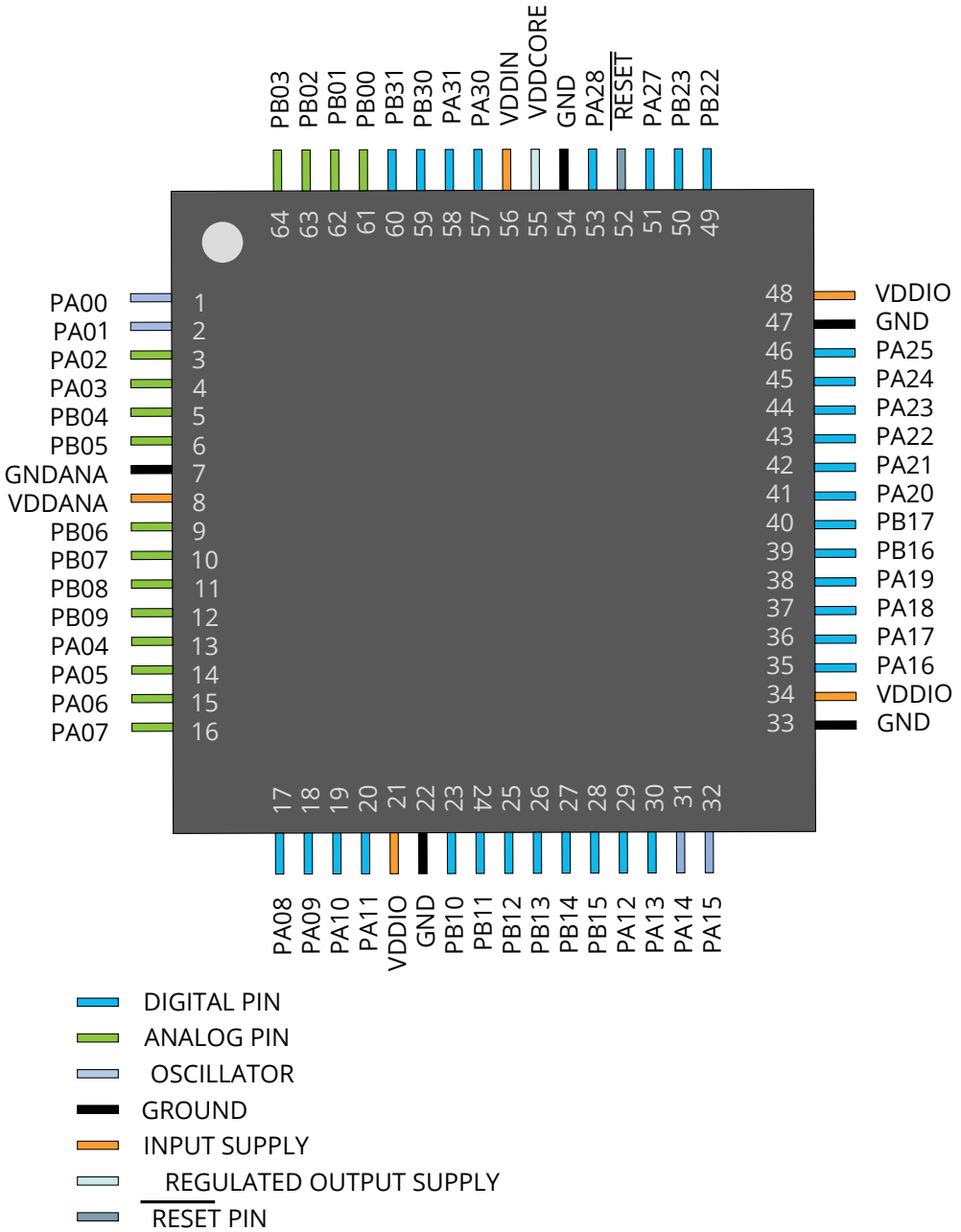




© 2026 Microchip Technology Inc. and its subsidiaries

## 4.3. SAM C21J / SAM C20J

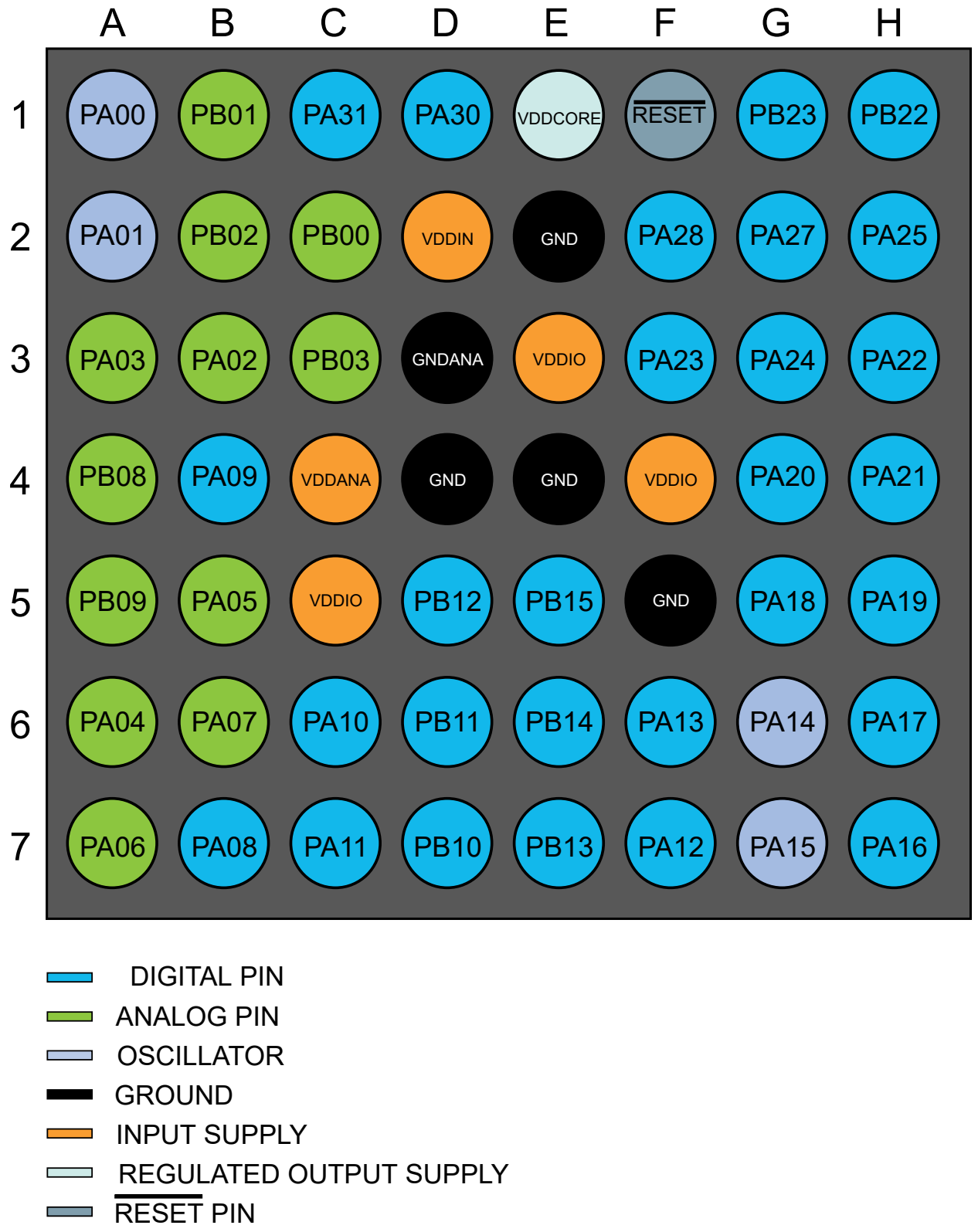
### 4.3.1. VQFN64/TQFP64





© 2026 Microchip Technology Inc. and its subsidiaries

### 4.3.2. WLCSP56

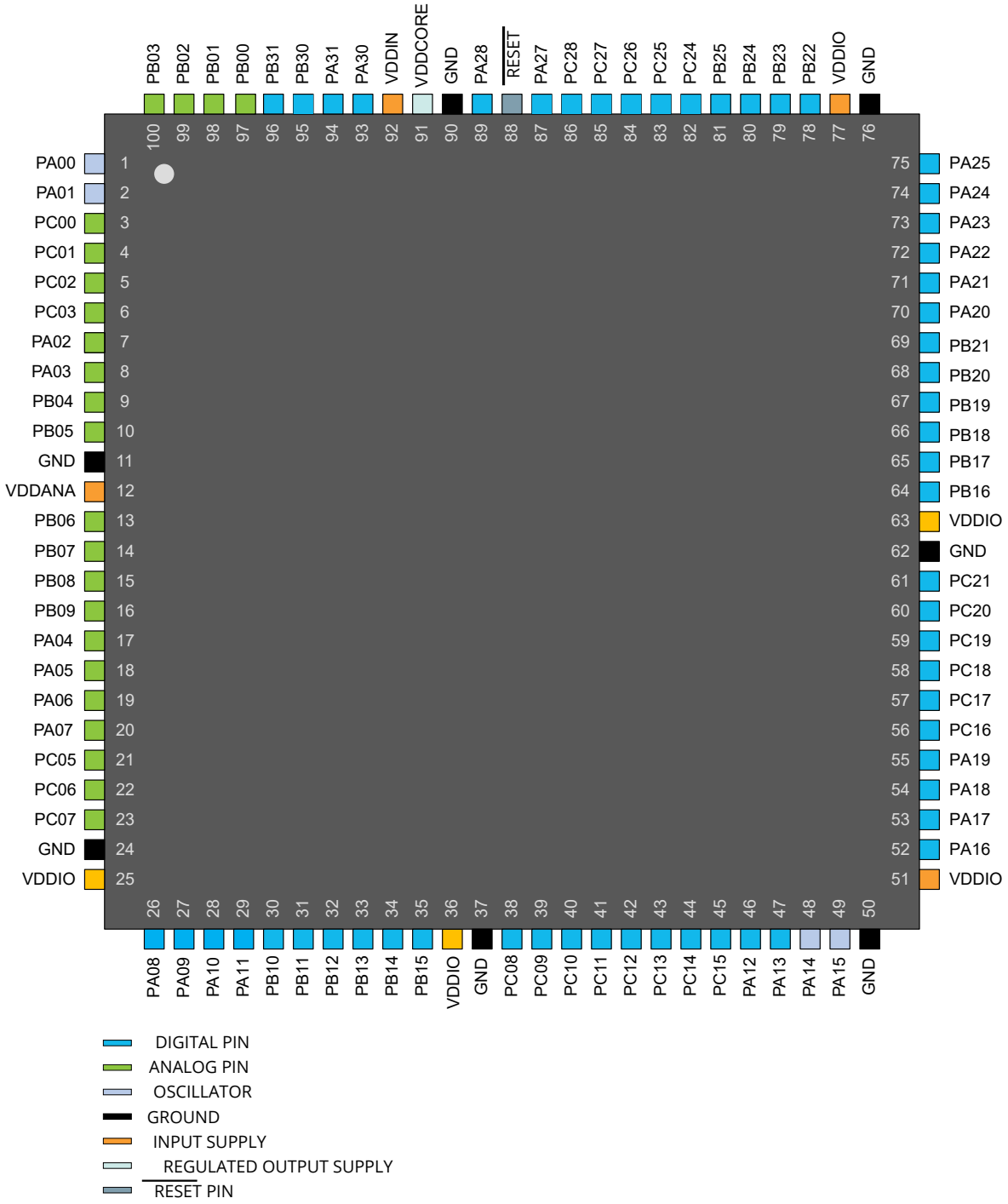




© 2026 Microchip Technology Inc. and its subsidiaries

## 4.4. SAM C21N / SAM C20N

### 4.4.1. TQFP100





© 2026 Microchip Technology Inc. and its subsidiaries

# 5. Signal Descriptions List

The following tables provide the details on signal names classified by peripheral.

**Table 5-1.** Signal Descriptions List - SAM C20/C21

| Signal Name | Function | Type | Active Level |
|---|---|---|---|
| **Analog Comparators - AC** | | | |
| AIN[7:0] | AC Analog Inputs | Analog | |
| CMP[2:0] | AC Comparator Outputs | Digital | |
| **Analog Digital Converter - ADCx** | | | |
| AIN[19:0] | ADC Analog Inputs | Analog | |
| VREFA | ADC Voltage External Reference A | Analog | |
| **Digital Analog Converter - DAC** | | | |
| VOUT | DAC Voltage output | Analog | |
| VREFA | DAC Voltage External Reference A | Analog | |
| **Sigma-Delta Analog Digital Converter - SDADC** | | | |
| AINN[2:0] | SDADC Analog Negative Inputs | Analog | |
| AINP[2:0] | SDADC Analog Positive Inputs | Analog | |
| VREFB | SDADC Voltage External Reference B | Analog | |
| **External Interrupt Controller - EIC** | | | |
| EXTINT[15:0] | External Interrupts inputs | Digital | |
| NMI | External Non-Maskable Interrupt input | Digital | |
| **Generic Clock Generator - GCLK** | | | |
| GCLK_IO[7:0] | Generic Clock (source clock inputs or generic clock generator output) | Digital | |
| **Custom Control Logic - CCL** | | | |
| IN[11:0] | Logic Inputs | Digital | |
| OUT[3:0] | Logic Outputs | Digital | |
| **Power Manager - PM** | | | |
| $\overline{\text{RESET}}$ | Reset input | Digital | Low |
| **Serial Communication Interface - SERCOMx** | | | |
| PAD[3:0] | SERCOM Inputs/Outputs Pads | Digital | |
| **Oscillators Control - OSCCTRL** | | | |
| XIN | Crystal or external clock Input | Analog/Digital | |
| XOUT | Crystal Output | Analog | |
| **32.768 kHz Oscillators Control - OSC32KCTRL** | | | |
| XIN32 | 32.768 kHz Crystal or external clock Input | Analog/Digital | |
| XOUT32 | 32.768 kHz Crystal Output | Analog | |
| **Timer Counter - TCx** | | | |
| WO[1:0] | Waveform/PWM Outputs/ Capture Inputs | Digital | |
| **Timer Counter - TCCx** | | | |
| WO[7:0] | Waveform/PWM Outputs | Digital | |
| **Peripheral Touch Controller - PTC** | | | |
| X[15:0] | PTC Input | Analog | |
| Y[15:0] | PTC Input | Analog | |
| **General Purpose I/O - PORT** | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 5-1.** Signal Descriptions List - SAM C20/C21 (continued)

| Signal Name | Function | Type | Active Level |
|---|---|---|---|
| PA25 - PA00 | Parallel I/O Controller I/O Port A | Digital | |
| PA28 - PA27 | Parallel I/O Controller I/O Port A | Digital | |
| PA31 - PA30 | Parallel I/O Controller I/O Port A | Digital | |
| PB17 - PB00 | Parallel I/O Controller I/O Port B | Digital | |
| PB21 - PB19 | Parallel I/O Controller I/O Port B | Digital | |
| PB25 - PB22 | Parallel I/O Controller I/O Port B | Digital | |
| PB31 - PB30 | Parallel I/O Controller I/O Port B | Digital | |
| PC03 - PC-00 | Parallel I/O Controller I/O Port C | Digital | |
| PC21 - PC05 | Parallel I/O Controller I/O Port C | Digital | |
| PC28 - PC24 | Parallel I/O Controller I/O Port C | Digital | |
| **Controller Area Network - CAN** | | | |
| TX | CAN Transmit Line | Digital | |
| RX | CAN Receive Line | Digital | |


© 2026 Microchip Technology Inc. and its subsidiaries

# 6. I/O Multiplexing and Considerations

## 6.1. Multiplexed Signals

Each pin is by default controlled by the PORT as a general purpose I/O, and alternatively it can be assigned to one of the peripheral functions, such as A, B, C, D, E, F, G, H, or I, as shown in the first row of the following table. To enable a peripheral function on a pin, the Peripheral Multiplexer Enable bit in the Pin Configuration register corresponding to that pin (PINCFGn.PMUXEN, n = 0-31) in the PORT must be written to one. The selection of peripheral function A to H is done by writing to the Peripheral Multiplexing Odd and Even bits in the Peripheral Multiplexing register (PMUXn.PMUXE/O) in the PORT.

**Table 6-1.** PORT Function Multiplexing for SAM C21 N

| Pin | I/O Pin | Supply | A | B[1,2] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | EIC | REF | ADC0 | ADC1 | SDADC | AC | PTC | DAC | SERCOM | SERCOM-ALT | TC | TCC | COM | AC/GCLK | CCL |
| 1 | PA00 | VDDANA | EXTINT[0] | | | | | | | | | SERCOM1/PAD[0] | TC2/WO[0] | | | CMP[2] | |
| 2 | PA01 | VDDANA | EXTINT[1] | | | | | | | | | SERCOM1/PAD[1] | TC2/WO[1] | | | CMP[3] | |
| 3 | PC00 | VDDANA | EXTINT[8] | | AIN[8] | | | | | | | | | | | | |
| 4 | PC01 | VDDANA | EXTINT[9] | | AIN[9] | | | | | | | | | | | | |
| 5 | PC02 | VDDANA | EXTINT[10] | | AIN[10] | | | | | | | | | | | | |
| 6 | PC03 | VDDIO | EXTINT[11] | | AIN[11] | | | | | | SERCOM7/PAD[0] | | | TCC2/WO[0] | | | |
| 7 | PA02 | VDDANA | EXTINT[2] | | AIN[0] | | | AIN[4] | Y[0] | VOUT | | | | | | | |
| 8 | PA03 | VDDANA | EXTINT[3] | ADC-DAC/VREFA | AIN[1] | | | | Y[1] | | | | | | | | |
| 9 | PB04 | VDDANA | EXTINT[4] | | | AIN[6] | | AIN[5] | Y[10] | | | | | | | | |
| 10 | PB05 | VDDANA | EXTINT[5] | | | AIN[7] | | AIN[6] | Y[11] | | | | | | | | |
| 13 | PB06 | VDDIO | EXTINT[6] | | | AIN[8] | AINN[2] | AIN[7] | Y[12] | | SERCOM7/PAD[1] | | | | | | CCL2/IN[6] |
| 14 | PB07 | VDDIO | EXTINT[7] | | | AIN[9] | AINP[2] | | Y[13] | | SERCOM7/PAD[3] | SERCOM7/PAD[2] | | | | | CCL2/IN[7] |
| 15 | PB08 | VDDIO | EXTINT[8] | | AIN[2] | AIN[4] | AINN[1] | | Y[14] | | SERCOM7/PAD[2] | SERCOM7/PAD[3] | TC4/WO[0] | | | | CCL2/IN[8] |
| 16 | PB09 | VDDANA | EXTINT[9] | | AIN[3] | AIN[5] | AINP[1] | | Y[15] | | | SERCOM4/PAD[1] | TC4/WO[1] | | | | CCL2/OUT[2] |
| 17 | PA04 | VDDANA | EXTINT[4] | SDADC/VREFB | AIN[4] | | | AIN[0] | Y[2] | | | SERCOM0/PAD[0] | TC0/WO[0] | | | | CCL0/IN[0] |
| 18 | PA05 | VDDANA | EXTINT[5] | | AIN[5] | | | AIN[1] | Y[3] | | | SERCOM0/PAD[1] | TC0/WO[1] | | | | CCL0/IN[1] |
| 19 | PA06 | VDDANA | EXTINT[6] | | AIN[6] | | AINN[0] | AIN[2] | Y[4] | | | SERCOM0/PAD[2] | TC1/WO[0] | | | | CCL0/IN[2] |
| 20 | PA07 | VDDANA | EXTINT[7] | | AIN[7] | | AINP[0] | AIN[3] | Y[5] | | | SERCOM0/PAD[3] | TC1/WO[1] | | | | CCL0/OUT[0] |
| 21 | PC05 | VDDANA | EXTINT[13] | | | | | | | | SERCOM6/PAD[3] | | | TCC2/WO[1] | | | |
| 22 | PC06 | VDDANA | EXTINT[14] | | | | | | | | SERCOM6/PAD[0] | | | | | | |
| 23 | PC07 | VDDANA | EXTINT[15] | | | | | | | | SERCOM6/PAD[1] | | | | | | |
| 26 | PA08 | VDDIO | NMI | | | AIN[10] | | | X[0]/Y[16] | | SERCOM0/PAD[0] | SERCOM2/PAD[0] | TC0/WO[0] | TCC0/WO[0] | | | CCL1/IN[3] |
| 27 | PA09 | VDDIO | EXTINT[9] | | | AIN[11] | | | X[1]/Y[17] | | SERCOM0/PAD[1] | SERCOM2/PAD[1] | TC0/WO[1] | TCC0/WO[1] | | | CCL1/IN[4] |
| 28 | PA10 | VDDIO | EXTINT[10] | | | | | | X[2]/Y[18] | | SERCOM0/PAD[2] | SERCOM2/PAD[2] | TC1/WO[0] | TCC0/WO[2] | | GCLK_IO[4] | CCL1/IN[5] |
| 29 | PA11 | VDDIO | EXTINT[11] | | | | | | X[3]/Y[19] | | SERCOM0/PAD[3] | SERCOM2/PAD[3] | TC1/WO[1] | TCC0/WO[3] | | GCLK_IO[5] | CCL1/OUT[1] |
| 30 | PB10 | VDDIO | EXTINT[10] | | | | | | | | | SERCOM4/PAD[2] | TC5/WO[0] | TCC0_WO4 | | GCLK_IO[4] | CCL1/IN[5] |
| 31 | PB11 | VDDIO | EXTINT[11] | | | | | | | | | SERCOM4/PAD[3] | TC5/WO[1] | TCC0_WO5 | | GCLK_IO[5] | CCL1/OUT[1] |
| 32 | PB12 | VDDIO | EXTINT[12] | | | | | | X[12]/Y[28] | | SERCOM4/PAD[0] | | TC4/WO[0] | TCC0_WO6 | CAN1/TX | GCLK_IO[6] | |
| 33 | PB13 | VDDIO | EXTINT[13] | | | | | | X[13]/Y[29] | | SERCOM4/PAD[1] | | TC4/WO[1] | TCC0_WO7 | CAN1/RX | GCLK_IO[7] | |
| 34 | PB14 | VDDIO | EXTINT[14] | | | | | | X[14]/Y[30] | | SERCOM4/PAD[2] | | TC5/WO[0] | | CAN1/TX | GCLK_IO[0] | CCL3/IN[9] |
| 35 | PB15 | VDDIO | EXTINT[15] | | | | | | X[15]/Y[31] | | SERCOM4/PAD[3] | | TC5/WO[1] | | CAN1/RX | GCLK_IO[1] | CCL3/IN[10] |
| 38 | PC08 | VDDIO | EXTINT[0] | | | | | | | | SERCOM6/PAD[0] | SERCOM7/PAD[0] | | | | | |
| 39 | PC09 | VDDIO | EXTINT[1] | | | | | | | | SERCOM6/PAD[1] | SERCOM7/PAD[1] | | | | | |
| 40 | PC10 | VDDIO | EXTINT[2] | | | | | | | | SERCOM6/PAD[2] | SERCOM7/PAD[2] | | | | | |
| 41 | PC11 | VDDIO | EXTINT[3] | | | | | | | | SERCOM6/PAD[3] | SERCOM7/PAD[3] | | | | | |
| 42 | PC12 | VDDIO | EXTINT[4] | | | | | | | | SERCOM7/PAD[0] | | | | | | |
| 43 | PC13 | VDDIO | EXTINT[5] | | | | | | | | SERCOM7/PAD[1] | | | | | | |
| 44 | PC14 | VDDIO | EXTINT[6] | | | | | | | | SERCOM7/PAD[2] | | | | | | |
| 45 | PC15 | VDDIO | EXTINT[7] | | | | | | | | SERCOM7/PAD[3] | | | | | | |
| 46 | PA12 | VDDIO | EXTINT[12] | | | | | | | | SERCOM2/PAD[0] | SERCOM4/PAD[0] | TC2/WO[0] | TCC0_WO6 | | CMP[0] | |
| 47 | PA13 | VDDIO | EXTINT[13] | | | | | | | | SERCOM2/PAD[1] | SERCOM4/PAD[1] | TC2/WO[1] | TCC0_WO7 | | CMP[1] | |
| 48 | PA14 | VDDIO | EXTINT[14] | | | | | | | | SERCOM2/PAD[2] | SERCOM4/PAD[2] | TC3/WO[0] | | | GCLK_IO[0] | |
| 49 | PA15 | VDDIO | EXTINT[15] | | | | | | | | SERCOM2/PAD[3] | SERCOM4/PAD[3] | TC3/WO[1] | | | GCLK_IO[1] | |
| 52 | PA16 | VDDIO | EXTINT[0] | | | | | | X[4]/Y[20] | | SERCOM1/PAD[0] | SERCOM3/PAD[0] | TC2/WO[0] | TCC1/WO[0] | | GCLK_IO[2] | CCL0/IN[0] |
| 53 | PA17 | VDDIO | EXTINT[1] | | | | | | X[5]/Y[21] | | SERCOM1/PAD[1] | SERCOM3/PAD[1] | TC2/WO[1] | TCC1/WO[1] | | GCLK_IO[3] | CCL0/IN[1] |
| 54 | PA18 | VDDIO | EXTINT[2] | | | | | | X[6]/Y[22] | | SERCOM1/PAD[2] | SERCOM3/PAD[2] | TC3/WO[0] | TCC1/WO[2] | | CMP[0] | CCL0/IN[2] |
| 55 | PA19 | VDDIO | EXTINT[3] | | | | | | X[7]/Y[23] | | SERCOM1/PAD[3] | SERCOM3/PAD[3] | TC3/WO[1] | TCC1/WO[3] | | CMP[1] | CCL0/OUT[0] |
| 56 | PC16 | VDDIO | EXTINT[8] | | | | | | | | SERCOM6/PAD[0] | | | | | | |
| 57 | PC17 | VDDIO | EXTINT[9] | | | | | | | | SERCOM6/PAD[1] | | | | | | |
| 58 | PC18 | VDDIO | EXTINT[10] | | | | | | | | SERCOM6/PAD[2] | | | | | | |
| 59 | PC19 | VDDIO | EXTINT[11] | | | | | | | | SERCOM6/PAD[3] | | | | | | |
| 60 | PC20 | VDDIO | EXTINT[12] | | | | | | | | | | | | | | CCL3/IN[9] |
| 61 | PC21 | VDDIO | EXTINT[13] | | | | | | | | | | | | | | CCL3/IN[10] |
| 64 | PB16 | VDDIO | EXTINT[0] | | | | | | | | SERCOM5/PAD[0] | | TC6/WO[0] | | | GCLK_IO[2] | CCL3/IN[11] |
| 65 | PB17 | VDDIO | EXTINT[1] | | | | | | | | SERCOM5/PAD[1] | | TC6/WO[1] | | | GCLK_IO[3] | CCL3/OUT[3] |
| 66 | PB18 | VDDIO | EXTINT[2] | | | | | | | | SERCOM5/PAD[2] | SERCOM3/PAD[2] | | | | GCLK_IO[4] | |
| 67 | PB19 | VDDIO | EXTINT[3] | | | | | | | | SERCOM5/PAD[3] | SERCOM3/PAD[3] | | | | GCLK_IO[5] | |
| 68 | PB20 | VDDIO | EXTINT[4] | | | | | | | | SERCOM3/PAD[0] | SERCOM2/PAD[0] | | | | GCLK_IO[6] | |
| 69 | PB21 | VDDIO | EXTINT[5] | | | | | | | | SERCOM3/PAD[1] | SERCOM2/PAD[1] | | | | GCLK_IO[7] | |
| 70 | PA20 | VDDIO | EXTINT[4] | | | | | | X[8]/Y[24] | | SERCOM5/PAD[2] | SERCOM3/PAD[2] | TC7/WO[0] | TCC2/WO[0] | | GCLK_IO[4] | |
| 71 | PA21 | VDDIO | EXTINT[5] | | | | | | X[9]/Y[25] | | SERCOM5/PAD[3] | SERCOM3/PAD[3] | TC7/WO[1] | TCC2/WO[1] | | GCLK_IO[5] | |
| 72 | PA22 | VDDIO | EXTINT[6] | | | | | | X[10]/Y[26] | | SERCOM3/PAD[0] | SERCOM5/PAD[0] | TC4/WO[0] | TCC1/WO[0] | CAN0/TX | GCLK_IO[6] | CCL2/IN[6] |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-1.** PORT Function Multiplexing for SAM C21 N (continued)

| Pin | I/O Pin | Supply | A | B[1,2] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | EIC | REF | ADC0 | ADC1 | SDADC | AC | PTC | DAC | SERCOM | SERCOM-ALT | TC | TCC | COM | AC/GCLK | CCL |
| 73 | PA23 | VDDIO | EXTINT[7] | | | | | | X[11]/Y[27] | | SERCOM3/PAD[1] | SERCOM5/PAD[1] | TC4/WO[1] | TCC1/WO[1] | CAN0/RX | GCLK_IO[7] | CCL2/IN[7] |
| 74 | PA24 | VDDIO | EXTINT[12] | | | | | | | | SERCOM3/PAD[2] | SERCOM5/PAD[2] | TC5/WO[0] | TCC2/WO[0] | CAN0/TX | CMP[2] | CCL2/IN[8] |
| 75 | PA25 | VDDIO | EXTINT[13] | | | | | | | | SERCOM3/PAD[3] | SERCOM5/PAD[3] | TC5/WO[1] | TCC2/WO[1] | CAN0/RX | CMP[3] | CCL2/OUT[2] |
| 78 | PB22 | VDDIO | EXTINT[6] | | | | | | | | SERCOM0/PAD[2] | SERCOM5/PAD[2] | TC7/WO[0] | TCC1/WO[2] | | GCLK_IO[0] | CCL0/IN[0] |
| 79 | PB23 | VDDIO | EXTINT[7] | | | | | | | | SERCOM0/PAD[3] | SERCOM5/PAD[3] | TC7/WO[1] | TCC1/WO[3] | | GCLK_IO[1] | CCL0/OUT[0] |
| 80 | PB24 | VDDIO | EXTINT[8] | | | | | | | | SERCOM0/PAD[0] | SERCOM4/PAD[0] | | | | CMP[0] | |
| 81 | PB25 | VDDIO | EXTINT[9] | | | | | | | | SERCOM0/PAD[1] | SERCOM4/PAD[1] | | | | CMP[1] | |
| 82 | PC24 | VDDIO | EXTINT[0] | | | | | | | | SERCOM0/PAD[2] | SERCOM4/PAD[2] | | | | | |
| 83 | PC25 | VDDIO | EXTINT[1] | | | | | | | | SERCOM0/PAD[3] | SERCOM4/PAD[3] | | | | | |
| 84 | PC26 | VDDIO | EXTINT[2] | | | | | | | | | | | | | | |
| 85 | PC27 | VDDIO | EXTINT[3] | | | | | | | | | SERCOM1/PAD[0] | | | | | CCL1/IN[4] |
| 86 | PC28 | VDDIO | EXTINT[4] | | | | | | | | | SERCOM1/PAD[1] | | | | | CCL1/IN[5] |
| 87 | PA27 | VDDIN | EXTINT[15] | | | | | | | | | | | | | GCLK_IO[0] | |
| 89 | PA28 | VDDIN | EXTINT[8] | | | | | | | | | | | | | GCLK_IO[0] | |
| 93 | PA30 | VDDIN | EXTINT[10] | | | | | | | | | SERCOM1/PAD[2] | TC1/WO[0] | | CORTEX_M0P/SWCLK | GCLK_IO[0] | CCL1/IN[3] |
| 94 | PA31 | VDDIN | EXTINT[11] | | | | | | | | | SERCOM1/PAD[3] | TC1/WO[1] | | CORTEX_M0P/SWDIO | | CCL1/OUT[1] |
| 95 | PB30 | VDDIN | EXTINT[14] | | | | | | | | SERCOM1/PAD[0] | SERCOM5/PAD[0] | TC0/WO[0] | | | CMP[2] | |
| 96 | PB31 | VDDIN | EXTINT[15] | | | | | | | | SERCOM1/PAD[1] | SERCOM5/PAD[1] | TC0/WO[1] | | | CMP[3] | |
| 97 | PB00 | VDDANA | EXTINT[0] | | | AIN[0] | | | Y[6] | | | SERCOM5/PAD[2] | TC7/WO[0] | | | | CCL0/IN[1] |
| 98 | PB01 | VDDANA | EXTINT[1] | | | AIN[1] | | | Y[7] | | | SERCOM5/PAD[3] | TC7/WO[1] | | | | CCL0/IN[2] |
| 99 | PB02 | VDDANA | EXTINT[2] | | | AIN[2] | | | Y[8] | | | SERCOM5/PAD[0] | TC6/WO[0] | | | | CCL0/OUT[0] |
| 100 | PB03 | VDDANA | EXTINT[3] | | | AIN[3] | | | Y[9] | | | SERCOM5/PAD[1] | TC6/WO[1] | | | | |

**Note:**

1. All analog pin functions are on peripheral function B. Peripheral function B must be selected to disable the digital control of the pin.
2. Only some pins can be used in SERCOM $I^2C$ mode. For additional information, refer to SERCOM I2C Pins.

**Table 6-2.** PORT Function Multiplexing for SAM C21 E/G/J

| Pin[1] | | | I/O Pin | Supply | A | B[2,3] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SAM C21E | SAM C21G | SAM C21J | | | EIC | REF | ADC0 | ADC1 | AC | PTC | DAC | SDADC | SERCOM[2,3,4] | SERCOM-ALT[4] | TC<br>TCC | TCC | COM | AC/GCLK | CCL |
| 1 | 1 | 1 | PA00 | VDDANA | EXTINT[0] | | | | | | | | | SERCOM1/PAD[0] | TCC2/WO[0] | | | CMP[2] | |
| 2 | 2 | 2 | PA01 | VDDANA | EXTINT[1] | | | | | | | | | SERCOM1/PAD[1] | TCC2/WO[1] | | | CMP[3] | |
| 3 | 3 | 3 | PA02 | VDDANA | EXTINT[2] | | AIN[0] | | AIN[4] | Y[0] | VOUT | | | | | | | | |
| 4 | 4 | 4 | PA03 | VDDANA | EXTINT[3] | ADC-DAC/VREFA | AIN[1] | | AIN[5] | Y[1] | | | | | | | | | |
| | | 5 | PB04 | VDDANA | EXTINT[4] | | | AIN[6] | | Y[10] | | | | | | | | | |
| | | 6 | PB05 | VDDANA | EXTINT[5] | | | AIN[7] | AIN[6] | Y[11] | | | | | | | | | |
| | | 9 | PB06 | VDDANA | EXTINT[6] | | | AIN[8] | AIN[7] | Y[12] | | AINN[2] | | | | | | | CCL2/IN[6] |
| | | 10 | PB07 | VDDANA | EXTINT[7] | | | AIN[9] | | Y[13] | | AINP[2] | | | | | | | CCL2/IN[7] |
| | 7 | 11 | PB08 | VDDANA | EXTINT[8] | | AIN[2] | AIN[4] | | Y[14] | | AINN[1] | | SERCOM4/PAD[0] | TC0/WO[0] | | | | CCL2/IN[8] |
| | 8 | 12 | PB09 | VDDANA | EXTINT[9] | | AIN[3] | AIN[5] | | Y[15] | | AINP[1] | | SERCOM4/PAD[1] | TC0WO[1] | | | | CCL2/OUT[2] |
| 5 | 9 | 13 | PA04 | VDDANA | EXTINT[4] | SDADC / VREFB | AIN[4] | | AIN[0] | Y[2] | | | | SERCOM0/PAD[0] | TCC0/WO[0] | | | | CCL0/IN[0] |
| 6 | 10 | 14 | PA05 | VDDANA | EXTINT[5] | | AIN[5] | | AIN[1] | Y[3] | | | | SERCOM0/PAD[1] | TCC0/WO[1] | | | | CCL0/IN[1] |
| 7 | 11 | 15 | PA06 | VDDANA | EXTINT[6] | | AIN[6] | | AIN[2] | Y[4] | | AINN[0] | | SERCOM0/PAD[2] | TCC1/WO[0] | | | | CCL0/IN[2] |
| 8 | 12 | 16 | PA07 | VDDANA | EXTINT[7] | | AIN[7] | | AIN[3] | Y[5] | | AINP[0] | | SERCOM0/PAD[3] | TCC1/WO[1] | | | | CCL0/OUT[0] |
| 11 | 13 | 17 | PA08 | VDDIO | NMI | | AIN[8] | AIN[10] | | X[0]/Y[16] | | | SERCOM0/PAD[0] | SERCOM2/PAD[0] | TCC0/WO[0] | TCC1/WO[2] | | | CCL1/IN[3] |
| 12 | 14 | 18 | PA09 | VDDIO | EXTINT[9] | | AIN[9] | AIN[11] | | X[1]/Y[17] | | | SERCOM0/PAD[1] | SERCOM2/PAD[1] | TCC0/WO[1] | TCC1/WO[3] | | | CCL1/IN[4] |
| 13 | 15 | 19 | PA10 | VDDIO | EXTINT[10] | | AIN[10] | | | X[2]/Y[18] | | | SERCOM0/PAD[2] | SERCOM2/PAD[2] | TCC1/WO[0] | TCC0/WO[2] | | GCLK_IO[4] | CCL1/IN[5] |
| 14 | 16 | 20 | PA11 | VDDIO | EXTINT[11] | | AIN[11] | | | X[3]/Y[19] | | | SERCOM0/PAD[3] | SERCOM2/PAD[3] | TCC1/WO[1] | TCC0/WO[3] | | GCLK_IO[5] | CCL1/OUT[1] |
| | 19 | 23 | PB10 | VDDIO | EXTINT[10] | | | | | | | | | SERCOM4/PAD[2] | TC1/WO[0] | TCC0/WO[4] | CAN1/TX | GCLK_IO[4] | CCL1/IN[5] |
| | 20 | 24 | PB11 | VDDIO | EXTINT[11] | | | | | | | | | SERCOM4/PAD[3] | TC1/WO[1] | TCC0/WO[5] | CAN1/RX | GCLK_IO[5] | CCL1/OUT[1] |
| | | 25 | PB12 | VDDIO | EXTINT[12] | | | | | X[12]/Y[28] | | | SERCOM4/PAD[0] | | TC0/WO[0] | TCC0/WO[6] | | GCLK_IO[6] | |
| | | 26 | PB13 | VDDIO | EXTINT[13] | | | | | X[13]/Y[29] | | | SERCOM4/PAD[1] | | TC0/WO[1] | TCC0/WO[7] | | GCLK_IO[7] | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-2.** PORT Function Multiplexing for SAM C21 E/G/J (continued)

| | Pin[1] | | I/O Pin | Supply | A | B[2,3] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SAM C21E | SAM C21G | SAM C21J | | | EIC | REF | ADC0 | ADC1 | AC | PTC | DAC | SDADC | SERCOM[2,3,4] | SERCOM-ALT[4] | TC<br>TCC | TCC | COM | AC/GCLK | CCL |
| | | 27 | PB14 | VDDIO | EXTINT[14] | | | | | X[14]/Y[30] | | | SERCOM4/PAD[2] | | TC1/WO[0] | | CAN1/TX | GCLK_IO[0] | CCL3/IN[9] |
| | | 28 | PB15 | VDDIO | EXTINT[15] | | | | | X[15]/Y[31] | | | SERCOM4/PAD[3] | | TC1/WO[1] | | CAN1/RX | GCLK_IO[1] | CCL3/IN[10] |
| | 21 | 29 | PA12 | VDDIO | EXTINT[12] | | | | | | | | SERCOM2/PAD[0] | SERCOM4/PAD[0] | TCC2/WO[0] | TCC0/WO[6] | | AC/CMP[0] | |
| | 22 | 30 | PA13 | VDDIO | EXTINT[13] | | | | | | | | SERCOM2/PAD[1] | SERCOM4/PAD[1] | TCC2/WO[1] | TCC0/WO[7] | | AC/CMP[1] | |
| 15 | 23 | 31 | PA14 | VDDIO | EXTINT[14] | | | | | | | | SERCOM2/PAD[2] | SERCOM4/PAD[2] | TC4/WO[0] | TCC0/WO[4] | | GCLK_IO[0] | |
| 16 | 24 | 32 | PA15 | VDDIO | EXTINT[15] | | | | | | | | SERCOM2/PAD[3] | SERCOM4/PAD[3] | TC4/WO[1] | TCC0/WO[5] | | GCLK_IO[1] | |
| 17 | 25 | 35 | PA16 | VDDIO | EXTINT[0] | | | | | X[4]/Y[20] | | | SERCOM1/PAD[0] | SERCOM3/PAD[0] | TCC2/WO[0] | TCC0/WO[6] | | GCLK_IO[2] | CCL0/IN[0] |
| 18 | 26 | 36 | PA17 | VDDIO | EXTINT[1] | | | | | X[5]/Y[21] | | | SERCOM1/PAD[1] | SERCOM3/PAD[1] | TCC2/WO[1] | TCC0/WO[7] | | GCLK_IO[3] | CCL0/IN[1] |
| 19 | 27 | 37 | PA18 | VDDIO | EXTINT[2] | | | | | X[6]/Y[22] | | | SERCOM1/PAD[2] | SERCOM3/PAD[2] | TC4/WO[0] | TCC0/WO[2] | | AC/CMP[0] | CCL0/IN[2] |
| 20 | 28 | 38 | PA19 | VDDIO | EXTINT[3] | | | | | X[7]/Y[23] | | | SERCOM1/PAD[3] | SERCOM3/PAD[3] | TC4/WO[1] | TCC0/WO[3] | | AC/CMP[1] | CCL0/OUT[0] |
| | | 39 | PB16 | VDDIO | EXTINT[0] | | | | | | | | SERCOM5/PAD[0] | | TC2/WO[0] | TCC0/WO[4] | | GCLK_IO[2] | CCL3/IN[11] |
| | | 40 | PB17 | VDDIO | EXTINT[1] | | | | | | | | SERCOM5/PAD[1] | | TC2/WO[1] | TCC0/WO[5] | | GCLK_IO[3] | CCL3/OUT[3] |
| | 29 | 41 | PA20 | VDDIO | EXTINT[4] | | | | | X[8]/Y[24] | | | SERCOM5/PAD[2] | SERCOM3/PAD[2] | TC3/WO[0] | TCC0/WO[6] | | GCLK_IO[4] | |
| | 30 | 42 | PA21 | VDDIO | EXTINT[5] | | | | | X[9]/Y[25] | | | SERCOM5/PAD[3] | SERCOM3/PAD[3] | TC3/WO[1] | TCC0/WO[7] | | GCLK_IO[5] | |
| 21 | 31 | 43 | PA22 | VDDIO | EXTINT[6] | | | | | X[10]/Y[26] | | | SERCOM3/PAD[0] | SERCOM5/PAD[0] | TC0/WO[0] | TCC0/WO[4] | | GCLK_IO[6] | CCL2/IN[6] |
| 22 | 32 | 44 | PA23 | VDDIO | EXTINT[7] | | | | | X[11]/Y[27] | | | SERCOM3/PAD[1] | SERCOM5/PAD[1] | TC0/WO[1] | TCC0/WO[5] | | GCLK_IO[7] | CCL2/IN[7] |
| 23 | 33 | 45 | PA24 | VDDIO | EXTINT[12] | | | | | | | | SERCOM3/PAD[2] | SERCOM5/PAD[2] | TC1/WO[0] | TCC1/WO[2] | CAN0/TX | AC/CMP[2] | CCL2/IN[8] |
| 24 | 34 | 46 | PA25 | VDDIO | EXTINT[13] | | | | | | | | SERCOM3/PAD[3] | SERCOM5/PAD[3] | TC1/WO[1] | TCC1/WO[3] | CAN0/RX | AC/CMP[3] | CCL2/OUT[2] |
| | 37 | 49 | PB22 | VDDIO | EXTINT[6] | | | | | | | | | SERCOM5/PAD[2] | TC3/WO[0] | | CAN0/TX | GCLK_IO[0] | CCL0/IN[0] |
| | 38 | 50 | PB23 | VDDIO | EXTINT[7] | | | | | | | | | SERCOM5/PAD[3] | TC3/WO[1] | | CAN0/RX | GCLK_IO[1] | CCL0/OUT[0] |
| 25 | 39 | 51 | PA27 | VDDIN | EXTINT[15] | | | | | | | | | | | | | GCLK_IO[0] | |
| 27 | 41 | 53 | PA28 | VDDIN | EXTINT[8] | | | | | | | | | | | | | GCLK_IO[0] | |
| 31 | 45 | 57 | PA30 | VDDIN | EXTINT[10] | | | | | | | | | SERCOM1/PAD[2] | TCC1/WO[0] | | CORTEX_M0P/SWCLK | GCLK_IO[0] | CCL1/IN[3] |
| 32 | 46 | 58 | PA31 | VDDIN | EXTINT[11] | | | | | | | | | SERCOM1/PAD[3] | TCC1/WO[1] | | CORTEX_M0P/SWDIO | | CCL1/OUT[1] |
| | | 59 | PB30 | VDDIN | EXTINT[14] | | | | | | | | | SERCOM5/PAD[0] | TCC0/WO[0] | TCC1/WO[2] | | AC/CMP[2] | |
| | | 60 | PB31 | VDDIN | EXTINT[15] | | | | | | | | | SERCOM5/PAD[1] | TCC0/WO[1] | TCC1/WO[3] | | AC/CMP[3] | |
| | | 61 | PB00 | VDDANA | EXTINT[0] | | | AIN[0] | | Y[6] | | | | SERCOM5/PAD[2] | TC3/WO[0] | | | | CCL0/IN[1] |
| | | 62 | PB01 | VDDANA | EXTINT[1] | | | AIN[1] | | Y[7] | | | | SERCOM5/PAD[3] | TC3/WO[1] | | | | CCL0/IN[2] |
| | 47 | 63 | PB02 | VDDANA | EXTINT[2] | | | AIN[2] | | Y[8] | | | | SERCOM5/PAD[0] | TC2/WO[0] | | | | CCL0/OUT[0] |
| | 48 | 64 | PB03 | VDDANA | EXTINT[3] | | | AIN[3] | | Y[9] | | | | SERCOM5/PAD[1] | TC2/WO[1] | | | | |

**Note:**

1. Use the SAM C21J pinout muxing for the WLCSP56 package.
2. All analog pin functions are on peripheral function B. Peripheral function B must be selected to disable the digital control of the pin.
3. Only some pins can be used in SERCOM $I^2C$ mode. For additional information, refer to SERCOM I2C Pins.
4. SERCOM4 and SERCOM5 are not supported on SAM C21E.

**Table 6-3.** PORT Function Multiplexing for SAM C20 N

| Pin | I/O Pin | Supply | A | B[1,2] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | EIC | REF | ADC0 | ADC1 | SDADC | AC | PTC | DAC | SERCOM | SERCOM-ALT | TC | TCC | COM | AC/GCLK | CCL |
| 1 | PA00 | VDDANA | EXTINT[0] | | | | | | | | | SERCOM1/PAD[0] | TC2/WO[0] | | | CMP[2] | |
| 2 | PA01 | VDDANA | EXTINT[1] | | | | | | | | | SERCOM1/PAD[1] | TC2/WO[1] | | | CMP[3] | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-3.** PORT Function Multiplexing for SAM C20 N (continued)

| Pin | I/O Pin | Supply | A | B[1,2] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | EIC | REF | ADC0 | ADC1 | SDADC | AC | PTC | DAC | SERCOM | SERCOM-ALT | TC | TCC | COM | AC/GCLK | CCL |
| 3 | PC00 | VDDANA | EXTINT[8] | | AIN[8] | | | | | | | | | | | | |
| 4 | PC01 | VDDANA | EXTINT[9] | | AIN[9] | | | | | | | | | | | | |
| 5 | PC02 | VDDANA | EXTINT[10] | | AIN[10] | | | | | | | | | | | | |
| 6 | PC03 | VDDIO | EXTINT[11] | | AIN[11] | | | | | | SERCOM7/PAD[0] | | | TCC2/WO[0] | | | |
| 7 | PA02 | VDDANA | EXTINT[2] | | AIN[0] | | | AIN[4] | Y[0] | VOUT | | | | | | | |
| 8 | PA03 | VDDANA | EXTINT[3] | ADC-DAC/VREFA | AIN[1] | | | | Y[1] | | | | | | | | |
| 9 | PB04 | VDDANA | EXTINT[4] | | | AIN[6] | | AIN[5] | Y[10] | | | | | | | | |
| 10 | PB05 | VDDANA | EXTINT[5] | | | AIN[7] | | AIN[6] | Y[11] | | | | | | | | |
| 13 | PB06 | VDDIO | EXTINT[6] | | | AIN[8] | INN[2] | AIN[7] | Y[12] | | SERCOM7/PAD[1] | | | | | | CCL2/IN[6] |
| 14 | PB07 | VDDIO | EXTINT[7] | | | AIN[9] | INP[2] | | Y[13] | | SERCOM7/PAD[3] | SERCOM7/PAD[2] | | | | | CCL2/IN[7] |
| 15 | PB08 | VDDIO | EXTINT[8] | | AIN[2] | AIN[4] | INN[1] | | Y[14] | | SERCOM7/PAD[2] | SERCOM7/PAD[3] | TC4/WO[0] | | | | CCL2/IN[8] |
| 16 | PB09 | VDDANA | EXTINT[9] | | AIN[3] | AIN[5] | INP[1] | | Y[15] | | | SERCOM4/PAD[1] | TC4/WO[1] | | | | CCL2/OUT[2] |
| 17 | PA04 | VDDANA | EXTINT[4] | | AIN[4] | | | AIN[0] | Y[2] | | | SERCOM0/PAD[0] | TC0/WO[0] | | | | CCL0/IN[0] |
| 18 | PA05 | VDDANA | EXTINT[5] | | AIN[5] | | | AIN[1] | Y[3] | | | SERCOM0/PAD[1] | TC0/WO[1] | | | | CCL0/IN[1] |
| 19 | PA06 | VDDANA | EXTINT[6] | | AIN[6] | | INN[0] | AIN[2] | Y[4] | | | SERCOM0/PAD[2] | TC1/WO[0] | | | | CCL0/IN[2] |
| 20 | PA07 | VDDANA | EXTINT[7] | | AIN[7] | | INP[0] | AIN[3] | Y[5] | | | SERCOM0/PAD[3] | TC1/WO[1] | | | | CCL0/OUT[0] |
| 21 | PC05 | VDDANA | EXTINT[13] | | | | | | | | SERCOM6/PAD[3] | | | TCC2/WO[1] | | | |
| 22 | PC06 | VDDANA | EXTINT[14] | | | | | | | | SERCOM6/PAD[0] | | | | | | |
| 23 | PC07 | VDDANA | EXTINT[15] | | | | | | | | SERCOM6/PAD[1] | | | | | | |
| 26 | PA08 | VDDIO | NMI | | | AIN[10] | | | X[0]/Y[16] | | SERCOM0/PAD[0] | SERCOM2/PAD[0] | TC0/WO[0] | TCC0/WO[0] | | | CCL1/IN[3] |
| 27 | PA09 | VDDIO | EXTINT[9] | | | AIN[11] | | | X[1]/Y[17] | | SERCOM0/PAD[1] | SERCOM2/PAD[1] | TC0/WO[1] | TCC0/WO[1] | | | CCL1/IN[4] |
| 28 | PA10 | VDDIO | EXTINT[10] | | | | | | X[2]/Y[18] | | SERCOM0/PAD[2] | SERCOM2/PAD[2] | TC1/WO[0] | TCC0/WO[2] | | GCLK_IO[4] | CCL1/IN[5] |
| 29 | PA11 | VDDIO | EXTINT[11] | | | | | | X[3]/Y[19] | | SERCOM0/PAD[3] | SERCOM2/PAD[3] | TC1/WO[1] | TCC0/WO[3] | | GCLK_IO[5] | CCL1/OUT[1] |
| 30 | PB10 | VDDIO | EXTINT[10] | | | | | | | | | SERCOM4/PAD[2] | TC5/WO[0] | TCC0_WO4 | | GCLK_IO[4] | CCL1/IN[5] |
| 31 | PB11 | VDDIO | EXTINT[11] | | | | | | | | | SERCOM4/PAD[3] | TC5/WO[1] | TCC0_WO5 | | GCLK_IO[5] | CCL1/OUT[1] |
| 32 | PB12 | VDDIO | EXTINT[12] | | | | | | X[12]/Y[28] | | SERCOM4/PAD[0] | | TC4/WO[0] | TCC0_WO6 | | GCLK_IO[6] | |
| 33 | PB13 | VDDIO | EXTINT[13] | | | | | | X[13]/Y[29] | | SERCOM4/PAD[1] | | TC4/WO[1] | TCC0_WO7 | | GCLK_IO[7] | |
| 34 | PB14 | VDDIO | EXTINT[14] | | | | | | X[14]/Y[30] | | SERCOM4/PAD[2] | | TC5/WO[0] | | | GCLK_IO[0] | CCL3/IN[9] |
| 35 | PB15 | VDDIO | EXTINT[15] | | | | | | X[15]/Y[31] | | SERCOM4/PAD[3] | | TC5/WO[1] | | | GCLK_IO[1] | CCL3/IN[10] |
| 38 | PC08 | VDDIO | EXTINT[0] | | | | | | | | SERCOM6/PAD[0] | SERCOM7/PAD[0] | | | | | |
| 39 | PC09 | VDDIO | EXTINT[1] | | | | | | | | SERCOM6/PAD[1] | SERCOM7/PAD[1] | | | | | |
| 40 | PC10 | VDDIO | EXTINT[2] | | | | | | | | SERCOM6/PAD[2] | SERCOM7/PAD[2] | | | | | |
| 41 | PC11 | VDDIO | EXTINT[3] | | | | | | | | SERCOM6/PAD[3] | SERCOM7/PAD[3] | | | | | |
| 42 | PC12 | VDDIO | EXTINT[4] | | | | | | | | SERCOM7/PAD[0] | | | | | | |
| 43 | PC13 | VDDIO | EXTINT[5] | | | | | | | | SERCOM7/PAD[1] | | | | | | |
| 44 | PC14 | VDDIO | EXTINT[6] | | | | | | | | SERCOM7/PAD[2] | | | | | | |
| 45 | PC15 | VDDIO | EXTINT[7] | | | | | | | | SERCOM7/PAD[3] | | | | | | |
| 46 | PA12 | VDDIO | EXTINT[12] | | | | | | | | SERCOM2/PAD[0] | SERCOM4/PAD[0] | TC2/WO[0] | TCC0_WO6 | | CMP[0] | |
| 47 | PA13 | VDDIO | EXTINT[13] | | | | | | | | SERCOM2/PAD[1] | SERCOM4/PAD[1] | TC2/WO[1] | TCC0_WO7 | | CMP[1] | |
| 48 | PA14 | VDDIO | EXTINT[14] | | | | | | | | SERCOM2/PAD[2] | SERCOM4/PAD[2] | TC3/WO[0] | | | GCLK_IO[0] | |
| 49 | PA15 | VDDIO | EXTINT[15] | | | | | | | | SERCOM2/PAD[3] | SERCOM4/PAD[3] | TC3/WO[1] | | | GCLK_IO[1] | |
| 52 | PA16 | VDDIO | EXTINT[0] | | | | | | X[4]/Y[20] | | SERCOM1/PAD[0] | SERCOM3/PAD[0] | TC2/WO[0] | TCC1/WO[0] | | GCLK_IO[2] | CCL0/IN[0] |
| 53 | PA17 | VDDIO | EXTINT[1] | | | | | | X[5]/Y[21] | | SERCOM1/PAD[1] | SERCOM3/PAD[1] | TC2/WO[1] | TCC1/WO[1] | | GCLK_IO[3] | CCL0/IN[1] |
| 54 | PA18 | VDDIO | EXTINT[2] | | | | | | X[6]/Y[22] | | SERCOM1/PAD[2] | SERCOM3/PAD[2] | TC3/WO[0] | TCC1/WO[2] | | CMP[0] | CCL0/IN[2] |
| 55 | PA19 | VDDIO | EXTINT[3] | | | | | | X[7]/Y[23] | | SERCOM1/PAD[3] | SERCOM3/PAD[3] | TC3/WO[1] | TCC1/WO[3] | | CMP[1] | CCL0/OUT[0] |
| 56 | PC16 | VDDIO | EXTINT[8] | | | | | | | | SERCOM6/PAD[0] | | | | | | |
| 57 | PC17 | VDDIO | EXTINT[9] | | | | | | | | SERCOM6/PAD[1] | | | | | | |
| 58 | PC18 | VDDIO | EXTINT[10] | | | | | | | | SERCOM6/PAD[2] | | | | | | |
| 59 | PC19 | VDDIO | EXTINT[11] | | | | | | | | SERCOM6/PAD[3] | | | | | | |
| 60 | PC20 | VDDIO | EXTINT[12] | | | | | | | | | | | | | | CCL3/IN[9] |
| 61 | PC21 | VDDIO | EXTINT[13] | | | | | | | | | | | | | | CCL3/IN[10] |
| 64 | PB16 | VDDIO | EXTINT[0] | | | | | | | | SERCOM5/PAD[0] | | TC6/WO[0] | | | GCLK_IO[2] | CCL3/IN[11] |
| 65 | PB17 | VDDIO | EXTINT[1] | | | | | | | | SERCOM5/PAD[1] | | TC6/WO[1] | | | GCLK_IO[3] | CCL3/OUT[3] |
| 66 | PB18 | VDDIO | EXTINT[2] | | | | | | | | SERCOM5/PAD[2] | SERCOM3/PAD[2] | | | | GCLK_IO[4] | |
| 67 | PB19 | VDDIO | EXTINT[3] | | | | | | | | SERCOM5/PAD[3] | SERCOM3/PAD[3] | | | | GCLK_IO[5] | |
| 68 | PB20 | VDDIO | EXTINT[4] | | | | | | | | SERCOM3/PAD[0] | SERCOM2/PAD[0] | | | | GCLK_IO[6] | |
| 69 | PB21 | VDDIO | EXTINT[5] | | | | | | | | SERCOM3/PAD[1] | SERCOM2/PAD[1] | | | | GCLK_IO[7] | |
| 70 | PA20 | VDDIO | EXTINT[4] | | | | | | X[8]/Y[24] | | SERCOM5/PAD[2] | SERCOM3/PAD[2] | TC7/WO[0] | TCC2/WO[0] | | GCLK_IO[4] | |
| 71 | PA21 | VDDIO | EXTINT[5] | | | | | | X[9]/Y[25] | | SERCOM5/PAD[3] | SERCOM3/PAD[3] | TC7/WO[1] | TCC2/WO[1] | | GCLK_IO[5] | |
| 72 | PA22 | VDDIO | EXTINT[6] | | | | | | X[10]/Y[26] | | SERCOM3/PAD[0] | SERCOM5/PAD[0] | TC4/WO[0] | TCC1/WO[0] | | GCLK_IO[6] | CCL2/IN[6] |
| 73 | PA23 | VDDIO | EXTINT[7] | | | | | | X[11]/Y[27] | | SERCOM3/PAD[1] | SERCOM5/PAD[1] | TC4/WO[1] | TCC1/WO[1] | | GCLK_IO[7] | CCL2/IN[7] |
| 74 | PA24 | VDDIO | EXTINT[12] | | | | | | | | SERCOM3/PAD[2] | SERCOM5/PAD[2] | TC5/WO[0] | TCC2/WO[0] | | CMP[2] | CCL2/IN[8] |
| 75 | PA25 | VDDIO | EXTINT[13] | | | | | | | | SERCOM3/PAD[3] | SERCOM5/PAD[3] | TC5/WO[1] | TCC2/WO[1] | | CMP[3] | CCL2/OUT[2] |
| 78 | PB22 | VDDIO | EXTINT[6] | | | | | | | | SERCOM0/PAD[2] | SERCOM5/PAD[2] | TC7/WO[0] | TCC1/WO[2] | | GCLK_IO[0] | CCL0/IN[0] |
| 79 | PB23 | VDDIO | EXTINT[7] | | | | | | | | SERCOM0/PAD[3] | SERCOM5/PAD[3] | TC7/WO[1] | TCC1/WO[3] | | GCLK_IO[1] | CCL0/OUT[0] |
| 80 | PB24 | VDDIO | EXTINT[8] | | | | | | | | SERCOM0/PAD[0] | SERCOM4/PAD[0] | | | | CMP[0] | |
| 81 | PB25 | VDDIO | EXTINT[9] | | | | | | | | SERCOM0/PAD[1] | SERCOM4/PAD[1] | | | | CMP[1] | |
| 82 | PC24 | VDDIO | EXTINT[0] | | | | | | | | SERCOM0/PAD[2] | SERCOM4/PAD[2] | | | | | |
| 83 | PC25 | VDDIO | EXTINT[1] | | | | | | | | SERCOM0/PAD[3] | SERCOM4/PAD[3] | | | | | |
| 84 | PC26 | VDDIO | EXTINT[2] | | | | | | | | | | | | | | |
| 85 | PC27 | VDDIO | EXTINT[3] | | | | | | | | | SERCOM1/PAD[0] | | | | | CCL1/IN[4] |
| 86 | PC28 | VDDIO | EXTINT[4] | | | | | | | | | SERCOM1/PAD[1] | | | | | CCL1/IN[5] |
| 87 | PA27 | VDDIN | EXTINT[15] | | | | | | | | | | | | | GCLK_IO[0] | |
| 89 | PA28 | VDDIN | EXTINT[8] | | | | | | | | | | | | | GCLK_IO[0] | |
| 93 | PA30 | VDDIN | EXTINT[10] | | | | | | | | | SERCOM1/PAD[2] | TC1/WO[0] | | CORTEX_M0P/SWCLK | GCLK_IO[0] | CCL1/IN[3] |
| 94 | PA31 | VDDIN | EXTINT[11] | | | | | | | | | SERCOM1/PAD[3] | TC1/WO[1] | | CORTEX_M0P/SWDIO | | CCL1/OUT[1] |
| 95 | PB30 | VDDIN | EXTINT[14] | | | | | | | | SERCOM1/PAD[0] | SERCOM5/PAD[0] | TC0/WO[0] | | | CMP[2] | |
| 96 | PB31 | VDDIN | EXTINT[15] | | | | | | | | SERCOM1/PAD[1] | SERCOM5/PAD[1] | TC0/WO[1] | | | CMP[3] | |
| 97 | PB00 | VDDANA | EXTINT[0] | | | AIN[0] | | | Y[6] | | | SERCOM5/PAD[2] | TC7/WO[0] | | | | CCL0/IN[1] |
| 98 | PB01 | VDDANA | EXTINT[1] | | | AIN[1] | | | Y[7] | | | SERCOM5/PAD[3] | TC7/WO[1] | | | | CCL0/IN[2] |
| 99 | PB02 | VDDANA | EXTINT[2] | | | AIN[2] | | | Y[8] | | | SERCOM5/PAD[0] | TC6/WO[0] | | | | CCL0/OUT[0] |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-3.** PORT Function Multiplexing for SAM C20 N (continued)

| Pin | I/O Pin | Supply | A | B[1,2] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | EIC | REF | ADC0 | ADC1 | SDADC | AC | PTC | DAC | SERCOM | SERCOM-ALT | TC | TCC | COM | AC/GCLK | CCL |
| 100 | PB03 | VDDANA | EXTINT[3] | | | AIN[3] | | | Y[9] | | | SERCOM5/PAD[1] | TC6/WO[1] | | | | |

**Note:**

1. All analog pin functions are on peripheral function B. Peripheral function B must be selected to disable the digital control of the pin.
2. Only some pins can be used in SERCOM $I^2C$ mode. For additional information, refer to SERCOM I2C Pins.

**Table 6-4.** PORT Function Multiplexing for SAM C20 E/G/J

| Pin[1] | | | I/O Pin | Supply | A | B[2,3] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SAM C20E | SAM C20G | SAM C20J | | | EIC | REF | ADC0 | ADC1 | AC | PTC | DAC | SDADC | SERCOM[2,3] | SERCOM-ALT | TC<br>TCC | TCC | COM | AC/GCLK | CCL |
| 1 | 1 | 1 | PA00 | VDDANA | EXTINT[0] | | | | | | | | | SERCOM1/PAD[0] | TCC2/WO[0] | | | CMP[2] | |
| 2 | 2 | 2 | PA01 | VDDANA | EXTINT[1] | | | | | | | | | SERCOM1/PAD[1] | TCC2/WO[1] | | | CMP[3] | |
| 3 | 3 | 3 | PA02 | VDDANA | EXTINT[2] | | AIN[0] | | AIN[4] | Y[0] | VOUT | | | | | | | | |
| 4 | 4 | 4 | PA03 | VDDANA | EXTINT[3] | ADC-DAC/VREFA | AIN[1] | | AIN[5] | Y[1] | | | | | | | | | |
| | | 5 | PB04 | VDDANA | EXTINT[4] | | | AIN[6] | | Y[10] | | | | | | | | | |
| | | 6 | PB05 | VDDANA | EXTINT[5] | | | AIN[7] | AIN[6] | Y[11] | | | | | | | | | |
| | | 9 | PB06 | VDDANA | EXTINT[6] | | | AIN[8] | AIN[7] | Y[12] | | INN[2] | | | | | | | CCL2/IN[6] |
| | | 10 | PB07 | VDDANA | EXTINT[7] | | | AIN[9] | | Y[13] | | INP[2] | | | | | | | CCL2/IN[7] |
| | 7 | 11 | PB08 | VDDANA | EXTINT[8] | | AIN[2] | AIN[4] | | Y[14] | | INN[1] | | - | TC0/WO[0] | | | | CCL2/IN[8] |
| | 8 | 12 | PB09 | VDDANA | EXTINT[9] | | AIN[3] | AIN[5] | | Y[15] | | INP[1] | | - | TC0WO[1] | | | | CCL2/OUT[2] |
| 5 | 9 | 13 | PA04 | VDDANA | EXTINT[4] | | AIN[4] | | AIN[0] | Y[2] | | | | SERCOM0/PAD[0] | TCC0/WO[0] | | | | CCL0/IN[0] |
| 6 | 10 | 14 | PA05 | VDDANA | EXTINT[5] | | AIN[5] | | AIN[1] | Y[3] | | | | SERCOM0/PAD[1] | TCC0/WO[1] | | | | CCL0/IN[1] |
| 7 | 11 | 15 | PA06 | VDDANA | EXTINT[6] | | AIN[6] | | AIN[2] | Y[4] | | INN[0] | | SERCOM0/PAD[2] | TCC1/WO[0] | | | | CCL0/IN[2] |
| 8 | 12 | 16 | PA07 | VDDANA | EXTINT[7] | | AIN[7] | | AIN[3] | Y[5] | | INP[0] | | SERCOM0/PAD[3] | TCC1/WO[1] | | | | CCL0/OUT[0] |
| 11 | 13 | 17 | PA08 | VDDIO | NMI | | AIN[8] | AIN[10] | | X[0]/Y[16] | | | SERCOM0/PAD[0] | SERCOM2/PAD[0] | TCC0/WO[0] | TCC1/WO[2] | | | CCL1/IN[3] |
| 12 | 14 | 18 | PA09 | VDDIO | EXTINT[9] | | AIN[9] | AIN[11] | | X[1]/Y[17] | | | SERCOM0/PAD[1] | SERCOM2/PAD[1] | TCC0/WO[1] | TCC1/WO[3] | | | CCL1/IN[4] |
| 13 | 15 | 19 | PA10 | VDDIO | EXTINT[10] | | AIN[10] | | | X[2]/Y[18] | | | SERCOM0/PAD[2] | SERCOM2/PAD[2] | TCC1/WO[0] | TCC0/WO[2] | | GCLK_IO[4] | CCL1/IN[5] |
| 14 | 16 | 20 | PA11 | VDDIO | EXTINT[11] | | AIN[11] | | | X[3]/Y[19] | | | SERCOM0/PAD[3] | SERCOM2/PAD[3] | TCC1/WO[1] | TCC0/WO[3] | | GCLK_IO[5] | CCL1/OUT[1] |
| | 19 | 23 | PB10 | VDDIO | EXTINT[10] | | | | | | | | | - | TC1/WO[0] | TCC0/WO[4] | | GCLK_IO[4] | CCL1/IN[5] |
| | 20 | 24 | PB11 | VDDIO | EXTINT[11] | | | | | | | | | - | TC1/WO[1] | TCC0/WO[5] | | GCLK_IO[5] | CCL1/OUT[1] |
| | | 25 | PB12 | VDDIO | EXTINT[12] | | | | | X[12]/Y[28] | | | - | | TC0/WO[0] | TCC0/WO[6] | | GCLK_IO[6] | |
| | | 26 | PB13 | VDDIO | EXTINT[13] | | | | | X[13]/Y[29] | | | - | | TC0/WO[1] | TCC0/WO[7] | | GCLK_IO[7] | |
| | | 27 | PB14 | VDDIO | EXTINT[14] | | | | | X[14]/Y[30] | | | - | | TC1/WO[0] | | | GCLK_IO[0] | CCL3/IN[9] |
| | | 28 | PB15 | VDDIO | EXTINT[15] | | | | | X[15]/Y[31] | | | - | | TC1/WO[1] | | | GCLK_IO[1] | CCL3/IN[10] |
| | 21 | 29 | PA12 | VDDIO | EXTINT[12] | | | | | | | | SERCOM2/PAD[0] | - | TCC2/WO[0] | TCC0/WO[6] | | AC/CMP[0] | |
| | 22 | 30 | PA13 | VDDIO | EXTINT[13] | | | | | | | | SERCOM2/PAD[1] | - | TCC2/WO[1] | TCC0/WO[7] | | AC/CMP[1] | |
| 15 | 23 | 31 | PA14 | VDDIO | EXTINT[14] | | | | | | | | SERCOM2/PAD[2] | - | TC4/WO[0] | TCC0/WO[4] | | GCLK_IO[0] | |
| 16 | 24 | 32 | PA15 | VDDIO | EXTINT[15] | | | | | | | | SERCOM2/PAD[3] | - | TC4/WO[1] | TCC0/WO[5] | | GCLK_IO[1] | |
| 17 | 25 | 35 | PA16 | VDDIO | EXTINT[0] | | | | | X[4]/Y[20] | | | SERCOM1/PAD[0] | SERCOM3/PAD[0] | TCC2/WO[0] | TCC0/WO[6] | | GCLK_IO[2] | CCL0/IN[0] |
| 18 | 26 | 36 | PA17 | VDDIO | EXTINT[1] | | | | | X[5]/Y[21] | | | SERCOM1/PAD[1] | SERCOM3/PAD[1] | TCC2/WO[1] | TCC0/WO[7] | | GCLK_IO[3] | CCL0/IN[1] |
| 19 | 27 | 37 | PA18 | VDDIO | EXTINT[2] | | | | | X[6]/Y[22] | | | SERCOM1/PAD[2] | SERCOM3/PAD[2] | TC4/WO[0] | TCC0/WO[2] | | AC/CMP[0] | CCL0/IN[2] |
| 20 | 28 | 38 | PA19 | VDDIO | EXTINT[3] | | | | | X[7]/Y[23] | | | SERCOM1/PAD[3] | SERCOM3/PAD[3] | TC4/WO[1] | TCC0/WO[3] | | AC/CMP[1] | CCL0/OUT[0] |
| | | 39 | PB16 | VDDIO | EXTINT[0] | | | | | | | | - | | TC2/WO[0] | TCC0/WO[4] | | GCLK_IO[2] | CCL3/IN[11] |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-4.** PORT Function Multiplexing for SAM C20 E/G/J (continued)

| | Pin[1] | | I/O Pin | Supply | A | B[2,3] | | | | | | | C | D | E | F | G | H | I |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SAM C20E | SAM C20G | SAM C20J | | | EIC | REF | ADC0 | ADC1 | AC | PTC | DAC | SDADC | SERCOM[2,3] | SERCOM-ALT | TC<br>TCC | TCC | COM | AC/GCLK | CCL |
| | | 40 | PB17 | VDDIO | EXTINT[1] | | | | | | | | - | | TC2/WO[1] | TCC0/<br>WO[5] | | GCLK_IO[3] | CCL3/<br>OUT[3] |
| | 29 | 41 | PA20 | VDDIO | EXTINT[4] | | | | | X[8]/Y[24] | | | - | SERCOM3/<br>PAD[2] | TC3/WO[0] | TCC0/<br>WO[6] | | GCLK_IO[4] | |
| | 30 | 42 | PA21 | VDDIO | EXTINT[5] | | | | | X[9]/Y[25] | | | - | SERCOM3/<br>PAD[3] | TC3/WO[1] | TCC0/<br>WO[7] | | GCLK_IO[5] | |
| 21 | 31 | 43 | PA22 | VDDIO | EXTINT[6] | | | | | X[10]/Y[26] | | | SERCOM3/<br>PAD[0] | - | TC0/WO[0] | TCC0/<br>WO[4] | | GCLK_IO[6] | CCL2/<br>IN[6] |
| 22 | 32 | 44 | PA23 | VDDIO | EXTINT[7] | | | | | X[11]/Y[27] | | | SERCOM3/<br>PAD[1] | - | TC0/WO[1] | TCC0/<br>WO[5] | | GCLK_IO[7] | CCL2/<br>IN[7] |
| 23 | 33 | 45 | PA24 | VDDIO | EXTINT[12] | | | | | | | | SERCOM3/<br>PAD[2] | - | TC1/WO[0] | TCC1/<br>WO[2] | | AC/CMP[2] | CCL2/<br>IN[8] |
| 24 | 34 | 46 | PA25 | VDDIO | EXTINT[13] | | | | | | | | SERCOM3/<br>PAD[3] | - | TC1/WO[1] | TCC1/<br>WO[3] | | AC/CMP[3] | CCL2/<br>OUT[2] |
| | 37 | 49 | PB22 | VDDIN | EXTINT[6] | | | | | | | | | - | TC3/WO[0] | | | GCLK_IO[0] | CCL0/<br>IN[0] |
| | 38 | 50 | PB23 | VDDIN | EXTINT[7] | | | | | | | | | - | TC3/WO[1] | | | GCLK_IO[1] | CCL0/<br>OUT[0] |
| 25 | 39 | 51 | PA27 | VDDIN | EXTINT[15] | | | | | | | | | | | | | GCLK_IO[0] | |
| 27 | 41 | 53 | PA28 | VDDIN | EXTINT[8] | | | | | | | | | | | | | GCLK_IO[0] | |
| 31 | 45 | 57 | PA30 | VDDIN | EXTINT[10] | | | | | | | | | SERCOM1/<br>PAD[2] | TCC1/WO[0] | | CORTEX_M0P/<br>SWCLK | GCLK_IO[0] | CCL1/<br>IN[3] |
| 32 | 46 | 58 | PA31 | VDDIN | EXTINT[11] | | | | | | | | | SERCOM1/<br>PAD[3] | TCC1/WO[1] | | CORTEX_M0P/<br>SWDIO | | CCL1/<br>OUT[1] |
| | | 59 | PB30 | VDDIN | EXTINT[14] | | | | | | | | | - | TCC0/WO[0] | TCC1/<br>WO[2] | | AC/CMP[2] | |
| | | 60 | PB31 | VDDIN | EXTINT[15] | | | | | | | | | - | TCC0/WO[1] | TCC1/<br>WO[3] | | AC/CMP[3] | |
| | | 61 | PB00 | VDDANA | EXTINT[0] | | | AIN[0] | | Y[6] | | | | - | TC3/WO[0] | | | | CCL0/<br>IN[1] |
| | | 62 | PB01 | VDDANA | EXTINT[1] | | | AIN[1] | | Y[7] | | | | - | TC3/WO[1] | | | | CCL0/<br>IN[2] |
| | 47 | 63 | PB02 | VDDANA | EXTINT[2] | | | AIN[2] | | Y[8] | | | | - | TC2/WO[0] | | | | CCL0/<br>OUT[0] |
| | 48 | 64 | PB03 | VDDANA | EXTINT[3] | | | AIN[3] | | Y[9] | | | | - | TC2/WO[1] | | | | |

**Note:**

1. Use the SAM C21J pinout muxing for the WLCSP56 package.
2. All analog pin functions are on peripheral function B. Peripheral function B must be selected to disable the digital control of the pin.
3. Only some pins can be used in SERCOM $I^2C$ mode. For additional information, refer to SERCOM I2C Pins.

**Related Links**

SERCOM I2C Pins

## 6.2. Other Functions

### 6.2.1. Oscillator Pinout

The oscillators are not mapped to the normal PORT functions and their multiplexing are controlled by registers in the Oscillators Controller (OSCCTRL) and in the 32K Oscillators Controller (OSC32KCTRL).

**Table 6-5.** Oscillator Pinout

| Oscillator | Supply | Signal | I/O pin |
|---|---|---|---|
| XOSC | VDDIO | XIN | PA14 |
| | | XOUT | PA15 |
| XOSC32K | VDDANA | XIN32 | PA00 |
| | | XOUT32 | PA01 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 6.2.2. Serial Wire Debug Interface Pinout

Only the SWCLK pin is mapped to the normal PORT functions. A debugger cold-plugging or hot-plugging detection will automatically switch the SWDIO port to the SWDIO function.

**Table 6-6.** Serial Wire Debug Interface Pinout

| Signal | Supply | I/O pin |
|---|---|---|
| SWCLK | VDDIN | PA30 |
| SWDIO | VDDIN | PA31 |

### 6.2.3. SERCOM $I^2C$ Pins

**Table 6-7.** SERCOM Pins Supporting $I^2C$

| Package | Pins Supporting $I^2C$ |
|---|---|
| 32-pin | PA08, PA09, PA16, PA17, PA22, PA23 |
| 48-pin | PA08, PA09, PA12, PA13, PA16, PA17, PA22, PA23 |
| 64-pin | PA08, PA09, PA12, PA13, PA16, PA17, PA22, PA23, PB12, PB13, PB16, PB17, PB30, PB31 |
| 100-pin | PA08, PA09, PA16, PA17, PB12, PB13, PB16, PB17 |

### 6.2.4. GPIO Clusters

**Table 6-8.** GPIO Clusters

| Package | Cluster | GPIO | Supplies Pin connected to the cluster | |
|---|---|---|---|---|
| 100 pins | 1 | PB31 PB30 PA31 PA30 PA28 PA27 | VDDIN (92) | GND (90) |
| | 2 | PC28 PC27 PC26 PC25 PC24 PB25 PB24 PB23 PB22 | VDDIO (77) | GND( 76 ) |
| | 3 | PA25 PA24 PA23 PA22 PA21 PA20 PB21 PB20 PB19 PB18 PB17 PB16 | VDDIO(63+77) | GND(62+76) |
| | 4 | PC21 PC20 PC19 PC18 PC17 PC16 PA19 PA18 PA17 PA16 | VDDIO(51+63) | GND(50+62) |
| | 5 | PA15 PA14 PA13 PA12 PC14 PC13 PC12 PC11 PC10 PC09 PC08 | VDDIO(36+51) | GND(37+50) |
| | 6 | PB15 PB13 PB12 PB11 PB10 PA11 PA10 PA09 PA08 | VDDIO(25+36) | GND(24+37) |
| | 7 | PC07 PC06 PC05 PA07 PA06 PA05 PA04 PB09 PB05 PB04 PA03 PA02 PC02 PC01 PC00 PA01 PA00 PB03 PB02 PB01 PB00 | VDDANA (12) | GNDANA (11) |
| | 8 | PC15 | VDDIO(25) | GND(37+50) |
| | 9 | PB14 | VDDIO(25) | GND(24+37) |
| | 10 | PB08 PB07 PB06 PC03 | VDDIO(25) | GNDANA (11) |
| 64 pins | 1 | PB31 PB30 PA31 PA30 PA28 PA27 | VDDIN (56) | GND (54) |
| | 2 | PB23 PB22 | VDDIO (48) | GND (54+47) |
| | 3 | PA25 PA24 PA23 PA22 PA21 PA20 PB17 PB16 PA19 PA18 PA17 PA16 | VDDIO (48+34) | GND (47+33) |
| | 4 | PA15 PA14 PA13 PA12 PB15 PB14 PB13 PB12 PB11 PB10 | VDDIO (34+21) | GND (33+22) |
| | 5 | PA11 PA10 PA08 PA09 | VDDIO (21) | GND (22) |
| | 6 | PA07 PA06 PA05 PA04 PB09 PB08 PB07 PB06 PB05 PB04 PA03 PA02 PA01 PA00 PB03 PB02 PB01 PB00 | VDDANA (8) | GNDANA (7) |
| 48 pins | 1 | PA31 PA30 PA28 PA27 | VDDIN (44) | GND (42) |
| | 2 | PB23 PB22 | VDDIO (36) | GND (42+35) |
| | 3 | PA25 PA24 PA23 PA22 PA21 PA20 PA19 PA18 PA17 PA16 PA15 PA14 PA13 PA12 PB11 PB10 | VDDIO (36+17) | GND (35+18) |
| | 4 | PA11 PA10 PA08 PA09 | VDDIO (17) | GND (18) |
| | 5 | PA07 PA06 PA05 PA04 PB09 PB08 PA03 PA02 PA01 PA00 PB03 PB02 | VDDANA (6) | GNDANA (5) |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 6-8.** GPIO Clusters (continued)

| Package | Cluster | GPIO | Supplies Pin connected to the cluster | |
|---|---|---|---|---|
| 32 pins | 1 | PA31 PA30 PA28 PA27 | VDDIN (30) | GND (28) |
| | 2 | PA25 PA24 PA23 PA22 PA19 PA18 PA17 PA16 PA15 PA14 PA11 PA10 PA08 PA09 | VDDANA (9) | GND (28+10) |
| | 3 | PA07 PA06 PA05 PA04 PA03 PA02 PA01 PA00 | VDDANA (9) | GND (28+10) |

### 6.2.5. TCC Configurations

The SAM C20/C21 devices have three instances of the Timer/Counter for Control applications (TCC) peripheral, , TCC[2:0]. The following table lists the features for each TCC instance.

**Table 6-9.** TCC Configuration Summary

| TCC# | Channels (CC_NUM) | Waveform Output (WO_NUM) | Counter size | Fault | Dithering | Output matrix | Dead Time Insertion (DTI) | SWAP | Pattern generation | Host/Client Operation |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 4 | 8 | 24-bit | Yes | Yes | Yes | Yes | Yes | Yes | Host |
| 1 | 2 | 4 | 24-bit | Yes | Yes | - | - | - | Yes | Client |
| 2 | 2 | 2 | 16-bit | Yes | - | - | - | - | - | - |

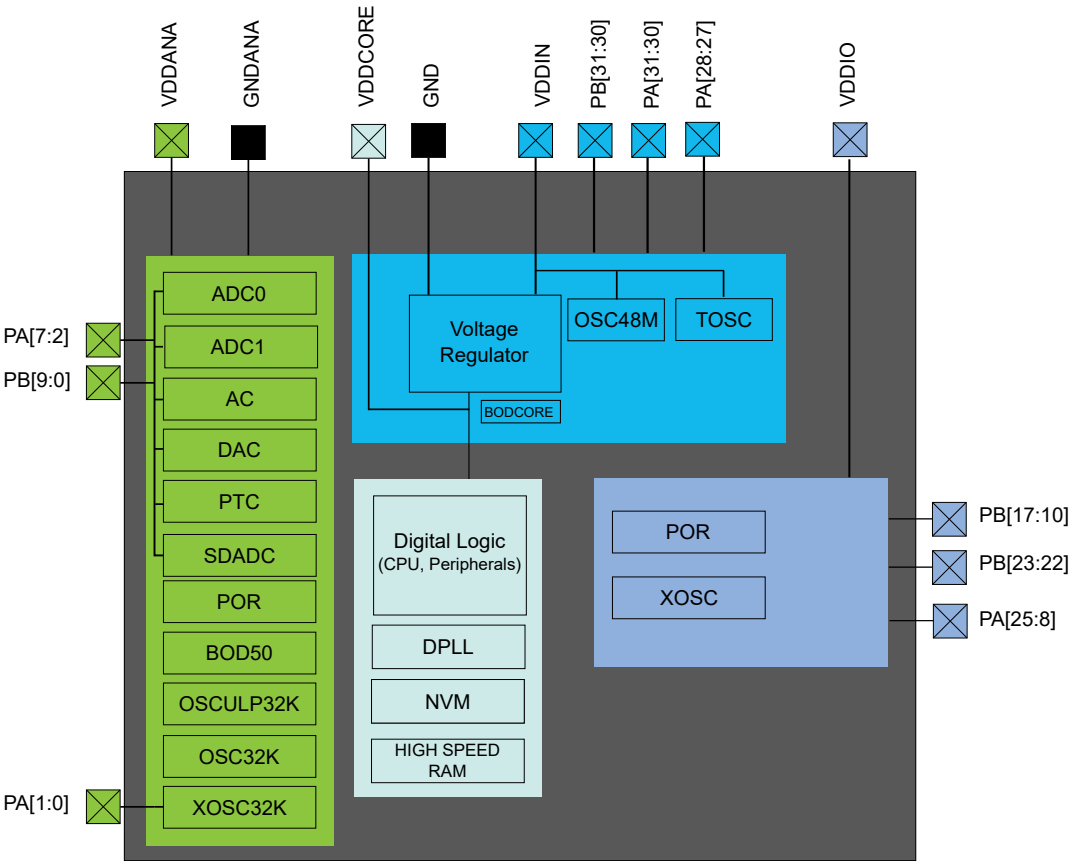
**Note:** The number of CC registers (CC_NUM) for each TCC corresponds to the number of compare/capture channels, therefore a TCC can have more Waveform Outputs (WO_NUM) than CC registers.


© 2026 Microchip Technology Inc. and its subsidiaries

# 7. Power Supply and Start-Up Considerations

## 7.1. Power Domain Overview

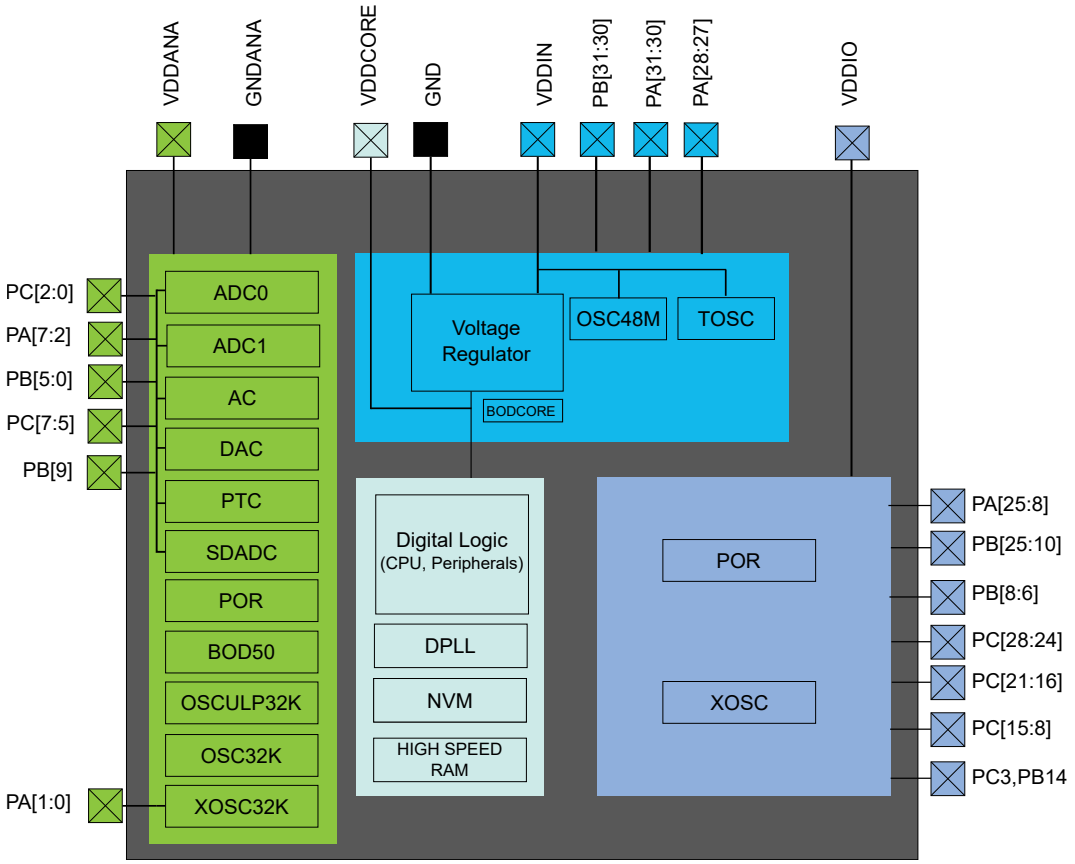
**Figure 7-1.** Power Domain Overview, SAM C20/C21 E/G/J





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 7-2.** Power Domain Overview, SAM C20/C21 N




## 7.2. Power Supply Considerations

### 7.2.1. Power Supplies, SAM C21/SAM C20

The SAM C21 has the following power supply pins:

- VDDIO: Powers I/O lines and XOSC. Voltage is 2.70V to 5.50V.
- VDDIN: Powers I/O lines and the OSC48M, TOSC and internal regulator. Voltage is 2.70V to 5.50V.
- VDDANA: Powers I/O lines and the ADC, AC, DAC, PTC, SDADC, OSCULP32K, OSC32K, and XOSC32K. Voltage is 2.70V to 5.50V.
- VDDCORE: Internal regulated voltage output. Powers the core, memories, peripherals, and FDPLL96M. Voltage is 1.2V typical.

The same voltage must be applied to both the VDDIN and VDDANA pins. This common voltage is referred to as $V_{DD}$ in the datasheet. VDDIO must always be less than or equal to VDDIN.

The ground pins, GND, are common to VDDCORE, VDDIO and VDDIN. The ground pin for VDDANA is GNDANA.

For decoupling recommendations for the different power supplies, refer to the schematic checklist.

The SAM C20 has the following power supply pins:

- VDDIO: Powers I/O lines and XOSC. Voltage is 2.70V to 5.50V.
- VDDIN: Powers I/O lines and the OSC48M, TOSC and internal regulator. Voltage is 2.70V to 5.50V.
- VDDANA: Powers I/O lines and the ADC, AC, PTC, OSCULP32K, OSC32K, and XOSC32K. Voltage is 2.70V to 5.50V.


© 2026 Microchip Technology Inc. and its subsidiaries

- VDDCORE: Internal regulated voltage output. Powers the core, memories, peripherals, and FDPLL96M. Voltage is 1.2V typical.

The same voltage must be applied to both VDDIN and VDDANA. This common voltage is referred to as $V_{DD}$ in the datasheet. VDDIO must always be less than or equal to VDDIN.

The ground pins, GND, are common to VDDCORE, VDDIO and VDDIN. The ground pin for VDDANA is GNDANA.

For decoupling recommendations for the different power supplies, refer to the schematic checklist.

### 7.2.2. Voltage Regulator

The SAM C20/C21 voltage regulators have two modes:

- Normal mode: This is the default mode when CPU and peripherals are running.
- Low Power (LP) mode: This default mode is used when the chip is in standby mode.

### 7.2.3. Typical Powering Schematics

The SAM C20/C21 use a single supply from 2.70V to 5.50V or dual supply mode where VDDIO is supplied separately from VDDIN.

The recommended power supply connections are shown in Figure 51-1, and Figure 51-2.

### 7.2.4. Power-Up Sequence - Maximum Rise Rate

The rise rate of the power supply must not exceed the values described in Electrical Characteristics.

## 7.3. Power-Up

This section summarizes the power-up sequence of the SAM C20/C21. The behavior after power-up is controlled by the Power Manager.

### 7.3.1. Starting of Clocks

After power-up, the device is set to its initial state and kept in reset, until the power has stabilized throughout the device. Once the power has stabilized, the device will use a 4 MHz clock. This clock is derived from the 48 MHz Internal Oscillator (OSC48M), which is configured to provide a 4 MHz clock and used as a clock source for generic clock generator '0'. Generic clock generator '0' is the main clock for the Power Manager (PM).

Some synchronous system clocks are active, allowing software execution.

Refer to the "Clock Mask Register" in the MCLK - Main Clock for the list of default peripheral clocks running. Synchronous system clocks that are running are by default not divided and receive a 4 MHz clock through generic clock generator '0'. Other generic clocks are disabled.

### 7.3.2. I/O Pins

After power-up, the I/O pins are tri-stated, with the exception of PA30. After power-up, PA30 is internally pulled up to avoid false detection of a debugger if the pin is left unconnected.

### 7.3.3. Fetching of Initial Instructions

After reset has been released, the CPU starts fetching PC and SP values from the reset address, 0x00000000, which points to the first executable address in the internal Flash. The code read from the internal Flash is free to configure the clock system and clock sources. Refer to the "Arm Architecture Reference Manual" for additional information on CPU start up, which is available for download at http://www.arm.com.

## 7.4. Power-on Reset and Brown-out Detector

The SAM C20/C21 embed three features to monitor, warn, and reset the device:


© 2026 Microchip Technology Inc. and its subsidiaries

- POR: Power-on Reset on VDDIN and VDDIO
- BODVDD: Brown-out Detector on VDDIN
- BODCORE: Voltage Regulator Internal Brown-out Detector on VDDCORE. The Voltage Regulator Internal BOD is calibrated in production and its calibration configuration is stored in the NVM User Row. For additional information, refer to the NVM user Row Mapping. This configuration should not be changed if the user row is written to assure the correct behavior of the BODCORE. This configuration is automatically copied at boot-up in the BODCORE registers.

### 7.4.1. Power-on Reset (POR) on VDDIN

POR monitors VDDIN. It is always activated and monitors voltage at startup and also during all the sleep modes. If VDDIN goes below the threshold voltage, the entire chip is reset.

### 7.4.2. Power-on Reset (POR) on VDDIO

POR monitors VDDIO. It is always activated and monitors voltage at startup and also during all the sleep modes. If VDDIO goes below the threshold voltage, all IOs supplied by VDDIO are reset.

### 7.4.3. Brown-out Detector (BOD) on VDDIN

BODVDD monitors VDDIN.

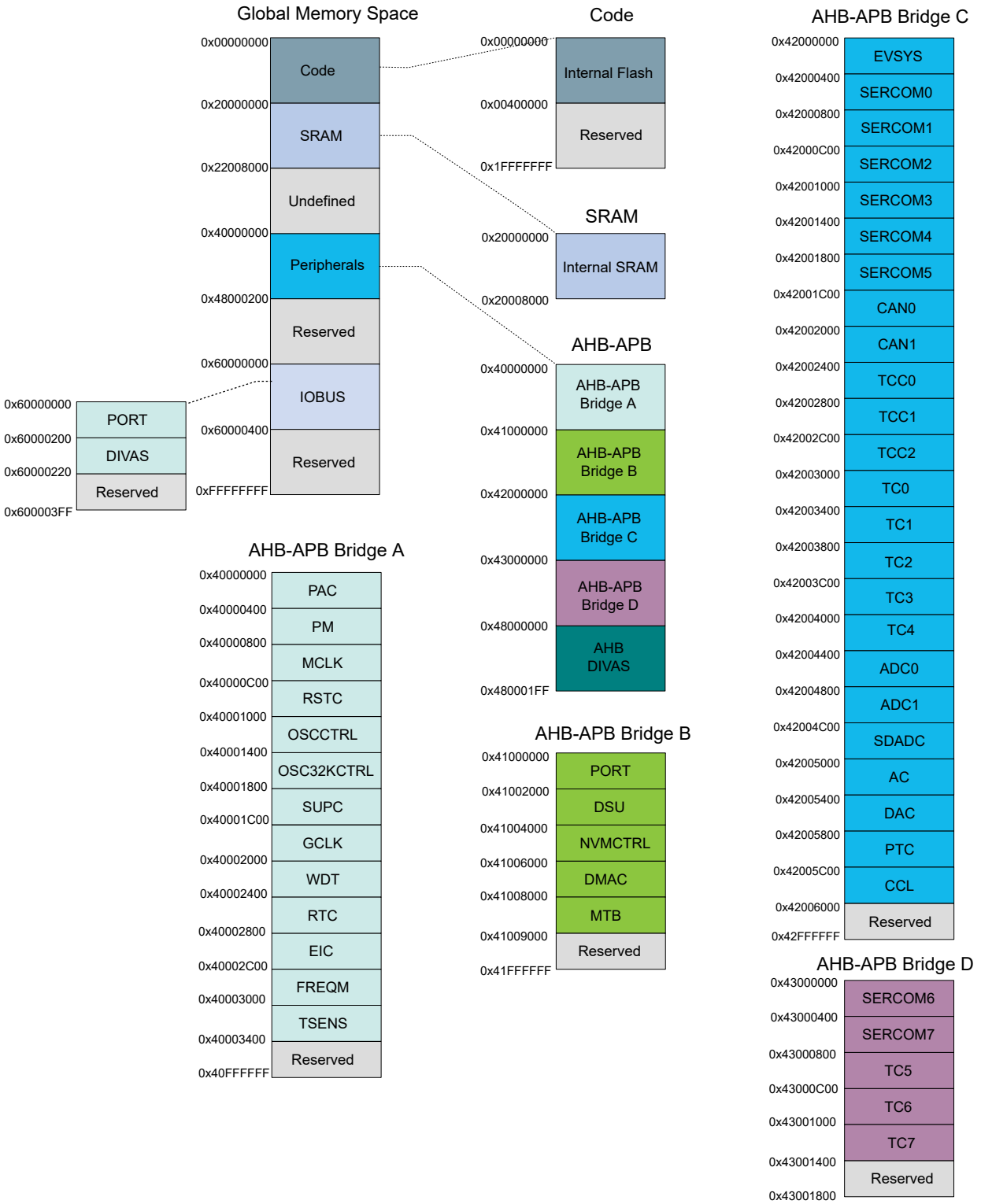
### 7.4.4. Brown-out Detector (BOD) on VDDCORE

Once the device has started up, BODCORE monitors the internal VDDCORE.


© 2026 Microchip Technology Inc. and its subsidiaries

# 8. Product Mapping

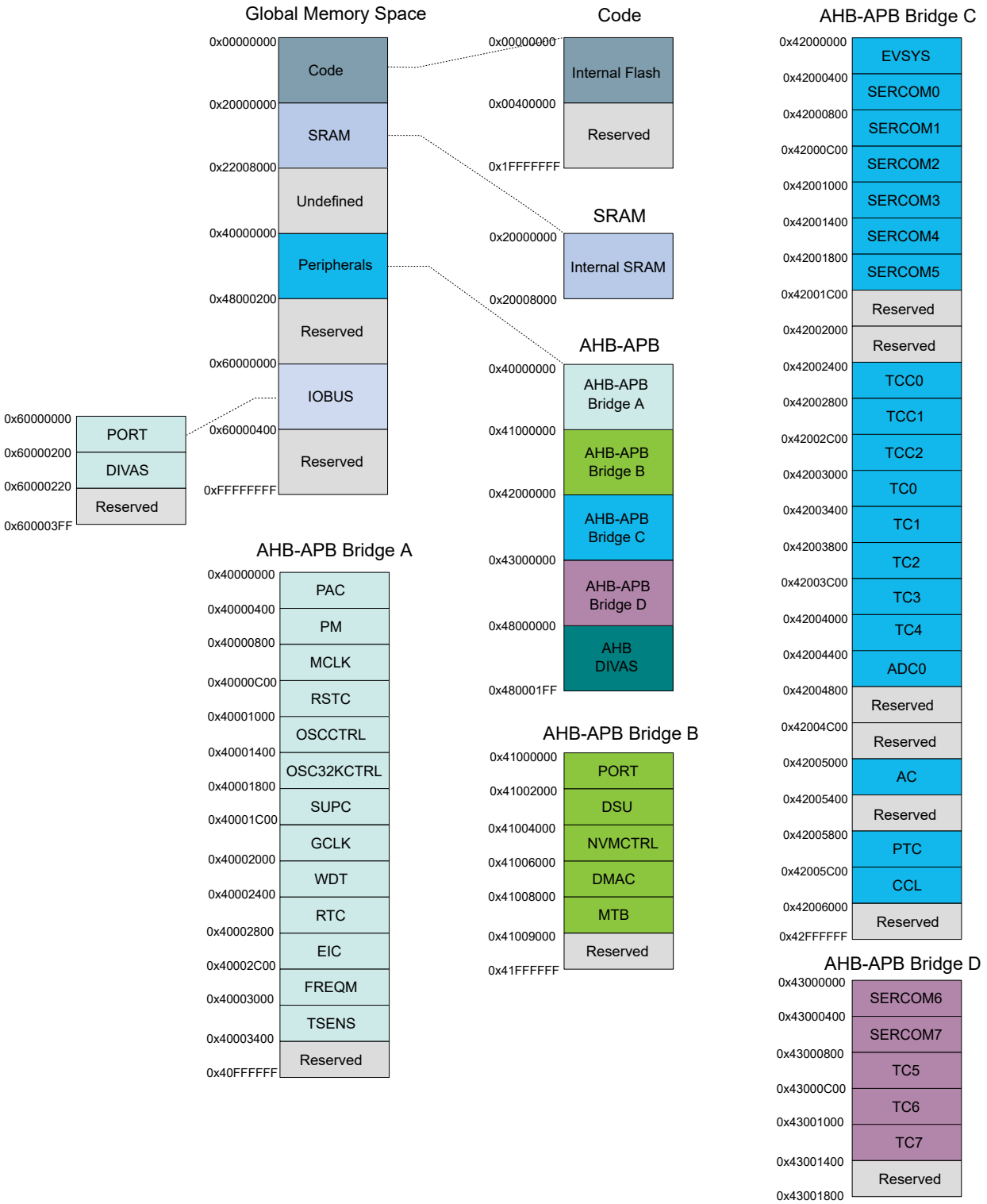
**Figure 8-1.** SAM C21 N Product Mapping





© 2026 Microchip Technology Inc. and its subsidiaries

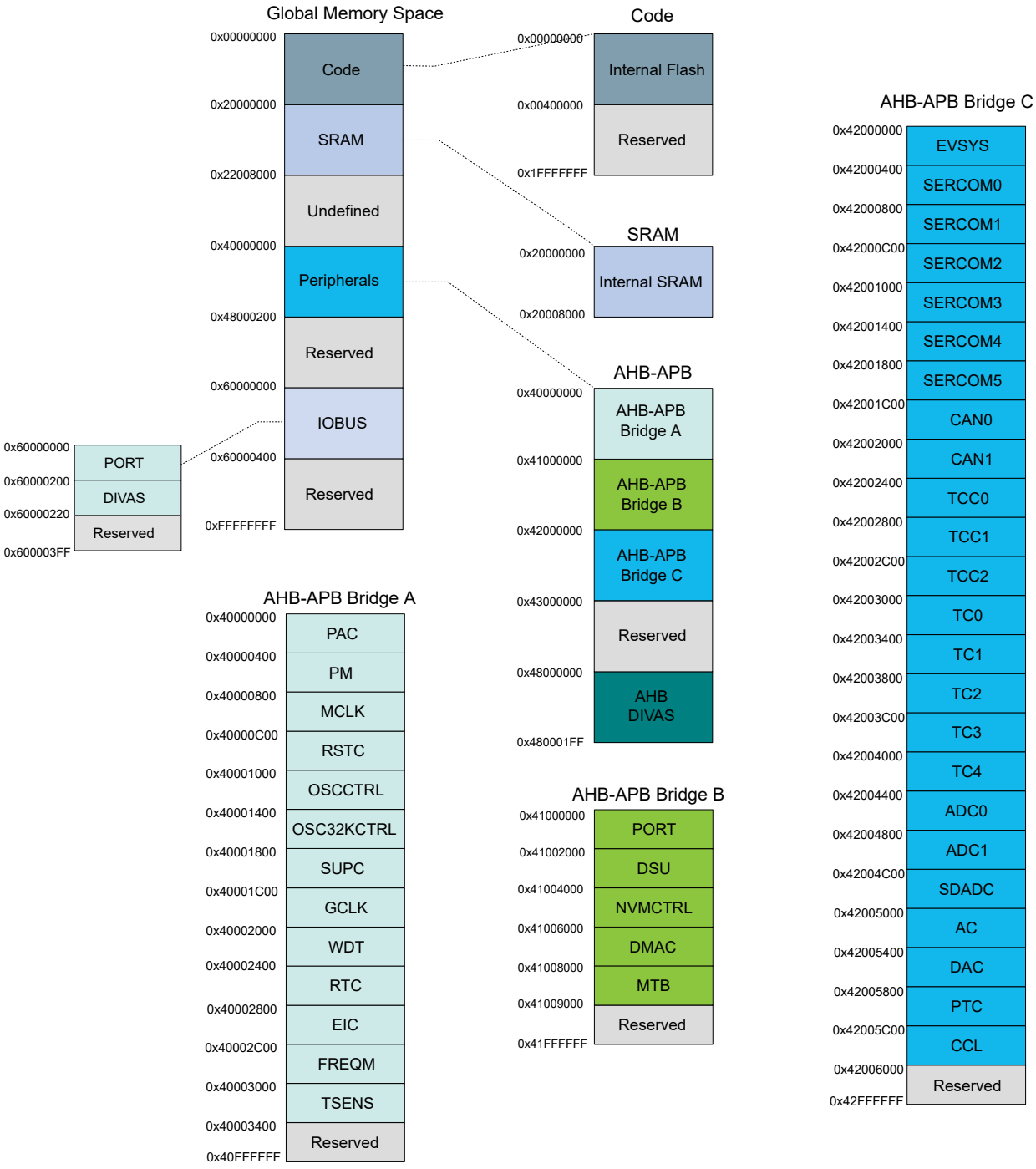
**Figure 8-2.** SAM C20 N Product Mapping





© 2026 Microchip Technology Inc. and its subsidiaries

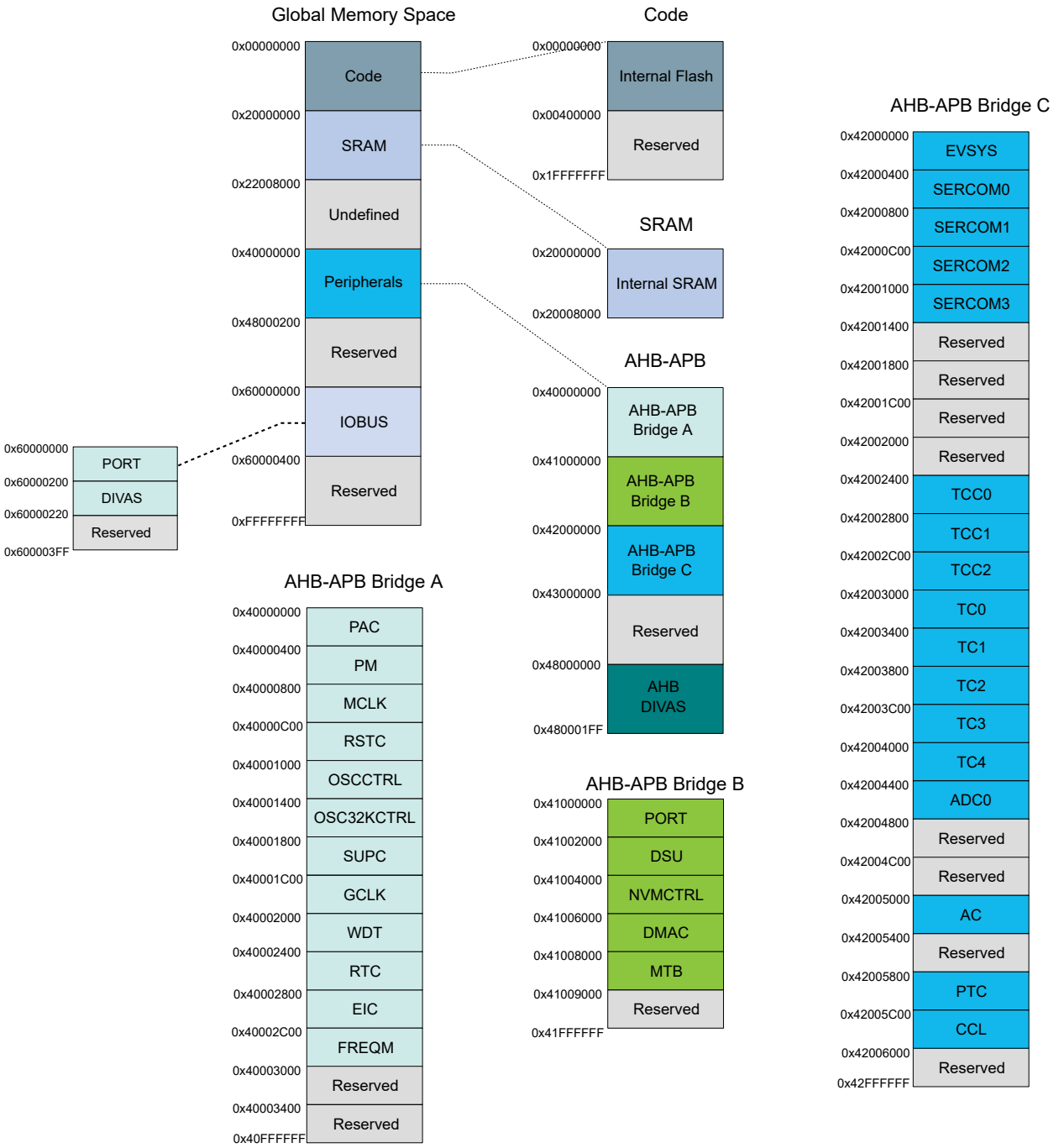
**Figure 8-3.** SAM C21 E/G/J Product Mapping





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 8-4.** SAM C20 E/G/J Product Mapping





© 2026 Microchip Technology Inc. and its subsidiaries

# 9. Memories

## 9.1. Embedded Memories

- Internal high-speed Flash with read-while-write capability on section of the array
- Internal high-speed RAM, single-cycle access at full speed

## 9.2. Physical Memory Map

The high-speed bus is implemented as a bus matrix. All high-speed bus addresses are fixed, and they are never remapped in any way, even during boot. The 32-bit physical address space is mapped as follows:

**Table 9-1.** SAM C20/C21 Physical Memory Map(1)

| Memory | Start address | Size | Size | Size | Size |
|---|---|---|---|---|---|
| | | x18 | x17 | x16 | x15 |
| Embedded Flash | 0x00000000 | 256 Kbytes | 128 Kbytes | 64 Kbytes | 32 Kbytes |
| Embedded RWW section | 0x00400000 | 8 Kbytes | 4 Kbytes | 2 Kbytes | 1 Kbytes |
| Embedded high-speed SRAM | 0x20000000 | 32 Kbytes | 16 Kbytes | 8 Kbytes | 4 Kbytes |
| AHB-APB Bridge A | 0x40000000 | 64 Kbytes | 64 Kbytes | 64 Kbytes | 64 Kbytes |
| AHB-APB Bridge B | 0x41000000 | 64 Kbytes | 64 Kbytes | 64 Kbytes | 64 Kbytes |
| AHB-APB Bridge C | 0x42000000 | 64 Kbytes | 64 Kbytes | 64 Kbytes | 64 Kbytes |
| AHB-APB Bridge D | 0x43000000 | 64 Kbytes | – | – | – |
| AHB DIVAS | 0x48000000 | 64 Kbytes | 64 Kbytes | 64 Kbytes | 64 Kbytes |
| IOBUS | 0x60000000 | 64 Kbytes | 64 Kbytes | 64 Kbytes | 64 Kbytes |

Note: 1. x = SAM C20/C21 G/J/E/N. The N-series (100-pin devices) does not include x16 and x15 option.

**Table 9-2.** SAM C20/C21 Flash Memory Parameters(1)

| Device | Flash size (FLASH_PM) | Number of pages (FLASH_P) | Page size (FLASH_W) |
|---|---|---|---|
| x18 | 256Kbytes | 4096 | 64 bytes |
| x17 | 128Kbytes | 2048 | 64 bytes |
| x16 | 64Kbytes | 1024 | 64 bytes |
| x15 | 32Kbytes | 512 | 64 bytes |

Note: 1. x = SAM C20/C21 G/J/E/N. The N-series (100-pin devices) does not include x16 and x15 option.

**Table 9-3.** SAM C20/C21 RWW Section Parameters(1)

| Device | Flash size (FLASH_PM) | Number of pages (FLASH_P) | Page size (FLASH_W) |
|---|---|---|---|
| x18 | 8Kbytes | 128 | 64 bytes |
| x17 | 4Kbytes | 64 | 64 bytes |
| x16 | 2Kbytes | 32 | 64 bytes |
| x15 | 1Kbytes | 16 | 64 bytes |

Note: 1. x = SAM C20/C21 G/J/E/N. The N-series (100-pin devices) does not include x16 and x15 option.

## 9.3. NVM User Row Mapping

The first two 32-bit words of the NVM User Row contains calibration data that are automatically read at device power on.


© 2026 Microchip Technology Inc. and its subsidiaries

The NVM User Row can be read at address 0x804000.

To write the NVM User Row, refer to the NVMCTRL - Non-Volatile Memory Controller.

Note that when writing to the user row the values do not get loaded by the other modules on the device until a device reset occurs.

**Table 9-4.** NVM User Row Mapping

| Bit Position | Name | Usage | Production setting | Related Peripheral Register |
|---|---|---|---|---|
| 2:0 | BOOTPROT | Used to select one of eight different bootloader sizes. | 0x7 | NA, See table 27-2 |
| 3 | Reserved | - | 0x1 | - |
| 6:4 | EEPROM | Used to select one of eight different EEPROM Emulation area sizes. | 0x7 | NA, See table 27-3 |
| 7 | Reserved | - | 0x1 | - |
| 13:8 | BODVDD Level | BODVDD Threshold Level at power on. | 0x8 | SUPC.BODVDD.LEVEL |
| 14 | BODVDD Disable | BODVDD Disable at power on. | 0x0 (=BODVDD enabled) | SUPC.BODVDD.ENABLE |
| 16:15 | BODVDD Action | BODVDD Action at power on. | 0x1 | SUPC.BODVDD.ACTION |
| 25:17 | BODCORE calibration | DO NOT CHANGE (1) | 0xA8 | - |
| 26 | WDT Enable | WDT Enable at power on. | 0x0 | WDT.CTRLA.ENABLE |
| 27 | WDT Always-On | WDT Always-On at power on. | 0x0 | WDT.CTRLA.ALWAYSON |
| 31:28 | WDT Period | WDT Period at power on. | 0xB | WDT.CONFIG.PER |
| 35:32 | WDT Window | WDT Window mode time-out at power on. | 0xB | WDT.CONFIG.WINDOW |
| 39:36 | WDT EWOFFSET | WDT Early Warning Interrupt Time Offset at power on. | 0xB | WDT.EWCTRL.EWOFFSET |
| 40 | WDT WEN | WDT Timer Window Mode Enable at power on. | 0x0 | WDT.CTRLA.WEN |
| 41 | BODVDD Hysteresis | BODVDD Hysteresis configuration at power on. | 0x0 | SUPC.BODVDD.HYSTERESIS |
| 42 | BODCORE calibration | DO NOT CHANGE (1) | 0x0 | - |
| 47:43 | Reserved | - | 0x1F | - |
| 63:48 | LOCK | NVM Region Lock Bits. | 0xFFFF | NVMCTRL.LOCK |

**Note:**

1. BODCORE is calibrated in production and its calibration parameters must not be changed to ensure the correct device behavior.

**Related Links**

NVMCTRL – Nonvolatile Memory Controller

CTRLA

Control A

CONFIG

Configuration

EWCTRL

Early Warning Control

BODVDD

VDD Brown-Out Detector (BODVDD) Control


© 2026 Microchip Technology Inc. and its subsidiaries

## 9.4. NVM Software Calibration Area Mapping

The NVM Software Calibration Area contains calibration data that are measured and written during production test. These calibration values should be read by the application software and written back to the corresponding register.

The NVM Software Calibration Area can be read at address 0x806020.

The NVM Software Calibration Area can not be written.

**Table 9-5.** NVM Software Calibration Area Mapping

| Bit Position | Name | Description |
|---|---|---|
| 2:0 | ADC0 BIASREFBUF | ADC0 Linearity Calibration. Should be written to ADC0 CALIB.BIASREFBUF. |
| 5:3 | ADC0 BIASCOMP | ADC0 Bias Calibration. Should be written to ADC0 CALIB.BIASCOMP. |
| 8:6 | ADC1 BIASREFBUF | ADC1 Linearity Calibration. Should be written to ADC1 CALIB.BIASREFBUF. |
| 11:9 | ADC1 BIASCOMP | ADC1 Bias Calibration. Should be written to ADC1 CALIB.BIASCOMP. |
| 18:12 | OSC32K CAL | OSC32K Calibration. Should be written to OSC32KCTRL OSC32K.CALIB. |
| 40:19 | CAL48M 5V | OSC48M Calibration: VDD range 3.6V to 5.5V. The CAL48M register is loaded from the NVM User Row at startup. |
| 62:41 | CAL48M 3V3 | OSC48M Calibration: VDD range 2.7V to 3.6V. The CAL48M register is loaded from the NVM User Row at startup. |
| 63 | Reserved | - |

### Related Links

CAL48M

OSC48M Calibration

## 9.5. NVM Temperature Calibration Area Mapping, SAM C21

The NVM Temperature Calibration Area contains calibration data that are measured and written during production test. These calibration values should be read by the application software and written back to the corresponding register.

The NVM Temperature Calibration Area can be read at address 0x806030.

The NVM Temperature Calibration Area can not be written.

**Table 9-6.** NVM Temperature Calibration Area Mapping, SAM C21

| Bit Position | Name | Description |
|---|---|---|
| 5:0 | TSENS TCAL | TSENS Temperature Calibration. Should be written to TSENS CAL.TCAL. |
| 11:6 | TSENS FCAL | TSENS Frequency Calibration. Should be written to TSENS CAL.FCAL. |
| 35:12 | TSENS GAIN | TSENS Gain Calibration. Should be written to the TSENS GAIN register. |
| 59:36 | TSENS OFFSET | TSENS Offset Calibration. Should be written to TSENS OFFSET register. |
| 63:60 | Reserved | |

### Related Links

CAL

Calibration

GAIN

Gain

OFFSET

Offset


© 2026 Microchip Technology Inc. and its subsidiaries

## 9.6. Serial Number

Each device has a unique 128-bit serial number which is a concatenation of four 32-bit words contained at the following addresses:

Word 0: 0x0080A00C

Word 1: 0x0080A040

Word 2: 0x0080A044

Word 3: 0x0080A048

The uniqueness of the serial number is guaranteed only when using all 128 bits.


© 2026 Microchip Technology Inc. and its subsidiaries

# 10. Processor and Architecture

## 10.1. Cortex M0+ Processor

The SAM C20/C21 implement the Arm Cortex-M0+ processor, based on the ARMv6 Architecture and Thumb®-2 ISA. The Cortex M0+ is 100% instruction set compatible with its predecessor, the Cortex-M0 core, and upward compatible to Cortex-M3 and M4 cores. The implemented Arm Cortex-M0+ is revision r0p1. For additional information, refer to www.arm.com.

### 10.1.1. Cortex M0+ Configuration

**Table 10-1.** Cortex M0+ Configuration

| Features | Cortex M0+ options | SAM C20/C21 configurations |
|---|---|---|
| Interrupts | External interrupts 0-32 | 32 |
| Data endianness | Little-endian or big-endian | Little-endian |
| SysTick timer | Present or absent | Present |
| Number of watchpoint comparators | 0, 1, 2 | 2 |
| Number of breakpoint comparators | 0, 1, 2, 3, 4 | 4 |
| Halting debug support | Present or absent | Present |
| Multiplier | Fast or small | Fast (single cycle) |
| Single-cycle I/O port | Present or absent | Present |
| Wake-up interrupt controller | Supported or not supported | Not supported |
| Vector Table Offset Register | Present or absent | Present |
| Unprivileged/Privileged support | Present or absent | Present |
| Memory Protection Unit | Not present or 8-region | 8-region |
| Reset all registers | Present or absent | Absent |
| Instruction fetch width | 16-bit only or mostly 32-bit | 32-bit |

The Arm Cortex-M0+ core has two bus interfaces:

- Single 32-bit AMBA-3 AHB-Lite system interface that provides connections to peripherals and all system memory, which includes Flash and RAM.
- Single 32-bit I/O port bus interfacing to the PORT and DIVAS with 1-cycle loads and stores.

### 10.1.2. Cortex-M0+ Peripherals

- System Control Space (SCS)
  - The processor provides debug through registers in the SCS. Refer to the Cortex-M0+ Technical Reference Manual for details (www.arm.com).
- Nested Vectored Interrupt Controller (NVIC)
  - External interrupt signals connect to the NVIC, and the NVIC prioritizes the interrupts. Software can set the priority of each interrupt. The NVIC and the Cortex-M0+ processor core are closely coupled, providing low latency interrupt processing and efficient processing of late arriving interrupts. Refer to Nested Vector Interrupt Controller and the Cortex-M0+ Technical Reference Manual for details (www.arm.com).
- System Timer (SysTick)
  - The System Timer is a 24-bit timer clocked by CLK_CPU that extends the functionality of both the processor and the NVIC. Refer to the Cortex-M0+ Technical Reference Manual for details (www.arm.com).
- System Control Block (SCB)


© 2026 Microchip Technology Inc. and its subsidiaries

  - The System Control Block provides system implementation information, and system control. This includes configuration, control, and reporting of the system exceptions. Refer to the Cortex-M0+ Devices Generic User Guide for details (www.arm.com).
- Micro Trace Buffer (MTB)
  - The CoreSight MTB-M0+ (MTB) provides a simple execution trace capability to the Cortex-M0+ processor. Refer to section Micro Trace Buffer and the CoreSight MTB-M0+ Technical Reference Manual for details (www.arm.com).
- Memory Protection Unit (MPU)
  - The Memory Protection Unit divides the memory map into a number of regions, and defines the location, size, access permissions and memory attributes of each region. Refer to the Cortex-M0+ Devices Generic User Guide for details (http://www.arm.com)

### 10.1.3. Cortex-M0+ Address Map

**Table 10-2.** Cortex-M0+ Address Map

| Address | Peripheral |
|---|---|
| 0xE000E000 | System Control Space (SCS) |
| 0xE000E010 | System Timer (SysTick) |
| 0xE000E100 | Nested Vectored Interrupt Controller (NVIC) |
| 0xE000ED00 | System Control Block (SCB) |
| 0x41008000 | Micro Trace Buffer (MTB) |

**Related Links**
Product Mapping

### 10.1.4. I/O Interface

#### 10.1.4.1. Overview

Because accesses to the AMBA® AHB-Lite™ and the single cycle I/O interface can be made concurrently, the Cortex-M0+ processor can fetch the next instructions while accessing the I/Os. This enables single cycle I/O accesses to be sustained for as long as needed.

#### 10.1.4.2. Description

Direct access to PORT registers and DIVAS registers.

## 10.2. Nested Vector Interrupt Controller

### 10.2.1. Overview

The Nested Vectored Interrupt Controller (NVIC) in the SAM C20/C21 supports 32 interrupt lines with four different priority levels. For more details, refer to the Cortex-M0+ Technical Reference Manual (www.arm.com).

### 10.2.2. Interrupt Line Mapping

Each of the interrupt lines is connected to one peripheral instance, as shown in the table below. Each peripheral can have one or more interrupt flags, located in the peripheral's Interrupt Flag Status and Clear (INTFLAG) register.

The interrupt flag is set when the interrupt condition occurs. Each interrupt in the peripheral can be individually enabled by writing a one to the corresponding bit in the peripheral's Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the peripheral's Interrupt Enable Clear (INTENCLR) register.

An interrupt request is generated from the peripheral when the interrupt flag is set and the corresponding interrupt is enabled.


© 2026 Microchip Technology Inc. and its subsidiaries

The interrupt requests for one peripheral are ORed together on system level, generating one interrupt request for each peripheral. An interrupt request will set the corresponding interrupt pending bit in the NVIC interrupt pending registers (SETPEND/CLRPEND bits in ISPR/ICPR).

For the NVIC to activate the interrupt, it must be enabled in the NVIC interrupt enable register (SETENA/CLRENA bits in ISER/ICER). The NVIC interrupt priority registers IPR0-IPR7 provide a priority field for each interrupt.

**Table 10-3.** Interrupt Line Mapping, SAM C21

| Peripheral Source | NVIC Line | |
|---|---|---|
| EIC NMI – External Interrupt Controller | | NMI |
| MCLK - Main Clock<br>OSCCTRL - Oscillators Controller<br>OSC32KCTRL - 32.768 kHz Oscillators Controller<br>SUPC - Supply Controller<br>PAC - Protection Access Controller | | 0 |
| WDT – Watchdog Timer | | 1 |
| RTC – Real Time Clock | | 2 |
| EIC – External Interrupt Controller | | 3 |
| FREQM – Frequency Meter | | 4 |
| TSENS – Temperature Sensor | | 5 |
| NVMCTRL – Non-Volatile Memory Controller | | 6 |
| DMAC - Direct Memory Access Controller | | 7 |
| EVSYS – Event System | | 8 |
| SERCOM0 – Serial Communication Controller 0<br>SERCOM6 – Serial Communication Controller 6 (only for SAM C21 N) | | 9 |
| SERCOM1 – Serial Communication Controller 1<br>SERCOM7 – Serial Communication Controller 7 (only for SAM C21 N) | | 10 |
| SERCOM2 – Serial Communication Controller 2 | | 11 |
| SERCOM3 – Serial Communication Controller 3 | | 12 |
| SERCOM4 – Serial Communication Controller 4 (only for SAM C21 G/J/N) | | 13 |
| SERCOM5 – Serial Communication Controller 5 (only for SAM C21 G/J/N) | | 14 |
| CAN0 – Controller Area Network 0 | Line 0 | 15 |
| | Line 1 | |
| CAN1 – Controller Area Network 1 (only for SAM C21 G/J/N) | Line 0 | 16 |
| | Line 1 | |
| TCC0 – Timer Counter for Control 0 | | 17 |
| TCC1 – Timer Counter for Control 1 | | 18 |
| TCC2 – Timer Counter for Control 2 | | 19 |
| TC0 – Timer Counter 0<br>TC5 – Timer Counter 5 (only for SAM C21 N) | | 20 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 10-3.** Interrupt Line Mapping, SAM C21 (continued)

| Peripheral Source | NVIC Line |
|---|---|
| TC1 – Timer Counter 1<br>TC6 – Timer Counter 6 (only for SAM C21 N) | 21 |
| TC2 – Timer Counter 2<br>TC7 – Timer Counter 7 (only for SAM C21 N) | 22 |
| TC3 – Timer Counter 3 | 23 |
| TC4 – Timer Counter 4 | 24 |
| ADC0 – Analog-to-Digital Converter 0 | 25 |
| ADC1 – Analog-to-Digital Converter 1 | 26 |
| AC – Analog Comparator | 27 |
| DAC – Digital-to-Analog Converter | 28 |
| SDADC – Sigma-Delta Analog-to-Digital Converter 1 | 29 |
| PTC – Peripheral Touch Controller | 30 |
| Reserved | 31 |

**Table 10-4.** Interrupt Line Mapping, SAM C20

| Peripheral Source | NVIC Line |
|---|---|
| EIC NMI – External Interrupt Controller | NMI |
| MCLK - Main Clock<br>OSCCTRL - Oscillators Controller<br>OSC32KCTRL - 32.768 kHz Oscillators Controller<br>SUPC - Supply Controller<br>PAC - Protection Access Controller | 0 |
| WDT – Watchdog Timer | 1 |
| RTC – Real Time Clock | 2 |
| EIC – External Interrupt Controller | 3 |
| FREQM – Frequency Meter | 4 |
| Reserved | 5 |
| NVMCTRL – Non-Volatile Memory Controller | 6 |
| DMAC - Direct Memory Access Controller | 7 |
| EVSYS – Event System | 8 |
| SERCOM0 – Serial Communication Controller 0<br>SERCOM6 – Serial Communication Controller 6 (only for SAM C20 N) | 9 |
| SERCOM1 – Serial Communication Controller 1<br>SERCOM7 – Serial Communication Controller 7 (only for SAM C20 N) | 10 |
| SERCOM2 – Serial Communication Controller 2 | 11 |
| SERCOM3 – Serial Communication Controller 3 | 12 |
| SERCOM4 – Serial Communication Controller 4 (only for SAM C20 N) | 13 |
| SERCOM5 – Serial Communication Controller 5 (only for SAM C20 N) | 14 |
| Reserved | 15 |
| Reserved | 16 |
| TCC0 – Timer Counter for Control 0 | 17 |
| TCC1 – Timer Counter for Control 1 | 18 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 10-4.** Interrupt Line Mapping, SAM C20 (continued)

| Peripheral Source | NVIC Line |
|---|---|
| TCC2 – Timer Counter for Control 2 | 19 |
| TC0 – Timer Counter 0<br>TC5 – Timer Counter 5 (only for SAM C20 N) | 20 |
| TC1 – Timer Counter 1<br>TC6 – Timer Counter 6 (only for SAM C20 N) | 21 |
| TC2 – Timer Counter 2<br>TC7 – Timer Counter 7 (only for SAM C20 N) | 22 |
| TC3 – Timer Counter 3 | 23 |
| TC4 – Timer Counter 4 | 24 |
| ADC0 – Analog-to-Digital Converter 0 | 25 |
| Reserved | 26 |
| AC – Analog Comparator | 27 |
| Reserved | 28 |
| Reserved | 29 |
| PTC – Peripheral Touch Controller | 30 |
| Reserved | 31 |

## 10.3. Micro Trace Buffer

### 10.3.1. Features

- Program flow tracing for the Cortex-M0+ processor
- MTB SRAM can be used for both trace and general purpose storage by the processor
- The position and size of the trace buffer in SRAM is configurable by software
- CoreSight compliant

### 10.3.2. Overview

When enabled, the MTB records changes in program flow, reported by the Cortex-M0+ processor over the execution trace interface shared between the Cortex-M0+ processor and the CoreSight MTB-M0+. This information is stored as trace packets in the SRAM by the MTB. An off-chip debugger can extract the trace information using the Debug Access Port to read the trace information from the SRAM. The debugger can then reconstruct the program flow from this information.

The MTB simultaneously stores trace information into the SRAM, and gives the processor access to the SRAM. The MTB ensures that trace write accesses have priority over processor accesses.

The execution trace packet consists of a pair of 32-bit words that the MTB generates when it detects the processor PC value changes non-sequentially. A non-sequential PC change can occur during branch instructions or during exception entry. See the CoreSight MTB-M0+ Technical Reference Manual for more details on the MTB execution trace packet format.

Tracing is enabled when the MASTER.EN bit in the Host Trace Control Register is 1. There are various ways to set the bit to 1 to start tracing, or to 0 to stop tracing. See the CoreSight Cortex-M0+ Technical Reference Manual for more details on the Trace start and stop and for a detailed description of the MTB's MASTER register. The MTB can be programmed to stop tracing automatically when the memory fills to a specified watermark level or to start or stop tracing by writing directly to the MASTER.EN bit. If the watermark mechanism is not being used and the trace buffer overflows, then the buffer wraps around overwriting previous trace packets.


© 2026 Microchip Technology Inc. and its subsidiaries

The base address of the MTB registers is 0x41008000; this address is also written in the CoreSight ROM Table. The offset of each register from the base address is fixed and as defined by the CoreSight MTB-M0+ Technical Reference Manual. The MTB has 4 programmable registers to control the behavior of the trace features:

- POSITION: Contains the trace write pointer and the wrap bit,
- MASTER: Contains the main trace enable bit and other trace control fields,
- FLOW: Contains the WATERMARK address and the AUTOSTOP and AUTOHALT control bits,
- BASE: Indicates where the SRAM is located in the processor memory map. This register is provided to enable auto discovery of the MTB SRAM location, by a debug agent.

See the CoreSight MTB-M0+ Technical Reference Manual for a detailed description of these registers.

## 10.4. High-Speed Bus System

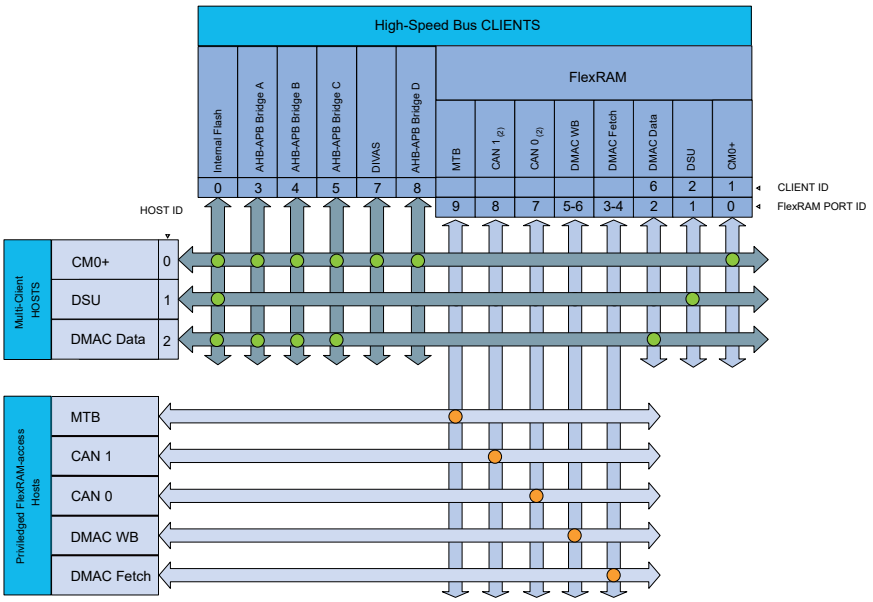
### 10.4.1. Features

High-Speed Bus Matrix has the following features:

- Symmetric crossbar bus switch implementation
- Allows concurrent accesses from different hosts to different client
- 32-bit data bus
- Operation at a 1-to-1 clock frequency with the bus hosts

### 10.4.2. Configuration

**Figure 10-1.** Host-Client Relation High-Speed Bus Matrix, SAM C20/C21




1. The AHB-APB bridge D is available only on SAM C20/C21 N.
2. The CAN peripheral is available only on C21.

**Table 10-5.** Bus Matrix Hosts

| Bus Matrix Hosts | Host ID |
|---|---|
| CM0+ - Cortex M0+ Processor | 0 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 10-5.** Bus Matrix Hosts (continued)

| Bus Matrix Hosts | Host ID |
|---|---|
| DSU - Device Service Unit | 1 |
| DMAC - Direct Memory Access Controller / Data Access | 2 |

**Table 10-6.** Bus Matrix Clients

| Bus Matrix Clients | Client ID |
|---|---|
| Internal Flash Memory | 0 |
| SRAM Port 4 - CM0+ Access | 1 |
| SRAM Port 6 - DSU Access | 2 |
| AHB-APB Bridge A | 3 |
| AHB-APB Bridge B | 4 |
| AHB-APB Bridge C | 5 |
| SRAM Port 5 - DMAC Data Access | 6 |
| DIVAS - Divide Accelerator | 7 |
| AHB-APB Bridge D | 8 |

**Table 10-7.** SRAM Port Connections

| SRAM Port Connection | Port ID | Connection Type |
|---|---|---|
| CM0+ - Cortex M0+ Processor | 0 | Bus Matrix |
| DSU - Device Service Unit | 1 | Bus Matrix |
| DMAC - Direct Memory Access Controller - Data Access | 2 | Bus Matrix |
| DMAC - Direct Memory Access Controller - Fetch Access 0 | 3 | Direct |
| DMAC - Direct Memory Access Controller - Fetch Access 1 | 4 | Direct |
| DMAC - Direct Memory Access Controller - Write-Back Access 0 | 5 | Direct |
| DMAC - Direct Memory Access Controller - Write-Back Access 1 | 6 | Direct |
| CAN0 - Controller Area Network 0 | 7 | Direct |
| CAN1 - Controller Area Network 1 | 8 | Direct |
| MTB - Micro Trace Buffer | 9 | Direct |

### 10.4.3. SRAM Quality of Service

To ensure that Hosts with latency requirements get sufficient priority when accessing RAM, the different Hosts can be configured to have a given priority for different type of access.

The Quality of Service (QoS) level is independently selected for each Host accessing the RAM. For any access to the RAM, the RAM also receives the QoS level. The QoS levels and their corresponding bit values for the QoS level configuration is shown in below.

**Table 10-8.** Quality of Service Level Configuration

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Background (no sensitive operation) |
| 0x1 | LOW | Sensitive Bandwidth |
| 0x2 | MEDIUM | Sensitive Latency |
| 0x3 | HIGH | Critical Latency |

If a Host is configured with QoS level DISABLE (0x0) or LOW (0x1), there will be minimum latency of one cycle for the RAM access.

The priority order for concurrent accesses are decided by two factors: first the QoS level for the Host and second a static priority as shown in the Table 10-7. The lowest port ID has the highest static priority.


© 2026 Microchip Technology Inc. and its subsidiaries

The MTB has fixed QoS level HIGH (0x3) and the DSU has fixed QoS level LOW (0x1).

The CPU QoS level can be written/read at address 0x4100A114, bits [1:0]. Its reset value is 0x3. (The APBB.HMATRIXHS bit need to be set first.)

Refer to different Host QOSCTRL registers for configuring QoS for the other Hosts (for SAM C21: CAN, DMAC; for SAM C20: DMAC).


© 2026 Microchip Technology Inc. and its subsidiaries

# 11. PAC - Peripheral Access Controller

## 11.1. Overview

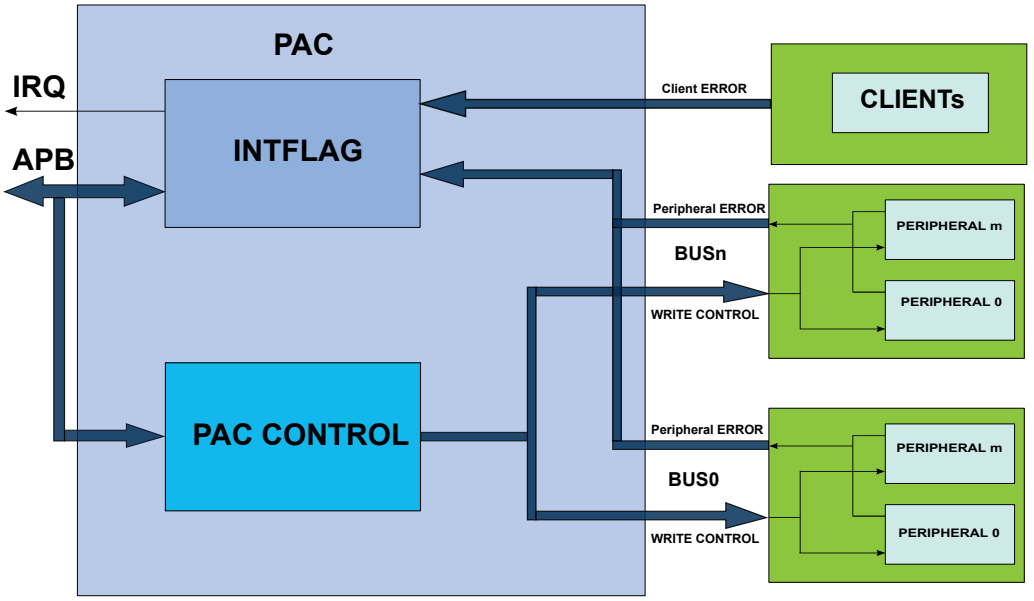
The Peripheral Access Controller provides an interface for the locking and unlocking of peripheral registers within the device. It reports all violations that could happen when accessing a peripheral: write protected access, illegal access, enable protected access, access when clock synchronization or software reset is on-going. These errors are reported in a unique interrupt flag for a peripheral. The PAC module also reports errors occurring at the client bus level, when an access to a non-existing address is detected.

## 11.2. Features

- Manages write protection access and reports access errors for the peripheral modules or bridges.

## 11.3. Block Diagram

**Figure 11-1.** PAC Block Diagram




## 11.4. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 11.4.1. IO Lines

Not applicable.

### 11.4.2. Power Management

The PAC can continue to operate in any Sleep mode where the selected source clock is running. The PAC interrupts can be used to wake up the device from Sleep modes. The events can trigger other operations in the system without exiting sleep modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.4.3. Clocks

The PAC bus clock (CLK_PAC_APB) can be enabled and disabled in the Main Clock module. The default state of CLK_PAC_APB can be found in the related links.

**Related Links**
MCLK – Main Clock
Peripheral Clock Masking

### 11.4.4. DMA

Not applicable.

### 11.4.5. Interrupts

The interrupt request line is connected to the Interrupt Controller (NVIC). Using the PAC interrupt requires the Interrupt Controller to be configured first.

**Table 11-1.** Interrupt Lines

| Instances | NVIC Line |
|---|---|
| PAC | PACERR |

**Related Links**
Nested Vector Interrupt Controller

### 11.4.6. Events

The events are connected to the Event System, which may need configuration.

**Related Links**
Event System (EVSYS)

### 11.4.7. Debug Operation

When the CPU is halted in debug mode, write protection of all peripherals is disabled and the PAC continues normal operation.

### 11.4.8. Register Access Protection

All registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following PAC registers:

- Write Control (WRCTRL) register
- AHB Client Bus Interrupt Flag Status and Clear (INTFLAGAHB) register
- Peripheral Interrupt Flag Status and Clear n (INTFLAG A/B/C...) registers

Optional write protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write Protection" property in each individual register description.

PAC write protection does not apply to accesses through an external debugger.

## 11.5. Functional Description

### 11.5.1. Principle of Operation

The Peripheral Access Control module allows the user to set a write protection on peripheral modules and generate an interrupt in case of a peripheral access violation. The peripheral's protection can be set, cleared or locked at the user discretion. A set of Interrupt Flag and Status registers informs the user on the status of the violation in the peripherals. In addition, client bus errors can be also reported in the cases where reserved area is accessed by the application.


© 2026 Microchip Technology Inc. and its subsidiaries

## 11.5.2. Basic Operation

### 11.5.2.1. Initialization

After reset, the PAC is enabled.

### 11.5.2.2. Initialization, Enabling and Resetting

The PAC is always enabled after reset.

Only a hardware reset will reset the PAC module.

### 11.5.2.3. Operations

The PAC module allows the user to set, clear or lock the write protection status of all peripherals on all Peripheral Bridges.

If a peripheral register violation occurs, the Peripheral Interrupt Flag n registers (INTFLAGn) are updated to inform the user on the status of the violation in the peripherals connected to the Peripheral Bridge n (n = A,B,C ...). The corresponding Peripheral Write Control Status n register (STATUSn) gives the state of the write protection for all peripherals connected to the corresponding Peripheral Bridge n. Refer to Peripheral Access Errors for details.

The PAC module also report the errors occurring at client bus level when an access to reserved area is detected. AHB Client Bus Interrupt Flag register (INTFLAGAHB) informs the user on the status of the violation in the corresponding client. Refer to the AHB Client Bus Errors for details.

### 11.5.2.4. Peripheral Access Errors

The following events will generate a Peripheral Access Error:

- Protected write: To avoid unexpected writes to a peripheral's registers, each peripheral can be write protected. Only the registers denoted as “PAC Write-Protection” in the module’s datasheet can be protected. If a peripheral is not write protected, write data accesses are performed normally. If a peripheral is write protected and if a write access is attempted, data will not be written and peripheral returns an access error. The corresponding interrupt flag bit in the INTFLAGn register will be set.
- Illegal access: Access to an unimplemented register within the module.
- Synchronized write error: For write-synchronized registers an error will be reported if the register is written while a synchronization is ongoing.

When any of the INTFLAGn registers bit are set, an interrupt will be requested if the PAC interrupt enable bit is set.

### 11.5.2.5. Write Access Protection Management

Peripheral access control can be enabled or disabled by writing to the WRCTRL register.

The data written to the WRCTRL register is composed of two fields; WRCTRL.PERID and WRCTRL.KEY. The WRCTRL.PERID is an unique identifier corresponding to a peripheral. The WRCTRL.KEY is a key value that defines the operation to be done on the control access bit. These operations can be “clear protection”, “set protection” and “set and lock protection bit”.

The “clear protection” operation will remove the write access protection for the peripheral selected by WRCTRL.PERID. Write accesses are allowed for the registers in this peripheral.

The “set protection” operation will set the write access protection for the peripheral selected by WRCTRL.PERID. Write accesses are not allowed for the registers with write protection property in this peripheral.

The “set and lock protection” operation will set the write access protection for the peripheral selected by WRCTRL.PERID and locks the access rights of the selected peripheral registers. The write access protection will only be cleared by a hardware reset.

The peripheral access control status can be read from the corresponding STATUSn register.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 11.5.2.6. Write Access Protection Management Errors

Only word-wise writes to the WRCTRL register will effectively change the access protection. Other type of accesses will have no effect and will cause a PAC write access error. This error is reported in the INTFLAGA.PAC bit.

PAC also offers an additional safety feature for correct program execution with an interrupt generated on double write clear protection or double write set protection. If a peripheral is write protected and a subsequent set protection operation is detected then the PAC returns an error, and similarly for a double clear protection operation.

In addition, an error is generated when writing a “set and lock” protection to a write-protected peripheral or when a write access is done to a locked set protection. This can be used to ensure that the application follows the intended program flow by always following a write protect with an unprotect and conversely. However in applications where a write protected peripheral is used in several contexts, e.g. interrupt, care should be taken so that either the interrupt can not happen while the main application or other interrupt levels manipulates the write protection status or when the interrupt handler needs to unprotect the peripheral based on the current protection status by reading the STATUS register.

The errors generated while accessing the PAC module registers (eg. key error, double protect error...) will set the INTFLAGA.PAC flag.

#### 11.5.2.7. AHB Client Bus Errors

The PAC module reports errors occurring at the AHB Client bus level. These errors are generated when an access is performed at an address where no client (bridge or peripheral) is mapped . These errors are reported in the corresponding bits of the INTFLAGAHB register.

#### 11.5.2.8. Generating Events

The PAC module can also generate an event when any of the Interrupt Flag registers bit are set. To enable the PAC event generation, the control bit EVCTRL.ERREO must be set a '1'.

### 11.5.3. DMA Operation

Not applicable.

### 11.5.4. Interrupts

The PAC has the following interrupt source:

- Error (ERR): Indicates that a peripheral access violation occurred in one of the peripherals controlled by the PAC module, or a bridge error occurred in one of the bridges reported by the PAC
  - This interrupt is a synchronous wake-up source.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAGAHB and INTFLAGn) registers is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the PAC is reset. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAGAHB and INTFLAGn registers to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Nested Vector Interrupt Controller
Sleep Mode Controller

### 11.5.5. Events

The PAC can generate the following output event:

- Error (ERR): Generated when one of the interrupt flag registers bits is set

Writing a '1' to an Event Output bit in the Event Control Register (EVCTRL.ERREO) enables the corresponding output event. Writing a '0' to this bit disables the corresponding output event.

### 11.5.6. Sleep Mode Operation

In Sleep mode, the PAC is kept enabled if an available bus host (CPU, DMA) is running. The PAC will continue to catch access errors from the module and generate interrupts or events.

### 11.5.7. Synchronization

Not applicable.


© 2026 Microchip Technology Inc. and its subsidiaries

## 11.6. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | WRCTRL | 7:0 | PERID[7:0] | | | | | | | |
| | | 15:8 | PERID[15:8] | | | | | | | |
| | | 23:16 | KEY[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | EVCTRL | 7:0 | | | | | | | | ERREO |
| 0x05 ... 0x07 | Reserved | | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | | | | | | | | ERR |
| 0x09 | INTENSET | 7:0 | | | | | | | | ERR |
| 0x0A ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | INTFLAGAHB | 7:0 | DIVAS | LPRAMDMAC | HPB2 | HPB0 | HPB1 | HSRAMDSU | HSRAMCM0P | FLASH |
| | | 15:8 | | | | | | | | HPB3 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | INTFLAGA | 7:0 | GCLK | SUPC | OSC32KCTRL | OSCCTRL | RSTC | MCLK | PM | PAC |
| | | 15:8 | | | | TSENS | FREQM | EIC | RTC | WDT |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | INTFLAGB | 7:0 | | | | MTB | DMAC | NVMCTRL | DSU | PORT |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | INTFLAGC | 7:0 | CAN0 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| | | 15:8 | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | CAN1 |
| | | 23:16 | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| | | 31:24 | | | | | | | | |
| 0x20 | INTFLAGD | 7:0 | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 ... 0x33 | Reserved | | | | | | | | | |
| 0x34 | STATUSA | 7:0 | GCLK | SUPC | OSC32KCTRL | OSCCTRL | | MCLK | PM | PAC |
| | | 15:8 | | | | TSENS | FREQM | EIC | RTC | WDT |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x38 | STATUSB | 7:0 | | | | MTB | DMAC | NVMCTRL | DSU | PORT |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x3C | STATUSC | 7:0 | CAN0 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| | | 15:8 | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | CAN1 |
| | | 23:16 | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| | | 31:24 | | | | | | | | |
| 0x40 | STATUSD | 7:0 | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 11.7. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to the related links.


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.1. Write Control

**Name:** WRCTRL
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | KEY[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PERID[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PERID[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:16 – KEY[7:0]** Peripheral Access Control Key
These bits define the peripheral access control key:

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | No action |
| 0x1 | CLEAR | Clear the peripheral write control |
| 0x2 | SET | Set the peripheral write control |
| 0x3 | LOCK | Set and lock the peripheral write control until the next hardware reset |

**Bits 15:0 – PERID[15:0]** Peripheral Identifier
The PERID represents the peripheral whose control is changed using the WRCTRL.KEY. The Peripheral Identifier is calculated following formula:
$PERID = 32^{*}\,\mathrm{BridgeNumber} + \mathrm{N}$

Where BridgeNumber represents the Peripheral Bridge Number (0 for Peripheral Bridge A, 1 for Peripheral Bridge B, etc). N represents the peripheral index from the respective Bridge Number, which can be found in the tables of Section 12 Peripheral Configuration Summary in the "PAC, Index" column.

**Table 11-2.** PERID Values

| Periph. Bridge Name | BridgeNumber | PERID Values |
|---|---|---|
| A | 0 | 0+N |
| B | 1 | 32+N |
| C | 2 | 64+N |
| D | 3 | 96+N |


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.2. Event Control

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | ERREO |
| Access | | | | | | | | RW |
| Reset | | | | | | | | 0 |

**Bit 0 – ERREO** Peripheral Access Error Event Output
This bit indicates if the Peripheral Access Error Event Output is enabled or disabled. When enabled, an event will be generated when one of the interrupt flag registers bits (INTFLAGAHB, INTFLAGn) is set:

| Value | Description |
|---|---|
| 0 | Peripheral Access Error Event Output is disabled. |
| 1 | Peripheral Access Error Event Output is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.3. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | ERR |
| Access | | | | | | | | RW |
| Reset | | | | | | | | 0 |

**Bit 0 – ERR** Peripheral Access Error Interrupt Disable
This bit indicates that the Peripheral Access Error Interrupt is disabled and an interrupt request will be generated when one of the interrupt flag registers bits (INTFLAGAHB, INTFLAGn) is set:
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Peripheral Access Error interrupt Enable bit and disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | Peripheral Access Error interrupt is disabled. |
| 1 | Peripheral Access Error interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.4. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | ERR |
| Access | | | | | | | | RW |
| Reset | | | | | | | | 0 |

**Bit 0 – ERR** Peripheral Access Error Interrupt Enable
This bit indicates that the Peripheral Access Error Interrupt is enabled and an interrupt request will be generated when one of the interrupt flag registers bits (INTFLAGAHB, INTFLAGn) is set:
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Peripheral Access Error interrupt Enable bit and enables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | Peripheral Access Error interrupt is disabled. |
| 1 | Peripheral Access Error interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.5. AHB Client Bus Interrupt Flag Status and Clear

**Name:** INTFLAGAHB
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** –

This flag is cleared by writing a '1' to the flag.

This flag is set when an access error is detected by the CLIENT n, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the corresponding INTFLAGAHB interrupt flag.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIVAS | LPRAMDMAC | HPB2 | HPB0 | HPB1 | HSRAMDSU | HSRAMCM0P | FLASH |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 8 – HPB3** Interrupt Flag for CLIENT APBD

**Bit 7 – DIVAS** Interrupt Flag for CLIENT DIVAS

**Bit 6 – LPRAMDMAC** Interrupt Flag for CLIENT LPRAMDMAC

**Bit 5 – HPB2** Interrupt Flag for CLIENT APBC

**Bit 4 – HPB0** Interrupt Flag for CLIENT APBA

**Bit 3 – HPB1** Interrupt Flag for CLIENT APBB

**Bit 2 – HSRAMDSU** Interrupt Flag for CLIENT HSRAMDSU

**Bit 1 – HSRAMCM0P** Interrupt Flag for CLIENT HSRAMCM0P

**Bit 0 – FLASH** Interrupt Flag for CLIENT FLASH


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.6. Peripheral Interrupt Flag Status and Clear A

**Name:** INTFLAGA
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** –

This flag is cleared by writing a one to the flag.

This flag is set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGA bit, and will generate an interrupt request if INTENCLR/SET.ERR is one.

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the corresponding INTFLAGA interrupt flag.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | TSENS | FREQM | EIC | RTC | WDT |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GCLK | SUPC | OSC32KCTRL | OSCCTRL | RSTC | MCLK | PM | PAC |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 12 – TSENS** Interrupt Flag for TSENS

**Bit 11 – FREQM** Interrupt Flag for FREQM

**Bit 10 – EIC** Interrupt Flag for EIC

**Bit 9 – RTC** Interrupt Flag for RTC

**Bit 8 – WDT** Interrupt Flag for WDT

**Bit 7 – GCLK** Interrupt Flag for GCLK

**Bit 6 – SUPC** Interrupt Flag for SUPC

**Bit 5 – OSC32KCTRL** Interrupt Flag for OSC32KCTRL

**Bit 4 – OSCCTRL** Interrupt Flag for OSCCTRL

**Bit 3 – RSTC** Interrupt Flag for RSTC


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 2 – MCLK** Interrupt Flag for MCLK

**Bit 1 – PM** Interrupt Flag for PM

**Bit 0 – PAC** Interrupt Flag for PAC


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.7. Peripheral Interrupt Flag Status and Clear B

**Name:** INTFLAGB
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** –

This flag is cleared by writing a '1' to the flag.

This flag is set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGB bit, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the corresponding INTFLAGB interrupt flag.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | MTB | DMAC | NVMCTRL | DSU | PORT |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bit 4 – MTB** Interrupt Flag for MTB

**Bit 3 – DMAC** Interrupt Flag for DMAC

**Bit 2 – NVMCTRL** Interrupt Flag for NVMCTRL

**Bit 1 – DSU** Interrupt Flag for DSU

**Bit 0 – PORT** Interrupt Flag for PORT


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.8. Peripheral Interrupt Flag Status and Clear C

**Name:** INTFLAGC
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** –

This flag is cleared by writing a one to the flag.

This flag is set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGC bit, and will generate an interrupt request if INTENCLR/SET.ERR is one.

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the corresponding INTFLAGC interrupt flag.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | CAN1 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CAN0 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 23 – CCL** Interrupt Flag for CCL

**Bit 22 – PTC** Interrupt Flag for PTC

**Bit 21 – DAC** Interrupt Flag for DAC

**Bit 20 – AC** Interrupt Flag for AC

**Bit 19 – SDADC** Interrupt Flag for SDADC

**Bits 17, 18 – ADC** Interrupt Flag for ADCn [n=1..0]

**Bits 12, 13, 14, 15, 16 – TC** Interrupt Flag for TCn [n = 4..0]

**Bits 9, 10, 11 – TCC** Interrupt Flag for TCCn [n = 2..0]

**Bits 7, 8 – CAN** Interrupt Flag for CAN

**Bits 1, 2, 3, 4, 5, 6 – SERCOM** Interrupt Flag for SERCOMn [n = 5..0]


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 0 – EVSYS** Interrupt Flag for EVSYS


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.9. Peripheral Interrupt Flag Status and Clear D

**Name:** INTFLAGD
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** –

This flag is cleared by writing a one to the flag.

This flag is set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGD bit, and will generate an interrupt request if INTENCLR/SET.ERR is one.

Writing a zero to this bit has no effect.

Writing a one to this bit will clear the corresponding INTFLAGD interrupt flag.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 2, 3, 4 – TC** Interrupt Flag for TCn [n = 7..5]

**Bits 0, 1 – SERCOM** Interrupt Flag for SERCOMn [n = 7..6]


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.10. Peripheral Write Protection Status A

**Name:** STATUSA
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** –

Writing to this register has no effect.

Reading STATUS register returns peripheral write protection status:

| Value | Description |
|---|---|
| 0 | Peripheral is not write protected. |
| 1 | Peripheral is write protected. |

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | TSENS | FREQM | EIC | RTC | WDT |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GCLK | SUPC | OSC32KCTRL | OSCCTRL | | MCLK | PM | PAC |
| Access | R | R | R | R | | R | R | R |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bit 12 – TSENS** Peripheral TSENS Write Protection Status

**Bit 11 – FREQM** Peripheral FREQM Write Protection Status

**Bit 10 – EIC** Peripheral EIC Write Protection Status

**Bit 9 – RTC** Peripheral RTC Write Protection Status

**Bit 8 – WDT** Peripheral WDT Write Protection Status

**Bit 7 – GCLK** Peripheral GCLK Write Protection Status

**Bit 6 – SUPC** Peripheral SUPC Write Protection Status

**Bit 5 – OSC32KCTRL** Peripheral OSC32KCTRL Write Protection Status

**Bit 4 – OSCCTRL** Peripheral OSCCTRL Write Protection Status

**Bit 2 – MCLK** Peripheral MCLK Write Protection Status


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – PM** Peripheral PM Write Protection Status

**Bit 0 – PAC** Peripheral PAC Write Protection Status


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.11. Peripheral Write Protection Status B

**Name:** STATUSB
**Offset:** 0x38
**Reset:** 0x00000002
**Property:** –

Writing to this register has no effect.

Reading the STATUS register returns peripheral write protection status:

| Value | Description |
|---|---|
| 0 | Peripheral is not write protected. |
| 1 | Peripheral is write protected. |

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | MTB | DMAC | NVMCTRL | DSU | PORT |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 1 | 0 |

**Bit 4 – MTB** Peripheral MTB Write Protection Status

**Bit 3 – DMAC** Peripheral DMAC Write Protection Status

**Bit 2 – NVMCTRL** Peripheral NVMCTRL Write Protection Status

**Bit 1 – DSU** Peripheral DSU Write Protection Status

**Bit 0 – PORT** Peripheral PORT Write Protection Status


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.12. Peripheral Write Protection Status C

**Name:** STATUSC
**Offset:** 0x3C
**Reset:** 0x00000000
**Property:** –

Writing to this register has no effect.

Reading STATUS register returns peripheral write protection status:

| Value | Description |
|---|---|
| 0 | Peripheral is not write protected. |
| 1 | Peripheral is write protected. |

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | CAN1 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CAN0 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 23 – CCL** Peripheral CCL Write Protection Status

**Bit 22 – PTC** Peripheral PTC Write Protection Status

**Bit 21 – DAC** Peripheral DAC Write Protection Status

**Bit 20 – AC** Peripheral AC Write Protection Status

**Bit 19 – SDADC** Peripheral SDADC Write Protection Status

**Bits 17, 18 – ADC** Peripheral ADCn [n=1..0] Write Protection Status

**Bits 12, 13, 14, 15, 16 – TC** Peripheral TCn Write Protection Status [n = 4..0]

**Bits 9, 10, 11 – TCC** Peripheral TCCn [n = 2..0] Write Protection Status TCCn [n = 2..0]

**Bits 7, 8 – CAN** Peripheral CAN Write Protection Status

**Bits 1, 2, 3, 4, 5, 6 – SERCOM** Peripheral SERCOMn Write Protection Status [n = 5..0]


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 0 – EVSYS** Peripheral EVSYS Write Protection Status


© 2026 Microchip Technology Inc. and its subsidiaries

### 11.7.13. Peripheral Write Protection Status D

**Name:** STATUSD
**Offset:** 0x40
**Reset:** 0x00000000
**Property:** –

Writing to this register has no effect.

Reading STATUS register returns peripheral write protection status:

| Value | Description |
|---|---|
| 0 | Peripheral is not write protected. |
| 1 | Peripheral is write protected. |

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 2, 3, 4 – TC** Peripheral TCn Write Protection Status [n = 7..5]

**Bits 0, 1 – SERCOM** Peripheral SERCOMn Write Protection Status [n = 7..6]


© 2026 Microchip Technology Inc. and its subsidiaries

# 12. Peripherals Configuration Summary

## 12.1. SAM C20/C21 N

**Table 12-1.** Peripherals Configuration Summary SAM C21 N

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| AHB-APB Bridge A | 0x40000000 | | 0 | Y | | | | | | | | | N/A |
| PAC | 0x40000000 | 0 | 10 | Y | 0 | Y | | 0 | N | | 86 : ACCERR | | N/A |
| PM | 0x40000400 | 0 | | | 1 | Y | | 1 | N | | | | N/A |
| MCLK | 0x40000800 | 0 | | | 2 | Y | | 2 | N | | | | Y |
| RSTC | 0x40000C00 | | | | 3 | Y | | 3 | N | | | | N/A |
| OSCCTRL | 0x40001000 | 0 | | | 4 | Y | 0: FDPLL96M clk source<br>1: FDPLL96M 32kHz | 4 | N | | 1: XOSC_FAIL | | Y |
| OSC32KCTRL | 0x40001400 | 0 | | | 5 | Y | | 5 | N | | 2: XOSC32K_FAIL | | Y |
| SUPC | 0x40001800 | 0 | | | 6 | Y | | 6 | N | | | | N/A |
| GCLK | 0x40001C00 | | | | 7 | Y | | 7 | N | | | | N/A |
| WDT | 0x40002000 | 1 | | | 8 | Y | | 8 | N | | | | Y |
| RTC | 0x40002400 | 2 | | | 9 | Y | | 9 | N | | 3: CMP0/ALARM0<br>4: CMP1<br>5: OVF5-1<br>6:13: PER0-7 | | Y |
| EIC | 0x40002800 | 3, NMI | | | 10 | Y | 2 | 10 | N | | 14-29: EXTINT0-16 | | Y |
| FREQM | 0x40002C00 | 4 | | | 11 | Y | 3: Measure<br>4: Reference | 11 | N | | | | N/A |
| TSENS | 0x40003000 | 5 | | | 12 | N | 5 | 12 | N | 0: START | 30: WINMON | 1: RESRDY | A |
| AHB-APB Bridge B | 0x41000000 | | 1 | Y | | | | | | | | | N/A |
| PORT | 0x41000000 | | | | 0 | Y | | 0 | N | 1-4 : EV0-3 | | | Y |
| DSU | 0x41002000 | | 3 | Y | 1 | Y | | 1 | Y | | | | N/A |
| NVMCTRL | 0x41004000 | 6 | 5 | Y | 2 | Y | 39 | 2 | N | | | | Y |
| DMAC | 0x41006000 | 7 | 7 | Y | | | | 3 | N | 5-8: CH0-3 | 31-34: CH0-3 | | Y |
| MTB | 0x41008000 | | | | | | | | N | 45: START<br>46: STOP | | | N/A |
| AHB-APB Bridge C | 0x42000000 | | 2 | Y | | | | | | | | | N/A |
| EVSYS | 0x42000000 | 8 | | | 0 | N | 6-17: one per CHANNEL | 0 | N | | | | Y |
| SERCOM0 | 0x42000400 | 9 | | | 1 | N | 19: CORE<br>18: SLOW | 1 | N | | | 2: RX<br>3: TX | Y |
| SERCOM1 | 0x42000800 | 10 | | | 2 | N | 20: CORE<br>18: SLOW | 2 | N | | | 4: RX<br>5: TX | Y |
| SERCOM2 | 0x42000C00 | 11 | | | 3 | N | 21: CORE<br>18: SLOW | 3 | N | | | 6: RX<br>7: TX | Y |
| SERCOM3 | 0x42001000 | 12 | | | 4 | N | 22: CORE<br>18: SLOW | 4 | N | | | 8: RX<br>9: TX | Y |
| SERCOM4 | 0x42001400 | 13 | | | 5 | N | 23: CORE<br>18: SLOW | 5 | N | | | 10: RX<br>11: TX | Y |
| SERCOM5 | 0x42001800 | 14 | | | 6 | N | 25: CORE<br>24: SLOW | 6 | N | | | 12: RX<br>13: TX | Y |
| CAN0 | 0x42001C00 | 15 | 8 | N | | | 26 | 7 | | | | 14: DEBUG | N/A |
| CAN1 | 0x42002000 | 16 | 9 | N | | | 27 | 8 | | | | 15: DEBUG | N/A |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-1.** Peripherals Configuration Summary SAM C21 N (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| TCC0 | 0x42002400 | 17 | | | 9 | N | 28 | 9 | N | 9-10: EV0-1<br>11-14: MC0-3 | 35: OVF<br>36: TRG<br>37: CNT<br>38-41: MC0-3 | 16: OVF<br>17-20: MC0-3 | Y |
| TCC1 | 0x42002800 | 18 | | | 10 | N | 28 | 10 | N | 15-16: EV0-1<br>17-18: MC0-1 | 42: OVF<br>43: TRG<br>44: CNT<br>45-46: MC0-1 | 21: OVF<br>22-23: MC0-1 | Y |
| TCC2 | 0x42002C00 | 19 | | | 11 | N | 29 | 11 | N | 19-20: EV0-1<br>21-22: MC0-1 | 47: OVF<br>48: TRG<br>49: CNT<br>50-51: MC0-1 | 24: OVF<br>25-26: MC0-1 | Y |
| TC0 | 0x42003000 | 20 | | | 12 | N | 30 | 12 | N | 23: EVU | 52: OVF<br>53-54: MC0-1 | 27: OVF<br>28-29: MC0-1 | Y |
| TC1 | 0x42003400 | 21 | | | 13 | N | 30 | 13 | N | 24: EVU | 55: OVF<br>56-57: MC0-1 | 30: OVF<br>31-32: MC0-1 | Y |
| TC2 | 0x42003800 | 22 | | | 14 | N | 31 | 14 | N | 25: EVU | 58: OVF<br>59-60: MC0-1 | 33: OVF<br>34-35: MC0-1 | Y |
| TC3 | 0x42003C00 | 23 | | | 15 | N | 31 | 15 | N | 26: EVU | 61: OVF<br>62-63: MC0-1 | 36: OVF<br>37-38: MC0-1 | Y |
| TC4 | 0x42004000 | 24 | | | 16 | N | 32 | 16 | N | 27: EVU | 64: OVF<br>65-66: MC0-1 | 39: OVF<br>40-41: MC0-1 | Y |
| ADC0 | 0x42004400 | 25 | | | 17 | N | 33 | 17 | N | 28: START<br>29: SYNC | 67: RESRDY<br>68: WINMON | 42: RESRDY | Y |
| ADC1 | 0x42004800 | 26 | | | 18 | N | 34 | 18 | N | 30: START<br>31: SYNC | 69: RESRDY<br>70: WINMON | 43: RESRDY | Y |
| SDADC | 0x42004C00 | 29 | | | 19 | N | 35 | 19 | N | 32: START<br>33: FLUSH | 71: RESRDY<br>72: WINMON | 44: RESRDY | Y |
| AC | 0x42005000 | 27 | | | 20 | N | 40 | 20 | N | 34-37: SOC0-3 | 73-76: COMP0-3<br>77-78: WIN0-1 | | Y |
| DAC | 0x42005400 | 28 | | | 21 | N | 36 | 21 | N | 38: START | 79: EMPTY | 45: EMPTY | Y |
| PTC | 0x42005800 | 30 | | | 22 | N | 37 | 22 | N | 39: STCONV | 80: EOC<br>81: WCOMP | EOC: 46<br>WCOMP: 47<br>SEQ: 48 | |
| CCL | 0x42005C00 | | | | 23 | N | 38 | 23 | N | 40-43 : LUTIN0-3 | 82-85: LUTOUT0-3 | | Y |
| AHB-APB Bridge D | 0x43000000 | | 13 | Y | 0 | | | | | | | | N/A |
| SERCOM6 | 0x43000000 | 9 | | | 0 | N | 41: CORE<br>18: SLOW | 0 | N | | | 49: RX<br>50: TX | Y |
| SERCOM7 | 0x43000400 | 10 | | | 1 | N | 42: CORE<br>18: SLOW | 1 | N | | | 51: RX<br>52: TX | Y |
| TC5 | 0x43000800 | 20 | | | 2 | N | 43 | 2 | N | 47: EVU | 88: OVF<br>89-90: MC0-1 | 53: OVF<br>54-55: MC0-1 | Y |
| TC6 | 0x43000C00 | 21 | | | 3 | N | 44 | 3 | N | 48:EVU | 91: OVF<br>92-93: MC0-1 | 56: OVF<br>57-58: MC0-1 | Y |
| TC7 | 0x43001000 | 22 | | | 4 | N | 45 | 4 | N | 49:EVU | 94: OVF<br>95-96: MC0-1 | 59: OVF<br>60-61: MC0-1 | Y |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-1.** Peripherals Configuration Summary SAM C21 N (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| DIVAS | 0x48000000 | | 12 | Y | | | | | | | | | N/A |

**Table 12-2.** Peripherals Configuration Summary SAM C20 N

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| AHB-APB Bridge A | 0x40000000 | | 0 | Y | | | | | | | | | N/A |
| PAC | 0x40000000 | 0 | 10 | Y | 0 | Y | | 0 | N | | 86 : ACCERR | | N/A |
| PM | 0x40000400 | 0 | | | 1 | Y | | 1 | N | | | | N/A |
| MCLK | 0x40000800 | 0 | | | 2 | Y | | 2 | N | | | | Y |
| RSTC | 0x40000C00 | | | | 3 | Y | | 3 | N | | | | N/A |
| OSCCTRL | 0x40001000 | 0 | | | 4 | Y | 0: FDPLL96M clk source<br>1: FDPLL96M 32kHz | 4 | N | | 1: XOSC_FAIL | | Y |
| OSC32KCTRL | 0x40001400 | 0 | | | 5 | Y | | 5 | N | | 2: XOSC32K_FAIL | | Y |
| SUPC | 0x40001800 | 0 | | | 6 | Y | | 6 | N | | | | N/A |
| GCLK | 0x40001C00 | | | | 7 | Y | | 7 | N | | | | N/A |
| WDT | 0x40002000 | 1 | | | 8 | Y | | 8 | N | | | | Y |
| RTC | 0x40002400 | 2 | | | 9 | Y | | 9 | N | | 3: CMP0/ALARM0<br>4: CMP1<br>5: OVF5-1<br>6:13: PER0-7 | | Y |
| EIC | 0x40002800 | 3, NMI | | | 10 | Y | 2 | 10 | N | | 14-29: EXTINT0-15 | | Y |
| FREQM | 0x40002C00 | 4 | | | 11 | Y | 3: Measure<br>4: Reference | 11 | N | | | | N/A |
| AHB-APB Bridge B | 0x41000000 | | 1 | Y | | | | | | | | | N/A |
| PORT | 0x41000000 | | | | 0 | Y | | 0 | N | 1-4 : EV0-3 | | | Y |
| DSU | 0x41002000 | | 3 | Y | 1 | Y | | 1 | Y | | | | N/A |
| NVMCTRL | 0x41004000 | 6 | 5 | Y | 2 | Y | 39 | 2 | N | | | | Y |
| DMAC | 0x41006000 | 7 | 7 | Y | | | | 3 | N | 5-8: CH0-3 | 31-34: CH0-3 | | Y |
| MTB | 0x41008000 | | | | | | | | N | 45: START<br>46: STOP | | | N/A |
| AHB-APB Bridge C | 0x42000000 | | 2 | Y | | | | | | | | | N/A |
| EVSYS | 0x42000000 | 8 | | | 0 | N | 6-17: one per CHANNEL | 0 | N | | | | Y |
| SERCOM0 | 0x42000400 | 9 | | | 1 | N | 19: CORE<br>18: SLOW | 1 | N | | | 2: RX<br>3: TX | Y |
| SERCOM1 | 0x42000800 | 10 | | | 2 | N | 20: CORE<br>18: SLOW | 2 | N | | | 4: RX<br>5: TX | Y |
| SERCOM2 | 0x42000C00 | 11 | | | 3 | N | 21: CORE<br>18: SLOW | 3 | N | | | 6: RX<br>7: TX | Y |
| SERCOM3 | 0x42001000 | 12 | | | 4 | N | 22: CORE<br>18: SLOW | 4 | N | | | 8: RX<br>9: TX | Y |
| SERCOM4 | 0x42001400 | 13 | | | 5 | N | 23: CORE<br>18: SLOW | 5 | N | | | 10: RX<br>11: TX | Y |
| SERCOM5 | 0x42001800 | 14 | | | 6 | N | 25: CORE<br>24: SLOW | 6 | N | | | 12: RX<br>13: TX | Y |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-2.** Peripherals Configuration Summary SAM C20 N (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| TCC0 | 0x42002400 | 17 | | | 9 | N | 28 | 9 | N | 9-10: EV0-1<br>11-14: MC0-3 | 35: OVF<br>36: TRG<br>37: CNT<br>38-41: MC0-3 | 16: OVF<br>17-20: MC0-3 | Y |
| TCC1 | 0x42002800 | 18 | | | 10 | N | 28 | 10 | N | 15-16: EV0-1<br>17-18: MC0-1 | 42: OVF<br>43: TRG<br>44: CNT<br>45-46: MC0-1 | 21: OVF<br>22-23: MC0-1 | Y |
| TCC2 | 0x42002C00 | 19 | | | 11 | N | 29 | 11 | N | 19-20: EV0-1<br>21-22: MC0-1 | 47: OVF<br>48: TRG<br>49: CNT<br>50-51: MC0-1 | 24: OVF<br>25-26: MC0-1 | Y |
| TC0 | 0x42003000 | 20 | | | 12 | N | 30 | 12 | N | 23: EVU | 52: OVF<br>53-54: MC0-1 | 27: OVF<br>28-29: MC0-1 | Y |
| TC1 | 0x42003400 | 21 | | | 13 | N | 30 | 13 | N | 24: EVU | 55: OVF<br>56-57: MC0-1 | 30: OVF<br>31-32: MC0-1 | Y |
| TC2 | 0x42003800 | 22 | | | 14 | N | 31 | 14 | N | 25: EVU | 58: OVF<br>59-60: MC0-1 | 33: OVF<br>34-35: MC0-1 | Y |
| TC3 | 0x42003C00 | 23 | | | 15 | N | 31 | 15 | N | 26: EVU | 61: OVF<br>62-63: MC0-1 | 36: OVF<br>37-38: MC0-1 | Y |
| TC4 | 0x42004000 | 24 | | | 16 | N | 32 | 16 | N | 27: EVU | 64: OVF<br>65-66: MC0-1 | 39: OVF<br>40-41: MC0-1 | Y |
| ADC0 | 0x42004400 | 25 | | | 17 | N | 33 | 17 | N | 28: START<br>29: SYNC | 67: RESRDY<br>68: WINMON | 42: RESRDY | Y |
| AC | 0x42005000 | 27 | | | 20 | N | 40 | 20 | N | 34-37: SOC0-3 | 73-76: COMP0-3<br>77-78: WIN0-1 | | Y |
| PTC | 0x42005800 | 30 | | | 22 | N | 37 | 22 | N | 39: STCONV | 80: EOC<br>81: WCOMP | EOC: 46<br>WCOMP: 47<br>SEQ: 48 | |
| CCL | 0x42005C00 | | | | 23 | N | 38 | 23 | N | 40-43 : LUTIN0-3 | 82-85: LUTOUT0-3 | | Y |
| AHB-APB Bridge D | 0x43000000 | | 13 | Y | 0 | | | | | | | | N/A |
| SERCOM6 | 0x43000000 | 9 | | | 0 | N | 41: CORE<br>18: SLOW | 0 | N | | | 49: RX<br>50: TX | Y |
| SERCOM7 | 0x43000400 | 10 | | | 1 | N | 42: CORE<br>18: SLOW | 1 | N | | | 51: RX<br>52: TX | Y |
| TC5 | 0x43000800 | 20 | | | 2 | N | 43 | 2 | N | 47: EVU | 88: OVF<br>89-90: MC0-1 | 53: OVF<br>54-55: MC0-1 | Y |
| TC6 | 0x43000C00 | 21 | | | 3 | N | 44 | 3 | N | 48:EVU | 91: OVF<br>92-93: MC0-1 | 56: OVF<br>57-58: MC0-1 | Y |
| TC7 | 0x43001000 | 22 | | | 4 | N | 45 | 4 | N | 49:EVU | 94: OVF<br>95-96: MC0-1 | 59: OVF<br>60-61: MC0-1 | Y |
| DIVAS | 0x48000000 | | 12 | Y | | | | | | | | | N/A |


© 2026 Microchip Technology Inc. and its subsidiaries

## 12.2. SAM C20/C21 E/G/J

**Table 12-3.** Peripherals Configuration Summary SAM C21 E/G/J

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| AHB-APB Bridge A | 0x40000000 | | 0 | Y | | | | | | | | | N/A |
| PAC | 0x44000000 | 0 | 10 | Y | 0 | Y | | 0 | N | | 86 : ACCERR | | N/A |
| PM | 0x40000400 | 0 | | | 1 | Y | | 1 | N | | | | N/A |
| MCLK | 0x40000800 | 0 | | | 2 | Y | | 2 | N | | | | Y |
| RSTC | 0x40000C00 | | | | 3 | Y | | 3 | N | | | | N/A |
| OSCCTRL | 0x40001000 | 0 | | | 4 | Y | 0: FDPLL96M clk source<br>1: FDPLL96M 32kHz | 4 | N | | 1: XOSC_FAIL | | Y |
| OSC32KCTRL | 0x40001400 | 0 | | | 5 | Y | | 5 | N | | 2: XOSC32K_FAIL | | Y |
| SUPC | 0x40001800 | 0 | | | 6 | Y | | 6 | N | | | | N/A |
| GCLK | 0x40001C00 | | | | 7 | Y | | 7 | N | | | | N/A |
| WDT | 0x40002000 | 1 | | | 8 | Y | | 8 | N | | | | Y |
| RTC | 0x40002400 | 2 | | | 9 | Y | | 9 | N | | 3: CMP0/ALARM0<br>3: CMP1<br>4: OVF<br>5-12: PER0-7 | | Y |
| EIC | 0x40002800 | 3, NMI | | | 10 | Y | 2 | 10 | N | | 14-29: EXTINT0-15 | | Y |
| FREQM | 0x40002C00 | 4 | | | 11 | Y | 3: Measure<br>4: Reference | 11 | N | | | | N/A |
| TSENS | 0x40003000 | 5 | | | 12 | N | 5 | 12 | N | 0: START | 30: WINMON | 1: RESRDY | A |
| AHB-APB Bridge B | 0x41000000 | | 1 | Y | | | | | | | | | N/A |
| PORT | 0x41000000 | | | | 0 | Y | | 0 | N | 1-4 : EV0-3 | | | Y |
| DSU | 0x41002000 | | 3 | Y | 1 | Y | | 1 | Y | | | | N/A |
| NVMCTRL | 0x41004000 | 6 | 5 | Y | 2 | Y | 39 | 2 | N | | | | Y |
| DMAC | 0x41006000 | 7 | 7 | Y | | | | 3 | N | 5-8: CH0-3 | 31-34: CH0-3 | | Y |
| MTB | 0x41008000 | | | | | | | | N | 44: START<br>45: STOP | | | N/A |
| AHB-APB Bridge C | 0x42000000 | | 2 | Y | | | | | | | | | N/A |
| EVSYS | 0x42000000 | 8 | | | 0 | N | 6-17: one per CHANNEL | 0 | N | | | | Y |
| SERCOM0 | 0x42000400 | 9 | | | 1 | N | 19: CORE<br>18: SLOW | 1 | N | | | 2: RX<br>3: TX | Y |
| SERCOM1 | 0x42000800 | 10 | | | 2 | N | 20: CORE<br>18: SLOW | 2 | N | | | 4: RX<br>5: TX | Y |
| SERCOM2 | 0x42000C00 | 11 | | | 3 | N | 21: CORE<br>18: SLOW | 3 | N | | | 6: RX<br>7: TX | Y |
| SERCOM3 | 0x42001000 | 12 | | | 4 | N | 22: CORE<br>18: SLOW | 4 | N | | | 8: RX<br>9: TX | Y |
| SERCOM4 | 0x42001400 | 13 | | | 5 | N | 23: CORE<br>18: SLOW | 5 | N | | | 10: RX<br>11: TX | Y |
| SERCOM5 | 0x42001800 | 14 | | | 6 | N | 25: CORE<br>24: SLOW | 6 | N | | | 12: RX<br>13: TX | Y |
| CAN0 | 0x42001C00 | 15 | 8 | N | | | 26 | | | | | 14: DEBUG | N/A |
| CAN1 | 0x42002000 | 16 | 9 | N | | | 27 | | | | | 15: DEBUG | N/A |
| TCC0 | 0x42002400 | 17 | | | 9 | N | 28 | 9 | N | 9-10: EV0-1<br>11-14: MC0-3 | 35: OVF<br>36: TRG<br>37: CNT<br>38-40: MC0-3 | 16: OVF<br>17-20: MC0-3 | Y |
| TCC1 | 0x42002800 | 18 | | | 10 | N | 28 | 10 | N | 15-16: EV0-1<br>17-18: MC0-1 | 42: OVF<br>43: TRG<br>44: CNT<br>45-46: MC0-1 | 21: OVF<br>22-23: MC0-1 | Y |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-3.** Peripherals Configuration Summary SAM C21 E/G/J (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| TCC2 | 0x42002C00 | 19 | | | 11 | N | 29 | 11 | N | 19-20: EV0-1 21-22: MC0-1 | 47: OVF 48: TRG 49: CNT 50-51: MC0-1 | 24: OVF 25-26: MC0-1 | Y |
| TC0 | 0x42003000 | 20 | | | 12 | N | 30 | 12 | N | 23: EVU | 52: OVF 52-53: MC0-1 | 27: OVF 28-29: MC0-1 | Y |
| TC1 | 0x42003400 | 21 | | | 13 | N | 30 | 13 | N | 24: EVU | 55: OVF 55-56: MC0-1 | 30: OVF 21-32: MC0-1 | Y |
| TC2 | 0x42003800 | 22 | | | 14 | N | 31 | 14 | N | 25: EVU | 58: OVF 58-59: MC0-1 | 33: OVF 23-35: MC0-1 | Y |
| TC3 | 0x42003C00 | 23 | | | 15 | N | 31 | 15 | N | 26: EVU | 61: OVF 61-62: MC0-1 | 36: OVF 37-38: MC0-1 | Y |
| TC4 | 0x42004000 | 24 | | | 16 | N | 32 | 16 | N | 27: EVU | 64: OVF 64-65: MC0-1 | 39: OVF 40-41: MC0-1 | Y |
| ADC0 | 0x42004400 | 25 | | | 17 | N | 33 | 17 | N | 28: START 29: SYNC | 67: RESRDY 67: WINMON | 42: RESRDY | Y |
| ADC1 | 0x42004800 | 26 | | | 18 | N | 34 | 18 | N | 30: START 31: SYNC | 69: RESRDY 69: WINMON | 43: RESRDY | Y |
| SDADC | 0x42004C00 | 29 | | | 19 | N | 35 | 19 | N | 32: START 33: FLUSH | 71: RESRDY 71: WINMON | 44: RESRDY | Y |
| AC | 0x42005000 | 27 | | | 20 | N | 34 | 20 | N | 34-37: SOC0-3 | 73-76: COMP0-3 76-77: WIN0-1 | | Y |
| DAC | 0x42005400 | 28 | | | 21 | N | 36 | 21 | N | 38: START | 79: EMPTY | 45: EMPTY | Y |
| PTC | 0x42005800 | 30 | | | 22 | N | 37 | 22 | N | 39: STCONV | 80: EOC 80: WCOMP | EOC: 46 WCOMP: 47 SEQ: 48 | |
| CCL | 0x42005C00 | | | | 23 | N | 38 | 23 | N | 40-43 : LUTIN0-3 | 82-85: LUTOUT0-3 | | Y |
| DIVAS | 0x48000000 | | 12 | Y | | | | | | | | | N/A |

**Table 12-4.** Peripherals Configuration Summary SAM C20 E/G/J

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| AHB-APB Bridge A | 0x40000000 | | 0 | Y | | | | | | | | | N/A |
| PAC | 0x44000000 | 0 | 10 | Y | 0 | Y | | 0 | N | | 86 : ACCERR | | N/A |
| PM | 0x40000400 | 0 | | | 1 | Y | | 1 | N | | | | N/A |
| MCLK | 0x40000800 | 0 | | | 2 | Y | | 2 | N | | | | Y |
| RSTC | 0x40000C00 | | | | 3 | Y | | 3 | N | | | | N/A |
| OSCCTRL | 0x40001000 | 0 | | | 4 | Y | 0: FDPLL96M clk source 1: FDPLL96M 32kHz | 4 | N | | 1: XOSC_FAIL | | Y |
| OSC32KCTRL | 0x40001400 | 0 | | | 5 | Y | | 5 | N | | 2: XOSC32K_FAIL | | Y |
| SUPC | 0x40001800 | 0 | | | 6 | Y | | 6 | N | | | | N/A |
| GCLK | 0x40001C00 | | | | 7 | Y | | 7 | N | | | | N/A |
| WDT | 0x40002000 | 1 | | | 8 | Y | | 8 | N | | | | Y |
| RTC | 0x40002400 | 2 | | | 9 | Y | | 9 | N | | 3: CMP0/ALARM0 3: CMP1 4: OVF 5-12: PER0-7 | | Y |
| EIC | 0x40002800 | 3, NMI | | | 10 | Y | 2 | 10 | N | | 14-29: EXTINT0-15 | | Y |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-4.** Peripherals Configuration Summary SAM C20 E/G/J (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| FREQM | 0x40002C00 | 4 | | | 11 | Y | 3: Measure<br>4: Reference | 11 | N | | | | N/A |
| AHB-APB Bridge B | 0x41000000 | | 1 | Y | | | | | | | | | N/A |
| PORT | 0x41000000 | | | | 0 | Y | | 0 | N | 1-4 : EV0-3 | | | Y |
| DSU | 0x41002000 | | 3 | Y | 1 | Y | | 1 | Y | | | | N/A |
| NVMCTRL | 0x41004000 | 6 | 5 | Y | 2 | Y | 39 | 2 | N | | | | Y |
| DMAC | 0x41006000 | 7 | 7 | Y | | | | 3 | N | 5-8: CH0-3 | 31-34: CH0-3 | | Y |
| MTB | 0x41008000 | | | | | | | | N | 44: START<br>45: STOP | | | N/A |
| AHB-APB Bridge C | 0x42000000 | | 2 | Y | | | | | | | | | N/A |
| EVSYS | 0x42000000 | 8 | | | 0 | N | 6-17: one per CHANNEL | 0 | N | | | | Y |
| SERCOM0 | 0x42000400 | 9 | | | 1 | N | 19: CORE<br>18: SLOW | 1 | N | | | 2: RX<br>3: TX | Y |
| SERCOM1 | 0x42000800 | 10 | | | 2 | N | 20: CORE<br>18: SLOW | 2 | N | | | 4: RX<br>5: TX | Y |
| SERCOM2 | 0x42000C00 | 11 | | | 3 | N | 21: CORE<br>18: SLOW | 3 | N | | | 6: RX<br>7: TX | Y |
| SERCOM3 | 0x42001000 | 12 | | | 4 | N | 22: CORE<br>18: SLOW | 4 | N | | | 8: RX<br>9: TX | Y |
| SERCOM4 | 0x42001400 | 13 | | | 5 | N | 23: CORE<br>18: SLOW | 5 | N | | | 10: RX<br>11: TX | Y |
| SERCOM5 | 0x42001800 | 14 | | | 6 | N | 25: CORE<br>24: SLOW | 6 | N | | | 12: RX<br>13: TX | Y |
| TCC0 | 0x42002400 | 17 | | | 9 | N | 28 | 9 | N | 9-10: EV0-1<br>11-14: MC0-3 | 35: OVF<br>36: TRG<br>37: CNT<br>37-40: MC0-3 | 16: OVF<br>17-20: MC0-3 | Y |
| TCC1 | 0x42002800 | 18 | | | 10 | N | 28 | 10 | N | 15-16: EV0-1<br>17-18: MC0-1 | 42: OVF<br>43: TRG<br>44: CNT<br>44-45: MC0-1 | 21: OVF<br>22-23: MC0-1 | Y |
| TCC2 | 0x42002C00 | 19 | | | 11 | N | 29 | 11 | N | 19-20: EV0-1<br>21-22: MC0-1 | 47: OVF<br>48: TRG<br>49: CNT<br>49-50: MC0-1 | 24: OVF<br>25-26: MC0-1 | Y |
| TC0 | 0x42003000 | 20 | | | 12 | N | 30 | 12 | N | 23: EVU | 52: OVF<br>52-53: MC0-1 | 27: OVF<br>28-29: MC0-1 | Y |
| TC1 | 0x42003400 | 21 | | | 13 | N | 30 | 13 | N | 24: EVU | 55: OVF<br>55-56: MC0-1 | 30: OVF<br>21-32: MC0-1 | Y |
| TC2 | 0x42003800 | 22 | | | 14 | N | 31 | 14 | N | 25: EVU | 58: OVF<br>58-59: MC0-1 | 33: OVF<br>23-35: MC0-1 | Y |
| TC3 | 0x42003C00 | 23 | | | 15 | N | 31 | 15 | N | 26: EVU | 61: OVF<br>61-62: MC0-1 | 36: OVF<br>37-38: MC0-1 | Y |
| TC4 | 0x42004000 | 24 | | | 16 | N | 32 | 16 | N | 27: EVU | 64: OVF<br>64-65: MC0-1 | 39: OVF<br>40-41: MC0-1 | Y |
| ADC0 | 0x42004400 | 25 | | | 17 | N | 33 | 17 | N | 28: START<br>29: SYNC | 67: RESRDY<br>67: WINMON | 42: RESRDY | Y |
| AC | 0x42005000 | 27 | | | 20 | N | 34 | 20 | N | 34-37: SOC0-3 | 73-76: COMP0-3<br>76-77: WIN0-1 | | Y |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 12-4.** Peripherals Configuration Summary SAM C20 E/G/J (continued)

| Peripheral Name | Base Address | IRQ Line | AHB Clock | | APB Clock | | Generic Clock | PAC | | Events | | DMA | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Index | Enabled at Reset | Index | Enabled at Reset | Index | Index | Prot at Reset | User | Generator | Index | Sleep Walking |
| PTC | 0x42005800 | 30 | | | 22 | N | 37 | 22 | N | 39: STCONV | 80: EOC<br>80: WCOMP | EOC: 46<br>WCOMP: 47<br>SEQ: 48 | |
| CCL | 0x42005C00 | | | | 23 | N | 38 | 23 | N | 40-43 : LUTIN0-3 | 82-85: LUTOUT0-3 | | Y |
| DIVAS | 0x48000000 | | 12 | Y | | | | | | | | | N/A |


© 2026 Microchip Technology Inc. and its subsidiaries

# 13. DSU - Device Service Unit

## 13.1. Overview

The Device Service Unit (DSU) provides a means of detecting debugger probes. It enables the Arm Debug Access Port (DAP) to have control over multiplexed debug pads and CPU reset. The DSU also provides system-level services to debug adapters in an Arm debug system. It implements a CoreSight Debug ROM that provides device identification as well as identification of other debug components within the system. Hence, it complies with the Arm Peripheral Identification specification. The DSU also provides system services to applications that need memory testing, as required for IEC60730 Class B compliance, for example. The DSU can be accessed simultaneously by a debugger and the CPU, as it is connected on the High-Speed Bus Matrix. For security reasons, some of the DSU features will be limited or unavailable when the device is protected by the NVMCTRL security bit.

**Related Links**

System Services Availability when Accessed Externally and Device is Protected
NVMCTRL – Nonvolatile Memory Controller
Security Bit

## 13.2. Features

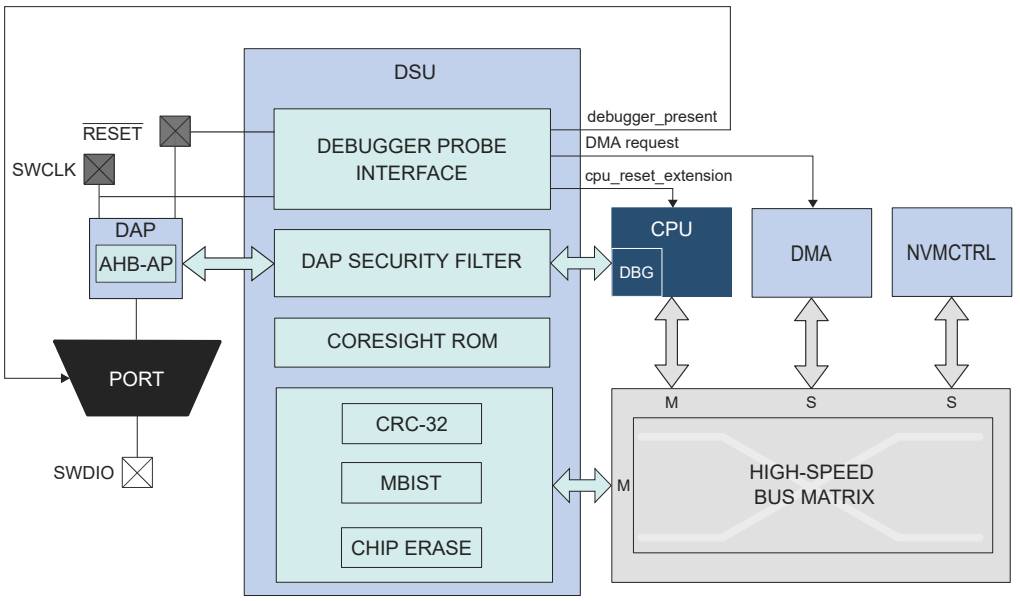
The following are key features of the DSU module:

- CPU reset extension
- Debugger probe detection (Cold- and Hot-Plugging)
- Chip-Erase command and status
- 32-bit cyclic redundancy check (CRC32) of any memory accessible through the bus matrix
- Arm CoreSight compliant device identification
- Two debug communications channels with DMA connection
- Debug access port security filter
- Onboard memory built-in self-test (MBIST)


© 2026 Microchip Technology Inc. and its subsidiaries

## 13.3. Block Diagram

**Figure 13-1.** DSU Block Diagram




## 13.4. Signal Description

The DSU uses three signals to function.

| Signal Name | Type | Description |
|---|---|---|
| $\overline{\text{RESET}}$ | Digital Input | External Reset pin |
| SWCLK | Digital Input | SW clock pin |
| SWDIO | Digital I/O | SW bidirectional data pin |

**Related Links**

I/O Multiplexing and Considerations

## 13.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 13.5.1. I/O Lines

The SWCLK pin is by default assigned to the DSU module to allow debugger probe detection and to stretch the CPU reset phase. For more information, refer to Debugger Probe Detection. The Hot-Plugging feature depends on the PORT configuration. If the SWCLK pin function is changed in the PORT or if the PORT_MUX is disabled, the Hot-Plugging feature is disabled until a power-reset or an external reset is performed.

### 13.5.2. Power Management

The DSU will continue to operate in Idle mode.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.5.3. Clocks

The DSU bus clocks (CLK_DSU_APB and CLK_DSU_AHB) can be enabled and disabled by the Main Clock Controller.

**Related Links**

PM - Power Manager
MCLK – Main Clock
Peripheral Clock Masking

### 13.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to DMAC – Direct Memory Access Controller for details.

### 13.5.5. Interrupts

Not applicable.

### 13.5.6. Events

Not applicable.

### 13.5.7. Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except for the following:

- Debug Communication Channel 0 register (DCC0)
- Debug Communication Channel 1 register (DCC1)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

**Related Links**

PAC - Peripheral Access Controller

### 13.5.8. Analog Connections

Not applicable.

## 13.6. Debug Operation

### 13.6.1. Principle of Operation

The DSU provides basic services to allow on-chip debug using the Arm Debug Access Port and the Arm processor debug resources:

- CPU reset extension
- Debugger probe detection

For additional information on Arm debug components, refer to the "*Arm Debug Interface v5 Architecture Specification*".

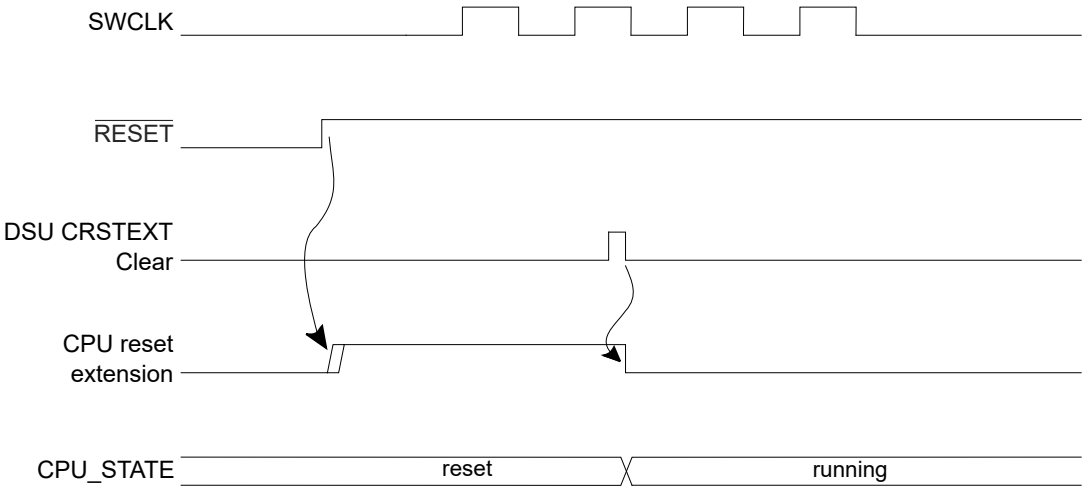
### 13.6.2. CPU Reset Extension

"CPU reset extension" refers to the extension of the reset phase of the CPU core after the external reset is released. This ensures that the CPU is not executing code at startup while a debugger is connects to the system. The debugger is detected on a $\overline{\text{RESET}}$ release event when SWCLK is low. At startup, SWCLK is internally pulled up to avoid false detection of a debugger if the SWCLK pin is left


© 2026 Microchip Technology Inc. and its subsidiaries

unconnected. When the CPU is held in the reset extension phase, the CPU Reset Extension bit of the Status A register (STATUSA.CRSTEXT) is set. To release the CPU, write a '1' to STATUSA.CRSTEXT. STATUSA.CRSTEXT will then be set to '0'. Writing a '0' to STATUSA.CRSTEXT has no effect. For security reasons, it is not possible to release the CPU reset extension when the device is protected by the NVMCTRL security bit. Trying to do so sets the Protection Error bit (PERR) of the Status A register (STATUSA.PERR).

**Figure 13-2.** Typical CPU Reset Extension Set and Clear Timing Diagram




**Related Links**

NVMCTRL – Nonvolatile Memory Controller
Security Bit

### 13.6.3. Debugger Probe Detection

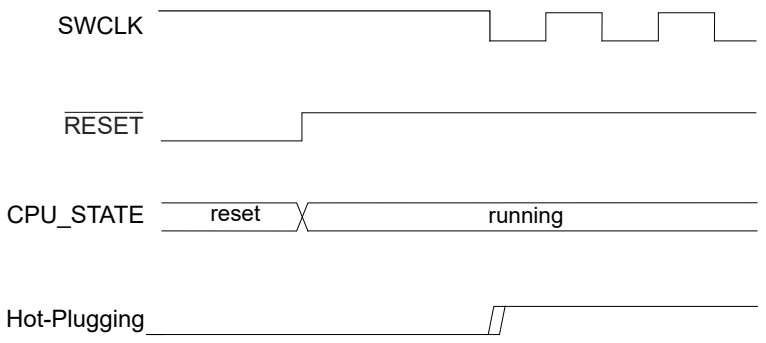
#### 13.6.3.1. Cold Plugging

Cold-Plugging is the detection of a debugger when the system is in reset. Cold-Plugging is detected when the CPU reset extension is requested, as described above.

#### 13.6.3.2. Hot Plugging

Hot-Plugging is the detection of a debugger probe when the system is not in reset. Hot-Plugging is not possible under reset because the detector is reset when POR or $\overline{\text{RESET}}$ are asserted. Hot-Plugging is active when a SWCLK falling edge is detected. The SWCLK pad is multiplexed with other functions and the user must ensure that its default function is assigned to the debug system. If the SWCLK function is changed, the Hot-Plugging feature is disabled until a power-reset or external reset occurs. Availability of the Hot-Plugging feature can be read from the Hot-Plugging Enable bit of the Status B register (STATUSB.HPE).

**Figure 13-3.** Hot-Plugging Detection Timing Diagram





© 2026 Microchip Technology Inc. and its subsidiaries

The presence of a debugger probe is detected when either Hot-Plugging or Cold-Plugging is detected. Once detected, the Debugger Present bit of the Status B register (STATUSB.DBGPRES) is set. For security reasons, Hot-Plugging is not available when the device is protected by the NVMCTRL security bit.

This detection requires that pads are correctly powered. Thus, at cold startup, this detection cannot be done until POR is released. If the device is protected, Cold-Plugging is the only way to detect a debugger probe, and so the external reset timing must be longer than the POR timing. If external reset is deasserted before POR release, the user must retry the procedure above until it gets connected to the device.

**Related Links**

NVMCTRL – Nonvolatile Memory Controller

Security Bit

## 13.7. Chip Erase

Chip-Erase consists of removing all sensitive information stored in the chip and clearing the NVMCTRL security bit. Therefore, all volatile memories and the Flash memory (including the EEPROM emulation area) will be erased. The Flash auxiliary rows, including the user row, will not be erased.

When the device is protected, the debugger must first reset the device in order to be detected. This ensures that internal registers are reset after the protected state is removed. The Chip-Erase operation is triggered by writing a '1' to the Chip-Erase bit in the Control register (CTRL.CE). This command will be discarded if the DSU is protected by the Peripheral Access Controller (PAC). Once issued, the module clears volatile memories prior to erasing the Flash array. To ensure that the Chip-Erase operation is completed, check the Done bit of the Status A register (STATUSA.DONE).

The Chip-Erase operation depends on clocks and power management features that can be altered by the CPU. For that reason, it is recommended to issue a Chip- Erase after a Cold-Plugging procedure to ensure that the device is in a known and safe state.

The recommended sequence is as follows:

1. Issue the Cold-Plugging procedure (refer to Cold Plugging). The device then:
   a. Detects the debugger probe.
   b. Holds the CPU in reset.
2. Issue the Chip-Erase command by writing a '1' to CTRL.CE. The device then:
   a. Clears the system volatile memories.
   b. Erases the whole Flash array (including the EEPROM emulation area, not including auxiliary rows).
   c. Erases the lock row, removing the NVMCTRL security bit protection.
3. Check for completion by polling STATUSA.DONE (read as '1' when completed).
4. Reset the device to let the NVMCTRL update the fuses.

## 13.8. Programming

Programming the Flash or RAM memories is only possible when the device is not protected by the NVMCTRL security bit. The programming procedure is as follows:

1. At power up, $\overline{\text{RESET}}$ is driven low by a debugger. The on-chip regulator holds the system in a POR state until the input supply is above the POR threshold (refer to Powe-On Reset (POR) characteristics). The system continues to be held in this static state until the internally regulated supplies have reached a safe operating state.
2. The PM starts, clocks are switched to the slow clock (Core Clock, System Clock, Flash Clock and any Bus Clocks that do not have clock gate control). Internal resets are maintained due to the external reset.


© 2026 Microchip Technology Inc. and its subsidiaries

3. The debugger maintains a low level on SWCLK. $\overline{\text{RESET}}$ is released, resulting in a debugger Cold-Plugging procedure.
4. The debugger generates a clock signal on the SWCLK pin, the Debug Access Port (DAP) receives a clock.
5. The CPU remains in Reset due to the Cold-Plugging procedure; meanwhile, the rest of the system is released.
6. A Chip-Erase is issued to ensure that the Flash is fully erased prior to programming.
7. Programming is available through the AHB-AP.
8. After the operation is completed, the chip can be restarted either by asserting $\overline{\text{RESET}}$ or toggling power. Make sure that the SWCLK pin is high when releasing $\overline{\text{RESET}}$ to prevent extending the CPU reset.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)
NVMCTRL – Nonvolatile Memory Controller
Security Bit

## 13.9. Intellectual Property Protection

Intellectual property protection consists of restricting access to internal memories from external tools when the device is protected, and this is accomplished by setting the NVMCTRL security bit. This protected state can be removed by issuing a Chip-Erase (refer to Chip Erase). When the device is protected, read/write accesses using the AHB-AP are limited to the DSU address range and DSU commands are restricted. When issuing a Chip-Erase, sensitive information is erased from volatile memory and Flash.

The DSU implements a security filter that monitors the AHB transactions inside the DAP. If the device is protected, then AHB-AP read/write accesses outside the DSU external address range are discarded, causing an error response that sets the ARM AHB-AP sticky error bits (refer to the ARM Debug Interface v5 Architecture Specification on www.arm.com).

The DSU is intended to be accessed either:

- Internally from the CPU, without any limitation, even when the device is protected
- Externally from a debug adapter, with some restrictions when the device is protected

For security reasons, DSU features have limitations when used from a debug adapter. To differentiate external accesses from internal ones, the first 0x100 bytes of the DSU register map has been mirrored at offset 0x100:

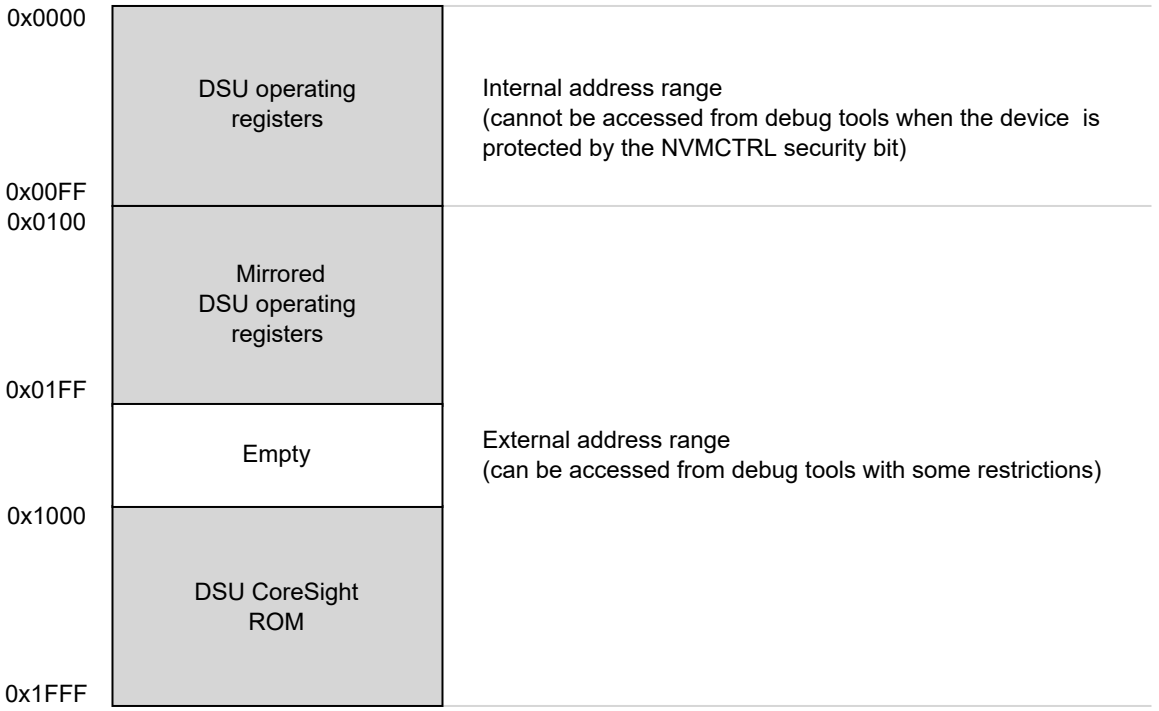
- The first 0x100 bytes form the internal address range
- The next 0x100 bytes form the external address range

When the device is protected, the DAP can only issue MEM-AP accesses in the DSU range 0x0100-0x2000.

The DSU operating registers are located in the 0x0000-0x00FF area and remapped in 0x0100-0x01FF to differentiate accesses coming from a debugger and the CPU. If the device is protected and an access is issued in the region 0x0100-0x01FF, it is subject to security restrictions. For more information, refer to the Table 13-1.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 13-4.** APB Memory Mapping




Some features not activated by APB transactions are not available when the device is protected:

**Table 13-1.** Feature Availability Under Protection

| Features | Availability when the device is protected |
|---|---|
| CPU Reset Extension | Yes |
| Clear CPU Reset Extension | No |
| Debugger Cold-Plugging | Yes |
| Debugger Hot-Plugging | No |

**Related Links**

NVMCTRL – Nonvolatile Memory Controller
Security Bit

## 13.10. Program and Debug Interface Disable (PDID)

After activating the Program and Debug Interface Disable (PDID), the only way to write to the reprogrammable Flash memory (non-volatile memory (NVM)) is from the code executing on the device. Chip erase or other re-programming attempts through the debug port will fail. Also, any attempt to read out any device content (NVM, RAM, and so on) will fail. Contact a local Microchip sales office for additional information about PDID functionality.

## 13.11. Device Identification

Device identification relies on the ARM CoreSight component identification scheme, which allows the chip to be identified as a SAM device implementing a DSU. The DSU contains identification registers to differentiate the device.

### 13.11.1. CoreSight Identification

A system-level ARM® CoreSight™ ROM table is present in the device to identify the vendor and the chip identification method. Its address is provided in the MEM-AP BASE register inside the ARM


© 2026 Microchip Technology Inc. and its subsidiaries

Debug Access Port. The CoreSight ROM implements a 64-bit conceptual ID composed as follows from the PID0 to PID7 CoreSight ROM Table registers:

**Figure 13-5.** Conceptual 64-bit Peripheral ID

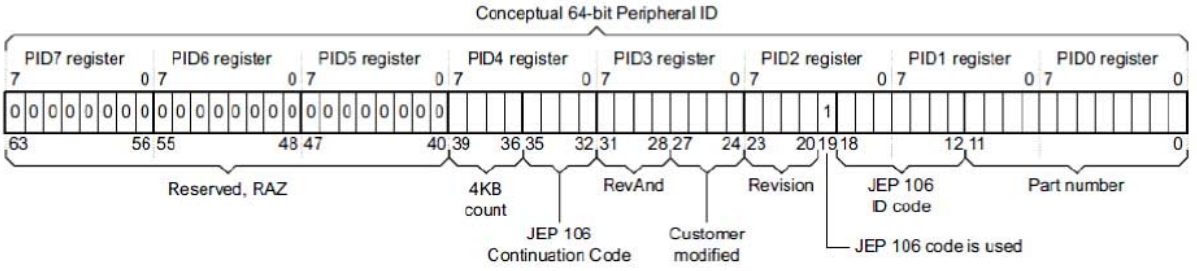



**Table 13-2.** Conceptual 64-Bit Peripheral ID Bit Descriptions

| Field | Size | Description | Location |
|---|---|---|---|
| JEP-106 CC code | 4 | Continuation code: 0x0 | PID4 |
| JEP-106 ID code | 7 | Device ID: 0x1F | PID1+PID2 |
| 4KB count | 4 | Indicates that the CoreSight component is a ROM: 0x0 | PID4 |
| RevAnd | 4 | Not used; read as 0 | PID3 |
| CUSMOD | 4 | Not used; read as 0 | PID3 |
| PARTNUM | 12 | Contains 0xCD0 to indicate that DSU is present | PID0+PID1 |
| REVISION | 4 | DSU revision (starts at 0x0 and increments by 1 at both major and minor revisions). Identifies DSU identification method variants. If 0x0, this indicates that device identification can be completed by reading the Device Identification register (DID) | PID2 |

For more information, refer to the ARM Debug Interface Version 5 Architecture Specification.

### 13.11.2. Chip Identification Method

The DSU DID register identifies the device by implementing the following information:

- Processor identification
- Product family identification
- Product series identification
- Device select

## 13.12. Functional Description

### 13.12.1. Principle of Operation

The DSU provides memory services, such as CRC32 or MBIST that require almost the same interface. Hence, the Address, Length and Data registers (ADDR, LENGTH, DATA) are shared. These shared registers must be configured first; then a command can be issued by writing the Control register. When a command is ongoing, other commands are discarded until the current operation is completed. Hence, the user must wait for the STATUSA.DONE bit to be set prior to issuing another one.

### 13.12.2. Basic Operation

#### 13.12.2.1. Initialization

The module is enabled by enabling its clocks. For more details, refer to Clocks. The DSU registers can be PAC write-protected.

**Related Links**

PAC - Peripheral Access Controller


© 2026 Microchip Technology Inc. and its subsidiaries

#### 13.12.2.2. Operation From a Debug Adapter

Debug adapters should access the DSU registers in the external address range 0x100 – 0x2000. If the device is protected by the NVMCTRL security bit, accessing the first 0x100 bytes causes the system to return an error. Refer to Intellectual Property Protection.

**Related Links**

NVMCTRL – Nonvolatile Memory Controller

Security Bit

#### 13.12.2.3. Operation From the CPU

There are no restrictions when accessing DSU registers from the CPU. However, the user should access DSU registers in the internal address range (0x0 – 0x100) to avoid external security restrictions. Refer to Intellectual Property Protection.

### 13.12.3. 32-bit Cyclic Redundancy Check CRC32

The DSU unit provides support for calculating a cyclic redundancy check (CRC32) value for a memory area (including Flash and AHB RAM).

When the CRC32 command is issued from:

- The internal range, the CRC32 can be operated at any memory location
- The external range, the CRC32 operation is restricted; DATA, ADDR, and LENGTH values are forced (see below)

**Table 13-3.** AMOD Bit Descriptions when Operating CRC32

| AMOD[1:0] | Short name | External range restrictions |
|---|---|---|
| 0 | ARRAY | CRC32 is restricted to the full Flash array area (EEPROM emulation area not included) DATA forced to 0xFFFFFFFF before calculation (no seed) |
| 1 | EEPROM | CRC32 of the whole EEPROM emulation area DATA forced to 0xFFFFFFFF before calculation (no seed) |
| 2-3 | Reserved | |

The algorithm employed is the industry standard CRC32 algorithm using the generator polynomial 0xEDB88320 (reversed representation).

#### 13.12.3.1. Starting CRC32 Calculation

CRC32 calculation for a memory range is started after writing the start address into the Address register (ADDR) and the size of the memory range into the Length register (LENGTH). Both must be word-aligned.

The initial value used for the CRC32 calculation must be written to the Data register (DATA). This value will usually be 0xFFFFFFFF, but can be, for example, the result of a previous CRC32 calculation if generating a common CRC32 of separate memory blocks.

Once completed, the calculated CRC32 value can be read out of the Data register. The read value must be complemented to match standard CRC32 implementations or kept non-inverted if used as starting point for subsequent CRC32 calculations.

The actual test is started by writing a '1' in the 32-bit Cyclic Redundancy Check bit of the Control register (CTRL.CRC). A running CRC32 operation can be canceled by resetting the module (writing '1' to CTRL.SWRST).

**Related Links**

NVMCTRL – Nonvolatile Memory Controller

Security Bit


© 2026 Microchip Technology Inc. and its subsidiaries

#### 13.12.3.2. Interpreting the Results

The user should monitor the Status A register. When the operation is completed, STATUSA.DONE is set. Then the Bus Error bit of the Status A register (STATUSA.BERR) must be read to ensure that no bus error occurred.

### 13.12.4. Debug Communication Channels

The Debug Communication Channels (DCCO and DCC1) consist of a pair of registers with associated handshake logic, accessible by both CPU and debugger even if the device is protected by the NVMCTRL security bit. The registers can be used to exchange data between the CPU and the debugger, during run time as well as in debug mode. This enables the user to build a custom debug protocol using only these registers.

The DCC0 and DCC1 registers are accessible when the protected state is active. When the device is protected, however, it is not possible to connect a debugger while the CPU is running (STATUSA.CRSTEXT is not writable and the CPU is held under Reset).

Two Debug Communication Channel status bits in the Status B registers (STATUS.DCCDx) indicate whether a new value has been written in DCC0 or DCC1. These bits, DCC0D and DCC1D, are located in the STATUSB registers. They are automatically set on write and cleared on read.

**Note:** The DCC0 and DCC1 registers are shared with the on-board memory testing logic (MBIST). Accordingly, DCC0 and DCC1 must not be used while performing MBIST operations.

**Related Links**

NVMCTRL – Nonvolatile Memory Controller

Security Bit

### 13.12.5. Testing of On-Board Memories MBIST

The DSU implements a feature for automatic testing of memory, also known as MBIST (memory built-in self test). This is primarily intended for production test of on-board memories. MBIST cannot be operated from the external address range when the device is protected by the NVMCTRL security bit. If an MBIST command is issued when the device is protected, a protection error is reported in the Protection Error bit in the Status A register (STATUSA.PERR).

1. Algorithm
   The algorithm used for testing is a type of March algorithm called "March LR". This algorithm can detect a wide range of memory defects, while still keeping a linear run time. The algorithm is :
   a. Write entire memory to '0', in any order.
   b. Bit by bit read '0', write '1', in descending order.
   c. Bit by bit read '1', write '0', read '0', write '1', in ascending order.
   d. Bit by bit read '1', write '0', in ascending order.
   e. Bit by bit read '0', write '1', read '1', write '0', in ascending order.
   f. Read '0' from entire memory, in ascending order.

   The specific implementation used as a run time which depends on the CPU clock frequency and the number of bytes tested in the RAM. The detected faults are:
   - Address decoder faults
   - Stuck-at faults
   - Transition faults
   - Coupling faults
   - Linked Coupling faults
2. Starting MBIST


© 2026 Microchip Technology Inc. and its subsidiaries

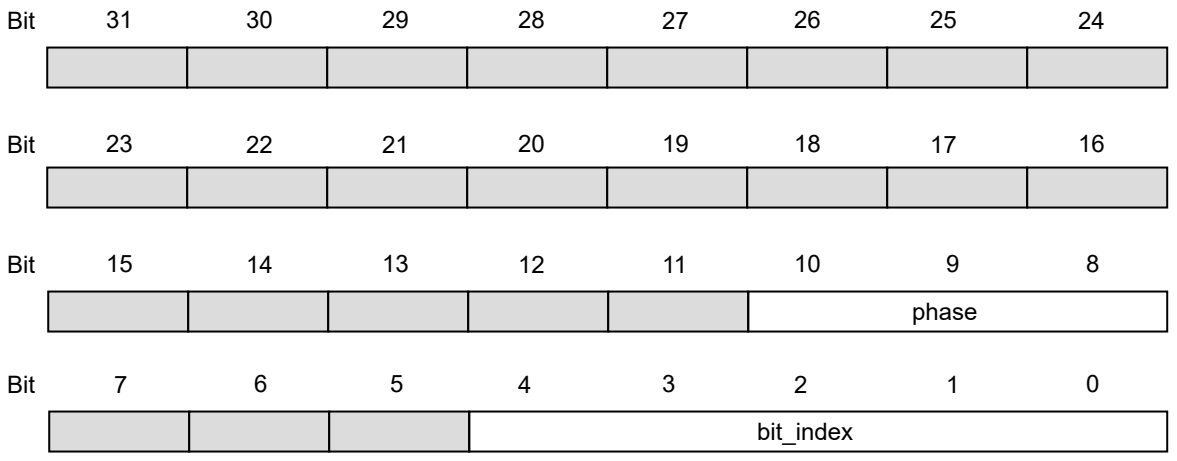
To test a memory, you must write the Start address of the memory to the ADDR.ADDR bit field, and the size of the memory into the Length register.

For best test coverage, an entire physical memory block must be tested at once. It is possible to test only a subset of a memory, but the test coverage will then be somewhat lower.

The actual test is started by writing a '1' to CTRL.MBIST. A running MBIST operation can be canceled by writing a '1' to CTRL.SWRST.

3. Interpreting the Results
   The tester must monitor the STATUSA register. When the operation is completed, STATUSA.DONE is set. There are two different modes:
   - ADDR.AMOD = 0: exit-on-error (default)
     In this mode, the algorithm terminates either when a fault is detected or on successful completion. In both cases, STATUSA.DONE is set. If an error was detected, STATUSA.FAIL will be set. Then the user can read the DATA and ADDR registers to locate the fault.
   - ADDR.AMOD = 1: pause-on-error
     In this mode, the MBIST algorithm is paused when an error is detected. In such a situation, only STATUSA.FAIL is asserted. The state machine waits for user to clear STATUSA.FAIL by writing a '1' in STATUSA.FAIL to resume. Prior to resuming, users can read the DATA and ADDR registers to locate the fault.
4. Locating Faults
   If the test stops with STATUSA.FAIL set, one or more bits failed the test. The test stops at the first detected error. The position of the failing bit can be found by reading the following registers:
   - ADDR: Address of the word containing the failing bit
   - DATA: contains data to identify which bit failed, and during which phase of the test it failed. The DATA register will in this case contains the following bit groups:

**Figure 13-6.** DATA bits Description When MBIST Operation Returns an Error




- bit_index: contains the bit number of the failing bit
- phase: indicates which phase of the test failed and the cause of the error, as listed in the following table.

**Table 13-4.** MBIST Operation Phases

| Phase | Test actions |
|---|---|
| 0 | Write all bits to zero. This phase cannot fail. Ascending order. |
| 1 | Read '0', write '1', descending order |
| 2 | Read '1', write '0', ascending order |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 13-4.** MBIST Operation Phases (continued)

| Phase | Test actions |
|---|---|
| 3 | Read '0', write '1', ascending order |
| 4 | Read '1', write '0', ascending order |
| 5 | Read '0', write '1', ascending order |
| 6 | Read '1', write '0', ascending order |
| 7 | Read all zeros. bit_index is not used. Ascending order. |

**Table 13-5.** AMOD Bit Descriptions for MBIST

| AMOD[1:0] | Description |
|---|---|
| 0x0 | Exit on Error |
| 0x1 | Pause on Error |
| 0x2, 0x3 | Reserved |

**Related Links**
NVMCTRL – Nonvolatile Memory Controller
Security Bit
Product Mapping

### 13.12.6. System Services Availability when Accessed Externally and Device is Protected

External access: Access performed in the DSU address offset 0x200-0x1FFF range.

Internal access: Access performed in the DSU address offset 0x000-0x100 range.

**Table 13-6.** Available Features when Operated From The External Address Range and Device is Protected

| Features | Availability From The External Address Range and Device is Protected |
|---|---|
| Chip-Erase command and status | Yes |
| CRC32 | Yes, only full array or full EEPROM |
| CoreSight Compliant Device identification | Yes |
| Debug communication channels | Yes |
| Testing of onboard memories (MBIST) | No |
| STATUSA.CRSTEXT clearing | No (STATUSA.PERR is set when attempting to do so) |


© 2026 Microchip Technology Inc. and its subsidiaries

## 13.13. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRL | 7:0 | Reserved | Reserved | | CE | MBIST | CRC | | SWRST |
| 0x01 | STATUSA | 7:0 | | | | PERR | FAIL | BERR | CRSTEXT | DONE |
| 0x02 | STATUSB | 7:0 | | | | HPE | DCCD1 | DCCD0 | DBGPRES | PROT |
| 0x03 | Reserved | | | | | | | | | |
| 0x04 | ADDR | 7:0 | ADDR[5:0] | | | | | | AMOD[1:0] | |
| | | 15:8 | ADDR[13:6] | | | | | | | |
| | | 23:16 | ADDR[21:14] | | | | | | | |
| | | 31:24 | ADDR[29:22] | | | | | | | |
| 0x08 | LENGTH | 7:0 | LENGTH[5:0] | | | | | | | |
| | | 15:8 | LENGTH[13:6] | | | | | | | |
| | | 23:16 | LENGTH[21:14] | | | | | | | |
| | | 31:24 | LENGTH[29:22] | | | | | | | |
| 0x0C | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | DATA[15:8] | | | | | | | |
| | | 23:16 | DATA[23:16] | | | | | | | |
| | | 31:24 | DATA[31:24] | | | | | | | |
| 0x10 | DCC0 | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | DATA[15:8] | | | | | | | |
| | | 23:16 | DATA[23:16] | | | | | | | |
| | | 31:24 | DATA[31:24] | | | | | | | |
| 0x14 | DCC1 | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | DATA[15:8] | | | | | | | |
| | | 23:16 | DATA[23:16] | | | | | | | |
| | | 31:24 | DATA[31:24] | | | | | | | |
| 0x18 | DID | 7:0 | DEVSEL[7:0] | | | | | | | |
| | | 15:8 | DIE[3:0] | | | | REVISION[3:0] | | | |
| | | 23:16 | FAMILY[0] | | SERIES[5:0] | | | | | |
| | | 31:24 | PROCESSOR[3:0] | | | | FAMILY[4:1] | | | |
| 0x1C ... 0x0FFF | Reserved | | | | | | | | | |
| 0x1000 | ENTRY0 | 7:0 | | | | | | | FMT | EPRES |
| | | 15:8 | ADDOFF[3:0] | | | | | | | |
| | | 23:16 | ADDOFF[11:4] | | | | | | | |
| | | 31:24 | ADDOFF[19:12] | | | | | | | |
| 0x1004 | ENTRY1 | 7:0 | | | | | | | FMT | EPRES |
| | | 15:8 | ADDOFF[3:0] | | | | | | | |
| | | 23:16 | ADDOFF[11:4] | | | | | | | |
| | | 31:24 | ADDOFF[19:12] | | | | | | | |
| 0x1008 | END | 7:0 | END[7:0] | | | | | | | |
| | | 15:8 | END[15:8] | | | | | | | |
| | | 23:16 | END[23:16] | | | | | | | |
| | | 31:24 | END[31:24] | | | | | | | |
| 0x100C ... 0x1FCB | Reserved | | | | | | | | | |
| 0x1FCC | MEMTYPE | 7:0 | | | | | | | | SMEMP |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FD0 | PID4 | 7:0 | FKBC[3:0] | | | | JEPCC[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FD4 ... 0x1FDF | Reserved | | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x1FE0 | PID0 | 7:0 | PARTNBL[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FE4 | PID1 | 7:0 | JEPIDCL[3:0] | | | | PARTNBH[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FE8 | PID2 | 7:0 | REVISION[3:0] | | | | JEPU | JEPIDCH[2:0] | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FEC | PID3 | 7:0 | REVAND[3:0] | | | | CUSMOD[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FF0 | CID0 | 7:0 | PREAMBLEB0[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FF4 | CID1 | 7:0 | CCLASS[3:0] | | | | PREAMBLE[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FF8 | CID2 | 7:0 | PREAMBLEB2[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1FFC | CID3 | 7:0 | PREAMBLEB3[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 13.14. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.1. Control

**Name:** CTRL
**Offset:** 0x0000
**Reset:** 0x00
**Property:** PAC Write Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | Reserved | Reserved | | CE | MBIST | CRC | | SWRST |
| Access | - | - | | W | W | W | | W |
| Reset | 0 | 0 | | 0 | 0 | 0 | | 0 |

**Bit 7 – Reserved** Must be set to 0.

**Bit 6 – Reserved** Must be set to 0.

**Bit 4 – CE** Chip Erase
Writing a '0' to this bit has no effect.
Writing a '1' to this bit starts the chip erase operation.

**Bit 3 – MBIST** Memory Built-In Self-Test
Writing a '0' to this bit has no effect.
Writing a '1' to this bit starts the memory BIST algorithm.

**Bit 2 – CRC** 32-Bit Cyclic Redundancy Check
Writing a '0' to this bit has no effect.
Writing a '1' to this bit starts the cyclic redundancy check algorithm.

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets the module.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.2. Status A

**Name:** STATUSA
**Offset:** 0x0001
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | PERR | FAIL | BERR | CRSTEXT | DONE |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bit 4 – PERR** Protection Error
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Protection Error bit.
This bit is set when a command that is not allowed in protected state is issued.

**Bit 3 – FAIL** Failure
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Failure bit.
This bit is set when a DSU operation failure is detected.

**Bit 2 – BERR** Bus Error
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Bus Error bit.
This bit is set when a bus error is detected.

**Bit 1 – CRSTEXT** CPU Reset Phase Extension
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the CPU Reset Phase Extension bit.
This bit is set when a debug adapter Cold-Plugging is detected, which extends the CPU reset phase.

**Bit 0 – DONE** Done
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Done bit.
This bit is set when a DSU operation is completed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.3. Status B

**Name:** STATUSB
**Offset:** 0x0002
**Reset:** 0x1X
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | HPE | DCCD1 | DCCD0 | DBGPRES | PROT |
| Access | | | | R | R | R | R | R |
| Reset | | | | 1 | 0 | 0 | 0 | 0 |

**Bit 4 – HPE** Hot-Plugging Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit has no effect.
This bit is set when Hot-Plugging is enabled.
This bit is cleared when Hot-Plugging is disabled. This is the case when the SWCLK function is changed. Only a power-reset or a external reset can set it again.

**Bits 2, 3 – DCCDx** Debug Communication Channel x Dirty [x=1..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit has no effect.
This bit is set when DCCx is written.
This bit is cleared when DCCx is read.

**Bit 1 – DBGPRES** Debugger Present
Writing a '0' to this bit has no effect.
Writing a '1' to this bit has no effect.
This bit is set when a debugger probe is detected.
This bit is never cleared.

**Bit 0 – PROT** Protected
Writing a '0' to this bit has no effect.
Writing a '1' to this bit has no effect.
This bit is set at power-up when the device is protected.
This bit is never cleared.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.4. Address

**Name:** ADDR
**Offset:** 0x0004
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[29:22] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[21:14] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[13:6] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[5:0] | | | | | | AMOD[1:0] | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:2 – ADDR[29:0]** Address
Initial word start address needed for memory operations.

**Bits 1:0 – AMOD[1:0]** Access Mode
The functionality of these bits is dependent on the operation mode.
Bit description when operating CRC32: refer to 32-bit Cyclic Redundancy Check CRC32
Bit description when testing onboard memories (MBIST): refer to Testing of On-Board Memories MBIST


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.5. Length

**Name:** LENGTH
**Offset:** 0x0008
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[29:22] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[21:14] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[13:6] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | LENGTH[5:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

**Bits 31:2 – LENGTH[29:0]** Length
Length in words needed for memory operations.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.6. Data

**Name:** DATA
**Offset:** 0x000C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DATA[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DATA[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DATA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DATA[31:0]** Data
Memory operation initial value or result value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.7. Debug Communication Channel 0

**Name:** DCC0
**Offset:** 0x0010
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DATA[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DATA[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DATA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DATA[31:0]** Data
Data register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.8. Debug Communication Channel 1

**Name:** DCC1
**Offset:** 0x0014
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DATA[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DATA[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DATA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DATA[31:0]** Data
Data register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.9. Device Identification

**Name:** DID
**Offset:** 0x0018
**Property:** PAC Write-Protection

The information in this register is related to the *Ordering Information*.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PROCESSOR[3:0] | | | | FAMILY[4:1] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | p | p | p | p | f | f | f | f |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | FAMILY[0] | | SERIES[5:0] | | | | | |
| Access | R | | R | R | R | R | R | R |
| Reset | f | | s | s | s | s | s | s |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIE[3:0] | | | | REVISION[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | d | d | d | d | r | r | r | r |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DEVSEL[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

**Bits 31:28 – PROCESSOR[3:0]** Processor
The value of this field defines the processor used on the device.

**Bits 27:23 – FAMILY[4:0]** Product Family
The value of this field corresponds to the product family part of the ordering code.

**Bits 21:16 – SERIES[5:0]** Product Series
The value of this field corresponds to the product series part of the ordering code.

**Bits 15:12 – DIE[3:0]** Die Number
Identifies the die family.

**Bits 11:8 – REVISION[3:0]** Revision Number
Identifies the die revision number. 0x0=rev.A, 0x1=rev.B etc.
**Note:** The device variant (last letter of the ordering number) is independent of the die revision (DSU.DID.REVISION): The device variant denotes functional differences, whereas the die revision marks evolution of the die.

**Bits 7:0 – DEVSEL[7:0]** Device Selection
This bit field identifies a device within a product family and product series.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.10. CoreSight ROM Table Entry 0

**Name:** ENTRY0
**Offset:** 0x1000
**Reset:** 0xXXXXX00X
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[19:12] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[11:4] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[3:0] | | | | | | | |
| Access | R | R | R | R | | | | |
| Reset | x | x | x | x | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | FMT | EPRES |
| Access | | | | | | | R | R |
| Reset | | | | | | | 1 | x |

**Bits 31:12 – ADDOFF[19:0]** Address Offset
The base address of the component, relative to the base address of this ROM table.

**Bit 1 – FMT** Format
Always reads as '1', indicating a 32-bit ROM table.

**Bit 0 – EPRES** Entry Present
This bit indicates whether an entry is present at this location in the ROM table.
This bit is set at power-up if the device is not protected indicating that the entry is not present.
This bit is cleared at power-up if the device is not protected indicating that the entry is present.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.11. CoreSight ROM Table Entry 1

**Name:** ENTRY1
**Offset:** 0x1004
**Reset:** 0xXXXXX00X
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[19:12] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[11:4] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ADDOFF[3:0] | | | | | | | |
| Access | R | R | R | R | | | | |
| Reset | x | x | x | x | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | FMT | EPRES |
| Access | | | | | | | R | R |
| Reset | | | | | | | 1 | x |

**Bits 31:12 – ADDOFF[19:0]** Address Offset
The base address of the component, relative to the base address of this ROM table.

**Bit 1 – FMT** Format
Always read as '1', indicating a 32-bit ROM table.

**Bit 0 – EPRES** Entry Present
This bit indicates whether an entry is present at this location in the ROM table.
This bit is set at power-up if the device is not protected indicating that the entry is not present.
This bit is cleared at power-up if the device is not protected indicating that the entry is present.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.12. CoreSight ROM Table End

**Name:** END
**Offset:** 0x1008
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | END[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | END[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | END[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | END[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – END[31:0]** End Marker
Indicates the end of the CoreSight ROM table entries.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.13. CoreSight ROM Table Memory Type

**Name:** MEMTYPE
**Offset:** 0x1FCC
**Reset:** 0x0000000x
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SMEMP |
| Access | | | | | | | | R |
| Reset | | | | | | | | x |

**Bit 0 – SMEMP** System Memory Present

This bit indicates whether system memory is present on the bus that connects to the ROM table.
This bit is set at power-up if the device is not protected, indicating that the system memory is accessible from a debug adapter.
This bit is cleared at power-up if the device is protected, indicating that the system memory is not accessible from a debug adapter.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.14. Peripheral Identification 4

**Name:** PID4
**Offset:** 0x1FD0
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | FKBC[3:0] | | | | JEPCC[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

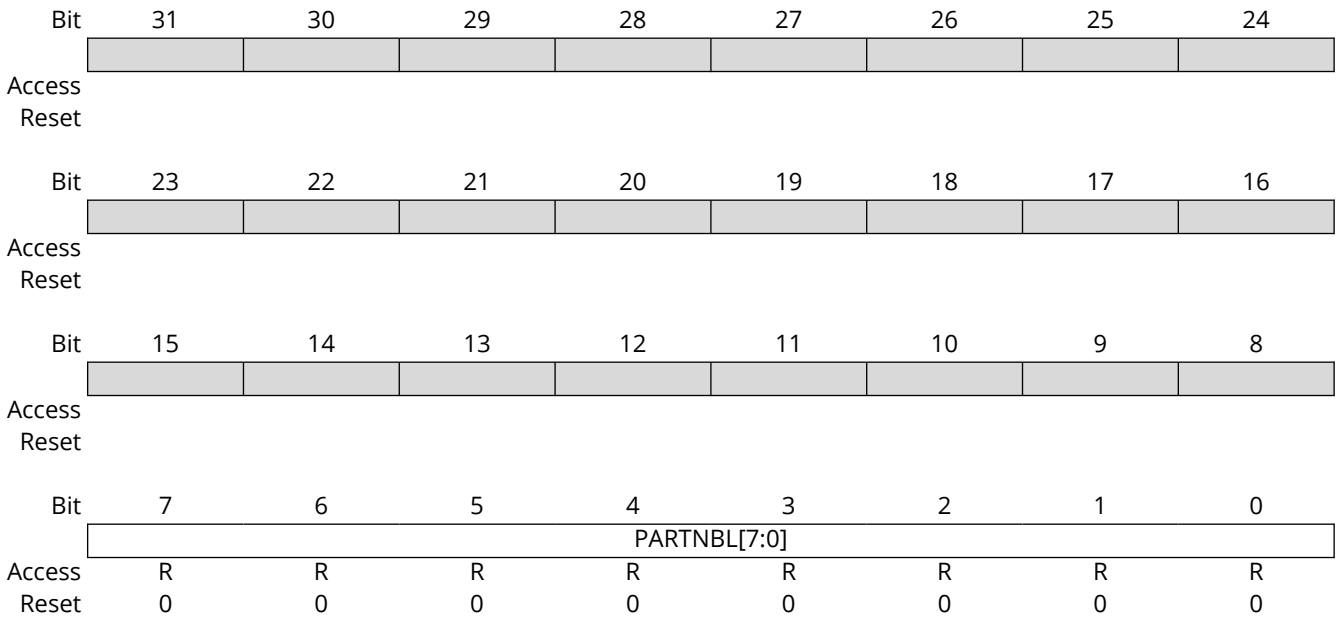
**Bits 7:4 – FKBC[3:0]** 4KB Count
These bits will always return zero when read, indicating that this debug component occupies one 4KB block.

**Bits 3:0 – JEPCC[3:0]** JEP-106 Continuation Code
These bits will always return zero when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.15. Peripheral Identification 0

**Name:** PID0
**Offset:** 0x1FE0
**Reset:** 0x00000000
**Property:** -




**Bits 7:0 – PARTNBL[7:0]** Part Number Low
These bits will always return 0xD0 when read, indicating that this device implements a DSU module instance.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.16. Peripheral Identification 1

**Name:** PID1
**Offset:** 0x1FE4
**Reset:** 0x000000FC
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | JEPIDCL[3:0] | | | | PARTNBH[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |

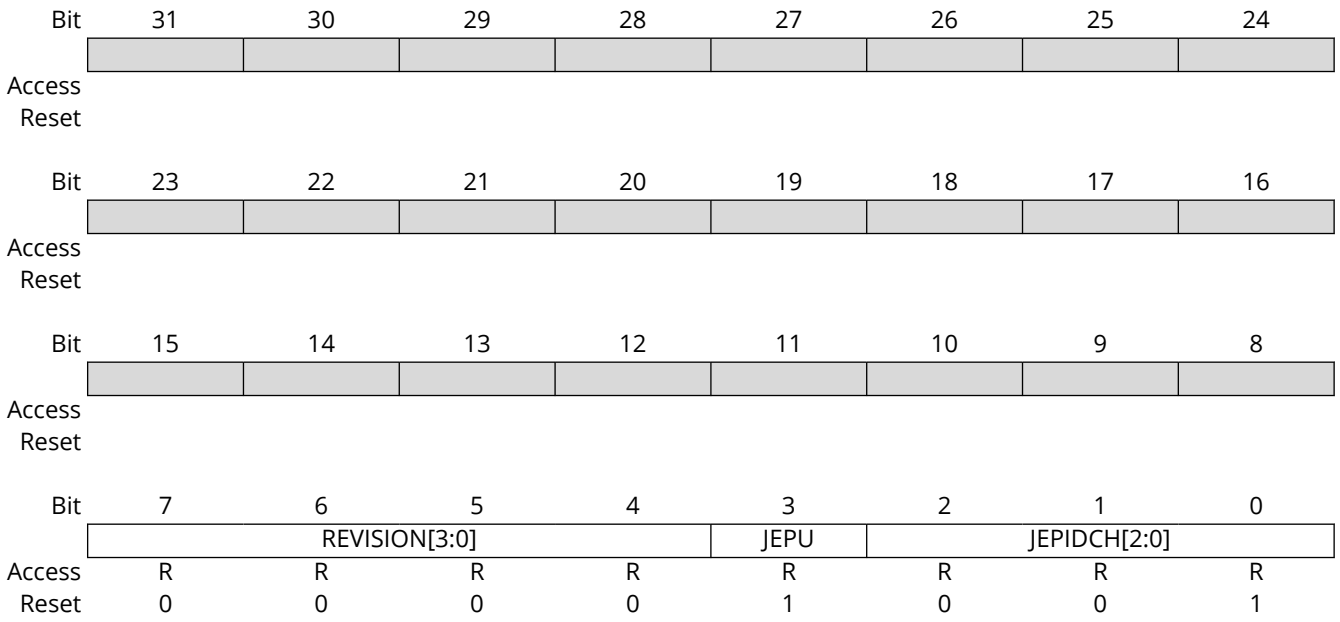
**Bits 7:4 – JEPIDCL[3:0]** Low part of the JEP-106 Identity Code
These bits will always return 0xF when read (JEP-106 identity code is 0x1F).

**Bits 3:0 – PARTNBH[3:0]** Part Number High
These bits will always return 0xC when read, indicating that this device implements a DSU module instance.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.17. Peripheral Identification 2

**Name:** PID2
**Offset:** 0x1FE8
**Reset:** 0x00000009
**Property:** -



| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | REVISION[3:0] | | | | JEPU | JEPIDCH[2:0] | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

**Bits 7:4 – REVISION[3:0]** Revision Number
Revision of the peripheral. Starts at 0x0 and increments by one at both major and minor revisions.

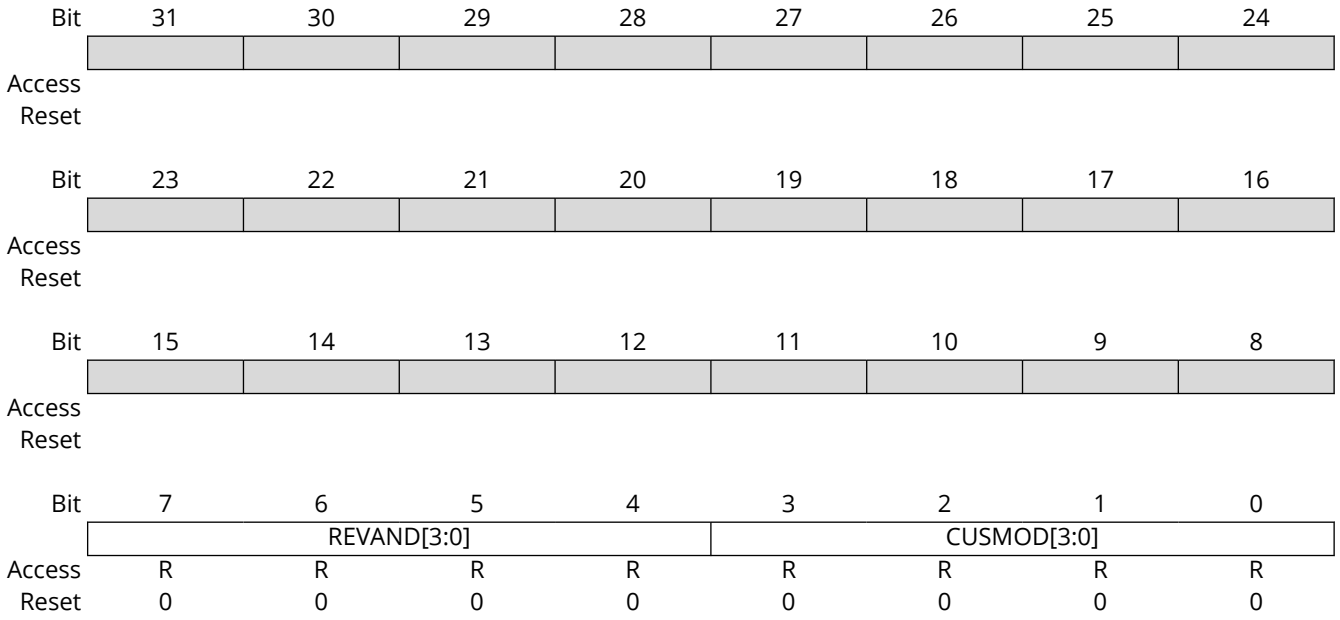
**Bit 3 – JEPU** JEP-106 Identity Code is used
This bit will always return one when read, indicating that JEP-106 code is used.

**Bits 2:0 – JEPIDCH[2:0]** JEP-106 Identity Code High
These bits will always return 0x1 when read, (JEP-106 identity code is 0x1F).


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.18. Peripheral Identification 3

**Name:** PID3
**Offset:** 0x1FEC
**Reset:** 0x00000000
**Property:** -




**Bits 7:4 – REVAND[3:0]** Revision Number
These bits will always return 0x0 when read.

**Bits 3:0 – CUSMOD[3:0]** ARM CUSMOD
These bits will always return 0x0 when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.19. Component Identification 0

**Name:** CID0
**Offset:** 0x1FF0
**Reset:** 0x0000000D
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PREAMBLEB0[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 |

**Bits 7:0 – PREAMBLEB0[7:0]** Preamble Byte 0
These bits will always return 0x0000000D when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.20. Component Identification 1

**Name:** CID1
**Offset:** 0x1FF4
**Reset:** 0x00000010
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CCLASS[3:0] | | | | PREAMBLE[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

**Bits 7:4 – CCLASS[3:0]** Component Class
These bits will always return 0x1 when read indicating that this ARM CoreSight component is ROM table (refer to the ARM Debug Interface v5 Architecture Specification at http://www.arm.com).

**Bits 3:0 – PREAMBLE[3:0]** Preamble
These bits will always return 0x00 when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.21. Component Identification 2

**Name:** CID2
**Offset:** 0x1FF8
**Reset:** 0x00000005
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PREAMBLEB2[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

**Bits 7:0 – PREAMBLEB2[7:0]** Preamble Byte 2
These bits will always return 0x00000005 when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 13.14.22. Component Identification 3

**Name:** CID3
**Offset:** 0x1FFC
**Reset:** 0x000000B1
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PREAMBLEB3[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

**Bits 7:0 – PREAMBLEB3[7:0]** Preamble Byte 3
These bits will always return 0x000000B1 when read.


© 2026 Microchip Technology Inc. and its subsidiaries

# 14. DIVAS – Divide and Square Root Accelerator

## 14.1. Overview

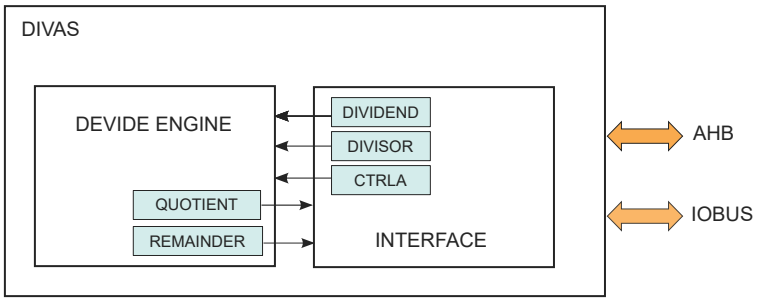
The Divide and Square Root Accelerator (DIVAS) is a programmable 32-bit signed or unsigned hardware divider and a 32-bit unsigned square root hardware engine. The DIVAS is connected to the high-speed bus matrix and may also be accessed using the low-latency CPU local bus (IOBUS; ARM® single-cycle I/O port). The DIVAS takes dividend and divisor values and returns the quotient and remainder when it is used as divider. The DIVAS takes unsigned input value and returns its square root and remainder when it is used as square root function.

## 14.2. Features

- Division accelerator for Cortex-M0+ systems
- 32-bit signed or unsigned integer division
- 32-bit unsigned square root
- 32-bit division in 2-16 cycles
- Programmable leading zero optimization
- Result includes quotient and remainder
- Result includes square root and remainder
- Busy and Divide-by-zero status
- Automatic start of operation when divisor or square root input is loaded

## 14.3. Block Diagram

**Figure 14-1.** DIVAS Block Diagram




## 14.4. Signal Description

Not applicable

## 14.5. Product Dependencies

In order to use this peripherial, other parts of the system must be configured correctly, as described below.

### 14.5.1. I/O Lines

Not applicable

### 14.5.2. Power Management

The DIVAS will not operate in any sleep mode .


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.5.3. Clocks

The DIVAS bus clock (CLK_DIVAS_AHB) can be enabled and disabled in the power manager, and the default state of CLK_DIVAS_AHB can be found in the Peripheral Clock Masking section in the Power Manager chapter.

### 14.5.4. DMA

Not applicable

### 14.5.5. Interrupts

Not applicable

### 14.5.6. Events

Not applicable

### 14.5.7. Debug Operation

Not applicable

### 14.5.8. Register Access Protection

Certain registers cannot be modified while DIVAS is busy. The following registers are write-protected while busy:

- Control A (CTRLA)
- Dividend (DIVIDEND)
- Divisor (DIVISOR)
- Square Root Input (SQRNUM)

Accessing these registers while protected will result in an error.

### 14.5.9. Analog Connections

Not applicable

### 14.5.10. CPU Local Bus

The CPU local bus (IOBUS) is an interface that connects the CPU directly to the DIVAS. It is a single-cycle bus interface, and does not support wait states. It supports byte, half word and word sizes. This bus is generally used for low latency. All registers can be read and written using this bus.

Since the IOBUS cannot wait for DIVAS to complete operation, the Quotient and Remainder registers must be only be read via the IOBUS while the Busy bit in the Status register (STATUS.BUSY) is zero to prevent incorrect data from being read.

## 14.6. Functional Description

### 14.6.1. Principle of Operation

The Divide and Square Root Accelerator (DIVAS) supports signed or unsigned hardware division of 32-bit values and unsigned square root of 32-bit value. It is accessible from the CPU via both the AHB bus and IOBUS. When the dividend and divide registers are programmed, the division starts and the result will be stored in the Result and Remainder registers. The Busy and Divide-by-zero status can be read from STATUS register.

When the square root input register (SQRNUM) is programmed, the square root function starts and the result will be stored in the Result and Remainder registers. The Busy status can be read from STATUS register.


© 2026 Microchip Technology Inc. and its subsidiaries

## 14.6.2. Basic Operation

### 14.6.2.1. Initialization

The DIVAS configuration cannot be modified while a divide operation is ongoing. The following bits must be written prior to starting a division:

- Sign selection bit in Control A register (CTRLA.SIGNED)
- Leading zero mode bit in Control A register (CTRLA.DLZ)

### 14.6.2.2. Performing Division

First write the dividend to DIVIDEND register. Writing the divisor to DIVISOR register starts the division and sets the busy bit in the Status register (STATUS.BUSY). When the division has completed, the STATUS.BUSY bit is cleared and the result will be stored in RESULT and REMAINDER registers.

The RESULT and REMAINDER registers can be read directly via the high-speed bus without checking first STATUS.BUSY. Wait states will be inserted on the high-speed bus until the operation is complete. The IOBUS does not support wait states. For accesses via the IOBUS, the STATUS.BUSY bit must be polled before reading the result from the RESULT and REM registers.

### 14.6.2.3. Operand Size

**Divide**

The DIVAS can perform 32-bit signed and unsigned division and the operation follows the equation as below.

$$RESULT[31:0] = DIVIDEND[31:0]/DIVISOR[31:0]$$

$$REMAINDER[31:0] = DIVIDEND[31:0]\,\%\,DIVISOR[31:0]$$

DIVAS completes 32-bit division in 2-16 cycles.

**Square Root**

The DIVAS can perform 32-bit unsigned division and the operation follows the equation as below.

$$RESULT\left[31:0\right] = \sqrt{SQRNUM[31:0]}$$

$$REMAINDER\left[31:0\right] = SQRNUM\left[31:0\right] - RESULT[31:0]^2$$

### 14.6.2.4. Signed Division

When CTRLA.SIGNED is one, both the input and the result will be in 2's complement format. The results of signed division are such that the remainder and dividend have the same sign and the quotient is negative if the dividend and divisor have opposite signs. 16-bit results are sign extended to 32-bits. Note that when the maximum negative number is divided by the minimum negative number, the resulting quotient overflows the signed integer range and will return the maximum negative number with no indication of the overflow. This occurs for 0x80000000 / 0xFFFFFFFF in 32-bit operation and 0x8000 / 0xFFFF in 16-bit operation.

### 14.6.2.5. Divide By Zero

A divide by zero fault occurs if the DIVISOR is programmed to zero. QUOTIENT will be zero and the REMAINDER is equal to DIVIDEND. Divide by zero sets the Divide-by-zero bit in the Status register (STATUS.DBZ) to one. STATUS.DBZ must be cleared by writing a one to it.

### 14.6.2.6. Leading Zero Optimization

Leading zero optimization can reduce the time it takes to complete a division by skipping leading zeros in the DIVIDEND (or leading ones in signed mode). Leading zero optimization is enabled by default and can be disabled by the Disable Leading Zero bit in the Control A register (CTRLA.DLZ). When CTRLA.DLZ is zero, 16-bit division completes in 2-8 cycles and 32-bit division completes in 2-16 cycles, depending on the dividend value. If deterministic timing is required, setting CTRLA.DLZ to one forces 16-bit division to always take 8 cycles and 32-bit division to always take 16 cycles.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 14.6.2.7. Unsigned Square Root

When the square root input register (SQRNUM) is programmed, the square root function starts and the result will be stored in the Result and Remainder registers. The Busy status can be read from STATUS register.


© 2026 Microchip Technology Inc. and its subsidiaries

## 14.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | DLZ | SIGNED |
| 0x01 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | STATUS | 7:0 | | | | | | | DBZ | BUSY |
| 0x05 ... 0x07 | Reserved | | | | | | | | | |
| 0x08 | DIVIDEND | 7:0 | DIVIDEND[7:0] | | | | | | | |
| | | 15:8 | DIVIDEND[15:8] | | | | | | | |
| | | 23:16 | DIVIDEND[23:16] | | | | | | | |
| | | 31:24 | DIVIDEND[31:24] | | | | | | | |
| 0x0C | DIVISOR | 7:0 | DIVISOR[7:0] | | | | | | | |
| | | 15:8 | DIVISOR[15:8] | | | | | | | |
| | | 23:16 | DIVISOR[23:16] | | | | | | | |
| | | 31:24 | DIVISOR[31:24] | | | | | | | |
| 0x10 | RESULT | 7:0 | RESULT[7:0] | | | | | | | |
| | | 15:8 | RESULT[15:8] | | | | | | | |
| | | 23:16 | RESULT[23:16] | | | | | | | |
| | | 31:24 | RESULT[31:24] | | | | | | | |
| 0x14 | REMAINDER | 7:0 | REMAINDER[7:0] | | | | | | | |
| | | 15:8 | REMAINDER[15:8] | | | | | | | |
| | | 23:16 | REMAINDER[23:16] | | | | | | | |
| | | 31:24 | REMAINDER[31:24] | | | | | | | |
| 0x18 | SQRNUM | 7:0 | SQRNUM[7:0] | | | | | | | |
| | | 15:8 | SQRNUM[15:8] | | | | | | | |
| | | 23:16 | SQRNUM[23:16] | | | | | | | |
| | | 31:24 | SQRNUM[31:24] | | | | | | | |

## 14.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DLZ | SIGNED |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – DLZ** Disable Leading Zero Optimization

| Value | Description |
|---|---|
| 0 | Enable leading zero optimization; 32-bit division takes 2-16 cycles. |
| 1 | Disable leading zero optimization; 32-bit division takes 16 cycles. |

**Bit 0 – SIGNED** Signed Division Enable

| Value | Description |
|---|---|
| 0 | Unsigned division. |
| 1 | Signed division. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.2. Status

**Name:** STATUS
**Offset:** 0x04
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DBZ | BUSY |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – DBZ** Disable-By-Zero
Writing a zero to this bit has no effect.
Writing a one to this bit clears DBZ to zero.

| Value | Description |
|---|---|
| 0 | A divide-by-zero fault has not occurred |
| 1 | A divide-by-zero fault has occurred |

**Bit 0 – BUSY** DIVAS Accelerator Busy
This bit is set when a value is written to the DIVISOR or SQRNUM registers.
This bit is cleared when either division or square root function completes and results are ready in the RESULT and REMAINDER registers.

| Value | Description |
|---|---|
| 0 | DIVAS is idle |
| 1 | DIVAS is busy with an ongoing division |


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.3. Dividend

**Name:** DIVIDEND
**Offset:** 0x08
**Reset:** 0x0000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIVIDEND[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIVIDEND[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIVIDEND[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIVIDEND[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIVIDEND[31:0]** Dividend Value

Holds the 32-bit dividend for the divide operation. If the Signed bit in Control A register (CTRLA.SIGNED) is zero, DIVIDEND is unsigned. If CTRLA.SIGNED = 1, DIVIDEND is signed two's complement. Refer to Performing Division, Operand Size and Signed Division.


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.4. Divisor

**Name:** DIVISOR
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIVISOR[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIVISOR[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIVISOR[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIVISOR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIVISOR[31:0]** Divisor Value

Holds the 32-bit divisor for the divide operation. If the Signed bit in Control A register (CTRLA.SIGNED) is zero, DIVISOR is unsigned. If CTRLA.SIGNED = 1, DIVISOR is signed two's complement. Writing the DIVISOR register will start the divide function. Refer to Performing Division, Operand Size and Signed Division.


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.5. Result

**Name:** RESULT
**Offset:** 0x10
**Reset:** 0x0000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – RESULT[31:0]** Result of Operation

Holds the 32-bit result of the last performed operation. For a divide operation this is the quotient. If the Signed bit in Control A register (CTRLA.SIGNED) is zero, the quotient is unsigned. If CTRLA.SIGNED = 1, the quotient is signed two's complement. For a square root operation this is the square root. Refer to Performing Division, Operand Size and Signed Division.


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.6. Remainder

**Name:** REMAINDER
**Offset:** 0x14
**Reset:** 0x0000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | REMAINDER[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | REMAINDER[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | REMAINDER[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | REMAINDER[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – REMAINDER[31:0]** Remainder of Operation

Holds the 32-bit remainder of the last performed operation. For a divide operation this is the division remainder. If the Signed bit in Control A register (CTRLA.SIGNED) is zero, the quotient is unsigned. If CTRLA.SIGNED = 1, the quotient is signed two's complement. For a square root operation this is the square root remainder. Refer to Performing Division, Operand Size and Signed Division.


© 2026 Microchip Technology Inc. and its subsidiaries

### 14.8.7. Square Root Input

**Name:** SQRNUM
**Offset:** 0x18
**Reset:** 0x0000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SQRNUM[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SQRNUM[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SQRNUM[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SQRNUM[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – SQRNUM[31:0]** Square Root Input
Holds the 32-bit unsigned input for the square root operation. Writing the SQRNUM register will start the square root function. Refer to Unsigned Square Root.

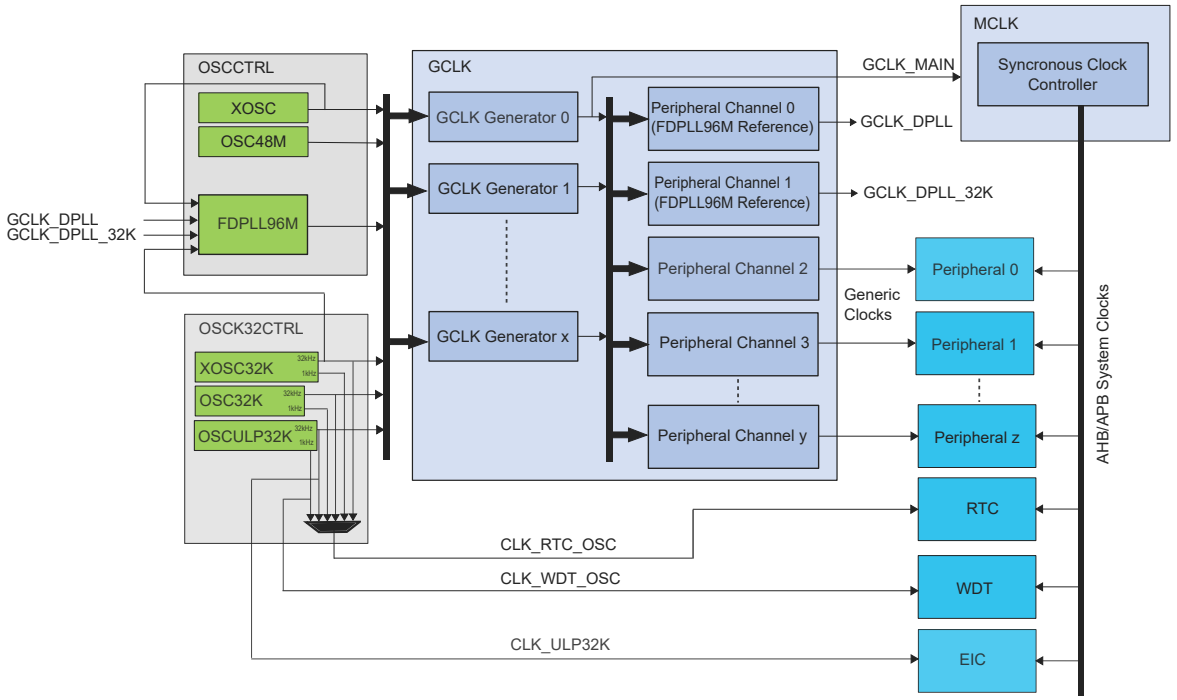

© 2026 Microchip Technology Inc. and its subsidiaries

# 15. Clock System

This chapter only aims to summarize the clock distribution and terminology in the SAM C20/C21 device. It will not explain every detail of its configuration. For in-depth documentation, see the referenced module chapters.

## 15.1. Clock Distribution

**Figure 15-1.** Clock distribution



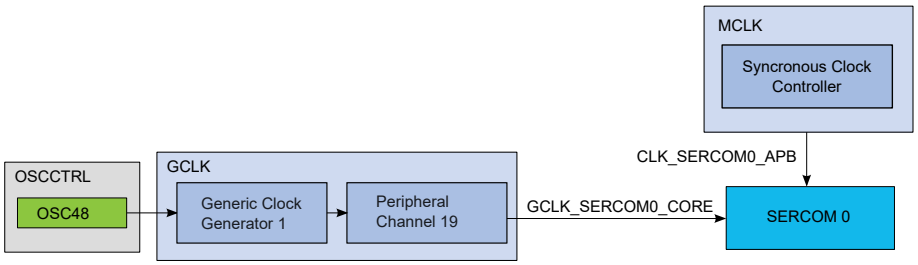

The clock system on the SAM C20/C21 consists of:

- Clock sources, controlled by OSCCTRL and OSC32KCTRL
  - A Clock source is the base clock signal used in the system. Example clock sources are the internal 48 MHz oscillator (OSC48M), External crystal oscillator (XOSC) and the Digital phase locked loop (FDPLL96M).
- Generic Clock Controller (GCLK) which controls the clock distribution system, made up of:
  - Generic Clock generators: A programmable prescaler, that can use any of the system clock sources as its source clock. The Generic Clock Generator 0, also called GCLK_MAIN, is the clock feeding the Power Manager used to generate synchronous clocks.
  - Generic Clocks: Typically the clock input of a peripheral on the system. The generic clocks, through the Generic Clock Multiplexer, can use any of the Generic Clock generators as its clock source. Multiple instances of a peripheral will typically have a separate generic clock for each instance.
- Main Clock controller (MCLK)
  - The MCLK controls synchronous clocks on the system. This includes the CPU, bus clocks (APB, AHB) as well as the synchronous (to the CPU) user interfaces of the peripherals. It contains clock masks that can turn on/off the user interface of a peripheral as well as prescalers for the CPU and bus clocks.


© 2026 Microchip Technology Inc. and its subsidiaries

The figure below shows an example where SERCOM0 is clocked by the OSC48M. The OSC48M is enabled, the Generic Clock Generator 1 uses the OSCLL48M as its clock source, and the generic clock 19, also called GCLK_SERCOM0_CORE, that is connected to SERCOM0 uses generator 1 as its source. The SERCOM0 interface, clocked by CLK_SERCOM0_APB, has been unmasked in the APBC Mask register in the MCLK.

**Figure 15-2.** Example of SERCOM clock




## 15.2. Synchronous and Asynchronous Clocks

As the CPU and the peripherals can be clocked from different clock sources, possibly with widely different clock speeds, some peripheral accesses by the CPU needs to be synchronized between the different clock domains. In these cases the peripheral includes a SYNCBUSY status register that can be used to check if a sync operation is in progress. As the nature of the synchronization might vary between different peripherals, detailed description for each peripheral can be found in the sub-chapter “synchronization” for each peripheral where this is necessary.

In the datasheet references to synchronous clocks are referring to the CPU and bus clocks, while asynchronous clocks are clock generated by generic clocks.

## 15.3. Register Synchronization

### 15.3.1. Overview

All peripherals are composed of one digital bus interface, which is connected to the APB or AHB bus and clocked using a corresponding synchronous clock, and one core clock, which is clocked using a generic clock. Access between these clock domains must be synchronized. As this mechanism is implemented in hardware the synchronization process takes place even if the different clocks domains are clocked from the same source and on the same frequency. All registers in the bus interface are accessible without synchronization. All core registers in the generic clock domain must be synchronized when written. Some core registers must be synchronized when read. Registers that need synchronization has this denoted in each individual register description.

**Note:** Due to asynchronicity between the main clock domain and the peripheral clock domains, all Read-Synchronized registers must ensure synchronization is completed before accessing them after waking up from standby sleep mode. The respective bit in the Synchronization Busy register (SYNCBUSY) will be cleared when the write-synchronization is complete.

### 15.3.2. General Write-Synchronization

Inside the same module, each core register, denoted by the Write-Synchronized property, use its own synchronization mechanism so that writing to different core registers can be done without waiting for the end of synchronization of previous core register access.

However a second write access to the same core register, while synchronization is on going, is discarded and an error is reported through the PAC. To write again to the same core register in the same module, user must wait for the end of synchronization.

For each core register, that can be written, a synchronization status bit is associated

**Example:**


© 2026 Microchip Technology Inc. and its subsidiaries

REGA, REGB are 8-bit core registers. REGC is 16-bit core register.

| Offset | Register |
|---|---|
| 0x00 | REGA |
| 0x01 | REGB |
| 0x02 | REGC |
| 0x03 | |

Since synchronization is per register, user can write REGA (8-bit access) then immediately write REGB (8-bit access) without error.

User can write REGC (16-bit access) without affecting REGA or REGB. But if user writes REGC in two consecutive 8-bit accesses, second write will be discarded and generate an error.

When user makes a 32-bit access to offset 0x00, all registers are written but REGA, REGB, REGC can be updated at a different time because of independent write synchronization

### 15.3.3. General Read-Synchronization

Before any read of a core register, the user must check that the related bit in SYNCBUSY register is cleared.

Read access to core register is always immediate but the return value is reliable only if a synchronization of this core register is not going.

### 15.3.4. Completion of Synchronization

The user can either poll SYNCBUSY register or use the Synchronization Ready interrupt (if available) to check when the synchronization is complete.

### 15.3.5. Enable Write-Synchronization

Writing to the Enable bit in the Control register (CTRL.ENABLE) will also trigger write-synchronization and set SYNCBUSY.ENABLE. CTRL.ENABLE will read its new value immediately after being written. The Synchronisation Ready interrupt (if available) cannot be used for Enable write-synchronization.

### 15.3.6. Software Reset Write-Synchronization

Writing a one to the Software Reset bit in CTRL (CTRL.SWRST) will also trigger write-synchronization and set SYNCBUSY.SWRST. When writing a one to the CTRL.SWRST bit it will immediately read as one. CTRL.SWRST and SYNCBUSY.SWRST will be cleared by hardware when the peripheral has been reset. Writing a zero to the CTRL.SWRST bit has no effect. The Synchronisation Ready interrupt (if available) cannot be used for Software Reset write-synchronization.

### 15.3.7. Synchronization Delay

The synchronization will delay the write or read access duration by a delay D, given by the equation:

$$5 \cdot P_{\mathrm{GCLK}} + 2 \cdot P_{\mathrm{APB}} < D < 6 \cdot P_{\mathrm{GCLK}} + 3 \cdot P_{\mathrm{APB}}$$

Where $P_{\mathrm{GCLK}}$ is the period of the generic clock and $P_{\mathrm{APB}}$ is the period of the peripheral bus clock. A normal peripheral bus register access duration is $2 \cdot P_{\mathrm{APB}}$.

## 15.4. Enabling a Peripheral

To enable a peripheral clocked by a generic clock, the following parts of the system needs to be configured:

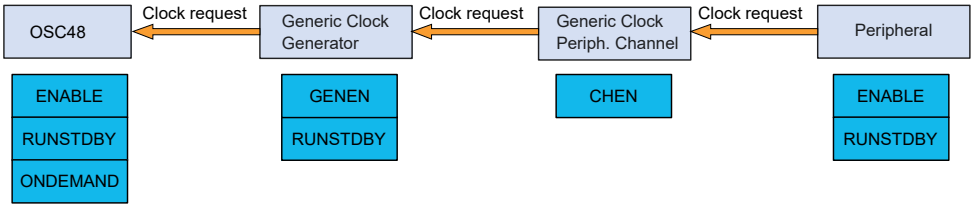
- A running clock source.
- A clock from the Generic Clock Generator must be configured to use one of the running clock sources, and the generator must be enabled.


© 2026 Microchip Technology Inc. and its subsidiaries

- The generic clock, through the Generic Clock Multiplexer, that connects to the peripheral needs to be configured with a running clock from the Generic Clock Generator, and the generic clock must be enabled.
- The user interface of the peripheral needs to be unmasked in the PM. If this is not done the peripheral registers will read as all 0's and any writes to the peripheral will be discarded.

## 15.5. On-demand, Clock Requests

**Figure 15-3.** Clock request routing




All the clock sources in the system can be run in an on-demand mode, where the clock source is in a stopped state when no peripherals are requesting the clock source. Clock requests propagate from the peripheral, via the GCLK, to the clock source. If one or more peripheral is using a clock source, the clock source will be started/kept running. As soon as the clock source is no longer needed and no peripheral have an active request the clock source will be stopped until requested again. For the clock request to reach the clock source, the peripheral, the generic clock and the clock from the Generic Clock Generator in-between must be enabled. The time taken from a clock request being asserted to the clock source being ready is dependent on the clock source startup time, clock source frequency as well as the divider used in the Generic Clock Generator. The total startup time from a clock request to the clock is available for the peripheral is:

```
Delay_start_max = Clock source startup time + 2 * clock source periods + 2 * divided clock
source periods
Delay_start_min = Clock source startup time + 1 * clock source period + 1 * divided clock
source periodDelay_start_min = Clock source startup time + 1 * clock source period + 1 *
divided clock source period
```

The delay for shutting down the clock source when there is no longer an active request is:

```
Delay_stop_min = 1 * divided clock source period + 1 * clock source period
Delay_stop_max = 2 * divided clock source periods + 2 * clock source periods
```

The On-Demand principle can be disabled individually for each clock source by clearing the ONDEMAND bit located in each clock source controller. The clock is always running whatever is the clock request. This has the effect to remove the clock source startup time at the cost of the power consumption.

In standby mode, the clock request mechanism is still working if the modules are configured to run in standby mode (RUNSTDBY bit).

## 15.6. Power Consumption vs. Speed

Due to the nature of the asynchronous clocking of the peripherals there are some considerations that needs to be taken if either targeting a low-power or a fast-acting system. If clocking a peripheral with a very low clock, the active power consumption of the peripheral will be lower. At the same time the synchronization to the synchronous (CPU) clock domain is dependent on the peripheral clock speed, and will be longer with a slower peripheral clock; giving lower response time and more time waiting for the synchronization to complete.

## 15.7. Clocks after Reset

On any reset the synchronous clocks start to their initial state:


© 2026 Microchip Technology Inc. and its subsidiaries

- OSC48M is enabled and divided by 12
- GCLK_MAIN uses OSC48M as source
- CPU and BUS clocks are undivided

On a power reset the GCLK starts to their initial state:

- All generic clock generators disabled except:
  - The generator 0 (GCLK_MAIN) using OSC48M as source, with no division
- All generic clocks disabled

On a user reset the GCLK starts to their initial state, except for:

- Generic clocks that are write-locked (WRTLOCK is written to one prior to reset)

On any reset the clock sources are reset to their initial state except the 32.768 kHz clock sources which are reset only by a power reset.


© 2026 Microchip Technology Inc. and its subsidiaries

# 16. GCLK - Generic Clock Controller

## 16.1. Overview

Depending on the application, peripherals may require specific clock frequencies to operate correctly. The Generic Clock controller (GCLK) features 9 Generic Clock Generators 0..8 that can provide a wide range of clock frequencies.

Generators can be set to use different external and internal oscillators as source. The clock of each Generator can be divided. The outputs from the Generators are used as sources for the Peripheral Channels, which provide the Generic Clocks (GCLK_PERIPH) to the peripheral modules, as shown in Figure 16-2. The number of peripheral clocks depends on how many peripherals the device has.
**Note:** The Generic Clock Generator 0 is always the direct source of the GCLK_MAIN signal.

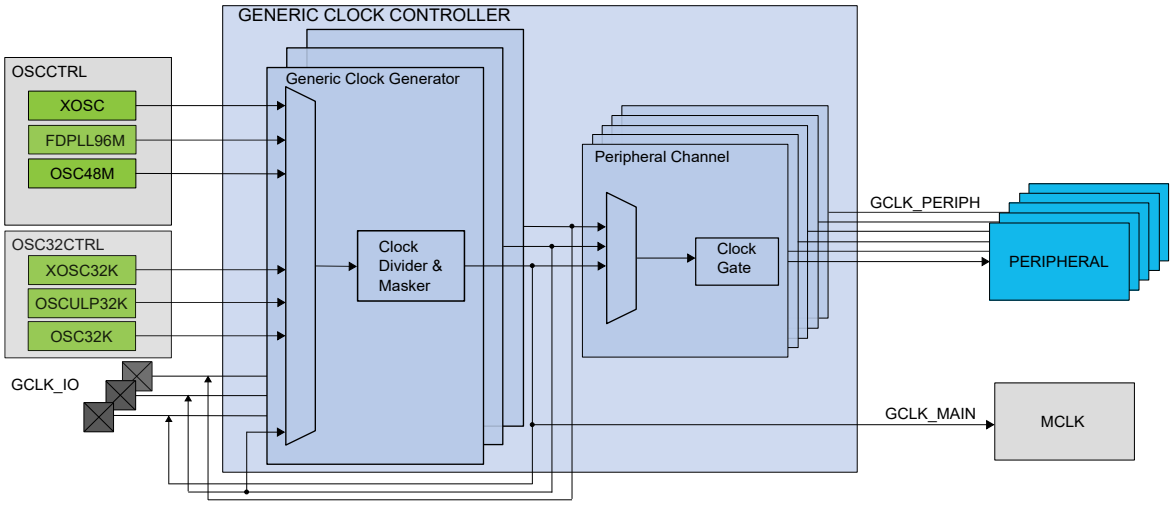
## 16.2. Features

- Provides a device-defined, configurable number of Peripheral Channel clocks
- Wide frequency range:
  - Various clock sources
  - Embedded dividers

## 16.3. Block Diagram

The generation of Peripheral Clock signals (GCLK_PERIPH) and the Main Clock (GCLK_MAIN) can be seen in Device Clocking Diagram.

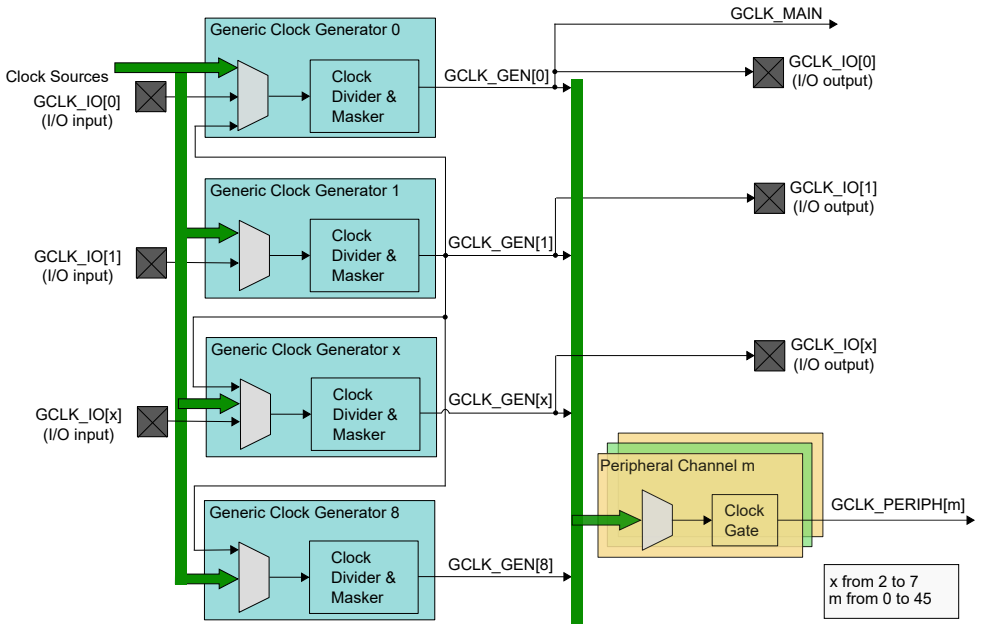
**Figure 16-1.** Device Clocking Diagram




The GCLK block diagram is shown below:


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 16-2.** Generic Clock Controller Block Diagram




## 16.4. Signal Description

**Table 16-1.** GCLK Signal Description

| Signal Name | Type | Description |
|---|---|---|
| GCLK_IO[7:0] | Digital I/O | Clock source for Generators when input<br>Generic Clock signal when output |

**Notes:**

1. One signal can be mapped on several pins.
2. Each GCLK_IO[x] signal is connected to the related Generic Clock Generator x, for x in [7:0]. However,GCLK_IO[8] does not exist, so the Generic Clock Generator 8 is not connected to an external pin.

**Related Links**

I/O Multiplexing and Considerations

## 16.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 16.5.1. I/O Lines

Using the GCLK I/O lines requires the I/O pins to be configured.

**Related Links**

PORT - I/O Pin Controller

### 16.5.2. Power Management

The GCLK can operate in sleep modes, if required. Refer to the sleep mode description in the Power Manager (PM) section.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
PM - Power Manager

### 16.5.3. Clocks

The GCLK bus clock (CLK_GCLK_APB) can be enabled and disabled in the Main Clock Controller (MCLK) and is enabled by default at reset.

**Related Links**
Peripheral Clock Masking
OSC32KCTRL – 32.768 kHz Oscillators Controller

### 16.5.4. DMA

Not applicable.

### 16.5.5. Interrupts

Not applicable.

### 16.5.6. Events

Not applicable.

### 16.5.7. Debug Operation

When the CPU is halted in debug mode the GCLK continues normal operation. If the GCLK is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 16.5.8. Register Access Protection

All registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC).

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 16.5.9. Analog Connections

Not applicable.

## 16.6. Functional Description

### 16.6.1. Principle of Operation

The GCLK module is comprised of nine Generic Clock Generators (Generators) sourcing up to 46 Peripheral Channels and the Main Clock signal GCLK_MAIN.

A clock source selected as input to a Generator can either be used directly or it can be prescaled in the Generator. A generator output is used by one or more Peripheral Channels to provide a peripheral generic clock signal (GCLK_PERIPH) to the peripherals.

### 16.6.2. Basic Operation

#### 16.6.2.1. Initialization

Before a Generator is enabled, the corresponding clock source must be enabled. The Peripheral clock must be configured as outlined by the following steps:


© 2026 Microchip Technology Inc. and its subsidiaries

1. The Generator must be enabled (GENCTRLn.GENEN=1) and the division factor must be set (GENTRLn.DIVSEL and GENCTRLn.DIV) by performing a single 32-bit write to the Generator Control register (GENCTRLn).
2. The Generic Clock for a peripheral must be configured by writing to the respective Peripheral Channel Control register (PCHCTRLm). The Generator used as the source for the Peripheral Clock must be written to the GEN bit field in the Peripheral Channel Control register (PCHCTRLm.GEN).

**Note:** Each Generator n is configured by one dedicated register GENCTRLn.

**Note:** Each Peripheral Channel m is configured by one dedicated register PCHCTRLm. See PCHCTRLm Mapping for the mapping of the peripheral to index m.

### 16.6.2.2. Enabling, Disabling, and Resetting

The GCLK module has no enable/disable bit to enable or disable the whole module.

The GCLK is reset by setting the Software Reset bit in the Control A register (CTRLA.SWRST) to 1. All registers in the GCLK will be reset to their initial state, except for Peripheral Channels and associated Generators that have their Write Lock bit set to 1 (PCHCTRLm.WRTLOCK). For further details, refer to Configuration Lock.

### 16.6.2.3. Generic Clock Generator

Each Generator (GCLK_GEN) can be set to run from one of nine different clock sources except GCLK_GEN[1], which can be set to run from one of eight sources. GCLK_GEN[1] is the only Generator that can be selected as source to others Generators.

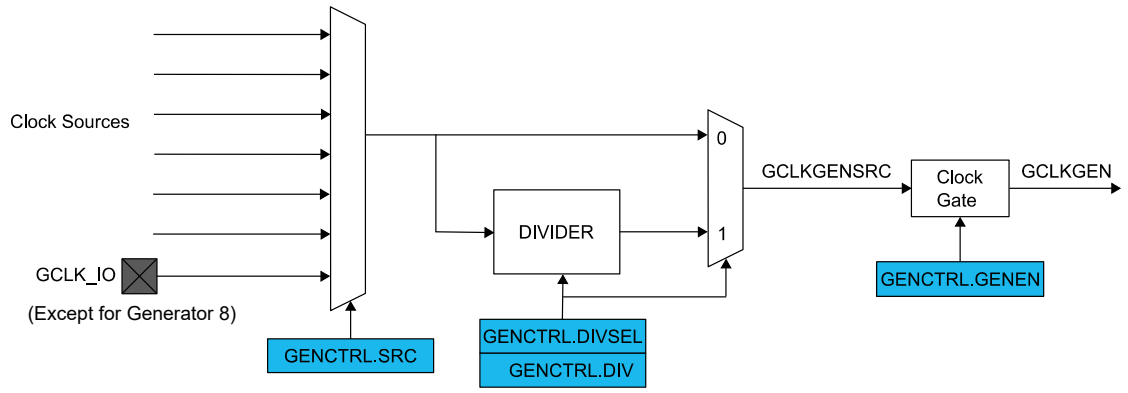
Each generator GCLK_GEN[x] (except GCLK_GEN[8]) can be connected to one specific pin GCLK_IO[x]. A pin GCLK_IO[x] can be set either to act as source to GCLK_GEN[x] or to output the clock signal generated by GCLK_GEN[x].

The selected source can be divided. Each Generator can be enabled or disabled independently.

Each GCLK_GEN clock signal can then be used as clock source for Peripheral Channels. Each Generator output can be allocated to one or more Peripherals.

GCLK_GEN[0] is used as GCLK_MAIN for the synchronous clock controller inside the Main Clock Controller. Refer to the Main Clock Controller description for details on the synchronous clock generation.

**Figure 16-3.** Generic Clock Generator




**Related Links**

MCLK – Main Clock


© 2026 Microchip Technology Inc. and its subsidiaries

#### 16.6.2.4. Enabling a Generator

A Generator is enabled by writing a '1' to the Generator Enable bit in the Generator Control register (GENCTRLn.GENEN=1).

#### 16.6.2.5. Disabling a Generator

A Generator is disabled by writing a '0' to GENCTRLn.GENEN. When GENCTRLn.GENEN=0, the GCLK_GEN[n] clock is disabled and gated.

#### 16.6.2.6. Selecting a Clock Source for the Generator

Each Generator can individually select a clock source by setting the Source Select bit group in the Generator Control register (GENCTRLn.SRC).

Changing from one clock source, for example A, to another clock source, B, can be done on the fly: If clock source B is not ready, the Generator will continue using clock source A. As soon as source B is ready, the Generator will switch to it. During the switching operation, the Generator maintains clock requests to both clock sources A and B, and will release source A as soon as the switch is done. The according bit in SYNCBUSY register (SYNCBUSY.GENCTRLn) will remain '1' until the switch operation is completed.

Before switching the Generic Clock Generator 0 (GCLKGEN0) from a clock source A to another clock source B, enable the ONDEMAND feature of the clock source A to ensure a proper transition from clock source A to clock source B.

Only Generator 1 can be used as a common source for all other generators.

#### 16.6.2.7. Changing the Clock Frequency

The selected source for a Generator can be divided by writing a division value in the Division Factor bit field of the Generator Control register (GENCTRLn.DIV). How the actual division factor is calculated is depending on the Divide Selection bit (GENCTRLn.DIVSEL).

If GENCTRLn.DIVSEL=0 and GENCTRLn.DIV is either 0 or 1, the output clock will be undivided.

**Note:** The number of available DIV bits may vary from Generator to Generator.

#### 16.6.2.8. Duty Cycle

When dividing a clock with an odd division factor, the duty-cycle will not be 50/50. Setting the Improve Duty Cycle bit of the Generator Control register (GENCTRLn.IDC) will result in a 50/50 duty cycle.

#### 16.6.2.9. External Clock

The output clock (GCLK_GEN) of each Generator can be sent to I/O pins (GCLK_IO).

If the Output Enable bit in the Generator Control register is set (GENCTRLn.OE = 1) and the generator is enabled (GENCTRLn.GENEN=1), the Generator requests its clock source and the GCLK_GEN clock is output to an I/O pin.
**Note:** The I/O pin (GCLK/IO[n]) must first be configured as output by writing the corresponding PORT registers.

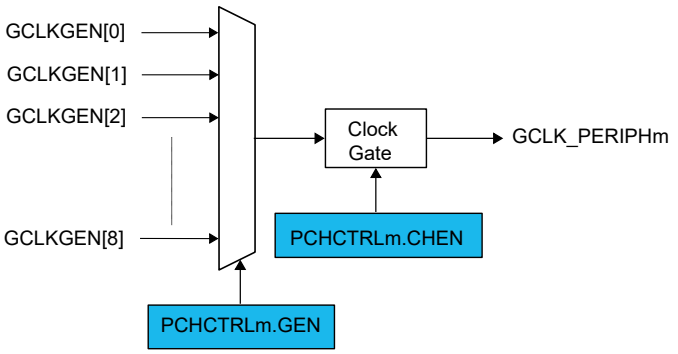
If GENCTRLn.OE is 0, the according I/O pin is set to an Output Off Value, which is selected by GENCTRLn.OOV: If GENCTRLn.OOV is '0', the output clock will be low. If this bit is '1', the output clock will be high.

In Standby mode, if the clock is output (GENCTRLn.OE=1), the clock on the I/O pin is frozen to the OOV value if the Run In Standby bit of the Generic Control register (GENCTRLn.RUNSTDBY) is zero. If GENCTRLn.RUNSTDBY is '1', the GCLKGEN clock is kept running and output to the I/O pin.


© 2026 Microchip Technology Inc. and its subsidiaries

### 16.6.3. Peripheral Clock

**Figure 16-4.** Peripheral Clock




#### 16.6.3.1. Enabling a Peripheral Clock

Before a Peripheral Clock is enabled, one of the Generators must be enabled (GENCTRLn.GENEN) and selected as source for the Peripheral Channel by setting the Generator Selection bits in the Peripheral Channel Control register (PCHCTRLm.GEN). Any available Generator can be selected as clock source for each Peripheral Channel. (See PCHCTRLm Mapping for the mapping of the peripheral to index m.)

When a Generator has been selected, the peripheral clock is enabled by setting the Channel Enable bit in the Peripheral Channel Control register, PCHCTRLm.CHEN = 1. The PCHCTRLm.CHEN bit must be synchronized to the generic clock domain. PCHCTRLm.CHEN will continue to read as its previous state until the synchronization is complete.

#### 16.6.3.2. Disabling a Peripheral Clock

A Peripheral Clock is disabled by writing PCHCTRLm.CHEN=0. The PCHCTRLm.CHEN bit must be synchronized to the Generic Clock domain. PCHCTRLm.CHEN will stay in its previous state until the synchronization is complete. The Peripheral Clock is gated when disabled.

**Related Links**

PCHCTRLm

Peripheral Channel Control

#### 16.6.3.3. Selecting the Clock Source for a Peripheral

When changing a peripheral clock source by writing to PCHCTRLm.GEN, the peripheral clock must be disabled before re-enabling it with the new clock source setting. This prevents glitches during the transition:

1. Disable the Peripheral Channel by writing PCHCTRLm.CHEN=0
2. Assert that PCHCTRLm.CHEN reads '0'
3. Change the source of the Peripheral Channel by writing PCHCTRLm.GEN
4. Re-enable the Peripheral Channel by writing PCHCTRLm.CHEN=1

**Related Links**

PCHCTRLm

Peripheral Channel Control

#### 16.6.3.4. Configuration Lock

The peripheral clock configuration can be locked for further write accesses by setting the Write Lock bit in the Peripheral Channel Control register PCHCTRLm.WRTLOCK=1). After this, all writing to the PCHCTRLm register will be ignored. It can only be unlocked by a Power Reset.


© 2026 Microchip Technology Inc. and its subsidiaries

The Generator source of a locked Peripheral Channel will be locked, too: The corresponding GENCTRLn register is locked, and can be unlocked only by a Power Reset.

The Generator 0 is one exception concerning the configuration lock. As it is used as GCLK_MAIN, it cannot be locked. It is reset by any Reset and will start up in a known configuration. The software reset (CTRLA.SWRST) can not unlock the registers.

In case of an external Reset, the Generator source will be disabled. Even if the WRTLOCK bit is written to '1' the peripheral channels are disabled (PCHCTRLm.CHEN set to '0') until the Generator source is enabled again. Then, the PCHCTRLm.CHEN are set to '1' again.

**Related Links**

CTRLA

Control A

## 16.6.4. Additional Features

### 16.6.4.1. Peripheral Clock Enable after Reset

The Generic Clock Controller must be able to provide a generic clock to some specific peripherals after a Reset. That means that the configuration of the Generators and Peripheral Channels after Reset is device-dependent.

Refer to GENCTRLn.SRC for details on GENCTRLn reset.

Refer to PCHCTRLm.SRC for details on PCHCTRLm reset.

## 16.6.5. Sleep Mode Operation

### 16.6.5.1. SleepWalking

The GCLK module supports the SleepWalking feature.

If the system is in a sleep mode where the Generic Clocks are stopped, a peripheral that needs its clock in order to execute a process must first request it from the Generic Clock Controller.

The Generic Clock Controller receives this request, determines which Generic Clock Generator is involved and which clock source needs to be awakened. It then wakes up the respective clock source, enables the Generator and Peripheral Channel stages successively, and delivers the clock to the peripheral.

The RUNSTDBY bit in the Generator Control register controls clock output to pin during standby sleep mode. If the bit is cleared, the Generator output is not available on pin. When set, the GCLK can continuously output the generator output to the GCLK_IO[n] pin. Refer to External Clock for details.

**Related Links**

PM - Power Manager

### 16.6.5.2. Minimize Power Consumption in Standby

The following table identifies when a Clock Generator is off in Standby Mode, minimizing the power consumption:

**Table 16-2.** Clock Generator n Activity in Standby Mode

| Request for Clock n present | GENCTRLn.RUNSTDBY | GENCTRLn.OE | Clock Generator n |
|---|---|---|---|
| yes | - | - | active |
| no | 1 | 1 | active |
| no | 1 | 0 | OFF |
| no | 0 | 1 | OFF |
| no | 0 | 0 | OFF |


© 2026 Microchip Technology Inc. and its subsidiaries

#### 16.6.5.3. Entering Standby Mode

There may occur a delay when the device is put into Standby, before the power is turned off. This delay is caused by running Clock Generators: if the Run in Standby bit in the Generator Control register (GENCTRLn.RUNSTDBY) is '0', GCLK must verify that the clock is turned of properly. The duration of this verification is frequency-dependent.

**Related Links**

PM - Power Manager

### 16.6.6. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

An exception is the Channel Enable bit in the Peripheral Channel Control registers (PCHCTRLm.CHEN). When changing this bit, the bit value must be read-back to ensure the synchronization is complete and to assert glitch free internal operation. Note that changing the bit value under ongoing synchronization will *not* generate an error.

The following registers are synchronized when written:

- Generic Clock Generator Control register (GENCTRLn)
- Control A register (CTRLA)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

**Related Links**

CTRLA

Control A

Register Synchronization

PCHCTRLm

Peripheral Channel Control


© 2026 Microchip Technology Inc. and its subsidiaries

## 16.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | | SWRST |
| 0x01<br>...<br>0x03 | Reserved | | | | | | | | | |
| 0x04 | SYNCBUSY | 7:0 | GENCTRL5 | GENCTRL4 | GENCTRL3 | GENCTRL2 | GENCTRL1 | GENCTRL0 | | SWRST |
| | | 15:8 | | | | | | GENCTRL8 | GENCTRL7 | GENCTRL6 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08<br>...<br>0x1F | Reserved | | | | | | | | | |
| 0x20 | GENCTRL0 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x24 | GENCTRL1 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x28 | GENCTRL2 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x2C | GENCTRL3 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x30 | GENCTRL4 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x34 | GENCTRL5 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x38 | GENCTRL6 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x3C | GENCTRL7 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x40 | GENCTRL8 | 7:0 | | | | SRC[4:0] | | | | |
| | | 15:8 | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | DIV[15:8] | | | | | | | |
| 0x44<br>...<br>0x7F | Reserved | | | | | | | | | |
| 0x80 | PCHCTRL0 | 7:0 | WRTLOCK | CHEN | | | GEN[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| ... | | | | | | | | | | |
| 0x0134 | PCHCTRL45 | 7:0 | WRTLOCK | CHEN | | | GEN[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

## 16.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 16.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SWRST |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – SWRST** Software Reset
Writing a zero to this bit has no effect.
Setting this bit to 1 will reset all registers in the GCLK to their initial state after a Power Reset, except for generic clocks and associated Generators that have their WRTLOCK bit in PCHCTRLm set to 1.
Refer to GENCTRL Reset Value for details on GENCTRLn register reset.
Refer to PCHCTRL Reset Value for details on PCHCTRLm register reset.
Due to synchronization, there is a waiting period between setting CTRLA.SWRST and a completed Reset. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing. |
| 1 | A Reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 16.8.2. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | GENCTRL8 | GENCTRL7 | GENCTRL6 |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GENCTRL5 | GENCTRL4 | GENCTRL3 | GENCTRL2 | GENCTRL1 | GENCTRL0 | | SWRST |
| Access | R | R | R | R | R | R | | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |

**Bits 2, 3, 4, 5, 6, 7, 8, 9, 10 – GENCTRL** Generator Control n Synchronization Busy
This bit is cleared when the synchronization of the Generator Control n register (GENCTRLn) between clock domains is complete, or when clock switching operation is complete.
This bit is set when the synchronization of the Generator Control n register (GENCTRLn) between clock domains is started.

**Bit 0 – SWRST** Software Reset Synchronization Busy
This bit is cleared when the synchronization of the CTRLA.SWRST register bit between clock domains is complete.
This bit is set when the synchronization of the CTRLA.SWRST register bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 16.8.3. Generator Control

**Name:** GENCTRLn
**Offset:** 0x20 + n*0x04 [n=0..8]
**Reset:** 0x00000106
**Property:** PAC Write-Protection, Write-Synchronized

GENCTRLn controls the settings of Generic Generator n (n=0..8). The reset value is 0x00000106 for Generator n=0, else 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIV[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIV[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | RUNSTDBY | DIVSEL | OE | OOV | IDC | GENEN |
| Access | | | | | | | | |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | SRC[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 31:16 – DIV[15:0]** Division Factor
These bits represent a division value for the corresponding Generator. The actual division factor used is dependent on the state of DIVSEL. The number of relevant DIV bits for each Generator can be seen in this table. Written bits outside of the specified range will be ignored.

**Table 16-3.** Division Factor Bits

| Generic Clock Generator | Division Factor Bits |
|---|---|
| Generator 0 | 8 division factor bits - DIV[7:0] |
| Generator 1 | 16 division factor bits - DIV[15:0] |
| Generator 2-8 | 8 division factor bits - DIV[7:0] |

**Bit 13 – RUNSTDBY** Run in Standby
This bit is used to keep the Generator running in Standby as long as it is configured to output to a dedicated GCLK_IO pin. If GENCTRLn.OE is zero, this bit has no effect and the generator will only be running if a peripheral requires the clock.

| Value | Description |
|---|---|
| 0 | The Generator is stopped in Standby and the GCLK_IO pin state (one or zero) will be dependent on the setting in GENCTRL.OOV. |
| 1 | The Generator is kept running and output to its dedicated GCLK_IO pin during Standby mode. |

**Bit 12 – DIVSEL** Divide Selection
This bit determines how the division factor of the clock source of the Generator will be calculated from DIV. If the clock source should not be divided, DIVSEL must be 0 and the GENCTRLn.DIV value must be either 0 or 1.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The Generator clock frequency equals the clock source frequency divided by GENCTRLn.DIV. |
| 1 | The Generator clock frequency equals the clock source frequency divided by 2^(N+1), where N is the Division Factor Bits for the selected generator (refer to GENCTRLn.DIV). |

**Bit 11 – OE** Output Enable

This bit is used to output the Generator clock output to the corresponding pin (GCLK_IO), as long as GCLK_IO is not defined as the Generator source in the GENCTRLn.SRC bit field. This feature only applies to GCLK Clock Generators 0 through 7. (GCLK Generator 8 does not have a GCLK_IO pin).

| Value | Description |
|---|---|
| 0 | No Generator clock signal on pin GCLK_IO. |
| 1 | The Generator clock signal is output on the corresponding GCLK_IO, unless GCLK_IO is selected as a generator source in the GENCTRLn.SRC bit field. |

**Bit 10 – OOV** Output Off Value

This bit is used to control the clock output value on pin (GCLK_IO) when the Generator is turned off or the OE bit is zero, as long as GCLK_IO is not defined as the Generator source in the GENCTRLn.SRC bit field. This feature only applies to GCLK Clock Generators 0 through 7. (GCLK Generator 8 does not have a GCLK_IO pin).

| Value | Description |
|---|---|
| 0 | The GCLK_IO will be LOW when generator is turned off or when the OE bit is zero. |
| 1 | The GCLK_IO will be HIGH when generator is turned off or when the OE bit is zero. |

**Bit 9 – IDC** Improve Duty Cycle

This bit is used to improve the duty cycle of the Generator output to 50/50 for odd division factors.

| Value | Description |
|---|---|
| 0 | Generator output clock duty cycle is not balanced to 50/50 for odd division factors. |
| 1 | Generator output clock duty cycle is 50/50. |

**Bit 8 – GENEN** Generator Enable

This bit is used to enable and disable the Generator.

| Value | Description |
|---|---|
| 0 | Generator is disabled. |
| 1 | Generator is enabled. |

**Bits 4:0 – SRC[4:0]** Generator Clock Source Selection

These bits select the Generator clock source, as shown in this table.

**Table 16-4.** Generator Clock Source Selection

| Value | Name | Description |
|---|---|---|
| 0x00 | XOSC | XOSC oscillator output |
| 0x01 | GCLK_IN | Generator input pad (GCLK_IO) |
| 0x02 | GCLK_GEN1 | Generic clock generator 1 output |
| 0x03 | OSCULP32K | OSCULP32K oscillator output |
| 0x04 | OSC32K | OSC32K oscillator output |
| 0x05 | XOSC32K | XOSC32K oscillator output |
| 0x06 | OSC48M | OSC48M oscillator output |
| 0x07 | DPLL96M | DPLL96M output |
| 0x08-0x1F | Reserved | Reserved for future use |

A Power Reset will reset all GENCTRLn registers. the Reset values of the GENCTRLn registers are shown in table below.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 16-5.** GENCTRLn Reset Value after a Power Reset

| GCLK Generator | Reset Value after a Power Reset |
|---|---|
| 0 | 0x00000106 |
| others | 0x00000000 |

A User Reset will reset the associated GENCTRL register unless the Generator is the source of a locked Peripheral Channel (PCHCTRLm.WRTLOCK=1). The reset values of the GENCTRL register are as shown in the table below.

**Table 16-6.** GENCTRLn Reset Value after a User Reset

| GCLK Generator | Reset Value after a User Reset |
|---|---|
| 0 | 0x00000106 |
| others | No change if the generator is used by a Peripheral Channel m with PCHCTRLm.WRTLOCK=1<br>else 0x00000000 |

### Related Links

PCHCTRLm

Peripheral Channel Control


© 2026 Microchip Technology Inc. and its subsidiaries

### 16.8.4. Peripheral Channel Control

**Name:** PCHCTRLm
**Offset:** 0x80 + m*0x04 [m=0..45]
**Reset:** 0x00000000
**Property:** PAC Write-Protection

PCHTRLm controls the settings of Peripheral Channel number m (m=0..45).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WRTLOCK | CHEN | | | GEN[3:0] | | | |
| Access | R/W | R/W | | | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | | | 0 | 0 | 0 | 0 |

**Bit 7 – WRTLOCK** Write Lock
After this bit is set to '1', further writes to the PCHCTRLm register will be discarded. The control register of the corresponding Generator n (GENCTRLn), as assigned in PCHCTRLm.GEN, will also be locked. It can only be unlocked by a Power Reset.
Note that Generator 0 cannot be locked.

| Value | Description |
|---|---|
| 0 | The Peripheral Channel register and the associated Generator register are not locked |
| 1 | The Peripheral Channel register and the associated Generator register are locked |

**Bit 6 – CHEN** Channel Enable
This bit is used to enable and disable a Peripheral Channel.
This bit is synchronized with the GCLK domain. It will continue to read its previous state until the synchronization is complete.

| Value | Description |
|---|---|
| 0 | The Peripheral Channel is disabled |
| 1 | The Peripheral Channel is enabled |

**Bits 3:0 – GEN[3:0]** Generator Selection
This bit field selects the Generator to be used as the source of a peripheral clock, as shown in the table below:

**Table 16-7.** Generator Selection

| Value | Description |
|---|---|
| 0x0 | Generic Clock Generator 0 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 16-7.** Generator Selection (continued)

| Value | Description |
|---|---|
| 0x1 | Generic Clock Generator 1 |
| 0x2 | Generic Clock Generator 2 |
| 0x3 | Generic Clock Generator 3 |
| 0x4 | Generic Clock Generator 4 |
| 0x5 | Generic Clock Generator 5 |
| 0x6 | Generic Clock Generator 6 |
| 0x7 | Generic Clock Generator 7 |
| 0x8 | Generic Clock Generator 8 |
| 0x9 - 0xF | Reserved |

**Table 16-8.** Reset Value after a User Reset or a Power Reset

| Reset | PCHCTRLm.GEN | PCHCTRLm.CHEN | PCHCTRLm.WRTLOCK |
|---|---|---|---|
| Power Reset | 0x0 | 0x0 | 0x0 |
| User Reset | If WRTLOCK= 0<br>: 0x0<br>If WRTLOCK = 1: no change | If WRTLOCK= 0<br>: 0x0<br>If WRTLOCK = 1: no change | No change |

A Power Reset will reset all the PCHCTRLm registers.
A User Reset will reset a PCHCTRL if WRTLOCK=0, or else, the content of that PCHCTRL remains unchanged.
PCHCTRL register Reset values are shown in the table PCHCTRLm Mapping.

**Table 16-9.** PCHCTRLm Mapping

| index(m) | Name | Description |
|---|---|---|
| 0 | GCLK_DPLL | FDPLL96M input clock source for reference |
| 1 | GCLK_DPLL_32K | FDPLL96M 32.768 kHz clock for FDPLL96M internal clock timer |
| 2 | GCLK_EIC | EIC |
| 3 | GCLK_FREQM_MSR | FREQM Measure |
| 4 | GCLK_FREQM_REF | FREQM Reference |
| 5 | GCLK_TSENS | TSENS |
| 6 | GCLK_EVSYS_CHANNEL_0 | EVSYS_CHANNEL_0 |
| 7 | GCLK_EVSYS_CHANNEL_1 | EVSYS_CHANNEL_1 |
| 8 | GCLK_EVSYS_CHANNEL_2 | EVSYS_CHANNEL_2 |
| 9 | GCLK_EVSYS_CHANNEL_3 | EVSYS_CHANNEL_3 |
| 10 | GCLK_EVSYS_CHANNEL_4 | EVSYS_CHANNEL_4 |
| 11 | GCLK_EVSYS_CHANNEL_5 | EVSYS_CHANNEL_5 |
| 12 | GCLK_EVSYS_CHANNEL_6 | EVSYS_CHANNEL_6 |
| 13 | GCLK_EVSYS_CHANNEL_7 | EVSYS_CHANNEL_7 |
| 14 | GCLK_EVSYS_CHANNEL_8 | EVSYS_CHANNEL_8 |
| 15 | GCLK_EVSYS_CHANNEL_9 | EVSYS_CHANNEL_9 |
| 16 | GCLK_EVSYS_CHANNEL_10 | EVSYS_CHANNEL_10 |
| 17 | GCLK_EVSYS_CHANNEL_11 | EVSYS_CHANNEL_11 |
| 18 | GCLK_SERCOM[0,1,2,3,4]_SLOW | SERCOM[0,1,2,3,4]_SLOW |
| 19 | GCLK_SERCOM0_CORE | SERCOM0_CORE |
| 20 | GCLK_SERCOM1_CORE | SERCOM1_CORE |
| 21 | GCLK_SERCOM2_CORE | SERCOM2_CORE |
| 22 | GCLK_SERCOM3_CORE | SERCOM3_CORE |
| 23 | GCLK_SERCOM4_CORE | SERCOM4_CORE |
| 24 | GCLK_SERCOM5_SLOW | SERCOM5_SLOW |
| 25 | GCLK_SERCOM5_CORE | SERCOM5_CORE |
| 26 | GCLK_CAN0 | CAN0 |
| 27 | GCLK_CAN1 | CAN1 |
| 28 | GCLK_TCC0, GCLK_TCC1 | TCC0,TCC1 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 16-9.** PCHCTRLm Mapping (continued)

| index(m) | Name | Description |
|---|---|---|
| 29 | GCLK_TCC2 | TCC2 |
| 30 | GCLK_TC0, GCLK_TC1 | TC0,TC1 |
| 31 | GCLK_TC2, GCLK_TC3 | TC2,TC3 |
| 32 | GCLK_TC4 | TC4 |
| 33 | GCLK_ADC0 | ADC0 |
| 34 | GCLK_ADC1 | ADC1 |
| 35 | GCLK_SDADC | SDADC |
| 36 | GCLK_DAC | DAC |
| 37 | GCLK_PTC | PTC |
| 38 | GCLK_CCL | CCL |
| 39 | - | Reserved |
| 40 | GCLK_AC | AC |
| 41 | GCLK_SERCOM6_CORE | SERCOM6_CORE |
| 42 | GCLK_SERCOM7_CORE | SERCOM7_CORE |
| 43 | GCLK_TC5 | TC5 |
| 44 | GCLK_TC6 | TC6 |
| 45 | GCLK_TC7 | TC7 |


© 2026 Microchip Technology Inc. and its subsidiaries

# 17. MCLK – Main Clock

## 17.1. Overview

The Main Clock (MCLK) controls the synchronous clock generation of the device.

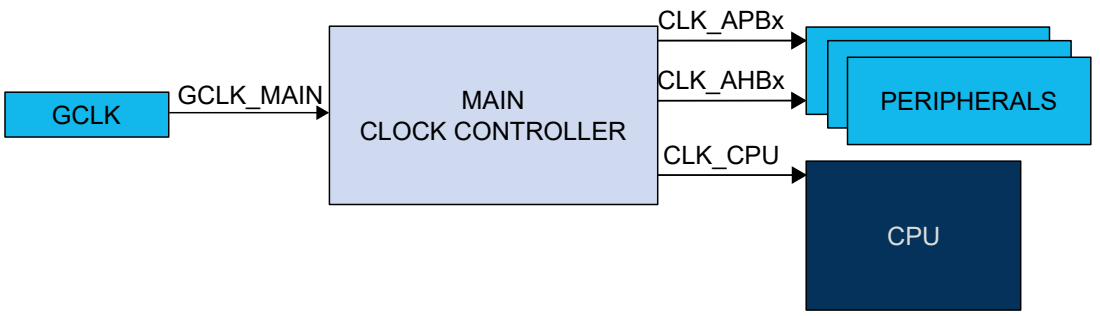
Using a clock provided by the Generic Clock Module (GCLK_MAIN), the Main Clock Controller provides synchronous system clocks to the CPU and the modules connected to the AHBx and the APBx buses. The synchronous system clocks are divided into a number of clock domains. Each clock domain can run at different frequencies, enabling the user to save power by running peripherals at a relatively low clock frequency, while maintaining high CPU performance or vice versa. In addition, the clock can be masked for individual modules, enabling the user to minimize power consumption.

## 17.2. Features

- Generates CPU, AHB, and APB system clocks
  - Clock source and division factor from GCLK
  - Clock prescaler with 1x to 128x division
- Safe run-time clock switching from GCLK
- Module-level clock gating through maskable peripheral clocks

## 17.3. Block Diagram

**Figure 17-1.** MCLK Block Diagram




## 17.4. Signal Description

Not applicable.

## 17.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 17.5.1. I/O Lines

Not applicable.

### 17.5.2. Power Management

The MCLK will operate in all sleep modes if a synchronous clock is required in these modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.5.3. Clocks

The MCLK bus clock (CLK_MCLK_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_MCLK_APB can be found in the Peripheral Clock Masking section. If this clock is disabled, it can only be re-enabled by a reset.

The Generic Clock GCLK_MAIN is required to generate the Main Clocks. GCLK_MAIN is configured in the Generic Clock Controller, and can be re-configured by the user if needed.

**Related Links**
GCLK - Generic Clock Controller
Peripheral Clock Masking

#### 17.5.3.1. Main Clock

The main clock GCLK_MAIN is the common source for the synchronous clocks. This is fed into the common 8-bit prescaler that is used to generate synchronous clocks to the CPU, AHBx, and APBx modules.

#### 17.5.3.2. CPU Clock

The CPU clock (CLK_CPU) is routed to the CPU. Halting the CPU clock inhibits the CPU from executing instructions.

#### 17.5.3.3. APBx and AHBx Clock

The APBx clocks (CLK_APBx) and the AHBx clocks (CLK_AHBx) are the root clock sources used by modules requiring a clock on the APBx and the AHBx buses. These clocks are always synchronous to the CPU clock, and can run even when the CPU clock is turned off in sleep mode. A clock gater is inserted after the common APB clock to gate any APBx clock of a module on APBx bus, as well as the AHBx clock serving that module.

#### 17.5.3.4. Clock Domains

The device has these synchronous clock domains:

- CPU synchronous clock domain (CPU Clock Domain). Frequency is $f_{CPU}$.

See also the related links for the clock domain partitioning.

**Related Links**
Peripheral Clock Masking

### 17.5.4. DMA

Not applicable.

### 17.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the MCLK interrupt requires the Interrupt Controller to be configured first.

### 17.5.6. Events

Not applicable.

### 17.5.7. Debug Operation

When the CPU is halted in debug mode, the MCLK continues normal operation. In sleep mode, the clocks generated from the MCLK are kept running to allow the debugger accessing any module. As a consequence, power measurements are incorrect in debug mode.

### 17.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:


© 2026 Microchip Technology Inc. and its subsidiaries

- Interrupt Flag register (INTFLAG)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 17.5.9. Analog Connections

Not applicable.

## 17.6. Functional Description

### 17.6.1. Principle of Operation

The GCLK_MAIN clock signal from the GCLK module is the source for the main clock, which in turn is the common root for the synchronous clocks for the CPU, APBx, and AHBx modules. The GCLK_MAIN is divided by an 8-bit prescaler. Each of the derived clocks can run from any divided or undivided main clock, ensuring synchronous clock sources for each clock domain. The clock domain (CPU) can be changed on the fly to respond to variable load in the application. The clocks for each module in a clock domain can be masked individually to avoid power consumption in inactive modules. Depending on the sleep mode, some clock domains can be turned off.

### 17.6.2. Basic Operation

#### 17.6.2.1. Initialization

After a Reset, the default clock source of the GCLK_MAIN clock is started and calibrated before the CPU starts running. The GCLK_MAIN clock is selected as the main clock without any prescaler division.

By default, only the necessary clocks are enabled.

**Related Links**
Peripheral Clock Masking

#### 17.6.2.2. Enabling, Disabling, and Resetting

The MCLK module is always enabled and cannot be reset.

#### 17.6.2.3. Selecting the Main Clock Source

Refer to the Generic Clock Controller description for details on how to configure the clock source of the GCLK_MAIN clock.

**Related Links**
GCLK - Generic Clock Controller

#### 17.6.2.4. Selecting the Synchronous Clock Division Ratio

The main clock GCLK_MAIN feeds an 8-bit prescaler, which can be used to generate the synchronous clocks. By default, the synchronous clocks run on the undivided main clock. The user can select a prescaler division for the CPU clock domain by writing the Division (DIV) bits in the CPU Clock Division register CPUDIV, resulting in a CPU clock domain frequency determined by this equation:

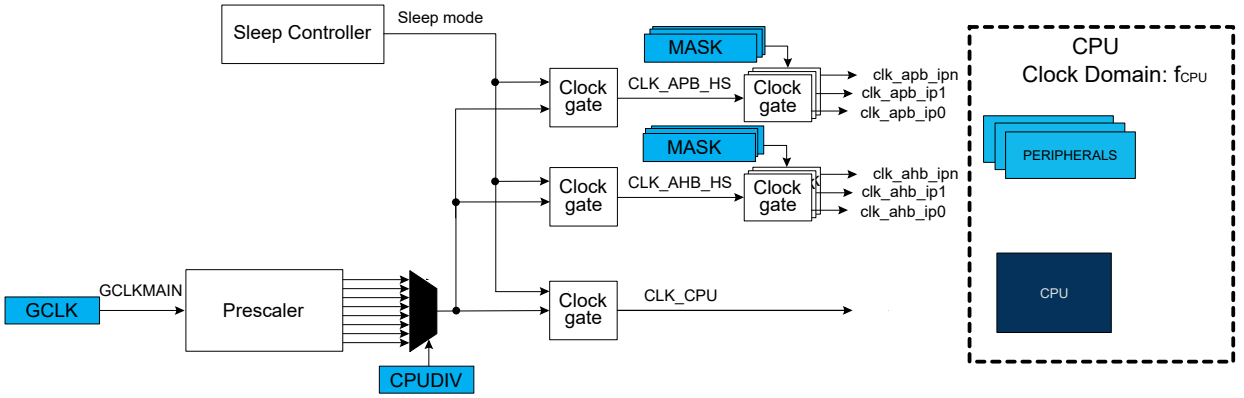
$$f_{CPU} = \frac{f_{main}}{CPUDIV}$$

If the application attempts to write forbidden values in CPUDIV register, registers are written but these bad values are not used and a violation is reported to the PAC module.


© 2026 Microchip Technology Inc. and its subsidiaries

Division bits (DIV) can be written without halting or disabling peripheral modules. Writing DIV bits allows a new clock setting to be written to all synchronous clocks belonging to the corresponding clock domain at the same time.

**Figure 17-2.** Synchronous Clock Selection and Prescaler




### 17.6.2.5. Clock Ready Flag

There is a slight delay between writing to CPUDIV until the new clock settings become effective.

During this interval, the Clock Ready flag in the Interrupt Flag Status and Clear register (INTFLAG.CKRDY) will return zero when read. If CKRDY in the INTENSET register is set to '1', the Clock Ready interrupt will be triggered when the new clock setting is effective.

**Related Links**

PAC - Peripheral Access Controller

### 17.6.2.6. Peripheral Clock Masking

It is possible to disable/enable the AHB or APB clock for a peripheral by writing the corresponding bit in the Clock Mask registers (AHBMASK and APBxMASK) to '0'/'1'. The default state of the peripheral clocks is given by the peripheral bit reset value in AHBMASK and APBxMASK registers.

When the APB clock is not provided to a module, its registers cannot be read or written. The module can be re-enabled later by writing the corresponding mask bit to '1'.

A module may be connected to several clock domains (for instance, AHB and APB), in which case it will have several mask bits.

Note that clocks should only be switched off if it is certain that the module will not be used: Switching off the clock for the NVM Controller (NVMCTRL) will cause a problem if the CPU needs to read from the Flash Memory. Switching off the clock to the MCLK module (which contains the mask registers) or the corresponding APBx bridge, will make it impossible to write the mask registers again. In this case, they can only be re-enabled by a system reset.

## 17.6.3. DMA Operation

Not applicable.

## 17.6.4. Interrupts

The peripheral has the following interrupt sources:

- Clock Ready (CKRDY): indicates that CPU clocks are ready. This interrupt is a synchronous wake-up source.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be enabled individually by writing a '1' to the corresponding enabling bit in the Interrupt


© 2026 Microchip Technology Inc. and its subsidiaries

Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding clearing bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled or the peripheral is reset. An interrupt flag is cleared by writing a '1' to the corresponding bit in the INTFLAG register. Each peripheral can have one interrupt request line per interrupt source or one common interrupt request line for all the interrupt sources. If the peripheral has one common interrupt request line for all the interrupt sources, the user must read the INTFLAG register to determine which interrupt condition is present.

**Related Links**

Overview
Sleep Mode Controller
PM - Power Manager

### 17.6.5. Events

Not applicable.

### 17.6.6. Sleep Mode Operation

In IDLE sleep mode, the MCLK is still running on the selected main clock.

In STANDBY sleep mode, the MCLK is frozen if no synchronous clock is required.


© 2026 Microchip Technology Inc. and its subsidiaries

## 17.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | | |
| 0x01 | INTENCLR | 7:0 | | | | | | | | CKRDY |
| 0x02 | INTENSET | 7:0 | | | | | | | | CKRDY |
| 0x03 | INTFLAG | 7:0 | | | | | | | | CKRDY |
| 0x04 | Reserved | | | | | | | | | |
| 0x05 | CPUDIV | 7:0 | CPUDIV[7:0] | | | | | | | |
| 0x06<br>...<br>0x0F | Reserved | | | | | | | | | |
| 0x10 | AHBMASK | 7:0 | DMAC | HSRAM | NVMCTRL | HMATRIXHS | DSU | APBC | APBB | APBA |
| | | 15:8 | | | APBD | DIVAS | | PAC | CAN1 | CAN0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | APBAMASK | 7:0 | GCLK | SUPC | OSC32KCTRL | OSCCTRL | RSTC | MCLK | PM | PAC |
| | | 15:8 | | | | TSENS | FREQM | EIC | RTC | WDT |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | APBBMASK | 7:0 | | | HMATRIXHS | | | NVMCTRL | DSU | PORT |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | APBCMASK | 7:0 | | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| | | 15:8 | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | |
| | | 23:16 | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| | | 31:24 | | | | | | | | |
| 0x20 | APBDMASK | 7:0 | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 17.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers can be write-protected optionally by the Peripheral Access Controller (PAC). This is denoted by the property "PAC Write-Protection" in each individual register description. Refer to the Register Access Protection for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection

All bits in this register are reserved.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.2. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CKRDY |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – CKRDY** Clock Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Clock Ready Interrupt Enable bit and the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Clock Ready interrupt is enabled and will generate an interrupt request when the Clock Ready Interrupt Flag is set. |
| 1 | The Clock Ready interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.3. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x02
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CKRDY |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – CKRDY** Clock Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Clock Ready Interrupt Enable bit and enable the Clock Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Clock Ready interrupt is disabled. |
| 1 | The Clock Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.4. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x03
**Reset:** 0x01
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CKRDY |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 1 |

**Bit 0 – CKRDY** Clock Ready
This flag is cleared by writing a '1' to the flag.
This flag is set when the synchronous CPU, APBx, and AHBx clocks have frequencies as indicated in the CPUDIV register and will generate an interrupt if INTENCLR/SET.CKRDY is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Clock Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.5. CPU Clock Division

**Name:** CPUDIV
**Offset:** 0x05
**Reset:** 0x01
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CPUDIV[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 7:0 – CPUDIV[7:0]** CPU Clock Division Factor
These bits define the division ratio of the main clock prescaler related to the CPU clock domain. Frequencies must never exceed the specified maximum frequency for each clock domain.

| Value | Name | Description |
|---|---|---|
| 0x01 | DIV1 | Divide by 1 |
| 0x02 | DIV2 | Divide by 2 |
| 0x04 | DIV4 | Divide by 4 |
| 0x08 | DIV8 | Divide by 8 |
| 0x10 | DIV16 | Divide by 16 |
| 0x20 | DIV32 | Divide by 32 |
| 0x40 | DIV64 | Divide by 64 |
| 0x80 | DIV128 | Divide by 128 |
| others | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.6. AHB Mask

**Name:** AHBMASK
**Offset:** 0x10
**Reset:** 0x000003CFF
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | APBD | DIVAS | | PAC | CAN1 | CAN0 |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 1 | 1 | | 1 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DMAC | HSRAM | NVMCTRL | HMATRIXHS | DSU | APBC | APBB | APBA |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bit 13 – APBD** APBD AHB Clock Enable. This bit field is only available for SAMC2x "N" series devices.

| Value | Description |
|---|---|
| 0 | The AHB clock for the APBD is stopped. |
| 1 | The AHB clock for the APBD is enabled. |

**Bit 12 – DIVAS** DIVAS AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the DIVAS is stopped. |
| 1 | The AHB clock for the DIVAS is enabled. |

**Bit 10 – PAC** PAC AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the PAC is stopped. |
| 1 | The AHB clock for the PAC is enabled. |

**Bit 9 – CAN1** CAN1 AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the CAN1 is stopped. |
| 1 | The AHB clock for the CAN1 is enabled. |

**Bit 8 – CAN0** CAN0 AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the CAN0 is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The AHB clock for the CAN0 is enabled. |

**Bit 7 – DMAC** DMAC AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the DMAC is stopped. |
| 1 | The AHB clock for the DMAC is enabled. |

**Bit 6 – HSRAM** HSRAM AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the HSRAM is stopped. |
| 1 | The AHB clock for the HSRAM is enabled. |

**Bit 5 – NVMCTRL** NVMCTRL AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the NVMCTRL is stopped. |
| 1 | The AHB clock for the NVMCTRL is enabled. |

**Bit 4 – HMATRIXHS** HMATRIXHS AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the HMATRIXHS is stopped. |
| 1 | The AHB clock for the HMATRIXHS is enabled. |

**Bit 3 – DSU** DSU AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the DSU is stopped. |
| 1 | The AHB clock for the DSU is enabled. |

**Bit 2 – APBC** APBC AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the APBC is stopped. |
| 1 | The AHB clock for the APBC is enabled |

**Bit 1 – APBB** APBB AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the APBB is stopped. |
| 1 | The AHB clock for the APBB is enabled. |

**Bit 0 – APBA** APBA AHB Clock Enable

| Value | Description |
|---|---|
| 0 | The AHB clock for the APBA is stopped. |
| 1 | The AHB clock for the APBA is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.7. APBA Mask

**Name:** APBAMASK
**Offset:** 0x14
**Reset:** 0x00000FFF
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | TSENS | FREQM | EIC | RTC | WDT |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GCLK | SUPC | OSC32KCTRL | OSCCTRL | RSTC | MCLK | PM | PAC |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bit 12 – TSENS** TSENS APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the TSENS is stopped. |
| 1 | The APBA clock for the TSENS is enabled. |

**Bit 11 – FREQM** FREQM APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the FREQM is stopped. |
| 1 | The APBA clock for the FREQM is enabled. |

**Bit 10 – EIC** EIC APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the EIC is stopped. |
| 1 | The APBA clock for the EIC is enabled. |

**Bit 9 – RTC** RTC APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the RTC is stopped. |
| 1 | The APBA clock for the RTC is enabled. |

**Bit 8 – WDT** WDT APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the WDT is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The APBA clock for the WDT is enabled. |

**Bit 7 – GCLK** GCLK APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the GCLK is stopped. |
| 1 | The APBA clock for the GCLK is enabled. |

**Bit 6 – SUPC** SUPC APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the SUPC is stopped. |
| 1 | The APBA clock for the SUPC is enabled. |

**Bit 5 – OSC32KCTRL** OSC32KCTRL APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the OSC32KCTRL is stopped. |
| 1 | The APBA clock for the OSC32KCTRL is enabled. |

**Bit 4 – OSCCTRL** OSCCTRL APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the OSCCTRL is stopped. |
| 1 | The APBA clock for the OSCCTRL is enabled. |

**Bit 3 – RSTC** RSTC APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the RSTC is stopped. |
| 1 | The APBA clock for the RSTC is enabled. |

**Bit 2 – MCLK** MCLK APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the MCLK is stopped. |
| 1 | The APBA clock for the MCLK is enabled. |

**Bit 1 – PM** PM APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the PM is stopped. |
| 1 | The APBA clock for the PM is enabled. |

**Bit 0 – PAC** PAC APBA Clock Enable

| Value | Description |
|---|---|
| 0 | The APBA clock for the PAC is stopped. |
| 1 | The APBA clock for the PAC is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.8. APBB Mask

**Name:** APBBMASK
**Offset:** 0x18
**Reset:** 0x00000007
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | HMATRIXHS | | | NVMCTRL | DSU | PORT |
| Access | | | R/W | | | R/W | R/W | R/W |
| Reset | | | 0 | | | 1 | 1 | 1 |

**Bit 5 – HMATRIXHS** HMATRIXHS APBB Clock Enable

| Value | Description |
|---|---|
| 0 | The APBB clock for the HMATRIXHS is stopped |
| 1 | The APBB clock for the HMATRIXHS is enabled |

**Bit 2 – NVMCTRL** NVMCTRL APBB Clock Enable

| Value | Description |
|---|---|
| 0 | The APBB clock for the NVMCTRL is stopped |
| 1 | The APBB clock for the NVMCTRL is enabled |

**Bit 1 – DSU** DSU APBB Clock Enable

| Value | Description |
|---|---|
| 0 | The APBB clock for the DSU is stopped |
| 1 | The APBB clock for the DSU is enabled |

**Bit 0 – PORT** PORT APBB Clock Enable

| Value | Description |
|---|---|
| 0 | The APBB clock for the PORT is stopped. |
| 1 | The APBB clock for the PORT is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.9. APBC Mask

**Name:** APBCMASK
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CCL | PTC | DAC | AC | SDADC | ADC1 | ADC0 | TC4 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TC3 | TC2 | TC1 | TC0 | TCC2 | TCC1 | TCC0 | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 | SERCOM1 | SERCOM0 | EVSYS |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 23 – CCL** CCL APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the CCL is stopped. |
| 1 | The APBC clock for the CCL is enabled. |

**Bit 22 – PTC** PTC APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the PTC is stopped. |
| 1 | The APBC clock for the PTC is enabled. |

**Bit 21 – DAC** DAC APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the DAC is stopped. |
| 1 | The APBC clock for the DAC is enabled. |

**Bit 20 – AC** AC APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the AC is stopped. |
| 1 | The APBC clock for the AC is enabled. |

**Bit 19 – SDADC** SDADC APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SDADC is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The APBC clock for the SDADC is enabled. |

**Bit 18 – ADC1** ADC1 APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the ADC1 is stopped. |
| 1 | The APBC clock for the ADC1 is enabled. |

**Bit 17 – ADC0** ADC0 APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the ADC0 is stopped. |
| 1 | The APBC clock for the ADC0 is enabled. |

**Bit 16 – TC4** TC4 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TC4 is stopped. |
| 1 | The APBC clock for the TC4 is enabled. |

**Bit 15 – TC3** TC3 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TC3 is stopped. |
| 1 | The APBC clock for the TC3 is enabled. |

**Bit 14 – TC2** TC2 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TC2 is stopped. |
| 1 | The APBC clock for the TC2 is enabled. |

**Bit 13 – TC1** TC1 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TC1 is stopped. |
| 1 | The APBC clock for the TC1 is enabled. |

**Bit 12 – TC0** TC0 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TC0 is stopped. |
| 1 | The APBC clock for the TC0 is enabled. |

**Bit 11 – TCC2** TCC2 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TCC2 is stopped. |
| 1 | The APBC clock for the TCC2 is enabled. |

**Bit 10 – TCC1** TCC1 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the TCC1 is stopped. |
| 1 | The APBC clock for the TCC1 is enabled. |

**Bit 9 – TCC0** TCC0 APBC Mask Clock Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The APBC clock for the TCC0 is stopped. |
| 1 | The APBC clock for the TCC0 is enabled. |

**Bit 6 – SERCOM5** SERCOM5 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM5 is stopped. |
| 1 | The APBC clock for the SERCOM5 is enabled. |

**Bit 5 – SERCOM4** SERCOM4 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM4 is stopped. |
| 1 | The APBC clock for the SERCOM4 is enabled. |

**Bit 4 – SERCOM3** SERCOM3 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM3 is stopped. |
| 1 | The APBC clock for the SERCOM3 is enabled. |

**Bit 3 – SERCOM2** SERCOM2 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM2 is stopped. |
| 1 | The APBC clock for the SERCOM2 is enabled. |

**Bit 2 – SERCOM1** SERCOM1 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM1 is stopped. |
| 1 | The APBC clock for the SERCOM1 is enabled. |

**Bit 1 – SERCOM0** SERCOM0 APBC Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the SERCOM0 is stopped. |
| 1 | The APBC clock for the SERCOM0 is enabled. |

**Bit 0 – EVSYS** EVSYS APBC Clock Enable

| Value | Description |
|---|---|
| 0 | The APBC clock for the EVSYS is stopped. |
| 1 | The APBC clock for the EVSYS is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 17.8.10. APBD Mask

**Name:** APBDMASK
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** PAC Write-Protection

**Note:** This register is only available for SAMC2x "N" series devices.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | TC7 | TC6 | TC5 | SERCOM7 | SERCOM6 |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bit 4 – TC7** TC7 APBD Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBD clock for the TC7 is stopped. |
| 1 | The APBD clock for the TC7 is enabled. |

**Bit 3 – TC6** TC6 APBD Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBD clock for the TC6 is stopped. |
| 1 | The APBD clock for the TC6 is enabled. |

**Bit 2 – TC5** TC5 APBd Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBD clock for the TC5 is stopped. |
| 1 | The APBD clock for the TC5 is enabled. |

**Bit 1 – SERCOM7** SERCOM7 APBD Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBD clock for the SERCOM7 is stopped. |
| 1 | The APBD clock for the SERCOM7 is enabled. |

**Bit 0 – SERCOM6** SERCOM6 APBD Mask Clock Enable

| Value | Description |
|---|---|
| 0 | The APBD clock for the SERCOM6 is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The APBD clock for the SERCOM6 is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 18. RSTC – Reset Controller

## 18.1. Overview

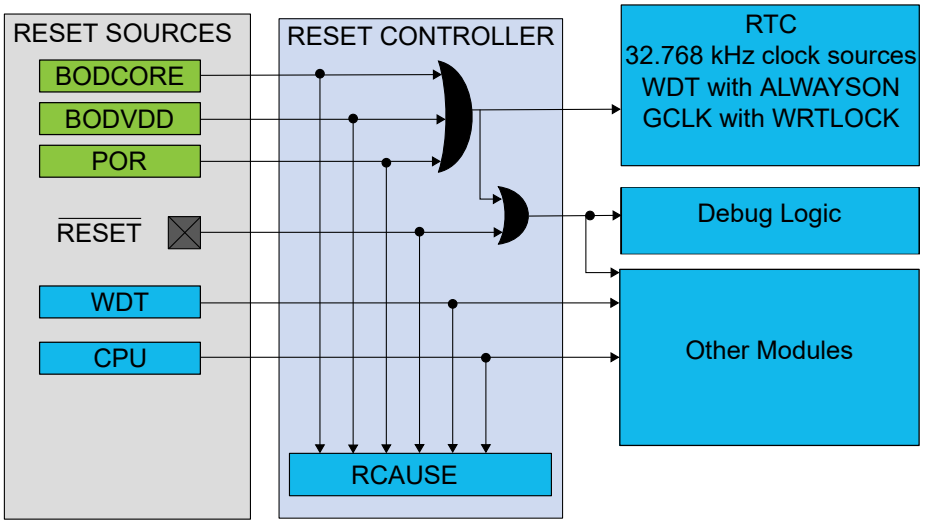
The Reset Controller (RSTC) manages the reset of the microcontroller. It issues a microcontroller reset, sets the device to its initial state and allows the reset source to be identified by software.

## 18.2. Features

- Reset the microcontroller and set it to an initial state according to the reset source
- Reset cause register for reading the reset source from the application code
- Multiple reset sources
  - Power supply reset sources: POR, BODCORE, BODVDD
  - User reset sources: External reset ($\overline{\text{RESET}}$), Watchdog reset, and System Reset Request

## 18.3. Block Diagram

**Figure 18-1.** Reset System




## 18.4. Signal Description

| Signal Name | Type | Description |
|---|---|---|
| $\overline{\text{RESET}}$ | Digital input | External reset pin |

One signal can be mapped on several pins.

**Related Links**

I/O Multiplexing and Considerations

## 18.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 18.5.1. I/O Lines

Not applicable.

### 18.5.2. Power Management

The Reset Controller module is always on.


© 2026 Microchip Technology Inc. and its subsidiaries

### 18.5.3. Clocks

The RSTC bus clock (CLK_RSTC_APB) can be enabled and disabled in the Main Clock Controller.

**Related Links**
MCLK – Main Clock
Peripheral Clock Masking

### 18.5.4. DMA

Not applicable.

### 18.5.5. Interrupts

Not applicable.

### 18.5.6. Events

Not applicable.

### 18.5.7. Debug Operation

When the CPU is halted in debug mode, the RSTC continues normal operation.

### 18.5.8. Register Access Protection

All registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC).

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 18.5.9. Analog Connections

Not applicable.

## 18.6. Functional Description

### 18.6.1. Principle of Operation

The Reset Controller collects the various Reset sources and generates Reset for the device.

### 18.6.2. Basic Operation

#### 18.6.2.1. Initialization

After a power-on Reset, the RSTC is enabled and the Reset Cause (RCAUSE) register indicates the POR source.

#### 18.6.2.2. Enabling, Disabling, and Resetting

The RSTC module is always enabled.

#### 18.6.2.3. Reset Causes and Effects

The latest Reset cause is available in RCAUSE register, and can be read during the application boot sequence in order to determine proper action.

These are the groups of Reset sources:

- Power supply Reset: Resets caused by an electrical issue. It covers POR and BODs Resets
- User Reset: Resets caused by the application. It covers external Resets, system Reset requests and watchdog Resets

The following table lists the parts of the device that are reset, depending on the Reset type.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 18-1.** Effects of the Different Reset Causes

| | Power Supply Reset | User Reset | |
|---|---|---|---|
| | POR, BODVDD, BODCORE | External Reset | WDT Reset, System Reset Request |
| RTC, OSC32KCTRL, RSTC | Reset | - | - |
| GCLK with WRTLOCK | Reset | - | - |
| Debug logic | Reset | Reset | - |
| Others | Reset | Reset | Reset |

The external Reset is generated when pulling the $\overline{\text{RESET}}$ pin low.

The POR, BODCORE, and BODVDD Reset sources are generated by their corresponding module in the Supply Controller Interface (SUPC).

The WDT Reset is generated by the Watchdog Timer.

The System Reset Request is a Reset generated by the CPU when asserting the SYSRESETREQ bit located in the Reset Control register of the CPU (for details refer to the ARM® Cortex™ Technical Reference Manual on http://www.arm.com).

**Note:** Refer to the *External Reset Characteristics* table in the *Timing Characteristics* section of the *Electrical Characteristics* chapter.

**Related Links**
WDT – Watchdog Timer
SUPC – Supply Controller

### 18.6.3. Additional Features
Not applicable.

### 18.6.4. DMA Operation
Not applicable.

### 18.6.5. Interrupts
Not applicable.

### 18.6.6. Events
Not applicable.

### 18.6.7. Sleep Mode Operation
The RSTC module is active in all sleep modes.


© 2026 Microchip Technology Inc. and its subsidiaries

## 18.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | RCAUSE | 7:0 | | SYST | WDT | EXT | | BODVDD | BODCORE | POR |

## 18.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.


© 2026 Microchip Technology Inc. and its subsidiaries

### 18.8.1. Reset Cause

**Name:** RCAUSE
**Offset:** 0x00
**Property:** –

When a Reset occurs, the bit corresponding to the Reset source is set to '1' and all other bits are written to '0'.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | SYST | WDT | EXT | | BODVDD | BODCORE | POR |
| Access | | R | R | R | | R | R | R |
| Reset | | x | x | x | | x | x | x |

**Bit 6 – SYST** System Reset Request
This bit is set if a System Reset Request has occurred. Refer to the Cortex processor documentation for more details.

**Bit 5 – WDT** Watchdog Reset
This bit is set if a Watchdog Timer Reset has occurred.

**Bit 4 – EXT** External Reset
This bit is set if an external Reset has occurred.

**Bit 2 – BODVDD** Brown Out VDD Detector Reset
This bit is set if a BODVDD Reset has occurred.

**Bit 1 – BODCORE** Brown Out CORE Detector Reset
This bit is set if a BODCORE Reset has occurred.

**Bit 0 – POR** Power On Reset
This bit is set if a POR has occurred.


© 2026 Microchip Technology Inc. and its subsidiaries

# 19. PM - Power Manager

**Related Links**

Sleep Mode Operation

## 19.1. Overview

The Power Manager (PM) controls the sleep modes of the device.

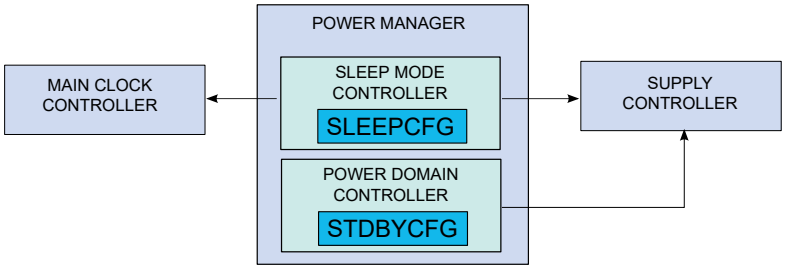
Various sleep modes are provided in order to fit power consumption requirements. This enables the PM to stop unused modules in order to save power. In active mode, the CPU is executing application code. When the device enters a sleep mode, program execution is stopped and some modules and clock domains are automatically switched off by the PM according to the sleep mode. The application code decides which sleep mode to enter and when. Interrupts from enabled peripherals and all enabled reset sources can restore the device from a sleep mode to active mode.

## 19.2. Features

- Power management control
  - Sleep modes: Idle, Standby

## 19.3. Block Diagram

**Figure 19-1.** PM Block Diagram




## 19.4. Signal Description

Not applicable.

## 19.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 19.5.1. I/O Lines

Not applicable.

### 19.5.2. Clocks

The PM bus clock (CLK_PM_APB) can be enabled and disabled in the Main Clock module. If this clock is disabled, it can only be re-enabled by a system reset.

### 19.5.3. DMA

Not applicable.

### 19.5.4. Interrupts

The interrupt request line is connected to the interrupt controller. Using the PM interrupt requires the interrupt controller to be configured first.


© 2026 Microchip Technology Inc. and its subsidiaries

### 19.5.5. Events

Not applicable.

### 19.5.6. Debug Operation

When the CPU is halted in debug mode, the PM continues normal operation. If standby sleep mode is requested by the system while in debug mode, the power domains are not turned off. As a consequence, power measurements while in debug mode are not relevant.

Hot plugging in standby mode is supported.

### 19.5.7. Register Access Protection

Registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC).

Optional PAC Write-Protection is denoted by the "PAC Write-Protection" property in each individual register description.

Write-protection does not apply to accesses through an external debugger.

### 19.5.8. Analog Connections

Not applicable.

## 19.6. Functional Description

### 19.6.1. Terminology

The following is a list of terms used to describe the Power Managemement features of this microcontroller.

#### 19.6.1.1. Sleep Modes

The device can be set in a sleep mode. In sleep mode, the CPU is stopped and the peripherals are either active or idle, according to the sleep mode depth:

- Idle sleep mode: The CPU is stopped. Synchronous clocks are stopped except when requested. The logic is retained.
- Standby sleep mode: The CPU is stopped as well as the peripherals.

### 19.6.2. Principle of Operation

In active mode, all clock domains and power domains are active, allowing software execution and peripheral operation. The PM Sleep Mode Controller allows to save power by choosing between different sleep modes depending on application requirements, see Sleep Mode Controller.

The PM Power Domain Controller allows to reduce the power consumption in standby mode even further.

### 19.6.3. Basic Operation

#### 19.6.3.1. Initialization

After a power-on reset, the PM is enabled, the device is in ACTIVE mode.

#### 19.6.3.2. Enabling, Disabling and Resetting

The PM is always enabled and can not be reset.

#### 19.6.3.3. Sleep Mode Controller

A Sleep mode is entered by executing the Wait For Interrupt instruction (WFI). The Sleep Mode bits in the Sleep Configuration register (SLEEPCFG.SLEEPMODE) select the level of the sleep mode.


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** A small latency happens between the store instruction and actual writing of the SLEEPCFG register due to bridges. Software must ensure that the SLEEPCFG register reads the desired value before issuing a WFI instruction.

**Table 19-1.** Sleep Mode Entry and Exit Table

| Mode | Mode Entry | Wake-Up Sources |
|---|---|---|
| IDLE0 | SLEEPCFG.SLEEPMODE = IDLE0 | Synchronous (2) (CAN included), asynchronous (1) |
| IDLE2 | SLEEPCFG.SLEEPMODE = IDLE2 | Synchronous (2) (CAN excluded), asynchronous (1) |
| STANDBY | SLEEPCFG.SLEEPMODE = STANDBY | Synchronous(3)(CAN excluded), Asynchronous(1) |

**Notes:**

1. Asynchronous: interrupt generated on GCLK generic clock, external clock, or external event.
2. Synchronous: interrupt generated on the APB clock.
3. Synchronous interrupt only for peripherals configured to run in standby.

**Note:** The type of wake-up sources (synchronous or asynchronous) is given in each module interrupt section.

The sleep modes (idle, standby) and their effect on the clocks activity, the regulator and the RAM state are described in the table and the sections below.

**Table 19-2.** Sleep Mode Overview

| Mode | CPU clock | AHB/APB clocks | Main clock | GCLK0 clock | GCLK1-8 clocks | Clock Sources | | Regulator | RAM |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | ONDEMAND = 0 | ONDEMAND = 1 | | |
| **IDLE0** | Stop | Run(1), CAN included | Run | Run | Run/Stop (3) | Run | Run/Stop (4) | Main | Normal |
| **IDLE2** | Stop | Run(2), CAN excluded | | | | | | | |
| **STANDBY** | Stop | Stop (5) | Stop (5) | Stop if RUNSTDBY=0 | | Stop if RUNSTDBY=0 | | LPVREG (8) | Low Power (9) |
| | | | | Stop/Run if RUNSTDBY=1 (6) | | Run if RUNSTDBY=1 | Stop/Run if RUNSTDBY=1 (7) | | |

**Notes:**

1. The AHB/APB clocks are running up to MCLK, and then provided only to the IPs requesting them and to the CAN (which is a specific IP with no clock request mechanism). For the other IPs not requesting their AHB/APB clocks, these are gated at MCLK output.
2. The AHB/APB clocks are running up to MCLK, and then provided only to the IPs requesting them. For the CAN and for the other IPs not requesting the clocks, they are gated at MCLK output.
3. Each GCLK1 to GCLK8 is running if the associated generated clock is requested by at least one IP. It is stopped if no IP is requesting this clock.
4. The clock source is running if the clock is requested by at least one GCLK Generator. It is stopped if no GCLK Generator is requesting this clock and will be restarted as soon as an IP requests a clock coming from a GCLK fed by this clock source.
5. The AHB/APB clocks are stopped, except if requested by at least one IP, and in this case, only provided to this/these IP(s) through GCLK0 and MCLK.
6. Each GCLK generators is stopped, except if the clock it generates is requested by at least one IP.
7. Each Clock Source is stopped, except if the clock it generates is requested by at least one GCLK Generator.


© 2026 Microchip Technology Inc. and its subsidiaries

8. Regulator state is programmable by using STDBYCFG.VREGSMOD bits.
9. RAM state is programmable by using STDBYCFG.BBIASHS bit.

#### 19.6.3.3.1. IDLE Mode

The IDLE mode allows power optimization with the fastest wake-up time.

The CPU is stopped.

The clock source feeding the GCLK generator 0, the GCLK generator 0, and the MCLK are kept active. The AHB/APB clocks are gated at the MCLK output, unless requested by a peripheral.

The other clock sources and the GCLK generators can be running or stopped depending on each clock source ONDEMAND bit, and depending on the peripherals requesting these clocks.

The CAN is a specific peripheral not featuring the AHB/APB clock request mechanism. As a consequence, it is clocked and can wakeup the system in IDLE0 mode, and not clocked and cannot wakeup the system in IDLE2 mode.

If an AHB/APB clock is masked in MCLK.AHBMASK or MCLK.APBxMASK, then it is gated at the output of the MCLK and not provided to the related peripheral (regardless of the related peripheral requesting it or not).

- Entering IDLE mode: The IDLE mode is entered by setting SLEEPCFG.SLEEPMODE = IDLE0/2 and executing the WFI instruction. Additionally, if the SLEEPONEXIT bit in the ARM Cortex System Control register (SCR) is set, the IDLE mode will also be entered when the CPU exits the lowest priority ISR. This mechanism can be useful for applications that only require the processor to run when an interrupt occurs. Before entering the IDLE mode, the user must configure the Sleep Configuration register.
- Exiting IDLE mode: The processor wakes the system up when it detects any non-masked interrupt with sufficient priority to cause exception entry. The system goes back to the ACTIVE mode. The CPU and affected modules are restarted.

In IDLE mode, the regulator and RAM operate in normal mode.

#### 19.6.3.3.2. STANDBY Mode

The STANDBY mode is the lowest power configuration while keeping the state of the logic and the content of the RAM.

This mode depends on (As depicted in the previous table):

- The peripherals running in standby and requesting their asynchronous GCLK clock or their synchronous AHB/APB clock
- The RUNSTDBY bit of the GCLK generators
- The RUNSTDBY/ONDEMAND bit combination of the clock sources

Each clock source and GCLK generator can be:

- Stopped during the whole standby
- Running during the whole standby
- Automatically woken up and switched off depending on the clocks requested by the peripherals during standby (SleepWalking). For example a peripheral can run during standby and request its GCLK asynchronous clock, which will wake up the related GCLK and clock source. Another peripheral may request its APB clock, which will wake up the MCLK, GCLK generator 0 and the related clock source running. (In this case the other AHB/APB clocks are kept gated at the MCLK output).

All features that don't require CPU intervention are supported in STANDBY mode. Here are examples:

- Autonomous peripherals features.


© 2026 Microchip Technology Inc. and its subsidiaries

- Features relying on Event System allowing autonomous communication between peripherals.
- Features relying on on-demand clock.
- DMA transfers.

**Entering STANDBY mode**: This mode is entered by setting SLEEPCFG.SLEEPMODE = STANDBY and by executing the WFI instruction. The SLEEPONEXIT feature is also available as in IDLE mode.

**Exiting STANDBY mode**: Any peripheral able to generate an asynchronous interrupt can wake up the system. For example, a peripheral running on a GCLK clock can trigger an interrupt. When the enabled asynchronous wake-up event occurs and the system is woken up, the device will either execute the interrupt service routine or continue the normal program execution according to the Priority Mask Register (PRIMASK) configuration of the CPU.

Depending on the configuration of these modules, the current consumption of the device in STANDBY mode can be slightly different.

## 19.6.4. Advanced Features

### 19.6.4.1. RAM Automatic Low Power Mode

The RAM is by default put in low power mode (back-biased) if the device is in standby sleep mode.

This behavior can be changed by configuring the Back Bias bit in the Standby Configuration register (STDBYCFG.BBIASHS), refer to the table below for details.

**Note:** In standby sleep mode, the RAM is put in low-power mode by default. This means that the RAM is back-biased, and the DMAC cannot access it. The DMAC can only access the RAM when it is not back biased (PM.STDBYCFG.BBIASxx=0x0).

**Table 19-3.** RAM Back-Biasing Mode

| STBYCDFG.BBIASHS | | RAM |
|---|---|---|
| 0x0 | No Back Biasing | RAM is not back-biased if the device is in standby sleep mode. |
| 0x1 | Standby Back Biasing mode | RAM is back-biased if the device is in standby sleep mode. |

### 19.6.4.2. Regulator Automatic Low-Power Mode

In Standby mode, the PM selects either the main or the low-power voltage regulator to supply the VDDCORE. By default, the low-power voltage regulator is used.

If a sleepwalking task is working on either asynchronous clocks (generic clocks) or synchronous clock (APB/AHB clocks), the main voltage regulator is used. This behavior can be changed by writing the Voltage Regulator Standby Mode bits in the Standby Configuration register (STDBYCFG.VREGSMOD). For additional information, refer to the following table.

**Table 19-4.** Regulator State in Sleep Mode

| Sleep Mode | STDBYCFG. VREGSMOD | SleepWalking (1) | Regulator State for VDDCORE |
|---|---|---|---|
| Active | - | - | main voltage regulator |
| Idle | - | - | main voltage regulator |
| Standby | 0x0: AUTO | NO | low-power regulator |
| | | YES | main voltage regulator |
| | 0x1: PERFORMANCE | - | main voltage regulator |
| | 0x2: LP (2) | (2) | low-power regulator |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. SleepWalking is running on a GCLK clock or synchronous clock. This is not related to the XOSC32K or OSCULP32K clocks.
2. Must only be used when SleepWalking is running on GCLK with 32 kHz source.

### 19.6.5. DMA Operation

Not applicable.

### 19.6.6. Interrupts

Not applicable.

### 19.6.7. Events

Not applicable.

### 19.6.8. Sleep Mode Operation

The Power Manager is always active.


© 2026 Microchip Technology Inc. and its subsidiaries

## 19.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | Reserved | | | | | | | | | |
| 0x01 | SLEEPCFG | 7:0 | | | | | | SLEEPMODE[2:0] | | |
| 0x02<br>...<br>0x07 | Reserved | | | | | | | | | |
| 0x08 | STDBYCFG | 7:0 | VREGSMOD[1:0] | | | | | | | |
| | | 15:8 | | | | | | BBIASHS | | |

## 19.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.


© 2026 Microchip Technology Inc. and its subsidiaries

### 19.8.1. Sleep Configuration

**Name:** SLEEPCFG
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SLEEPMODE[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – SLEEPMODE[2:0]** Sleep Mode
**Note:** A small latency happens between the store instruction and actual writing of the SLEEPCFG register due to bridges. Software has to make sure the SLEEPCFG register reads the wanted value before issuing Wait For Interrupt (WFI) instruction.

| Value | Name |
|---|---|
| 0x0 | IDLE0 |
| 0x1 | Reserved |
| 0x2 | IDLE2 |
| 0x3 | Reserved |
| 0x4 | STANDBY |
| 0x5 - 0x7 | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 19.8.2. Standby Configuration

**Name:** STDBYCFG
**Offset:** 0x08
**Reset:** 0x0400
**Property:** PAC Write-Protection

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BBIASHS | | |
| Access | | | | | | R/W | | |
| Reset | | | | | | 1 | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | VREGSMOD[1:0] | | | | | | | |
| Access | R/W | R/W | | | | | | |
| Reset | 0 | 0 | | | | | | |

**Bit 10 – BBIASHS** Back Bias for HMCRAMCHS
Refer to the RAM Automatic Low Power Mode for details.

| Value | Description |
|---|---|
| 0 | No Back Biasing Mode |
| 1 | Standby Back Biasing Mode |

**Bits 7:6 – VREGSMOD[1:0]** VREG Switching Mode
Refer to the Regulator Automatic Low-Power Mode for details.

| Value | Name | Description |
|---|---|---|
| 0x0 | AUTO | Automatic Mode |
| 0x1 | PERFORMANCE | Performance oriented |
| 0x2 | LP | Low Power consumption oriented |
| 0x3 | Reserved | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

# 20. OSCCTRL – Oscillators Controller

## 20.1. Overview

The Oscillators Controller (OSCCTRL) provides a user interface to the XOSC, OSC48M and FDPLL96M modules. (See OSC32KCTRL for the user interface to the 32.768 kHz oscillators.)

Through the interface registers, users can enable, disable, calibrate, and monitor the OSCCTRL oscillators. All oscillators statuses are collected in the Status (STATUS) register. They can additionally trigger interrupts upon status changes through the INTENSET, INTENCLR, and INTFLAG registers.

**Related Links**

INTENCLR

Interrupt Enable Clear

INTENSET

Interrupt Enable Set

INTFLAG

Interrupt Flag Status and Clear

STATUS

Status

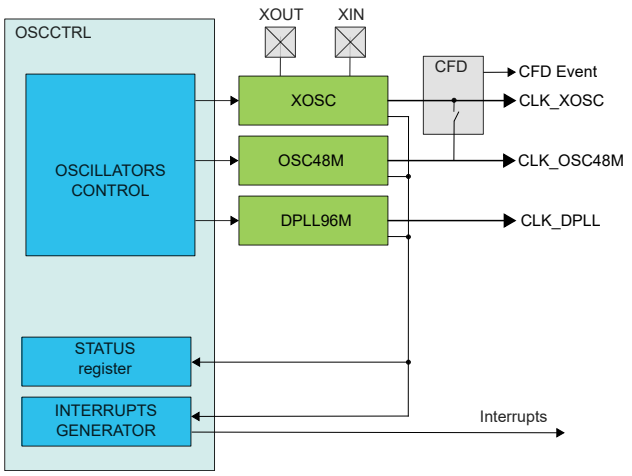
## 20.2. Features

- 0.4-32MHz Crystal Oscillator (XOSC)
  - Tunable gain control
  - Programmable start-up time
  - Crystal or external input clock on XIN I/O
  - Clock failure detection with safe clock switch
  - Clock failure event output
- 48MHz Internal Oscillator (OSC48M)
  - Fast start-up
  - Programmable start-up time
  - 4-bit linear divider available
- Fractional Digital Phase Locked Loop (FDPLL96M)
  - 48MHz to 96MHz output frequency
  - 32kHz to 2MHz reference clock
  - A selection of sources for the reference clock
  - Adjustable proportional integral controller
  - Fractional part used to achieve 1/16th of reference clock step


© 2026 Microchip Technology Inc. and its subsidiaries

## 20.3. Block Diagram

**Figure 20-1.** OSCCTRL Block Diagram




## 20.4. Signal Description

| Signal | Description | Type |
|---|---|---|
| XIN | Multipurpose Crystal Oscillator or external clock generator input | Analog input |
| XOUT | Multipurpose Crystal Oscillator output | Analog output |

The I/O lines are automatically selected when XOSC is enabled.

## 20.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 20.5.1. I/O Lines

I/O lines are configured by OSCCTRL when XOSC is enabled, and need no user configuration.

### 20.5.2. Power Management

The OSCCTRL can continue to operate in any sleep mode where the selected source clock is running. The OSCCTRL interrupts can be used to wake up the device from sleep modes. The events can trigger other operations in the system without exiting sleep modes.

**Related Links**

PM - Power Manager

### 20.5.3. Clocks

The OSCCTRL gathers controls for all device oscillators and provides clock sources to the Generic Clock Controller (GCLK). The available clock sources are: XOSC, OSC48M and FDPLL96M.

The OSCCTRL bus clock (CLK_OSCCTRL_APB) can be enabled and disabled in the Main Clock module (MCLK).

The OSC48M control logic uses the oscillator output, which is also asynchronous to the user interface clock (CLK_OSCCTRL_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**

MCLK – Main Clock

Peripheral Clock Masking


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.5.4. DMA

Not applicable.

### 20.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the OSCCTRL interrupts requires the interrupt controller to be configured first.

**Related Links**

Nested Vector Interrupt Controller

INTFLAG

Interrupt Flag Status and Clear

Sleep Mode Controller

### 20.5.6. Events

The events of this peripheral are connected to the Event System.

**Related Links**

Event System (EVSYS)

### 20.5.7. Debug Operation

When the CPU is halted in debug mode the OSCCTRL continues normal operation. If the OSCCTRL is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 20.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Status and Clear register (INTFLAG)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 20.5.9. Analog Connections

The 0.4-32MHz crystal must be connected between the XIN and XOUT pins, along with any required load capacitors.

## 20.6. Functional Description

### 20.6.1. Principle of Operation

XOSC, OSC48M, and FDPLL96M. are configured via OSCCTRL control registers. Through this interface, these oscillators are enabled, disabled, or have their calibration values updated.

The Status register gathers different status signals coming from the oscillators controlled by the OSCCTRL. These status signals can be used to generate system interrupts, and in some cases wake the system from Sleep mode, provided the corresponding interrupt is enabled.

### 20.6.2. External Multipurpose Crystal Oscillator (XOSC) Operation

The XOSC can operate in the following modes:

- External clock, with an external clock signal connected to the XIN pin
- Crystal oscillator, with an external 0.4-32 MHz crystal connected to the XIN and XOUT pins

The XOSC can be used as a clock source for generic clock generators. This is configured by the Generic Clock Controller (GCLK).


© 2026 Microchip Technology Inc. and its subsidiaries

At reset, the XOSC is disabled, and the XIN/XOUT pins can be used as General Purpose I/O (GPIO) pins or by other peripherals in the system. When XOSC is enabled, the operating mode determines the GPIO usage. When in crystal oscillator mode, the XIN and XOUT pins are controlled by the OSCCTRL, and GPIO functions are overridden on both pins. When in external clock mode, only the XIN pin will be overridden and controlled by the OSCCTRL, while the XOUT pin can still be used as a GPIO pin.

The XOSC is enabled by writing a '1' to the Enable bit in the External Multipurpose Crystal Oscillator Control register (XOSCCTRL.ENABLE).

To enable XOSC as an external crystal oscillator, the XTAL Enable bit (XOSCCTRL.XTALEN) must be written to '1'. If XOSCCTRL.XTALEN is zero, the external clock input on XIN will be enabled.

When in crystal oscillator mode (XOSCCTRL.XTALEN=1), the External Multipurpose Crystal Oscillator Gain (XOSCCTRL.GAIN) must be set to match the external crystal oscillator frequency. If the External Multipurpose Crystal Oscillator Automatic Amplitude Gain Control (XOSCCTRL.AMPGC) is '1', the oscillator amplitude will be automatically adjusted, and in most cases result in a lower power consumption.

The XOSC will behave differently in different sleep modes, based on the settings of XOSCCTRL.RUNSTDBY, XOSCCTRL.ONDEMAND, and XOSCCTRL.ENABLE. If XOSCCTRL.ENABLE=0, the XOSC will be always stopped. For XOSCCTRL.ENABLE=1, this table is valid:

**Table 20-1.** XOSC Sleep Behavior

| CPU Mode | XOSCCTRL.RUNSTDBY | XOSCCTRL.ONDEMAND | Sleep Behavior |
|---|---|---|---|
| Active or Idle | - | 0 | Always run |
| Active or Idle | - | 1 | Run if requested by peripheral |
| Standby | 1 | 0 | Always run |
| Standby | 1 | 1 | Run if requested by peripheral |
| Standby | 0 | - | Run if requested by peripheral |

After a hard reset, or when waking up from a sleep mode where the XOSC was disabled, the XOSC will need time to stabilize on the correct frequency, depending on the external crystal specification. This start-up time can be configured by changing the Oscillator Start-Up Time bit group (XOSCCTRL.STARTUP) in the External Multipurpose Crystal Oscillator Control register. During the start-up time, the oscillator output is masked to ensure that no unstable clock propagates to the digital logic.

The External Multipurpose Crystal Oscillator Ready bit in the Status register (STATUS.XOSCRDY) is set once the external clock or crystal oscillator is stable and ready to be used as a clock source. An interrupt is generated on a zero-to-one transition on STATUS.XOSCRDY if the External Multipurpose Crystal Oscillator Ready bit in the Interrupt Enable Set register (INTENSET.XOSCRDY) is set.

**Related Links**

GCLK - Generic Clock Controller

### 20.6.3. Clock Failure Detection Operation

The Clock Failure Detector (CFD) enables the user to monitor the external clock or crystal oscillator signal provided by the external oscillator (XOSC). The CFD detects failing operation of the XOSC clock with reduced latency, and supports switching to a safe clock source in case of clock failure. The user can also switch from the safe clock back to XOSC in case of recovery. The safe clock is derived from the OSC48M oscillator with a configurable prescaler. This allows configuring the safe clock in order to fulfill the operative conditions of the microcontroller.

In sleep modes, CFD operation is automatically disabled when the external oscillator is not requested to run by a peripheral, for additional information, refer to the “Table 23-1. XOSC Sleep Behavior”.


© 2026 Microchip Technology Inc. and its subsidiaries

The user interface registers allow to enable, disable, and configure the CFD. The Status register provides status flags on failure and clock switch conditions. The CFD can optionally trigger an interrupt or an event when a failure is detected.

### Clock Failure Detection

The CFD is disabled at reset. The CFD does not monitor the XOSC clock when the oscillator is disabled (XOSCCTRL.ENABLE = 0).

Before starting CFD operation, the user must start and enable the safe clock source (OSC48M oscillator).

CFD operation is started by writing a '1' to the CFD Enable bit in the External Oscillator Control register (XOCCTRL.CFDEN). After starting or restarting the XOSC, the CFD does not detect failure until the start-up time has elapsed. The start-up time is configured by the Oscillator Start-Up Time in the External Multipurpose Crystal Oscillator Control register (XOSCCTRL.STARTUP). Once the XOSC Start-Up Time is elapsed, the XOSC clock is constantly monitored.

During a period of 4 safe clocks (monitor period), the CFD watches for a clock activity from the XOSC. There must be at least one rising and one falling XOSC clock edge during 4 safe clock periods to meet non-failure conditions. If no or insufficient activity is detected, the failure status is asserted: The Clock Failure Detector status bit in the Status register (STATUS.CLKFAIL) and the Clock Failure Detector interrupt flag bit in the Interrupt Flag register (INTFLAG.CLKFAIL) are set. If the CLKFAIL bit in the Interrupt Enable Set register (INTENSET.CLKFAIL) is set, an interrupt is generated as well. If the Event Output enable bit in the Event Control register (EVCTRL.CLKFAILEO) is set, an output event is generated, too.

After a clock failure is issued the monitoring of the XOSC clock continues, and the Clock Failure Detector status bit in the Status register (STATUS.CLKFAIL) reflects the current XOSC activity.

### Clock Switch

When a clock failure is detected, the XOSC clock is replaced by the safe clock in order to maintain an active clock during the XOSC clock failure. The safe clock source is the OSC48M oscillator clock. The safe clock source frequency can be scaled down by a configurable prescaler to ensure that the safe clock frequency does not exceed the operating conditions selected by the application. When the XOSC clock is switched to the safe clock, the Clock Switch bit in the Status register (STATUS.CLKSW) is set.

When the CFD has switched to the safe clock, the XOSC is not disabled. If desired, the application must take the necessary actions to disable the oscillator. The application must also take the necessary actions to configure the system clocks to continue normal operations.

If the application can recover the XOSC, the application can switch back to the XOSC clock by writing a '1' to Switch Back Enable bit in the Clock Failure Control register (XOSCCTRL.SWBACK). Once the XOSC clock is switched back, the Switch Back bit (XOSCCTRL.SWBACK) is cleared by hardware.

### Prescaler

The CFD has an internal configurable prescaler to generate the safe clock from the OSC48M oscillator. The prescaler size allows to scale down the OSC48M oscillator so the safe clock frequency is not higher than the XOSC clock frequency monitored by the CFD. The division factor is 2^P, with P being the value of the CFD Prescaler bits in the CFD Prescaler Register (CFDPRESC.CFDPRESC).

**Example 20-1.**

For an external crystal oscillator at 0.4 MHz and the OSC48M frequency at 16 MHz, the CFDPRESC.CFDPRESC value should be set scale down by more than factor 16/0.4=80, for example 128, for a safe clock of adequate frequency.


© 2026 Microchip Technology Inc. and its subsidiaries

### Event

If the Event Output Enable bit in the Event Control register (EVCTRL.CLKFAILEO) is set, the CFD clock failure will be output on the Event Output. When the CFD is switched to the safe clock, the CFD clock failure will not be output on the Event Output.

### Sleep Mode

The CFD is halted depending on configuration of the XOSC and the peripheral clock request. For further details, refer to the Sleep Behavior table above. The CFD interrupt can be used to wake up the device from sleep modes.

## 20.6.4. 48MHz Internal Oscillator (OSC48M) Operation

The OSC48M is an internal oscillator operating in open-loop mode and generating 48MHz frequency. The OSC48M frequency is selected by writing to the Division Factor field in the OSC48MDIV register (OSC48MDIV.DIV). OSC48M is enabled by writing '1' to the Oscillator Enable bit in the OSC48M Control register (OSC48MCTRL.ENABLE), and disabled by writing a '0' to this bit.

After enabling OSC48M, the OSC48M clock is output as soon as the oscillator is ready (STATUS.OSC48MRDY=1). User must ensure that the OSC48M is fully disabled before enabling it by reading STATUS.OSC48MRDY=0.

After reset, OSC48M is enabled and serves as the default clock source at 4MHz.

OSC48M will behave differently in different sleep modes based on the settings of OSC48MCTRL.RUNSTDBY, OSC48MCTRL.ONDEMAND, and OSC48MCTRL.ENABLE. If OSC48MCTRL.ENABLE=0, the OSC48M will be always stopped. For OSC48MCTRL.ENABLE=1, this table is valid:

**Table 20-2.** OSC48M Sleep Behavior

| CPU Mode | OSC48MCTRL.RUNSTDBY | OSC48MCTRL.ONDEMAND | Sleep Behavior |
|---|---|---|---|
| Active or Idle | - | 0 | Always run |
| Active or Idle | - | 1 | Run if requested by peripheral |
| Standby | 1 | 0 | Always run |
| Standby | 1 | 1 | Run if requested by peripheral |
| Standby | 0 | - | Run if requested by peripheral |

After a hard reset, or when waking up from a sleep mode where the OSC48M was disabled, the OSC48M will need time to stabilize on the correct frequency (See Electrical Characteristics). This start-up time can be configured by changing the Oscillator Start-Up Delay bit group (OSC48MSTUP.STARTUP) in the OSC48M Startup register. During the start-up time, the oscillator output is masked to ensure that no unstable clock propagates to the digital logic. The OSC48M Ready bit in the Status register (STATUS.OSC48MRDY) is set when the oscillator is stable and ready to be used as a clock source. An interrupt is generated on a zero-to-one transition on STATUS.OSC48MRDY if the OSC48M Ready bit in the Interrupt Enable Set register (INTENSET.OSC48MRDY) is set.

Faster start-up times are achievable by selecting shorter delays. However, the oscillator frequency may not stabilize within tolerances when short delays are used. If a fast start-up time is desired at the expense of initial accuracy, the division factor should be set to two or higher (OSC48MDIV.DIV > 0).

The OSC48M is used as a clock source for the generic clock generators.

### Related Links

GCLK - Generic Clock Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.6.5. Digital Phase Locked Loop (DPLL) Operation

The task of the DPLL is to maintain coherence between the input (reference) signal and the respective output frequency, CLK_DPLL, via phase comparison. The DPLL controller supports three independent sources of reference clocks:

- XOSC32K: this clock is provided by the 32.768 kHz External Crystal Oscillator (XOSC32K).
- XOSC: this clock is provided by the External Multipurpose Crystal Oscillator (XOSC).
- GCLK: this clock is provided by the Generic Clock Controller.

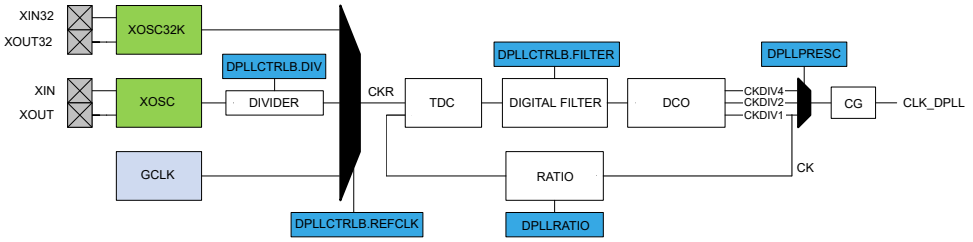
When the controller is enabled, the relationship between the reference clock frequency and the output clock frequency is:

$$f_{CK} = f_{CKR} \times \left(LDR + 1 + \frac{LDRFRAC}{16}\right) \times \frac{1}{2^{PRESC}}$$

Where,

$f_{CK}$ is the frequency of the DPLL output clock, LDR is the loop divider ratio integer part, LDRFRAC is the loop divider ratio fractional part, $f_{CKR}$ is the frequency of the selected reference clock, and PRESC is the output prescaler value.

**Figure 20-2.** DPLL Block Diagram




When the controller is disabled, the output clock is low. If the Loop Divider Ratio Fractional part bit field in the DPLL Ratio register (DPLLRATIO.LDRFRAC) is zero, the DPLL works in integer mode. Otherwise, the fractional mode is activated. Note that the fractional part has a negative impact on the jitter of the DPLL.

> Example (integer mode only): assuming $F_{CKR}$ = 32 kHz and $F_{CK}$ = 48 MHz, the multiplication ratio is 1500. It means that LDR will be set to 1499.

> Example (fractional mode): assuming $F_{CKR}$ = 32 kHz and $F_{CK}$ = 48.006 MHz, the multiplication ratio is 1500.1875 (1500 + 3/16). Thus LDR is set to 1499 and LDRFRAC to 3.

**Related Links**

GCLK - Generic Clock Controller
OSC32KCTRL – 32.768 kHz Oscillators Controller

#### 20.6.5.1. Basic Operation

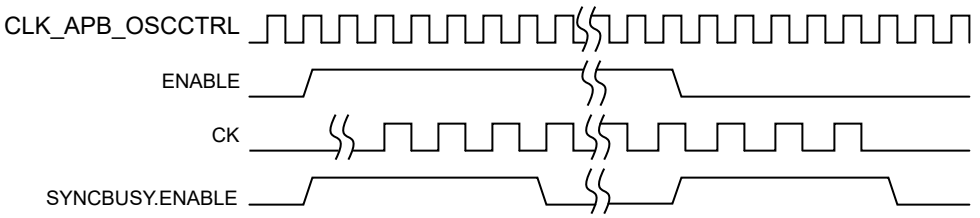
##### 20.6.5.1.1. Initialization, Enabling, Disabling, and Resetting

The DPLLC is enabled by writing a '1' to the Enable bit in the DPLL Control A register (DPLLCTRLA.ENABLE). The DPLLC is disabled by writing a zero to this bit.


© 2026 Microchip Technology Inc. and its subsidiaries

The DPLLSYNCBUSY.ENABLE is set when the DPLLCTRLA.ENABLE bit is modified. It is cleared when the DPLL output clock CK has sampled the bit at the high level after enabling the DPLL. When disabling the DPLL, DPLLSYNCBUSY.ENABLE is cleared when the output clock is no longer running.

**Figure 20-3.** Enable Synchronization Busy Operation




The frequency of the DPLL output clock CK is stable when the module is enabled and when the Lock bit in the DPLL Status register is set (DPLLSTATUS.LOCK).

When the Lock Time bit field in the DPLL Control B register (DPLLCTRLB.LTIME) is non-zero, a user defined lock time is used to validate the lock operation. In this case the lock time is constant. If DPLLCTRLB.LTIME=0, the lock signal is linked with the status bit of the DPLL, and the lock time varies depending on the filter selection and the final target frequency.

When the Wake Up Fast bit (DPLLCTRLB.WUF) is set, the wake up fast mode is activated. In this mode the clock gating cell is enabled at the end of the startup time. At this time the final frequency is not stable, as it is still during the acquisition period, but it allows to save several milliseconds. After first acquisition, the Lock Bypass bit (DPLLCTRLB.LBYPASS) indicates if the lock signal is discarded from the control of the clock gater (CG) generating the output clock CLK_DPLL.

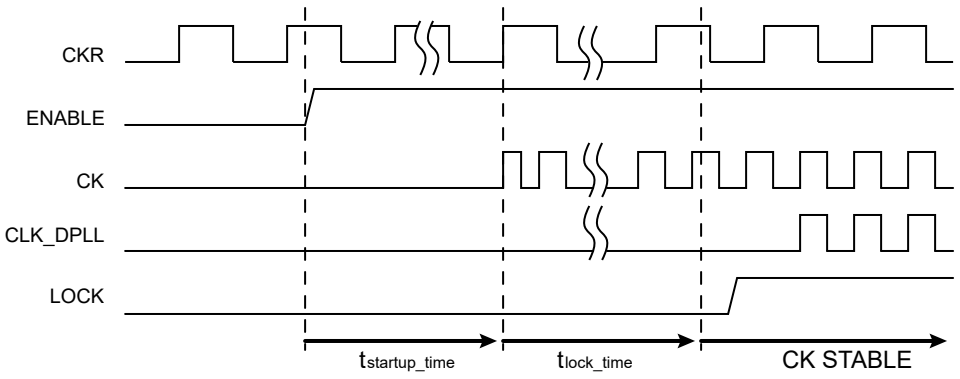
**Table 20-3.** CLK_DPLL Behavior from Startup to First Edge Detection

| WUF | LTIME | CLK_DPLL Behavior |
|---|---|---|
| 0 | 0 | Normal Mode: First Edge when lock is asserted |
| 0 | Not Equal To Zero | Lock Timer Timeout mode: First Edge when the timer down-counts to 0. |
| 1 | X | Wake Up Fast Mode: First Edge when CK is active (startup time) |

**Table 20-4.** CLK_DPLL Behavior after First Edge Detection

| LBYPASS | CLK_DPLL Behavior |
|---|---|
| 0 | Normal Mode: the CLK_DPLL is turned off when lock signal is low. |
| 1 | Lock Bypass Mode: the CLK_DPLL is always running, lock is irrelevant. |

**Figure 20-4.** CK and CLK_DPLL Output from DPLL Off Mode to Running Mode





© 2026 Microchip Technology Inc. and its subsidiaries

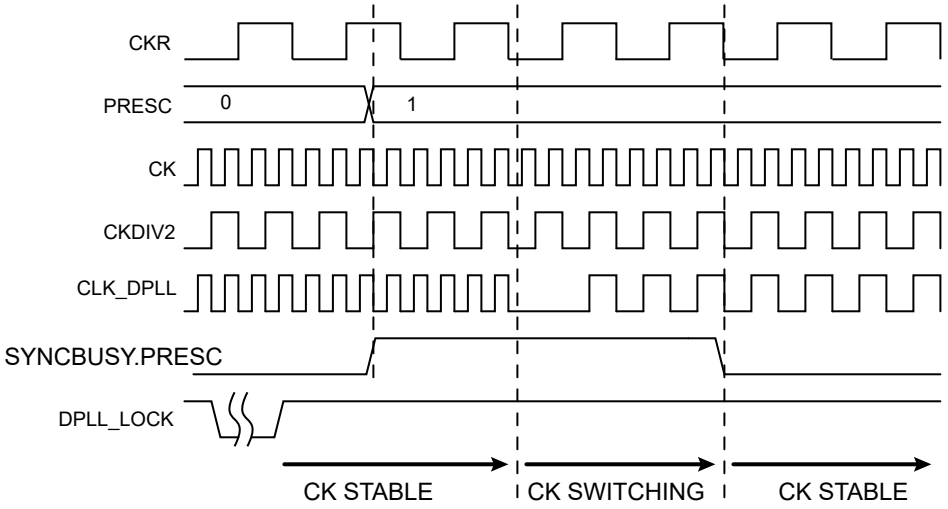
#### 20.6.5.1.2. Reference Clock Switching

When a software operation requires reference clock switching, the recommended procedure is to turn the DPLL into the standby mode, modify the DPLLCTRLB.REFCLK to select the desired reference source, and activate the DPLL again.

#### 20.6.5.1.3. Output Clock Prescaler

The DPLL controller includes an output prescaler. This prescaler provides three selectable output clocks CK, CKDIV2 and CKDIV4. The Prescaler bit field in the DPLL Prescaler register (DPLLPRESC.PRESC) is used to select a new output clock prescaler. When the prescaler field is modified, the DPLLSYNCBUSY.DPLLPRESC bit is set. It will be cleared by hardware when the synchronization is over.

**Figure 20-5.** Output Clock Switching Operation

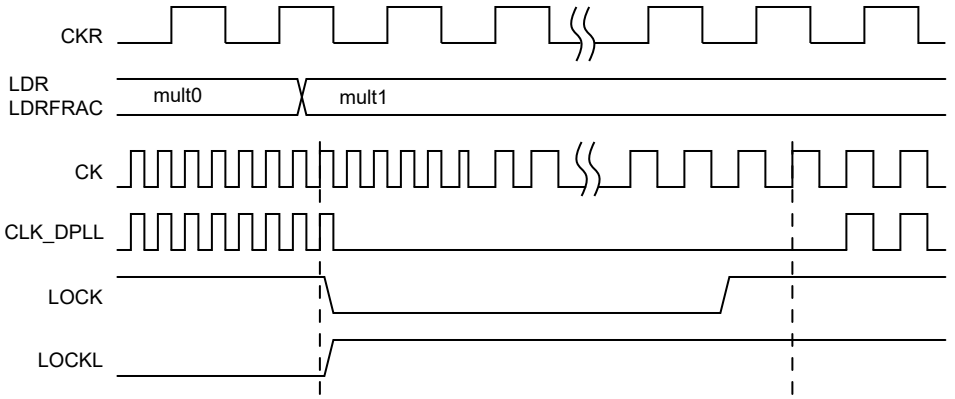



#### 20.6.5.1.4. Loop Divider Ratio Updates

The DPLL Controller supports on-the-fly update of the DPLL Ratio Control (DPLLRATIO) register, allowing to modify the loop divider ratio and the loop divider ratio fractional part when the DPLL is enabled.

STATUS.DPLLLDRTO is set when the DPLLRATIO register has been modified and the DPLL analog cell has successfully sampled the updated value. At that time the DPLLSTATUS.LOCK bit is cleared and set again by hardware when the output frequency reached a stable state.

**Figure 20-6.** RATIOCTRL register update operation





© 2026 Microchip Technology Inc. and its subsidiaries

#### 20.6.5.1.5. Digital Filter Selection

The PLL digital filter (PI controller) is automatically adjusted in order to provide a good compromise between stability and jitter. Nevertheless a software operation can override the filter setting using the Filter bit field in the DPLL Control B register (DPLLCTRLB.FILTER). The Low Power Enable bit (DPLLCTRLB.LPEN) can be use to bypass the Time to Digital Converter (TDC) module.

### 20.6.6. DMA Operation

Not applicable.

### 20.6.7. Interrupts

The OSCCTRL has the following interrupt sources:

- XOSCRDY - Multipurpose Crystal Oscillator Ready: A 0-to-1 transition on the STATUS.XOSCRDY bit is detected
- CLKFAIL - Clock Failure. A 0-to-1 transition on the STATUS.CLKFAIL bit is detected
- OSC48MRDY - 48MHz Internal Oscillator Ready: A 0-to-1 transition on the STATUS.OSC48MRDY bit is detected
- DPLL-related:
  - DPLLLOCKR - DPLL Lock Rise: A 0-to-1 transition of the STATUS.DPLLLOCKR bit is detected
  - DPLLLOCKF - DPLL Lock Fall: A 0-to-1 transition of the STATUS.DPLLLOCKF bit is detected
  - DPLLLTTO - DPLL Lock Timer Time-out: A 0-to-1 transition of the STATUS.DPLLLTTO bit is detected
  - DPLLLDRTO - DPLL Loop Divider Ratio Update Complete. A 0-to-1 transition of the STATUS.DPLLLDRTO bit is detected

All these interrupts are synchronous wake-up sources.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the OSCCTRL is reset. See the INTFLAG register for details on how to clear interrupt flags.

The OSCCTRL has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present. Refer to the INTFLAG register for details.

**Note:** The interrupts must be globally enabled for interrupt requests to be generated.

### 20.6.8. Events

The CFD can generate the following output event:

- Clock Failure (CLKFAIL): Generated when the Clock Failure status bit is set in the Status register (STATUS.CLKFAIL). The CFD event is not generated when the Clock Switch bit (STATUS.CLKSW) in the Status register is set.

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.CFDEO) enables the CFD output event. Writing a '0' to this bit disables the CFD output event. Refer to the *Event System* chapter for details on configuring the event system.


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.6.9. Synchronization

#### OSC48M

Due to the multiple clock domains, values in the OSC48M control registers need to be synchronized to other clock domains.

When executing an operation that requires synchronization, the relevant synchronization bit in the Synchronization Busy register (OSC48MSYNCBUSY) will be set immediately, and cleared when synchronization is complete.

The following registers need synchronization when written:

- OSC48M Divider register (OSC48MDIV)

#### DPLL96M

Due to the multiple clock domains, some registers in the DPLL96M must be synchronized when accessed.

When executing an operation that requires synchronization, the relevant synchronization bit in the Synchronization Busy register (DPLLSYNCBUSY) will be set immediately, and cleared when synchronization is complete.

The following bits need synchronization when written:

- Enable bit in control register A (DPLLCTRLA.ENABLE)
- DPLL Ratio register (DPLLRATIO)
- DPLL Prescaler register (DPLLPRESC)

**Related Links**

Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 20.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | INTENCLR | 7:0 | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| | | 15:8 | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | INTENSET | 7:0 | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| | | 15:8 | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTFLAG | 7:0 | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| | | 15:8 | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | STATUS | 7:0 | | | | OSC48MRDY | | CLKSW | CLKFAIL | XOSCRDY |
| | | 15:8 | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | XOSCCTRL | 7:0 | ONDEMAND | RUNSTDBY | | SWBEN | CFDEN | XTALEN | ENABLE | |
| | | 15:8 | STARTUP[3:0] | | | | AMPGC | GAIN[2:0] | | |
| 0x12 | CFDPRESC | 7:0 | | | | | | CFDPRESC[2:0] | | |
| 0x13 | EVCTRL | 7:0 | | | | | | | | CFDEO |
| 0x14 | OSC48MCTRL | 7:0 | ONDEMAND | RUNSTDBY | | | | | ENABLE | |
| 0x15 | OSC48MDIV | 7:0 | | | | | DIV[3:0] | | | |
| 0x16 | OSC48MSTUP | 7:0 | | | | | | STARTUP[2:0] | | |
| 0x17 | Reserved | | | | | | | | | |
| 0x18 | OSC48MSYNCBUSY | 7:0 | | | | | | OSC48MDIV | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | DPLLCTRLA | 7:0 | ONDEMAND | RUNSTDBY | | | | | ENABLE | |
| 0x1D ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | DPLLRATIO | 7:0 | LDR[7:0] | | | | | | | |
| | | 15:8 | | | | | LDR[11:8] | | | |
| | | 23:16 | | | | | LDRFRAC[3:0] | | | |
| | | 31:24 | | | | | | | | |
| 0x24 | DPLLCTRLB | 7:0 | | | REFCLK[1:0] | | WUF | LPEN | FILTER[1:0] | |
| | | 15:8 | | | | LBYPASS | | LTIME[2:0] | | |
| | | 23:16 | DIV[7:0] | | | | | | | |
| | | 31:24 | | | | | | DIV[10:8] | | |
| 0x28 | DPLLPRESC | 7:0 | | | | | | | PRESC[1:0] | |
| 0x29 ... 0x2B | Reserved | | | | | | | | | |
| 0x2C | DPLLSYNCBUSY | 7:0 | | | | | DPLLPRESC | DPLLRATIO | ENABLE | |
| 0x2D ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DPLLSTATUS | 7:0 | | | | | | | CLKRDY | LOCK |
| 0x31 ... 0x37 | Reserved | | | | | | | | | |
| 0x38 | CAL48M | 7:0 | | | FCAL[5:0] | | | | | |
| | | 15:8 | | | | | | | FRANGE[1:0] | |
| | | 23:16 | | | TCAL[5:0] | | | | | |
| | | 31:24 | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

## 20.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Write-protection is denoted by the "PAC Write-Protection" property in each individual register description. Refer to the Register Access Protection section and the PAC - Peripheral Access Controller chapter for details.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" or "Write-Synchronized" property in each individual register description. Refer to the section on Synchronization for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.1. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| Access | | | | R/W | | | R/W | R/W |
| Reset | | | | 0 | | | 0 | 0 |

**Bit 11 – DPLLLDRTO** DPLL Loop Divider Ratio Update Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the DPLL Loop Divider Ratio Update Complete Interrupt Enable bit, which disables the DPLL Loop Divider Ratio Update Complete interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Loop Divider Ratio Update Complete interrupt is disabled. |
| 1 | The DPLL Loop Divider Ratio Update Complete interrupt is enabled, and an interrupt request will be generated when the DPLL Loop Divider Ratio Update Complete Interrupt flag is set. |

**Bit 10 – DPLLLTO** DPLL Lock Timeout Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the DPLL Lock Timeout Interrupt Enable bit, which disables the DPLL Lock Timeout interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Lock Timeout interrupt is disabled. |
| 1 | The DPLL Lock Timeout interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Timeout Interrupt flag is set. |

**Bit 9 – DPLLLCKF** DPLL Lock Fall Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the DPLL Lock Fall Interrupt Enable bit, which disables the DPLL Lock Fall interrupt.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The DPLL Lock Fall interrupt is disabled. |
| 1 | The DPLL Lock Fall interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Fall Interrupt flag is set. |

**Bit 8 – DPLLLCKR** DPLL Lock Rise Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the DPLL Lock Rise Interrupt Enable bit, which disables the DPLL Lock Rise interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Lock Rise interrupt is disabled. |
| 1 | The DPLL Lock Rise interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Rise Interrupt flag is set. |

**Bit 4 – OSC48MRDY** OSC48M Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the OSC48M Ready Interrupt Enable bit, which disables the OSC48M Ready interrupt.

| Value | Description |
|---|---|
| 0 | The OSC48M Ready interrupt is disabled. |
| 1 | The OSC48M Ready interrupt is enabled, and an interrupt request will be generated when the OSC48M Ready Interrupt flag is set. |

**Bit 1 – CLKFAIL** Clock Failure Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the XOSC Clock Failure Interrupt Enable bit, which disables the XOSC Clock Failure interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC Clock Failure interrupt is disabled. |
| 1 | The XOSC Clock Failure interrupt is enabled, and an interrupt request will be generated when the XOSC Clock Failure Interrupt flag is set. |

**Bit 0 – XOSCRDY** XOSC Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the XOSC Ready Interrupt Enable bit, which disables the XOSC Ready interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC Ready interrupt is disabled. |
| 1 | The XOSC Ready interrupt is enabled, and an interrupt request will be generated when the XOSC Ready Interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.2. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| Access | | | | R/W | | | R/W | R/W |
| Reset | | | | 0 | | | 0 | 0 |

**Bit 11 – DPLLLDRTO** DPLL Loop Divider Ratio Update Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the DPLL Loop Ratio Update Complete Interrupt Enable bit, which enables the DPLL Loop Ratio Update Complete interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Loop Divider Ratio Update Complete interrupt is disabled. |
| 1 | The DPLL Loop Ratio Update Complete interrupt is enabled, and an interrupt request will be generated when the DPLL Loop Ratio Update Complete Interrupt flag is set. |

**Bit 10 – DPLLLTO** DPLL Lock Timeout Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the DPLL Lock Timeout Interrupt Enable bit, which enables the DPLL Lock Timeout interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Lock Timeout interrupt is disabled. |
| 1 | The DPLL Lock Timeout interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Timeout Interrupt flag is set. |

**Bit 9 – DPLLLCKF** DPLL Lock Fall Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the DPLL Lock Fall Interrupt Enable bit, which enables the DPLL Lock Fall interrupt.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The DPLL Lock Fall interrupt is disabled. |
| 1 | The DPLL Lock Fall interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Fall Interrupt flag is set. |

**Bit 8 – DPLLLCKR** DPLL Lock Rise Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the DPLL Lock Rise Interrupt Enable bit, which enables the DPLL Lock Rise interrupt.

| Value | Description |
|---|---|
| 0 | The DPLL Lock Rise interrupt is disabled. |
| 1 | The DPLL Lock Rise interrupt is enabled, and an interrupt request will be generated when the DPLL Lock Rise Interrupt flag is set. |

**Bit 4 – OSC48MRDY** OSC48M Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the OSC48M Ready Interrupt Enable bit, which enables the OSC48M Ready interrupt.

| Value | Description |
|---|---|
| 0 | The OSC48M Ready interrupt is disabled. |
| 1 | The OSC48M Ready interrupt is enabled, and an interrupt request will be generated when the OSC48M Ready Interrupt flag is set. |

**Bit 1 – CLKFAIL** XOSC Clock Failure Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the XOSC Clock Failure Interrupt Enable bit, which enables the XOSC Clock Failure Interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC Clock Failure Interrupt is disabled. |
| 1 | The XOSC Clock Failure Interrupt is enabled, and an interrupt request will be generated when the XOSC Clock Failure Interrupt flag is set. |

**Bit 0 – XOSCRDY** XOSC Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the XOSC Ready Interrupt Enable bit, which enables the XOSC Ready interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC Ready interrupt is disabled. |
| 1 | The XOSC Ready interrupt is enabled, and an interrupt request will be generated when the XOSC Ready Interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.3. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | OSC48MRDY | | | CLKFAIL | XOSCRDY |
| Access | | | | R/W | | | R/W | R/W |
| Reset | | | | 0 | | | 0 | 0 |

**Bit 11 – DPLLLDRTO** DPLL Loop Divider Ratio Update Complete
This flag is cleared by writing '1' to it.
This flag is set on 0-to-1 transition of the DPLL Loop Divider Ratio Update Complete bit in the Status register (STATUS.DPLLLDRTO) and will generate an interrupt request if INTENSET.DPLLLDRTO is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the DPLL Loop Divider Ratio Update Complete interrupt flag.

**Bit 10 – DPLLLTO** DPLL Lock Timeout
This flag is cleared by writing '1' to it.
This flag is set on 0-to-1 transition of the DPLL Lock Timeout bit in the Status register (STATUS.DPLLLTO) and will generate an interrupt request if INTENSET.DPLLLTO is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the DPLL Lock Timeout interrupt flag.

**Bit 9 – DPLLLCKF** DPLL Lock Fall
This flag is cleared by writing '1' to it.
This flag is set on 0-to-1 transition of the DPLL Lock Fall bit in the Status register (STATUS.DPLLLCKF) and will generate an interrupt request if INTENSET.DPLLLCKF is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the DPLL Lock Fall interrupt flag.

**Bit 8 – DPLLLCKR** DPLL Lock Rise
This flag is cleared by writing '1' to it.
This flag is set on 0-to-1 transition of the DPLL Lock Rise bit in the Status register (STATUS.DPLLLCKR) and will generate an interrupt request if INTENSET.DPLLLCKR is '1'.
Writing '0' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing '1' to this bit clears the DPLL Lock Rise interrupt flag.

**Bit 4 – OSC48MRDY** OSC48M Ready
This flag is cleared by writing '1' to it.
This flag is set on 0-to-1 transition of the OSC48M Ready bit in the Status register (STATUS.OSC48MRDY) and will generate an interrupt request if INTENSET.OSC48MRDY is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the OSC48M Ready interrupt flag.

**Bit 1 – CLKFAIL** XOSC Failure Detection
This flag is cleared by writing '1' to it.
This flag is set on a 0-to-1 transition of the XOSC Clock Failure bit in the Status register (STATUS.CLKFAIL) and will generate an interrupt request if INTENSET.CLKFAIL is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the XOSC Clock Fail interrupt flag.

**Bit 0 – XOSCRDY** XOSC Ready
This flag is cleared by writing '1' to it.
This flag is set on a 0-to-1 transition of the XOSC Ready bit in the Status register (STATUS.XOSCRDY) and will generate an interrupt request if INTENSET.XOSCRDY is '1'.
Writing '0' to this bit has no effect.
Writing '1' to this bit clears the XOSC Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.4. Status

**Name:** STATUS
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DPLLLDRTO | DPLLLTO | DPLLLCKF | DPLLLCKR |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | OSC48MRDY | | CLKSW | CLKFAIL | XOSCRDY |
| Access | | | | R | | R | R | R |
| Reset | | | | 0 | | 0 | 0 | 0 |

**Bit 11 – DPLLLDRTO** DPLL Loop Divider Ratio Update Complete

| Value | Description |
|---|---|
| 0 | DPLL Loop Divider Ratio Update Complete not detected. |
| 1 | DPLL Loop Divider Ratio Update Complete detected. |

**Bit 10 – DPLLLTO** DPLL Lock Timeout

| Value | Description |
|---|---|
| 0 | DPLL Lock time-out not detected. |
| 1 | DPLL Lock time-out detected. |

**Bit 9 – DPLLLCKF** DPLL Lock Fall

| Value | Description |
|---|---|
| 0 | DPLL Lock fall edge not detected. |
| 1 | DPLL Lock fall edge detected. |

**Bit 8 – DPLLLCKR** DPLL Lock Rise

| Value | Description |
|---|---|
| 0 | DPLL Lock rise edge not detected. |
| 1 | DPLL Lock fall edge detected. |

**Bit 4 – OSC48MRDY** OSC48M Ready

| Value | Description |
|---|---|
| 0 | OSC48M is not ready. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | OSC48M is stable and ready to be used as a clock source. |

**Bit 2 – CLKSW** XOSC Clock Switch

| Value | Description |
|---|---|
| 0 | XOSC is not switched and provides the external clock or crystal oscillator clock. |
| 1 | XOSC is switched and provides the safe clock. |

**Bit 1 – CLKFAIL** XOSC Clock Failure

**Note:** Once a first failure has been detected, it is logged in INTFLAG.CLKFAIL and will trigger an interrupt if enabled. After the detection of the first failure, the value of the STATUS.CLKFAIL bit becomes irrelevant and should be ignored until the XOSC clock is restored.

| Value | Description |
|---|---|
| 0 | No XOSC failure detected. |
| 1 | A XOSC failure was detected. |

**Bit 0 – XOSCRDY** XOSC Ready

| Value | Description |
|---|---|
| 0 | XOSC is not ready. |
| 1 | XOSC is stable and ready to be used as a clock source. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.5. External Multipurpose Crystal Oscillator (XOSC) Control

**Name:** XOSCCTRL
**Offset:** 0x10
**Reset:** 0x0080
**Property:** PAC Write-Protection

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | STARTUP[3:0] | | | | AMPGC | GAIN[2:0] | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | SWBEN | CFDEN | XTALEN | ENABLE | |
| Access | R/W | R/W | | R/W | R/W | R/W | R/W | |
| Reset | 1 | 0 | | 0 | 0 | 0 | 0 | |

**Bits 15:12 – STARTUP[3:0]** Start-Up Time
These bits select start-up time for the oscillator.
The OSCULP32K oscillator is used to clock the start-up counter.

**Table 20-5.** Start-Up Time for External Multipurpose Crystal Oscillator

| STARTUP[3:0] | Number of OSCULP32K Clock Cycles | Number of XOSC Clock Cycles | Approximate Equivalent Time [µs] |
|---|---|---|---|
| 0x0 | 1 | 3 | 31 |
| 0x1 | 2 | 3 | 61 |
| 0x2 | 4 | 3 | 122 |
| 0x3 | 8 | 3 | 244 |
| 0x4 | 16 | 3 | 488 |
| 0x5 | 32 | 3 | 977 |
| 0x6 | 64 | 3 | 1953 |
| 0x7 | 128 | 3 | 3906 |
| 0x8 | 256 | 3 | 7813 |
| 0x9 | 512 | 3 | 15625 |
| 0xA | 1024 | 3 | 31250 |
| 0xB | 2048 | 3 | 62500 µs |
| 0xC | 4096 | 3 | 125000 |
| 0xD | 8192 | 3 | 250000 |
| 0xE | 16384 | 3 | 500000 |
| 0xF | 32768 | 3 | 1000000 |

**Notes:**
1. Actual startup time is 1 OSCULP32K cycle + 3 XOSC cycles.
2. The given time neglects the three XOSC cycles before the OSCULP32K cycle.

**Bit 11 – AMPGC** Automatic Amplitude Gain Control
**Note:** The configuration of the oscillator gain is mandatory even if the AMPGC feature is enabled at start up.

| Value | Description |
|---|---|
| 0 | The automatic amplitude gain control is disabled. |
| 1 | The automatic amplitude gain control is enabled. Amplitude gain will be automatically adjusted during Crystal Oscillator operation. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 10:8 – GAIN[2:0]** Oscillator Gain

These bits select the gain for the oscillator. The listed maximum frequencies are recommendations, and might vary based on capacitive load and crystal characteristics. Those bits must be configured even when the Automatic Amplitude Gain Control is active.

| Value | Recommended Max Frequency [MHz] |
|---|---|
| 0x0 | 2 |
| 0x1 | 4 |
| 0x2 | 8 |
| 0x3 | 16 |
| 0x4 | 32 |
| 0x5-0x7 | Reserved |

**Bit 7 – ONDEMAND** On Demand Control

The On Demand operation mode allows the oscillator to be enabled or disabled, depending on peripheral clock requests.
If the ONDEMAND bit has been previously written to '1', the oscillator will be running only when requested by a peripheral. If there is no peripheral requesting the oscillator's clock source, the oscillator will be in a disabled state.
If On Demand is disabled, the oscillator will always be running when enabled.
In standby sleep mode, the On Demand operation is still active.

| Value | Description |
|---|---|
| 0 | The oscillator is always on, if enabled. |
| 1 | The oscillator is enabled when a peripheral is requesting the oscillator to be used as a clock source. The oscillator is disabled if no peripheral is requesting the clock source. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls how the XOSC behaves during Standby Sleep mode, together with the ONDEMAND bit:

| Value | Description |
|---|---|
| 0 | The XOSC is not running in Standby Sleep mode if no peripheral requests the clock. |
| 1 | The XOSC is running in Standby Sleep mode. If ONDEMAND = 1, the XOSC will be running when a peripheral is requesting the clock. If ONDEMAND = 0, the clock source will always be running in Standby Sleep mode. |

**Bit 4 – SWBEN** Clock Switch Back

This bit controls the XOSC output switch back to the external clock or crystal oscillator in case of clock recovery:

| Value | Description |
|---|---|
| 0 | The clock switch back is disabled. |
| 1 | The clock switch back is enabled. This bit is reset once the XOSC putput clock is switched back to the external clock or crystal oscillator. |

**Bit 3 – CFDEN** Clock Failure Detector Enable

This bit controls the clock failure detector:

| Value | Description |
|---|---|
| 0 | The Clock Failure Detector is disabled. |
| 1 | the Clock Failure Detector is enabled. |

**Bit 2 – XTALEN** Crystal Oscillator Enable

This bit controls the connections between the I/O pads and the external clock or crystal oscillator:

| Value | Description |
|---|---|
| 0 | External clock connected on XIN. XOUT can be used as general-purpose I/O. |
| 1 | Crystal connected to XIN/XOUT. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – ENABLE** Oscillator Enable

| Value | Description |
|---|---|
| 0 | The oscillator is disabled. |
| 1 | The oscillator is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.6. Clock Failure Detector Prescaler

**Name:** CFDPRESC
**Offset:** 0x12
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CFDPRESC[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – CFDPRESC[2:0]** Clock Failure Detector Prescaler
These bits select the prescaler for the clock failure detector.
The OSC48M oscillator is used to clock the CFD prescaler. The CFD safe clock frequency is the OSC48M frequency divided by 2^CFDPRESC.


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.7. Event Control

**Name:** EVCTRL
**Offset:** 0x13
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CFDEO |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – CFDEO** Clock Failure Detector Event Output Enable
This bit indicates whether the Clock Failure detector event output is enabled or not and an output event will be generated when the Clock Failure detector detects a clock failure

| Value | Description |
|---|---|
| 0 | Clock Failure detector event output is disabled and no event will be generated. |
| 1 | Clock Failure detector event output is enabled and an event will be generated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.8. 48MHz Internal Oscillator (OSC48M) Control

**Name:** OSC48MCTRL
**Offset:** 0x14
**Reset:** 0x82
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | | | ENABLE | |
| Access | R/W | R/W | | | | | R/W | |
| Reset | 1 | 0 | | | | | 1 | |

**Bit 7 – ONDEMAND** On Demand Control

The On Demand operation mode allows the oscillator to be enabled or disabled depending on peripheral clock requests.
If the ONDEMAND bit has been previously written to '1', the oscillator will only be running when requested by a peripheral. If there is no peripheral requesting the oscillator's clock source, the oscillator will be in a disabled state.
If On Demand is disabled the oscillator will always be running when enabled.
In standby sleep mode, the On Demand operation is still active.

| Value | Description |
|---|---|
| 0 | The oscillator is always on, if enabled. |
| 1 | The oscillator is enabled when a peripheral is requesting the oscillator to be used as a clock source. The oscillator is disabled if no peripheral is requesting the clock source. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls how the OSC48M behaves during standby sleep mode.

| Value | Description |
|---|---|
| 0 | The OSC48M is disabled in standby sleep mode if no peripheral requests the clock. |
| 1 | The OSC48M is not stopped in standby sleep mode. If ONDEMAND=1, the OSC48M will be running when a peripheral is requesting the clock. If ONDEMAND=0, the clock source will always be running in standby sleep mode. |

**Bit 1 – ENABLE** Oscillator Enable

| Value | Description |
|---|---|
| 0 | The oscillator is disabled. |
| 1 | The oscillator is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.9. OSC48M Divider

**Name:** OSC48MDIV
**Offset:** 0x15
**Reset:** 0x0B
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DIV[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 1 | 0 | 1 | 1 |

**Bits 3:0 – DIV[3:0]** Oscillator Divider Selection
These bits control the oscillator frequency range by adjusting the division ratio. The oscillator frequency is 48MHz divided by DIV+1.

| Value | Description |
|---|---|
| 0000 | 48MHz |
| 0001 | 24MHz |
| 0010 | 16MHz |
| 0011 | 12MHz |
| 0100 | 9.6MHz |
| 0101 | 8MHz |
| 0110 | 6.86MHz |
| 0111 | 6MHz |
| 1000 | 5.33MHz |
| 1001 | 4.8MHz |
| 1010 | 4.36MHz |
| 1011 | 4MHz |
| 1100 | 3.69MHz |
| 1101 | 3.43MHz |
| 1110 | 3.2MHz |
| 1111 | 3MHz |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.10. OSC48M Startup

**Name:** OSC48MSTUP
**Offset:** 0x16
**Reset:** 0x07
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | STARTUP[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 1 | 1 | 1 |

**Bits 2:0 – STARTUP[2:0]** Oscillator Startup Delay
These bits select the oscillator start-up delay in oscillator cycles.

**Table 20-6.** Oscillator Divider Selection

| STARTUP[2:0] | Number of OSCM48M Clock Cycles | Approximate Equivalent Time |
|---|---|---|
| 0x0 | 8 | 166ns |
| 0x1 | 16 | 333ns |
| 0x2 | 32 | 667ns |
| 0x3 | 64 | 1.333µs |
| 0x4 | 128 | 2.667µs |
| 0x5 | 256 | 5.333µs |
| 0x6 | 512 | 10.667µs |
| 0x7 | 1024 | 21.333µs |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.11. OSC48M Synchronization Busy

**Name:** OSC48MSYNCBUSY
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | OSC48MDIV | | |
| Access | | | | | | R/W | | |
| Reset | | | | | | 1 | | |

**Bit 2 – OSC48MDIV** Oscillator Divider Synchronization Status
This bit is set when OSC48MDIV register is written.
This bit is cleared when OSC48MDIV synchronization is completed.

| Value | Description |
|---|---|
| 0 | No synchronized access. |
| 1 | Synchronized access is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.12. DPLL Control A

**Name:** DPLLCTRLA
**Offset:** 0x1C
**Reset:** 0x80
**Property:** PAC Write-Protection, Write-Synchronized (ENABLE)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | | | ENABLE | |
| Access | R/W | R/W | | | | | R/W | |
| Reset | 1 | 0 | | | | | 0 | |

**Bit 7 – ONDEMAND** On Demand Clock Activation

The On Demand operation mode allows the DPLL to be enabled or disabled depending on peripheral clock requests.
If the ONDEMAND bit has been previously written to '1', the DPLL will only be running when requested by a peripheral. If there is no peripheral requesting the DPLL's clock source, the DPLL will be in a disabled state.
If On Demand is disabled the DPLL will always be running when enabled.
In standby sleep mode, the On Demand operation is still active.

| Value | Description |
|---|---|
| 0 | The DPLL is always on, if enabled. |
| 1 | The DPLL is enabled when a peripheral is requesting the DPLL to be used as a clock source. The DPLL is disabled if no peripheral is requesting the clock source. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls how the DPLL behaves during standby sleep mode:

| Value | Description |
|---|---|
| 0 | The DPLL is disabled in standby sleep mode if no peripheral requests the clock. |
| 1 | The DPLL is not stopped in standby sleep mode. If ONDEMAND=1, the DPLL will be running when a peripheral is requesting the clock. If ONDEMAND=0, the clock source will always be running in standby sleep mode. |

**Bit 1 – ENABLE** DPLL Enable

The software operation of enabling or disabling the DPLL takes a few clock cycles, so the DPLLSYNCBUSY.ENABLE status bit indicates when the DPLL is successfully enabled or disabled.

| Value | Description |
|---|---|
| 0 | The DPLL is disabled. |
| 1 | The DPLL is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.13. DPLL Ratio Control

**Name:** DPLLRATIO
**Offset:** 0x20
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LDRFRAC[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LDR[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | LDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 19:16 – LDRFRAC[3:0]** Loop Divider Ratio Fractional Part

Writing these bits selects the fractional part of the frequency multiplier. Due to synchronization there is a delay between writing these bits and the effect on the DPLL output clock. The value written will read back immediately and the DPLLRATIO bit in the DPLL Synchronization Busy register (DPLLSYNCBUSY.DPLLRATIO) will be set. DPLLSYNCBUSY.DPLLRATIO will be cleared when the operation is completed.

**Bits 11:0 – LDR[11:0]** Loop Divider Ratio

Writing these bits selects the integer part of the frequency multiplier. The value written to these bits will read back immediately, and the DPLLRATIO bit in the DPLL Synchronization busy register (DPLLSYNCBUSY.DPLLRATIO), will be set. DPLLSYNCBUSY.DPLLRATIO will be cleared when the operation is completed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.14. DPLL Control B

**Name:** DPLLCTRLB
**Offset:** 0x24
**Reset:** 0x00
**Property:** Enable-Protected, PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | DIV[10:8] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIV[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | LBYPASS | | LTIME[2:0] | | |
| Access | | | | R/W | | R/W | R/W | R/W |
| Reset | | | | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | REFCLK[1:0] | | WUF | LPEN | FILTER[1:0] | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 26:16 – DIV[10:0]** Clock Divider
These bits set the XOSC clock division factor and can be calculated with following formula:

$$f_{DIV} = \frac{f_{XOSC}}{2x(DIV + 1)}$$

**Bit 12 – LBYPASS** Lock Bypass

| Value | Description |
|---|---|
| 0 | DPLL Lock signal drives the DPLL controller internal logic. |
| 1 | DPLL Lock signal is always asserted. |

**Bits 10:8 – LTIME[2:0]** Lock Time
These bits select the lock time-out value:

| Value | Name | Description |
|---|---|---|
| 0x0 | Default | No time-out. Automatic lock. |
| 0x1 | Reserved | |
| 0x2 | Reserved | |
| 0x3 | Reserved | |
| 0x4 | 8MS | Time-out if no lock within 8ms |
| 0x5 | 9MS | Time-out if no lock within 9ms |
| 0x6 | 10MS | Time-out if no lock within 10ms |
| 0x7 | 11MS | Time-out if no lock within 11ms |

**Bits 5:4 – REFCLK[1:0]** Reference Clock Selection
Write these bits to select the DPLL clock reference:


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | XOSC32K | XOSC32K clock reference |
| 0x1 | XOSC | XOSC clock reference |
| 0x2 | GCLK | GCLK clock reference |
| 0x3 | Reserved | |

**Bit 3 – WUF** Wake Up Fast

| Value | Description |
|---|---|
| 0 | DPLL clock is output after startup and lock time. |
| 1 | DPLL clock is output after startup time. |

**Bit 2 – LPEN** Low-Power Enable

| Value | Description |
|---|---|
| 0 | The low-power mode is disabled. Time to Digital Converter is enabled. |
| 1 | The low-power mode is enabled. Time to Digital Converter is disabled. This will improve power consumption but increase the output jitter. |

**Bits 1:0 – FILTER[1:0]** Proportional Integral Filter Selection
These bits select the DPLL filter type:

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default filter mode |
| 0x1 | LBFILT | Low bandwidth filter |
| 0x2 | HBFILT | High bandwidth filter |
| 0x3 | HDFILT | High damping filter |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.15. DPLL Prescaler

**Name:** DPLLPRESC
**Offset:** 0x28
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | PRESC[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 1:0 – PRESC[1:0]** Output Clock Prescaler
These bits define the output clock prescaler setting.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV1 | DPLL output is divided by 1 |
| 0x1 | DIV2 | DPLL output is divided by 2 |
| 0x2 | DIV4 | DPLL output is divided by 4 |
| 0x3 | Reserved | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.16. DPLL Synchronization Busy

**Name:** DPLLSYNCBUSY
**Offset:** 0x2C
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DPLLPRESC | DPLLRATIO | ENABLE | |
| Access | | | | | R | R | R | |
| Reset | | | | | 0 | 0 | 0 | |

**Bit 3 – DPLLPRESC** DPLL Prescaler Synchronization Status

| Value | Description |
|---|---|
| 0 | The DPLLRESC register has been synchronized. |
| 1 | The DPLLRESC register value has changed and its synchronization is in progress. |

**Bit 2 – DPLLRATIO** DPLL Loop Divider Ratio Synchronization Status

| Value | Description |
|---|---|
| 0 | The DPLLRATIO register has been synchronized. |
| 1 | The DPLLRATIO register value has changed and its synchronization is in progress. |

**Bit 1 – ENABLE** DPLL Enable Synchronization Status

| Value | Description |
|---|---|
| 0 | The DPLLCTRLA.ENABLE bit has been synchronized. |
| 1 | The DPLLCTRLA.ENABLE bit value has changed and its synchronization is in progress. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.17. DPLL Status

**Name:** DPLLSTATUS
**Offset:** 0x30
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | CLKRDY | LOCK |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – CLKRDY** Output Clock Ready

| Value | Description |
|---|---|
| 0 | The DPLL output clock is off. |
| 1 | The DPLL output clock in on. |

**Bit 0 – LOCK** DPLL Lock status bit

| Value | Description |
|---|---|
| 0 | The DPLL Lock signal is cleared, when the DPLL is disabled or when the DPLL is trying to reach the target frequency. |
| 1 | The DPLL Lock signal is asserted when the desired frequency is reached. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 20.8.18. OSC48M Calibration

**Name:** CAL48M
**Offset:** 0x38
**Reset:** Calibrated value for VDD range 3.6 V to 5.5 V
**Property:** PAC Write-Protection

This register (bits 0 to 21) must be updated with the corresponding data in the NVM Software Calibration Area: CAL48M 5V or CAL48M 3V3, depending on the VDD range. Refer to NVM Software Calibration Area Mapping.

**Note:** This register is only available for Rev D silicon and above.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | TCAL[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | x | x | x | x | x | x |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | FRANGE[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | x | x |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | FCAL[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | x | x | x | x | x | x |

**Bits 21:16 – TCAL[5:0]** Temperature Calibration

**Bits 9:8 – FRANGE[1:0]** Frequency Range

**Bits 5:0 – FCAL[5:0]** Frequency Calibration

**Related Links**
NVM Software Calibration Area Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

# 21. OSC32KCTRL – 32.768 kHz Oscillators Controller

## 21.1. Overview

The 32.768 kHz Oscillators Controller (OSC32KCTRL) provides a user interface to the 32.768 kHz oscillators: XOSC32K, OSC32K, and OSCULP32K.

The OSC32KCTRL sub-peripherals can be enabled, disabled, calibrated, and monitored through interface registers.

All sub-peripheral statuses are collected in the Status register (STATUS). They can additionally trigger interrupts upon status changes via the INTENSET, INTENCLR, and INTFLAG registers.

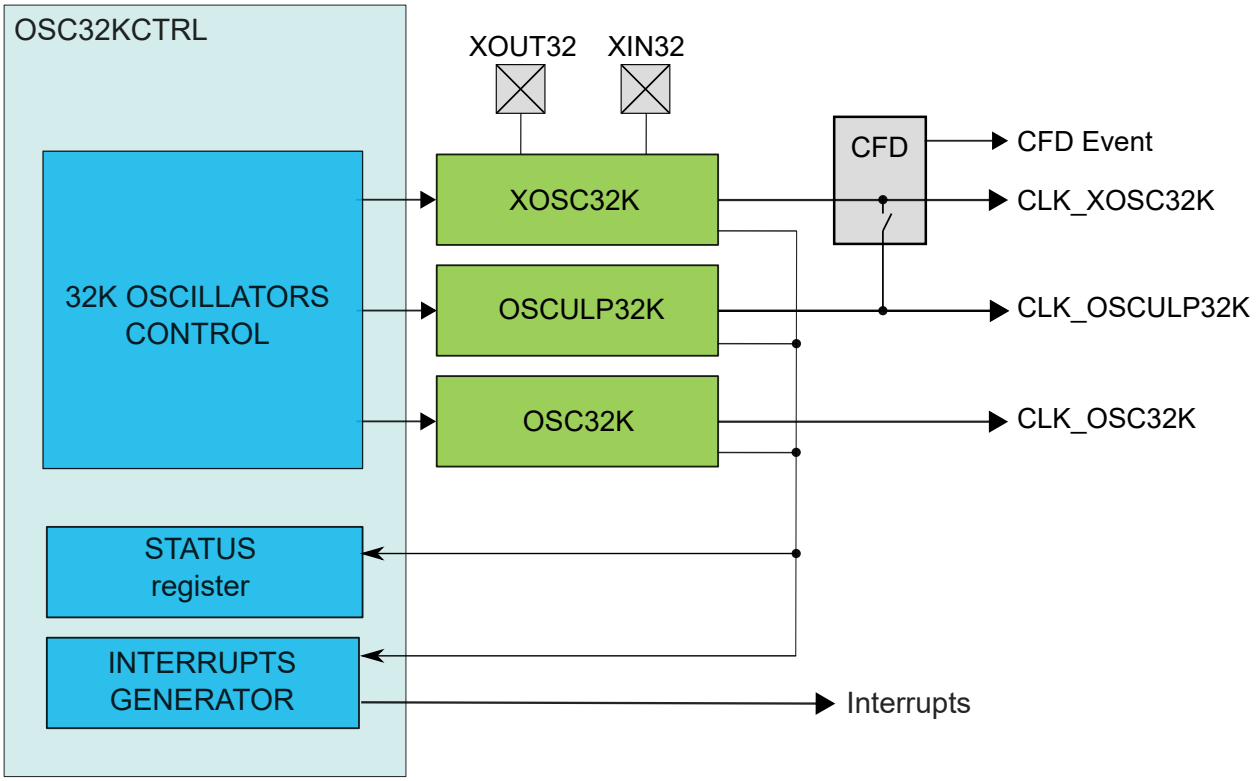
## 21.2. Features

- 32.768 kHz Crystal Oscillator (XOSC32K)
  - Programmable start-up time
  - Crystal or external input clock on XIN32 I/O
  - Clock failure detection with safe clock switch
  - Clock failure event output
- 32.768 kHz High Accuracy Internal Oscillator (OSC32K)
  - Frequency fine tuning
  - Programmable start-up time
- 32.768 kHz Ultra Low-Power Internal Oscillator (OSCULP32K)
  - Ultra low-power, always-on oscillator
  - Frequency fine tuning
- 1.024 kHz clock outputs available


© 2026 Microchip Technology Inc. and its subsidiaries

## 21.3. Block Diagram

**Figure 21-1.** OSC32KCTRL Block Diagram




## 21.4. Signal Description

| Signal | Description | Type |
|---|---|---|
| XIN32 | Analog Input | 32.768 kHz Crystal Oscillator or external clock input |
| XOUT32 | Analog Output | 32.768 kHz Crystal Oscillator output |

The I/O lines are automatically selected when XOSC32K is enabled.

**Note:** The signal of the external crystal oscillator may affect the jitter of neighboring pads.

## 21.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 21.5.1. I/O Lines

I/O lines are configured by OSC32KCTRL when XOSC32K is enabled, and need no user configuration.

### 21.5.2. Power Management

The OSC32KCTRL will continue to operate in any sleep mode where a 32.768 kHz oscillator is running as source clock. The OSC32KCTRL interrupts can be used to wake up the device from sleep modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.5.3. Clocks

The OSC32KCTRL gathers controls for all 32.768 kHz oscillators and provides clock sources to the Generic Clock Controller (GCLK), Real-Time Counter (RTC), and Watchdog Timer (WDT).

The available clock sources are: XOSC32K, OSC32K, and OSCULP32K.

The OSC32KCTRL bus clock (CLK_OSC32KCTRL_APB) can be enabled and disabled in the Main Clock module (MCLK).

**Related Links**
Peripheral Clock Masking

### 21.5.4. Interrupts

The interrupt request lines are connected to the interrupt controller. Using the OSC32KCTRL interrupts requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 21.5.5. Events

The events of this peripheral are connected to the Event System.

**Related Links**
Event System (EVSYS)

### 21.5.6. Debug Operation

When the CPU is halted in debug mode, OSC32KCTRL will continue normal operation. If OSC32KCTRL is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 21.5.7. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Status and Clear (INTFLAG) register

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 21.5.8. Analog Connections

The external 32.768 kHz crystal must be connected between the XIN32 and XOUT32 pins, along with any required load capacitors. For details on recommended oscillator characteristics and capacitor load, refer to the related links.

**Related Links**
Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

### 21.5.9. Calibration

The OSC32K calibration value from the production test must be loaded from the NVM Software Calibration Area into the OSC32K register (OSC32K.CALIB) by software to achieve specified accuracy.

**Related Links**
NVM Software Calibration Area Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

## 21.6. Functional Description

### 21.6.1. Principle of Operation

XOSC32K, OSC32K, and OSCULP32K are configured via OSC32KCTRL control registers. Through this interface, the sub-peripherals are enabled, disabled, or have their calibration values updated.

The STATUS register gathers different status signals coming from the sub-peripherals of OSC32KCTRL. The status signals can be used to generate system interrupts, and in some cases wake up the system from standby mode, provided the corresponding interrupt is enabled.

### 21.6.2. 32.768 kHz External Crystal Oscillator (XOSC32K) Operation

The XOSC32K can operate in two different modes:

- External clock, with an external clock signal connected to XIN32
- Crystal oscillator, with an external 32.768 kHz crystal connected between XIN32 and XOUT32

At reset, the XOSC32K is disabled, and the XIN32/XOUT32 pins can either be used as General Purpose I/O (GPIO) pins or by other peripherals in the system.

When XOSC32K is enabled, the operating mode determines the GPIO usage. When in crystal oscillator mode, the XIN32 and XOUT32 pins are controlled by the OSC32KCTRL, and GPIO functions are overridden on both pins. When in external clock mode, the only XIN32 pin will be overridden and controlled by the OSC32KCTRL, while the XOUT32 pin can still be used as a GPIO pin.

The XOSC32K is enabled by writing a '1' to the Enable bit in the 32.768 kHz External Crystal Oscillator Control register (XOSC32K.ENABLE=1). The XOSC32K is disabled by writing a '0' to the Enable bit in the 32.768 kHz External Crystal Oscillator Control register (XOSC32K.ENABLE=0).

To enable the XOSC32K as a crystal oscillator, the XTALEN bit in the 32.768 kHz External Crystal Oscillator Control register must be set (XOSC32K.XTALEN=1). If XOSC32K.XTALEN is '0', the external clock input will be enabled.

The XOSC32K 32.768 kHz output is enabled by setting the EN32K bit in the 32.768 kHz External Crystal Oscillator Control register (XOSC32K.EN32K=1). The XOSC32K also has a 1.024 kHz clock output, which can only be used by the RTC. This clock output is enabled by setting the EN1K bit in the 32.768 kHz External Crystal Oscillator Control register (XOSC32K.EN1K=1).

It is also possible to lock the XOSC32K configuration by setting the Write Lock bit in the 32.768 kHz External Crystal Oscillator Control register (XOSC32K.WRTLOCK=1). If set, the XOSC32K configuration is locked until a Power-On Reset (POR) is detected.

The XOSC32K will behave differently in different sleep modes based on the settings of XOSC32K.RUNSTDBY, XOSC32K.ONDEMAND, and XOSC32K.ENABLE. If XOSC32KCTRL.ENABLE=0, the XOSC32K will be always stopped. For XOS32KCTRL.ENABLE=1, this table is valid:

**Table 21-1.** XOSC32K Sleep Behavior

| CPU Mode | XOSC32K.RUNSTDBY | XOSC32K.ONDEMAND | Sleep Behavior of XOSC32K and CFD |
|---|---|---|---|
| Active or Idle | - | 0 | Always run |
| Active or Idle | - | 1 | Run if requested by peripheral |
| Standby | 1 | 0 | Always run |
| Standby | 1 | 1 | Run if requested by peripheral |
| Standby | 0 | - | Run if requested by peripheral |

As a crystal oscillator usually requires a very long start-up time, the 32.768 kHz External Crystal Oscillator will keep running across resets when XOSC32K.ONDEMAND=0, except for power-on reset (POR). After a reset or when waking up from a sleep mode where the XOSC32K was disabled, the XOSC32K will need time to stabilize on the correct frequency, depending on the external crystal


© 2026 Microchip Technology Inc. and its subsidiaries

specification. This start-up time can be configured by changing the Oscillator Start-Up Time bit group (XOSC32K.STARTUP) in the 32.768 kHz External Crystal Oscillator Control register. During the start-up time, the oscillator output is masked to ensure that no unstable clock propagates to the digital logic.

Once the external clock or crystal oscillator is stable and ready to be used as a clock source, the XOSC32K Ready bit in the Status register is set (STATUS.XOSC32KRDY=1). The transition of STATUS.XOSC32KRDY from '0' to '1' generates an interrupt if the XOSC32K Ready bit in the Interrupt Enable Set register is set (INTENSET.XOSC32KRDY=1).

The XOSC32K can be used as a source for Generic Clock Generators (GCLK) or for the Real-Time Counter (RTC). Before enabling the GCLK or the RTC module, the corresponding oscillator output must be enabled (XOSC32K.EN32K or XOSC32K.EN1K) in order to ensure proper operation. In the same way, the GCLK or RTC modules must be disabled before the clock selection is changed. For details on RTC clock configuration, refer also to Real-Time Counter Clock Selection.

**Related Links**

GCLK - Generic Clock Controller
RTC – Real-Time Counter

### 21.6.3. Clock Failure Detection Operation

The Clock Failure Detector (CFD) allows the user to monitor the external clock or crystal oscillator signal provided by the external oscillator (XOSC32K). The CFD detects failing operation of the XOSC32K clock with reduced latency, and supports switching to a safe clock source in case of clock failure. The user can also switch from the safe clock back to XOSC32K in case of recovery. The safe clock is derived from the OSCULP32K oscillator with a configurable prescaler. This allows configuring the safe clock in order to fulfill the operative conditions of the microcontroller.

In sleep modes, CFD operation is automatically disabled when the external oscillator is not requested to run by a peripheral. See the Sleep Behavior table above when this is the case.

The user interface registers allow to enable, disable, and configure the CFD. The Status register provides status flags on failure and clock switch conditions. The CFD can optionally trigger an interrupt or an event when a failure is detected.

**Clock Failure Detection**

The CFD is reset only at power-on (POR). The CFD does not monitor the XOSC32K clock when the oscillator is disabled (XOSC32K.ENABLE=0).

Before starting CFD operation, the user must start and enable the safe clock source (OSCULP32K oscillator).

CFD operation is started by writing a '1' to the CFD Enable bit in the External Oscillator Control register (CFDCTRL.CFDEN). After starting or restarting the XOSC32K, the CFD does not detect failure until the start-up time has elapsed. The start-up time is configured by the Oscillator Start-Up Time in the External Multipurpose Crystal Oscillator Control register (XOSC32K.STARTUP). Once the XOSC32K Start-Up Time is elapsed, the XOSC32K clock is constantly monitored.

During a period of 4 safe clocks (monitor period), the CFD watches for a clock activity from the XOSC32K. There must be at least one rising and one falling XOSC32K clock edge during 4 safe clock periods to meet non-failure conditions. If no or insufficient activity is detected, the failure status is asserted: The Clock Failure Detector status bit in the Status register (STATUS.CLKFAIL) and the Clock Failure Detector interrupt flag bit in the Interrupt Flag register (INTFLAG.CLKFAIL) are set. If the CLKFAIL bit in the Interrupt Enable Set register (INTENSET.CLKFAIL) is set, an interrupt is generated as well. If the Event Output enable bit in the Event Control register (EVCTRL.CFDEO) is set, an output event is generated, too.


© 2026 Microchip Technology Inc. and its subsidiaries

After a clock failure was issued the monitoring of the XOSC32K clock is continued, and the Clock Failure Detector status bit in the Status register (STATUS.CLKFAIL) reflects the current XOSC32K activity.

### Clock Switch

When a clock failure is detected, the XOSC32K clock is replaced by the safe clock in order to maintain an active clock during the XOSC32K clock failure. The safe clock source is the OSCULP32K oscillator clock. Both 32.768 kHz and 1.024 kHz outputs of the XOSC32K are replaced by the respective OSCULP32K 32.768 kHz and 1.024 kHz outputs. The safe clock source can be scaled down by a configurable prescaler to ensure that the safe clock frequency does not exceed the operating conditions selected by the application. When the XOSC32K clock is switched to the safe clock, the Clock Switch bit in the Status register (STATUS.CLKSW) is set.

When the CFD has switched to the safe clock, the XOSC32K is not disabled. If desired, the application must take the necessary actions to disable the oscillator. The application must also take the necessary actions to configure the system clocks to continue normal operations. In the case the application can recover the XOSC32K, the application can switch back to the XOSC32K clock by writing a '1' to Switch Back Enable bit in the Clock Failure Control register (CFDCTRL.SWBACK). Once the XOSC32K clock is switched back, the Switch Back bit (CFDCTRL.SWBACK) is cleared by hardware.

### Prescaler

The CFD has an internal configurable prescaler to generate the safe clock from the OSCULP32K oscillator. The prescaler size allows to scale down the OSCULP32K oscillator so the safe clock frequency is not higher than the XOSC32K clock frequency monitored by the CFD. The maximum division factor is 2.

The prescaler is applied on both outputs (32.768 kHz and 1.024 kHz) of the safe clock.

**Example 21-1.** Example

For an external crystal oscillator at 32.768 kHz and the OSCULP32K frequency is 32.768 kHz, the XOSC32K.CFDPRESC should be set to 0 for a safe clock of equal frequency.

### Event

If the Event Output Enable bit in the Event Control register (EVCTRL.CFDEO) is set, the CFD clock failure will be output on the Event Output. When the CFD is switched to the safe clock, the CFD clock failure will not be output on the Event Output.

### Sleep Mode

The CFD is halted depending on configuration of the XOSC32K and the peripheral clock request. For further details, refer to the Sleep Behavior table above. The CFD interrupt can be used to wake up the device from sleep modes.

## 21.6.4. 32.768 kHz Internal Oscillator (OSC32K) Operation

The OSC32K provides a tunable, low-speed, and low-power clock source.

At reset, the OSC32K is disabled. It can be enabled by setting the Enable bit in the 32.768 kHz Internal Oscillator Control register (OSC32K.ENABLE = 1). The OSC32K is disabled by clearing the Enable bit in the 32.768 kHz Internal Oscillator Control register (OSC32K.ENABLE = 0).

The frequency of the OSC32K oscillator is controlled by OSC32K.CALIB. Before using the OSC32K, this calibration field must be loaded with production calibration values from the NVM Software Calibration Area. When writing the calibration bits, the user must wait for the STATUS.OSC32KRDY bit to go high before the new value is committed to the oscillator.

The OSC32K has a 32.768 kHz output which is enabled by setting the EN32K bit in the 32.768 kHz Internal Oscillator Control register (OSC32K.EN32K = 1). The OSC32K also has a 1.024 kHz clock


© 2026 Microchip Technology Inc. and its subsidiaries

output. This is enabled by setting the EN1K bit in the 32.768 kHz Internal Oscillator Control register (OSC32K.EN1K).

It is also possible to lock the OSC32K configuration by setting the Write Lock bit in the 32.768 kHz Internal Oscillator Control register (OSC32K.WRTLOCK = 1). If set, the OSC32K configuration is locked until a Power-on Reset (POR) is detected.

The OSC32K will behave differently in different sleep modes based on the settings of OSC32K.RUNSTDBY, OSC32K.ONDEMAND, and OSC32K.ENABLE. If OSC32KCTRL.ENABLE = 0, the OSC32K will be always stopped. For OS32KCTRL.ENABLE = 1, this table is valid:

**Table 21-2.** OSC32K Sleep Behavior

| CPU Mode | OSC32KCTRL.RUNSTDBY | OSC32KCTRL.ONDEMAND | Sleep Behavior |
|---|---|---|---|
| Active or Idle | - | 0 | Always run |
| Active or Idle | - | 1 | Run if requested by peripheral |
| Standby | 1 | 0 | Always run |
| Standby | 1 | 1 | Run if requested by peripheral |
| Standby | 0 | - | Run if requested by peripheral |

The OSC32K requires a start-up time. For this reason, OSC32K will keep running across resets when OSC32K.ONDEMAND = 0, except for Power-on Reset (POR).

After such a reset, or when waking up from a Sleep mode where the OSC32K was disabled, the OSC32K will need time to stabilize on the correct frequency.

This startup time can be configured by changing the Oscillator Start-Up Time bit group (OSC32K.STARTUP) in the OSC32K Control register. During the start-up time, the oscillator output is masked to ensure that no unstable clock propagates to the digital logic.

Once the external clock or crystal oscillator is stable and ready to be used as a clock source, the OSC32K Ready bit in the Status register is set (STATUS.OSC32KRDY = 1). The transition of STATUS.OSC32KRDY from '0' to '1' generates an interrupt if the OSC32K Ready bit in the Interrupt Enable Set register is set (INTENSET.OSC32KRDY = 1).

The OSC32K can be used as a source for Generic Clock Generators (GCLK) or for the Real-Time Counter (RTC). Before enabling the GCLK or the RTC module, the corresponding oscillator output must be enabled (OSC32K.EN32K or OSC32K.EN1K) in order to ensure proper operation. In the same way, the GCLK or RTC modules must be disabled before the clock selection is changed.

**Related Links**

NVM Software Calibration Area Mapping
RTC – Real-Time Counter
Real-Time Counter Clock Selection

### 21.6.5. 32.768 kHz Ultra-Low-Power Internal Oscillator (OSCULP32K) Operation

The OSCULP32K provides a tunable, low-speed, and ultra-low-power clock source. The OSCULP32K is factory-calibrated under typical voltage and temperature conditions. The OSCULP32K should be preferred to the OSC32K whenever the power requirements are prevalent over frequency stability and accuracy.

The OSCULP32K is enabled by default after a power-on reset (POR) and will always run except during POR. The frequency of the OSCULP32K Oscillator is controlled by the value in the Calibration bits in the 32.768 kHz Ultra-Low-Power Internal Oscillator Control register (OSCULP32K.CALIB). This data is used to compensate for process variations.

OSCULP32K.CALIB is automatically loaded from Flash Factory Calibration during start-up. The calibration value can be overridden by the user by writing to OSCULP32K.CALIB.


© 2026 Microchip Technology Inc. and its subsidiaries

It is also possible to lock the OSCULP32K configuration by setting the Write Lock bit in the 32.768 kHz Ultra-Low-Power Internal Oscillator Control register (OSCULP32K.WRTLOCK = 1). If set, the OSCULP32K configuration is locked until a Power-on Reset (POR) is detected.

The OSCULP32K can be used as a source for Generic Clock Generators (GCLK) or for the Real-Time Counter (RTC). To ensure proper operation, the GCLK or RTC modules must be disabled before the clock selection is changed.

**Related Links**
RTC – Real-Time Counter
Real-Time Counter Clock Selection
GCLK - Generic Clock Controller

### 21.6.6. Watchdog Timer Clock Selection

The Watchdog Timer (WDT) uses the internal 1.024 kHz OSCULP32K output clock. This clock is running all the time and internally enabled when requested by the WDT module.

**Related Links**
WDT – Watchdog Timer

### 21.6.7. Real-Time Counter Clock Selection

Before enabling the RTC module, the RTC clock must be selected first. All oscillator outputs are valid as RTC clock. The selection is done in the RTC Control register (RTCCTRL). To ensure a proper operation, it is highly recommended to disable the RTC module first, before the RTC clock source selection is changed.

**Related Links**
RTC – Real-Time Counter

### 21.6.8. Interrupts

The OSC32KCTRL has the following interrupt sources:

- XOSC32KRDY - 32.768 kHz Crystal Oscillator Ready: A 0-to-1 transition on the STATUS.XOSC32KRDY bit is detected
- CLKFAIL - Clock Failure Detector: A 0-to-1 transition on the STATUS.CLKFAIL bit is detected
- OSC32KRDY - 32.768 kHz Internal Oscillator Ready: A 0-to-1 transition on the STATUS.OSC32KRDY bit is detected

All these interrupts are synchronous wake-up source.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs. Each interrupt can be enabled individually by setting the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by setting the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled or the OSC32KCTRL is reset. See the INTFLAG register for details on how to clear interrupt flags.

The OSC32KCTRL has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present. Refer to the INTFLAG register for details.

**Note:** Interrupts must be globally enabled for interrupt requests to be generated.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
PM - Power Manager
Nested Vector Interrupt Controller

### 21.6.9. Events

The CFD can generate the following output event:

- Clock Failure Detector (CLKFAIL): Generated when the Clock Failure Detector status bit is set in the Status register (STATUS.CLKFAIL). The CFD event is not generated when the Clock Switch bit (STATUS.SWBACK) in the Status register is set.

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.CFDEO) enables the CFD output event. Writing a '0' to this bit disables the CFD output event. Refer to the Event System chapter for details on configuring the event system.


© 2026 Microchip Technology Inc. and its subsidiaries

## 21.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | INTENCLR | 7:0 | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | INTENSET | 7:0 | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTFLAG | 7:0 | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | STATUS | 7:0 | | | | | CLKSW | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | RTCCTRL | 7:0 | | | | | | RTCSEL[2:0] | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | XOSC32K | 7:0 | ONDEMAND | RUNSTDBY | | EN1K | EN32K | XTALEN | ENABLE | |
| | | 15:8 | | | | WRTLOCK | | STARTUP[2:0] | | |
| 0x16 | CFDCTRL | 7:0 | | | | | | CFDPRESC | SWBACK | CFDEN |
| 0x17 | EVCTRL | 7:0 | | | | | | | | CFDEO |
| 0x18 | OSC32K | 7:0 | ONDEMAND | RUNSTDBY | | | EN1K | EN32K | ENABLE | |
| | | 15:8 | | | | WRTLOCK | | STARTUP[2:0] | | |
| | | 23:16 | | CALIB[6:0] | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | OSCULP32K | 7:0 | | | | | | | | |
| | | 15:8 | WRTLOCK | | | CALIB[4:0] | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 21.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register and the 8-bit halves of a 16-bit register can be accessed directly.

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC). Optional Write-Protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in the register description. Write-protection does not apply to accesses through an external debugger.

**Related Links**

PAC - Peripheral Access Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.1. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CLKFAIL** XOSC32K Clock Failure Detection Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the XOSC32K Clock Failure Interrupt Enable bit, which disables the XOSC32K Clock Failure interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC32K Clock Failure Detection is disabled. |
| 1 | The XOSC32K Clock Failure Detection is enabled. An interrupt request will be generated when the XOSC32K Clock Failure Detection interrupt flag is set. |

**Bit 1 – OSC32KRDY** OSC32K Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the OSC32K Ready Interrupt Enable bit, which disables the OSC32K Ready interrupt.

| Value | Description |
|---|---|
| 0 | The OSC32K Ready interrupt is disabled. |
| 1 | The OSC32K Ready interrupt is enabled. |

**Bit 0 – XOSC32KRDY** XOSC32K Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the XOSC32K Ready Interrupt Enable bit, which disables the XOSC32K Ready interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC32K Ready interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The XOSC32K Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.2. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CLKFAIL** XOSC32K Clock Failure Detection Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the XOSC32K Clock Failure Interrupt Enable bit, which enables the XOSC32K Clock Failure interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC32K Clock Failure Detection is disabled. |
| 1 | The XOSC32K Clock Failure Detection is enabled. An interrupt request will be generated when the XOSC32K Clock Failure Detection interrupt flag is set. |

**Bit 1 – OSC32KRDY** OSC32K Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the OSC32K Ready Interrupt Enable bit, which enables the OSC32K Ready interrupt.

| Value | Description |
|---|---|
| 0 | The OSC32K Ready interrupt is disabled. |
| 1 | The OSC32K Ready interrupt is enabled. |

**Bit 0 – XOSC32KRDY** XOSC32K Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the XOSC32K Ready Interrupt Enable bit, which enables the XOSC32K Ready interrupt.

| Value | Description |
|---|---|
| 0 | The XOSC32K Ready interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The XOSC32K Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.3. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CLKFAIL** XOSC32K Clock Failure Detection
This flag is cleared by writing a '1' to it.
This flag is set on a zero-to-one transition of the XOSC32K Clock Failure Detection bit in the Status register (STATUS.CLKFAIL) and will generate an interrupt request if INTENSET.CLKFAIL is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the XOSC32K Clock Failure Detection flag.

**Bit 1 – OSC32KRDY** OSC32K Ready
This flag is cleared by writing a '1' to it.
This flag is set by a zero-to-one transition of the OSC32K Ready bit in the Status register (STATUS.OSC32KRDY), and will generate an interrupt request if INTENSET.OSC32KRDY=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the OSC32K Ready interrupt flag.

**Bit 0 – XOSC32KRDY** XOSC32K Ready
This flag is cleared by writing a '1' to it.
This flag is set by a zero-to-one transition of the XOSC32K Ready bit in the Status register (STATUS.XOSC32KRDY), and will generate an interrupt request if INTENSET.XOSC32KRDY=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the XOSC32K Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.4. Status

**Name:** STATUS
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CLKSW | CLKFAIL | OSC32KRDY | XOSC32KRDY |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – CLKSW** XOSC32K Clock Switch

| Value | Description |
|---|---|
| 0 | XOSC32K is not switched and provided the crystal oscillator. |
| 1 | XOSC32K is switched to be provided by the safe clock. |

**Bit 2 – CLKFAIL** XOSC32K Clock Failure Detector

| Value | Description |
|---|---|
| 0 | XOSC32K is passing failure detection. |
| 1 | XOSC32K is not passing failure detection. |

**Bit 1 – OSC32KRDY** OSC32K Ready

| Value | Description |
|---|---|
| 0 | OSC32K is not ready. |
| 1 | OSC32K is stable and ready to be used as a clock source. |

**Bit 0 – XOSC32KRDY** XOSC32K Ready

| Value | Description |
|---|---|
| 0 | XOSC32K is not ready. |
| 1 | XOSC32K is stable and ready to be used as a clock source. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.5. RTC Clock Selection Control

**Name:** RTCCTRL
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | RTCSEL[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – RTCSEL[2:0]** RTC Clock Source Selection
These bits select the source for the RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | ULP1K | 1.024 kHz from 32.768 kHz internal ULP oscillator |
| 0x1 | ULP32K | 32.768 kHz from 32.768 kHz internal ULP oscillator |
| 0x2 | OSC1K | 1.024 kHz from 32.768 kHz internal oscillator |
| 0x3 | OSC32K | 32.768 kHz from 32.768 kHz internal oscillator |
| 0x4 | XOSC1K | 1.024 kHz from 32.768 kHz external oscillator |
| 0x5 | XOSC32K | 32.768 kHz from 32.768 kHz external crystal oscillator |
| 0x6 | Reserved | |
| 0x7 | Reserved | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.6. 32.768 kHz External Crystal Oscillator (XOSC32K) Control

**Name:** XOSC32K
**Offset:** 0x14
**Reset:** 0x00000080
**Property:** PAC Write-Protection

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | WRTLOCK | | STARTUP[2:0] | | |
| Access | | | | R/W | | R/W | R/W | R/W |
| Reset | | | | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | EN1K | EN32K | XTALEN | ENABLE | |
| Access | R/W | R/W | | R/W | R/W | R/W | R/W | |
| Reset | 1 | 0 | | 0 | 0 | 0 | 0 | |

**Bit 12 – WRTLOCK** Write Lock
This bit locks the XOSC32K register for future writes, effectively freezing the XOSC32K configuration.

| Value | Description |
|---|---|
| 0 | The XOSC32K configuration is not locked. |
| 1 | The XOSC32K configuration is locked. |

**Bits 10:8 – STARTUP[2:0]** Oscillator Start-Up Time
This bit field configures the time after which the XOSC32K clock will be propagated in the design. In order to let a stable clock propagate in the design, the right STARTUP time should be configured after considering the external crystal characteristics and the information provided in the XOSC32K Electrical Specifications section of the Electrical Characteristics chapter. The actual startup time is the number of selected OSCULP32K cycles + 3 XOSC32K cycles.

**Table 21-3.** Start-up Time for 32.768 kHz External Crystal Oscillator

| STARTUP[2:0] | Number of OSCULP32K Clock Cycles | Number of XOSC32K Clock Cycles | Approximate Equivalent Time [s] |
|---|---|---|---|
| 0x0 | 1 | 3 | 122 µs |
| 0x1 | 32 | 3 | 1.06 ms |
| 0x2 | 2048 | 3 | 62.6 ms |
| 0x3 | 4096 | 3 | 125 ms |
| 0x4 | 16384 | 3 | 500 ms |
| 0x5 | 32768 | 3 | 1 s |
| 0x6 | 65536 | 3 | 2 s |
| 0x7 | 131072 | 3 | 4 s |

**Bit 7 – ONDEMAND** On Demand Control
This bit controls how the XOSC32K behaves when a peripheral clock request is detected. For details, refer to XOSC32K Sleep Behavior.

**Bit 6 – RUNSTDBY** Run in Standby
This bit controls how the XOSC32K behaves during standby sleep mode. For details, refer to XOSC32K Sleep Behavior.

**Bit 4 – EN1K** 1.024 kHz Output Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The 1.024 kHz output is disabled. |
| 1 | The 1.024 kHz output is enabled, and available internally only for RTC. |

**Bit 3 – EN32K** 32.768 kHz Output Enable

| Value | Description |
|---|---|
| 0 | The 32.768 kHz output is disabled. |
| 1 | The 32.768 kHz output is enabled, and can be routed to GCLK/GCLK_IO. |

**Bit 2 – XTALEN** Crystal Oscillator Enable

This bit controls the connections between the I/O pads and the external clock or crystal oscillator.

| Value | Description |
|---|---|
| 0 | External clock connected on XIN32. XOUT32 can be used as general-purpose I/O. |
| 1 | Crystal connected to XIN32/XOUT32. |

**Bit 1 – ENABLE** Oscillator Enable

| Value | Description |
|---|---|
| 0 | The oscillator is disabled. |
| 1 | The oscillator is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.7. Clock Failure Detector Control

**Name:** CFDCTRL
**Offset:** 0x16
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CFDPRESC | SWBACK | CFDEN |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CFDPRESC** Clock Failure Detector Prescaler
This bit selects the prescaler for the Clock Failure Detector.

| Value | Description |
|---|---|
| 0 | The CFD safe clock frequency is the OSCULP32K frequency |
| 1 | The CFD safe clock frequency is the OSCULP32K frequency divided by 2 |

**Bit 1 – SWBACK** Clock Switch Back
This bit clontrols the XOSC32K output switch back to the external clock or crystal scillator in case of clock recovery.

| Value | Description |
|---|---|
| 0 | The clock switch is disabled. |
| 1 | The clock switch is enabled. This bit is reset when the XOSC32K output is switched back to the external clock or crystal oscillator. |

**Bit 0 – CFDEN** Clock Failure Detector Enable
This bit selects the Clock Failure Detector state.

| Value | Description |
|---|---|
| 0 | The CFD is disabled. |
| 1 | The CFD is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.8. Event Control

**Name:** EVCTRL
**Offset:** 0x17
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CFDEO |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – CFDEO** Clock Failure Detector Event Out Enable
This bit controls whether the Clock Failure Detector event output is enabled and an event will be generated when the CFD detects a clock failure.

| Value | Description |
|---|---|
| 0 | Clock Failure Detector Event output is disabled, no event will be generated. |
| 1 | Clock Failure Detector Event output is enabled, an event will be generated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.9. 32.768 kHz Internal Oscillator (OSC32K) Control

**Name:** OSC32K
**Offset:** 0x18
**Reset:** 0x0000 0080 (Writing action by User required)
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | CALIB[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | - | - | - | - | - | - | - |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | WRTLOCK | | STARTUP[2:0] | | |
| Access | | | | R/W | | R/W | R/W | R/W |
| Reset | | | | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | EN1K | EN32K | ENABLE | |
| Access | R/W | R/W | | | R/W | R/W | R/W | |
| Reset | 1 | 0 | | | 0 | 0 | 0 | |

**Bits 22:16 – CALIB[6:0]** Oscillator Calibration
These bits control the oscillator calibration. The calibration values must be loaded by the user from the NVM Software Calibration Area.

**Bit 12 – WRTLOCK** Write Lock
This bit locks the OSC32K register for future writes, effectively freezing the OSC32K configuration.

| Value | Description |
|---|---|
| 0 | The OSC32K configuration is not locked. |
| 1 | The OSC32K configuration is locked. |

**Bits 10:8 – STARTUP[2:0]** Oscillator Start-Up Time
These bits select start-up time for the oscillator.
The OSCULP32K oscillator is used as input clock to the start-up counter.

**Table 21-4.** Start-Up Time for 32.768 kHz Internal Oscillator

| STARTUP[2:0] | Number of OSC32K clock cycles | Approximate Equivalent Time [ms] |
|---|---|---|
| 0x0 | 3 | 0.092 |
| 0x1 | 4 | 0.122 |
| 0x2 | 6 | 0.183 |
| 0x3 | 10 | 0.305 |
| 0x4 | 18 | 0.549 |
| 0x5 | 34 | 1.038 |
| 0x6 | 66 | 2.014 |
| 0x7 | 130 | 3.967 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. Start-up time is given by STARTUP + three OSC32K cycles.
2. The given time assumes an XTAL frequency of 32.768 kHz.

**Bit 7 – ONDEMAND** On Demand Control
This bit controls how the OSC32K behaves when a peripheral clock request is detected. For details, refer to OSC32K Sleep Behavior.

**Bit 6 – RUNSTDBY** Run in Standby
This bit controls how the OSC32K behaves during standby sleep mode. For details, refer to OSC32K Sleep Behavior.

**Bit 3 – EN1K** 1.024 kHz Output Enable

| Value | Description |
|---|---|
| 0 | The 1.024 kHz output is disabled. |
| 1 | The 1.024 kHz output is enabled, and available internally only for RTC. |

**Bit 2 – EN32K** 32.768 kHz Output Enable

| Value | Description |
|---|---|
| 0 | The 32.768 kHz output is disabled. |
| 1 | The 32.768 kHz output is enabled, and can be routed to GCLK/GCLK_IO. |

**Bit 1 – ENABLE** Oscillator Enable

| Value | Description |
|---|---|
| 0 | The oscillator is disabled. |
| 1 | The oscillator is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 21.8.10. 32.768 kHz Ultra Low-Power Internal Oscillator (OSCULP32K) Control

**Name:** OSCULP32K
**Offset:** 0x1C
**Reset:** 0x0000XX06
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WRTLOCK | | | CALIB[4:0] | | | | |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | x | x | x | x | x |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 15 – WRTLOCK** Write Lock
This bit locks the OSCULP32K register for future writes to fix the OSCULP32K configuration.

| Value | Description |
|---|---|
| 0 | The OSCULP32K configuration is not locked. |
| 1 | The OSCULP32K configuration is locked. |

**Bits 12:8 – CALIB[4:0]** Oscillator Calibration
These bits control the oscillator calibration.
These bits are automatically loaded from Flash Factory Calibration at startup.


© 2026 Microchip Technology Inc. and its subsidiaries

# 22. SUPC – Supply Controller

## 22.1. Overview

The Supply Controller (SUPC) manages the voltage reference and power supply of the device.

The SUPC controls the voltage regulators for the core (VDDCORE) domain. It sets the voltage regulators according to the sleep modes, or the user configuration.

The SUPC embeds two Brown-Out Detectors: BODVDD monitors the voltage applied to the device (VDD) and BODCORE monitors the internal voltage to the core (VDDCORE). The BOD can monitor the supply voltage continuously (continuous mode) or periodically (sampling mode).

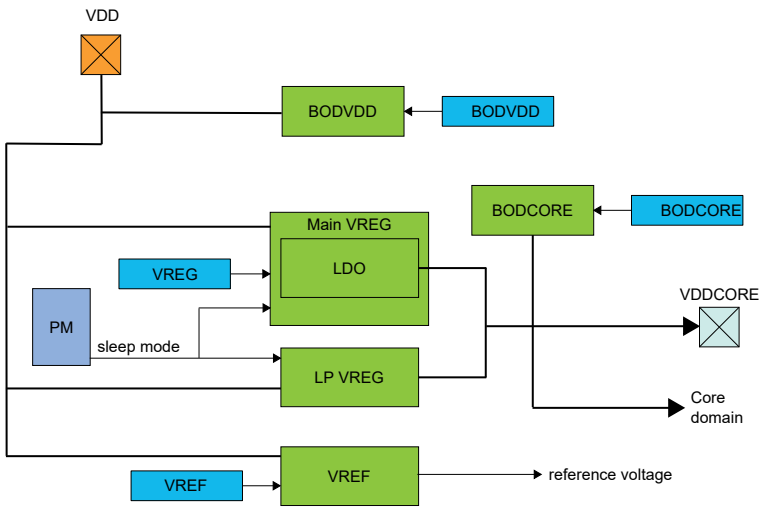
The SUPC generates also a selectable reference voltage which can be used by analog modules like the ADC, SDADC or DAC.

## 22.2. Features

- Voltage Regulator System
  - Main voltage regulator: LDO in active mode (MAINVREG)
  - Low Power voltage regulator in standby mode (LPVREG)
- Voltage Reference System
  - Reference voltage for ADC, SDADC and DAC
- VDD Brown-Out Detector (BODVDD)
  - Programmable threshold
  - Threshold value loaded from NVM User Row at startup
  - Triggers resets or interrupts. Action loaded from NVM User Row
  - Operating modes:
    - Continuous mode
    - Sampled mode for low power applications with programmable sample frequency
  - Hysteresis value from Flash User Calibration
- VDDCORE Brown-Out Detector (BODCORE)
  - Internal non-configurable Brown-Out Detector

## 22.3. Block Diagram

**Figure 22-1.** SUPC Block Diagram





© 2026 Microchip Technology Inc. and its subsidiaries

## 22.4. Signal Description

Not applicable.

**Related Links**

I/O Multiplexing and Considerations

## 22.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 22.5.1. I/O Lines

Not applicable.

### 22.5.2. Power Management

The SUPC can operate in all sleep modes.

**Related Links**

PM - Power Manager

### 22.5.3. Clocks

The SUPC bus clock (CLK_SUPC_APB) can be enabled and disabled in the Main Clock module (MCLK.APBAMASK.SUPC).

A 32.768 kHz clock, asynchronous to the user interface clock (CLK_SUPC_APB), is required to run BODVDD and BODCORE in sampled mode. Due to this asynchronicity, writing to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**

OSC32KCTRL – 32.768 kHz Oscillators Controller
Peripheral Clock Masking

### 22.5.4. DMA

Not applicable.

### 22.5.5. Interrupts

The interrupt request lines are connected to the interrupt controller. Using the SUPC interrupts requires the interrupt controller to be configured first.

**Related Links**

Nested Vector Interrupt Controller

### 22.5.6. Events

Not applicable.

### 22.5.7. Debug Operation

When the CPU is halted in debug mode, the SUPC continues normal operation. If the SUPC is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

If debugger cold-plugging is detected by the system, BODVDD and BODCORE resets will be masked. The BOD resets keep running under hot-plugging. This allows to correct a BODVDD user level too high for the available supply.


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.5.8. Register Access Protection

Registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC).

**Note:** Not all registers with write-access can be write-protected.

PAC Write-Protection is not available for the following registers:

- Interrupt Flag Status and Clear register (INTFLAG)

Optional PAC Write-Protection is denoted by the "PAC Write-Protection" property in each individual register description.

**Related Links**
PAC - Peripheral Access Controller

### 22.5.9. Analog Connections

Not applicable.

## 22.6. Functional Description

### 22.6.1. Voltage Regulator System Operation

#### 22.6.1.1. Enabling, Disabling, and Resetting

The LDO main voltage regulator is enabled after any Reset. The main voltage regulator (MAINVREG) cannot be disabled, therefore the Enable bit in the VREG register (VREG.ENABLE) must never be changed from its default (reset) value of one. The main voltage regulator output supply level is automatically defined by the Sleep mode selected in the Power Manager module.

**Related Links**
PM - Power Manager

#### 22.6.1.2. Initialization

After a Reset, the LDO voltage regulator supplying VDDCORE is enabled.

#### 22.6.1.3. Sleep Mode Operation

In standby mode, the low power voltage regulator (LPVREG) is used to supply VDDCORE.

When the Run in Standby bit in the VREG register (VREG.RUNSTDBY) is written to '1', VDDCORE is supplied by the main voltage regulator. The VDDCORE level is set to the active mode voltage level.

**Related Links**
Sleep Mode Controller

### 22.6.2. Voltage Reference System Operation

The INTREF internal reference voltage is generated by the bandgap in the SUPC. Refer to SUPC.VREF.SEL for voltage level selection.

#### 22.6.2.1. Initialization

The voltage reference output is disabled after any Reset.

#### 22.6.2.2. Enabling, Disabling, and Resetting

The voltage reference output is enabled or disabled by setting or clearing the Voltage Reference Output Enable bit in the Voltage Reference register (VREF.VREFOE).

#### 22.6.2.3. Selecting a Voltage Reference

The Voltage Reference Selection bit field in the VREF register (VREF.SEL) selects the voltage of INTREF (supplied by the bandgap) to be applied to analog modules, for example, ADC.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 22.6.2.4. Sleep Mode Operation

The Voltage Reference output behavior during sleep mode can be configured using the Run in Standby bit and the On Demand bit in the Voltage Reference register (VREF.RUNSTDBY, VREF.ONDEMAND), refer to the table below:

**Table 22-1.** VREF Sleep Mode Operation

| VREF.ONDEMAND | VREF.RUNSTDBY | Voltage Reference Sleep behavior |
|---|---|---|
| 0 | 0 | Always run in all sleep modes *except* Standby Sleep mode |
| 0 | 1 | Always run in all sleep modes *including* Standby Sleep mode |
| 1 | 0 | Only run if requested by the ADC, in all sleep modes *except* Standby Sleep mode |
| 1 | 1 | Only run if requested by the ADC, in all sleep modes *including* Standby Sleep mode |

### 22.6.3. Brown-Out Detectors

#### 22.6.3.1. Initialization

Before a Brown-Out Detector (BODVDD) is enabled, it must be configured, as outlined by the following:

- Set the BOD threshold level (BODVDD.LEVEL)
- Set the configuration in Active, Standby (BODVDD.ACTION, BODVDD.STDBYCFG)
- Set the prescaling value if the BOD will run in sampling mode (BODVDD.PSEL)
- Set the action and hysteresis (BODVDD.ACTION and BODVDD.HYST)

The BODVDD register is Enable-Protected, meaning that they can only be written when the BOD is disabled (BODVDD.ENABLE=0 and STATUS.BVDDSRDY=0). As long as the Enable bit is '1', any writes to Enable-Protected registers will be discarded, and an APB error will be generated. The Enable bits are not Enable-Protected.

#### 22.6.3.2. Enabling, Disabling, and Resetting

After power or user reset, the BODVDD and BODCORE register values are loaded from the NVM User Page.

The BODVDD is enabled by writing a '1' to the Enable bit in the BOD control register (BODVDD.ENABLE). The BOD is disabled by writing a '0' to the BODVDD.ENABLE.

**Related Links**

NVM User Row Mapping

#### 22.6.3.3. VDD Brown-Out Detector (BODVDD)

The VDD Brown-Out Detector (BODVDD) is able to monitor the VDD supply and compares the voltage with the brown-out threshold level set in the BODVDD Level field (BODVDD.LEVEL) in the BODVDD register.

When VDD crosses below the brown-out threshold level, the BODVDD can generate either an interrupt or a Reset, depending on the BODVDD Action bit field (BODVDD.ACTION).

The BODVDD detection status can be read from the BODVDD Detection bit in the Status register (STATUS.BODVDDDET).

At start-up or at Power-On Reset (POR), the BODVDD register values are loaded from the NVM User Row.

**Related Links**

NVM User Row Mapping
BODVDD - Brown Out Detector Characteristics


© 2026 Microchip Technology Inc. and its subsidiaries

#### 22.6.3.4. VDDCORE Brown-Out Detector (BODCORE)

The BODCORE is calibrated in production and its calibration configuration is stored in the NVM User Row. This configuration must not be changed to assure the correct behavior of the BODCORE. The BODCORE generates a reset when VDDCORE crosses below the preset brown-out level. The BODCORE is always disabled in Standby Sleep mode.

**Related Links**

NVM User Row Mapping

#### 22.6.3.5. Continuous Mode

Continuous mode is the default mode for BODVDD.

The BODVDD is continuously monitoring the VDD supply voltage if it is enabled (BODVDD.ENABLE=1) and if the BODVDD Configuration bit in the BODVDD register is cleared (BODVDD.ACTCFG=0 for active mode, BODVDD.STDBYCFG=0 for standby mode).

#### 22.6.3.6. Sampling Mode

The Sampling Mode is a low-power mode where the BODVDD is being repeatedly enabled on a sampling clock's ticks. The BODVDD will monitor the supply voltage for a short period of time and then go to a low-power disabled state until the next sampling clock tick.

Sampling mode is enabled in Active mode for BODVDD by writing the ACTCFG bit (BODVDD.ACTCFG=1). Sampling mode is enabled in Standby mode by writing to the STDBYCFG bit (BODVDD.STBYCFG=1). The frequency of the clock ticks ($F_{clksampling}$) is controlled by the Prescaler Select bit groups in the BODVDD register (BODVDD.PSEL).

$$F_{clksampling} = \frac{F_{clkprescaler}}{2^{(\mathrm{PSEL}+1)}}$$

The prescaler signal ($F_{clkprescaler}$) is a 1.024 kHz clock, output by the 32.768 kHz Ultra Low Power Oscillator OSCULP32K.

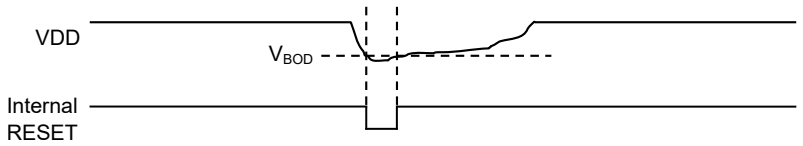
As the sampling clock is different from the APB clock domain, synchronization among the clocks is necessary. See also Synchronization.

#### 22.6.3.7. Hysteresis

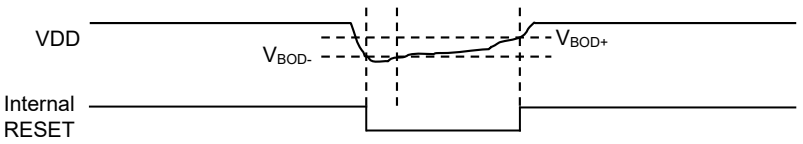
A hysteresis on the trigger threshold of a BOD will reduce the sensitivity to ripples on the monitored voltage: instead of switching $\overline{\text{RESET}}$ at each crossing of $V_{BOD}$, the thresholds for switching $\overline{\text{RESET}}$ on and off are separated ($V_{BOD-}$ and $V_{BOD+}$, respectively).

**Figure 22-2.** BOD Hysteresis Principle

Hysteresis OFF:




Hysteresis ON:




Enabling the BODVDD hysteresis by writing the Hysteresis bit in the BODVDD register (BODVDD.HYST) to '1' will add hysteresis to the BODVDD threshold level.


© 2026 Microchip Technology Inc. and its subsidiaries

The hysteresis functionality can be used in both Continuous and Sampling Mode (See the Electrical Characteristics sections for more information on the hysteresis values).

#### 22.6.3.8. Sleep Mode Operation

##### 22.6.3.8.1. Standby Mode

The BODVDD can be used in standby mode if the BOD is enabled and the corresponding Run in Standby bit is written to '1' (BODVDD.RUNSTDBY).

The BODVDD can be configured to work in either Continuous or Sampling Mode by writing a '1' to the Configuration in Standby Sleep Mode bit (BODVDD.STDBYCFG).

### 22.6.4. Interrupts

The SUPC has the following interrupt sources, which are either synchronous or asynchronous wake-up sources:

- BODVDD Ready (BODVDDRDY), synchronous
- BODVDD Detection (BODVDDDET), asynchronous
- BODVDD Synchronization Ready (BVDDSRDY), synchronous

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled, or the SUPC is reset. See the INTFLAG register for details on how to clear interrupt flags. The SUPC has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present.

**Note:** Interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**

Nested Vector Interrupt Controller

Sleep Mode Controller

### 22.6.5. Synchronization

The prescaler counters that are used to trigger brown-out detections operate asynchronously from the peripheral bus. As a consequence, the BODVDD Enable bit (BODVDD.ENABLE) need synchronization when written.

The Write-Synchronization of the Enable bit is triggered by writing a '1' to the Enable bit of the BODVDD Control register. The Synchronization Ready bit (STATUS.BVDDSRDY) in the STATUS register will be cleared when the Write-Synchronization starts, and set again when the Write-Synchronization is complete. Writing to the same register while the Write-Synchronization is ongoing (STATUS.BVDDSRDY is '0') will generate a PAC error without stalling the APB bus.


© 2026 Microchip Technology Inc. and its subsidiaries

## 22.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | INTENCLR | 7:0 | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | INTENSET | 7:0 | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTFLAG | 7:0 | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | STATUS | 7:0 | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | BODVDD | 7:0 | | RUNSTDBY | STDBYCFG | ACTION[1:0] | | HYST | ENABLE | |
| | | 15:8 | PSEL[3:0] | | | | | | | ACTCFG |
| | | 23:16 | | | LEVEL[5:0] | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | VREG | 7:0 | | RUNSTDBY | | | | | ENABLE | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | VREF | 7:0 | ONDEMAND | RUNSTDBY | | | | VREFOE | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | SEL[3:0] | | | |
| | | 31:24 | | | | | | | | |

## 22.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). PAC Write-protection is denoted by the "PAC Write-Protection" property in each individual register description. Refer to Register Access Protection for details.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. Refer to Synchronization for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.1. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – BVDDSRDY** BODVDD Synchronization Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the BODVDD Synchronization Ready Interrupt Enable bit, which disables the BODVDD Synchronization Ready interrupt.

| Value | Description |
|---|---|
| 0 | The BODVDD Synchronization Ready interrupt is disabled. |
| 1 | The BODVDD Synchronization Ready interrupt is enabled, and an interrupt request will be generated when the BODVDD Synchronization Ready Interrupt flag is set. |

**Bit 1 – BODVDDDET** BODVDD Detection Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the BODVDD Detection Interrupt Enable bit, which disables the BODVDD Detection interrupt.

| Value | Description |
|---|---|
| 0 | The BODVDD Detection interrupt is disabled. |
| 1 | The BODVDD Detection interrupt is enabled, and an interrupt request will be generated when the BODVDD Detection Interrupt flag is set. |

**Bit 0 – BODVDDRDY** BODVDD Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the BODVDD Ready Interrupt Enable bit, which disables the BODVDD Ready interrupt.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The BODVDD Ready interrupt is disabled. |
| 1 | The BODVDD Ready interrupt is enabled, and an interrupt request will be generated when the BODVDD Ready Interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.2. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – BVDDSRDY** BODVDD Synchronization Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the BODVDD Synchronization Ready Interrupt Enable bit, which enables the BODVDD Synchronization Ready interrupt.

| Value | Description |
|---|---|
| 0 | The BODVDD Synchronization Ready interrupt is disabled. |
| 1 | The BODVDD Synchronization Ready interrupt is enabled, and an interrupt request will be generated when the BODVDD Synchronization Ready Interrupt flag is set. |

**Bit 1 – BODVDDDET** BODVDD Detection Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the BODVDD Detection Interrupt Enable bit, which enables the BODVDD Detection interrupt.

| Value | Description |
|---|---|
| 0 | The BODVDD Detection interrupt is disabled. |
| 1 | The BODVDD Detection interrupt is enabled, and an interrupt request will be generated when the BODVDD Detection Interrupt flag is set. |

**Bit 0 – BODVDDRDY** BODVDD Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the BODVDD Ready Interrupt Enable bit, which enables the BODVDD Ready interrupt.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The BODVDD Ready interrupt is disabled. |
| 1 | The BODVDD Ready interrupt is enabled, and an interrupt request will be generated when the BODVDD Ready Interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.3. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x08
**Reset:** X determined from NVM User Row
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | x |

**Bit 2 – BVDDSRDY** BODVDD Synchronization Ready
This flag is cleared by writing a '1' to it.
This flag is set on a zero-to-one transition of the BODVDD Synchronization Ready bit in the Status register (STATUS.BVDDSRDY) and will generate an interrupt request if INTENSET.BVDDSRDY=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the BODVDD Synchronization Ready interrupt flag.

**Bit 1 – BODVDDDET** BODVDD Detection
This flag is cleared by writing a '1' to it.
This flag is set on a zero-to-one transition of the BODVDD Detection bit in the Status register (STATUS.BODVDDDET) and will generate an interrupt request if INTENSET.BODVDDDET=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the BODVDD Detection interrupt flag.

**Bit 0 – BODVDDRDY** BODVDD Ready
This flag is cleared by writing a '1' to it.
This flag is set on a zero-to-one transition of the BODVDD Ready bit in the Status register (STATUS.BODVDDRDY) and will generate an interrupt request if INTENSET.BODVDDRDY=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the BODVDD Ready interrupt flag.
The BODVDD can be enabled.

**Related Links**
NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.4. Status

**Name:** STATUS
**Offset:** 0x0C
**Reset:** Determined from NVM User Row
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BVDDSRDY | BODVDDDET | BODVDDRDY |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | y |

**Bit 2 – BVDDSRDY** BODVDD Synchronization Ready

| Value | Description |
|---|---|
| 0 | BODVDD synchronization is ongoing. |
| 1 | BODVDD synchronization is complete. |

**Bit 1 – BODVDDDET** BODVDD Detection

| Value | Description |
|---|---|
| 0 | No BODVDD detection. |
| 1 | BODVDD has detected that the I/O power supply is going below the BODVDD reference value. |

**Bit 0 – BODVDDRDY** BODVDD Ready

The BODVDD can be enabled at start-up from NVM User Row.
The state of this bit is only applicable in BODVDD continuous mode. In sampling mode, this bit is never set.

| Value | Description |
|---|---|
| 0 | BODVDD is not ready. |
| 1 | BODVDD is ready. |

**Related Links**

NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.5. VDD Brown-Out Detector (BODVDD) Control

**Name:** BODVDD
**Offset:** 0x10
**Reset:** x initially determined from NVM User Row after reset
**Property:** Write-Synchronized, Enable-Protected, PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | LEVEL[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | x | x | x | x | x | x |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PSEL[3:0] | | | | | | | ACTCFG |
| Access | R/W | R/W | R/W | R/W | | | | R/W |
| Reset | 0 | 0 | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | STDBYCFG | ACTION[1:0] | | HYST | ENABLE | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | |
| Reset | | 0 | 0 | x | x | x | x | |

**Bits 21:16 – LEVEL[5:0]** BODVDD Threshold Level on VDD
These bits set the triggering voltage threshold for the BODVDD when the BODVDD monitors the VDD .
These bits are loaded from NVM User Row at start-up.
This bit field is not synchronized.

**Bits 15:12 – PSEL[3:0]** Prescaler Select
Selects the prescaler divide-by output for the BODVDD sampling mode. The input clock comes from the OSCULP32K 1.024 kHz output.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | Divide clock by 2 |
| 0x1 | DIV4 | Divide clock by 4 |
| 0x2 | DIV8 | Divide clock by 8 |
| 0x3 | DIV16 | Divide clock by 16 |
| 0x4 | DIV32 | Divide clock by 32 |
| 0x5 | DIV64 | Divide clock by 64 |
| 0x6 | DIV128 | Divide clock by 128 |
| 0x7 | DIV256 | Divide clock by 256 |
| 0x8 | DIV512 | Divide clock by 512 |
| 0x9 | DIV1024 | Divide clock by 1024 |
| 0xA | DIV2048 | Divide clock by 2048 |
| 0xB | DIV4096 | Divide clock by 4096 |
| 0xC | DIV8192 | Divide clock by 8192 |
| 0xD | DIV16384 | Divide clock by 16384 |
| 0xE | DIV32768 | Divide clock by 32768 |
| 0xF | DIV65536 | Divide clock by 65536 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 8 – ACTCFG** BODVDD Configuration in Active Sleep Mode
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | In active mode, the BODVDD operates in continuous mode. |
| 1 | In active mode, the BODVDD operates in sampling mode. |

**Bit 6 – RUNSTDBY** Run in Standby
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | In standby sleep mode, the BODVDD is disabled. |
| 1 | In standby sleep mode, the BODVDD is enabled. |

**Bit 5 – STDBYCFG** BODVDD Configuration in Standby Sleep Mode
If the RUNSTDBY bit is set to '1', the STDBYCFG bit sets the BODVDD configuration in standby sleep mode.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | In standby sleep mode, the BODVDD is enabled and configured in continuous mode. |
| 1 | In standby sleep mode, the BODVDD is enabled and configured in sampling mode. |

**Bits 4:3 – ACTION[1:0]** BODVDD Action
These bits are used to select the BODVDD action when the supply voltage crosses below the BODVDD threshold.
These bits are loaded from NVM User Row at start-up.
This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RESET | The BODVDD generates a reset |
| 0x2 | INT | The BODVDD generates an interrupt |
| 0x3 | - | Reserved |

**Bit 2 – HYST** Hysteresis
This bit indicates whether hysteresis is enabled for the BODVDD threshold voltage.
This bit is loaded from NVM User Row at start-up.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | No hysteresis. |
| 1 | Hysteresis enabled. |

**Bit 1 – ENABLE** Enable
This bit is loaded from NVM User Row at start-up.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | BODVDD is enabled. |
| 1 | BODVDD is disabled. |

**Related Links**
Electrical Characteristics 85°C (SAM C20/C21 E/G/J)
NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.6. Voltage Regulator System (VREG) Control

**Name:** VREG
**Offset:** 0x18
**Reset:** 0x00000002
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | | | | ENABLE | |
| Access | R | R/W | R | R | R | R | R/W | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

**Bit 6 – RUNSTDBY** Run in Standby

| Value | Description |
|---|---|
| 0 | The voltage regulator is in Low-Power mode in Standby-Sleep mode. |
| 1 | The voltage regulator is in normal mode in Standby-Sleep mode. |

**Bit 1 – ENABLE** Main Voltage Regulator Enable

The Main Voltage Regulator is automatically enabled after every reset. The main voltage regulator (MAINVREG) cannot be disabled, hence this bit must never be changed from its reset value of one.

| Value | Description |
|---|---|
| 0 | The voltage regulator is disabled. |
| 1 | The voltage regulator is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 22.8.7. Voltage References System (VREF) Control

**Name:** VREF
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SEL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | | VREFOE | | |
| Access | R/W | R/W | | | | R/W | | |
| Reset | 0 | 0 | | | | 0 | | |

**Bits 19:16 – SEL[3:0]** Voltage Reference Selection
These bits select the Voltage Reference for the ADC/ SDADC/DAC.

| Value | Description |
|---|---|
| 0x0 | 1.024V voltage reference typical value |
| 0x2 | 2.048V voltage reference typical value |
| 0x3 | 4.096V voltage reference typical value |
| Others | Reserved |

**Bit 7 – ONDEMAND** On Demand Control
The On Demand operation mode allows to enable or disable the voltage reference depending on peripheral requests.

| Value | Description |
|---|---|
| 0 | The voltage reference is always on, if enabled. |
| 1 | The voltage reference is enabled when a peripheral is requesting it. The voltage reference is disabled if no peripheral is requesting it. |

**Bit 6 – RUNSTDBY** Run In Standby
The bit controls how the voltage reference behaves during Standby Sleep mode.

| Value | Description |
|---|---|
| 0 | The voltage reference is halted during Standby Sleep mode. |
| 1 | The voltage reference is not stopped in Standby Sleep mode. If VREF.ONDEMAND=1, the voltage reference will be running when a peripheral is requesting it. If VREF.ONDEMAND=0, the voltage reference will always be running in Standby Sleep mode. |

**Bit 2 – VREFOE** Voltage Reference Output Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The Voltage Reference output is not available as an ADC input channel. |
| 1 | The Voltage Reference output is routed to an ADC input channel. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 23. WDT – Watchdog Timer

## 23.1. Overview

The Watchdog Timer (WDT) is a system function for monitoring correct program operation. It makes it possible to recover from error situations such as runaway or deadlocked code. The WDT is configured to a predefined time-out period, and is constantly running when enabled. If the WDT is not cleared within the time-out period, it will issue a system reset. An early-warning interrupt is available to indicate an upcoming watchdog time-out condition.

The window mode makes it possible to define a time slot (or window) inside the total time-out period during which the WDT must be cleared. If the WDT is cleared outside this window, either too early or too late, a system reset will be issued. Compared to the normal mode, this can also catch situations where a code error causes the WDT to be cleared too frequently.

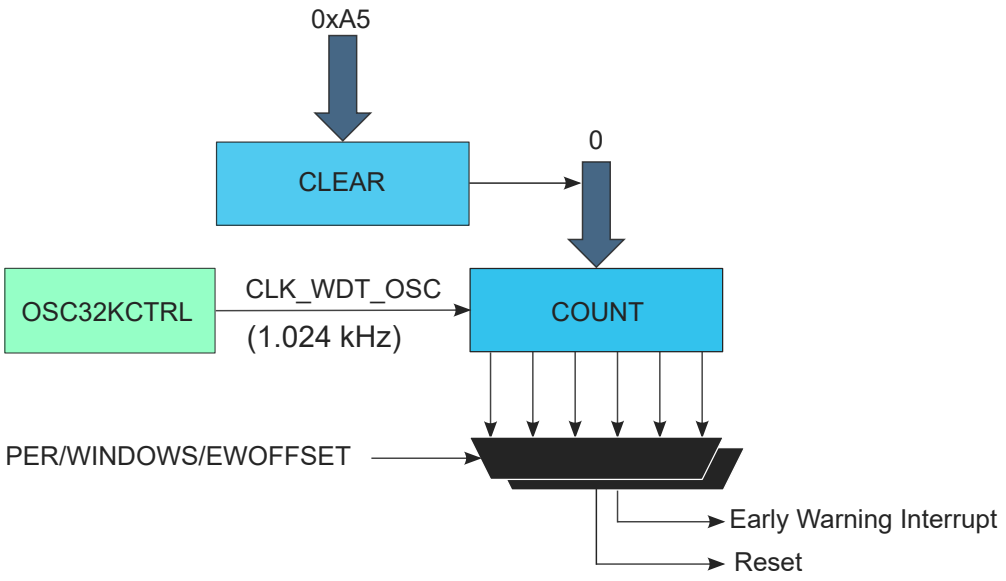
When enabled, the WDT will run in active, idle and standby modes. It is asynchronous and runs from a CPU-independent clock source. The WDT will continue operation and issue a system reset or interrupt even if the main clocks fail.

## 23.2. Features

- Issues a system reset if the Watchdog Timer is not cleared before its time-out period
- Early Warning interrupt generation
- Asynchronous operation from dedicated oscillator
- Two types of operation
  - Normal
  - Window mode
- Selectable time-out periods
  - From 8 cycles to 16,384 cycles in Normal mode
  - From 16 cycles to 32,768 cycles in Window mode
- Always-On capability

## 23.3. Block Diagram

**Figure 23-1.** WDT Block Diagram





© 2026 Microchip Technology Inc. and its subsidiaries

## 23.4. Signal Description

Not applicable.

## 23.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 23.5.1. I/O Lines

Not applicable.

### 23.5.2. Power Management

The WDT can continue to operate in any sleep mode where the selected source clock is running. The WDT interrupts can be used to wake up the device from sleep modes. The events can trigger other operations in the system without exiting sleep modes.

**Related Links**

PM - Power Manager

### 23.5.3. Clocks

The WDT bus clock (CLK_WDT_APB) can be enabled and disabled (masked) in the Main Clock module (MCLK).

A 1.024 kHz oscillator clock (CLK_WDT_OSC) is required to clock the WDT internal counter.

The CLK_WDT_OSC CLOCK is sourced from the clock of the internal Ultra Low-Power Oscillator (OSCULP32K). Due to ultra low-power design, the oscillator is not very accurate, therefore the exact time-out period may vary from device-to-device. This variation must be considered when designing software that uses the WDT to ensure that the time-out periods used are valid for all devices.

The counter clock CLK_WDT_OSC is asynchronous to the bus clock (CLK_WDT_APB). Due to this asynchronicity, writing to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**

Peripheral Clock Masking
OSC32KCTRL – 32.768 kHz Oscillators Controller

### 23.5.4. DMA

Not applicable.

### 23.5.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the WDT interrupt(s) requires the interrupt controller to be configured first.

**Related Links**

Nested Vector Interrupt Controller
Overview
Interrupt Line Mapping

### 23.5.6. Events

Not applicable.

### 23.5.7. Debug Operation

When the CPU is halted in debug mode the WDT will halt normal operation.


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Status and Clear (INTFLAG) register

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

### 23.5.9. Analog Connections

Not applicable.

## 23.6. Functional Description

### 23.6.1. Principle of Operation

The Watchdog Timer (WDT) is a system for monitoring correct program operation, making it possible to recover from error situations such as runaway code, by issuing a Reset. When enabled, the WDT is a constantly running timer that is configured to a predefined time-out period. Before the end of the time-out period, the WDT should be set back, or else, a system Reset is issued.

The WDT has two modes of operation, Normal mode and Window mode. Both modes offer the option of Early Warning interrupt generation. The description for each of the basic modes is given below. The settings in the Control A register (CTRLA) and the Interrupt Enable register (handled by INTENCLR/INTENSET) determine the mode of operation:

**Table 23-1.** WDT Operating Modes

| CTRLA.ENABLE | CTRLA.WEN | Interrupt Enable | Mode |
|---|---|---|---|
| 0 | x | x | Stopped |
| 1 | 0 | 0 | Normal mode |
| 1 | 0 | 1 | Normal mode with Early Warning interrupt |
| 1 | 1 | 0 | Window mode |
| 1 | 1 | 1 | Window mode with Early Warning interrupt |

### 23.6.2. Basic Operation

#### 23.6.2.1. Initialization

The following bits are enable-protected, meaning that they can only be written when the WDT is disabled (CTRLA.ENABLE=0):

- Control A register (CTRLA), except the Enable bit (CTRLA.ENABLE)
- Configuration register (CONFIG)
- Early Warning Interrupt Control register (EWCTRL)

Enable-protected bits in the CTRLA register can be written at the same time as CTRLA.ENABLE is written to '1', but not at the same time as CTRLA.ENABLE is written to '0'.

The WDT can be configured only while the WDT is disabled. The WDT is configured by defining the required Time-Out Period bits in the Configuration register (CONFIG.PER). If Window mode operation is desired, the Window Enable bit in the Control A register must be set (CTRLA.WEN=1) and the Window Period bits in the Configuration register (CONFIG.WINDOW) must be defined.

Enable-protection is denoted by the "Enable-Protected" property in the register description.

#### 23.6.2.2. Configurable Reset Values

After a Power-on Reset, some registers will be loaded with initial values from the NVM User Row.


© 2026 Microchip Technology Inc. and its subsidiaries

This includes the following bits and bit groups:

- Enable bit in the Control A register, CTRLA.ENABLE
- Always-On bit in the Control A register, CTRLA.ALWAYSON
- Watchdog Timer Windows Mode Enable bit in the Control A register, CTRLA.WEN
- Watchdog Timer Windows Mode Time-Out Period bits in the Configuration register, CONFIG.WINDOW
- Time-Out Period bits in the Configuration register, CONFIG.PER
- Early Warning Interrupt Time Offset bits in the Early Warning Interrupt Control register, EWCTRL.EWOFFSET

**Related Links**

NVM User Row Mapping

### 23.6.2.3. Enabling, Disabling, and Resetting

The WDT is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The WDT is disabled by writing a '0' to CTRLA.ENABLE.

The WDT can be disabled only if the Always-On bit in the Control A register (CTRLA.ALWAYSON) is '0'.

### 23.6.2.4. Normal Mode

In Normal mode operation, the length of a time-out period is configured in CONFIG.PER. The WDT is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). Once enabled, the WDT will issue a system reset if a time-out occurs. This can be prevented by clearing the WDT at any time during the time-out period.

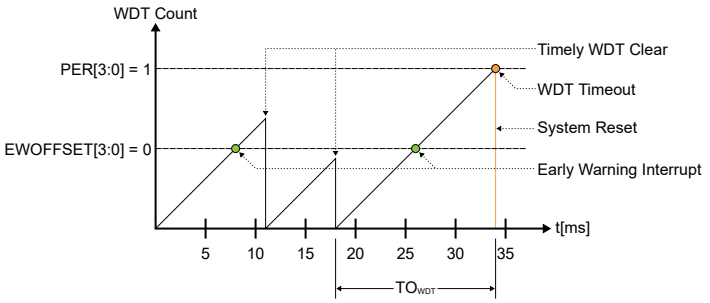
The WDT is cleared and a new WDT time-out period is started by writing 0xA5 to the Clear register (CLEAR). Writing any other value than 0xA5 to CLEAR will issue an immediate system reset.

There are 12 possible WDT time-out ($TO_{WDT}$) periods, selectable from 8ms to 16s.

By default, the early warning interrupt is disabled. If it is desired, the Early Warning Interrupt Enable bit in the Interrupt Enable register (INTENSET.EW) must be written to '1'. The Early Warning Interrupt is disabled again by writing a '1' to the Early Warning Interrupt bit in the Interrupt Enable Clear register (INTENCLR.EW).

If the Early Warning Interrupt is enabled, an interrupt is generated prior to a WDT time-out condition. In Normal mode, the Early Warning Offset bits in the Early Warning Interrupt Control register, EWCTRL.EWOFFSET, define the time when the early warning interrupt occurs. The Normal mode operation is illustrated in the figure Normal-Mode Operation.

**Figure 23-2.** Normal-Mode Operation




### 23.6.2.5. Window Mode

In Window mode operation, the WDT uses two different time specifications: the WDT can only be cleared by writing 0xA5 to the CLEAR register *after* the closed window time-out period ($TO_{WDTW}$),


© 2026 Microchip Technology Inc. and its subsidiaries

during the subsequent Normal time-out period ($TO_{WDT}$). If the WDT is cleared before the time window opens (before $TO_{WDTW}$ is over), the WDT will issue a system reset.

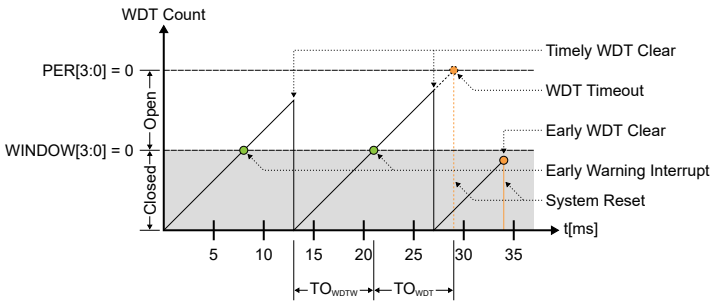
Both parameters $TO_{WDTW}$ and $TO_{WDT}$ are periods in a range from 8ms to 16s, so the total duration of the WDT time-out period is the sum of the two parameters.

The closed window period is defined by the Window Period bits in the Configuration register (CONFIG.WINDOW), and the open window period is defined by the Period bits in the Configuration register (CONFIG.PER).

By default, the Early Warning interrupt is disabled. If it is desired, the Early Warning Interrupt Enable bit in the Interrupt Enable register (INTENSET.EW) must be written to '1'. The Early Warning Interrupt is disabled again by writing a '1' to the Early Warning Interrupt bit in the Interrupt Enable Clear (INTENCLR.EW) register.

If the Early Warning interrupt is enabled in Window mode, the interrupt is generated at the start of the open window period, i.e. after $TO_{WDTW}$. The Window mode operation is illustrated in figure Window-Mode Operation.

**Figure 23-3.** Window-Mode Operation




### 23.6.3. DMA Operation

Not applicable.

### 23.6.4. Interrupts

The WDT has the following interrupt source:

- Early Warning (EW): Indicates that the counter is approaching the time-out condition.
  - This interrupt is an asynchronous wake-up source.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the WDT is reset. See the INTFLAG register description for details on how to clear interrupt flags. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

**Note:** Interrupts must be globally enabled for interrupt requests to be generated.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Nested Vector Interrupt Controller
Overview
Interrupt Line Mapping
PM - Power Manager
Sleep Mode Controller

### 23.6.5. Events

Not applicable.

### 23.6.6. Sleep Mode Operation

**Related Links**
CTRLA
Control A

### 23.6.7. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following registers are synchronized when written:

- Enable bit in Control A register (CTRLA.ENABLE)
- Window Enable bit in Control A register (CTRLA.WEN)
- Always-On bit in control Control A (CTRLA.ALWAYSON)
- Watchdog Clear register (CLEAR)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

### 23.6.8. Additional Features

#### 23.6.8.1. Always-On Mode

The Always-On mode is enabled by setting the Always-On bit in the Control A register (CTRLA.ALWAYSON=1). When the Always-On mode is enabled, the WDT runs continuously, regardless of the state of CTRLA.ENABLE. Once written, the Always-On bit can only be cleared by a power-on reset. The Configuration (CONFIG) and Early Warning Control (EWCTRL) registers are read-only registers while the CTRLA.ALWAYSON bit is set. Thus, the time period configuration bits (CONFIG.PER, CONFIG.WINDOW, EWCTRL.EWOFFSET) of the WDT cannot be changed.

Enabling or disabling Window mode operation by writing the Window Enable bit (CTRLA.WEN) is allowed while in Always-On mode, but note that CONFIG.PER cannot be changed.

The Interrupt Clear and Interrupt Set registers are accessible in the Always-On mode. The Early Warning interrupt can still be enabled or disabled while in the Always-On mode, but note that EWCTRL.EWOFFSET cannot be changed.

Table WDT Operating Modes With Always-On shows the operation of the WDT for CTRLA.ALWAYSON=1.

**Table 23-2.** WDT Operating Modes With Always-On

| WEN | Interrupt Enable | Mode |
|---|---|---|
| 0 | 0 | Always-on and normal mode |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 23-2.** WDT Operating Modes With Always-On (continued)

| WEN | Interrupt Enable | Mode |
|---|---|---|
| 0 | 1 | Always-on and normal mode with Early Warning interrupt |
| 1 | 0 | Always-on and window mode |
| 1 | 1 | Always-on and window mode with Early Warning interrupt |

#### 23.6.8.2. Early Warning

The Early Warning interrupt notifies that the WDT is approaching its time-out condition. The Early Warning interrupt behaves differently in Normal mode and in Window mode.

*In Normal mode*, the Early Warning interrupt generation is defined by the Early Warning Offset in the Early Warning Control register (EWCTRL.EWOFFSET). The Early Warning Offset bits define the number of CLK_WDT_OSC clocks before the interrupt is generated, relative to the start of the watchdog time-out period.

The user must take caution when programming the Early Warning Offset bits. If these bits define an Early Warning interrupt generation time greater than the watchdog time-out period, the watchdog time-out system reset is generated prior to the Early Warning interrupt. Consequently, the Early Warning interrupt will never be generated.

*In window mode*, the Early Warning interrupt is generated at the start of the open window period. In a typical application where the system is in sleep mode, the Early Warning interrupt can be used to wake up and clear the Watchdog Timer, after which the system can perform other tasks or return to sleep mode.

**Example:** If the WDT is operating in Normal mode with CONFIG.PER = 0x2 and EWCTRL.EWOFFSET = 0x1, the Early Warning interrupt is generated 16 CLK_WDT_OSC clock cycles after the start of the time-out period. The time-out system reset is generated 32 CLK_WDT_OSC clock cycles after the start of the watchdog time-out period.


© 2026 Microchip Technology Inc. and its subsidiaries

## 23.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ALWAYSON | | | | | WEN | ENABLE | |
| 0x01 | CONFIG | 7:0 | WINDOW[3:0] | | | | PER[3:0] | | | |
| 0x02 | EWCTRL | 7:0 | | | | | EWOFFSET[3:0] | | | |
| 0x03 | Reserved | | | | | | | | | |
| 0x04 | INTENCLR | 7:0 | | | | | | | | EW |
| 0x05 | INTENSET | 7:0 | | | | | | | | EW |
| 0x06 | INTFLAG | 7:0 | | | | | | | | EW |
| 0x07 | Reserved | | | | | | | | | |
| 0x08 | SYNCBUSY | 7:0 | | | | CLEAR | ALWAYSON | WEN | ENABLE | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | CLEAR | 7:0 | CLEAR[7:0] | | | | | | | |

## 23.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** x initially determined from NVM User Row after reset
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ALWAYSON | | | | | WEN | ENABLE | |
| Access | R/W | | | | | R/W | R/W | |
| Reset | x | | | | | x | x | |

**Bit 7 – ALWAYSON** Always-On
This bit allows the WDT to run continuously. After being set, this bit cannot be written to '0', and the WDT will remain enabled until a power-on Reset is received. When this bit is '1', the Control A register (CTRLA), the Configuration register (CONFIG) and the Early Warning Control register (EWCTRL) will be read-only, and any writes to these registers are not allowed.
Writing a '0' to this bit has no effect.
This bit is not Enable-Protected.
This bit is loaded from NVM User Row at start-up.

| Value | Description |
|---|---|
| 0 | The WDT is enabled and disabled through the ENABLE bit. |
| 1 | The WDT is enabled and can only be disabled by a power-on reset (POR). |

**Bit 2 – WEN** Watchdog Timer Window Mode Enable
This bit enables Window mode. It can only be written if the peripheral is disabled unless CTRLA.ALWAYSON=1.
This bit is loaded from NVM User Row at startup.

| Value | Description |
|---|---|
| 0 | Window mode is disabled (normal operation). |
| 1 | Window mode is enabled. |

**Bit 1 – ENABLE** Enable
This bit enables or disables the WDT. It can only be written if CTRLA.ALWAYSON=0.
Due to synchronization, there is delay between writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately, and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not Enable-Protected.
This bit is loaded from NVM User Row at startup.

| Value | Description |
|---|---|
| 0 | The WDT is disabled. |
| 1 | The WDT is enabled. |

**Related Links**
NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.2. Configuration

**Name:** CONFIG
**Offset:** 0x01
**Reset:** x initially determined from NVM User Row after reset
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINDOW[3:0] | | | | PER[3:0] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | x | x | x | x | x | x | x | x |

**Bits 7:4 – WINDOW[3:0]** Window Mode Time-Out Period
In Window mode, these bits determine the watchdog closed window period as a number of cycles of the 1.024 kHz CLK_WDT_OSC clock.
These bits are loaded from NVM User Row at start-up.

| Value | Name | Description |
|---|---|---|
| 0x0 | CYC8 | 8 clock cycles |
| 0x1 | CYC16 | 16 clock cycles |
| 0x2 | CYC32 | 32 clock cycles |
| 0x3 | CYC64 | 64 clock cycles |
| 0x4 | CYC128 | 128 clock cycles |
| 0x5 | CYC256 | 256 clock cycles |
| 0x6 | CYC512 | 512 clock cycles |
| 0x7 | CYC1024 | 1024 clock cycles |
| 0x8 | CYC2048 | 2048 clock cycles |
| 0x9 | CYC4096 | 4096 clock cycles |
| 0xA | CYC8192 | 8192 clock cycles |
| 0xB | CYC16384 | 16384 clock cycles |
| 0xC-0xF | Reserved | Reserved |

**Bits 3:0 – PER[3:0]** Time-Out Period
These bits determine the watchdog time-out period as a number of 1.024 kHz CLK_WDTOSC clock cycles. In Window mode operation, these bits define the open window period.
These bits are loaded from NVM User Row at startup.

| Value | Name | Description |
|---|---|---|
| 0x0 | CYC8 | 8 clock cycles |
| 0x1 | CYC16 | 16 clock cycles |
| 0x2 | CYC32 | 32 clock cycles |
| 0x3 | CYC64 | 64 clock cycles |
| 0x4 | CYC128 | 128 clock cycles |
| 0x5 | CYC256 | 256 clock cycles |
| 0x6 | CYC512 | 512 clock cycles |
| 0x7 | CYC1024 | 1024 clock cycles |
| 0x8 | CYC2048 | 2048 clock cycles |
| 0x9 | CYC4096 | 4096 clock cycles |
| 0xA | CYC8192 | 8192 clock cycles |
| 0xB | CYC16384 | 16384 clock cycles |
| 0xC - 0xF | - | Reserved |

**Related Links**
NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.3. Early Warning Control

**Name:** EWCTRL
**Offset:** 0x02
**Reset:** x initially determined from NVM User Row after reset
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | EWOFFSET[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | x | x | x | x |

**Bits 3:0 – EWOFFSET[3:0]** Early Warning Interrupt Time Offset
These bits determine the number of CLK_WDT_OSC clock cycles between the start of the watchdog time-out period and the generation of the Early Warning interrupt. These bits are loaded from NVM User Row at start-up.

| Value | Name | Description |
|---|---|---|
| 0x0 | CYC8 | 8 clock cycles |
| 0x1 | CYC16 | 16 clock cycles |
| 0x2 | CYC32 | 32 clock cycles |
| 0x3 | CYC64 | 64 clock cycles |
| 0x4 | CYC128 | 128 clock cycles |
| 0x5 | CYC256 | 256 clock cycles |
| 0x6 | CYC512 | 512 clock cycles |
| 0x7 | CYC1024 | 1024 clock cycles |
| 0x8 | CYC2048 | 2048 clock cycles |
| 0x9 | CYC4096 | 4096 clock cycles |
| 0xA | CYC8192 | 8192 clock cycles |
| 0xB - 0xF | Reserved | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | EW |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – EW** Early Warning Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Early Warning Interrupt Enable bit, which disables the Early Warning interrupt.

| Value | Description |
|---|---|
| 0 | The Early Warning interrupt is disabled. |
| 1 | The Early Warning interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | EW |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – EW** Early Warning Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit sets the Early Warning Interrupt Enable bit, which enables the Early Warning interrupt.

| Value | Description |
|---|---|
| 0 | The Early Warning interrupt is disabled. |
| 1 | The Early Warning interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x06
**Reset:** 0x00
**Property:** N/A

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | EW |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – EW** Early Warning
This flag is cleared by writing a '1' to it.
This flag is set when an Early Warning interrupt occurs, as defined by the EWOFFSET bit group in EWCTRL.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Early Warning interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.7. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | CLEAR | ALWAYSON | WEN | ENABLE | |
| Access | | | | R | R | R | R | |
| Reset | | | | 0 | 0 | 0 | 0 | |

**Bit 4 – CLEAR** Clear Synchronization Busy

| Value | Description |
|---|---|
| 0 | Write synchronization of the CLEAR register is complete. |
| 1 | Write synchronization of the CLEAR register is ongoing. |

**Bit 3 – ALWAYSON** Always-On Synchronization Busy

| Value | Description |
|---|---|
| 0 | Write synchronization of the CTRLA.ALWAYSON bit is complete. |
| 1 | Write synchronization of the CTRLA.ALWAYSON bit is ongoing. |

**Bit 2 – WEN** Window Enable Synchronization Busy

| Value | Description |
|---|---|
| 0 | Write synchronization of the CTRLA.WEN bit is complete. |
| 1 | Write synchronization of the CTRLA.WEN bit is ongoing. |

**Bit 1 – ENABLE** Enable Synchronization Busy

| Value | Description |
|---|---|
| 0 | Write synchronization of the CTRLA.ENABLE bit is complete. |
| 1 | Write synchronization of the CTRLA.ENABLE bit is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 23.8.8. Clear

**Name:** CLEAR
**Offset:** 0x0C
**Reset:** 0x00
**Property:** Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CLEAR[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – CLEAR[7:0]** Watchdog Clear
In Normal mode, writing 0xA5 to this register during the watchdog time-out period will clear the Watchdog Timer and the watchdog time-out period is restarted.
In Window mode, any writing attempt to this register before the time-out period started (i.e., during $TO_{WDTW}$) will issue an immediate system Reset. Writing 0xA5 during the time-out period $TO_{WDT}$ will clear the Watchdog Timer and the complete time-out sequence (first $TO_{WDTW}$ then $TO_{WDT}$) is restarted.
In both modes, writing any other value than 0xA5 will issue an immediate system Reset.


© 2026 Microchip Technology Inc. and its subsidiaries

# 24. RTC – Real-Time Counter

## 24.1. Overview

The Real-Time Counter (RTC) is a 32-bit counter with a 10-bit programmable prescaler that typically runs continuously to keep track of time. The RTC can wake up the device from sleep modes using the alarm/compare wake up, periodic wake up, or overflow wake up mechanisms.

The RTC can generate periodic peripheral events from outputs of the prescaler, as well as alarm/compare interrupts and peripheral events, which can trigger at any counter value. Additionally, the timer can trigger an overflow interrupt and peripheral event, and can be reset on the occurrence of an alarm/compare match. This allows periodic interrupts and peripheral events at very long and accurate intervals.

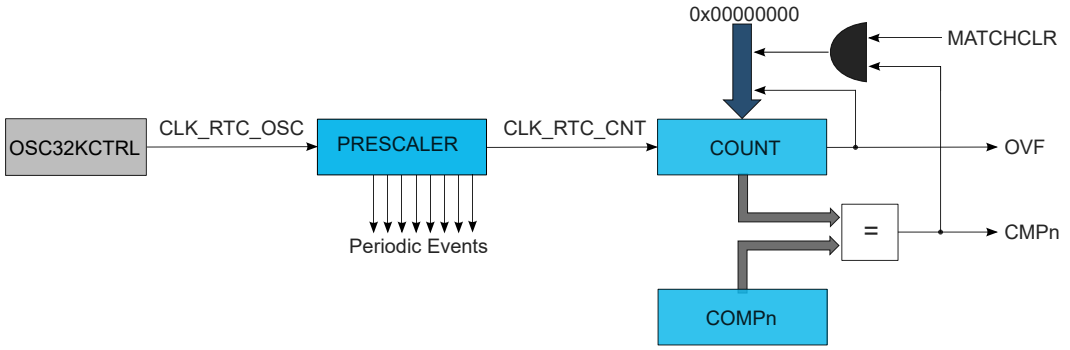
The 10-bit programmable prescaler can scale down the clock source. By this, a wide range of resolutions and time-out periods can be configured. With a 32.768 kHz clock source, the minimum counter tick interval is 30.5µs, and time-out periods can range up to 36 hours. For a counter tick interval of 1s, the maximum time-out period is more than 136 years.

## 24.2. Features

- 32-bit counter with 10-bit prescaler
- Multiple clock sources
- 32-bit or 16-bit counter mode
- One 32-bit or two 16-bit compare values
- Clock/Calendar mode
  - Time in seconds, minutes, and hours (12/24)
  - Date in day of month, month, and year
  - Leap year correction
- Digital prescaler correction/tuning for increased accuracy
- Overflow, alarm/compare match and prescaler interrupts and events
  - Optional clear on alarm/compare match

## 24.3. Block Diagram

**Figure 24-1.** RTC Block Diagram (Mode 0 — 32-Bit Counter)





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 24-2.** RTC Block Diagram (Mode 1 — 16-Bit Counter)

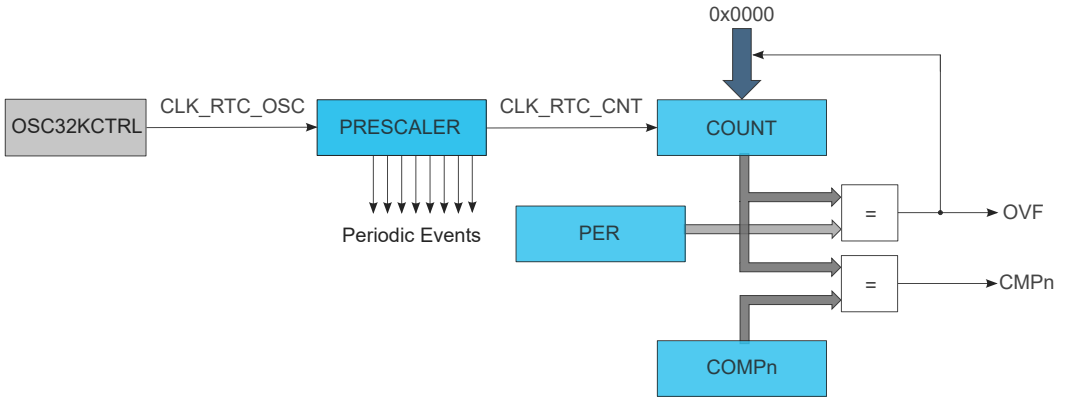
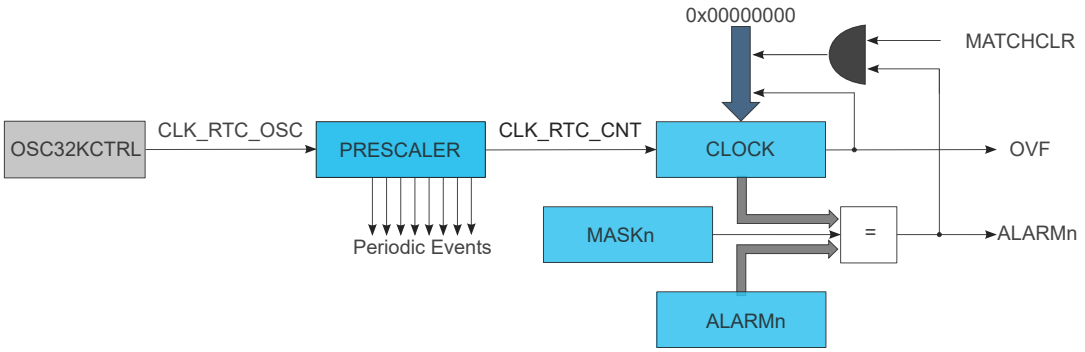



**Figure 24-3.** RTC Block Diagram (Mode 2 — Clock/Calendar)




**Related Links**

32-Bit Counter (Mode 0)
16-Bit Counter (Mode 1)
Clock/Calendar (Mode 2)

## 24.4. Signal Description

Not applicable.

## 24.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 24.5.1. I/O Lines

For more information on I/O configurations, refer to the "RTC Pinout" section.

Related Links: I/O Multiplexing and Considerations

### 24.5.2. Power Management

The RTC will continue to operate in any sleep modes where the selected source clock is running. The RTC interrupts can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes. Refer to the *Power Manager* for details on the different sleep modes.

The RTC can only be reset by a power on reset (POR) or by setting the Software Reset bit in the Control A register (CTRLA.SWRST=1).


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
PM - Power Manager

### 24.5.3. Clocks

The RTC bus clock (CLK_RTC_APB) can be enabled and disabled in the Main Clock module MCLK, and the default state of CLK_RTC_APB can be found in Peripheral Clock Masking section.

A 32.768 kHz or 1.024 kHz oscillator clock (CLK_RTC_OSC) is required to clock the RTC. This clock must be configured and enabled in the 32.768 kHz oscillator controller (OSC32KCTRL RTCCTRL.RTCSEL) before using the RTC.

This oscillator clock is asynchronous to the bus clock (CLK_RTC_APB). Due to this asynchronicity, writing to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
OSC32KCTRL – 32.768 kHz Oscillators Controller
Peripheral Clock Masking

### 24.5.4. DMA

Not applicable.

**Related Links**
DMAC – Direct Memory Access Controller

### 24.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the RTC interrupt requires the Interrupt Controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 24.5.6. Events

The events are connected to the *Event System*.

**Related Links**
Event System (EVSYS)

### 24.5.7. Debug Operation

When the CPU is halted in debug mode the RTC will halt normal operation. The RTC can be forced to continue operation during debugging. Refer to DBGCTRL for details.

### 24.5.8. Register Access Protection

All registers with write-access are optionally write-protected by the Peripheral Access Controller (PAC), except the following registers:

- Interrupt Flag Status and Clear (INTFLAG) register

Write-protection is denoted by the "PAC Write-Protection" property in the register description.

Write-protection does not apply to accesses through an external debugger. Refer to the PAC - Peripheral Access Controller for details.

**Related Links**
PAC - Peripheral Access Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.5.9. Analog Connections

A 32.768 kHz crystal can be connected to the XIN32 and XOUT32 pins, along with any required load capacitors. See the Electrical Characteristics Chapters for details on recommended crystal characteristics and load capacitors.

## 24.6. Functional Description

### 24.6.1. Principle of Operation

The RTC keeps track of time in the system and enables periodic events, as well as interrupts and events at a specified time. The RTC consists of a 10-bit prescaler that feeds a 32-bit counter. The actual format of the 32-bit counter depends on the RTC operating mode.

The RTC can function in one of these modes:

- Mode 0 - COUNT32: RTC serves as 32-bit counter
- Mode 1 - COUNT16: RTC serves as 16-bit counter
- Mode 2 - CLOCK: RTC serves as clock/calendar with alarm functionality

### 24.6.2. Basic Operation

#### 24.6.2.1. Initialization

The following bits are enable-protected, meaning that they can only be written when the RTC is disabled (CTRLA.ENABLE = 0):

- Operating Mode bits in the Control A register (CTRLA.MODE)
- Prescaler bits in the Control A register (CTRLA.PRESCALER)
- Clear on Match bit in the Control A register (CTRLA.MATCHCLR)
- Clock Representation bit in the Control A register (CTRLA.CLKREP)

The following registers are enable-protected:

- Event Control register (EVCTRL)

Enable-protected bits and registers can be changed only when the RTC is disabled (CTRLA.ENABLE = 0). If the RTC is enabled (CTRLA.ENABLE = 1), these operations are necessary: first write CTRLA.ENABLE = 0 and check whether the write synchronization has finished, then change the desired bit field value. Enable-protected bits in CTRLA register can be written at the same time as CTRLA.ENABLE is written to '1', but not at the same time as CTRLA.ENABLE is written to '0'.

Enable-protection is denoted by the "Enable-Protected" property in the register description.

The RTC prescaler divides the source clock for the RTC counter.

**Note:** In Clock/Calendar mode, the prescaler must be configured to provide a 1 Hz clock to the counter for correct operation.

The frequency of the RTC clock (CLK_RTC_CNT) is given by the following formula:

$$f_{CLK_RTC_CNT} = \frac{f_{CLK_RTC_OSC}}{2^{PRESCALER-1}}$$ when PRESCALER is not 0, else $f_{CLK_RTC_CNT} = f_{CLK_RTC_OSC}$

The frequency of the oscillator clock, CLK_RTC_OSC, is given by $f_{CLK_RTC_OSC}$, and $f_{CLK_RTC_CNT}$ is the frequency of the internal prescaled RTC clock, CLK_RTC_CNT.

#### 24.6.2.2. Enabling, Disabling, and Resetting

The RTC is enabled by setting the Enable bit in the Control A register (CTRLA.ENABLE=1). The RTC is disabled by writing CTRLA.ENABLE=0.


© 2026 Microchip Technology Inc. and its subsidiaries

The RTC is reset by setting the Software Reset bit in the Control A register (CTRLA.SWRST=1). All registers in the RTC, except DEBUG, will be reset to their initial state, and the RTC will be disabled. The RTC must be disabled before resetting it.

#### 24.6.2.3. 32-Bit Counter (Mode 0)

When the RTC Operating Mode bits in the Control A register (CTRLA.MODE) are written to 0x0, the counter operates in 32-bit Counter mode. The block diagram of this mode is shown in Figure 24-1. When the RTC is enabled, the counter will increment on every 0-to-1 transition of CLK_RTC_CNT. The counter will increment until it reaches the top value of 0xFFFFFFFF, and then wrap to 0x00000000. This sets the Overflow Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF).

The RTC counter value can be read from or written to the Counter Value register (COUNT) in 32-bit format.

The counter value is continuously compared with the 32-bit Compare register (COMP0). When a compare match occurs, the Compare 0 Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMP0) is set on the next 0-to-1 transition of CLK_RTC_CNT.

If the Clear on Match bit in the Control A register (CTRLA.MATCHCLR) is '1', the counter is cleared on the next counter cycle when a compare match with COMP0 occurs. This allows the RTC to generate periodic interrupts or events with longer periods than the prescaler events. Note that when CTRLA.MATCHCLR is '1', INTFLAG.CMP0 and INTFLAG.OVF will both be set simultaneously on a compare match with COMP0.

#### 24.6.2.4. 16-Bit Counter (Mode 1)

When the RTC Operating Mode bits in the Control A register (CTRLA.MODE) are written to 0x1, the counter operates in 16-bit Counter mode as shown in Figure 24-2. When the RTC is enabled, the counter will increment on every 0-to-1 transition of CLK_RTC_CNT. In 16-bit Counter mode, the 16-bit Period register (PER) holds the maximum value of the counter. The counter will increment until it reaches the PER value, and then wrap to 0x0000. This sets the Overflow Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF).

The RTC counter value can be read from or written to the Counter Value register (COUNT) in 16-bit format.

The counter value is continuously compared with the 16-bit Compare registers (COMPn, n=0..1). When a compare match occurs, the Compare n Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMPn, n=0..1) is set on the next 0-to-1 transition of CLK_RTC_CNT.

#### 24.6.2.5. Clock/Calendar (Mode 2)

When the RTC Operating Mode bits in the Control A register (CTRLA.MODE) are written to 0x2, the counter operates in Clock/Calendar mode, as shown in Figure 24-3. When the RTC is enabled, the counter will increment on every 0-to-1 transition of CLK_RTC_CNT. The selected clock source and RTC prescaler must be configured to provide a 1Hz clock to the counter for correct operation in this mode.

The time and date can be read from or written to the Clock Value register (CLOCK) in a 32-bit time/date format. Time is represented as:

- Seconds
- Minutes
- Hours

Hours can be represented in either 12- or 24-hour format, selected by the Clock Representation bit in the Control A register (CTRLA.CLKREP). This bit can be changed only while the RTC is disabled.

The date is represented in this form:

- Day as the numeric day of the month (starting at 1)


© 2026 Microchip Technology Inc. and its subsidiaries

- Month as the numeric month of the year (1 = January, 2 = February, etc.)
- Year as a value from 0x00 to 0x3F. This value must be added to a user-defined reference year. The reference year must be a leap year (2016, 2020 etc). Example: the year value 0x2D, added to a reference year 2016, represents the year 2061.

The RTC will increment until it reaches the top value of 23:59:59 December 31 of year value 0x3F, and then wrap to 00:00:00 January 1 of year value 0x00. This will set the Overflow Interrupt flag in the Interrupt Flag Status and Clear registers (INTFLAG.OVF).

The clock value is continuously compared with the 32-bit Alarm register (ALARM0). When an alarm match occurs, the Alarm 0 Interrupt flag in the Interrupt Flag Status and Clear registers (INTFLAG.ALARM0) is set on the next 0-to-1 transition of CLK_RTC_CNT. E.g. For a 1Hz clock counter, it means the Alarm 0 Interrupt flag is set with a delay of 1s after the occurrence of alarm match.

A valid alarm match depends on the setting of the Alarm Mask Selection bits in the Alarm 0 Mask register (MASK0.SEL). These bits determine which time/date fields of the clock and alarm values are valid for comparison and which are ignored.

If the Clear on Match bit in the Control A register (CTRLA.MATCHCLR) is set, the counter is cleared on the next counter cycle when an alarm match with ALARM0 occurs. This allows the RTC to generate periodic interrupts or events with longer periods than it would be possible with the prescaler events only (see Periodic Intervals).
**Note:** When CTRLA.MATCHCLR is 1, INTFLAG.ALARM0 and INTFLAG.OVF will both be set simultaneously on an alarm match with ALARM0.

### 24.6.3. DMA Operation

Not applicable.

### 24.6.4. Interrupts

The RTC has the following interrupt sources:

- Overflow (OVF): Indicates that the counter has reached its top value and wrapped to zero.
- Compare (CMPn): Indicates a match between the counter value and the compare register.
- Alarm (ALARM): Indicates a match between the clock value and the alarm register.
- Period n (PERn): The corresponding bit in the prescaler has toggled. Refer to Periodic Intervals for details.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by setting the corresponding bit in the Interrupt Enable Set register (INTENSET=1), and disabled by setting the corresponding bit in the Interrupt Enable Clear register (INTENCLR=1). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is raised and the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled or the RTC is reset. See the description of the INTFLAG registers for details on how to clear interrupt flags.

All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. Refer to the Nested Vector Interrupt Controller for details. The user must read the INTFLAG register to determine which interrupt condition is present.

**Note:** Interrupts must be globally enabled for interrupt requests to be generated. Refer to the Nested Vector Interrupt Controller for details.

**Related Links**
Nested Vector Interrupt Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.6.5. Events

The RTC can generate the following output events:

- Overflow (OVF): Generated when the counter has reached its top value and wrapped to zero.
- Compare (CMPn): Indicates a match between the counter value and the compare register.
- Alarm (ALARM): Indicates a match between the clock value and the alarm register.
- Period n (PERn): The corresponding bit in the prescaler has toggled. Refer to Periodic Intervals for details.
- Periodic Daily (PERD): Generated when the COUNT/CLOCK has incremented at a fixed period of time.

Setting the Event Output bit in the Event Control Register (EVCTRL.xxxEO=1) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. Refer to the EVSYS - Event System for details on configuring the event system.

**Related Links**
Event System (EVSYS)

### 24.6.6. Sleep Mode Operation

The RTC will continue to operate in any sleep mode where the source clock is active. The RTC *interrupts* can be used to wake up the device from a sleep mode. RTC *events* can trigger other operations in the system without exiting the sleep mode.

An interrupt request will be generated after the wake-up if the Interrupt Controller is configured accordingly. Otherwise the CPU will wake up directly, without triggering any interrupt. In this case, the CPU will continue executing right from the first instruction that followed the entry into sleep.

### 24.6.7. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in Control A register, CTRLA.SWRST
- Enable bit in Control A register, CTRLA.ENABLE

The following registers are synchronized when written:

- Counter Value register, COUNT
- Clock Value register, CLOCK
- Counter Period register, PER
- Compare n Value registers, COMPn
- Alarm n Value registers, ALARMn
- Frequency Correction register, FREQCORR
- Alarm n Mask register, MASKn

The following registers are synchronized when read:

- The Counter Value register, COUNT, if the Counter Read Sync Enable bit in CTRLA (CTRLA.COUNTSYNC) is '1'
- The Clock Value register, CLOCK, if the Clock Read Sync Enable bit in CTRLA (CTRLA.CLOCKSYNC) is '1'

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

**Related Links**

Register Synchronization

## 24.6.8. Additional Features

### 24.6.8.1. Periodic Intervals

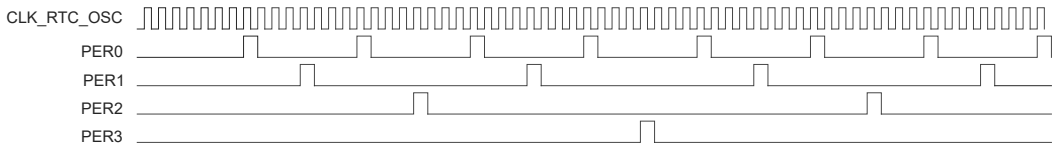
The RTC prescaler can generate interrupts and events at periodic intervals, allowing flexible system tick creation. Any of the upper eight bits of the prescaler (bits 2 to 9) can be the source of an interrupt/event. When one of the eight Periodic Event Output bits in the Event Control register (EVCTRL.PEREO[n=0..7]) is '1', an event is generated on the 0-to-1 transition of the related bit in the prescaler, resulting in a periodic event frequency of:

$$f_{PERIODIC(n)} = \frac{f_{CLK_RTC_OSC}}{2^{n+3}}$$

$f_{CLK_RTC_OSC}$ is the frequency of the internal prescaler clock CLK_RTC_OSC, and n is the position of the EVCTRL.PEREOn bit. For example, PER0 will generate an event every eight CLK_RTC_OSC cycles, PER1 every 16 cycles, etc. This is shown in the figure below.

Periodic events are independent of the prescaler setting used by the RTC counter, except if CTRLA.PRESCALER is zero. Then, no periodic events will be generated.

**Figure 24-4.** Example Periodic Events




**Note:** The same applies for interrupts enabled in INTENSET/CLR.

### 24.6.8.2. Frequency Correction

The RTC Frequency Correction module employs periodic counter corrections to compensate for a too-slow or too-fast oscillator. Frequency correction requires that CTRLA.PRESCALER is greater than 1.

The digital correction circuit adds or subtracts cycles from the RTC prescaler to adjust the frequency in approximately 1ppm steps. Digital correction is achieved by adding or skipping a single count in the prescaler once every 4096 CLK_RTC_OSC cycles. The Value bit group in the Frequency Correction register (FREQCORR.VALUE) determines the number of times the adjustment is applied over 240 of these periods. The resulting correction is as follows:

$$\text{Correction in ppm} = \frac{\text{FREQCORR.VALUE}}{4096 \cdot 240} \cdot 10^6 \text{ppm}$$

This results in a resolution of 1.017ppm.

The Sign bit in the Frequency Correction register (FREQCORR.SIGN) determines the direction of the correction. A positive value will add counts and increase the period (reducing the frequency), and a negative value will reduce counts per period (speeding up the frequency).

Digital correction also affects the generation of the periodic events from the prescaler. When the correction is applied at the end of the correction cycle period, the interval between the previous periodic event and the next occurrence may also be shortened or lengthened depending on the correction value.


© 2026 Microchip Technology Inc. and its subsidiaries

## 24.7. Register Summary - Mode 0 - 32-Bit Counter

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | MATCHCLR | | | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | PRESCALER[3:0] | | | |
| 0x02 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | | | | | | | CMPEO0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | CMP0 |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | CMP0 |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | CMP0 |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | | COMP0 | | COUNT | FREQCORR | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | FREQCORR | 7:0 | SIGN | VALUE[6:0] | | | | | | |
| 0x15 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| | | 23:16 | COUNT[23:16] | | | | | | | |
| | | 31:24 | COUNT[31:24] | | | | | | | |
| 0x1C ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | COMP0 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |
| | | 23:16 | COMP[23:16] | | | | | | | |
| | | 31:24 | COMP[31:24] | | | | | | | |

## 24.8. Register Description - Mode 0 - 32-Bit Counter

This Register Description section is valid if the RTC is in COUNT32 mode (CTRLA.MODE=0).

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.1. Control A in COUNT32 mode (CTRLA.MODE=0)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | | | | PRESCALER[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MATCHCLR | | | | MODE[1:0] | | ENABLE | SWRST |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 15 – COUNTSYNC** COUNT Read Synchronization Enable
The COUNT register requires synchronization when reading. Disabling the synchronization will prevent reading valid values from the COUNT register.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | COUNT read synchronization is disabled |
| 1 | COUNT read synchronization is enabled |

**Bits 11:8 – PRESCALER[3:0]** Prescaler
These bits define the prescaling factor for the RTC clock source (GCLK_RTC) to generate the counter clock (CLK_RTC_CNT). Periodic events and interrupts are not available when the prescaler is off.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x1 | DIV1 | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x2 | DIV2 | CLK_RTC_CNT = GCLK_RTC/2 |
| 0x3 | DIV4 | CLK_RTC_CNT = GCLK_RTC/4 |
| 0x4 | DIV8 | CLK_RTC_CNT = GCLK_RTC/8 |
| 0x5 | DIV16 | CLK_RTC_CNT = GCLK_RTC/16 |
| 0x6 | DIV32 | CLK_RTC_CNT = GCLK_RTC/32 |
| 0x7 | DIV64 | CLK_RTC_CNT = GCLK_RTC/64 |
| 0x8 | DIV128 | CLK_RTC_CNT = GCLK_RTC/128 |
| 0x9 | DIV256 | CLK_RTC_CNT = GCLK_RTC/256 |
| 0xA | DIV512 | CLK_RTC_CNT = GCLK_RTC/512 |
| 0xB | DIV1024 | CLK_RTC_CNT = GCLK_RTC/1024 |
| 0xC-0xF | - | Reserved |

**Bit 7 – MATCHCLR** Clear on Match
This bit defines if the counter is cleared or not on a match.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The counter is not cleared on a Compare/Alarm 0 match |
| 1 | The counter is cleared on a Compare/Alarm 0 match |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 3:2 – MODE[1:0]** Operating Mode

This bit group defines the operating mode of the RTC.
This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT32 | Mode 0: 32-bit counter |
| 0x1 | COUNT16 | Mode 1: 16-bit counter |
| 0x2 | CLOCK | Mode 2: Clock/calendar |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** Enable

Due to synchronization there is a delay between writing CTRLA.ENABLE and until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled |
| 1 | The peripheral is enabled |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the RTC (except DBGCTRL) to their initial state, and the RTC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay between writing CTRLA.SWRST and until the reset is complete. CTRLA.SWRST will be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is not reset operation ongoing |
| 1 | The reset operation is ongoing |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.2. Event Control in COUNT32 mode (CTRLA.MODE=0)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | | | | | | | CMPEO0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVFEO** Overflow Event Output Enable

| Value | Description |
|---|---|
| 0 | Overflow event is disabled and will not be generated. |
| 1 | Overflow event is enabled and will be generated for every overflow. |

**Bit 8 – CMPEO0** Compare 0 Event Output Enable

| Value | Description |
|---|---|
| 0 | Compare 0 event is disabled and will not be generated. |
| 1 | Compare 0 event is enabled and will be generated for every compare match. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PEREOn** Periodic Interval n Event Output Enable [n = 7..0]

| Value | Description |
|---|---|
| 0 | Periodic Interval n event is disabled and will not be generated. |
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.3. Interrupt Enable Clear in COUNT32 mode (CTRLA.MODE=0)

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | CMP0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bit 8 – CMP0** Compare 0 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Compare 0 Interrupt Enable bit, which disables the Compare 0 interrupt.

| Value | Description |
|---|---|
| 0 | The Compare 0 interrupt is disabled. |
| 1 | The Compare 0 interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Periodic Interval n Interrupt Enable bit, which disables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.4. Interrupt Enable Set in COUNT32 mode (CTRLA.MODE=0)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | CMP0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bit 8 – CMP0** Compare 0 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Compare 0 Interrupt Enable bit, which enables the Compare 0 interrupt.

| Value | Description |
|---|---|
| 0 | The Compare 0 interrupt is disabled. |
| 1 | The Compare 0 interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Periodic Interval n Interrupt Enable bit, which enables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.5. Interrupt Flag Status and Clear in COUNT32 mode (CTRLA.MODE=0)

**Name:** INTFLAG
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | CMP0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after an overflow condition occurs, and an interrupt request will be generated if INTENCLR/SET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

**Bit 8 – CMP0** Compare 0
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after a match with the compare condition, and an interrupt request will be generated if INTENCLR/SET.COMP0 is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare 0 interrupt flag.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n [n = 7..0]
This flag is cleared by writing a '1' to the flag.
This flag is set on the 0-to-1 transition of prescaler bit [n+2], and an interrupt request will be generated if INTENCLR/SET.PERn is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Periodic Interval n interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.6. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The RTC is halted when the CPU is halted by an external debugger. |
| 1 | The RTC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.7. Synchronization Busy in COUNT32 mode (CTRLA.MODE=0)

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | | | | | | | |
| Access | R | | | | | | | |
| Reset | 0 | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | COMP0 | | COUNT | FREQCORR | ENABLE | SWRST |
| Access | | | R | | R | R | R | R |
| Reset | | | 0 | | 0 | 0 | 0 | 0 |

**Bit 15 – COUNTSYNC** Count Read Sync Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.COUNTSYNC bit is complete. |
| 1 | Write synchronization for CTRLA.COUNTSYNC bit is ongoing. |

**Bit 5 – COMP0** Compare 0 Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for COMP0 register is complete. |
| 1 | Write synchronization for COMP0 register is ongoing. |

**Bit 3 – COUNT** Count Value Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Read/write synchronization for COUNT register is complete. |
| 1 | Read/write synchronization for COUNT register is ongoing. |

**Bit 2 – FREQCORR** Frequency Correction Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for FREQCORR register is complete. |
| 1 | Write synchronization for FREQCORR register is ongoing. |

**Bit 1 – ENABLE** Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.ENABLE bit is complete. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Write synchronization for CTRLA.ENABLE bit is ongoing. |

**Bit 0 – SWRST** Software Reset Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.8. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.FREQCORR must be checked to ensure the FREQCORR register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SIGN | VALUE[6:0] | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – SIGN** Correction Sign

| Value | Description |
|---|---|
| 0 | The correction value is positive, i.e., frequency will be decreased. |
| 1 | The correction value is negative, i.e., frequency will be increased. |

**Bits 6:0 – VALUE[6:0]** Correction Value
These bits define the amount of correction applied to the RTC prescaler.

| Value | Description |
|---|---|
| 0 | Correction is disabled and the RTC frequency is unchanged. |
| 1 - 127 | The RTC frequency is adjusted according to the value. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.9. Counter Value in COUNT32 mode (CTRLA.MODE = 0)

**Name:** COUNT
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

**Notes:**

1. This register is read-synchronized and write-synchronized: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.
2. This register must be written with 32-bit accesses only.
3. Prior to any read access, this register must be synchronized by the user by writing the COUNT Read Synchronization Enable bit in the Control A register (CTRLA.COUNTSYNC = 1).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – COUNT[31:0]** Counter Value
These bits define the value of the 32-bit RTC counter in mode 0.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.8.10. Compare 0 Value in COUNT32 mode (CTRLA.MODE=0)

**Name:** COMP0
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.COMP0 must be checked to ensure the COMP0 register synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | COMP[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | COMP[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COMP[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COMP[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – COMP[31:0]** Compare Value
The 32-bit value of COMP0 is continuously compared with the 32-bit COUNT value. When a match occurs, the Compare 0 interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMP0) is set on the next counter cycle, and the counter value is cleared if CTRLA.MATCHCLR is '1'.


© 2026 Microchip Technology Inc. and its subsidiaries

## 24.9. Register Summary - Mode 1 - 16-Bit Counter

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | PRESCALER[3:0] | | | |
| 0x02 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | | | | | | CMPEO1 | CMPEO0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | CMP1 | CMP0 |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | CMP1 | CMP0 |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | CMP1 | CMP0 |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | COMP1 | COMP0 | PER | COUNT | FREQCORR | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | FREQCORR | 7:0 | SIGN | VALUE[6:0] | | | | | | |
| 0x15 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| 0x1A ... 0x1B | Reserved | | | | | | | | | |
| 0x1C | PER | 7:0 | PER[7:0] | | | | | | | |
| | | 15:8 | PER[15:8] | | | | | | | |
| 0x1E ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | COMP0 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |
| 0x22 | COMP1 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |

## 24.10. Register Description - Mode 1 - 16-Bit Counter

This Register Description section is valid if the RTC is in COUNT16 mode (CTRLA.MODE=1).

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.1. Control A in COUNT16 mode (CTRLA.MODE=1)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | | | | PRESCALER[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MODE[1:0] | | ENABLE | SWRST |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 15 – COUNTSYNC** COUNT Read Synchronization Enable
The COUNT register requires synchronization when reading. Disabling the synchronization will prevent reading valid values from the COUNT register.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | COUNT read synchronization is disabled |
| 1 | COUNT read synchronization is enabled |

**Bits 11:8 – PRESCALER[3:0]** Prescaler
These bits define the prescaling factor for the RTC clock source (GCLK_RTC) to generate the counter clock (CLK_RTC_CNT). Periodic events and interrupts are not available when the prescaler is off.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x1 | DIV1 | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x2 | DIV2 | CLK_RTC_CNT = GCLK_RTC/2 |
| 0x3 | DIV4 | CLK_RTC_CNT = GCLK_RTC/4 |
| 0x4 | DIV8 | CLK_RTC_CNT = GCLK_RTC/8 |
| 0x5 | DIV16 | CLK_RTC_CNT = GCLK_RTC/16 |
| 0x6 | DIV32 | CLK_RTC_CNT = GCLK_RTC/32 |
| 0x7 | DIV64 | CLK_RTC_CNT = GCLK_RTC/64 |
| 0x8 | DIV128 | CLK_RTC_CNT = GCLK_RTC/128 |
| 0x9 | DIV256 | CLK_RTC_CNT = GCLK_RTC/256 |
| 0xA | DIV512 | CLK_RTC_CNT = GCLK_RTC/512 |
| 0xB | DIV1024 | CLK_RTC_CNT = GCLK_RTC/1024 |
| 0xC-0xF | - | Reserved |

**Bits 3:2 – MODE[1:0]** Operating Mode
This field defines the operating mode of the RTC. This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT32 | Mode 0: 32-bit counter |
| 0x1 | COUNT16 | Mode 1: 16-bit counter |
| 0x2 | CLOCK | Mode 2: Clock/calendar |
| 0x3 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – ENABLE** Enable

Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled |
| 1 | The peripheral is enabled |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the RTC (except DBGCTRL) to their initial state, and the RTC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST will be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is not reset operation ongoing |
| 1 | The reset operation is ongoing |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.2. Event Control in COUNT16 mode (CTRLA.MODE=1)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | | | | | | CMPEO1 | CMPEO0 |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVFEO** Overflow Event Output Enable

| Value | Description |
|---|---|
| 0 | Overflow event is disabled and will not be generated. |
| 1 | Overflow event is enabled and will be generated for every overflow. |

**Bits 8, 9 – CMPEOn** Compare n Event Output Enable [n = 1..0]

| Value | Description |
|---|---|
| 0 | Compare n event is disabled and will not be generated. |
| 1 | Compare n event is enabled and will be generated for every compare match. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PEREOn** Periodic Interval n Event Output Enable [n = 7..0]

| Value | Description |
|---|---|
| 0 | Periodic Interval n event is disabled and will not be generated. |
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.3. Interrupt Enable Clear in COUNT16 mode (CTRLA.MODE=1)

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | CMP1 | CMP0 |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bits 8, 9 – CMPn** Compare n Interrupt Enable [n = 1..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Compare n Interrupt Enable bit, which disables the Compare n interrupt.

| Value | Description |
|---|---|
| 0 | The Compare n interrupt is disabled. |
| 1 | The Compare n interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Periodic Interval n Interrupt Enable bit, which disables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.4. Interrupt Enable Set in COUNT16 mode (CTRLA.MODE=1)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | CMP1 | CMP0 |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bits 8, 9 – CMPn** Compare n Interrupt Enable [n = 1..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Compare n Interrupt Enable bit, which and enables the Compare n interrupt.

| Value | Description |
|---|---|
| 0 | The Compare n interrupt is disabled. |
| 1 | The Compare n interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Periodic Interval n Interrupt Enable bit, which enables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.5. Interrupt Flag Status and Clear in COUNT16 mode (CTRLA.MODE=1)

**Name:** INTFLAG
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | CMP1 | CMP0 |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after an overflow condition occurs, and an interrupt request will be generated if INTENCLR/SET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

**Bits 8, 9 – CMPn** Compare n [n = 1..0]
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after a match with the compare condition, and an interrupt request will be generated if INTENCLR/SET.COMPn is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n [n = 7..0]
This flag is cleared by writing a '1' to the flag.
This flag is set on the 0-to-1 transition of prescaler bit [n+2], and an interrupt request will be generated if INTENCLR/SET.PERx is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Periodic Interval n interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.6. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The RTC is halted when the CPU is halted by an external debugger. |
| 1 | The RTC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.7. Synchronization Busy in COUNT16 mode (CTRLA.MODE=1)

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | | | | | | | |
| Access | R | | | | | | | |
| Reset | 0 | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | COMP1 | COMP0 | PER | COUNT | FREQCORR | ENABLE | SWRST |
| Access | | R/W | R/W | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – COUNTSYNC** Count Read Sync Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.COUNTSYNC bit is complete. |
| 1 | Write synchronization for CTRLA.COUNTSYNC bit is ongoing. |

**Bits 5, 6 – COMPn** Compare n Synchronization Busy Status [n = 1..0]

| Value | Description |
|---|---|
| 0 | Write synchronization for COMPn register is complete. |
| 1 | Write synchronization for COMPn register is ongoing. |

**Bit 4 – PER** Period Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for PER register is complete. |
| 1 | Write synchronization for PER register is ongoing. |

**Bit 3 – COUNT** Count Value Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Read/write synchronization for COUNT register is complete. |
| 1 | Read/write synchronization for COUNT register is ongoing. |

**Bit 2 – FREQCORR** Frequency Correction Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for FREQCORR register is complete. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Write synchronization for FREQCORR register is ongoing. |

**Bit 1 – ENABLE** Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.ENABLE bit is complete. |
| 1 | Write synchronization for CTRLA.ENABLE bit is ongoing. |

**Bit 0 – SWRST** Software Reset Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.8. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.FREQCORR must be checked to ensure the FREQCORR register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SIGN | VALUE[6:0] | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – SIGN** Correction Sign

| Value | Description |
|---|---|
| 0 | The correction value is positive, i.e., frequency will be decreased. |
| 1 | The correction value is negative, i.e., frequency will be increased. |

**Bits 6:0 – VALUE[6:0]** Correction Value
These bits define the amount of correction applied to the RTC prescaler.

| Value | Description |
|---|---|
| 0 | Correction is disabled and the RTC frequency is unchanged. |
| 1 - 127 | The RTC frequency is adjusted according to the value. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.9. Counter Value in COUNT16 mode (CTRLA.MODE = 1)

**Name:** COUNT
**Offset:** 0x18
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

**Notes:**

1. This register is read-synchronized and write-synchronized: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.
2. This register must be written with 16-bit accesses only.
3. Prior to any read access, this register must be synchronized by the user by writing the COUNT Read Synchronization Enable bit in the Control A register (CTRLA.COUNTSYNC = 1).

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – COUNT[15:0]** Counter Value
These bits define the value of the 16-bit RTC counter in COUNT16 mode (CTRLA.MODE = 1).


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.10. Counter Period in COUNT16 mode (CTRLA.MODE=1)

**Name:** PER
**Offset:** 0x1C
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.PER must be checked to ensure the PER register synchronization is complete.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PER[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – PER[15:0]** Counter Period
These bits define the value of the 16-bit RTC period in COUNT16 mode (CTRLA.MODE=1).


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.10.11. Compare n Value in COUNT16 mode (CTRLA.MODE=1)

**Name:** COMPn
**Offset:** 0x20 + n*0x02 [n=0..1]
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.COMPn must be checked to ensure the COMPn register synchronization is complete.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COMP[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COMP[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – COMP[15:0]** Compare Value

The 16-bit value of COMPn is continuously compared with the 16-bit COUNT value. When a match occurs, the Compare n interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMPn) is set on the next counter cycle.


© 2026 Microchip Technology Inc. and its subsidiaries

## 24.11. Register Summary - Mode 2 - Clock/Calendar

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | MATCHCLR | CLKREP | | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | CLOCKSYNC | | | | PRESCALER[3:0] | | | |
| 0x02 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | | | | | | | ALARMEO |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | ALARM0 |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | ALARM0 |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | | | | | | | ALARM0 |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | | ALARM0 | | CLOCK | FREQCORR | ENABLE | SWRST |
| | | 15:8 | CLOCKSYNC | | | | MASK0 | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | FREQCORR | 7:0 | SIGN | VALUE[6:0] | | | | | | |
| 0x15 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | CLOCK | 7:0 | MINUTE[1:0] | | SECOND[5:0] | | | | | |
| | | 15:8 | HOUR[3:0] | | | | MINUTE[5:2] | | | |
| | | 23:16 | MONTH[1:0] | | DAY[4:0] | | | | | HOUR[4] |
| | | 31:24 | YEAR[5:0] | | | | | | MONTH[3:2] | |
| 0x1C ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | ALARM | 7:0 | MINUTE[1:0] | | SECOND[5:0] | | | | | |
| | | 15:8 | HOUR[3:0] | | | | MINUTE[5:2] | | | |
| | | 23:16 | MONTH[1:0] | | DAY[4:0] | | | | | HOUR[4] |
| | | 31:24 | YEAR[5:0] | | | | | | MONTH[3:2] | |
| 0x24 | MASK | 7:0 | | | | | | SEL[2:0] | | |

## 24.12. Register Description - Mode 2 - Clock/Calendar

This Register Description section is valid if the RTC is in Clock/Calendar mode (CTRLA.MODE=2).

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.1. Control A in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CLOCKSYNC | | | | PRESCALER[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MATCHCLR | CLKREP | | | MODE[1:0] | | ENABLE | SWRST |
| Access | R/W | R/W | | | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | | | 0 | 0 | 0 | 0 |

**Bit 15 – CLOCKSYNC** CLOCK Read Synchronization Enable
The CLOCK register requires synchronization when reading. Disabling the synchronization will prevent reading valid values from the CLOCK register.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | CLOCK read synchronization is disabled |
| 1 | CLOCK read synchronization is enabled |

**Bits 11:8 – PRESCALER[3:0]** Prescaler
These bits define the prescaling factor for the RTC clock source (GCLK_RTC) to generate the counter clock (CLK_RTC_CNT). Periodic events and interrupts are not available when the prescaler is off.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x1 | DIV1 | CLK_RTC_CNT = GCLK_RTC/1 |
| 0x2 | DIV2 | CLK_RTC_CNT = GCLK_RTC/2 |
| 0x3 | DIV4 | CLK_RTC_CNT = GCLK_RTC/4 |
| 0x4 | DIV8 | CLK_RTC_CNT = GCLK_RTC/8 |
| 0x5 | DIV16 | CLK_RTC_CNT = GCLK_RTC/16 |
| 0x6 | DIV32 | CLK_RTC_CNT = GCLK_RTC/32 |
| 0x7 | DIV64 | CLK_RTC_CNT = GCLK_RTC/64 |
| 0x8 | DIV128 | CLK_RTC_CNT = GCLK_RTC/128 |
| 0x9 | DIV256 | CLK_RTC_CNT = GCLK_RTC/256 |
| 0xA | DIV512 | CLK_RTC_CNT = GCLK_RTC/512 |
| 0xB | DIV1024 | CLK_RTC_CNT = GCLK_RTC/1024 |
| 0xC-0xF | - | Reserved |

**Bit 7 – MATCHCLR** Clear on Match
This bit is valid only in Mode 0 (COUNT32) and Mode 2 (CLOCK). This bit can be written only when the peripheral is disabled. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The counter is not cleared on a Compare/Alarm 0 match |
| 1 | The counter is cleared on a Compare/Alarm 0 match |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 6 – CLKREP** Clock Representation

This bit is valid only in Mode 2 and determines how the hours are represented in the Clock Value (CLOCK) register. This bit can be written only when the peripheral is disabled. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | 24 Hour |
| 1 | 12 Hour (AM/PM) |

**Bits 3:2 – MODE[1:0]** Operating Mode

This field defines the operating mode of the RTC. This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT32 | Mode 0: 32-bit counter |
| 0x1 | COUNT16 | Mode 1: 16-bit counter |
| 0x2 | CLOCK | Mode 2: Clock/calendar |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** Enable

Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled |
| 1 | The peripheral is enabled |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the RTC, except DBGCTRL, to their initial state, and the RTC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST will be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is not reset operation ongoing |
| 1 | The reset operation is ongoing |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.2. Event Control in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | | | | | | | ALARMEO |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVFEO** Overflow Event Output Enable

| Value | Description |
|---|---|
| 0 | Overflow event is disabled and will not be generated. |
| 1 | Overflow event is enabled and will be generated for every overflow. |

**Bit 8 – ALARMEO** Alarm 0 Event Output Enable

| Value | Description |
|---|---|
| 0 | Alarm 0 event is disabled and will not be generated. |
| 1 | Alarm 0 event is enabled and will be generated for every compare match. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PEREOn** Periodic Interval n Event Output Enable [n = 7..0]

| Value | Description |
|---|---|
| 0 | Periodic Interval n event is disabled and will not be generated. |
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.3. Interrupt Enable Clear in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | ALARM0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bit 8 – ALARM0** Alarm 0 Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Alarm 0 Interrupt Enable bit, which disables the Alarm interrupt.

| Value | Description |
|---|---|
| 0 | The Alarm 0 interrupt is disabled. |
| 1 | The Alarm 0 interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Periodic Interval n Interrupt Enable bit, which disables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.4. Interrupt Enable Set in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | ALARM0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bit 8 – ALARM0** Alarm 0 Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Alarm 0 Interrupt Enable bit, which enables the Alarm 0 interrupt.

| Value | Description |
|---|---|
| 0 | The Alarm 0 interrupt is disabled. |
| 1 | The Alarm 0 interrupt is enabled. |

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n Interrupt Enable [n = 7..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Periodic Interval n Interrupt Enable bit, which enables the Periodic Interval n interrupt.

| Value | Description |
|---|---|
| 0 | Periodic Interval n interrupt is disabled. |
| 1 | Periodic Interval n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.5. Interrupt Flag Status and Clear in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** INTFLAG
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | | | | | | | ALARM0 |
| Access | R/W | | | | | | | R/W |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – OVF** Overflow
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after an overflow condition occurs, and an interrupt request will be generated if INTENCLR/SET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.

**Bit 8 – ALARM0** Alarm 0
This flag is cleared by writing a '1' to the flag.
This flag is set on the next CLK_RTC_CNT cycle after a match with the compare condition, and an interrupt request will be generated if INTENCLR/SET.ALARM0 is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Alarm 0 interrupt flag.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PERn** Periodic Interval n [n = 7..0]
This flag is cleared by writing a '1' to the flag.
This flag is set on the 0-to-1 transition of prescaler bit [n+2], and an interrupt request will be generated if INTENCLR/SET.PERx is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Periodic Interval n interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.6. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The RTC is halted when the CPU is halted by an external debugger. |
| 1 | The RTC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.7. Synchronization Busy in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CLOCKSYNC | | | | MASK0 | | | |
| Access | R | | | | R | | | |
| Reset | 0 | | | | 0 | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | ALARM0 | | CLOCK | FREQCORR | ENABLE | SWRST |
| Access | | | R | | R | R | R | R |
| Reset | | | 0 | | 0 | 0 | 0 | 0 |

**Bit 15 – CLOCKSYNC** Clock Read Sync Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.CLOCKSYNC bit is complete. |
| 1 | Write synchronization for CTRLA.CLOCKSYNC bit is ongoing. |

**Bit 11 – MASK0** Mask 0 Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for MASK0 register is complete. |
| 1 | Write synchronization for MASK0 register is ongoing. |

**Bit 5 – ALARM0** Alarm 0 Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for ALARM0 register is complete. |
| 1 | Write synchronization for ALARM0 register is ongoing. |

**Bit 3 – CLOCK** Clock Register Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Read/write synchronization for CLOCK register is complete. |
| 1 | Read/write synchronization for CLOCK register is ongoing. |

**Bit 2 – FREQCORR** Frequency Correction Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for FREQCORR register is complete. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Write synchronization for FREQCORR register is ongoing. |

**Bit 1 – ENABLE** Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.ENABLE bit is complete. |
| 1 | Write synchronization for CTRLA.ENABLE bit is ongoing. |

**Bit 0 – SWRST** Software Reset Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.8. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.FREQCORR must be checked to ensure the FREQCORR register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SIGN | VALUE[6:0] | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – SIGN** Correction Sign

| Value | Description |
|---|---|
| 0 | The correction value is positive, i.e., frequency will be decreased. |
| 1 | The correction value is negative, i.e., frequency will be increased. |

**Bits 6:0 – VALUE[6:0]** Correction Value
These bits define the amount of correction applied to the RTC prescaler.

| Value | Description |
|---|---|
| 0 | Correction is disabled and the RTC frequency is unchanged. |
| 1 - 127 | The RTC frequency is adjusted according to the value. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.9. Clock Value in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** CLOCK
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

**Notes:**

1. This register is read-synchronized and write-synchronized: SYNCBUSY.CLOCK must be checked to ensure the CLOCK register synchronization is complete.
2. This register must be written with 32-bit accesses only.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | YEAR[5:0] | | | | | | MONTH[3:2] | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | MONTH[1:0] | | DAY[4:0] | | | | | HOUR[4] |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | HOUR[3:0] | | | | MINUTE[5:2] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MINUTE[1:0] | | SECOND[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:26 – YEAR[5:0]** Year
The year offset with respect to the reference year (defined in software).
The year is considered a leap year if YEAR[1:0] is zero.

**Bits 25:22 – MONTH[3:0]** Month
1 – January
2 – February
...
12 – December

**Bits 21:17 – DAY[4:0]** Day
Day starts at 1 and ends at 28, 29, 30, or 31, depending on the month and year.

**Bits 16:12 – HOUR[4:0]** Hour
When CTRLA.CLKREP=0, the Hour bit group is in 24-hour format, with values 0-23. When CTRLA.CLKREP=1, HOUR[3:0] has values 1-12, and HOUR[4] represents AM (0) or PM (1).

**Bits 11:6 – MINUTE[5:0]** Minute
0 – 59

**Bits 5:0 – SECOND[5:0]** Second
0 – 59


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.10. Alarm Value in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** ALARM
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

The 32-bit value of ALARM is continuously compared with the 32-bit CLOCK value, based on the masking set by MASK.SEL. When a match occurs, the Alarm n interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.ALARM) is set on the next counter cycle, and the counter is cleared if CTRLA.MATCHCLR is '1'.
**Note:** This register is write-synchronized: SYNCBUSY.ALARM0 must be checked to ensure the ALARM register synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | YEAR[5:0] | | | | | | MONTH[3:2] | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | MONTH[1:0] | | DAY[4:0] | | | | | HOUR[4] |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | HOUR[3:0] | | | | MINUTE[5:2] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MINUTE[1:0] | | SECOND[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:26 – YEAR[5:0]** Year
The alarm year. Years are only matched if MASK.SEL is 6

**Bits 25:22 – MONTH[3:0]** Month
The alarm month. Months are matched only if MASK.SEL is greater than 4.

**Bits 21:17 – DAY[4:0]** Day
The alarm day. Days are matched only if MASK.SEL is greater than 3.

**Bits 16:12 – HOUR[4:0]** Hour
The alarm hour. Hours are matched only if MASK.SEL is greater than 2.

**Bits 11:6 – MINUTE[5:0]** Minute
The alarm minute. Minutes are matched only if MASK.SEL is greater than 1.

**Bits 5:0 – SECOND[5:0]** Second
The alarm second. Seconds are matched only if MASK.SEL is greater than 0.


© 2026 Microchip Technology Inc. and its subsidiaries

### 24.12.11. Alarm Mask in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** MASK
**Offset:** 0x24
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.MASK0 must be checked to ensure the MASK register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SEL[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – SEL[2:0]** Alarm Mask Selection
These bits define which bit groups of ALARM are valid.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Alarm Disabled |
| 0x1 | SS | Match seconds only |
| 0x2 | MMSS | Match seconds and minutes only |
| 0x3 | HHMMSS | Match seconds, minutes, and hours only |
| 0x4 | DDHHMMSS | Match seconds, minutes, hours, and days only |
| 0x5 | MMDDHHMMSS | Match seconds, minutes, hours, days, and months only |
| 0x6 | YYMMDDHHMMSS | Match seconds, minutes, hours, days, months, and years |
| 0x7 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

# 25. DMAC – Direct Memory Access Controller

## 25.1. Overview

The Direct Memory Access Controller (DMAC) contains both a Direct Memory Access (DMA) engine and a Cyclic Redundancy Check (CRC) engine. The DMAC can transfer data between memories and peripherals and therefore, off-load these tasks from the CPU. It enables high data transfer rates with minimum CPU intervention, and frees up CPU time. With access to all peripherals, the DMAC can handle automatic transfer of data between communication modules.

The DMA part of the DMAC has several DMA channels, which can receive different types of transfer triggers and generate transfer requests from the DMA channels to the arbiter (Refer to the Block Diagram). The arbiter will select one DMA channel at a time to act as the active channel. When an active channel has been selected, the fetch engine of the DMAC will fetch a transfer descriptor from the SRAM and store it in the internal memory of the active channel, which will then execute the data transmission.

An ongoing data transfer of an active channel can be interrupted by a higher prioritized DMA channel. The DMAC will write back the updated transfer descriptor from the internal memory of the active channel to SRAM, and grant the higher prioritized channel a start transfer as the new active channel. Once a DMA channel is done with its transfer, interrupts and events can be generated optionally.

The DMAC has four bus interfaces:

- The *data transfer bus* is used for performing the actual DMA transfer.
- The *AHB/APB Bridge bus* is used when writing and reading the I/O registers of the DMAC.
- The *descriptor fetch bus* is used by the fetch engine to fetch transfer descriptors before data transfer can be started or continued.
- The *write-back bus* is used to write the transfer descriptor back to SRAM.

All buses are AHB host interfaces except the AHB/APB Bridge bus, which is an APB client interface.

The CRC engine can be used by software to detect an accidental error in the transferred data and to take corrective action, such as requesting the data to be sent again or simply not using the incorrect data.

## 25.2. Features

- Data transfer from:
  - Peripheral to peripheral
  - Peripheral to memory
  - Memory to peripheral
  - Memory to memory
- Transfer trigger sources
  - Software
  - Events from Event System
  - Dedicated requests from peripherals
- SRAM based transfer descriptors
  - Single transfer using one descriptor
  - Multi-buffer or circular buffer modes by linking multiple descriptors
- Up to 12 channels
  - Enable 12 independent transfers

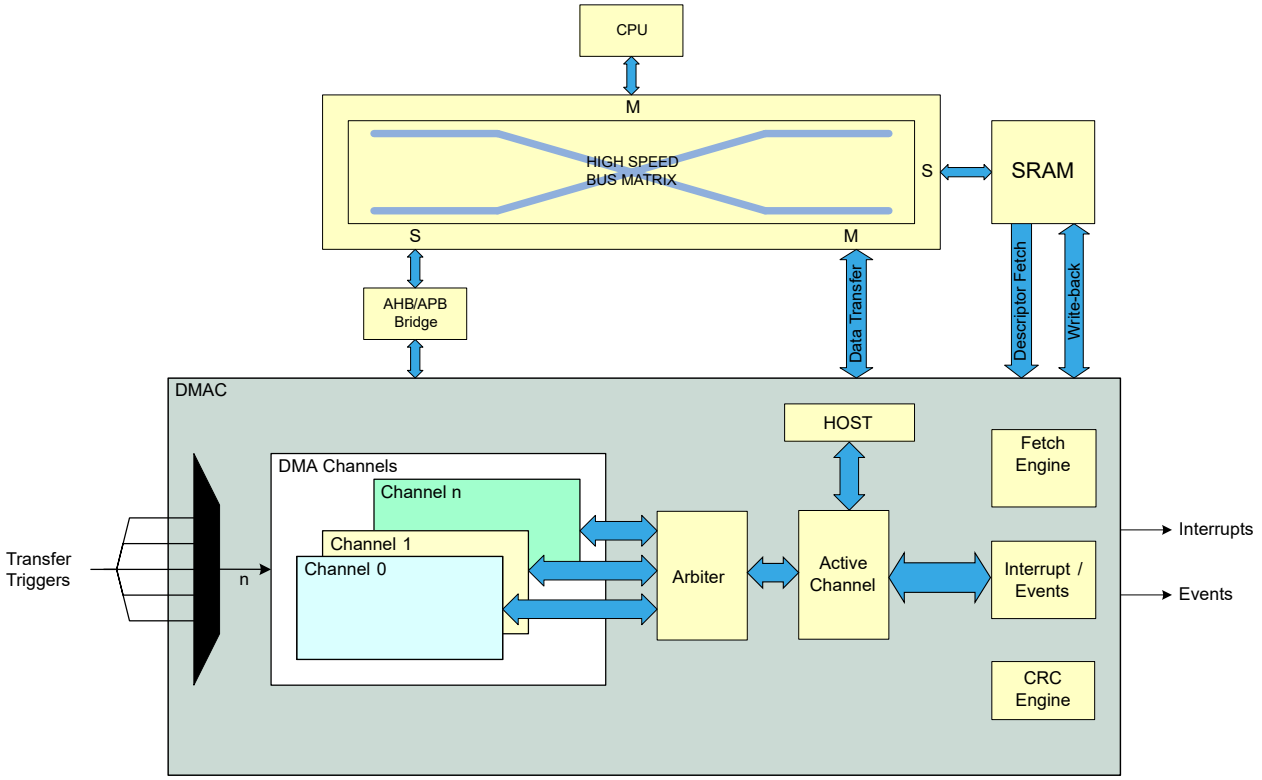

© 2026 Microchip Technology Inc. and its subsidiaries

  - Automatic descriptor fetch for each channel
  - Suspend/resume operation support for each channel
- Flexible arbitration scheme
  - 4 configurable priority levels for each channel
  - Fixed or round-robin priority scheme within each priority level
- From 1 to 256KB data transfer in a single block transfer
- Multiple addressing modes
  - Static
  - Configurable increment scheme
- Optional interrupt generation
  - On block transfer complete
  - On error detection
  - On channel suspend
- 4 event inputs
  - One event input for each of the 4 least significant DMA channels
  - Can be selected to trigger normal transfers, periodic transfers or conditional transfers
  - Can be selected to suspend or resume channel operation
- 4 event outputs
  - One output event for each of the 4 least significant DMA channels
  - Selectable generation on AHB, block, or transaction transfer complete
- Error management supported by write-back function
  - Dedicated Write-Back memory section for each channel to store ongoing descriptor transfer
- CRC polynomial software selectable to
  - CRC-16 (CRC-CCITT)
  - CRC-32 (IEEE® 802.3)


© 2026 Microchip Technology Inc. and its subsidiaries

## 25.3. Block Diagram

**Figure 25-1.** DMAC Block Diagram




## 25.4. Signal Description

Not applicable.

## 25.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 25.5.1. I/O Lines

Not applicable.

### 25.5.2. Power Management

The DMAC will continue to operate in any sleep mode where the selected source clock is running. The DMAC's interrupts can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes. On hardware or software reset, all registers are set to their reset value.

**Related Links**

PM - Power Manager

### 25.5.3. Clocks

The DMAC bus clock (CLK_DMAC_AHB) must be configured and enabled in the Main Clock module before using the DMAC.

This bus clock (CLK_DMAC_AHB) is always synchronous to the module clock (CLK_DMAC_AHB), but can be divided by a prescaler and may run even when the module clock is turned off.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Peripheral Clock Masking

### 25.5.4. DMA
Not applicable.

### 25.5.5. Interrupts
The interrupt request line is connected to the interrupt controller. Using the DMAC interrupt requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 25.5.6. Events
The events are connected to the event system.

**Related Links**
Event System (EVSYS)

### 25.5.7. Debug Operation
When the CPU is halted in debug mode the DMAC will halt normal operation. The DMAC can be forced to continue operation during debugging. Refer to DBGCTRL for details.

### 25.5.8. Register Access Protection
All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Pending register (INTPEND)
- Channel ID register (CHID)
- Channel Interrupt Flag Status and Clear register (CHINTFLAG)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 25.5.9. Analog Connections
Not applicable.

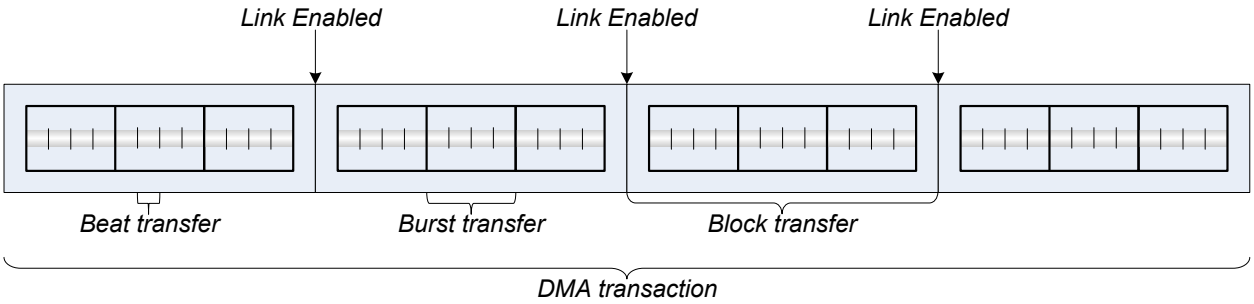
## 25.6. Functional Description

### 25.6.1. Principle of Operation
The DMAC consists of a DMA module and a CRC module.

#### 25.6.1.1. DMA
The DMAC can transfer data between memories and peripherals without interaction from the CPU. The data transferred by the DMAC are called transactions, and these transactions can be split into smaller data transfers. The following figure shows the relationship between the different transfer sizes:


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-2.** DMA Transfer Sizes




- Beat transfer: The size of one data transfer bus access, and the size is selected by writing the Beat Size bit group in the Block Transfer Control register (BTCTRL.BEATSIZE)
- Block transfer: The amount of data one transfer descriptor can transfer, and the amount can range from 1 to 64k beats. A block transfer can be interrupted.
- Transaction: The DMAC can link several transfer descriptors by having the first descriptor pointing to the second and so forth, as shown in the figure above. A DMA transaction is the complete transfer of all blocks within a linked list.

A transfer descriptor describes how a block transfer should be carried out by the DMAC, and it must remain in SRAM. For further details on the transfer descriptor refer to Transfer Descriptors.

The figure above shows several block transfers linked together, which are called linked descriptors. For further information about linked descriptors, refer to Linked Descriptors.

A DMA transfer is initiated by an incoming transfer trigger on one of the DMA channels. This trigger can be configured to be either a software trigger, an event trigger, or one of the dedicated peripheral triggers. The transfer trigger will result in a DMA transfer request from the specific channel to the arbiter. If there are several DMA channels with pending transfer requests, the arbiter chooses which channel is granted access to become the active channel. The DMA channel granted access as the active channel will carry out the transaction as configured in the transfer descriptor. A current transaction can be interrupted by a higher prioritized channel, but will resume the block transfer when the according DMA channel is granted access as the active channel again.

For each beat transfer, an optional output event can be generated. For each block transfer, optional interrupts and an optional output event can be generated. When a transaction is completed, depending on the configuration, the DMA channel will either be suspended or disabled.

#### 25.6.1.2. CRC

The internal CRC engine supports two commonly used CRC polynomials: CRC-16 (CRC-CCITT) and CRC-32 (IEEE 802.3). It can be used on a selectable DMA channel, or on the I/O interface. Refer to CRC Operation for details.

### 25.6.2. Basic Operation

#### 25.6.2.1. Initialization

The following DMAC registers are enable-protected, meaning that they can only be written when the DMAC is disabled (CTRL.DMAENABLE=0):

- Descriptor Base Memory Address register (BASEADDR)
- Write-Back Memory Base Address register (WRBADDR)

The following DMAC bit is enable-protected, meaning that it can only be written when both the DMAC and CRC are disabled (CTRL.DMAENABLE=0 and CTRL.CRCENABLE=0):

- Software Reset bit in Control register (CTRL.SWRST)


© 2026 Microchip Technology Inc. and its subsidiaries

The following DMA channel register is enable-protected, meaning that it can only be written when the corresponding DMA channel is disabled (CHCTRLA.ENABLE=0):

- Channel Control B (CHCTRLB) register, except the Command bit (CHCTRLB.CMD) and the Channel Arbitration Level bit (CHCTRLB.LVL)

The following DMA channel bit is enable-protected, meaning that it can only be written when the corresponding DMA channel is disabled:

- Channel Software Reset bit in Channel Control A register (CHCTRLA.SWRST)

The following CRC registers are enable-protected, meaning that they can only be written when the CRC is disabled (CTRL.CRCENABLE=0):

- CRC Control register (CRCCTRL)
- CRC Checksum register (CRCCHKSUM)

Enable-protection is denoted by the "Enable-Protected" property in the register description.

Before the DMAC is enabled it must be configured, as outlined by the following steps:

- The SRAM address of where the descriptor memory section is located must be written to the Description Base Address (BASEADDR) register
- The SRAM address of where the write-back section should be located must be written to the Write-Back Memory Base Address (WRBADDR) register
- Priority level x of the arbiter can be enabled by setting the Priority Level x Enable bit in the Control register (CTRL.LVLENx=1)

Before a DMA channel is enabled, the DMA channel and the corresponding first transfer descriptor must be configured, as outlined by the following steps:

- DMA channel configurations
  - The channel number of the DMA channel to configure must be written to the Channel ID (CHID) register
  - Trigger action must be selected by writing the Trigger Action bit group in the Channel Control B register (CHCTRLB.TRIGACT)
  - Trigger source must be selected by writing the Trigger Source bit group in the Channel Control B register (CHCTRLB.TRIGSRC)
- Transfer Descriptor
  - The size of each access of the data transfer bus must be selected by writing the Beat Size bit group in the Block Transfer Control register (BTCTRL.BEATSIZE)
  - The transfer descriptor must be made valid by writing a one to the Valid bit in the Block Transfer Control register (BTCTRL.VALID)
  - Number of beats in the block transfer must be selected by writing the Block Transfer Count (BTCNT) register
  - Source address for the block transfer must be selected by writing the Block Transfer Source Address (SRCADDR) register
  - Destination address for the block transfer must be selected by writing the Block Transfer Destination Address (DSTADDR) register

If CRC calculation is needed, the CRC engine must be configured before it is enabled, as outlined by the following steps:

- The CRC input source must selected by writing the CRC Input Source bit group in the CRC Control register (CRCCTRL.CRCSRC)
- The type of CRC calculation must be selected by writing the CRC Polynomial Type bit group in the CRC Control register (CRCCTRL.CRCPOLY)


© 2026 Microchip Technology Inc. and its subsidiaries

- If I/O is selected as input source, the beat size must be selected by writing the CRC Beat Size bit group in the CRC Control register (CRCCTRL.CRCBEATSIZE)

**Related Links**

BASEADDR
Descriptor Memory Section Base Address

CHCTRLA
Channel Control A

CHCTRLB
Channel Control B

CRCCHKSUM
CRC Checksum

CRCCTRL
CRC Control

CTRL
Control

WRBADDR
Write-Back Memory Section Base Address

BTCTRL
Block Transfer Control

BTCNT
Block Transfer Count

DSTADDR
Block Transfer Destination Address

SRCADDR
Block Transfer Source Address

### 25.6.2.2. Enabling, Disabling, and Resetting

The DMAC is enabled by writing the DMA Enable bit in the Control register (CTRL.DMAENABLE) to '1'. The DMAC is disabled by writing a '0' to CTRL.DMAENABLE.

A DMA channel is enabled by writing the Enable bit in the Channel Control A register (CHCTRLA.ENABLE) to '1', after writing the corresponding channel id to the Channel ID bit group in the Channel ID register (CHID.ID). A DMA channel is disabled by writing a '0' to CHCTRLA.ENABLE.

The CRC is enabled by writing a '1' to the CRC Enable bit in the Control register (CTRL.CRCENABLE). The CRC is disabled by writing a '0' to CTRL.CRCENABLE.

The DMAC is reset by writing a '1' to the Software Reset bit in the Control register (CTRL.SWRST) while the DMAC and CRC are disabled. All registers in the DMAC except DBGCTRL will be reset to their initial state.

A DMA channel is reset by writing a '1' to the Software Reset bit in the Channel Control A register (CHCTRLA.SWRST), after writing the corresponding channel id to the Channel ID bit group in the Channel ID register (CHID.ID). The channel registers will be reset to their initial state. The corresponding DMA channel must be disabled in order for the reset to take effect.

### 25.6.2.3. Transfer Descriptors

Together with the channel configurations the transfer descriptors decides how a block transfer should be executed. Before a DMA channel is enabled (CHCTRLA.ENABLE is written to one), and receives a transfer trigger, its first transfer descriptor has to be initialized and valid (BTCTRL.VALID). The first transfer descriptor describes the first block transfer of a transaction.

All transfer descriptors must reside in SRAM. The addresses stored in the Descriptor Memory Section Base Address (BASEADDR) and Write-Back Memory Section Base Address (WRBADDR)

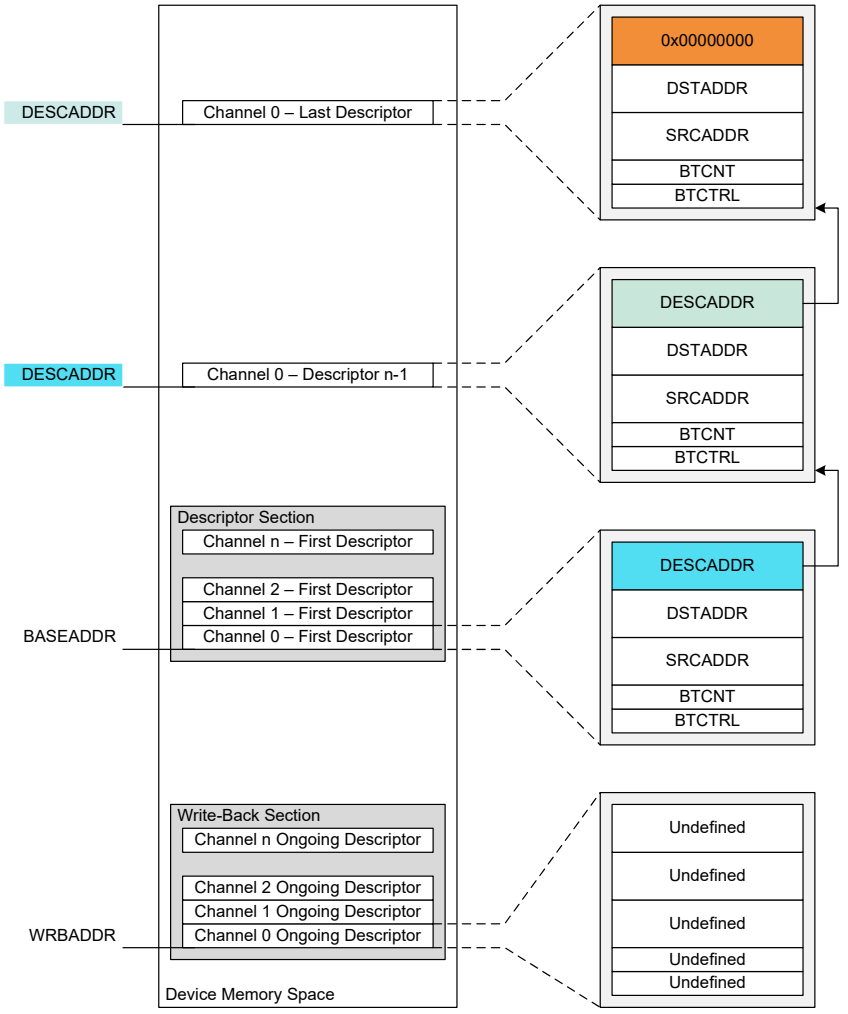

© 2026 Microchip Technology Inc. and its subsidiaries

registers tell the DMAC where to find the descriptor memory section and the write-back memory section.

The descriptor memory section is where the DMAC expects to find the first transfer descriptors for all DMA channels. As BASEADDR points only to the first transfer descriptor of channel 0 (see figure below), all first transfer descriptors must be stored in a contiguous memory section, where the transfer descriptors must be ordered according to their channel number. For further details on linked descriptors, refer to Linked Descriptors.

The write-back memory section is the section where the DMAC stores the transfer descriptors for the ongoing block transfers. WRBADDR points to the ongoing transfer descriptor of channel 0. All ongoing transfer descriptors will be stored in a contiguous memory section where the transfer descriptors are ordered according to their channel number. The figure below shows an example of linked descriptors on DMA channel 0. For further details on linked descriptors, refer to Linked Descriptors.

**Figure 25-3.** Memory Sections




The size of the descriptor and write-back memory sections is dependent on the number of the most significant enabled DMA channel *m*, as shown below:

$$Size = 128\text{bits} \cdot (m + 1)$$

For memory optimization, it is recommended to always use the less significant DMA channels if not all channels are required.


© 2026 Microchip Technology Inc. and its subsidiaries

The descriptor and write-back memory sections can either be two separate memory sections, or they can share memory section (BASEADDR=WRBADDR). The benefit of having them in two separate sections, is that the same transaction for a channel can be repeated without having to modify the first transfer descriptor. The benefit of having descriptor memory and write-back memory in the same section is that it requires less SRAM. In addition, the latency from fetching the first descriptor of a transaction to the first burst transfer is executed, is reduced.

### 25.6.2.4. Arbitration

If a DMA channel is enabled and not suspended when it receives a transfer trigger, it will send a transfer request to the arbiter. When the arbiter receives the transfer request it will include the DMA channel in the queue of channels having pending transfers, and the corresponding Pending Channel x bit in the Pending Channels registers (PENDCH.PENDCHx) will be set. Depending on the arbitration scheme, the arbiter will choose which DMA channel will be the next active channel. The active channel is the DMA channel being granted access to perform its next burst transfer. When the arbiter has granted a DMA channel access to the DMAC, the corresponding bit PENDCH.PENDCHx will be cleared. See also the following figure.

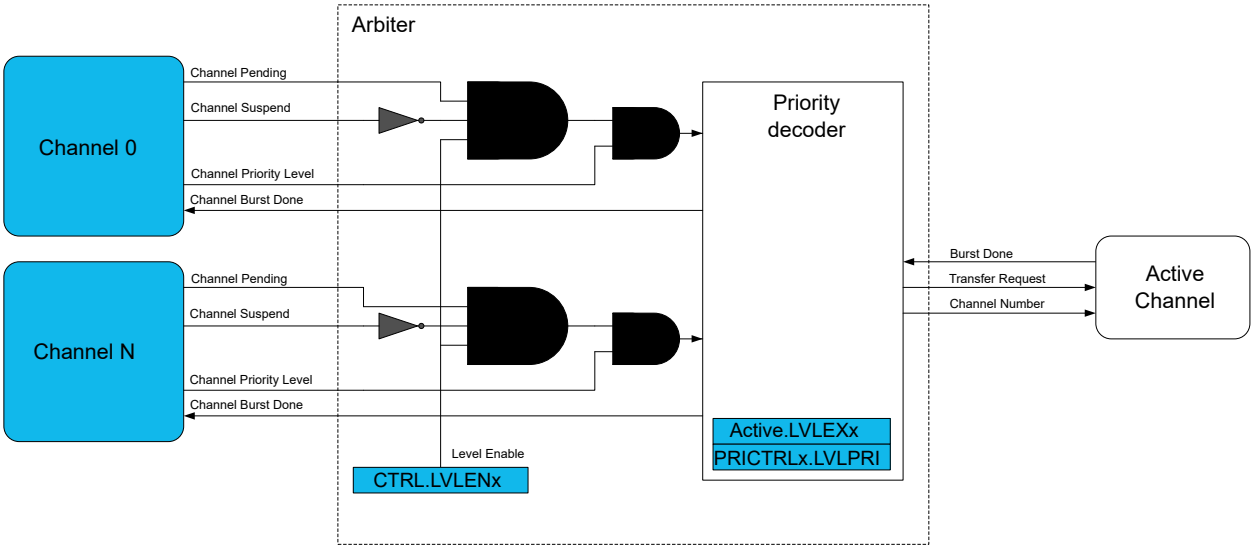
If the upcoming burst transfer is the first for the transfer request, the corresponding Busy Channel x bit in the Busy Channels register will be set (BUSYCH.BUSYCHx=1), and it will remain '1' for the subsequent granted burst transfers.

When the channel has performed its granted burst transfer(s) it will be either fed into the queue of channels with pending transfers, set to be waiting for a new transfer trigger, suspended, or disabled. This depends on the channel and block transfer configuration. If the DMA channel is fed into the queue of channels with pending transfers, the corresponding BUSYCH.BUSYCHx will remain '1'. If the DMA channel is set to wait for a new transfer trigger, suspended, or disabled, the corresponding BUSYCH.BUSYCHx will be cleared.

If a DMA channel is suspended while it has a pending transfer, it will be removed from the queue of pending channels, but the corresponding PENDCH.PENDCHx will remain set. When the same DMA channel is resumed, it will be added to the queue of pending channels again.

If a DMA channel gets disabled (CHCTRLA.ENABLE=0) while it has a pending transfer, it will be removed from the queue of pending channels, and the corresponding PENDCH.PENDCHx will be cleared.

**Figure 25-4.** Arbiter Overview




#### Priority Levels


© 2026 Microchip Technology Inc. and its subsidiaries

When a channel level is pending or the channel is transferring data, the corresponding Level Executing bit is set in the Active Channel and Levels register (ACTIVE.LVLEXx).

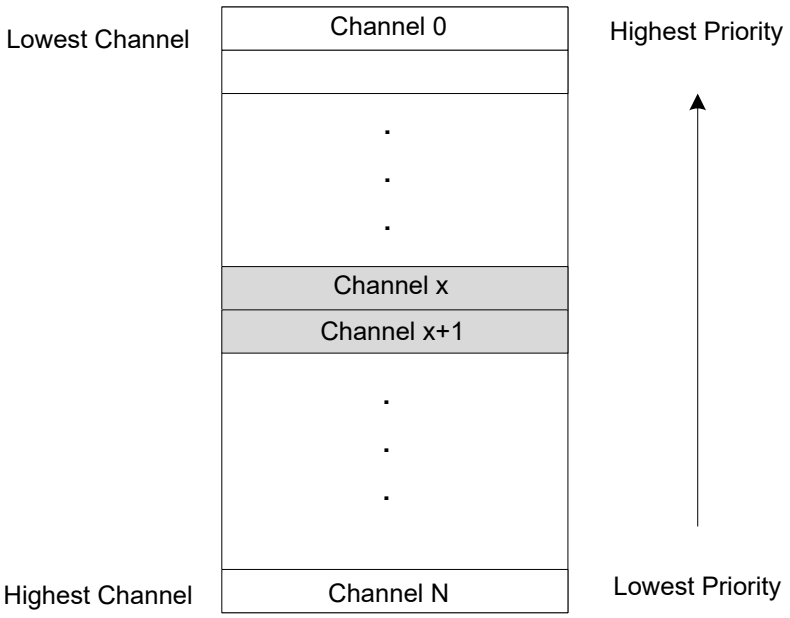
Each DMA channel supports a 4-level priority scheme. The priority level for a channel is configured by writing to the Channel Arbitration Level bit group in the Channel Control B register (CHCTRLB.LVL). As long as all priority levels are enabled, a channel with a higher priority level number will have priority over a channel with a lower priority level number. Each priority level x is enabled by setting the corresponding Priority Level x Enable bit in the Control register (CTRL.LVLENx=1).

Within each priority level the DMAC's arbiter can be configured to prioritize statically or dynamically:

*Static Arbitration* within a priority level is selected by writing a '0' to the Level x Round-Robin Scheduling Enable bit in the Priority Control 0 register (PRICTRL0.RRLVLENx).

When static arbitration is selected, the arbiter will prioritize a low channel number over a high channel number as shown in the figure below. When using the static arbitration there is a risk of high channel numbers never being granted access as the active channel. This can be avoided using a dynamic arbitration scheme.

**Figure 25-5.** Static Priority Scheduling

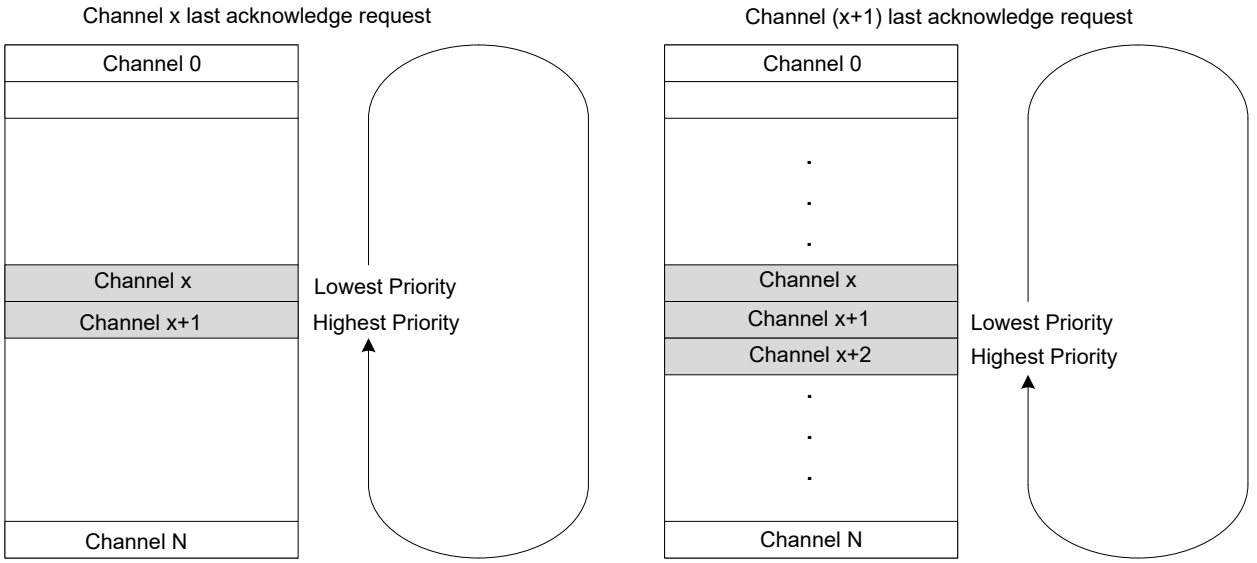



*Dynamic Arbitration* within a priority level is selected by writing a '1' to PRICTRL0.RRLVLENx.

The dynamic arbitration scheme in the DMAC is round-robin. With the round-robin scheme, the channel number of the last channel being granted access will have the lowest priority the next time the arbiter has to grant access to a channel within the same priority level, as shown in Figure 25-6. The channel number of the last channel being granted access as the active channel is stored in the Level x Channel Priority Number bit group in the Priority Control 0 register (PRICTRL0.LVLPRIx) for the corresponding priority level.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-6.** Dynamic (Round-Robin) Priority Scheduling




### 25.6.2.5. Data Transmission

Before the DMAC can perform a data transmission, a DMA channel has to be configured and enabled, its corresponding transfer descriptor has to be initialized, and the arbiter has to grant the DMA channel access as the active channel.

Once the arbiter has granted a DMA channel access as the active channel (refer to DMA Block Diagram section) the transfer descriptor for the DMA channel will be fetched from SRAM using the fetch bus, and stored in the internal memory for the active channel. For a new block transfer, the transfer descriptor will be fetched from the descriptor memory section (BASEADDR); For an ongoing block transfer, the descriptor will be fetched from the write-back memory section (WRBADDR). By using the data transfer bus, the DMAC will read the data from the current source address and write it to the current destination address. For further details on how the current source and destination addresses are calculated, refer to the section on Addressing.

The arbitration procedure is performed after each burst transfer. If the current DMA channel is granted access again, the block transfer counter (BTCNT) of the internal transfer descriptor will be decremented by the number of beats in a burst transfer, the optional output event Beat will be generated if configured and enabled, and the active channel will perform a new burst transfer. If a different DMA channel than the current active channel is granted access, the block transfer counter value will be written to the write-back section before the transfer descriptor of the newly granted DMA channel is fetched into the internal memory of the active channel.

When a block transfer has come to its end (BTCNT is zero), the Valid bit in the Block Transfer Control register will be cleared (BTCTRL.VALID=0) before the entire transfer descriptor is written to the write-back memory. The optional interrupts, Channel Transfer Complete and Channel Suspend, and the optional output event Block, will be generated if configured and enabled. After the last block transfer in a transaction, the Next Descriptor Address register (DESCADDR) will hold the value 0x00000000, and the DMA channel will either be suspended or disabled, depending on the configuration in the Block Action bit group in the Block Transfer Control register (BTCTRL.BLOCKACT). If the transaction has further block transfers pending, DESCADDR will hold the SRAM address to the next transfer descriptor to be fetched. The DMAC will fetch the next descriptor into the internal memory of the active channel and write its content to the write-back section for the channel, before the arbiter gets to choose the next active channel.


© 2026 Microchip Technology Inc. and its subsidiaries

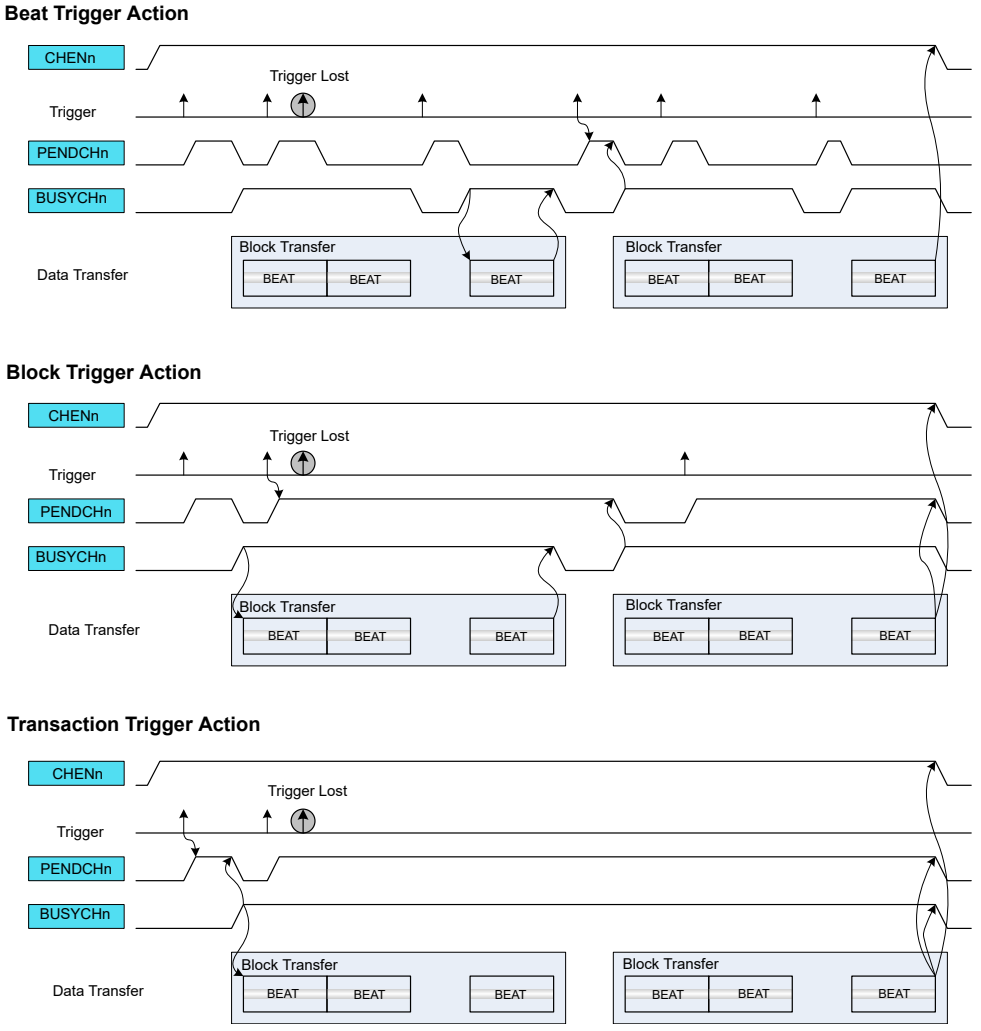
### 25.6.2.6. Transfer Triggers and Actions

A DMA transfer through a DMA channel can be started only when a DMA transfer request is detected, and the DMA channel has been granted access to the DMA. A transfer request can be triggered from software, from a peripheral, or from an event. There are dedicated Trigger Source selections for each DMA Channel Control B (CHCTRLB.TRIGSRC).

The trigger actions are available in the Trigger Action bit group in the Channel Control B register (CHCTRLB.TRIGACT). By default, a trigger generates a request for a block transfer operation. If a single descriptor is defined for a channel, the channel is automatically disabled when a block transfer has been completed. If a list of linked descriptors is defined for a channel, the channel is automatically disabled when the last descriptor in the list is executed. If the list still has descriptors to execute, the channel will be waiting for the next block transfer trigger. When enabled again, the channel will wait for the next block transfer trigger. The trigger actions can also be configured to generate a request for a beat transfer (CHCTRLB.TRIGACT=0x2) or transaction transfer (CHCTRLB.TRIGACT=0x3) instead of a block transfer (CHCTRLB.TRIGACT=0x0).

Figure 25-7 shows an example where triggers are used with two linked block descriptors.

**Figure 25-7.** Trigger Action and Transfers




If the trigger source generates a transfer request for a channel during an ongoing transfer, the new transfer request will be kept pending (CHSTATUS.PEND=1), and the new transfer can start after the ongoing one is done. Only one pending transfer can be kept per channel. If the trigger source


© 2026 Microchip Technology Inc. and its subsidiaries

generates more transfer requests while one is already pending, the additional ones will be lost. All channels pending status flags are also available in the Pending Channels register (PENDCH).

When the transfer starts, the corresponding Channel Busy status flag is set in Channel Status register (CHSTATUS.BUSY). When the trigger action is complete, the Channel Busy status flag is cleared. All channel busy status flags are also available in the Busy Channels register (BUSYCH) in DMAC.

#### 25.6.2.7. Addressing

Each block transfer must have a source address and a destination address defined. The source address is set by writing the Transfer Source Address (SRCADDR) register, the destination address is set by writing the Transfer Destination Address (DSTADDR) register.

The addressing of this DMAC module can be static or incremental, for either source or destination of a block transfer, or both.

Incrementation for the source address of a block transfer is enabled by writing the Source Address Incrementation Enable bit in the Block Transfer Control register (BTCTRL.SRCINC = 1). The step size of the incrementation is configurable and can be chosen by writing the Step Selection bit in the Block Transfer Control register (BTCTRL.STEPSEL = 1) and writing the desired step size in the Address Increment Step Size bit group in the Block Transfer Control register (BTCTRL.STEPSIZE). If BTCTRL.STEPSEL = 0, the step size for the source incrementation will be the size of one beat.

When source address incrementation is configured (BTCTRL.SRCINC = 1), SRCADDR is calculated as follows:

If BTCTRL.STEPSEL = 1:

$$\mathrm{SRCADDR} = \mathrm{SRCADDR}_{START} + BTCNT \cdot BEATSIZE \cdot 2^{\mathrm{STEPSIZE}}$$

If BTCTRL.STEPSEL = 0:

$$\mathrm{SRCADDR} = \mathrm{SRCADDR}_{START} + BTCNT \cdot BEATSIZE$$

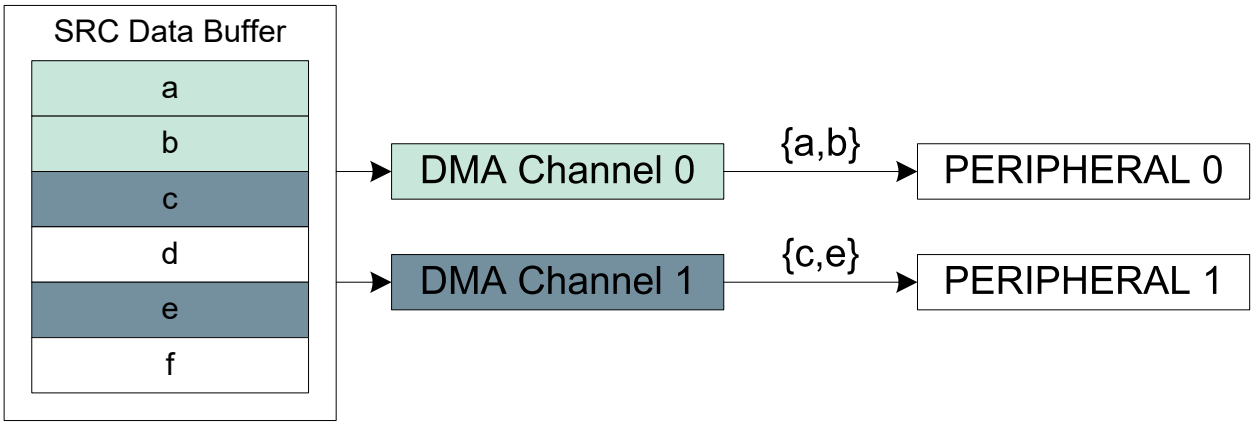
Where,

- $\mathrm{SRCADDR}_{START}$ is the source address of the first beat transfer in the block transfer.
- BTCNT is the initial number of beats remaining in the block transfer.
- BEATSIZE is the configured number of bytes in a beat.
- STEPSIZE is the configured number of beats for each incrementation.

The following figure shows an example where DMA channel 0 is configured to increment the source address by one beat after each beat transfer (BTCTRL.SRCINC = 1), and DMA channel 1 is configured to increment the source address by two beats (BTCTRL.SRCINC = 1, BTCTRL.STEPSEL = 1, and BTCTRL.STEPSIZE = 1). As the destination address for both channels are peripherals, destination incrementation is disabled (BTCTRL.DSTINC = 0).


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-8.** Source Address Increment




Incrementation for the destination address of a block transfer is enabled by setting the Destination Address Incrementation Enable bit in the Block Transfer Control register (BTCTRL.DSTINC = 1). The step size of the incrementation is configurable by clearing BTCTRL.STEPSEL = 0 and writing BTCTRL.STEPSIZE to the desired step size. If BTCTRL.STEPSEL = 1, the step size for the destination incrementation will be the size of one beat.

When the destination address incrementation is configured (BTCTRL.DSTINC = 1), DSTADDR must be set and calculated as follows:

If BTCTRL.STEPSEL = 1:

$$\mathrm{DSTADDR} = \mathrm{DSTADDR}_{START} + BTCNT \cdot 2^{\mathrm{BEATSIZE}}$$

If BTCTRL.STEPSEL = 0:

$$\mathrm{DSTADDR} = \mathrm{DSTADDR}_{START} + BTCNT \cdot 2^{(\mathrm{BEATSIZE} + \mathrm{STEPSIZE})}$$

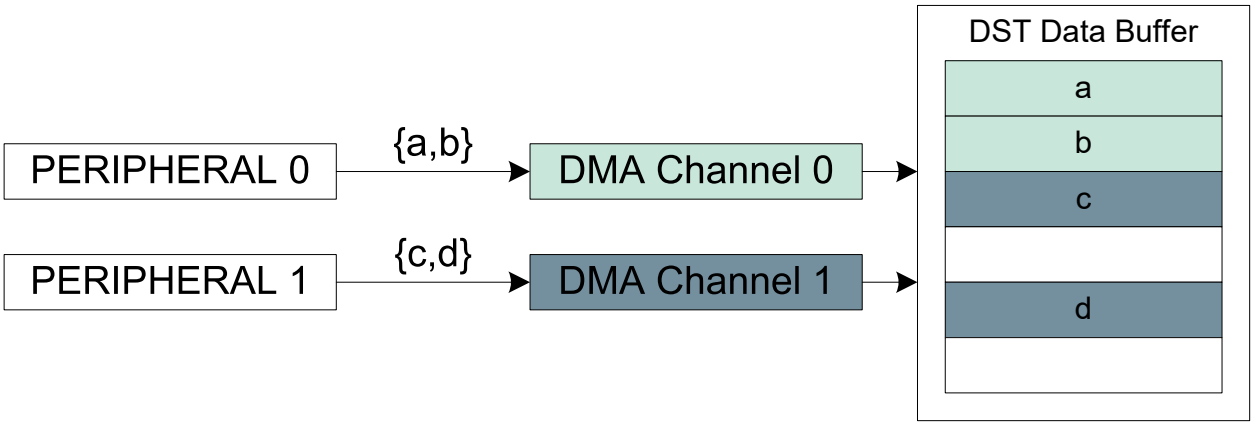
Where,

- $\mathrm{DSTADDR}_{START}$ is the destination address of the first beat transfer in the block transfer.
- BTCNT is the initial number of beats remaining in the block transfer.
- BEATSIZE is the configured number of bytes in a beat.
- STEPSIZE is the configured number of beats for each incrementation.

The followiong figure shows an example where DMA channel 0 is configured to increment destination address by one beat (BTCTRL.DSTINC = 1) and DMA channel 1 is configured to increment destination address by two beats (BTCTRL.DSTINC = 1, BTCTRL.STEPSEL = 0, and BTCTRL.STEPSIZE = 1). As the source address for both channels are peripherals, source incrementation is disabled (BTCTRL.SRCINC = 0).


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-9.** Destination Address Increment




#### 25.6.2.8. Error Handling

If a bus error is received from an AHB client during a DMA data transfer, the corresponding active channel is disabled and the corresponding Channel Transfer Error Interrupt flag in the Channel Interrupt Status and Clear register (CHINTFLAG.TERR) is set. If enabled, the optional transfer error interrupt is generated. The transfer counter will not be decremented and its current value is written-back in the write-back memory section before the channel is disabled.

When the DMAC fetches an invalid descriptor (BTCTRL.VALID=0) or when the channel is resumed and the DMA fetches the next descriptor with null address (DESCADDR=0x00000000), the corresponding channel operation is suspended, the Channel Suspend Interrupt Flag in the Channel Interrupt Flag Status and Clear register (CHINTFLAG.SUSP) is set, and the Channel Fetch Error bit in the Channel Status register (CHSTATUS.FERR) is set. If enabled, the optional suspend interrupt is generated.

### 25.6.3. Additional Features

#### 25.6.3.1. Linked Descriptors

A transaction can consist of either a single block transfer or of several block transfers. When a transaction consists of several block transfers it is done with the help of linked descriptors.

Figure 25-3 illustrates how linked descriptors work. When the first block transfer is completed on DMA channel 0, the DMAC fetches the next transfer descriptor, which is pointed to by the value stored in the Next Descriptor Address (DESCADDR) register of the first transfer descriptor. Fetching the next transfer descriptor (DESCADDR) is continued until the last transfer descriptor. When the block transfer for the last transfer descriptor is executed and DESCADDR=0x00000000, the transaction is terminated. For further details on how the next descriptor is fetched from SRAM, refer to section Data Transmission.

##### 25.6.3.1.1. Adding Descriptor to the End of a List

To add a new descriptor at the end of the descriptor list, create the descriptor in SRAM, with DESCADDR=0x00000000 indicating that it is the new last descriptor in the list, and modify the DESCADDR value of the current last descriptor to the address of the newly created descriptor.

##### 25.6.3.1.2. Modifying a Descriptor in a List

In order to add descriptors to a linked list, the following actions must be performed:

1. Enable the Suspend interrupt for the DMA channel.
2. Enable the DMA channel.
3. Reserve memory space in SRAM to configure a new descriptor.
4. Configure the new descriptor:


© 2026 Microchip Technology Inc. and its subsidiaries

- Set the next descriptor address (DESCADDR)
- Set the destination address (DSTADDR)
- Set the source address (SRCADDR)
- Configure the block transfer control (BTCTRL) including
  - Optionally enable the Suspend block action
  - Set the descriptor VALID bit

5. Clear the VALID bit for the existing list and for the descriptor which has to be updated.
6. Read DESCADDR from the Write-Back memory.
   - If the DMA has not already fetched the descriptor which requires changes (i.e., DESCADDR is wrong):
     - Update the DESCADDR location of the descriptor from the List
     - Optionally clear the Suspend block action
     - Set the descriptor VALID bit to '1'
     - Optionally enable the Resume software command
   - If the DMA is executing the same descriptor as the one which requires changes:
     - Set the Channel Suspend software command and wait for the Suspend interrupt
     - Update the next descriptor address (DESCRADDR) in the write-back memory
     - Clear the interrupt sources and set the Resume software command
     - Update the DESCADDR location of the descriptor from the List
     - Optionally clear the Suspend block action
     - Set the descriptor VALID bit to '1'
7. Go to step 4 if needed.

#### 25.6.3.1.3. Adding a Descriptor Between Existing Descriptors

To insert a new descriptor 'C' between two existing descriptors ('A' and 'B'), the descriptor currently executed by the DMA must be identified.

1. If DMA is executing descriptor B, descriptor C cannot be inserted.
2. If DMA has not started to execute descriptor A, follow the steps:
   a. Set the descriptor A VALID bit to '0'.
   b. Set the DESCADDR value of descriptor A to point to descriptor C instead of descriptor B.
   c. Set the DESCADDR value of descriptor C to point to descriptor B.
   d. Set the descriptor A VALID bit to '1'.
3. If DMA is executing descriptor A:
   a. Apply the software suspend command to the channel and
   b. Perform steps 2.1 through 2.4.
   c. Apply the software resume command to the channel.

### 25.6.3.2. Channel Suspend

The channel operation can be suspended at any time by software by writing a '1' to the Suspend command in the Command bit field of Channel Control B register (CHCTRLB.CMD). After the ongoing burst transfer is completed, the channel operation is suspended and the suspend command is automatically cleared.

When suspended, the Channel Suspend Interrupt flag in the Channel Interrupt Status and Clear register is set (CHINTFLAG.SUSP=1) and the optional suspend interrupt is generated.


© 2026 Microchip Technology Inc. and its subsidiaries

By configuring the block action to suspend by writing Block Action bit group in the Block Transfer Control register (BTCTRL.BLOCKACT is 0x2 or 0x3), the DMA channel will be suspended after it has completed a block transfer. The DMA channel will be kept enabled and will be able to receive transfer triggers, but it will be removed from the arbitration scheme.

If an invalid transfer descriptor (BTCTRL.VALID=0) is fetched from SRAM, the DMA channel will be suspended, and the Channel Fetch Error bit in the Channel Status register(CHASTATUS.FERR) will be set.

**Note:** Only enabled DMA channels can be suspended. If a channel is disabled when it is attempted to be suspended, the internal suspend command will be ignored.

For more details on transfer descriptors, refer to section Transfer Descriptors.

**Related Links**

CHCTRLB

Channel Control B

CHINTFLAG

Channel Interrupt Flag Status and Clear

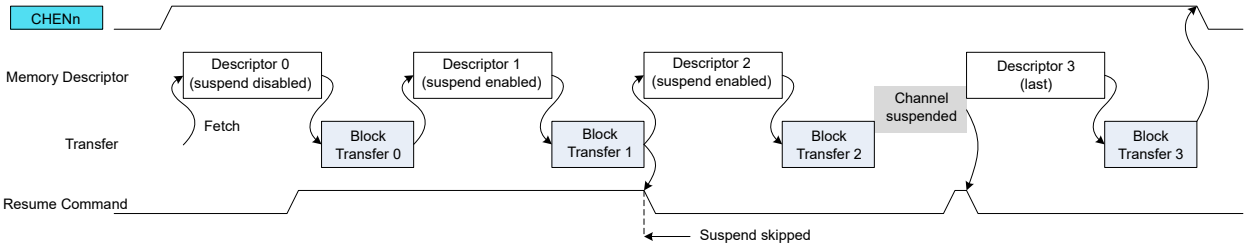
BTCTRL

Block Transfer Control

### 25.6.3.3. Channel Resume and Next Suspend Skip

A channel operation can be resumed by software by setting the Resume command in the Command bit field of the Channel Control B register (CHCTRLB.CMD). If the channel is already suspended, the channel operation resumes from where it previously stopped when the Resume command is detected. When the Resume command is issued before the channel is suspended, the next suspend action is skipped and the channel continues the normal operation.

**Figure 25-10.** Channel Suspend/Resume Operation




**Related Links**

CHCTRLB

Channel Control B

### 25.6.3.4. Event Input Actions

The event input actions are available only on the least significant DMA channels. For details on channels with event input support, refer to the in the Event system documentation.

Before using event input actions, the event controller must be configured first according to the following table, and the Channel Event Input Enable bit in the Channel Control B register (CHCTRLB.EVIE) must be written to '1'. Refer also to Events.

**Table 25-1.** Event Input Action

| Action | CHCTRLB.EVACT | CHCTRLB.TRGSRC |
|---|---|---|
| None | NOACT | - |
| Normal Transfer | TRIG | DISABLE |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 25-1.** Event Input Action (continued)

| Action | CHCTRLB.EVACT | CHCTRLB.TRGSRC |
|---|---|---|
| Conditional Transfer on Strobe | TRIG | any peripheral |
| Conditional Transfer | CTRIG | |
| Conditional Block Transfer | CBLOCK | |
| Channel Suspend | SUSPEND | |
| Channel Resume | RESUME | |
| Skip Next Block Suspend | SSKIP | |

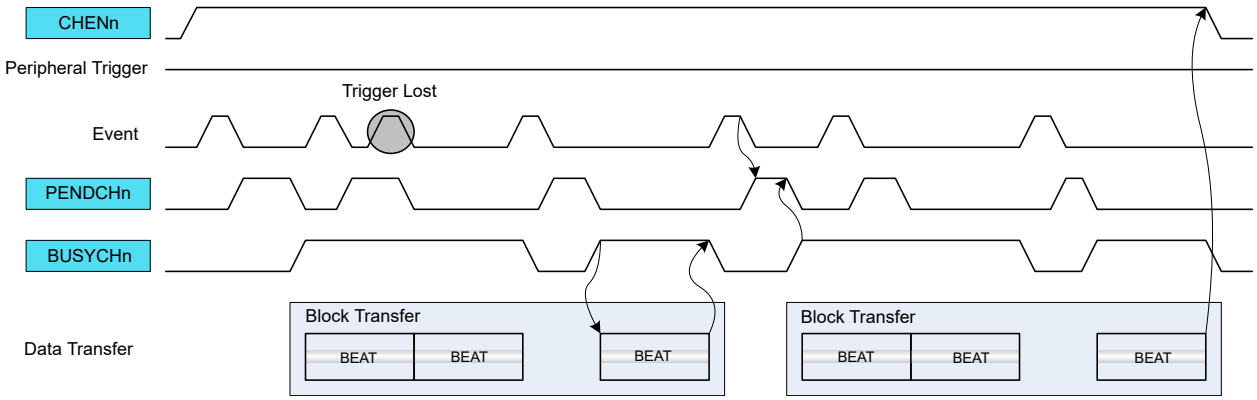
### Normal Transfer

The event input is used to trigger a beat or burst transfer on peripherals.

The event is acknowledged as soon as the event is received. When received, both the Channel Pending status bit in the Channel Status register (CHSTATUS.PEND) and the corresponding Channel n bit in the Pending Channels register (PENDCH.PENDCHn) are set. If the event is received while the channel is pending, the event trigger is lost.

The figure below shows an example where beat transfers are enabled by internal events.

**Figure 25-11.** Beat Event Trigger Action




### Conditional Transfer on Strobe

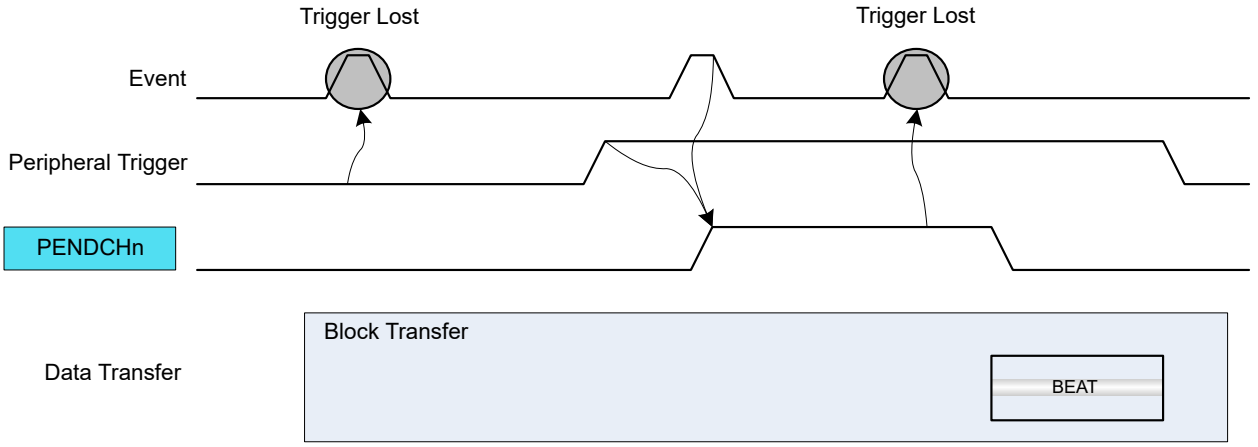
The event input is used to trigger a transfer on peripherals with pending transfer requests. This event action is intended to be used with peripheral triggers, e.g. for timed communication protocols or periodic transfers between peripherals: only when the peripheral trigger coincides with the occurrence of a (possibly cyclic) event the transfer is issued.

The event is acknowledged as soon as the event is received. The peripheral trigger request is stored internally when the previous trigger action is completed (i.e. the channel is not pending) and when an active event is received. If the peripheral trigger is active, the DMA will wait for an event before the peripheral trigger is internally registered. When both event and peripheral transfer trigger are active, both CHSTATUS.PEND and PENDCH.PENDCHn are set. A software trigger will now trigger a transfer.

The figure below shows an example where the peripheral beat transfer is started by a conditional strobe event action.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-12.** Periodic Event with Beat Peripheral Triggers



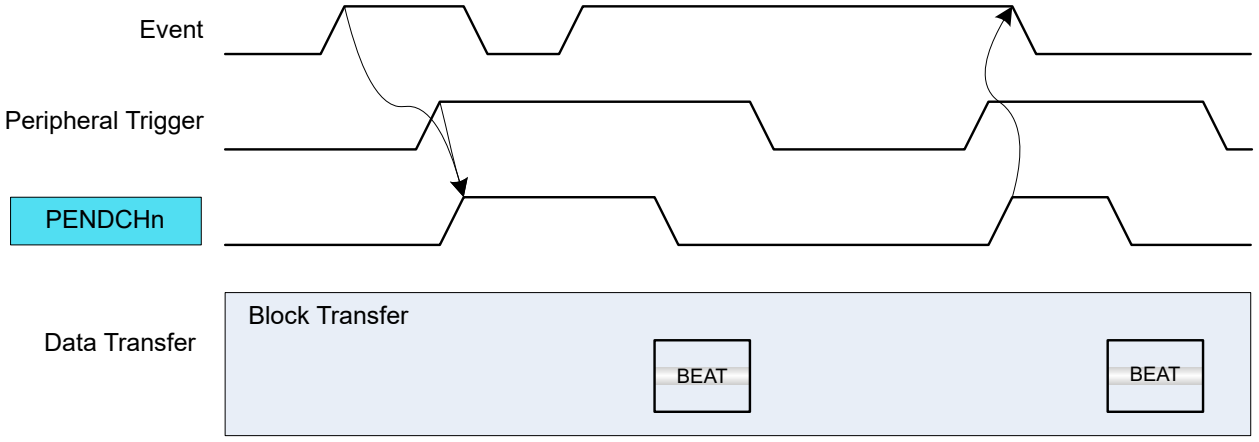

## Conditional Transfer

The event input is used to trigger a conditional transfer on peripherals with pending transfer requests. For example, this type of event can be used for peripheral-to-peripheral transfers, where one peripheral is the source of event and the second peripheral is the source of the trigger.

Each peripheral trigger is stored internally when the event is received. When the peripheral trigger is stored internally, the Channel Pending status bit is set (CHSTATUS.PEND), the respective Pending Channel n Bit in the Pending Channels register is set (PENDCH.PENDCHn), and the event is acknowledged. A software trigger will now trigger a transfer.

The figure below shows an example where conditional event is enabled with peripheral beat trigger requests.

**Figure 25-13.** Conditional Event with Beat Peripheral Triggers




## Conditional Block Transfer

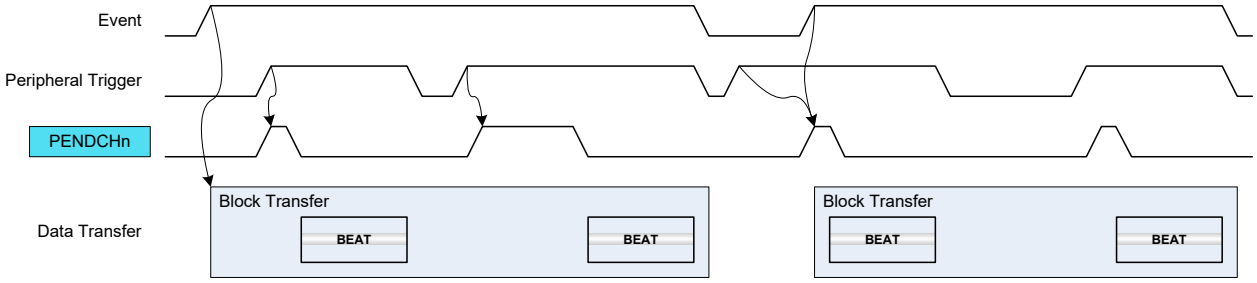
The event input is used to trigger a conditional block transfer on peripherals.

Before starting transfers within a block, an event must be received. When received, the event is acknowledged when the block transfer is completed. A software trigger will trigger a transfer.

The figure below shows an example where conditional event block transfer is started with peripheral beat trigger requests.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-14.** Conditional Block Transfer with Beat Peripheral Triggers




### Channel Suspend

The event input is used to suspend an ongoing channel operation. The event is acknowledged when the current AHB access is completed. For further details on Channel Suspend, refer to Channel Suspend.

### Channel Resume

The event input is used to resume a suspended channel operation. The event is acknowledged as soon as the event is received and the Channel Suspend Interrupt Flag (CHINTFLAG.SUSP) is cleared. For further details refer to Channel Suspend.

### Skip Next Block Suspend

This event can be used to skip the next block suspend action. If the channel is suspended before the event rises, the channel operation is resumed and the event is acknowledged. If the event rises before a suspend block action is detected, the event is kept until the next block suspend detection. When the block transfer is completed, the channel continues the operation (not suspended) and the event is acknowledged.

## 25.6.3.5. Event Output Selection

Event output selection is available only for the least significant DMA channels. The pulse width of an event output from a channel is one AHB clock cycle.

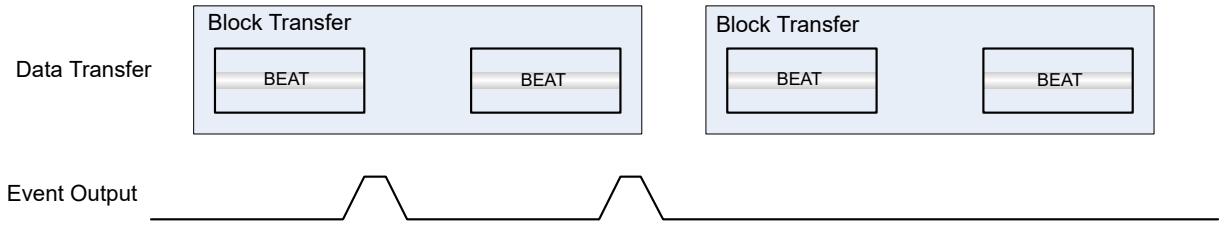
The output of channel events is enabled by writing a '1' to the Channel Event Output Enable bit in the Control B register (CHCTRLB.EVOE). The event output cause is selected by writing to the Event Output Selection bits in the Block Transfer Control register (BTCTRL.EVOSEL). It is possible to generate events after each block transfer (BTCTRL.EVOSEL=0x1) or beat transfer (BTCTRL.EVOSEL=0x3). To enable an event being generated when a transaction is complete, the block event selection must be set in the last transfer descriptor only.

Figure 25-15 shows an example where the event output generation is enabled in the first block transfer, and disabled in the second block.

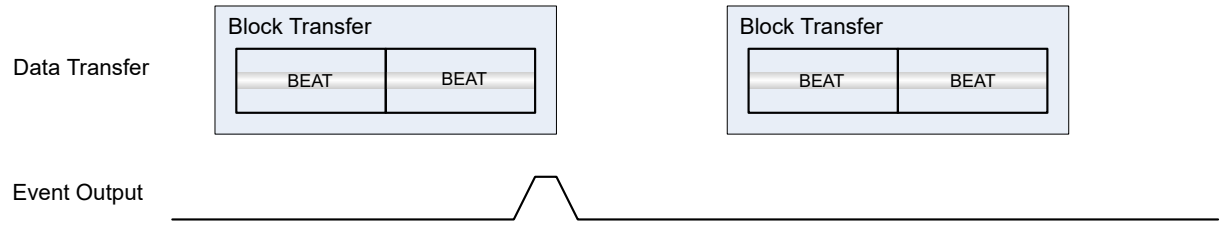

© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-15.** Event Output Generation

**Beat Event Output**




**Block Event Output**




**Related Links**

CHCTRLB

Channel Control B

BTCTRL

Block Transfer Control

### 25.6.3.6. Aborting Transfers

Transfers on any channel can be aborted gracefully by software by disabling the corresponding DMA channel. It is also possible to abort all ongoing or pending transfers by disabling the DMAC.

When a DMA channel disable request or DMAC disable request is detected:

- Ongoing transfers of the active channel will be disabled when the ongoing beat transfer is completed and the write-back memory section is updated. This prevents transfer corruption before the channel is disabled.
- All other enabled channels will be disabled in the next clock cycle.

The corresponding Channel Enable bit in the Channel Control A register is cleared (CHCTRLA.ENABLE=0) when the channel is disabled.

The corresponding DMAC Enable bit in the Control register is cleared (CTRL.DMAENABLE=0) when the entire DMAC module is disabled.

### 25.6.3.7. CRC Operation

A cyclic redundancy check (CRC) is an error detection technique used to find errors in data. It is commonly used to determine whether the data during a transmission, or data present in data and program memories has been corrupted or not. A CRC takes a data stream or a block of data as input and generates a 16- or 32-bit output that can be appended to the data and used as a checksum.

When the data is received, the device or application repeats the calculation: If the new CRC result does not match the one calculated earlier, the block contains a data error. The application will then detect this and may take a corrective action, such as requesting the data to be sent again or simply not using the incorrect data.


© 2026 Microchip Technology Inc. and its subsidiaries

The CRC engine in DMAC supports two commonly used CRC polynomials: CRC-16 (CRC-CCITT) and CRC-32 (IEEE 802.3). Typically, applying CRC-n (CRC-16 or CRC-32) to a data block of arbitrary length will detect any single alteration that is ≤n bits in length, and will detect the fraction 1-2-n of all longer error bursts.

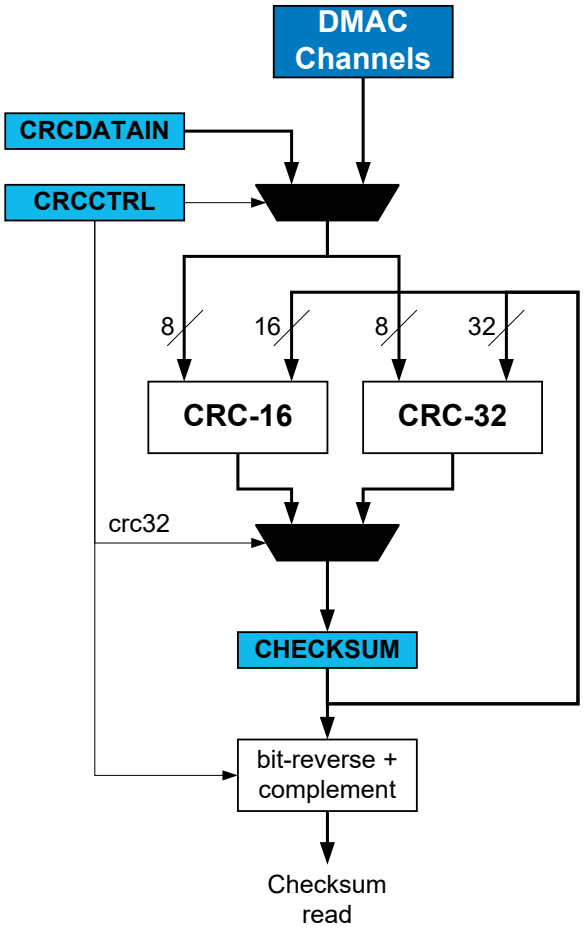
- CRC-16:
  - Polynomial: $x^{16}+ x^{12}+ x^5+ 1$
  - Hex value: 0x1021
- CRC-32:
  - Polynomial: $x^{32}+x^{26}+ x^{23}+ x^{22}+x^{16}+ x^{12}+ x^{11}+ x^{10}+ x^8+ x^7+ x^5+ x^4+ x^2+ x + 1$
  - Hex value: 0x04C11DB7

The data source for the CRC engine can either be one of the DMA channels or the APB bus interface, and must be selected by writing to the CRC Input Source bits in the CRC Control register (CRCCTRL.CRCSRC). The CRC engine then takes data input from the selected source and generates a checksum based on these data. The checksum is available in the CRC Checksum register (CRCCHKSUM). When CRC-32 polynomial is used, the final checksum read is bit reversed and complemented, as shown in Figure 25-16.

The CRC polynomial is selected by writing to the CRC Polynomial Type bit in the CRC Control register (CRCCTRL.CRCPOLY), the default is CRC-16. The CRC engine operates on byte only. When the DMA is used as data source for the CRC engine, the DMA channel beat size setting will be used. When used with APB bus interface, the application must select the CRC Beat Size bit field of CRC Control register (CRCCTRL.CRCBEATSIZE). 8-, 16-, or 32-bit bus transfer access type is supported. The corresponding number of bytes will be written in the CRCDATAIN register and the CRC engine will operate on the input data in a byte by byte manner.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 25-16.** CRC Generator Block Diagram




| | |
|---|---|
| **CRC on DMA data** | CRC-16 or CRC-32 calculations can be performed on data passing through any DMA channel. Once a DMA channel is selected as the source, the CRC engine will continuously generate the CRC on the data passing through the DMA channel. The checksum is available for readout once the DMA transaction is completed or aborted. A CRC can also be generated on SRAM, Flash, or I/O memory by passing these data through a DMA channel. If the latter is done, the destination register for the DMA data can be the data input (CRCDATAIN) register in the CRC engine. |
| **CRC using the I/O interface** | Before using the CRC engine with the I/O interface, the application must set the CRC Beat Size bits in the CRC Control register (CRCCTRL.CRCBEATSIZE). 8/16/32-bit bus transfer type can be selected. |

CRC can be performed on any data by loading them into the CRC engine using the CPU and writing the data to the CRCDATAIN register. Using this method, an arbitrary number of bytes can be written to the register by the CPU, and CRC is done continuously for each byte. This means if a 32-bit data is written to the CRCDATAIN register the CRC engine takes four cycles to calculate the CRC. The CRC complete is signaled by a set CRCBUSY bit in the CRCSTATUS register. New data can be written only when CRCBUSY flag is not set.

### 25.6.4. DMA Operation

Not applicable.

### 25.6.5. Interrupts

The DMAC channels have the following interrupt sources:


© 2026 Microchip Technology Inc. and its subsidiaries

- Transfer Complete (TCMPL): Indicates that a block transfer is completed on the corresponding channel. Refer to Data Transmission for details.
- Transfer Error (TERR): Indicates that a bus error has occurred during a burst transfer, or that an invalid descriptor has been fetched. Refer to Error Handling for details.
- Channel Suspend (SUSP): Indicates that the corresponding channel has been suspended. Refer to Channel Suspend and Data Transmission for details.

Each interrupt source has an Interrupt flag associated with it. The Interrupt flag in the Channel Interrupt Flag Status and Clear (CHINTFLAG) register is set when the Interrupt condition occurs. Each interrupt can be individually enabled by setting the corresponding bit in the Channel Interrupt Enable Set register (CHINTENSET=1), and disabled by setting the corresponding bit in the Channel Interrupt Enable Clear register (CHINTENCLR=1). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the Interrupt flag is cleared, the interrupt is disabled, the DMAC is reset or the corresponding DMA channel is reset. See CHINTFLAG for details on how to clear Interrupt flags. All interrupt requests are ORed together on system level to generate one combined interrupt request to the NVIC.

The user must read the Channel Interrupt Status (INTSTATUS) register to identify the channels with pending interrupts and must read the Channel Interrupt Flag Status and Clear (CHINTFLAG) register to determine which Interrupt condition is present for the corresponding channel. It is also possible to read the Interrupt Pending register (INTPEND), which provides the lowest channel number with pending interrupt and the respective Interrupt flags.

**Note:** Interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**
Nested Vector Interrupt Controller

### 25.6.6. Events

The DMAC can generate the following output events:

- Channel (CH): Generated when a block transfer for a given channel has been completed, or when a beat transfer within a block transfer for a given channel has been completed. Refer to *Event Output Selection* for details.

Setting the Channel Event Output Enable bit (CHEVCTRLx.EVOE = 1) enables the corresponding output event configured in the Event Output Selection bit group in the Block Transfer Control register (BTCTRL.EVOSEL). Clearing CHEVCTRLx.EVOE = 0 disables the corresponding output event.

The DMAC can take the following actions on an input event:

- Transfer and Periodic Transfer Trigger (TRIG): normal transfer or periodic transfers on peripherals are enabled
- Conditional Transfer Trigger (CTRIG): conditional transfers on peripherals are enabled
- Conditional Block Transfer Trigger (CBLOCK): conditional block transfers on peripherals are enabled
- Channel Suspend Operation (SUSPEND): suspend a channel operation
- Channel Resume Operation (RESUME): resume a suspended channel operation
- Skip Next Block Suspend Action (SSKIP): skip the next block suspend transfer condition
- Increase Priority (INCPRI): increase channel priority

Setting the Channel Event Input Enable bit (CHEVCTRLx.EVIE = 1) enables the corresponding action on input event. Clearing this bit disables the corresponding action on input event. Note that several actions can be enabled for incoming events. If several events are connected to the peripheral, any


© 2026 Microchip Technology Inc. and its subsidiaries

enabled action will be taken for any of the incoming events. For further details on event input actions, refer to *Event Input Actions*.
**Note:** Event input and outputs are not available for every channel. Refer to the Features section for more information.

**Related Links**
Event System (EVSYS)
CHCTRLB
Channel Control B
BTCTRL
Block Transfer Control

### 25.6.7. Sleep Mode Operation

Each DMA channel can be configured to operate in any sleep mode. To be able to run in standby, the RUNSTDBY bit in Channel Control A register (CHCTRLA.RUNSTDBY) must be written to '1'. The DMAC can wake up the device using interrupts from any sleep mode or perform actions through the Event System.

For channels with CHCTRLA.RUNSTDBY = 0, it is up to software to stop DMA transfers on these channels and wait for completion before going to standby mode using the following sequence:

1. Suspend the DMAC channels for which CHCTRLA.RUNSTDBY = 0.
2. Check the SYNCBUSY bits of registers accessed by the DMAC channels being suspended.
3. Go to sleep.
4. When the device wakes up, resume the suspended channels.

**Note:** In Stand-by Sleep mode, the DMAC can only access RAM when it is not back biased (PM.STDBYCFG.BBIASxx = 0x0)

### 25.6.8. Synchronization

Not applicable.


© 2026 Microchip Technology Inc. and its subsidiaries

## 25.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRL | 7:0 | | | | | | CRCENABLE | DMAENABLE | SWRST |
| | | 15:8 | | | | | LVLEN3 | LVLEN2 | LVLEN1 | LVLEN0 |
| 0x02 | CRCCTRL | 7:0 | | | | | CRCPOLY[1:0] | | CRCBEATSIZE[1:0] | |
| | | 15:8 | | | CRCSRC[5:0] | | | | | |
| 0x04 | CRCDATAIN | 7:0 | CRCDATAIN[7:0] | | | | | | | |
| | | 15:8 | CRCDATAIN[15:8] | | | | | | | |
| | | 23:16 | CRCDATAIN[23:16] | | | | | | | |
| | | 31:24 | CRCDATAIN[31:24] | | | | | | | |
| 0x08 | CRCCHKSUM | 7:0 | CRCCHKSUM[7:0] | | | | | | | |
| | | 15:8 | CRCCHKSUM[15:8] | | | | | | | |
| | | 23:16 | CRCCHKSUM[23:16] | | | | | | | |
| | | 31:24 | CRCCHKSUM[31:24] | | | | | | | |
| 0x0C | CRCSTATUS | 7:0 | | | | | | | CRCZERO | CRCBUSY |
| 0x0D | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0E | QOSCTRL | 7:0 | | | DQOS[1:0] | | FQOS[1:0] | | WRBQOS[1:0] | |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SWTRIGCTRL | 7:0 | SWTRIG7 | SWTRIG6 | SWTRIG5 | SWTRIG4 | SWTRIG3 | SWTRIG2 | SWTRIG1 | SWTRIG0 |
| | | 15:8 | | | | | SWTRIG11 | SWTRIG10 | SWTRIG9 | SWTRIG8 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | PRICTRL0 | 7:0 | RRLVLEN0 | | | | LVLPRI0[3:0] | | | |
| | | 15:8 | RRLVLEN1 | | | | LVLPRI1[3:0] | | | |
| | | 23:16 | RRLVLEN2 | | | | LVLPRI2[3:0] | | | |
| | | 31:24 | RRLVLEN3 | | | | LVLPRI3[3:0] | | | |
| 0x18 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | INTPEND | 7:0 | | | | | ID[3:0] | | | |
| | | 15:8 | PEND | BUSY | FERR | | | SUSP | TCMPL | TERR |
| 0x22 ... 0x23 | Reserved | | | | | | | | | |
| 0x24 | INTSTATUS | 7:0 | CHINT7 | CHINT6 | CHINT5 | CHINT4 | CHINT3 | CHINT2 | CHINT1 | CHINT0 |
| | | 15:8 | | | | | CHINT11 | CHINT10 | CHINT9 | CHINT8 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x28 | BUSYCH | 7:0 | BUSYCH7 | BUSYCH6 | BUSYCH5 | BUSYCH4 | BUSYCH3 | BUSYCH2 | BUSYCH1 | BUSYCH0 |
| | | 15:8 | | | | | BUSYCH11 | BUSYCH10 | BUSYCH9 | BUSYCH8 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x2C | PENDCH | 7:0 | PENDCH7 | PENDCH6 | PENDCH5 | PENDCH4 | PENDCH3 | PENDCH2 | PENDCH1 | PENDCH0 |
| | | 15:8 | | | | | PENDCH11 | PENDCH10 | PENDCH9 | PENDCH8 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x30 | ACTIVE | 7:0 | | | | | LVLEX3 | LVLEX2 | LVLEX1 | LVLEX0 |
| | | 15:8 | ABUSY | | | ID[4:0] | | | | |
| | | 23:16 | BTCNT[7:0] | | | | | | | |
| | | 31:24 | BTCNT[15:8] | | | | | | | |
| 0x34 | BASEADDR | 7:0 | BASEADDR[7:0] | | | | | | | |
| | | 15:8 | BASEADDR[15:8] | | | | | | | |
| | | 23:16 | BASEADDR[23:16] | | | | | | | |
| | | 31:24 | BASEADDR[31:24] | | | | | | | |
| 0x38 | WRBADDR | 7:0 | WRBADDR[7:0] | | | | | | | |
| | | 15:8 | WRBADDR[15:8] | | | | | | | |
| | | 23:16 | WRBADDR[23:16] | | | | | | | |
| | | 31:24 | WRBADDR[31:24] | | | | | | | |
| 0x3C ... 0x3E | Reserved | | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x3F | CHID | 7:0 | | | | | ID[3:0] | | | |
| 0x40 | CHCTRLA | 7:0 | | RUNSTDBY | | | | | ENABLE | SWRST |
| 0x41<br>...<br>0x43 | Reserved | | | | | | | | | |
| 0x44 | CHCTRLB | 7:0 | | LVL[1:0] | | EVOE | EVIE | EVACT[2:0] | | |
| | | 15:8 | | | TRIGSRC[5:0] | | | | | |
| | | 23:16 | TRIGACT[1:0] | | | | | | | |
| | | 31:24 | | | | | | | CMD[1:0] | |
| 0x48<br>...<br>0x4B | Reserved | | | | | | | | | |
| 0x4C | CHINTENCLR | 7:0 | | | | | | SUSP | TCMPL | TERR |
| 0x4D | CHINTENSET | 7:0 | | | | | | SUSP | TCMPL | TERR |
| 0x4E | CHINTFLAG | 7:0 | | | | | | SUSP | TCMPL | TERR |
| 0x4F | CHSTATUS | 7:0 | | | | | | FERR | BUSY | PEND |

## 25.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.1. Control

**Name:** CTRL
**Offset:** 0x00
**Reset:** 0x00X0
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LVLEN3 | LVLEN2 | LVLEN1 | LVLEN0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CRCENABLE | DMAENABLE | SWRST |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 8, 9, 10, 11 – LVLENx** Priority Level x Enable [x=0..3]
When this bit is set, all requests with the corresponding level will be fed into the arbiter block. When cleared, all requests with the corresponding level will be ignored.
For details on arbitration schemes, refer to the Arbitration section.
These bits are not enable-protected.

| Value | Description |
|---|---|
| 0 | Transfer requests for Priority level x will not be handled |
| 1 | Transfer requests for Priority level x will be handled |

**Bit 2 – CRCENABLE** CRC Enable
Writing a '0' to this bit will disable the CRC calculation when the CRC Status Busy flag is cleared (CRCSTATUS. CRCBUSY). The bit is zero when the CRC is disabled.
Writing a '1' to this bit will enable the CRC calculation.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The CRC calculation is disabled |
| 1 | The CRC calculation is enabled |

**Bit 1 – DMAENABLE** DMA Enable
Setting this bit will enable the DMA module.
Writing a '0' to this bit will disable the DMA module. When writing a '0' during an ongoing transfer, the bit will not be cleared until the internal data transfer buffer is empty and the DMA transfer is aborted. The internal data transfer buffer will be empty once the ongoing burst transfer is completed.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled |
| 1 | The peripheral is enabled |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit when both the DMAC and the CRC module are disabled (DMAENABLE and CRCENABLE are '0') resets all registers in the DMAC (except DBGCTRL) to their initial state. If either the DMAC or CRC module is enabled, the Reset request will be ignored and the DMAC will return an access error.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing |
| 1 | A Reset operation is ongoing |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.2. CRC Control

**Name:** CRCCTRL
**Offset:** 0x02
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | CRCSRC[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CRCPOLY[1:0] | | CRCBEATSIZE[1:0] | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 13:8 – CRCSRC[5:0]** CRC Input Source

These bits select the input source for generating the CRC, as shown in the table below. The selected source is locked until either the CRC generation is completed or the CRC module is disabled. This means the CRCSRC cannot be modified when the CRC operation is ongoing. The lock is signaled by the CRCBUSY Status bit. CRC generation complete is generated and signaled from the selected source when used with the DMA channel.

| Value | Name | Description |
|---|---|---|
| 0x00 | NOACT | No action |
| 0x01 | IO | I/O interface |
| 0x02-0x1F | - | Reserved |
| 0x20 | CHN | DMA channel 0 |
| 0x21 | CHN | DMA channel 1 |
| 0x22 | CHN | DMA channel 2 |
| 0x23 | CHN | DMA channel 3 |
| 0x24 | CHN | DMA channel 4 |
| 0x25 | CHN | DMA channel 5 |
| 0x26 | CHN | DMA channel 6 |
| 0x27 | CHN | DMA channel 7 |
| 0x28 | CHN | DMA channel 8 |
| 0x29 | CHN | DMA channel 9 |
| 0x2A | CHN | DMA channel 10 |
| 0x2B | CHN | DMA channel 11 |

**Bits 3:2 – CRCPOLY[1:0]** CRC Polynomial Type

These bits define the size of the data transfer for each bus access when the CRC is used with I/O interface, as shown in the table below.

| Value | Name | Description |
|---|---|---|
| 0x0 | CRC16 | CRC-16 (CRC-CCITT) |
| 0x1 | CRC32 | CRC32 (IEEE 802.3) |
| 0x2-0x3 | | Reserved |

**Bits 1:0 – CRCBEATSIZE[1:0]** CRC Beat Size

These bits define the size of the data transfer for each bus access when the CRC is used with I/O interface.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | BYTE | 8-bit bus transfer |
| 0x1 | HWORD | 16-bit bus transfer |
| 0x2 | WORD | 32-bit bus transfer |
| 0x3 | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.3. CRC Data Input

**Name:** CRCDATAIN
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CRCDATAIN[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CRCDATAIN[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CRCDATAIN[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CRCDATAIN[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CRCDATAIN[31:0]** CRC Data Input

These bits store the data for which the CRC checksum is computed. A new CRC Checksum is ready (CRCBEAT+ 1) clock cycles after the CRCDATAIN register is written.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.4. CRC Checksum

**Name:** CRCCHKSUM
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

The CRCCHKSUM represents the 16- or 32-bit checksum value and the generated CRC. The register is reset to zero by default, but it is possible to reset all bits to one by writing the CRCCHKSUM register directly. It is possible to write this register only when the CRC module is disabled. If CRC-32 is selected and the CRC Status Busy flag is cleared (i.e., CRC generation is completed or aborted), the bit reversed (bit 31 is swapped with bit 0, bit 30 with bit 1, etc.) and complemented result will be read from CRCCHKSUM. If CRC-16 is selected or the CRC Status Busy flag is set (i.e., CRC generation is ongoing), CRCCHKSUM will contain the actual content.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CRCCHKSUM[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CRCCHKSUM[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CRCCHKSUM[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CRCCHKSUM[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CRCCHKSUM[31:0]** CRC Checksum
These bits store the generated CRC result. The 16 MSB bits are always read zero when CRC-16 is enabled.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.5. CRC Status

**Name:** CRCSTATUS
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | CRCZERO | CRCBUSY |
| Access | | | | | | | R | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – CRCZERO** CRC Zero

This bit is cleared when a new CRC source is selected.
This bit is set when the CRC generation is complete and the CRC Checksum is zero.
When running CRC-32 and appending the checksum at the end of the packet (as little endian), the final checksum should be 0x2144df1c, and not zero. However, if the checksum is complemented before it is appended (as little endian) to the data, the final result in the checksum register will be zero. See the description of CRCCHKSUM to read out different versions of the checksum.

**Bit 0 – CRCBUSY** CRC Module Busy

This flag is cleared by writing a one to it when used with I/O interface. When used with a DMA channel, the bit is set when the corresponding DMA channel is enabled, and cleared when the corresponding DMA channel is disabled. This register bit cannot be cleared by the application when the CRC is used with a DMA channel.
This bit is set when a source configuration is selected and as long as the source is using the CRC module.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.6. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0D
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The DMAC is halted when the CPU is halted by an external debugger. |
| 1 | The DMAC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.7. Quality of Service Control

**Name:** QOSCTRL
**Offset:** 0x0E
**Reset:** 0x2A
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | DQOS[1:0] | | FQOS[1:0] | | WRBQOS[1:0] | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 1 | 0 | 1 | 0 | 1 | 0 |

**Bits 5:4 – DQOS[1:0]** Data Transfer Quality of Service
These bits define the memory priority access during the data transfer operation.

| DQOS[1:0] | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Background (no sensitive operation) |
| 0x1 | LOW | Sensitive Bandwidth |
| 0x2 | MEDIUM | Sensitive Latency |
| 0x3 | HIGH | Critical Latency |

**Bits 3:2 – FQOS[1:0]** Fetch Quality of Service
These bits define the memory priority access during the fetch operation.

| FQOS[1:0] | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Background (no sensitive operation) |
| 0x1 | LOW | Sensitive Bandwidth |
| 0x2 | MEDIUM | Sensitive Latency |
| 0x3 | HIGH | Critical Latency |

**Bits 1:0 – WRBQOS[1:0]** Write-Back Quality of Service
These bits define the memory priority access during the write-back operation.

| WRBQOS[1:0] | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Background (no sensitive operation) |
| 0x1 | LOW | Sensitive Bandwidth |
| 0x2 | MEDIUM | Sensitive Latency |
| 0x3 | HIGH | Critical Latency |

**Related Links**
SRAM Quality of Service


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.8. Software Trigger Control

**Name:** SWTRIGCTRL
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SWTRIG11 | SWTRIG10 | SWTRIG9 | SWTRIG8 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SWTRIG7 | SWTRIG6 | SWTRIG5 | SWTRIG4 | SWTRIG3 | SWTRIG2 | SWTRIG1 | SWTRIG0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – SWTRIGn** Channel n Software Trigger [n = 11..0]
This bit is cleared when the Channel Pending bit in the Channel Status register (CHSTATUS.PEND) for the corresponding channel is either set, or by writing a '1' to it.
This bit is set if CHSTATUS.PEND is already '1' when writing a '1' to that bit.
Writing a '0' to this bit will clear the bit.
Writing a '1' to this bit will generate a DMA software trigger on channel x, if CHSTATUS.PEND=0 for channel x. CHSTATUS.PEND will be set and SWTRIGn will remain cleared.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.9. Priority Control 0

**Name:** PRICTRL0
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | RRLVLEN3 | | | | LVLPRI3[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | RRLVLEN2 | | | | LVLPRI2[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RRLVLEN1 | | | | LVLPRI1[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RRLVLEN0 | | | | LVLPRI0[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 31 – RRLVLEN3** Level 3 Round-Robin Arbitration Enable
This bit controls which arbitration scheme is selected for DMA channels with priority level 3. For details on arbitration schemes, refer to Arbitration.

| Value | Description |
|---|---|
| 0 | Static arbitration scheme for channels with level 3 priority. |
| 1 | Round-robin arbitration scheme for channels with level 3 priority. |

**Bits 27:24 – LVLPRI3[3:0]** Level 3 Channel Priority Number
When round-robin arbitration is enabled (PRICTRL0.RRLVLEN3=1) for priority level 3, this register holds the channel number of the last DMA channel being granted access as the active channel with priority level 3.
When static arbitration is enabled (PRICTRL0.RRLVLEN3=0) for priority level 3, and the value of this bit group is non-zero, it will not affect the static priority scheme.
This bit group is not reset when round-robin arbitration gets disabled (PRICTRL0.RRLVLEN3 written to '0').

**Bit 23 – RRLVLEN2** Level 2 Round-Robin Arbitration Enable
This bit controls which arbitration scheme is selected for DMA channels with priority level 2. For details on arbitration schemes, refer to Arbitration.

| Value | Description |
|---|---|
| 0 | Static arbitration scheme for channels with level 2 priority. |
| 1 | Round-robin arbitration scheme for channels with level 2 priority. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 19:16 – LVLPRI2[3:0]** Level 2 Channel Priority Number
When round-robin arbitration is enabled (PRICTRL0.RRLVLEN2=1) for priority level 2, this register holds the channel number of the last DMA channel being granted access as the active channel with priority level 2.
When static arbitration is enabled (PRICTRL0.RRLVLEN2=0) for priority level 2, and the value of this bit group is non-zero, it will not affect the static priority scheme.
This bit group is not reset when round-robin arbitration gets disabled (PRICTRL0.RRLVLEN2 written to '0').

**Bit 15 – RRLVLEN1** Level 1 Round-Robin Scheduling Enable
For details on arbitration schemes, refer to Arbitration.

| Value | Description |
|---|---|
| 0 | Static arbitration scheme for channels with level 1 priority. |
| 1 | Round-robin arbitration scheme for channels with level 1 priority. |

**Bits 11:8 – LVLPRI1[3:0]** Level 1 Channel Priority Number
When round-robin arbitration is enabled (PRICTRL0.RRLVLEN1=1) for priority level 1, this register holds the channel number of the last DMA channel being granted access as the active channel with priority level 1.
When static arbitration is enabled (PRICTRL0.RRLVLEN1=0) for priority level 1, and the value of this bit group is non-zero, it will not affect the static priority scheme.
This bit group is not reset when round-robin arbitration gets disabled (PRICTRL0.RRLVLEN1 written to '0').

**Bit 7 – RRLVLEN0** Level 0 Round-Robin Scheduling Enable
For details on arbitration schemes, refer to Arbitration.

| Value | Description |
|---|---|
| 0 | Static arbitration scheme for channels with level 0 priority. |
| 1 | Round-robin arbitration scheme for channels with level 0 priority. |

**Bits 3:0 – LVLPRI0[3:0]** Level 0 Channel Priority Number
When round-robin arbitration is enabled (PRICTRL0.RRLVLEN0=1) for priority level 0, this register holds the channel number of the last DMA channel being granted access as the active channel with priority level 0.
When static arbitration is enabled (PRICTRL0.RRLVLEN0=0) for priority level 0, and the value of this bit group is non-zero, it will not affect the static priority scheme.
This bit group is not reset when round-robin arbitration gets disabled (PRICTRL0.RRLVLEN0 written to '0').


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.10. Interrupt Pending

**Name:** INTPEND
**Offset:** 0x20
**Reset:** 0x0000
**Property:** -

This register allows the user to identify the lowest DMA channel with pending interrupt.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PEND | BUSY | FERR | | | SUSP | TCMPL | TERR |
| Access | R | R | R | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ID[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 15 – PEND** Pending
This bit will read '1' when the channel selected by Channel ID field (ID) is pending.

**Bit 14 – BUSY** Busy
This bit will read '1' when the channel selected by Channel ID field (ID) is busy.

**Bit 13 – FERR** Fetch Error
This bit will read '1' when the channel selected by Channel ID field (ID) fetched an invalid descriptor.

**Bit 10 – SUSP** Channel Suspend
This bit will read '1' when the channel selected by Channel ID field (ID) has pending Suspend interrupt.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel ID (ID) Suspend interrupt flag.

**Bit 9 – TCMPL** Transfer Complete
This bit will read '1' when the channel selected by Channel ID field (ID) has pending Transfer Complete interrupt.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel ID (ID) Transfer Complete interrupt flag.

**Bit 8 – TERR** Transfer Error
This bit is read one when the channel selected by Channel ID field (ID) has pending Transfer Error interrupt.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel ID (ID) Transfer Error interrupt flag.

**Bits 3:0 – ID[3:0]** Channel ID
These bits store the lowest channel number with pending interrupts. The number is valid if Suspend (SUSP), Transfer Complete (TCMPL) or Transfer Error (TERR) bits are set. The Channel ID field is refreshed when a new channel (with channel number less than the current one) with pending interrupts is detected, or when the application clears the corresponding channel interrupt sources. When no pending channels interrupts are available, these bits will always return zero value when read.
When the bits are written, indirect access to the corresponding Channel Interrupt Flag register is enabled.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.11. Interrupt Status

**Name:** INTSTATUS
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CHINT11 | CHINT10 | CHINT9 | CHINT8 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CHINT7 | CHINT6 | CHINT5 | CHINT4 | CHINT3 | CHINT2 | CHINT1 | CHINT0 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – CHINTn** Channel n Pending Interrupt [n=11..0]
This bit is set when Channel n has a pending interrupt/the interrupt request is received.
This bit is cleared when the corresponding Channel n interrupts are disabled or the interrupts sources are cleared.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.12. Busy Channels

**Name:** BUSYCH
**Offset:** 0x28
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | BUSYCH11 | BUSYCH10 | BUSYCH9 | BUSYCH8 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BUSYCH7 | BUSYCH6 | BUSYCH5 | BUSYCH4 | BUSYCH3 | BUSYCH2 | BUSYCH1 | BUSYCH0 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – BUSYCHn** Busy Channel n [x=11..0]
This bit is cleared when the channel trigger action for DMA channel n is complete, when a bus error for DMA channel n is detected, or when DMA channel n is disabled.
This bit is set when DMA channel n starts a DMA transfer.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.13. Pending Channels

**Name:** PENDCH
**Offset:** 0x2C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | PENDCH11 | PENDCH10 | PENDCH9 | PENDCH8 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PENDCH7 | PENDCH6 | PENDCH5 | PENDCH4 | PENDCH3 | PENDCH2 | PENDCH1 | PENDCH0 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 – PENDCH** Pending Channel n [n=11..0]
This bit is cleared when trigger execution defined by channel trigger action settings for DMA channel n is started, when a bus error for DMA channel n is detected or when DMA channel n is disabled. For details on trigger action settings, refer to CHCTRLB.TRIGACT.
This bit is set when a transfer is pending on DMA channel n.

**Related Links**
CHCTRLB
Channel Control B


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.14. Active Channel and Levels

**Name:** ACTIVE
**Offset:** 0x30
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BTCNT[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | BTCNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ABUSY | | | ID[4:0] | | | | |
| Access | R | | | R | R | R | R | R |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LVLEX3 | LVLEX2 | LVLEX1 | LVLEX0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 31:16 – BTCNT[15:0]** Active Channel Block Transfer Count
These bits hold the 16-bit block transfer count of the ongoing transfer. This value is stored in the active channel and written back in the corresponding Write-Back channel memory location when the arbiter grants a new channel access. The value is valid only when the active channel Active Busy flag (ABUSY) is set.

**Bit 15 – ABUSY** Active Channel Busy
This bit is cleared when the active transfer count is written back in the write-back memory section.
This bit is set when the next descriptor transfer count is read from the write-back memory section.

**Bits 12:8 – ID[4:0]** Active Channel ID
These bits hold the channel index currently stored in the active channel registers. The value is updated each time the arbiter grants a new channel transfer access request.

**Bits 0, 1, 2, 3 – LVLEXx** Level x Channel Trigger Request Executing [x=3..0]
This bit is set when a level-x channel trigger request is executing or pending.
This bit is cleared when no request is pending or being executed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.15. Descriptor Memory Section Base Address

**Name:** BASEADDR
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** PAC Write Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BASEADDR[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | BASEADDR[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | BASEADDR[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BASEADDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – BASEADDR[31:0]** Descriptor Memory Base Address
These bits store the Descriptor memory section base address. The value must be 64-bit aligned.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.16. Write-Back Memory Section Base Address

**Name:** WRBADDR
**Offset:** 0x38
**Reset:** 0x00000000
**Property:** PAC Write Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | WRBADDR[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WRBADDR[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WRBADDR[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WRBADDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – WRBADDR[31:0]** Write-Back Memory Base Address
These bits store the Write-Back memory base address. The value must be 64-bit aligned.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.17. Channel ID

**Name:** CHID
**Offset:** 0x3F
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ID[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – ID[3:0]** Channel ID
These bits define the channel number that will be affected by the channel registers (CH*). Before reading or writing a channel register, the channel ID bit group must be written first.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.18. Channel Control A

**Name:** CHCTRLA
**Offset:** 0x40
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | | | | ENABLE | SWRST |
| Access | R | R/W | R | R | R | R | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 6 – RUNSTDBY** Channel run in standby
This bit is used to keep the DMAC channel running in standby mode.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The DMAC channel is halted in standby. |
| 1 | The DMAC channel continues to run in standby. |

**Bit 1 – ENABLE** Channel Enable
Writing a '0' to this bit during an ongoing transfer, the bit will not be cleared until the internal data transfer buffer is empty and the DMA transfer is aborted. The internal data transfer buffer will be empty once the ongoing burst transfer is completed.
Writing a '1' to this bit will enable the DMA channel.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | DMA channel is disabled. |
| 1 | DMA channel is enabled. |

**Bit 0 – SWRST** Channel Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets the channel registers to their initial state. The bit can be set when the channel is disabled (ENABLE=0). Writing a '1' to this bit will be ignored as long as ENABLE=1. This bit is automatically cleared when the reset is completed.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.19. Channel Control B

**Name:** CHCTRLB
**Offset:** 0x44
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | CMD[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TRIGACT[1:0] | | | | | | | |
| Access | R/W | R/W | | | | | | |
| Reset | 0 | 0 | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | TRIGSRC[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | LVL[1:0] | | EVOE | EVIE | EVACT[2:0] | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 25:24 – CMD[1:0]** Software Command
These bits define the software commands. Refer to Channel Suspend and Channel Resume and Next Suspend Skip.
These bits are not enable-protected.

| CMD[1:0] | Name | Description |
|---|---|---|
| 0x0 | NOACT | No action |
| 0x1 | SUSPEND | Channel suspend operation |
| 0x2 | RESUME | Channel resume operation |
| 0x3 | - | Reserved |

**Bits 23:22 – TRIGACT[1:0]** Trigger Action
These bits define the trigger action used for a transfer.

| TRIGACT[1:0] | Name | Description |
|---|---|---|
| 0x0 | BLOCK | One trigger required for each block transfer |
| 0x1 | - | Reserved |
| 0x2 | BEAT | One trigger required for each beat transfer |
| 0x3 | TRANSACTION | One trigger required for each transaction |

**Bits 13:8 – TRIGSRC[5:0]** Trigger Source
These bits define the peripheral trigger which is source of the transfer. For details on trigger selection and trigger modes, refer to Transfer Triggers and Actions and CHCTRLB.TRIGACT.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 25-2.** Peripheral Trigger Source

| Value | Name | Description |
|---|---|---|
| 0x00 | DISABLE | Only software/event triggers |
| 0x01 | TSENS | TSENS Result Ready Trigger |
| 0x02 | SERCOM0 RX | SERCOM0 RX Trigger |
| 0x03 | SERCOM0 TX | SERCOM0TX Trigger |
| 0x04 | SERCOM1 RX | SERCOM1 RX Trigger |
| 0x05 | SERCOM1 TX | SERCOM1 TX Trigger |
| 0x06 | SERCOM2 RX | SERCOM2 RX Trigger |
| 0x07 | SERCOM2 TX | SERCOM2 TX Trigger |
| 0x08 | SERCOM3 RX | SERCOM3 RX Trigger |
| 0x09 | SERCOM3 TX | SERCOM3 TX Trigger |
| 0x0A | SERCOM4 RX | SERCOM4 RX Trigger |
| 0x0B | SERCOM4 TX | SERCOM4 TX Trigger |
| 0x0C | SERCOM5 RX | SERCOM5 RX Trigger |
| 0x0D | SERCOM5 TX | SERCOM5 TX Trigger |
| 0x0E | CAN0 DEBUG | CAN0 Debug Trigger |
| 0x0F | CAN1 DEBUG | CAN1 Debug Trigger |
| 0x10 | TCC0 OVF | TCC0 Overflow Trigger |
| 0x11 | TCC0 MC0 | TCC0 Match/Compare 0 Trigger |
| 0x12 | TCC0 MC1 | TCC0 Match/Compare 1 Trigger |
| 0x13 | TCC0 MC2 | TCC0 Match/Compare 2 Trigger |
| 0x14 | TCC0 MC3 | TCC0 Match/Compare 3 Trigger |
| 0x15 | TCC1 OVF | TCC1 Overflow Trigger |
| 0x16 | TCC1 MC0 | TCC1 Match/Compare 0 Trigger |
| 0x17 | TCC1 MC1 | TCC1 Match/Compare 1 Trigger |
| 0x18 | TCC2 OVF | TCC2 Overflow Trigger |
| 0x19 | TCC2 MC0 | TCC2 Match/Compare 0 Trigger |
| 0x1A | TCC2 MC1 | TCC2 Match/Compare 1 Trigger |
| 0x1B | TC0 OVF | TC0 Overflow Trigger |
| 0x1C | TC0 MC0 | TC0 Match/Compare 0 Trigger |
| 0x1D | TC0 MC1 | TC0 Match/Compare 1 Trigger |
| 0x1E | TC1 OVF | TC1 Overflow Trigger |
| 0x1F | TC1 MC0 | TC1 Match/Compare 0 Trigger |
| 0x20 | TC1 MC1 | TC1 Match/Compare 1 Trigger |
| 0x21 | TC2 OVF | TC2 Overflow Trigger |
| 0x22 | TC2 MC0 | TC2 Match/Compare 0 Trigger |
| 0x23 | TC2 MC1 | TC2 Match/Compare 1 Trigger |
| 0x24 | TC3 OVF | TC3 Overflow Trigger |
| 0x25 | TC3 MC0 | TC3 Match/Compare 0 Trigger |
| 0x26 | TC3 MC1 | TC3 Match/Compare 1 Trigger |
| 0x27 | TC4 OVF | TC4 Overflow Trigger |
| 0x28 | TC4 MC0 | TC4 Match/Compare 0 Trigger |
| 0x29 | TC4 MC1 | TC4 Match/Compare 1 Trigger |
| 0x2A | ADC0 RESRDY | ADC0 Result Ready Trigger |
| 0x2B | ADC1 RESRDY | ADC1 Result Ready Trigger |
| 0x2C | SDADC RESRDY | SDADC Result Ready Trigger |
| 0x2D | DAC EMPTY | DAC Empty Trigger |
| 0x2E | PTC EOC | PTC End of Conversion Trigger |
| 0x2F | PTC WCOMP | PTC Window Compare Trigger |
| 0x30 | PTC SEQ | PTC Sequence Trigger |
| 0x31 | SERCOM6 RX | SERCOM6 RX Trigger |
| 0x32 | SERCOM6 TX | SERCOM6 TX Trigger |
| 0x33 | SERCOM7 RX | SERCOM6 RX Trigger |
| 0x34 | SERCOM7 TX | SERCOM6 TX Trigger |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 25-2.** Peripheral Trigger Source (continued)

| Value | Name | Description |
|---|---|---|
| 0x35 | TC5 OVF | TC5 Overflow Trigger |
| 0x36 | TC5 MC0 | TC5 Match/Compare 0 Trigger |
| 0x37 | TC5 MC1 | TC5 Match/Compare 1 Trigger |
| 0x38 | TC6 OVF | TC6 Overflow Trigger |
| 0x39 | TC6 MC0 | TC6 Match/Compare 0 Trigger |
| 0x3A | TC6 MC1 | TC6 Match/Compare 1 Trigger |
| 0x3B | TC7 OVF | TC7 Overflow Trigger |
| 0x3C | TC7 MC0 | TC7 Match/Compare 0 Trigger |
| 0x3D | TC7 MC1 | TC7 Match/Compare 1 Trigger |

**Bits 6:5 – LVL[1:0]** Channel Arbitration Level

These bits define the arbitration level used for the DMA channel, where a high level has priority over a low level. For further details on arbitration schemes, refer to Arbitration.
These bits are not enable-protected.

| TRIGACT[1:0] | Name | Description |
|---|---|---|
| 0x0 | LVL0 | Channel Priority Level 0 |
| 0x1 | LVL1 | Channel Priority Level 1 |
| 0x2 | LVL2 | Channel Priority Level 2 |
| 0x3 | LVL3 | Channel Priority Level 3 |

**Bit 4 – EVOE** Channel Event Output Enable

This bit indicates if the Channel event generation is enabled. The event will be generated for every condition defined in the descriptor Event Output Selection (BTCTRL.EVOSEL).
This bit is available only for the least significant DMA channels. Refer to table: *User Multiplexer Selection* and *Event Generator Selection* of the Event System for details.

| Value | Description |
|---|---|
| 0 | Channel event generation is disabled. |
| 1 | Channel event generation is enabled. |

**Bit 3 – EVIE** Channel Event Input Enable

This bit is available only for the least significant DMA channels. Refer to table: *User Multiplexer Selection* and *Event Generator Selection* of the Event System for details.

| Value | Description |
|---|---|
| 0 | Channel event action will not be executed on any incoming event. |
| 1 | Channel event action will be executed on any incoming event. |

**Bits 2:0 – EVACT[2:0]** Event Input Action

These bits define the event input action, as shown below. The action is executed only if the corresponding EVIE bit in CHCTRLB register of the channel is set.
These bits are available only for the least significant DMA channels. Refer to table: *User Multiplexer Selection* and *Event Generator Selection* of the Event System for details.

| EVACT[2:0] | Name | Description |
|---|---|---|
| 0x0 | NOACT | No action |
| 0x1 | TRIG | Normal Transfer and Conditional Transfer on Strobe trigger |
| 0x2 | CTRIG | Conditional transfer trigger |
| 0x3 | CBLOCK | Conditional block transfer |
| 0x4 | SUSPEND | Channel suspend operation |
| 0x5 | RESUME | Channel resume operation |
| 0x6 | SSKIP | Skip next block suspend action |
| 0x7 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**

Channel

CHANNELn

Channel n Control


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.20. Channel Interrupt Enable Clear

**Name:** CHINTENCLR
**Offset:** 0x4C
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Channel Interrupt Enable Set (CHINTENSET) register.
This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SUSP | TCMPL | TERR |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – SUSP** Channel Suspend Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel Suspend Interrupt Enable bit, which disables the Channel Suspend interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Suspend interrupt is disabled. |
| 1 | The Channel Suspend interrupt is enabled. |

**Bit 1 – TCMPL** Channel Transfer Complete Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel Transfer Complete Interrupt Enable bit, which disables the Channel Transfer Complete interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Transfer Complete interrupt is disabled. When block action is set to none, the TCMPL flag will not be set when a block transfer is completed. |
| 1 | The Channel Transfer Complete interrupt is enabled. |

**Bit 0 – TERR** Channel Transfer Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel Transfer Error Interrupt Enable bit, which disables the Channel Transfer Error interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Transfer Error interrupt is disabled. |
| 1 | The Channel Transfer Error interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.21. Channel Interrupt Enable Set

**Name:** CHINTENSET
**Offset:** 0x4D
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Channel Interrupt Enable Clear (CHINTENCLR) register.
This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SUSP | TCMPL | TERR |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – SUSP** Channel Suspend Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Channel Suspend Interrupt Enable bit, which enables the Channel Suspend interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Suspend interrupt is disabled. |
| 1 | The Channel Suspend interrupt is enabled. |

**Bit 1 – TCMPL** Channel Transfer Complete Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Channel Transfer Complete Interrupt Enable bit, which enables the Channel Transfer Complete interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Transfer Complete interrupt is disabled. |
| 1 | The Channel Transfer Complete interrupt is enabled. |

**Bit 0 – TERR** Channel Transfer Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Channel Transfer Error Interrupt Enable bit, which enables the Channel Transfer Error interrupt.

| Value | Description |
|---|---|
| 0 | The Channel Transfer Error interrupt is disabled. |
| 1 | The Channel Transfer Error interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.22. Channel Interrupt Flag Status and Clear

**Name:** CHINTFLAG
**Offset:** 0x4E
**Reset:** 0x00
**Property:** -

This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SUSP | TCMPL | TERR |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – SUSP** Channel Suspend
This flag is cleared by writing a '1' to it.
This flag is set when a block transfer with suspend block action is completed, when a software suspend command is executed, when a suspend event is received or when an invalid descriptor is fetched by the DMA.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Channel Suspend interrupt flag for the corresponding channel.
For details on available software commands, refer to CHCTRLB.CMD.
For details on available event input actions, refer to CHCTRLB.EVACT.
For details on available block actions, refer to BTCTRL.BLOCKACT.

**Bit 1 – TCMPL** Channel Transfer Complete
This flag is cleared by writing a '1' to it.
This flag is set when a block transfer is completed and the corresponding interrupt block action is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Transfer Complete interrupt flag for the corresponding channel.

**Bit 0 – TERR** Channel Transfer Error
This flag is cleared by writing a '1' to it.
This flag is set when a bus error is detected during a beat transfer or when the DMAC fetches an invalid descriptor.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Transfer Error interrupt flag for the corresponding channel.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.8.23. Channel Status

**Name:** CHSTATUS
**Offset:** 0x4F
**Reset:** 0x00
**Property:** -

This register affects the DMA channel that is selected in the Channel ID register (CHID.ID).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | FERR | BUSY | PEND |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – FERR** Channel Fetch Error
This bit is cleared when a software resume command is executed.
This bit is set when an invalid descriptor is fetched.

**Bit 1 – BUSY** Channel Busy
This bit is cleared when the channel trigger action is completed, when a bus error is detected or when the channel is disabled.
This bit is set when the DMA channel starts a DMA transfer.

**Bit 0 – PEND** Channel Pending
This bit is cleared when the channel trigger action is started, when a bus error is detected or when the channel is disabled. For details on trigger action settings, refer to CHCTRLB.TRIGACT.
This bit is set when a transfer is pending on the DMA channel, as soon as the transfer request is received.


© 2026 Microchip Technology Inc. and its subsidiaries

## 25.9. Register Summary - SRAM

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | BTCTRL | 7:0 | | | | BLOCKACT[1:0] | | EVOSEL[1:0] | | VALID |
| | | 15:8 | STEPSIZE[2:0] | | | STEPSEL | DSTINC | SRCINC | BEATSIZE[1:0] | |
| 0x02 | BTCNT | 7:0 | BTCNT[7:0] | | | | | | | |
| | | 15:8 | BTCNT[15:8] | | | | | | | |
| 0x04 | SRCADDR | 7:0 | SRCADDR[7:0] | | | | | | | |
| | | 15:8 | SRCADDR[15:8] | | | | | | | |
| | | 23:16 | SRCADDR[23:16] | | | | | | | |
| | | 31:24 | SRCADDR[31:24] | | | | | | | |
| 0x08 | DSTADDR | 7:0 | DSTADDR[7:0] | | | | | | | |
| | | 15:8 | DSTADDR[15:8] | | | | | | | |
| | | 23:16 | DSTADDR[23:16] | | | | | | | |
| | | 31:24 | DSTADDR[31:24] | | | | | | | |
| 0x0C | DESCADDR | 7:0 | DESCADDR[7:0] | | | | | | | |
| | | 15:8 | DESCADDR[15:8] | | | | | | | |
| | | 23:16 | DESCADDR[23:16] | | | | | | | |
| | | 31:24 | DESCADDR[31:24] | | | | | | | |

## 25.10. Register Description - SRAM

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.10.1. Block Transfer Control

**Name:** BTCTRL
**Offset:** 0x00
**Property:** -

The BTCTRL register offset is relative to (BASEADDR or WRBADDR) + Channel Number * 0x10

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | STEPSIZE[2:0] | | | STEPSEL | DSTINC | SRCINC | BEATSIZE[1:0] | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | BLOCKACT[1:0] | | EVOSEL[1:0] | | VALID |
| Access | | | | - | - | - | - | - |
| Reset | | | | | | | | |

**Bits 15:13 – STEPSIZE[2:0]** Address Increment Step Size
These bits select the address increment step size. The setting apply to source or destination address, depending on STEPSEL setting.

| Value | Name | Description |
|---|---|---|
| 0x0 | X1 | Next ADDR = ADDR + (Beat size in byte) * 1 |
| 0x1 | X2 | Next ADDR = ADDR + (Beat size in byte) * 2 |
| 0x2 | X4 | Next ADDR = ADDR + (Beat size in byte) * 4 |
| 0x3 | X8 | Next ADDR = ADDR + (Beat size in byte) * 8 |
| 0x4 | X16 | Next ADDR = ADDR + (Beat size in byte) * 16 |
| 0x5 | X32 | Next ADDR = ADDR + (Beat size in byte) * 32 |
| 0x6 | X64 | Next ADDR = ADDR + (Beat size in byte) * 64 |
| 0x7 | X128 | Next ADDR = ADDR + (Beat size in byte) * 128 |

**Bit 12 – STEPSEL** Step Selection
This bit selects if source or destination addresses are using the step size settings.

| Value | Name | Description |
|---|---|---|
| 0x0 | DST | Step size settings apply to the destination address |
| 0x1 | SRC | Step size settings apply to the source address |

**Bit 11 – DSTINC** Destination Address Increment Enable
Writing a '0' to this bit will disable the destination address incrementation. The address will be kept fixed during the data transfer.
Writing a '1' to this bit will enable the destination address incrementation. By default, the destination address is incremented by 1. If the STEPSEL bit is cleared, flexible step-size settings are available in the STEPSIZE register.

| Value | Description |
|---|---|
| 0 | The Destination Address Increment is disabled. |
| 1 | The Destination Address Increment is enabled. |

**Bit 10 – SRCINC** Source Address Increment Enable
Writing a '0' to this bit will disable the source address incrementation. The address will be kept fixed during the data transfer.
Writing a '1' to this bit will enable the source address incrementation. By default, the source address is incremented by 1. If the STEPSEL bit is set, flexible step-size settings are available in the STEPSIZE register.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The Source Address Increment is disabled. |
| 1 | The Source Address Increment is enabled. |

**Bits 9:8 – BEATSIZE[1:0]** Beat Size

These bits define the size of one beat. A beat is the size of one data transfer bus access, and the setting apply to both read and write accesses.

| Value | Name | Description |
|---|---|---|
| 0x0 | BYTE | 8-bit bus transfer |
| 0x1 | HWORD | 16-bit bus transfer |
| 0x2 | WORD | 32-bit bus transfer |
| other | | Reserved |

**Bits 4:3 – BLOCKACT[1:0]** Block Action

These bits define what actions the DMAC should take after a block transfer has completed.

| BLOCKACT[1:0] | Name | Description |
|---|---|---|
| 0x0 | NOACT | Channel will be disabled if it is the last block transfer in the transaction |
| 0x1 | INT | Channel will be disabled if it is the last block transfer in the transaction and block interrupt |
| 0x2 | SUSPEND | Channel suspend operation is completed |
| 0x3 | BOTH | Both channel suspend operation and block interrupt |

**Bits 2:1 – EVOSEL[1:0]** Event Output Selection

These bits define the event output selection.

| EVOSEL[1:0] | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Event generation disabled |
| 0x1 | BLOCK | Event strobe when block transfer complete |
| 0x2 | | Reserved |
| 0x3 | BEAT | Event strobe when beat transfer complete |

**Bit 0 – VALID** Descriptor Valid

Writing a '0' to this bit in the Descriptor or Write-Back memory will suspend the DMA channel operation when fetching the corresponding descriptor.
The bit is automatically cleared in the Write-Back memory section when channel is aborted, when an error is detected during the block transfer, or when the block transfer is completed.

| Value | Description |
|---|---|
| 0 | The descriptor is not valid. |
| 1 | The descriptor is valid. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.10.2. Block Transfer Count

**Name:** BTCNT
**Offset:** 0x02
**Property:** -

The BTCNT register offset is relative to (BASEADDR or WRBADDR) + Channel Number * 0x10

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | BTCNT[15:8] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BTCNT[7:0] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

**Bits 15:0 – BTCNT[15:0]** Block Transfer Count
This bit group holds the 16-bit block transfer count.
During a transfer, the internal counter value is decremented by one after each beat transfer. The internal counter is written to the corresponding write-back memory section for the DMA channel when the DMA channel loses priority, is suspended or gets disabled. The DMA channel can be disabled by a complete transfer, a transfer error or by software.


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.10.3. Block Transfer Source Address

**Name:** SRCADDR
**Offset:** 0x04
**Property:** -

The SRCADDR register offset is relative to (BASEADDR or WRBADDR) + Channel Number * 0x10

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SRCADDR[31:24] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SRCADDR[23:16] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SRCADDR[15:8] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SRCADDR[7:0] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

**Bits 31:0 – SRCADDR[31:0]** Transfer Source Address

This bit field holds the block transfer source address.
When source address incrementation is disabled (BTCTRL.SRCINC=0), SRCADDR corresponds to the last beat transfer address in the block transfer.
When source address incrementation is enabled (BTCTRL.SRCINC=1), SRCADDR is calculated as follows:
If BTCTRL.STEPSEL=1:

$$\mathrm{SRCADDR} = \mathrm{SRCADDR}_{START} + BTCNT \cdot \left(BEATSIZE + 1\right) \cdot 2^{\mathrm{STEPSIZE}}$$

If BTCTRL.STEPSEL=0:

$$\mathrm{SRCADDR} = \mathrm{SRCADDR}_{START} + BTCNT \cdot \left(BEATSIZE + 1\right)$$

- $\mathrm{SRCADDR}_{START}$ is the source address of the first beat transfer in the block transfer
- BTCNT is the initial number of beats remaining in the block transfer
- BEATSIZE is the configured number of bytes in a beat
- STEPSIZE is the configured number of beats for each incrementation


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.10.4. Block Transfer Destination Address

**Name:** DSTADDR
**Offset:** 0x08
**Property:** -

The DSTADDR register offset is relative to (BASEADDR or WRBADDR) + Channel Number * 0x10

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DSTADDR[31:24] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DSTADDR[23:16] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DSTADDR[15:8] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DSTADDR[7:0] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

**Bits 31:0 – DSTADDR[31:0]** Transfer Destination Address

This bit field holds the block transfer destination address.
When destination address incrementation is disabled (BTCTRL.DSTINC=0), DSTADDR corresponds to the last beat transfer address in the block transfer.
When destination address incrementation is enabled (BTCTRL.DSTINC=1), DSTADDR is calculated as follows:
If BTCTRL.STEPSEL=1:

$$DSTADDR = DSTADDR_{START} + BTCNT \bullet (BEATSIZE + 1)$$

If BTCTRL.STEPSEL=0:

$$DSTADDR = DSTADDR_{START} + BTCNT \bullet \left(BEATSIZE + 1\right) \bullet 2^{STEPSIZE}$$

- $DSTADDR_{START}$ is the destination address of the first beat transfer in the block transfer
- BTCNT is the initial number of beats remaining in the block transfer
- BEATSIZE is the configured number of bytes in a beat
- STEPSIZE is the configured number of beats for each incrementation


© 2026 Microchip Technology Inc. and its subsidiaries

### 25.10.5. Next Descriptor Address

**Name:** DESCADDR
**Offset:** 0x0C
**Property:** -

The DESCADDR register offset is relative to (BASEADDR or WRBADDR) + Channel Number * 0x10

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DESCADDR[31:24] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DESCADDR[23:16] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DESCADDR[15:8] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DESCADDR[7:0] | | | | | | | |
| Access | - | - | - | - | - | - | - | - |
| Reset | | | | | | | | |

**Bits 31:0 – DESCADDR[31:0]** Next Descriptor Address
This bit group holds the SRAM address of the next descriptor. The value must be 64-bit aligned. If the value of this SRAM register is 0x00000000, the transaction will be terminated when the DMAC tries to load the next transfer descriptor.


© 2026 Microchip Technology Inc. and its subsidiaries

# 26. EIC – External Interrupt Controller

## 26.1. Overview

The External Interrupt Controller (EIC) allows external pins to be configured as interrupt lines. Each interrupt line can be individually masked and can generate an interrupt on rising, falling, or both edges, or on high or low levels. Each external pin has a configurable filter to remove spikes. Each external pin can also be configured to be asynchronous in order to wake up the device from sleep modes where all clocks have been disabled. External pins can also generate an event.

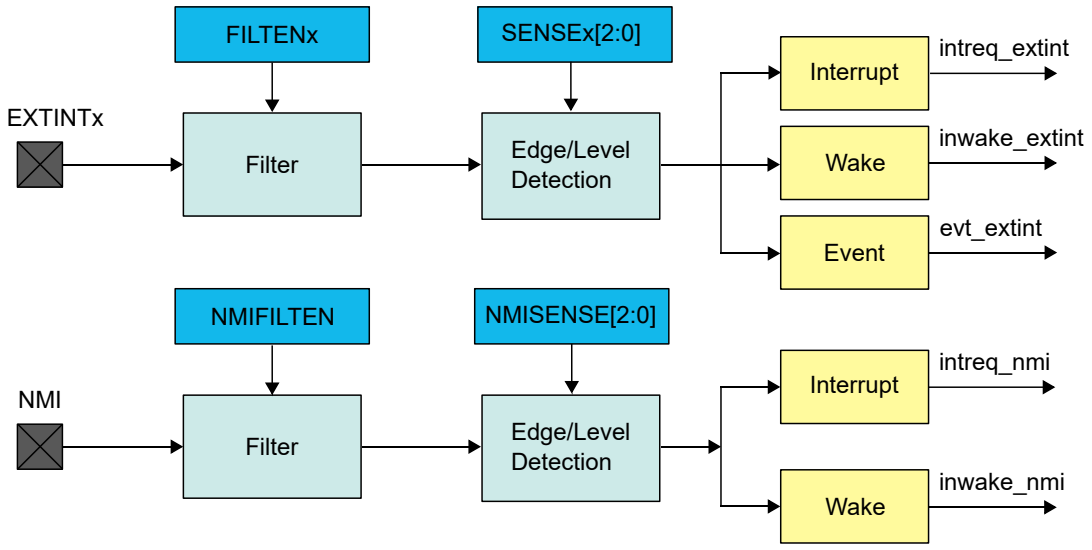
A separate non-maskable interrupt (NMI) is also supported. It has properties similar to the other external interrupts, but is connected to the NMI request of the CPU, enabling it to interrupt any other interrupt mode.

## 26.2. Features

- Up to 16 external pins (EXTINTx), plus one non-maskable pin (NMI)
- Dedicated, Individually Maskable Interrupt for Each Pin
- Interrupt on Rising, Falling, or Both Edges
- Synchronous or Asynchronous Edge Detection mode
- Interrupt pin Debouncing (Only available on SAM C20/C21 N variants)
- Interrupt on High or Low Levels
- Asynchronous Interrupts for Sleep Modes Without Clock
- Filtering of External Pins
- Event Generation from EXTINTx

## 26.3. Block Diagram

**Figure 26-1.** EIC Block Diagram




## 26.4. Signal Description

| Signal Name | Type | Description |
|---|---|---|
| EXTINT[15..0] | Digital Input | External interrupt pin |
| NMI | Digital Input | Non-maskable interrupt pin |


© 2026 Microchip Technology Inc. and its subsidiaries

One signal may be available on several pins.

**Related Links**
I/O Multiplexing and Considerations

## 26.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 26.5.1. I/O Lines

Using the EIC's I/O lines requires the I/O pins to be configured.

**Related Links**
PORT - I/O Pin Controller

### 26.5.2. Power Management

All interrupts are available down to STANDBY sleep mode, but the EIC can be configured to automatically mask some interrupts in order to prevent device wake-up.

The EIC will continue to operate in any sleep mode where the selected source clock is running. The EIC's interrupts can be used to wake up the device from sleep modes. Events connected to the Event System can trigger other operations in the system without exiting sleep modes.

**Related Links**
PM - Power Manager

### 26.5.3. Clocks

The EIC bus clock (CLK_EIC_APB) can be enabled and disabled by the Main Clock Controller, the default state of CLK_EIC_APB can be found in the Peripheral Clock Masking section.

Some optional functions need a peripheral clock, which can either be a generic clock (GCLK_EIC, for wider frequency selection) or a Ultra Low-Power 32.768 kHz clock (CLK_ULP32K, for highest power efficiency). One of the clock sources must be configured and enabled before using the peripheral:

GCLK_EIC is configured and enabled in the Generic Clock Controller (GCLK).

CLK_ULP32K is provided by the internal Ultra Low-Power (OSCULP32K) Oscillator in the OSC32KCTRL module.

Both GCLK_EIC and CLK_ULP32K are asynchronous to the user interface clock (CLK_EIC_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
MCLK – Main Clock
Peripheral Clock Masking
GCLK - Generic Clock Controller
OSC32KCTRL – 32.768 kHz Oscillators Controller

### 26.5.4. DMA

Not applicable.

### 26.5.5. Interrupts

There are several interrupt request lines, some (the number depends on the product variant) for the external interrupts (EXTINT) and one for Non-Maskable Interrupt (NMI).

Each EXTINT interrupt request line is connected to the interrupt controller. Using the EIC interrupt requires the interrupt controller to be configured first.


© 2026 Microchip Technology Inc. and its subsidiaries

The NMI interrupt request line is connected to the interrupt controller, but does not require the interrupt to be configured.

**Related Links**
Nested Vector Interrupt Controller

### 26.5.6. Events

The events are connected to the Event System. Using the events requires the Event System to be configured first.

**Related Links**
Event System (EVSYS)

### 26.5.7. Debug Operation

When the CPU is halted in debug mode, the EIC continues normal operation. If the EIC is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 26.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Status and Clear register (INTFLAG)
- Non-Maskable Interrupt Flag Status and Clear register (NMIFLAG)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 26.5.9. Analog Connections

Not applicable.

## 26.6. Functional Description

### 26.6.1. Principle of Operation

The EIC detects edge or level condition to generate interrupts to the CPU interrupt controller or events to the Event System. Each external interrupt pin (EXTINT) can be filtered using majority vote filtering, clocked by GCLK_EIC or by CLK_ULP32K.

**Related Links**
External Pin Processing

### 26.6.2. Basic Operation

#### 26.6.2.1. Initialization

The EIC must be initialized in the following order:

1. Enable CLK_EIC_APB.
2. If required, configure the NMI by writing the Non-Maskable Interrupt Control register (NMICTRL).
3. Enable GCLK_EIC or CLK_ULP32K when one of the following configuration is selected:
   - The NMI uses edge detection or filtering
   - One or more EXTINT uses filtering


© 2026 Microchip Technology Inc. and its subsidiaries

- One or more EXTINT uses synchronous edge detection
- One or more EXTINT uses debouncing (Only available on SAM C20/C21 N variants)

GCLK_EIC is used when a frequency higher than 32.768 kHz is required for filtering.

CLK_ULP32K is recommended when power consumption is the priority. For CLK_ULP32K write a '1' to the Clock Selection bit in the Control A register (CTRLA.CKSEL).

4. Configure the EIC input sense and filtering by writing the configuration register (CONFIG0 or CONFIG1).
5. Optionally, enable the asynchronous mode.
6. Optionally, enable the debouncer mode (Only available on SAM C20/C21 N variants).
7. Enable the EIC by writing a '1' to CTRLA.ENABLE.

The following bits are enable-protected, that is, it can only be written when the EIC is disabled (CTRLA.ENABLE = 0):

- Clock Selection bit in the Control A register (CTRLA.CKSEL)

The following registers are enable-protected:

- Event Control register (EVCTRL)
- Configuration register (CONFIGn)
- External Interrupt Asynchronous Mode register (ASYNCH)
- Debouncer Enable register (DEBOUNCEN)
- Debounce Prescaler register (DPRESCALER)

Enable-protected bits in the CTRLA register can be written simultaneously when setting CTRLA.ENABLE to '1', but not at the same time as CTRLA.ENABLE is being cleared.

Enable-protection is denoted by the "Enable-Protected" property in the register description.

**Related Links**

CONFIGn

External Interrupt Sense Configuration n

#### 26.6.2.2. Enabling, Disabling, and Resetting

The EIC is enabled by writing a '1' the Enable bit in the Control A register (CTRLA.ENABLE). The EIC is disabled by writing CTRLA.ENABLE to '0'.

The EIC is reset by setting the Software Reset bit in the Control register (CTRLA.SWRST). All registers in the EIC will be reset to their initial state, and the EIC will be disabled.

Refer to the CTRLA register description for details.

### 26.6.3. External Pin Processing

Each external pin can be configured to generate an interrupt/event on edge detection (rising, falling or both edges) or level detection (high or low). The sense of external interrupt pins is configured by writing the Input Sense x bits in the Configuration register (CONFIGn.SENSEx). The corresponding interrupt flag (INTFLAG.EXTINT[x]) in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition is met.

When the interrupt flag has been cleared in edge-sensitive mode, INTFLAG.EXTINT[x] will only be set if a new interrupt condition is met.

In level-sensitive mode, when interrupt has been cleared, INTFLAG.EXTINT[x] will be set immediately if the EXTINTx pin still matches the interrupt condition.

Each external pin can be filtered by a majority vote filtering, clocked by GCLK_EIC or CLK_ULP32K. Filtering is enabled if bit Filter Enable x in the configuration register (CONFIGn.FILTENx) is written to


© 2026 Microchip Technology Inc. and its subsidiaries

'1'. The majority vote filter samples the external pin three times with GCLK_EIC or CLK_ULP32K and outputs the value when two or more samples are equal.

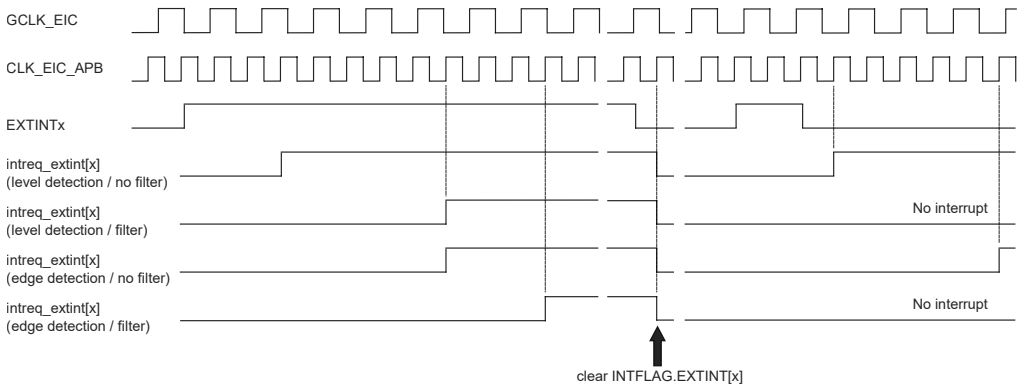
**Table 26-1.** Majority Vote Filter Logic

| Samples [0, 1, 2] | Filter Output |
|---|---|
| [0,0,0] | 0 |
| [0,0,1] | 0 |
| [0,1,0] | 0 |
| [0,1,1] | 1 |
| [1,0,0] | 0 |
| [1,0,1] | 1 |
| [1,1,0] | 1 |
| [1,1,1] | 1 |

When an external interrupt is configured for level detection and when filtering is disabled, detection is done asynchronously. Level detection and asynchronous edge detection do not require GCLK_EIC or CLK_ULP32K, and can generate asynchronous interrupts and events.

If filtering, synchronous edge detection, or debouncing is enabled, the EIC automatically requests GCLK_EIC or CLK_ULP32K to operate. The selection between these two clocks is done by writing the Clock Selection bits in the Control A register (CTRLA.CKSEL). GCLK_EIC must be enabled in the GCLK module. In these modes the external pin is sampled at the EIC clock rate, thus pulses with duration lower than four EIC clock periods may not be properly detected.

**Figure 26-2.** Interrupt Detection Latency by modes (Rising Edge)




Detection latency depends on the detection mode.

**Table 26-2.** Detection Latency

| Detection mode | Latency (worst case) |
|---|---|
| Level without filter (Asynchronous) | Five CLK_EIC_APB periods |
| Level with filter (Synchronous) | Four GCLK_EIC/CLK_ULP32K periods + five CLK_EIC_APB periods |
| Edge without filter (Asynchronous) | Five CLK_EIC_APB periods |
| Edge without filter (Synchronous) | Four GCLK_EIC/CLK_ULP32K periods + five CLK_EIC_APB periods |
| Edge with filter (Synchronous) | Six GCLK_EIC/CLK_ULP32K periods + five CLK_EIC_APB periods |

## Related Links

GCLK - Generic Clock Controller

CONFIGn

External Interrupt Sense Configuration n


© 2026 Microchip Technology Inc. and its subsidiaries

## 26.6.4. Additional Features

### 26.6.4.1. Non-Maskable Interrupt (NMI)

The non-maskable interrupt pin can also generate an interrupt on edge or level detection, but it is configured with the dedicated NMI Control register (NMICTRL). To select the sense for NMI, write to the NMISENSE bit group in the NMI Control register (NMICTRL.NMISENSE). NMI filtering is enabled by writing a '1' to the NMI Filter Enable bit (NMICTRL.NMIFILTEN).

If edge detection or filtering is required, enable GCLK_EIC or CLK_ULP32K.

NMI detection is enabled only by the NMICTRL.NMISENSE value, and the EIC module is not required to be enabled.

When an NMI is detected, the Non-maskable Interrupt flag in the NMI Flag Status and Clear register is set (NMIFLAG.NMI). NMI interrupt generation is always enabled, and NMIFLAG.NMI generates an interrupt request when set.

### 26.6.4.2. Asynchronous Edge Detection Mode (No Debouncing)

The EXTINT edge detection operates synchronously or asynchronously, as selected by the Asynchronous Control Mode bit for external pin x in the External Interrupt Asynchronous Mode register (ASYNCH.ASYNCH[x]). The EIC edge detection is operated synchronously when the Asynchronous Control Mode bit (ASYNCH.ASYNCH[x]) is '0' (default value). It is operated asynchronously when ASYNCH.ASYNCH[x] is written to '1'.

In *Synchronous Edge Detection Mode*, the external interrupt (EXTINT) or the non-maskable interrupt (NMI) pins are sampled using the EIC clock as defined by the Clock Selection bit in the Control A register (CTRLA.CKSEL). The External Interrupt flag (INTFLAG.EXTINT[x]) or Non-Maskable Interrupt flag (NMIFLAG.NMI) is set when the last sampled state of the pin differs from the previously sampled state. The EIC clock is needed in this mode.

The Synchronous Edge Detection Mode can be used in Idle and Standby sleep modes.

In *Asynchronous Edge Detection Mode*, the external interrupt (EXTINT) pins or the non-maskable interrupt (NMI) pins set the External Interrupt flag or Non-Maskable Interrupt flag (INTFLAG.EXTINT[x] or NMIFLAG) directly. The EIC clock is not needed in this mode.

The asynchronous edge detection mode can be used in Idle and Standby sleep modes.

### 26.6.4.3. Interrupt Pin Debouncing (only available on SAM C20/C21 N variants)

The external interrupt pin (EXTINT) edge detection can use a debouncer to improve input noise immunity. When selected, the debouncer can work in the synchronous mode or the asynchronous mode, depending on the configuration of the ASYNCH.ASYNCH[x] bit for the pin. The debouncer uses the EIC clock as defined by the CTRLA.CKSEL bit to clock the debouncing circuitry. The debouncing time frame is set with the debouncer prescaler DPRESCALER.PRESCALERm, which provides the *low-frequency clock* tick that is used to reject higher frequency signals.

The Debouncing mode for the EXTINT x pin can be selected only if the Sense bits in the configuration register (CONFIGn.SENSEx) are set to RISE, FALL or BOTH. If the debouncing mode for the EXTINT x pin is selected, the filter mode for that pin (CONFIGn.FILTENx) can not be selected.

The debouncer manages an internal “valid pin state” that depends on the external interrupt (EXTINT) pin transitions, the debouncing mode, and the debouncer prescaler frequency. The valid pin state reflects the pin value after debouncing. The external interrupt (EXTINT) pin is sampled continously on EIC clock. The sampled value is evaluated on each *low-frequency clock* tick to detect a transitional edge when the sampled value is different of the current valid pin state. The sampled value is evaluated on each EIC clock when DPRESCALER.TICKON = 0 or on each *low frequency clock* tick when DPRESCALER.TICKON = 1, to detect a bounce when the sampled value is equal to the current valid pin state. Transitional edge detection increments the transition counter of the EXTINT pin, while bounce detection resets the transition counter. The transition counter must exceed the transition count threshold as defined by the DPRESCALER.STATESm bitfield. In the synchronous


© 2026 Microchip Technology Inc. and its subsidiaries

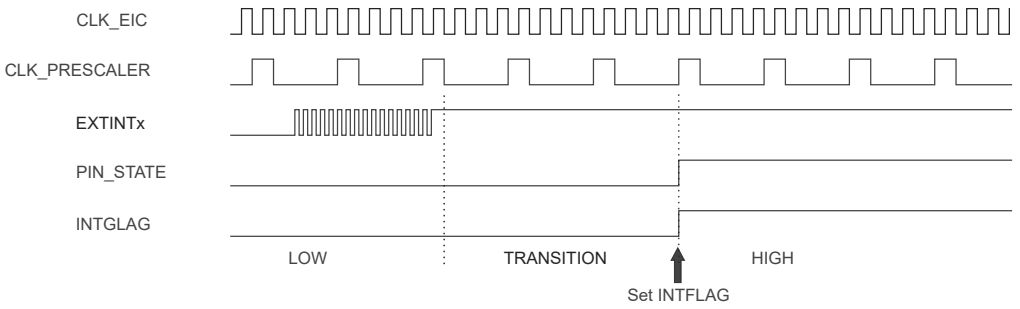
mode the threshold is 4 when DPRESCALER.STATESm = 0, or 8 when DPRESCALER.STATESm = 1. In the asynchronous mode the threshold is 4.

The valid pin state for the pins can be accessed by reading the PINSTATE register for both synchronous or asynchronous debouncing mode.

**Synchronous edge detection:** In this mode the external interrupt (EXTINT) pin is sampled continously on the EIC clock.

- A pin edge transition will be validated when the sampled value is consistently different of the current valid pin state for 4 (or 8 depending on bit DPRESCALER.STATESm) consecutive ticks of the low frequency clock.
- Any pin sample, at the *low-frequency clock* tick rate, with a value opposite to the current valid pin state will increment the transition counter.
- Any pin sample, at EIC clock rate (when DPRESCALER.TICKON = 0) or the *low-frequency clock* tick (when DPRESCALER.TICKON = 1), with a value identical to the current valid pin state will return the transition counter to zero.
- When the transition counter meets the count threshold, the pin edge transition is validated and the pin state PINSTATE.PINSTATE[x] is changed to the detected level.
- The external interrupt flag (INTFLAG.EXTINT[x]) is set when the pin state PINSTATE.PINSTATE[x] is changed.

**Figure 26-3.** EXTINT Pin Synchronous Debouncing (Rising Edge)



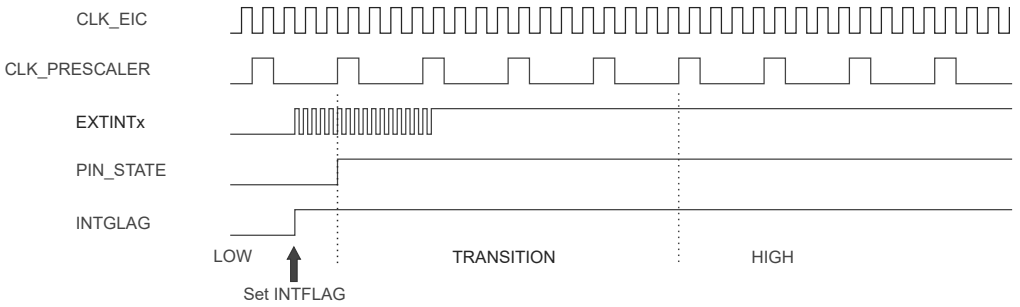

In the Synchronous Edge Detection mode, the EIC clock is required. The Synchronous Edge Detection mode can be used in Idle and Standby Sleep modes.

**Asynchronous edge detection:** In this mode, the external interrupt (EXTINT) pin drives an asynchronous edges detector which triggers any rising or falling edge on the pin:

- Any edge detection that indicates a transition from the current valid pin state will immediately set the valid pin state PINSTATE.PINSTATE[x] to the detected level.
- The external interrupt flag (INTFLAG.EXTINT[x] is immediately changed.
- The edge detector will then be idle until no other rising or falling edge transition is detected during 4 consecutive ticks of the low-frequency clock.
- Any rising or falling edge transition detected during the idle state will return the transition counter to 0.
- After 4 consecutive ticks of the low-frequency clock without bounce detected, the edge detector is ready for a new detection.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 26-4.** EXTINT Pin Asynchronous Debouncing (Rising Edge)




In this mode, the EIC clock is requested. The asynchronous Edge Detection mode can be used in Idle and Standby Sleep modes.

### 26.6.5. DMA Operation

Not applicable.

### 26.6.6. Interrupts

The EIC has the following interrupt sources:

- External interrupt (EXTINTx) pins. See Basic Operation.
- Non-maskable interrupt (NMI) pin. See Additional Features.

Each interrupt source has an associated Interrupt flag. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when an Interrupt condition occurs (NMIFLAG for NMI). Each interrupt, except NMI, can be individually enabled by setting the corresponding bit in the Interrupt Enable Set register (INTENSET = 1), and disabled by setting the corresponding bit in the Interrupt Enable Clear register (INTENCLR = 1). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the EIC is reset. See the INTFLAG register for details on how to clear Interrupt flags. The EIC has one interrupt request line for each external interrupt (EXTINTx) and one line for NMI. The user must read the INTFLAG (or NMIFLAG) register to determine which Interrupt condition is present.

**Notes:**

1. Interrupts must be globally enabled for interrupt requests to be generated.
2. If an external interrupt (EXTINT) is common on two or more I/O pins, only one will be active (the first one programmed).

**Related Links**

Processor and Architecture

### 26.6.7. Events

The EIC can generate the following output events:

- External event from pin (EXTINT0-7).

Setting an Event Output Control register (EVCTRL.EXTINTEO) enables the corresponding output event. Clearing this bit disables the corresponding output event. Refer to *Event System* for details on configuring the Event System.

When the condition on pin EXTINTx matches the configuration in the CONFIGn.SENSEx bit field, the corresponding event is generated, if enabled.

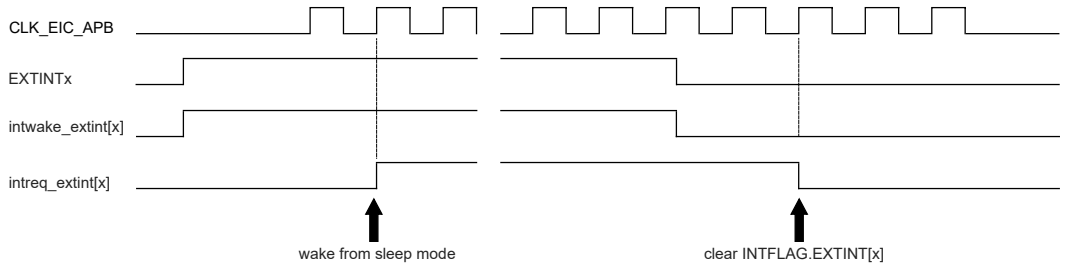

© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

### 26.6.8. Sleep Mode Operation

In sleep modes, an EXTINTx pin can wake up the device if the corresponding condition matches the configuration in the CONFIGn register, and the corresponding bit in the Interrupt Enable Set register (INTENSET) is written to '1'.

**Figure 26-5.** Wake-up Operation Example (High-Level Detection, No Filter, Interrupt Enable Set)




**Related Links**
CONFIGn
External Interrupt Sense Configuration n

### 26.6.9. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in control register (CTRLA.SWRST)
- Enable bit in control register (CTRLA.ENABLE)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

## 26.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | CKSEL | | | ENABLE | SWRST |
| 0x01 | NMICTRL | 7:0 | | | | NMIASYNCH | NMIFILTEN | NMISENSE[2:0] | | |
| 0x02 | NMIFLAG | 7:0 | | | | | | | | NMI |
| | | 15:8 | | | | | | | | |
| 0x04 | SYNCBUSY | 7:0 | | | | | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | EVCTRL | 7:0 | EXTINTEO[7:0] | | | | | | | |
| | | 15:8 | EXTINTEO[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | INTENCLR | 7:0 | EXTINT[7:0] | | | | | | | |
| | | 15:8 | EXTINT[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | INTENSET | 7:0 | EXTINT[7:0] | | | | | | | |
| | | 15:8 | EXTINT[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | INTFLAG | 7:0 | EXTINT[7:0] | | | | | | | |
| | | 15:8 | EXTINT[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | ASYNCH | 7:0 | ASYNCH[7:0] | | | | | | | |
| | | 15:8 | ASYNCH[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | CONFIG0 | 7:0 | FILTEN1 | SENSE1[2:0] | | | FILTEN0 | SENSE0[2:0] | | |
| | | 15:8 | FILTEN3 | SENSE3[2:0] | | | FILTEN2 | SENSE2[2:0] | | |
| | | 23:16 | FILTEN5 | SENSE5[2:0] | | | FILTEN4 | SENSE4[2:0] | | |
| | | 31:24 | FILTEN7 | SENSE7[2:0] | | | FILTEN6 | SENSE6[2:0] | | |
| 0x20 | CONFIG1 | 7:0 | FILTEN1 | SENSE1[2:0] | | | FILTEN0 | SENSE0[2:0] | | |
| | | 15:8 | FILTEN3 | SENSE3[2:0] | | | FILTEN2 | SENSE2[2:0] | | |
| | | 23:16 | FILTEN5 | SENSE5[2:0] | | | FILTEN4 | SENSE4[2:0] | | |
| | | 31:24 | FILTEN7 | SENSE7[2:0] | | | FILTEN6 | SENSE6[2:0] | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DEBOUNCEN | 7:0 | DEBOUNCEN[7:0] | | | | | | | |
| | | 15:8 | DEBOUNCEN[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x34 | DPRESCALER | 7:0 | STATES1 | PRESCALER1[2:0] | | | STATES0 | PRESCALER0[2:0] | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | TICKON |
| | | 31:24 | | | | | | | | |
| 0x38 | PINSTATE | 7:0 | PINSTATE[7:0] | | | | | | | |
| | | 15:8 | PINSTATE[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 26.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | CKSEL | | | ENABLE | SWRST |
| Access | | | | RW | | | RW | W |
| Reset | | | | 0 | | | 0 | 0 |

**Bit 4 – CKSEL** Clock Selection
The EIC can be clocked either by GCLK_EIC (when a frequency higher than 32KHz is required for filtering) or by CLK_ULP32K (when power consumption is the priority).
This bit is not Write-Synchronized.

| Value | Description |
|---|---|
| 0 | The EIC is clocked by GCLK_EIC. |
| 1 | The EIC is clocked by CLK_ULP32K. |

**Bit 1 – ENABLE** Enable
Due to synchronization there is a delay between writing to CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable bit in the Synchronization Busy register will be set (SYNCBUSY.ENABLE=1). SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not Enable-Protected.
This bit is Write-Synchronized.

| Value | Description |
|---|---|
| 0 | The EIC is disabled. |
| 1 | The EIC is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the EIC to their initial state, and the EIC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the Reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the Reset is complete.
This bit is not Enable-Protected.
This bit is Write-Synchronized.

| Value | Description |
|---|---|
| 0 | There is no ongoing reset operation. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.2. Non-Maskable Interrupt Control

**Name:** NMICTRL
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | NMIASYNCH | NMIFILTEN | NMISENSE[2:0] | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bit 4 – NMIASYNCH** NMI Asynchronous Edge Detection Mode
The NMI edge detection can be operated synchronously or asynchronously to the EIC clock.

| Value | Description |
|---|---|
| 0 | The NMI edge detection is synchronously operated. |
| 1 | The NMI edge detection is asynchronously operated. |

**Bit 3 – NMIFILTEN** Non-Maskable Interrupt Filter Enable

| Value | Description |
|---|---|
| 0 | NMI filter is disabled. |
| 1 | NMI filter is enabled. |

**Bits 2:0 – NMISENSE[2:0]** Non-Maskable Interrupt Sense Configuration
These bits define on which edge or level the NMI triggers.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No detection |
| 0x1 | RISE | Rising-edge detection |
| 0x2 | FALL | Falling-edge detection |
| 0x3 | BOTH | Both-edge detection |
| 0x4 | HIGH | High-level detection |
| 0x5 | LOW | Low-level detection |
| 0x6 – 0x7 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.3. Non-Maskable Interrupt Flag Status and Clear

**Name:** NMIFLAG
**Offset:** 0x02
**Reset:** 0x0000

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | NMI |
| Access | | | | | | | | RW |
| Reset | | | | | | | | 0 |

**Bit 0 – NMI** Non-Maskable Interrupt
This flag is cleared by writing a '1' to it.
This flag is set when the NMI pin matches the NMI sense configuration, and will generate an interrupt request.
Writing a '0' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.4. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x04
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** Enable Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.ENABLE bit is complete. |
| 1 | Write synchronization for CTRLA.ENABLE bit is ongoing. |

**Bit 0 – SWRST** Software Reset Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.5. Event Control

**Name:** EVCTRL
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EXTINTEO[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTINTEO[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – EXTINTEO[15:0]** External Interrupt Event Output Enable
The bit x of EXTINTEO enables the event associated with the EXTINTx pin.

| Value | Description |
|---|---|
| 0 | Event from pin EXTINTx is disabled. |
| 1 | Event from pin EXTINTx is enabled and will be generated when EXTINTx pin matches the external interrupt sensing configuration. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.6. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – EXTINT[15:0]** External Interrupt Enable
The bit x of EXTINT disables the interrupt associated with the EXTINTx pin.
Writing a '0' to bit x has no effect.
Writing a '1' to bit x will clear the External Interrupt Enable bit x, which disables the external interrupt EXTINTx.

| Value | Description |
|---|---|
| 0 | The external interrupt x is disabled. |
| 1 | The external interrupt x is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.7. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – EXTINT[15:0]** External Interrupt Enable
The bit x of EXTINT enables the interrupt associated with the EXTINTx pin.
Writing a '0' to bit x has no effect.
Writing a '1' to bit x will set the External Interrupt Enable bit x, which enables the external interrupt EXTINTx.

| Value | Description |
|---|---|
| 0 | The external interrupt x is disabled. |
| 1 | The external interrupt x is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.8. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTINT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – EXTINT[15:0]** External Interrupt
The flag bit x is cleared by writing a '1' to it.
This flag is set when EXTINTx pin matches the external interrupt sense configuration and will generate an interrupt request if INTENCLR/SET.EXTINT[x] is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the External Interrupt x flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.9. External Interrupt Asynchronous Mode

**Name:** ASYNCH
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ASYNCH[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ASYNCH[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – ASYNCH[15:0]** Asynchronous Edge Detection Mode
The bit x of ASYNCH set the Asynchronous Edge Detection Mode for the interrupt associated with the EXTINTx pin.

| Value | Description |
|---|---|
| 0 | The EXTINT x edge detection is synchronously operated. |
| 1 | The EXTINT x edge detection is asynchronously operated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.10. External Interrupt Sense Configuration n

**Name:** CONFIGn
**Offset:** 0x1C + n*0x04 [n=0..1]
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | FILTEN7 | SENSE7[2:0] | | | FILTEN6 | SENSE6[2:0] | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | FILTEN5 | SENSE5[2:0] | | | FILTEN4 | SENSE4[2:0] | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FILTEN3 | SENSE3[2:0] | | | FILTEN2 | SENSE2[2:0] | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | FILTEN1 | SENSE1[2:0] | | | FILTEN0 | SENSE0[2:0] | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 3, 7, 11, 15, 19, 23, 27, 31 – FILTENx** Filter Enable x [x=7..0]
**Note:** The filter must be disabled if the asynchronous detection is enabled.

| Value | Description |
|---|---|
| 0 | Filter is disabled for EXTINT[n*8+x] input. |
| 1 | Filter is enabled for EXTINT[n*8+x] input. |

**Bits 0:2, 4:6, 8:10, 12:14, 16:18, 20:22, 24:26, 28:30 – SENSEx** Input Sense Configuration x [x=7..0]
These bits define on which edge or level the interrupt or event for EXTINT[n*8+x] will be generated.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No detection |
| 0x1 | RISE | Rising-edge detection |
| 0x2 | FALL | Falling-edge detection |
| 0x3 | BOTH | Both-edge detection |
| 0x4 | HIGH | High-level detection |
| 0x5 | LOW | Low-level detection |
| 0x6 – 0x7 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.11. Debouncer Enable

**Name:** DEBOUNCEN
**Offset:** 0x30
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

**Important:** This register is only available on SAM C20/C21 N variants.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DEBOUNCEN[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DEBOUNCEN[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – DEBOUNCEN[15:0]** Debouncer Enable
The bit x of DEBOUNCEN set the Debounce mode for the interrupt associated with the EXTINTx pin.

| Value | Description |
|---|---|
| 0 | The EXTINT x edge input is not debounced. |
| 1 | The EXTINT x edge input is debounced. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.12. Debouncer Prescaler

**Name:** DPRESCALER
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

**Important:** This register is only available on SAM C20/C21 N variants.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | TICKON |
| Access | | | | | | | | RW |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | STATES1 | PRESCALER1[2:0] | | | STATES0 | PRESCALER0[2:0] | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 16 – TICKON** Pin Sampler frequency selection
This bit selects the clock used for the sampling of bounce during transition detection.

| Value | Description |
|---|---|
| 0 | The bounce sampler is using GCLK_EIC. |
| 1 | The bounce sampler is using the low frequency clock. |

**Bits 3, 7 – STATESx** Debouncer number of states x
This bit selects the number of samples by the debouncer low frequency clock needed to validate a transition from current pin state to next pin state in synchronous debouncing mode for pins EXTINT[7+(8x):8x].

| Value | Description |
|---|---|
| 0 | The number of low frequency samples is 3. |
| 1 | The number of low frequency samples is 7. |

**Bits 0:2, 4:6 – PRESCALERx** Debouncer Prescaler x
These bits select the debouncer low frequency clock for pins EXTINT[7+(8x):8x].

| Value | Name | Description |
|---|---|---|
| 0x0 | F/2 | EIC clock divided by 2 |
| 0x1 | F/4 | EIC clock divided by 4 |
| 0x2 | F/8 | EIC clock divided by 8 |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x3 | F/16 | EIC clock divided by 16 |
| 0x4 | F/32 | EIC clock divided by 32 |
| 0x5 | F/64 | EIC clock divided by 64 |
| 0x6 | F/128 | EIC clock divided by 128 |
| 0x7 | F/256 | EIC clock divided by 256 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 26.8.13. Pin State

**Name:** PINSTATE
**Offset:** 0x38
**Reset:** 0x00000000

**Important:** This register is only available on SAM C20/C21 N variants.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PINSTATE[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PINSTATE[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – PINSTATE[15:0]** Pin State
These bits return the valid pin state of the debounced external interrupt pin EXTINTx.


© 2026 Microchip Technology Inc. and its subsidiaries

# 27. NVMCTRL – Nonvolatile Memory Controller

## 27.1. Overview

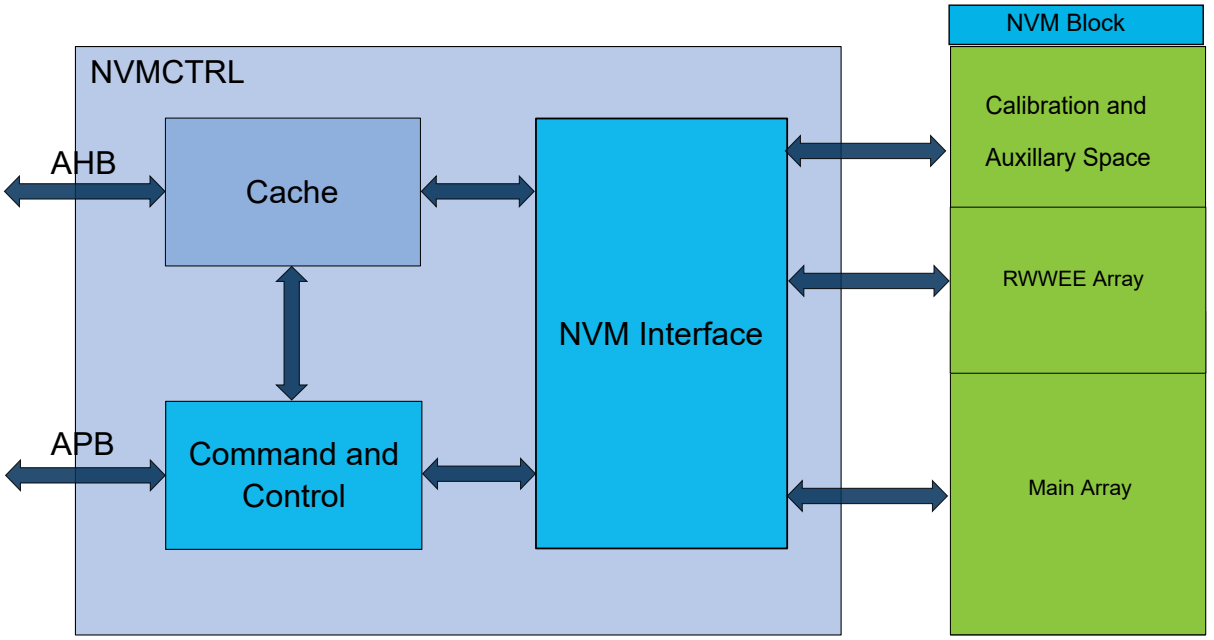
Nonvolatile Memory (NVM) is a reprogrammable Flash memory that retains program and data storage even with power off. It embeds a main array and a separate smaller array intended for Read While Write EEPROM emulation (RWWEE , standing for Read (the main array) while Write (the EEPROM) ) that can be programmed while reading the main array. The NVM Controller (NVMCTRL) connects to the AHB and APB bus interfaces for system access to the NVM block. The AHB interface is used for reads and writes to the NVM block, while the APB interface is used for commands and configuration.

## 27.2. Features

- 32-bit AHB interface for reads and writes
- Dedicated RWWEE array
- All NVM sections are memory mapped to the AHB, including calibration and system configuration
- 32-bit APB interface for commands and control
- Programmable wait states for read optimization
- 16 regions can be individually protected or unprotected
- Additional protection for bootloader
- Supports device protection through a security bit
- Interface to power manager for power-down of Flash blocks in sleep modes
- Can optionally wake-up on exit from sleep or on first access
- Direct-mapped cache

## 27.3. Block Diagram

**Figure 27-1.** Block Diagram





© 2026 Microchip Technology Inc. and its subsidiaries

## 27.4. Signal Description

Not applicable.

## 27.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described in the following sections.

### 27.5.1. Power Management

The NVMCTRL will continue to operate in any Sleep mode where the selected source clock is running. The NVMCTRL interrupts can be used to wake-up the device from Sleep modes.

The Power Manager will automatically put the NVM block into a Low-power state when entering Sleep mode. This is based on the Control B register SLEEPPRM bit setting. (Refer to the CTRLB.SLEEPPRM register description for more details.) The NVM block goes into Low-power mode automatically when the device enters STANDBY mode regardless of SLEEPPRM. The NVM Page Buffer is lost when the NVM goes into Low-power mode, so a write command must be issued prior entering the NVM Low-power mode. NVMCTRL SLEEPPRM can be disabled to avoid such loss when the CPU goes into sleep except if the device goes into STANDBY mode for which there is no way to retain the Page Buffer.

**Related Links**

PM - Power Manager

### 27.5.2. Clocks

Two synchronous clocks are used by the NVMCTRL. One is provided by the AHB bus (CLK_NVMCTRL_AHB) and the other is provided by the APB bus (CLK_NVMCTRL_APB). For higher system frequencies, a programmable number of wait states has to be configured in CTRLB.RWS. Refer to the "NVM Characteristics" chapter of the Electrical Characterisitics section for the exact number of wait states to be used for a particular frequency range. When changing the AHB bus frequency, the user shall ensure that the NVM Controller is configured with the proper number of wait states. For example when switching to a higher AHB frequency, the number of wait states shall be adapted to the future frequency first, and then only can the frequency be increased to the new value.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

### 27.5.3. Interrupts

The NVM Controller interrupt request line is connected to the interrupt controller. Using the NVMCTRL interrupt requires the interrupt controller to be programmed first.

### 27.5.4. Debug Operation

When an external debugger forces the CPU into debug mode, the peripheral continues normal operation.

Access to the NVM block can be protected by the security bit. In this case, the NVM block will not be accessible. See the section on the NVMCTRL Security Bit for details.

### 27.5.5. Register Access Protection

All registers with write-access are optionally write-protected by the Peripheral Access Controller (PAC), except the following registers:

- Interrupt Flag Status and Clear register (INTFLAG)
- Status register (STATUS)


© 2026 Microchip Technology Inc. and its subsidiaries

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

**Related Links**
PAC - Peripheral Access Controller

### 27.5.6. Analog Connections

Not applicable.

## 27.6. Functional Description

### 27.6.1. Principle of Operation

The NVM Controller is a client on the AHB and APB buses. It responds to commands, read requests and write requests, based on user configuration.

#### 27.6.1.1. Initialization

After power up, the NVM Controller goes through a power-up sequence. During this time, access to the NVM Controller from the AHB bus is halted. Upon power-up completion, the NVM Controller is operational without any need for user configuration.

### 27.6.2. Memory Organization

Refer to the Physical Memory Map for memory sizes and addresses for each device.

The NVM is organized into rows, where each row contains four pages, as shown in the NVM Row Organization figure. One page is made of 64 bytes, that is sixteen 32-bit words, or height 64-bit double words. One row is made of 4 pages, that is 256 bytes, or sixty four 32-bit words, or sixteen 64-bit double words. The NVM has a row-erase granularity, while the write granularity is by page. In other words, a single row erase will erase all four pages in the row, while four write operations are used to write the complete row.

**Figure 27-2.** NVM Row Organization

| | | | | |
|---|---|---|---|---|
| Row n | Page (n*4) + 3 | Page (n*4) + 2 | Page (n*4) + 1 | Page (n*4) + 0 |

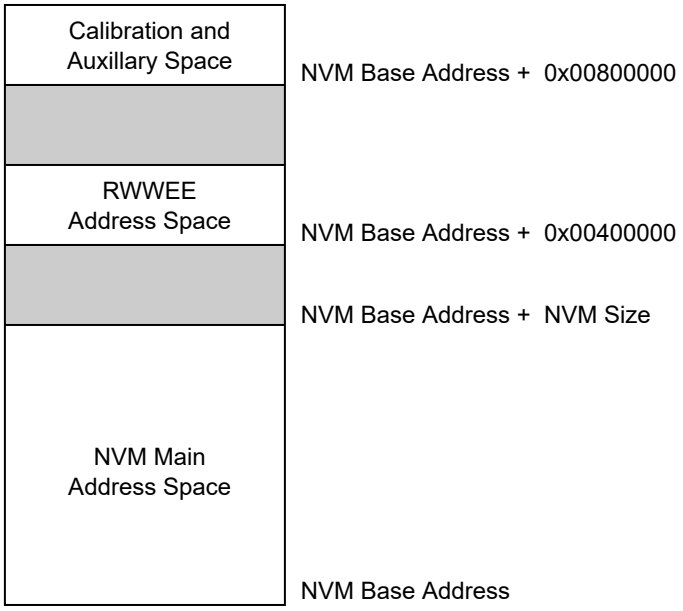
The NVM block contains a calibration and auxiliary space, a RWWEE section, and a main array that are memory mapped. Refer to the NVM Organization figure below for details.

The calibration and auxiliary space contains factory calibration and system configuration information. These spaces can be read from the AHB bus in the same way as the main NVM main address space.

In addition, a boot loader section can be allocated at the beginning of the main array, and an EEPROM Emulation section can be allocated at the end of the NVM main address space.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 27-3.** NVM Memory Address Space



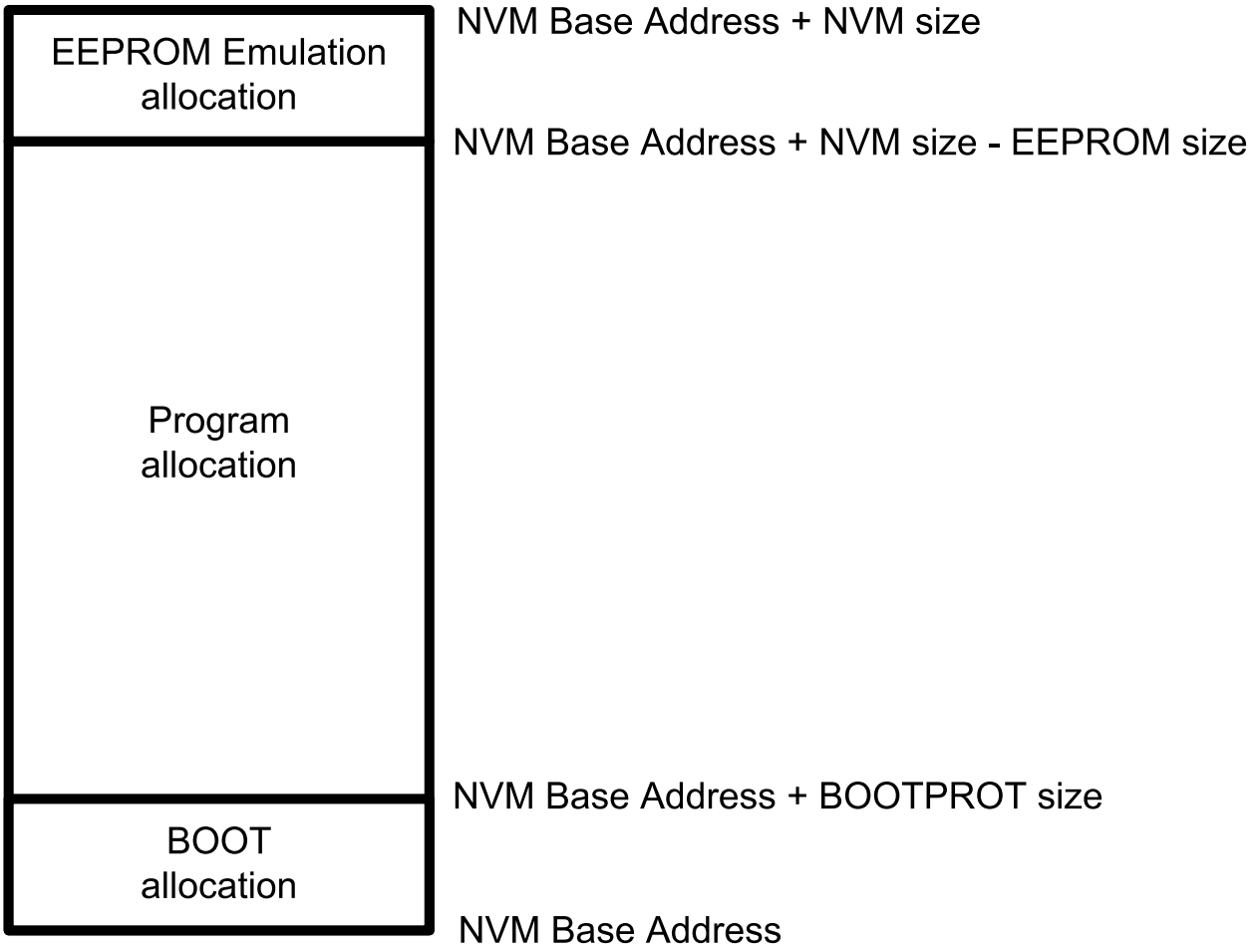

The lower rows in the NVM main address space can be allocated as a boot loader section by using the NVM User Row BOOTPROT fuses, and the upper rows can be allocated to EEPROM Emulation section by using the NVM User Row EEPROM Size fuses, as shown in the figure below.

The boot loader section is protected by the Lock bits corresponding to this address space and by the BOOTPROT[2:0] fuse. The EEPROM Emulation rows can be written regardless of the region lock status.

The number of rows protected by BOOTPROT is given in Boot Loader Size, the number of rows allocated to the EEPROM Emulation area are specified in EEPROM Size.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 27-4.** EEPROM and Boot Loader Memory Allocation




**Related Links**
Physical Memory Map

### 27.6.3. Region Lock Bits

The NVM block is grouped into 16 equally sized regions. The region size is dependent on the Flash memory size, and is given in the table below. Each region has a dedicated lock bit preventing writing and erasing pages in the region. After production, all regions will be unlocked.

**Table 27-1.** Region Size

| Memory Size [KB] | Region Size [KB] |
|---|---|
| 256 | 16 |
| 128 | 8 |
| 64 | 4 |
| 32 | 2 |

To lock or unlock a region, the Lock Region and Unlock Region commands are provided. Writing one of these commands will temporarily lock/unlock the region containing the address loaded in the ADDR register. ADDR can be written by software, or the automatically loaded value from a write operation can be used. The new setting will stay in effect until the next Reset, or until the setting is changed again using the Lock and Unlock commands. The current status of the lock can be determined by reading the LOCK register.


© 2026 Microchip Technology Inc. and its subsidiaries

To change the default lock/unlock setting for a region, the user configuration section of the auxiliary space must be written using the Write Auxiliary Page command. Writing to the auxiliary space will take effect after the next Reset. Therefore, a boot of the device is needed for changes in the lock/unlock setting to take effect. Refer to the Physical Memory Map for calibration and auxiliary space address mapping.

**Related Links**

Physical Memory Map

### 27.6.4. Command and Data Interface

The NVM Controller is addressable from the APB bus, while the NVM main address space is addressable from the AHB bus. Read and automatic page write operations are performed by addressing the NVM main address space or the RWWEE address space directly, while other operations such as manual page writes and row erases must be performed by issuing commands through the NVM Controller.

To issue a command, the CTRLA.CMD bits must be written along with the CTRLA.CMDEX value. When a command is issued, INTFLAG.READY will be cleared until the command has completed. Any commands written while INTFLAG.READY is low will be ignored.

The CTRLB register must be used to control the power reduction mode, read wait states, and the write mode.

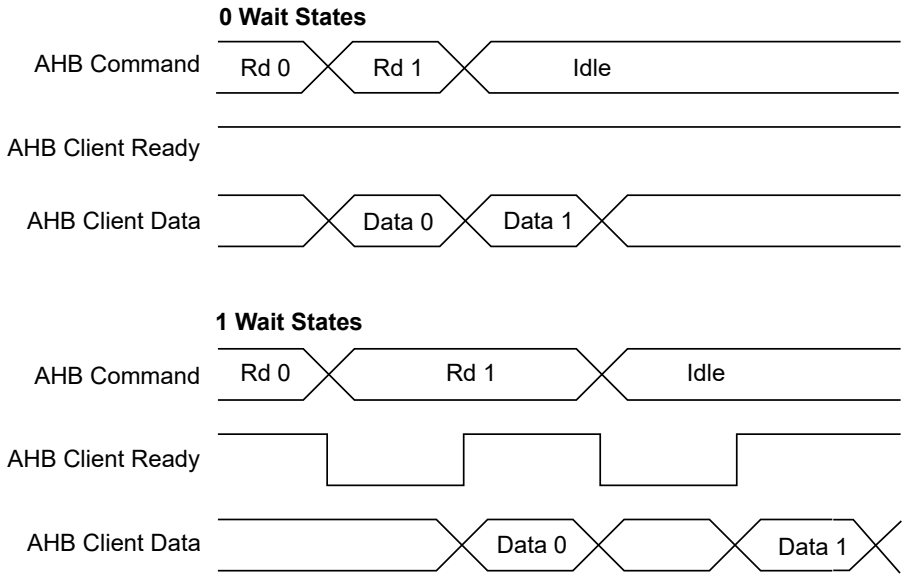
#### 27.6.4.1. NVM Read

Reading from the NVM main address space is performed via the AHB bus by addressing the NVM main address space or auxiliary address space directly. Read data is available after the configured number of Read Wait states (CTRLB.RWS) set in the NVM Controller.

The number of cycles data are delayed to the AHB bus is determined by the Read Wait states. Examples of using zero and one Wait states are shown in the following figure.

Reading the NVM main address space while a programming or erase operation is ongoing on the NVM main array results in an AHB bus stall until the end of the operation. Reading the NVM main array does not stall the bus when the RWWEE array is being programmed or erased.

**Figure 27-5.** Read Wait State Examples





© 2026 Microchip Technology Inc. and its subsidiaries

#### 27.6.4.2. RWWEE Read

Reading from the RWWEE address space is performed via the AHB bus by addressing the RWWEE address space directly.

Read timings are similar to regular NVM read timings when access size is Byte or half-Word. The AHB data phase is twice as long in case of full-Word-size access.

It is not possible to read the RWWEE area while the NVM main array is being written or erased, whereas the RWWEE area can be written or erased while the main array is being read.

The RWWEE address space is not cached, therefore it is recommended to limit access to this area for performance and power consumption considerations.

#### 27.6.4.3. NVM Write

The NVM Controller requires that an erase must be done before programming. The entire NVM main address space and the RWWEE address space can be erased by a debugger Chip Erase command. Alternatively, rows can be individually erased by the Erase Row command or the RWWEE Erase Row command to erase the NVM main address space or the RWWEE address space, respectively.

After programming the NVM main array, the region that the page resides in can be locked to prevent spurious write or erase sequences. Locking is performed on a per-region basis, and so, locking a region will lock all pages inside the region.

Data to be written to the NVM block are first written to and stored in an internal buffer called the *page buffer*. The page buffer contains the same number of bytes as an NVM page. Writes to the page buffer must be 16 or 32 bits. 8-bit writes to the page buffer are not allowed and will cause a system exception.

Internally, writes to the page buffer are on a 64-bit basis through the page buffer load data register (PBLDATA1 and PBLDATA0). The PBLDATA register is a holding register for writes to the same 64-bit page buffer section. Data within a 64-bit section can be written in any order. Crossing a 64-bit boundary will reset the PBLDATA register to all ones. The following example assumes startup from reset where the current address is 0 and PBLDATA is all ones. Only 64 bits of the page buffer are written at a time, but 128 bits are shown for reference.

**Sequential 32-bit Write Example:**

- 32-bit 0x1 written to address 0
  - Page buffer[127:0] = {0xFFFFFFFF_FFFFFFFF, PBLDATA[63:32], 0x00000001}
  - PBLDATA[63:0] = {PBLDATA[63:32], 0x00000001}
- 32-bit 0x2 written to address 1
  - Page buffer[127:0] = {0xFFFFFFFF_FFFFFFFF, 0x00000002, PBLDATA[31:0]}
  - PBLDATA[63:0] = {0x00000002, PBLDATA[31:0]}
- 32-bit 0x3 written to address 2 (crosses 64-bit boundary)
  - Page buffer[127:0] = 0xFFFFFFFF_00000003_00000002_00000001
  - PBLDATA[63:0] = 0xFFFFFFFF_00000003

Random access writes to 32-bit words within the page buffer will overwrite the opposite word within the same 64-bit section with ones. In the following example, notice that 0x00000001 is overwritten with 0xFFFFFFFF from the third write due to the 64-bit boundary crossing. Only 64 bits of the page buffer are written at a time, but 128 bits are shown for reference.

**Random Access 32-bit Write Example:**

- 32-bit 0x1 written to address 2
  - Page buffer[127:0] = 0xFFFFFFFF_00000001_FFFFFFFF_FFFFFFFF
  - PBLDATA[63:0] = 0xFFFFFFFF_00000001


© 2026 Microchip Technology Inc. and its subsidiaries

- 32-bit 0x2 written to address 1
  - Page buffer[127:0] = 0xFFFFFFFF_00000001_00000002_FFFFFFFF
  - PBLDATA[63:0] = 0x00000002_FFFFFFFF
- 32-bit 0x3 written to address 3
  - Page buffer[127:0] = 0x00000003_FFFFFFFF_00000002_FFFFFFFF
  - PBLDATA[63:0] = 0x00000003_0xFFFFFFFF

Both the NVM main array and the RWWEE array share the same page buffer. Writing to the NVM block via the AHB bus is performed by a load operation to the page buffer. For each AHB bus write, the address is stored in the ADDR register. After the page buffer has been loaded with the required number of bytes, the page can be written to the NVM main array or the RWWEE array by setting CTRLA.CMD to 'Write Page' or 'RWWEE Write Page', respectively, and setting the key value to CMDEX. The LOAD bit in the STATUS register indicates whether the page buffer has been loaded or not. Before writing the page to memory, the accessed row must be erased.

Automatic page writes are enabled by writing the manual Write bit to zero (CTRLB.MANW=0). This will trigger a write operation to the page addressed by ADDR when the last location of the page is written.

Because the address is automatically stored in ADDR during the I/O bus write operation, the last given address will be present in the ADDR register. There is no need to load the ADDR register manually, unless a different page in memory is to be written.

#### 27.6.4.3.1. Procedure for Manual Page Writes (CTRLB.MANW = 1)

The row to be written to must be erased before the write command is given.

1. Write to the page buffer by addressing the NVM main address space directly.
2. Issue a Read-Memory-Barrier from the NVMCTRL AHB0 interface. A single read to AHB0 after writing the whole page buffer.
3. Write the page buffer to memory: CTRL.CMD = 'Write Page' and CMDEX.
4. The READY bit in the INTFLAG register will be low while programming is in progress and access through the AHB will be stalled.

#### 27.6.4.3.2. Procedure for Automatic Page Writes (CTRLB.MANW = 0)

The row to be written to must be erased before the last write to the page buffer is performed.

Note that partially written pages must be written with a manual write.

- Write to the page buffer by addressing the NVM main address space directly.
  When the last location in the page buffer is written, the page is automatically written to NVM main address space.
- INTFLAG.READY will be zero while programming is in progress and access through the AHB will be stalled.

### 27.6.4.4. Page Buffer Clear

The page buffer is automatically set to all '1' after a page write is performed. If a partial page has been written and it is desired to clear the contents of the page buffer, the Page Buffer Clear command can be used.

### 27.6.4.5. Erase Row

Before a page can be written, the row containing that page must be erased. The Erase Row command can be used to erase the desired row in the NVM main address space. The RWWEE Erase Row can be used to erase the desired row in the RWWEE array. Erasing the row sets all bits to '1'. If the row resides in a region that is locked, the erase will not be performed and the Lock Error bit in the Status register (STATUS.LOCKE) will be set.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 27.6.4.5.1. Procedure for Erase Row

- Write the address of the row to erase to ADDR. Any address within the row can be used.
- Issue an Erase Row command.

**Note:** The NVM Address bit field in the Address register (ADDR.ADDR) uses 16-bit addressing.

### 27.6.4.6. Lock and Unlock Region

These commands are used to lock and unlock regions as detailed in section Region Lock Bits.

### 27.6.4.7. Set and Clear Power Reduction Mode

The NVM Controller and block can be taken in and out of power reduction mode through the Set and Clear Power Reduction Mode commands. When the NVM Controller and block are in power reduction mode, the Power Reduction Mode bit in the Status register (STATUS.PRM) is set.

## 27.6.5. NVM User Configuration

The NVM user configuration resides in the auxiliary space. Refer to the Physical Memory Map of the device for calibration and auxiliary space address mapping (See NVM User Row Mapping for more information).

The bootloader resides in the main array starting at offset zero. The allocated boot loader section is write-protected.

**Table 27-2.** Boot Loader Size

| BOOTPROT [2:0] | Rows Protected by BOOTPROT | Boot Loader Size in Bytes |
|---|---|---|
| 0x7(1) | None | 0 |
| 0x6 | 2 | 512 |
| 0x5 | 4 | 1024 |
| 0x4 | 8 | 2048 |
| 0x3 | 16 | 4096 |
| 0x2 | 32 | 8192 |
| 0x1 | 64 | 16384 |
| 0x0 | 128 | 32768 |

**Note:**

1. Default value is 0x7.

The EEPROM[2:0] bits indicate the EEPROM size, see the table below. The EEPROM resides in the upper rows of the NVM main address space and is writable, regardless of the region lock status.

**Table 27-3.** EEPROM Emulation Area Size

| EEPROM[2:0] | Rows Allocated to EEPROM | EEPROM Size in Bytes |
|---|---|---|
| 7 | None | 0 |
| 6 | 1 | 256 |
| 5 | 2 | 512 |
| 4 | 4 | 1024 |
| 3 | 8 | 2048 |
| 2 | 16 | 4096 |
| 1 | 32 | 8192 |
| 0 | 64 | 16384 |

**Note:**

1. Default value is 0x7.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Physical Memory Map

### 27.6.6. Security Bit

The security bit allows the entire chip to be locked from external access for code security. The security bit can be written by a dedicated command, Set Security Bit (SSB). Once set, the only way to clear the security bit is through a debugger Chip Erase command. After issuing the SSB command, the PROGE error bit can be checked.

In order to increase the security level it is recommended to enable the internal BODVDD when the security bit is set.

**Related Links**
DSU - Device Service Unit

### 27.6.7. Cache

The NVM Controller cache reduces the device power consumption and improves system performance when wait states are required. Only the NVM main array address space is cached. It is a direct-mapped cache that implements 8 lines of 64 bits (i.e., 64 Bytes). NVM Controller cache can be enabled by writing a '0' to the Cache Disable bit in the Control B register (CTRLB.CACHEDIS).

The cache can be configured to three different modes using the Read Mode bit group in the Control B register (CTRLB.READMODE).

The INVALL command can be issued using the Command bits in the Control A register to invalidate all cache lines (CTRLA.CMD=INVALL). Commands affecting NVM content automatically invalidate cache lines.


© 2026 Microchip Technology Inc. and its subsidiaries

## 27.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | CMD[6:0] | | | | | | |
| | | 15:8 | CMDEX[7:0] | | | | | | | |
| 0x02 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | CTRLB | 7:0 | MANW | | | RWS[3:0] | | | | |
| | | 15:8 | | | | | | | SLEEPPRM[1:0] | |
| | | 23:16 | | | | | | CACHEDIS | READMODE[1:0] | |
| | | 31:24 | | | | | | | | |
| 0x08 | PARAM | 7:0 | NVMP[7:0] | | | | | | | |
| | | 15:8 | NVMP[15:8] | | | | | | | |
| | | 23:16 | RWWEEP[3:0] | | | | | PSZ[2:0] | | |
| | | 31:24 | RWWEEP[11:4] | | | | | | | |
| 0x0C | INTENCLR | 7:0 | | | | | | | ERROR | READY |
| 0x0D ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | INTENSET | 7:0 | | | | | | | ERROR | READY |
| 0x11 ... 0x13 | Reserved | | | | | | | | | |
| 0x14 | INTFLAG | 7:0 | | | | | | | ERROR | READY |
| 0x15 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | STATUS | 7:0 | | | | NVME | LOCKE | PROGE | LOAD | PRM |
| | | 15:8 | | | | | | | | SB |
| 0x1A ... 0x1B | Reserved | | | | | | | | | |
| 0x1C | ADDR | 7:0 | ADDR[7:0] | | | | | | | |
| | | 15:8 | ADDR[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | ADDR[16] |
| | | 31:24 | | | | | | | | |
| 0x20 | LOCK | 7:0 | LOCK[7:0] | | | | | | | |
| | | 15:8 | LOCK[15:8] | | | | | | | |
| 0x22 ... 0x27 | Reserved | | | | | | | | | |
| 0x28 | PBLDATA0 | 7:0 | PBLDATA[7:0] | | | | | | | |
| | | 15:8 | PBLDATA[15:8] | | | | | | | |
| | | 23:16 | PBLDATA[23:16] | | | | | | | |
| | | 31:24 | PBLDATA[31:24] | | | | | | | |
| 0x2C | PBLDATA1 | 7:0 | PBLDATA[7:0] | | | | | | | |
| | | 15:8 | PBLDATA[15:8] | | | | | | | |
| | | 23:16 | PBLDATA[23:16] | | | | | | | |
| | | 31:24 | PBLDATA[31:24] | | | | | | | |

## 27.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** PAC Write-Protection

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CMDEX[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | CMD[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:8 – CMDEX[7:0]** Command Execution

When this bit group is written to the key value 0xA5, the command written to CMD will be executed. If a value different from the key value is tried, the write will not be performed and the Programming Error bit in the Status register (STATUS.PROGE) will be set. PROGE is also set if a previously written command is not completed yet.

The key value must be written at the same time as CMD. If a command is issued through the APB bus on the same cycle as an AHB bus access, the AHB bus access will be given priority. The command will then be executed when the NVM block and the AHB bus are idle.

INTFLAG.READY must be '1' when the command is issued.

Bit 0 of the CMDEX bit group will read back as '1' until the command is issued.

**Note:** The NVM Address bit field in the Address register (ADDR.ADDR) uses 16-bit addressing.

**Bits 6:0 – CMD[6:0]** Command

These bits define the command to be executed when the CMDEX key is written.

| CMD[6:0] | Group Configuration | Description |
|---|---|---|
| 0x00-0x01 | - | Reserved |
| 0x02 | ER | Erase Row - Erases the row addressed by the ADDR register in the NVM main array. |
| 0x03 | - | Reserved |
| 0x04 | WP | Write Page - Writes the contents of the page buffer to the page addressed by the ADDR register. |
| 0x05 | EAR | Erase Auxiliary Row - Erases the auxiliary row addressed by the ADDR register. This command can be given only when the Security bit is not set and only to the User Configuration Row. |
| 0x06 | WAP | Write Auxiliary Page - Writes the contents of the page buffer to the page addressed by the ADDR register. This command can be given only when the Security bit is not set and only to the User Configuration Row. |
| 0x07-0x0E | - | Reserved |
| 0x0F | - | Reserved |
| 0x1A-0x19 | - | Reserved |
| 0x1A | RWWEEER | RWWEE Erase Row - Erases the row addressed by the ADDR register in the RWWEE array. |
| 0x1B | - | Reserved |
| 0x1C | RWWEEWP | RWWEE Write Page - Writes the contents of the page buffer to the page addressed by the ADDR register in the RWWEE array. |
| 0x1D-0x3F | - | Reserved |
| 0x40 | LR | Lock Region - Locks the region containing the address location in the ADDR register. |
| 0x41 | UR | Unlock Region - Unlocks the region containing the address location in the ADDR register. |
| 0x42 | SPRM | Sets the Power Reduction mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

| **CMD** (continued) | | |
|---|---|---|
| **CMD[6:0]** | **Group Configuration** | **Description** |
| 0x43 | CPRM | Clears the Power Reduction mode. |
| 0x44 | PBC | Page Buffer Clear - Clears the page buffer. |
| 0x45 | SSB | Locks device from external access for code security. |
| 0x46 | INVALL | Invalidates all cache lines. |
| 0x47-0x7F | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.2. Control B

**Name:** CTRLB
**Offset:** 0x04
**Reset:** 0x00000080
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CACHEDIS | READMODE[1:0] | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SLEEPPRM[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MANW | | | RWS[3:0] | | | | |
| Access | R/W | | | R/W | R/W | R/W | R/W | |
| Reset | 1 | | | 0 | 0 | 0 | 0 | |

**Bit 18 – CACHEDIS** Cache Disable
This bit is used to disable the cache.

| Value | Description |
|---|---|
| 0 | The cache is enabled |
| 1 | The cache is disabled |

**Bits 17:16 – READMODE[1:0]** NVMCTRL Read Mode

| Value | Name | Description |
|---|---|---|
| 0x0 | NO_MISS_PENALTY | The NVM Controller (cache system) does not insert wait states on a cache miss. Gives the best system performance. |
| 0x1 | LOW_POWER | Reduces power consumption of the cache system, but inserts a wait state each time there is a cache miss. This mode may not be relevant if CPU performance is required, as the application will be stalled and may lead to increased run time. |
| 0x2 | DETERMINISTIC | The cache system ensures that a cache hit or miss takes the same amount of time, determined by the number of programmed Flash wait states. This mode can be used for real-time applications that require deterministic execution timings. |
| 0x3 | Reserved | |

**Bits 9:8 – SLEEPPRM[1:0]** Power Reduction Mode during Sleep
Indicates the Power Reduction Mode during sleep.

| Value | Name | Description |
|---|---|---|
| 0x0 | WAKEUPACCESS | NVM block enters low-power mode when entering sleep.<br>NVM block exits low-power mode upon first access. |
| 0x1 | WAKEUPINSTANT | NVM block enters low-power mode when entering sleep.<br>NVM block exits low-power mode when exiting sleep. |
| 0x2 | Reserved | |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x3 | DISABLED | Auto power reduction disabled. |

**Bit 7 – MANW** Manual Write

Note that reset value of this bit is '1'.

| Value | Description |
|---|---|
| 0 | Writing to the last word in the page buffer will initiate a write operation to the page addressed by the last write operation. This includes writes to memory and auxiliary rows. |
| 1 | Write commands must be issued through the CTRLA.CMD register. |

**Bits 4:1 – RWS[3:0]** NVM Read Wait States

These bits control the number of wait states for a read operation. '0' indicates zero wait states, '1' indicates one wait state, etc., up to 15 wait states.

This register is initialized to 0 wait states. Software can change this value based on the NVM access time and system frequency.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.3. NVM Parameter

**Name:** PARAM
**Offset:** 0x08
**Reset:** 0x000XXXXX
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | RWWEEP[11:4] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | RWWEEP[3:0] | | | | | PSZ[2:0] | | |
| Access | R | R | R | R | | R | R | R |
| Reset | 0 | 0 | 0 | 0 | | x | x | x |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NVMP[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | NVMP[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

**Bits 31:20 – RWWEEP[11:0]** RWWEE Pages
Indicates the number of pages in the RWWEE address space.

**Bits 18:16 – PSZ[2:0]** Page Size
Indicates the page size. Not all devices of the device families will provide all the page sizes indicated in the table.

| Value | Name | Description |
|---|---|---|
| 0x0 – 0x2 | Reserved | - |
| 0x3 | 64 | 64 bytes |
| 0x4 – 0x7 | Reserved | - |

**Bits 15:0 – NVMP[15:0]** NVM Pages
Indicates the number of pages in the NVM main address space.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ERROR | READY |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ERROR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the ERROR interrupt enable.
This bit will read as the current value of the ERROR interrupt enable.

**Bit 0 – READY** NVM Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the READY interrupt enable.
This bit will read as the current value of the READY interrupt enable.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x10
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ERROR | READY |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ERROR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit sets the ERROR interrupt enable.
This bit will read as the current value of the ERROR interrupt enable.

**Bit 0 – READY** NVM Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit sets the READY interrupt enable.
This bit will read as the current value of the READY interrupt enable.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x14
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ERROR | READY |
| Access | | | | | | | R/W | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ERROR** Error

This flag is set on the occurrence of an NVME, LOCKE or PROGE error.
This bit can be cleared by writing a '1' to its bit location.

| Value | Description |
|---|---|
| 0 | No errors have been received since the last clear. |
| 1 | At least one error has occurred since the last clear. |

**Bit 0 – READY** NVM Ready

| Value | Description |
|---|---|
| 0 | The NVM controller is busy programming or erasing. |
| 1 | The NVM controller is ready to accept a new command. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.7. Status

**Name:** STATUS
**Offset:** 0x18
**Reset:** 0x0X00
**Property:** –

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SB |
| Access | | | | | | | | R |
| Reset | | | | | | | | x |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | NVME | LOCKE | PROGE | LOAD | PRM |
| Access | | | | R/W | R/W | R/W | R/W | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bit 8 – SB** Security Bit Status

| Value | Description |
|---|---|
| 0 | The Security bit is inactive. |
| 1 | The Security bit is active. |

**Bit 4 – NVME** NVM Error
This bit can be cleared by writing a '1' to its bit location.

| Value | Description |
|---|---|
| 0 | No programming or erase errors have been received from the NVM controller since this bit was last cleared. |
| 1 | At least one error has been registered from the NVM Controller since this bit was last cleared. |

**Bit 3 – LOCKE** Lock Error Status
This bit can be cleared by writing a '1' to its bit location.

| Value | Description |
|---|---|
| 0 | No programming of any locked lock region has happened since this bit was last cleared. |
| 1 | Programming of at least one locked lock region has happened since this bit was last cleared. |

**Bit 2 – PROGE** Programming Error Status
This bit can be cleared by writing a '1' to its bit location.

| Value | Description |
|---|---|
| 0 | No invalid commands or bad keywords were written in the NVM Command register since this bit was last cleared. |
| 1 | An invalid command and/or a bad keyword was/were written in the NVM Command register since this bit was last cleared. |

**Bit 1 – LOAD** NVM Page Buffer Active Loading
This bit indicates that the NVM page buffer has been loaded with one or more words. Immediately after an NVM load has been performed, this flag is set. It remains set until a page write or a page buffer clear (PBCLR) command is given.
This bit can be cleared by writing a '1' to its bit location.

**Bit 0 – PRM** Power Reduction Mode
This bit indicates the current NVM power reduction state. The NVM block can be set in power reduction mode in two ways: through the command interface or automatically when entering sleep with SLEEPPRM set accordingly.


© 2026 Microchip Technology Inc. and its subsidiaries

PRM can be cleared in three ways: through AHB access to the NVM block, through the command interface (SPRM and CPRM) or when exiting sleep with SLEEPPRM set accordingly.

| Value | Description |
|---|---|
| 0 | NVM is not in power reduction mode. |
| 1 | NVM is in power reduction mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.8. Address

**Name:** ADDR
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | ADDR[16] |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 16:0 – ADDR[16:0]** NVM Address

ADDR drives the hardware half-word offset from the start address of the corresponding NVM section when a command is executed using CMDEX. This register is also automatically updated when writing to the page buffer. The effective address for the operation is Start address of the section + 2*ADDR.

**Example:**

For erasing the 3rd row in the Flash memory, spanning from 0x00000200 to 0x000002FF, ADDR must be written with the half-word offset address of any half-word within this range, that is any value between 0x100 and 0x17F.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.9. Lock Section

**Name:** LOCK
**Offset:** 0x20
**Reset:** 0xXXXX
**Property:** –

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | LOCK[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | LOCK[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

**Bits 15:0 – LOCK[15:0]** Region Lock Bits
To set or clear these bits, the CMD register must be used.
Default state after erase will be unlocked (0xFFFF).
Default state after reset will be loaded from the NVM User Row.

| Value | Description |
|---|---|
| 0 | The corresponding lock region is locked. |
| 1 | The corresponding lock region is not locked. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.10. Page Buffer Load Data 0

**Name:** PBLDATA0
**Offset:** 0x28
**Reset:** 0xFFFFFFFF
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 31:0 – PBLDATA[31:0]** Page Buffer Load Data
The PBLDATA register is a holding register for partial AHB writes to the same 64-bit page buffer section. Page buffer loads are performed on a 64-bit basis.
This is a read only register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 27.8.11. Page Buffer Load Data 1

**Name:** PBLDATA1
**Offset:** 0x2C
**Reset:** 0xFFFFFFFF
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PBLDATA[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 31:0 – PBLDATA[31:0]** Page Buffer Load Data (Bits 63:32])
The PBLDATA register is a holding register for partial AHB writes to the same 64-bit page buffer section. Page buffer loads are performed on a 64-bit basis.
This is a read only register.


© 2026 Microchip Technology Inc. and its subsidiaries

# 28. PORT - I/O Pin Controller

## 28.1. Overview

The I/O Pin Controller (PORT) controls the I/O pins of the device. The I/O pins are organized in a series of groups, collectively referred to as a PORT group. Each PORT group can have up to 32 pins that can be configured and controlled individually or as a group. The number of PORT groups on a device may depend on the package or number of pins. Each pin may either be used for general purpose I/O under direct application control or be assigned to an embedded device peripheral. When used for general purpose I/O, each pin can be configured as input or output, with a highly configurable driver and pull settings.

All I/O pins have true read-modify-write functionality when used for general purpose I/O. The direction or the output value of one or more pins may be changed (set, Reset or toggled) explicitly without unintentionally changing the state of any other pins in the same port group by a single, atomic 8-, 16- or 32-bit write.

The PORT is connected to the high-speed bus matrix through an AHB/APB bridge. The Pin Direction, Data Output Value and Data Input Value registers may also be accessed using the low-latency CPU local bus (IOBUS; ARM® single-cycle I/O port). .

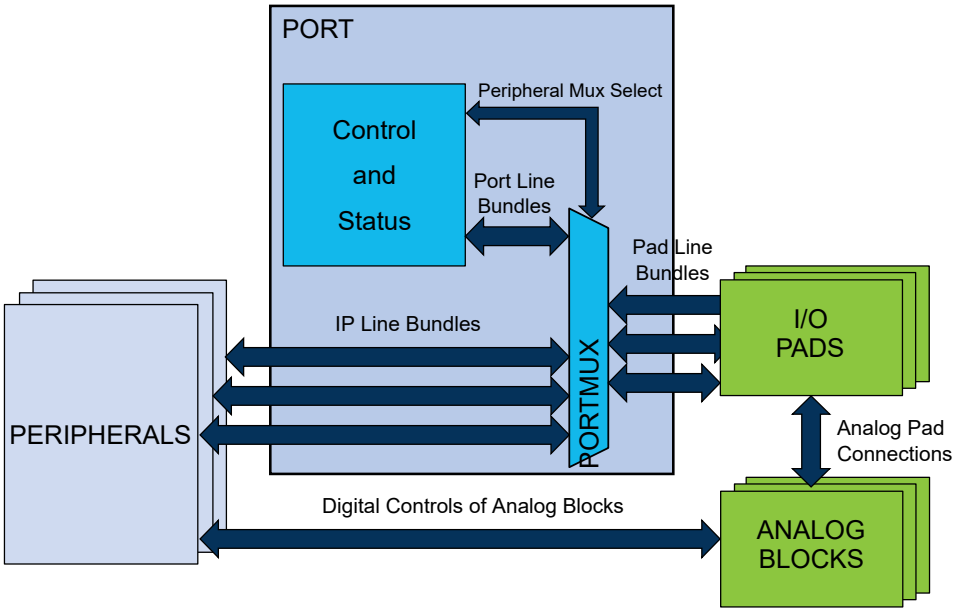
## 28.2. Features

- Selectable Input and Output Configuration for Each Individual Pin
- Software-controlled Multiplexing of Peripheral Functions on I/O Pins
- Flexible Pin Configuration Through a Dedicated Pin Configuration Register
- Configurable Output Driver and Pull Settings:
  - Totem-pole (push-pull)
  - Pull configuration
  - Driver strength
- Configurable Input Buffer and Pull Settings:
  - Internal pull up or pull down
  - Input sampling criteria
  - Input buffer can be disabled if not needed for low-power consumption
  - Read-Modify-Write support for output value (OUTCLR/OUTSET/OUTTGL) and pin direction (DIRCLR/DIRSET/DIRTGL)
- Input Event:
  - Up to four input event pins for each PORT group
  - SET/CLEAR/TOGGLE event actions for each event input on output value of a pin
  - Can be output to pin


© 2026 Microchip Technology Inc. and its subsidiaries

## 28.3. Block Diagram

**Figure 28-1.** PORT Block Diagram




## 28.4. Signal Description

**Table 28-1.** Signal description for PORT

| Signal name | Type | Description |
|---|---|---|
| Pxy | Digital I/O | General-purpose I/O pin y in group x |

Refer to the *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**
I/O Multiplexing and Considerations

## 28.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly as follows.

### 28.5.1. I/O Lines

The I/O lines of the PORT are mapped to pins of the physical device. The following naming scheme is used:

Each line bundle with up to 32 lines is assigned an identifier 'xy', with letter x=A, B, C... and two-digit number y=00, 01, ...31. Examples: A24, C03.

PORT pins are labeled 'Pxy' accordingly, for example PA24, PC03. This identifies each pin in the device uniquely.

Each pin may be controlled by one or more peripheral multiplexer settings, which allow the pad to be routed internally to a dedicated peripheral function. When the setting is enabled, the selected peripheral has control over the output state of the pad, as well as the ability to read the current physical pad state. Refer to *I/O Multiplexing and Considerations* for details.

Device-specific configurations may cause some lines (and the corresponding Pxy pin) not to be implemented.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
I/O Multiplexing and Considerations

### 28.5.2. Power Management

During Reset, all PORT lines are configured as inputs with input buffers, output buffers and pull disabled.

The PORT peripheral will continue operating in any Sleep mode where its source clock is running.

### 28.5.3. Clocks

The PORT bus clock (CLK_PORT_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_PORT_APB can be found in the *Peripheral Clock Masking* section in *MCLK – Main Clock*.

The PORT requires an APB clock, which may be divided from the CPU main clock and allows the CPU to access the registers of PORT through the high-speed matrix and the AHB/APB bridge.

One clock cycle latency can be observed on the APB access in case of concurrent PORT accesses.

**Related Links**
MCLK – Main Clock

### 28.5.4. DMA

Not applicable.

### 28.5.5. Interrupts

Not applicable.

### 28.5.6. Events

The events of this peripheral are connected to the Event System.

**Related Links**
Event System (EVSYS)

### 28.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation.

### 28.5.8. Register Access Protection

All registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC).

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

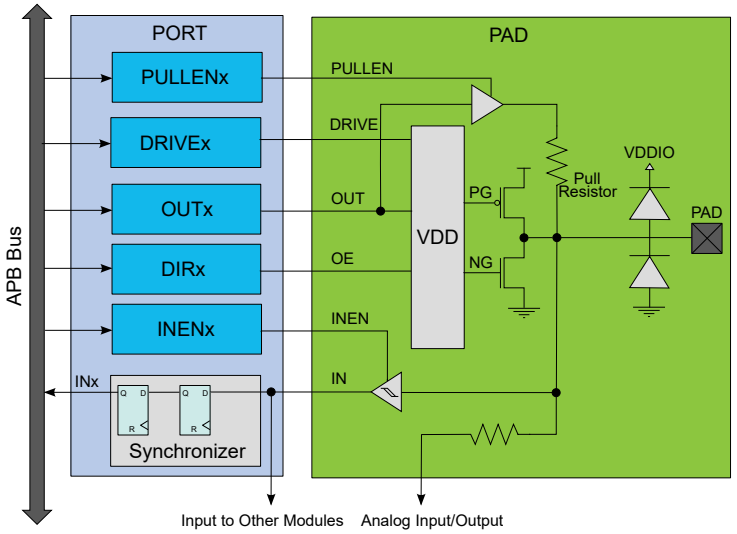
### 28.5.9. Analog Connections

Analog functions are connected directly between the analog blocks and the I/O pads using analog buses. Selecting an analog peripheral function for a given pin will disable the corresponding digital features of the pad.


© 2026 Microchip Technology Inc. and its subsidiaries

## 28.6. Functional Description

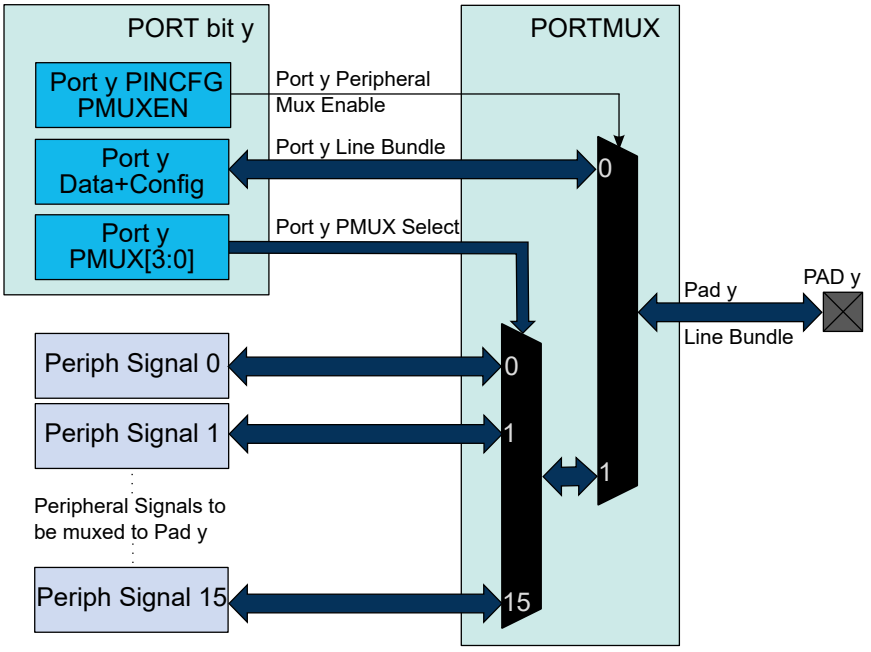
**Figure 28-2.** Overview of the PORT




### 28.6.1. Principle of Operation

Each PORT group of up to 32 pins is controlled by the registers in PORT, as described in the figure below. These registers in PORT are duplicated for each PORT group, with increasing base addresses using an offset of 0x80 between groups. The number of PORT groups may depend on the package or number of pins.

**Figure 28-3.** Overview of the peripheral functions multiplexing




The I/O pins of the device are controlled by the PORT peripheral registers. Each port pin has a corresponding bit in the Data Direction (DIR) and Data Output Value (OUT) registers to enable that pin as an output and to define the Output state.


© 2026 Microchip Technology Inc. and its subsidiaries

The direction of each pin in a PORT group is configured by the DIR register. If a bit in DIR is set to '1', the corresponding pin is configured as an output pin. If a bit in DIR is set to '0', the corresponding pin is configured as an input pin.

When the direction is set as output, the corresponding bit in the OUT register will set the level of the pin. If bit y in OUT is written to '1', pin y is driven HIGH. If bit y in OUT is written to '0', pin y is driven LOW. Pin configuration can be set by the Pin Configuration (PINCFGy) registers, with y = 00, 01, ..31 representing the pin position within the group.

The Data Input Value (IN) is set as the input value of a port pin with resynchronization to the PORT clock. To reduce power consumption, these input synchronizers can be clocked only when system requires reading the input value, as specified in the SAMPLING field of the Control register (CTRL). The value of the pin can always be read, whether the pin is configured as input or output. If the Input Enable bit in the Pin Configuration registers (PINCFGy.INEN) is '0', the input value will not be sampled.

In PORT, the Peripheral Multiplexer Enable bit in the PINCFGy register (PINCFGy.PMUXEN) can be written to '1' to enable the connection between peripheral functions and individual I/O pins. The Peripheral Multiplexing n (PMUXn) registers select the peripheral function for the corresponding pin. This will override the connection between the PORT and that I/O pin, and connect the selected peripheral signal to the particular I/O pin instead of the PORT line bundle.

## 28.6.2. Basic Operation

### 28.6.2.1. Initialization

After reset, all standard function device I/O pads are connected to the PORT with outputs tri-stated and input buffers disabled, even if there is no clock running.

However, specific pins, such as those used for connection to a debugger, may be configured differently, as required by their special function.

### 28.6.2.2. Operation

Each I/O pin Pxy can be controlled by the registers in PORT. Each PORT group x has its own set of PORT registers, with a base address at byte address (PORT + 0x80 * group index) (A corresponds to group index 0, B to 1, etc...). Within that set of registers, the pin index is y, from 0 to 31.

Refer to *I/O Multiplexing and Considerations* for details on available pin configuration and PORT groups.

#### Configuring Pins as Output

To use pin Pxy as an *output*, write bit y of the DIR register to '1'. This can also be done by writing bit y in the DIRSET register to '1' - this will avoid disturbing the configuration of other pins in that group. The y bit in the OUT register must be written to the desired output value.

Similarly, writing an OUTSET bit to '1' will set the corresponding bit in the OUT register to '1'. Writing a bit in OUTCLR to '1' will set that bit in OUT to zero. Writing a bit in OUTTGL to '1' will toggle that bit in OUT.

#### Configuring Pins as Input

To use pin Pxy as an *input*, bit y in the DIR register must be written to '0'. This can also be done by writing bit y in the DIRCLR register to '1' - this will avoid disturbing the configuration of other pins in that group. The input value can be read from bit y in register IN as soon as the INEN bit in the Pin Configuration register (PINCFGy.INEN) is written to '1'.

By default, the input synchronizer is clocked only when an input read is requested. This will delay the read operation by two cycles of the PORT clock. To remove the delay, the input synchronizers for each PORT group of eight pins can be configured to be always active, but this will increase power consumption. This is enabled by writing '1' to the corresponding SAMPLINGn bit field of the CTRL register, see CTRL.SAMPLING for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### Using Alternative Peripheral Functions

To use pin Pxy as one of the available peripheral functions, the corresponding PMUXEN bit of the PINCFGy register must be '1'. The PINCFGy register for pin Pxy is at byte offset (PINCFG0 + y).

The peripheral function can be selected by setting the PMUXO or PMUXE in the PMUXn register. The PMUXO/PMUXE is at byte offset PMUX0 + (y/2). The chosen peripheral must also be configured and enabled.

### Related Links

I/O Multiplexing and Considerations

## 28.6.3. I/O Pin Configuration

The Pin Configuration register (PINCFGy) is used for additional I/O pin configuration. A pin can be set in a totem-pole or pull configuration.

As pull configuration is done through the Pin Configuration register, all intermediate PORT states during switching of pin direction and pin values are avoided.

The I/O pin configurations are described further in this chapter, and summarized in Table 28-2.

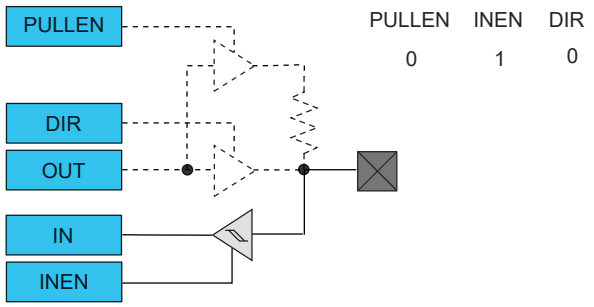
### 28.6.3.1. Pin Configurations Summary

**Table 28-2.** Pin Configurations Summary

| DIR | INEN | PULLEN | OUT | Configuration |
|---|---|---|---|---|
| 0 | 0 | 0 | X | Reset or analog I/O: all digital disabled |
| 0 | 0 | 1 | 0 | Pull-down; input buffer disabled |
| 0 | 0 | 1 | 1 | Pull-up; input buffer disabled |
| 0 | 1 | 0 | X | Input |
| 0 | 1 | 1 | 0 | Input with pull-down |
| 0 | 1 | 1 | 1 | Input with pull-up |
| 1 | 0 | X | X | Output; input buffer disabled |
| 1 | 1 | X | X | Output; input buffer enabled |

### 28.6.3.2. Input Configuration

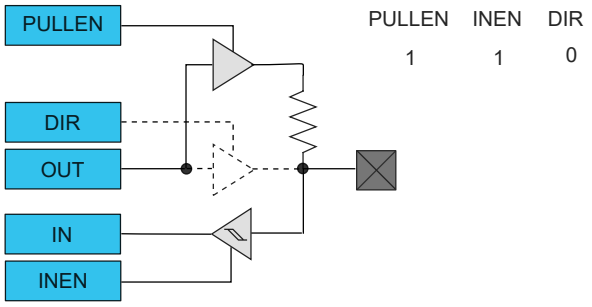
**Figure 28-4.** I/O configuration - Standard Input





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 28-5.** I/O Configuration - Input with Pull




**Note:** When pull is enabled, the pull value is defined by the OUT value.

### 28.6.3.3. Totem-Pole Output

When configured for totem-pole (push-pull) output, the pin is driven low or high according to the corresponding bit setting in the OUT register. In this configuration there is no current limitation for sink or source other than what the pin is capable of. If the pin is configured for input, the pin will float if no external pull is connected.

**Note:** Enabling the output driver will automatically disable pull.

**Figure 28-6.** I/O Configuration - Totem-Pole Output with Disabled Input

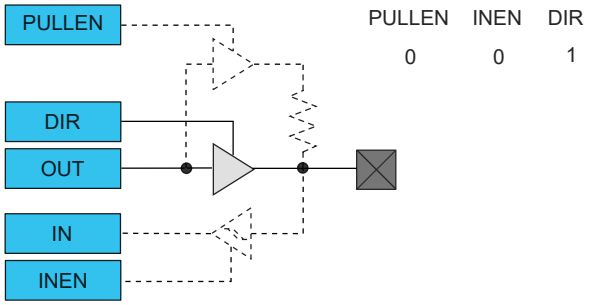
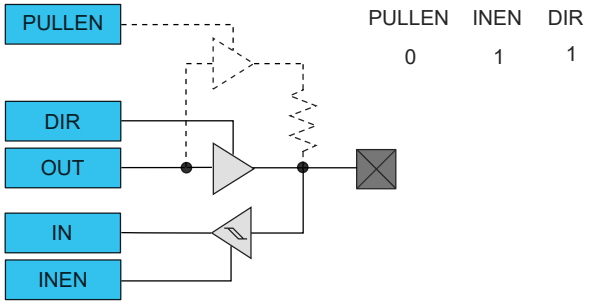


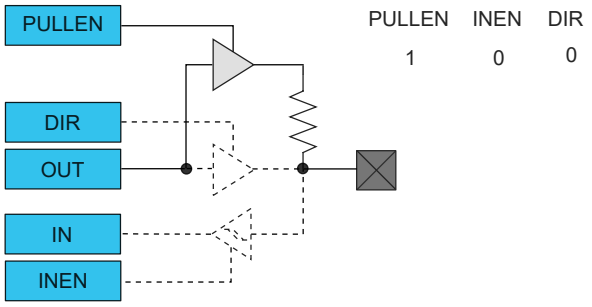

**Figure 28-7.** I/O Configuration - Totem-Pole Output with Enabled Input





© 2026 Microchip Technology Inc. and its subsidiaries

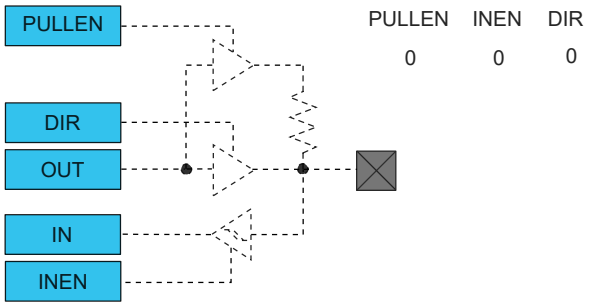
**Figure 28-8.** I/O Configuration - Output with Pull




### 28.6.3.4. Digital Functionality Disabled

Neither Input nor Output functionality are enabled.

**Figure 28-9.** I/O Configuration - Reset or Analog I/O: Digital Output, Input and Pull Disabled




## 28.6.4. Events

The PORT allows input events to control individual I/O pins. These input events (EVU0-3) are generated by the EVSYS module and can originate from a different clock domain than the PORT module's clock domain.

The PORT can perform the following actions:

- Output (OUT): I/O pin will be set when the incoming event has a high level ('1') and cleared when the incoming event has a low-level ('0').
- Set (SET): I/O pin will be set when an incoming event is detected.
- Clear (CLR): I/O pin will be cleared when an incoming event is detected.
- Toggle (TGL): I/O pin will toggle when an incoming event is detected.

The Output event is sent to the pin without any internal latency. For SET, CLEAR, and TOGGLE event actions, the action will be executed up to three clock cycles after a rising edge.

**Note:** In Standby mode, only the Out action is possible, and the Set, Clear, and Toggle actions are not available.

The event actions can be configured with the Event Action m bit group in the Event Input Control register( EVCTRL.EVACTn). Writing a '1' to a PORT Event Enable Input m of the Event Control register (EVCTRL.PORTEIn) enables the corresponding action on input event. Writing '0' to this bit disables the corresponding action on input event. Several actions can be enabled for incoming events. If several events are connected to the peripheral, any enabled action will be taken for any of the incoming events. Refer to the section " EVSYS – Event System" for additional information on configuring the Event System.

Each event input can address one and only one I/O pin at a time. The selection of the pin is indicated by the PORT Event Pin Identifier of the Event Input Control register (EVCTR.PIDn). On the other hand, one I/O pin can be addressed by up to four different input events. To avoid action conflict on the


© 2026 Microchip Technology Inc. and its subsidiaries

output value of the register (OUT) of this particular I/O pin, only one action is performed according to the table below. This truth table can be applied to any SET/CLR/TGL configuration from two to four active input events.

**Table 28-3.** Priority on Simultaneous SET/CLR/TGL Event Actions

| EVACT0 | EVACT1 | EVACT2 | EVACT3 | Executed Event Action |
|---|---|---|---|---|
| SET | SET | SET | SET | SET |
| CLR | CLR | CLR | CLR | CLR |
| All Other Combinations | | | | TGL |

Be careful when the event is Output (OUT) to pin. Because the events are received asynchronously, the I/O pin may have unpredictable levels, depending on the timing of when the events are received. When several events are output to the same pin, the lowest event line will get the access. All other events will be ignored.

**Related Links**
Event System (EVSYS)

### 28.6.5. PORT Access Priority

The PORT is accessed by different systems:

- The ARM® CPU through the ARM® single-cycle I/O port (IOBUS)
- The ARM® CPU through the high-speed matrix and the AHB/APB bridge (APB)
- EVSYS through four asynchronous input events

The CPU local bus (IOBUS) is an interface that connects the CPU directly to the PORT. It is a single-cycle bus interface, which does not support wait states. It supports 8-bit, 16-bit and 32-bit sizes.

This bus is generally used for low-latency operation. The Data Direction (DIR) and Data Output Value (OUT) registers can be read, written, set, cleared or be toggled using this bus, and the Data Input Value (IN) registers can be read.

Since the IOBUS cannot wait for IN register resynchronization, the Control register (CTRL) must be configured to continuous sampling of all pins that need to be read via the IOBUS in order to prevent stale data from being read.

**Note:** Refer to the Product Mapping chapter for the PORT IOBUS address.

The following priority is adopted:

1. ARM® CPU IOBUS (No wait tolerated).
2. APB.
3. EVSYS input events, except for events with EVCTRL.EVACTn = OUT, where the output pin follows the event input signal, independently of the OUT register value.

**Note:** One clock cycle latency can be observed on the APB access in case of concurrent PORT accesses.

For input events that require different actions on the same I/O pin, refer to Events


© 2026 Microchip Technology Inc. and its subsidiaries

## 28.7. Register Summary

The I/O pins are assembled in pin groups with up to 32 pins. Group 0 consists of the PA pins, and group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing between groups. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | DIR | 7:0 | DIR[7:0] | | | | | | | |
| | | 15:8 | DIR[15:8] | | | | | | | |
| | | 23:16 | DIR[23:16] | | | | | | | |
| | | 31:24 | DIR[31:24] | | | | | | | |
| 0x04 | DIRCLR | 7:0 | DIRCLR[7:0] | | | | | | | |
| | | 15:8 | DIRCLR[15:8] | | | | | | | |
| | | 23:16 | DIRCLR[23:16] | | | | | | | |
| | | 31:24 | DIRCLR[31:24] | | | | | | | |
| 0x08 | DIRSET | 7:0 | DIRSET[7:0] | | | | | | | |
| | | 15:8 | DIRSET[15:8] | | | | | | | |
| | | 23:16 | DIRSET[23:16] | | | | | | | |
| | | 31:24 | DIRSET[31:24] | | | | | | | |
| 0x0C | DIRTGL | 7:0 | DIRTGL[7:0] | | | | | | | |
| | | 15:8 | DIRTGL[15:8] | | | | | | | |
| | | 23:16 | DIRTGL[23:16] | | | | | | | |
| | | 31:24 | DIRTGL[31:24] | | | | | | | |
| 0x10 | OUT | 7:0 | OUT[7:0] | | | | | | | |
| | | 15:8 | OUT[15:8] | | | | | | | |
| | | 23:16 | OUT[23:16] | | | | | | | |
| | | 31:24 | OUT[31:24] | | | | | | | |
| 0x14 | OUTCLR | 7:0 | OUTCLR[7:0] | | | | | | | |
| | | 15:8 | OUTCLR[15:8] | | | | | | | |
| | | 23:16 | OUTCLR[23:16] | | | | | | | |
| | | 31:24 | OUTCLR[31:24] | | | | | | | |
| 0x18 | OUTSET | 7:0 | OUTSET[7:0] | | | | | | | |
| | | 15:8 | OUTSET[15:8] | | | | | | | |
| | | 23:16 | OUTSET[23:16] | | | | | | | |
| | | 31:24 | OUTSET[31:24] | | | | | | | |
| 0x1C | OUTTGL | 7:0 | OUTTGL[7:0] | | | | | | | |
| | | 15:8 | OUTTGL[15:8] | | | | | | | |
| | | 23:16 | OUTTGL[23:16] | | | | | | | |
| | | 31:24 | OUTTGL[31:24] | | | | | | | |
| 0x20 | IN | 7:0 | IN[7:0] | | | | | | | |
| | | 15:8 | IN[15:8] | | | | | | | |
| | | 23:16 | IN[23:16] | | | | | | | |
| | | 31:24 | IN[31:24] | | | | | | | |
| 0x24 | CTRL | 7:0 | SAMPLING[7:0] | | | | | | | |
| | | 15:8 | SAMPLING[15:8] | | | | | | | |
| | | 23:16 | SAMPLING[23:16] | | | | | | | |
| | | 31:24 | SAMPLING[31:24] | | | | | | | |
| 0x28 | WRCONFIG | 7:0 | PINMASK[7:0] | | | | | | | |
| | | 15:8 | PINMASK[15:8] | | | | | | | |
| | | 23:16 | | DRVSTR | | | | PULLEN | INEN | PMUXEN |
| | | 31:24 | HWSEL | WRPINCFG | | WRPMUX | | PMUX[3:0] | | |
| 0x2C | EVCTRL | 7:0 | PORTEI0 | EVACT0[1:0] | | PID0[4:0] | | | | |
| | | 15:8 | PORTEI1 | EVACT1[1:0] | | PID1[4:0] | | | | |
| | | 23:16 | PORTEI2 | EVACT2[1:0] | | PID2[4:0] | | | | |
| | | 31:24 | PORTEI3 | EVACT3[1:0] | | PID3[4:0] | | | | |
| 0x30 | PMUX0 | 7:0 | PMUXO[3:0] | | | | PMUXE[3:0] | | | |
| ... | | | | | | | | | | |
| 0x3F | PMUX15 | 7:0 | PMUXO[3:0] | | | | PMUXE[3:0] | | | |
| 0x40 | PINCFG0 | 7:0 | | DRVSTR | | | | PULLEN | INEN | PMUXEN |
| ... | | | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x5F | PINCFG31 | 7:0 | | DRVSTR | | | | PULLEN | INEN | PMUXEN |

## 28.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.1. Data Direction

**Name:** DIR
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to configure one or more I/O pins as an input or output. This register can be manipulated without doing a read-modify-write operation by using the Data Direction Toggle (DIRTGL), Data Direction Clear (DIRCLR) and Data Direction Set (DIRSET) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIR[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIR[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIR[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIR[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIR[31:0]** Port Data Direction
These bits set the data direction for the individual I/O pins in the PORT group.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group is configured as an input. |
| 1 | The corresponding I/O pin in the PORT group is configured as an output. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.2. Data Direction Clear

**Name:** DIRCLR
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to set one or more I/O pins as an input, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Direction (DIR), Data Direction Toggle (DIRTGL) and Data Direction Set (DIRSET) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIRCLR[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIRCLR[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIRCLR[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIRCLR[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIRCLR[31:0]** Port Data Direction Clear
Writing a '0' to a bit has no effect.
Writing a '1' to a bit will clear the corresponding bit in the DIR register, which configures the I/O pin as an input.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group will keep its configuration. |
| 1 | The corresponding I/O pin in the PORT group is configured as input. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.3. Data Direction Set

**Name:** DIRSET
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to set one or more I/O pins as an output, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Direction (DIR), Data Direction Toggle (DIRTGL) and Data Direction Clear (DIRCLR) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIRSET[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIRSET[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIRSET[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIRSET[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIRSET[31:0]** Port Data Direction Set
Writing '0' to a bit has no effect.
Writing '1' to a bit will set the corresponding bit in the DIR register, which configures the I/O pin as an output.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group will keep its configuration. |
| 1 | The corresponding I/O pin in the PORT group is configured as an output. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.4. Data Direction Toggle

**Name:** DIRTGL
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to toggle the direction of one or more I/O pins, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Direction (DIR), Data Direction Set (DIRSET) and Data Direction Clear (DIRCLR) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DIRTGL[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DIRTGL[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIRTGL[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DIRTGL[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – DIRTGL[31:0]** Port Data Direction Toggle
Writing '0' to a bit has no effect.
Writing '1' to a bit will toggle the corresponding bit in the DIR register, which reverses the direction of the I/O pin.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group will keep its configuration. |
| 1 | The direction of the corresponding I/O pin is toggled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.5. Data Output Value

**Name:** OUT
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register sets the data output drive value for the individual I/O pins in the PORT.

This register can be manipulated without doing a read-modify-write operation by using the Data Output Value Clear (OUTCLR), Data Output Value Set (OUTSET), and Data Output Value Toggle (OUTTGL) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | OUT[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OUT[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OUT[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OUT[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – OUT[31:0]** PORT Data Output Value
For pins configured as outputs via the Data Direction register (DIR), these bits set the logical output drive level.
For pins configured as inputs via the Data Direction register (DIR) and with pull enabled via the Pull Enable bit in the Pin Configuration register (PINCFG.PULLEN), these bits will set the input pull direction.

| Value | Description |
|---|---|
| 0 | The I/O pin output is driven low, or the input is connected to an internal pull-down. |
| 1 | The I/O pin output is driven high, or the input is connected to an internal pull-up. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.6. Data Output Value Clear

**Name:** OUTCLR
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to set one or more output I/O pin drive levels low, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Output Value (OUT), Data Output Value Toggle (OUTTGL) and Data Output Value Set (OUTSET) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | OUTCLR[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OUTCLR[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OUTCLR[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OUTCLR[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – OUTCLR[31:0]** PORT Data Output Value Clear
Writing '0' to a bit has no effect.
Writing '1' to a bit will clear the corresponding bit in the OUT register. Pins configured as outputs via the Data Direction register (DIR) will be set to low output drive level. Pins configured as inputs via DIR and with pull enabled via the Pull Enable bit in the Pin Configuration register (PINCFG.PULLEN) will set the input pull direction to an internal pull-down.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group will keep its configuration. |
| 1 | The corresponding I/O pin output is driven low, or the input is connected to an internal pull-down. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.7. Data Output Value Set

**Name:** OUTSET
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to set one or more output I/O pin drive levels high, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Output Value (OUT), Data Output Value Toggle (OUTTGL) and Data Output Value Clear (OUTCLR) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | OUTSET[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OUTSET[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OUTSET[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OUTSET[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – OUTSET[31:0]** PORT Data Output Value Set
Writing '0' to a bit has no effect.
Writing '1' to a bit will set the corresponding bit in the OUT register, which sets the output drive level high for I/O pins configured as outputs via the Data Direction register (DIR). For pins configured as inputs via Data Direction register (DIR) with pull enabled via the Pull Enable register (PULLEN), these bits will set the input pull direction to an internal pull-up.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the group will keep its configuration. |
| 1 | The corresponding I/O pin output is driven high, or the input is connected to an internal pull-up. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.8. Data Output Value Toggle

**Name:** OUTTGL
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to toggle the drive level of one or more output I/O pins, without doing a read-modify-write operation. Changes in this register will also be reflected in the Data Output Value (OUT), Data Output Value Set (OUTSET) and Data Output Value Clear (OUTCLR) registers.

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | OUTTGL[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OUTTGL[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OUTTGL[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OUTTGL[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – OUTTGL[31:0]** PORT Data Output Value Toggle
Writing '0' to a bit has no effect.
Writing '1' to a bit will toggle the corresponding bit in the OUT register, which inverts the output drive level for I/O pins configured as outputs via the Data Direction register (DIR). For pins configured as inputs via Data Direction register (DIR) with pull enabled via the Pull Enable register (PULLEN), these bits will toggle the input pull direction.

| Value | Description |
|---|---|
| 0 | The corresponding I/O pin in the PORT group will keep its configuration. |
| 1 | The corresponding OUT bit value is toggled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.9. Data Input Value

**Name:** IN
**Offset:** 0x20
**Reset:** 0x00000000

**Tip:** The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | IN[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | IN[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | IN[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IN[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – IN[31:0]** PORT Data Input Value

These bits are cleared when the corresponding I/O pin input sampler detects a logical low level on the input pin.
These bits are set when the corresponding I/O pin input sampler detects a logical high level on the input pin.


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.10. Control

**Name:** CTRL
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** PAC Write-Protection

**Tip:** The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SAMPLING[31:24] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SAMPLING[23:16] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SAMPLING[15:8] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SAMPLING[7:0] | | | | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – SAMPLING[31:0]** Input Sampling Mode
Configures the input sampling functionality of the I/O pin input samplers, for pins configured as inputs via the Data Direction register (DIR).
The input samplers are enabled and disabled in sub-groups of eight. Thus if any pins within a byte request continuous sampling, all pins in that eight pin sub-group will be continuously sampled.

| Value | Description |
|---|---|
| 0 | On demand sampling of I/O pin is enabled. |
| 1 | Continuous sampling of I/O pin is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.11. Write Configuration

**Name:** WRCONFIG
**Offset:** 0x28
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Only

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

This write-only register is used to configure several pins simultaneously with the same configuration and/or peripheral multiplexing.

In order to avoid side effect of non-atomic access, 8-bit or 16-bit writes to this register will have no effect. Reading this register always returns zero.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | HWSEL | WRPINCFG | | WRPMUX | PMUX[3:0] | | | |
| Access | W | W | | W | W | W | W | W |
| Reset | 0 | 0 | | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | DRVSTR | | | | PULLEN | INEN | PMUXEN |
| Access | | W | | | | W | W | W |
| Reset | | 0 | | | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PINMASK[15:8] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PINMASK[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 31 – HWSEL** Half-Word Select
This bit selects the half-word field of a 32-PORT group to be reconfigured in the atomic write operation.
This bit will always read as zero.

| Value | Description |
|---|---|
| 0 | The lower 16 pins of the PORT group will be configured. |
| 1 | The upper 16 pins of the PORT group will be configured. |

**Bit 30 – WRPINCFG** Write PINCFG
This bit determines whether the atomic write operation will update the Pin Configuration register (PINCFGy) or not for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits.
Writing '0' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing '1' to this bit updates the configuration of the selected pins with the written WRCONFIG.DRVSTR, WRCONFIG.PULLEN, WRCONFIG.INEN, WRCONFIG.PMUXEN, and WRCONFIG.PINMASK values.
This bit will always read as zero.

| Value | Description |
|---|---|
| 0 | The PINCFGy registers of the selected pins will not be updated. |
| 1 | The PINCFGy registers of the selected pins will be updated. |

**Bit 28 – WRPMUX** Write PMUX
This bit determines whether the atomic write operation will update the Peripheral Multiplexing register (PMUXn) or not for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits.
Writing '0' to this bit has no effect.
Writing '1' to this bit updates the pin multiplexer configuration of the selected pins with the written WRCONFIG. PMUX value.
This bit will always read as zero.

| Value | Description |
|---|---|
| 0 | The PMUXn registers of the selected pins will not be updated. |
| 1 | The PMUXn registers of the selected pins will be updated. |

**Bits 27:24 – PMUX[3:0]** Peripheral Multiplexing
These bits determine the new value written to the Peripheral Multiplexing register (PMUXn) for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits, when the WRCONFIG.WRPMUX bit is set.
These bits will always read as zero.

**Bit 22 – DRVSTR** Output Driver Strength Selection
This bit determines the new value written to PINCFGy.DRVSTR for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits, when the WRCONFIG.WRPINCFG bit is set.
This bit will always read as zero.

**Bit 18 – PULLEN** Pull Enable
This bit determines the new value written to PINCFGy.PULLEN for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits, when the WRCONFIG.WRPINCFG bit is set.
This bit will always read as zero.

**Bit 17 – INEN** Input Enable
This bit determines the new value written to PINCFGy.INEN for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits, when the WRCONFIG.WRPINCFG bit is set.
This bit will always read as zero.

**Bit 16 – PMUXEN** Peripheral Multiplexer Enable
This bit determines the new value written to PINCFGy.PMUXEN for all pins selected by the WRCONFIG.PINMASK and WRCONFIG.HWSEL bits, when the WRCONFIG.WRPINCFG bit is set.
This bit will always read as zero.

**Bits 15:0 – PINMASK[15:0]** Pin Mask for Multiple Pin Configuration
These bits select the pins to be configured within the half-word group selected by the WRCONFIG.HWSEL bit.
These bits will always read as zero.

| Value | Description |
|---|---|
| 0 | The configuration of the corresponding I/O pin in the half-word group will be left unchanged. |
| 1 | The configuration of the corresponding I/O pin in the half-word PORT group will be updated. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.12. Event Input Control

**Name:** EVCTRL
**Offset:** 0x2C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

**Tip:** The I/O pins are assembled in pin groups ("PORT groups") with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

There are up to four input event pins for each PORT group. Each byte of this register addresses one Event input pin.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PORTEI3 | EVACT3[1:0] | | PID3[4:0] | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | PORTEI2 | EVACT2[1:0] | | PID2[4:0] | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| | PORTEI1 | EVACT1[1:0] | | PID1[4:0] | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | PORTEI0 | EVACT0[1:0] | | PID0[4:0] | | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7, 15, 23, 31 – PORTEIx** PORT Event Input Enable x [x = 3..0]

| Value | Description |
|---|---|
| 0 | The event action x (EVACTx) will not be triggered on any incoming event. |
| 1 | The event action x (EVACTx) will be triggered on any incoming event. |

**Bits 5:6, 13:14, 21:22, 29:30 – EVACTx** PORT Event Action x [x = 3..0]
These bits define the event action the PORT will perform on event input x. See also Table 28-4.

**Bits 0:4, 8:12, 16:20, 24:28 – PIDx** PORT Event Pin Identifier x [x = 3..0]
These bits define the I/O pin on which the event action will be performed, according to Table 28-5.

**Table 28-4.** PORT Event x Action ( x = [3..0] )

| Value | Name | Description |
|---|---|---|
| 0x0 | OUT | Output register of pin will be set to level of event. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 28-4.** PORT Event x Action ( x = [3..0] ) (continued)

| Value | Name | Description |
|---|---|---|
| 0x1 | SET | Set output register of pin on event. |
| 0x2 | CLR | Clear output register of pin on event. |
| 0x3 | TGL | Toggle output register of pin on event. |

**Table 28-5.** PORT Event x Pin Identifier ( x = [3..0] )

| Value | Name | Description |
|---|---|---|
| 0x0 | PIN0 | Event action to be executed on PIN 0. |
| 0x1 | PIN1 | Event action to be executed on PIN 1. |
| ... | ... | ... |
| 0x31 | PIN31 | Event action to be executed on PIN 31. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.13. Peripheral Multiplexing n

**Name:** PMUX
**Offset:** 0x30 + n*0x01 [n=0..15]
**Property:** PAC Write-Protection

**Tip:** The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

There are up to 16 Peripheral Multiplexing registers in each group, one for every set of two subsequent I/O lines. The n denotes the number of the set of I/O lines.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PMUXO[3:0] | | | | PMUXE[3:0] | | | |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:4 – PMUXO[3:0]** Peripheral Multiplexing for Odd-Numbered Pin
These bits select the peripheral function for odd-numbered pins (2*n + 1) of a PORT group, if the corresponding PINCFGy.PMUXEN bit is '1'.
Not all possible values for this selection may be valid. For more details, refer to the *I/O Multiplexing and Considerations*.

| PMUXO[3:0] | Name | Description |
|---|---|---|
| 0x0 | A | Peripheral function A selected |
| 0x1 | B | Peripheral function B selected |
| 0x2 | C | Peripheral function C selected |
| 0x3 | D | Peripheral function D selected |
| 0x4 | E | Peripheral function E selected |
| 0x5 | F | Peripheral function F selected |
| 0x6 | G | Peripheral function G selected |
| 0x7 | H | Peripheral function H selected |
| 0x8 | I | Peripheral function I selected |
| 0x9 | - | Reserved |
| 0xA | - | Reserved |
| 0xB | - | Reserved |
| 0xC | - | Reserved |
| 0xD | - | Reserved |
| 0xE-0xF | - | Reserved |

**Bits 3:0 – PMUXE[3:0]** Peripheral Multiplexing for Even-Numbered Pin
These bits select the peripheral function for even-numbered pins (2*n) of a PORT group, if the corresponding PINCFGy.PMUXEN bit is '1'.
Not all possible values for this selection may be valid. For more details, refer to the *I/O Multiplexing and Considerations.*

| PMUXE[3:0] | Name | Description |
|---|---|---|
| 0x0 | A | Peripheral function A selected |
| 0x1 | B | Peripheral function B selected |
| 0x2 | C | Peripheral function C selected |


© 2026 Microchip Technology Inc. and its subsidiaries

| **PMUXE** (continued) | | |
|---|---|---|
| PMUXE[3:0] | Name | Description |
| 0x3 | D | Peripheral function D selected |
| 0x4 | E | Peripheral function E selected |
| 0x5 | F | Peripheral function F selected |
| 0x6 | G | Peripheral function G selected |
| 0x7 | H | Peripheral function H selected |
| 0x8 | I | Peripheral function I selected |
| 0x9 | - | Reserved |
| 0xA | - | Reserved |
| 0xB | - | Reserved |
| 0xC | - | Reserved |
| 0xD | - | Reserved |
| 0xE-0xF | - | Reserved |

**Related Links**

I/O Multiplexing and Considerations


© 2026 Microchip Technology Inc. and its subsidiaries

### 28.8.14. Pin Configuration

**Name:** PINCFG
**Offset:** 0x40 + n*0x01 [n=0..31]
**Reset:** 0x40
**Property:** PAC Write-Protection

**Tip:** The I/O pins are assembled in pin groups (”PORT groups”) with up to 32 pins. Group 0 consists of the PA pins, group 1 is for the PB pins, etc. Each pin group has its own PORT registers, with a 0x80 address spacing. For example, the register address offset for the Data Direction (DIR) register for group 0 (PA00 to PA31) is 0x00, and the register address offset for the DIR register for group 1 (PB00 to PB31) is 0x80.

There are up to 32 Pin Configuration registers in each PORT group, one for each I/O line.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | DRVSTR | | | | PULLEN | INEN | PMUXEN |
| Access | | RW | | | | RW | RW | RW |
| Reset | | 1 | | | | 0 | 0 | 0 |

**Bit 6 – DRVSTR** Output Driver Strength Selection
This bit controls the output driver strength of an I/O pin configured as an output.
**Note:** The reset value for DRVSTR is 0 for PA24 and PA25 for devices with USB.

| Value | Description |
|---|---|
| 0 | Pin drive strength is set to normal drive strength. |
| 1 | Pin drive strength is set to stronger drive strength. |

**Bit 2 – PULLEN** Pull Enable
This bit enables the internal pull-up or pull-down resistor of an I/O pin configured as an input.

| Value | Description |
|---|---|
| 0 | Internal pull resistor is disabled, and the input is in a high-impedance configuration. |
| 1 | Internal pull resistor is enabled, and the input is driven to a defined logic level in the absence of external input. |

**Bit 1 – INEN** Input Enable
This bit controls the input buffer of an I/O pin configured as either an input or output.
Writing a zero to this bit disables the input buffer completely, preventing read-back of the Physical Pin state when the pin is configured as either an input or output.

| Value | Description |
|---|---|
| 0 | Input buffer for the I/O pin is disabled, and the input value will not be sampled. |
| 1 | Input buffer for the I/O pin is enabled, and the input value will be sampled when required. |

**Bit 0 – PMUXEN** Peripheral Multiplexer Enable
This bit enables or disables the peripheral multiplexer selection set in the Peripheral Multiplexing register (PMUXn) to enable or disable alternative peripheral control over an I/O pin direction and output drive value.
Writing a zero to this bit allows the PORT to control the pad direction through the Data Direction register (DIR) and output drive value through the Data Output Value register (OUT). The peripheral multiplexer value in PMUXn is ignored. Writing '1' to this bit enables the peripheral selection in PMUXn to control the pad. In this configuration, the Physical Pin state may still be read from the Data Input Value register (IN) if PINCFGn.INEN is set.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The peripheral multiplexer selection is disabled, and the PORT registers control the direction and output drive value. |
| 1 | The peripheral multiplexer selection is enabled, and the selected peripheral function controls the direction and output drive value. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 29. Event System (EVSYS)

## 29.1. Overview

The Event System allows autonomous, low-latency, and configurable communication between peripherals.

Several peripherals can be configured to generate and/or respond to signals known as events. The exact condition to generate an event, or the action taken upon receiving an event, is specific to each peripheral. Peripherals that respond to events are called event users. Peripherals that generate events are called event generators. A peripheral can have one or more event generators and can have one or more event users.

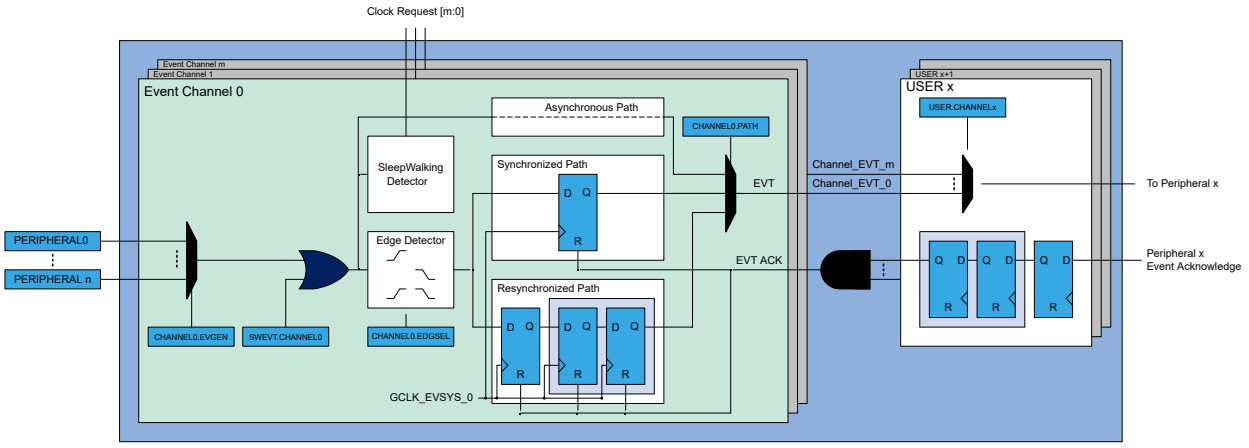
Communication is made without CPU intervention and without consuming system resources, such as bus or RAM bandwidth. This reduces the load on the CPU and other system resources, compared to a traditional interrupt-based system.

## 29.2. Features

- 12 configurable event channels:
  - Can be connected to any event generator
  - Can provide a pure asynchronous, resynchronized, or synchronous path
- 95 Event Generators
- 47 Event Users
- Configurable Edge Detector
- Peripherals can be Event Generators, Event Users, or both
- SleepWalking and interrupt for operation in sleep modes
- Software Event Generation
- Each Event User can choose which channel to respond to
- Each Event User can choose which channel to respond to, and several Event Users can share the same channel and therefore answer to the same event

## 29.3. Block Diagram

**Figure 29-1.** Event System Block Diagram




## 29.4. Signal Description

Not applicable.


© 2026 Microchip Technology Inc. and its subsidiaries

## 29.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 29.5.1. I/O Lines

Not applicable.

### 29.5.2. Power Management

The EVSYS can be used to wake up the CPU from all sleep modes, even if the clock used by the EVSYS channel and the EVSYS bus clock are disabled. Refer to the *PM – Power Manager* for details on the different sleep modes.

Although the clock for the EVSYS is stopped, the device still can wake up the EVSYS clock. Some event generators can generate an event when their clocks are stopped. The generic clock for the channel (GCLK_EVSYS_CHANNEL_n) will be restarted if that channel uses a synchronized path or a resynchronized path. It does not need to wake the system from sleep.

**Related Links**
PM - Power Manager

### 29.5.3. Clocks

The EVSYS bus clock (CLK_EVSYS_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_EVSYS_APB can be found in *Peripheral Clock Masking*.

Each EVSYS channel has a dedicated generic clock (GCLK_EVSYS_CHANNEL_n). These are used for event detection and propagation for each channel. These clocks must be configured and enabled in the generic clock controller before using the EVSYS. Refer to *GCLK - Generic Clock Controller* for details.

**Related Links**
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 29.5.4. DMA

Not applicable.

### 29.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the EVSYS interrupts requires the interrupt controller to be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

### 29.5.6. Events

Not applicable.

### 29.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging - refer to the Debug Control (DBGCTRL) register for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.5.8. Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except for the following:

- Channel Status (CHSTATUS)
- Interrupt Flag Status and Clear register (INTFLAG)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 29.5.9. Analog Connections

Not applicable.

## 29.6. Functional Description

### 29.6.1. Principle of Operation

The Event System consists of several channels which route the internal events from peripherals (generators) to other internal peripherals or I/O pins (users). Each event generator can be selected as source for multiple channels, but a channel cannot be set to use multiple event generators at the same time.

A channel path can be configured in asynchronous, synchronous or resynchronized mode of operation. The mode of operation must be selected based on the requirements of the application.

When using synchronous or resynchronized path, the Event System includes options to transfer events to users when rising, falling or both edges are detected on event generators.

For further details, refer to the Channel Path section of this chapter.

### 29.6.2. Basic Operation

#### 29.6.2.1. Initialization

Before enabling event routing within the system, the Event Users Multiplexer and Event Channels must be selected in the Event System (EVSYS), and the two peripherals that generate and use the event must be configured. Follow these steps to configure the event:

1. In the peripheral generating the event, enable output of event by writing a '1' to the respective Event Output Enable bit ("EO") in the peripheral's Event Control register, for example, AC.EVCTRL.WINEO0 or RTC.EVCTRL.OVFEO.
2. Configure the EVSYS:
   a. Configure the Event User multiplexer by writing the respective EVSYS.USERm register, refer to User Multiplexer Setup.
   b. Configure the Event Channel by writing the respective EVSYS.CHANNELn register, refer to Event System Channel.
3. Configure the action to be executed by the event user peripheral by writing to the Event Action bits (EVACT) in the respective Event control register, for example, TC.EVCTRL.EVACT, PDEC.EVCTRL.EVACT.
   **Note:** This step is not applicable for all the peripherals.
4. In the event user peripheral, enable event input by writing a '1' to the respective Event Input Enable bit ("EI") in the peripheral's Event Control register, for example, AC.EVCTRL.IVEI0, ADC.EVCTRL.STARTEI.

#### 29.6.2.2. Enabling, Disabling, and Resetting

The EVSYS is always enabled.


© 2026 Microchip Technology Inc. and its subsidiaries

The EVSYS is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the EVSYS will be reset to their initial state and all ongoing events will be canceled.

Refer to CTRLA.SWRST register for details.

#### 29.6.2.3. User Multiplexer Setup

The user multiplexer defines the channel to be connected to which event user. Each user multiplexer is dedicated to one event user. A user multiplexer receives all event channels output and must be configured to select one of these channels, as shown in Block Diagram section. The channel is selected with the Channel bit group in the User register (USERm.CHANNEL).

The user multiplexer must always be configured before the channel. A list of all user multiplexers is found in the User (USERm) register description.

#### 29.6.2.4. Event System Channel

An event channel can select one event from a list of event generators. Depending on configuration, the selected event could be synchronized, resynchronized or asynchronously sent to the users. When synchronization or resynchronization is required, the channel includes an internal edge detector, allowing the Event System to generate internal events when rising, falling or both edges are detected on the selected event generator.

An event channel is able to generate internal events for the specific software commands. A channel block diagram is shown in *Block Diagram* section.

#### 29.6.2.5. Event Generators

Each event channel can receive the events form all event generators. All event generators are listed in the Event Generator bit field in the Channel n register (CHANNELn.EVGEN). For details on event generation, refer to the corresponding module chapter. The channel event generator is selected by the Event Generator bit group in the Channel register (CHANNELn.EVGEN). By default, the channels are not connected to any event generators (ie, CHANNELn.EVGEN = 0)

#### 29.6.2.6. Channel Path

There are three different ways to propagate the event from an event generator:

- Asynchronous path
- Synchronous path
- Resynchronized path

The path is decided by writing to the Path Selection bit group of the Channel register (CHANNELn.PATH).

**Asynchronous Path**

When using the asynchronous path, the events are propagated from the event generator to the event user without intervention from the Event System. The GCLK for this channel (GCLK_EVSYS_CHANNEL_n) is not mandatory, meaning that an event will be propagated to the user without any clock latency.

When the asynchronous path is selected, the channel cannot generate any interrupts, and the Channel Status register (CHSTATUS) is always zero. The edge detection is not required and must be disabled by software. Each peripheral event user has to select which event edge must trigger internal actions. For further details, refer to each peripheral chapter description.

**Synchronous Path**

The synchronous path should be used when the event generator and the event channel share the same generator for the generic clock. If they do not share the same clock, a logic change from the event generator to the event channel might not be detected in the channel, which means that the event will not be propagated to the event user. For details on generic clock generators, refer to *GCLK - Generic Clock Controller*.


© 2026 Microchip Technology Inc. and its subsidiaries

When using the synchronous path, the channel is able to generate interrupts. The channel busy n bit in the Channel Status register (CHSTATUS.CHBUSYn) are also updated and available for use.

**Resynchronized Path**

The resynchronized path are used when the event generator and the event channel do not share the same generator for the generic clock. When the resynchronized path is used, resynchronization of the event from the event generator is done in the channel. For details on generic clock generators, refer to *GCLK - Generic Clock Controller*.

When the resynchronized path is used, the channel is able to generate interrupts. The channel busy n bits in the Channel Status register (CHSTATUS.CHBUSYn) are also updated and available for use.

**Related Links**

GCLK - Generic Clock Controller

#### 29.6.2.7. Edge Detection

When synchronous or resynchronized paths are used, edge detection must be enabled. The event system can execute edge detection in three different ways:

- Generate an event only on the rising edge
- Generate an event only on the falling edge
- Generate an event on rising and falling edges.

Edge detection is selected by writing to the Edge Selection bit group of the Channel register (CHANNELn.EDGSEL).

#### 29.6.2.8. Event Latency

The latency from event generator to event user depends on the channel's configuration:

- Asynchronous Path: The maximum routing latency of an external event is related to the internal signal routing and it is device dependent.
- Synchronous Path: The maximum routing latency of an external event is one GCLK_EVSYS_CHANNEL_n clock cycle.
- Resynchronized Path: The maximum routing latency of an external event is three GCLK_EVSYS_CHANNEL_n clock cycles.

The maximum propagation latency of a user event to the peripheral clock core domain is three peripheral clock cycles.

The event generators, event channel and event user clocks ratio must be selected in relation with the internal event latency constraints. Events propagation or event actions in peripherals may be lost if the clock setup violates the internal latencies.

#### 29.6.2.9. The Overrun Channel n Interrupt

The Overrun Channel n Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAGn.OVR) will be set, and the optional interrupt will be generated in the following cases:

- One or more event users on channel n is not ready when there is a new event
- An event occurs when the previous event on channel m has not been handled by all event users connected to that channel

The flag will only be set when using synchronous or resynchronized paths. In the case of asynchronous path, the INTFLAGn.OVR is always read as zero.

#### 29.6.2.10. The Event Detected Channel n Interrupt

The Event Detected Channel n Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAGn.EVD) is set when an event coming from the event generator configured on channel n is detected.


© 2026 Microchip Technology Inc. and its subsidiaries

The flag will only be set when using a synchronous or resynchronized path. In the case of an asynchronous path, the INTFLAGn.EVD is always zero.

#### 29.6.2.11. Channel Status

The Channel Status register (CHSTATUS) shows the status of the channels when using a synchronous or resynchronized path. There are two different status bits in CHSTATUS for each of the available channels:

- The CHSTATUSn.BUSYCH bit will be set when an event on the corresponding channel n has not been handled by all event users connected to that channel.
- The CHSTATUSn.RDYUSR bit will be set when all event users connected to the corresponding channel are ready to handle incoming events on that channel.

#### 29.6.2.12. Software Event

A software event can be initiated on a channel by setting the Channel n bit in the Software Event register (SWEVT.CHANNELn) to '1'. Then the software event can be serviced as any event generator; i.e., when a bit is set to '1', an event will be generated on the respective channel.

### 29.6.3. Interrupts

The EVSYS has the following interrupt sources:

- Overrun Channel n interrupt (OVRn): for details, refer to The Overrun Channel n Interrupt.
- Event Detected Channel n interrupt (EVDn): for details, refer to The Event Detected Channel n Interrupt.

These interrupts events are asynchronous wake-up sources. See *Sleep Mode Controller*. Each interrupt source has an interrupt flag which is in the Interrupt Flag Status and Clear (INTFLAG) register. The flag is set when the interrupt is issued. Each interrupt event can be individually enabled by setting a '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by setting a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt event is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt event is active until the interrupt flag is cleared, the interrupt is disabled, or the Event System is reset. See INTFLAG for details on how to clear interrupt flags.

All interrupt events from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. Refer to the *Nested Vector Interrupt Controller* for details. The event user must read the INTFLAG register to determine what the interrupt condition is.

Note that interrupts must be globally enabled for interrupt requests to be generated. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

Sleep Mode Controller

### 29.6.4. Sleep Mode Operation

The EVSYS can generate interrupts to wake up the device from any sleep modes.

To be able to run in Standby mode, the Run in Standby bit in the Channel register (CHANNELn.RUNSTDBY) must be set to '1'. When the Generic Clock On Demand bit in Channel register (CHANNELn.ONDEMAND) is set to '1' and the event generator is detected, the event channel will request its clock (GCLK_EVSYS_CHANNEL_n). The event latency for a resynchronized channel path will increase by two GCLK_EVSYS_CHANNEL_n clock (that is, up to five GCLK_EVSYS_CHANNEL_n clock cycles).

A channel will behave differently in different sleep modes regarding to CHANNELn.RUNSTDBY and CHANNELn.ONDEMAND, as shown in the table below:


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 29-1.** Event Channel Sleep Behavior

| CHANNELn.PATH | CHANNELn.ONDEMAND | CHANNELn.RUNSTDBY | Sleep Behavior |
|---|---|---|---|
| ASYNC | 0 | 0 | Only run in Idle Sleep mode if an event must be propagated. Disabled in Standby Sleep mode. |
| SYNC/RESYNC | 0 | 1 | Run in Idle and Standby Sleep modes. |
| SYNC/RESYNC | 1 | 0 | Only run in Idle Sleep mode if an event must be propagated. Disabled in Standby Sleep mode. Two GCLK_EVSYS_n latency added in the RESYNC path before the event is propagated internally. |
| SYNC/RESYNC | 1 | 1 | Run in Idle and Standby Sleep modes. Two GCLK_EVSYS_n latency added in the RESYNC path before the event is propagated internally. |


© 2026 Microchip Technology Inc. and its subsidiaries

## 29.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | | SWRST |
| 0x01 ... 0x0B | Reserved | | | | | | | | | |
| 0x0C | CHSTATUS | 7:0 | USRRDYn[7:0] | | | | | | | |
| | | 15:8 | | | | | USRRDYn[11:8] | | | |
| | | 23:16 | CHBUSYn[7:0] | | | | | | | |
| | | 31:24 | | | | | CHBUSYn[11:8] | | | |
| 0x10 | INTENCLR | 7:0 | OVRn[7:0] | | | | | | | |
| | | 15:8 | | | | | OVRn[11:8] | | | |
| | | 23:16 | EVDn[7:0] | | | | | | | |
| | | 31:24 | | | | | EVDn[11:8] | | | |
| 0x14 | INTENSET | 7:0 | OVRn[7:0] | | | | | | | |
| | | 15:8 | | | | | OVRn[11:8] | | | |
| | | 23:16 | EVDn[7:0] | | | | | | | |
| | | 31:24 | | | | | EVDn[11:8] | | | |
| 0x18 | INTFLAG | 7:0 | OVRn[7:0] | | | | | | | |
| | | 15:8 | | | | | OVRn[11:8] | | | |
| | | 23:16 | EVDn[7:0] | | | | | | | |
| | | 31:24 | | | | | EVDn[11:8] | | | |
| 0x1C | SWEVT | 7:0 | CHANNELn[7:0] | | | | | | | |
| | | 15:8 | | | | | CHANNELn[11:8] | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 | CHANNEL0 | 7:0 | EVGEN[7:0] | | | | | | | |
| | | 15:8 | ONDEMAND | RUNSTDBY | | | EDGSEL[1:0] | | PATH[1:0] | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| ... | | | | | | | | | | |
| 0x4C | CHANNEL11 | 7:0 | EVGEN[7:0] | | | | | | | |
| | | 15:8 | ONDEMAND | RUNSTDBY | | | EDGSEL[1:0] | | PATH[1:0] | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x50 ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | USER0 | 7:0 | CHANNEL[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| ... | | | | | | | | | | |
| 0x0144 | USER49 | 7:0 | CHANNEL[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 29.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Refer to *Register Access Protection* and *PAC - Peripheral Access Controller*.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SWRST |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – SWRST** Software Reset
Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the EVSYS to their initial state.
**Note:** Before applying a Software Reset it is recommended to disable the event generators.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.2. Channel Status

**Name:** CHSTATUS
**Offset:** 0x0C
**Reset:** 0x000000FF
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CHBUSYn[11:8] | | | |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CHBUSYn[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | USRRDYn[11:8] | | | |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | USRRDYn[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 27:16 – CHBUSYn[11:0]** Channel Busy n [n = 11..0]
This bit is cleared when channel n is idle.
This bit is set if an event on channel n has not been handled by all event users connected to channel n.

**Bits 11:0 – USRRDYn[11:0]** User Ready for Channel n [n = 11..0]
This bit is cleared when at least one of the event users connected to the channel is not ready.
This bit is set when all event users connected to channel n are ready to handle incoming events on channel n.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.3. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | EVDn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EVDn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVRn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVRn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 27:16 – EVDn[11:0]** Event Detected Channel n Interrupt Enable [n = 11..0]
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Event Detected Channel n Interrupt Enable bit, which disables the Event Detected Channel n interrupt.

| Value | Description |
|---|---|
| 0 | The Event Detected Channel n interrupt is disabled. |
| 1 | The Event Detected Channel n interrupt is enabled. |

**Bits 11:0 – OVRn[11:0]** Overrun Channel n Interrupt Enable[n = 11..0]
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Overrun Channel n Interrupt Enable bit, which disables the Overrun Channel n interrupt.

| Value | Description |
|---|---|
| 0 | The Overrun Channel n interrupt is disabled. |
| 1 | The Overrun Channel n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.4. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | EVDn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EVDn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVRn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVRn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 27:16 – EVDn[11:0]** Event Detected Channel n Interrupt Enable [n = 11..0]
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Event Detected Channel n Interrupt Enable bit, which enables the Event Detected Channel n interrupt.

| Value | Description |
|---|---|
| 0 | The Event Detected Channel n interrupt is disabled. |
| 1 | The Event Detected Channel n interrupt is enabled. |

**Bits 11:0 – OVRn[11:0]** Overrun Channel n Interrupt Enable [n = 11..0]
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Overrun Channel n Interrupt Enable bit, which disables the Overrun Channel n interrupt.

| Value | Description |
|---|---|
| 0 | The Overrun Channel n interrupt is disabled. |
| 1 | The Overrun Channel n interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.5. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | EVDn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EVDn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVRn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVRn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 27:16 – EVDn[11:0]** Event Detected Channel n [n=11..0]
This flag is set on the next CLK_EVSYS_APB cycle when an event is being propagated through the channel, and an interrupt request will be generated if INTENCLR/SET.EVDn is '1'.
When the event channel path is asynchronous, the EVDn interrupt flag will not be set.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Event Detected Channel n interrupt flag.

**Bits 11:0 – OVRn[11:0]** Overrun Channel n [n=11..0]
This flag is set on the next CLK_EVSYS_APB cycle after an overrun channel condition occurs, and an interrupt request will be generated if INTENCLR/SET.OVRn is '1'.
There are two possible overrun channel conditions:

- One or more of the event users on channel n are not ready when a new event occurs.
- An event happens when the previous event on channel n has not yet been handled by all event users.

When the event channel path is asynchronous, the OVRn interrupt flag will not be set.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Overrun Detected Channel n interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.6. Software Event

**Name:** SWEVT
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CHANNELn[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CHANNELn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 11:0 – CHANNELn[11:0]** Channel n Software [n=11..0] Selection
Writing '0' to this bit has no effect.
Writing '1' to this bit will trigger a software event for the channel n.
These bits will always return zero when read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.7. Channel n Control

**Name:** CHANNELn
**Offset:** 0x20 + n*0x04 [n=0..11]
**Reset:** 0x00008000
**Property:** PAC Write-Protection

This register allows the user to configure channel n. To write to this register, do a single, 32-bit write of all the configuration data.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | EDGSEL[1:0] | | PATH[1:0] | |
| Access | R/W | R/W | | | R/W | R/W | R/W | R/W |
| Reset | 1 | 0 | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EVGEN[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – ONDEMAND** Generic Clock On Demand

| Value | Description |
|---|---|
| 0 | Generic clock for a channel is always on, if the channel is configured and generic clock source is enabled. |
| 1 | Generic clock is requested on demand while an event is handled |

**Bit 14 – RUNSTDBY** Run in Standby
This bit is used to define the behavior during standby sleep mode.

| Value | Description |
|---|---|
| 0 | The channel is disabled in standby sleep mode. |
| 1 | The channel is not stopped in standby sleep mode and depends on the CHANNEL.ONDEMAND |

**Bits 11:10 – EDGSEL[1:0]** Edge Detection Selection
These bits set the type of edge detection to be used on the channel.
These bits must be written to zero when using the asynchronous path.

| Value | Name | Description |
|---|---|---|
| 0x0 | NO_EVT_OUTPUT | No event output when using the resynchronized or synchronous path |
| 0x1 | RISING_EDGE | Event detection only on the rising edge of the signal from the event generator |
| 0x2 | FALLING_EDGE | Event detection only on the falling edge of the signal from the event generator |
| 0x3 | BOTH_EDGES | Event detection on rising and falling edges of the signal from the event generator |

**Bits 9:8 – PATH[1:0]** Path Selection
These bits are used to choose which path will be used by the selected channel.
The path choice can be limited by the channel source, see the table in USERm.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | SYNCHRONOUS | Synchronous path |
| 0x1 | RESYNCHRONIZED | Resynchronized path |
| 0x2 | ASYNCHRONOUS | Asynchronous path |
| 0x3 | - | Reserved |

**Bits 7:0 – EVGEN[7:0]** Event Generator
These bits are used to choose the event generator to connect to the selected channel.

**Table 29-2.** Event Generators

| Value | Event Generator | Description |
|---|---|---|
| 0x00 | NONE | No event generator selected |
| 0x01 | OSCCTRL FAIL | XOSC Clock Failure |
| 0x02 | OSC32KCTRL FAIL | XOSC32K Clock Failure |
| 0x03 | RTC CMP0 | Compare 0 (mode 0 and 1) or Alarm 0 (mode 2) |
| 0x04 | RTC CMP1 | Compare 1 |
| 0x05 | RTC OVF | Overflow |
| 0x06 | RTC PER0 | Period 0 |
| 0x07 | RTC PER1 | Period 1 |
| 0x08 | RTC PER2 | Period 2 |
| 0x09 | RTC PER3 | Period 3 |
| 0x0A | RTC PER4 | Period 4 |
| 0x0B | RTC PER5 | Period 5 |
| 0x0C | RTC PER6 | Period 6 |
| 0x0D | RTC PER7 | Period 7 |
| 0x0E | EIC EXTINT0 | External Interrupt 0 |
| 0x0F | EIC EXTINT1 | External Interrupt 1 |
| 0x10 | EIC EXTINT2 | External Interrupt 2 |
| 0x11 | EIC EXTINT3 | External Interrupt 3 |
| 0x12 | EIC EXTINT4 | External Interrupt 4 |
| 0x13 | EIC EXTINT5 | External Interrupt 5 |
| 0x14 | EIC EXTINT6 | External Interrupt 6 |
| 0x15 | EIC EXTINT7 | External Interrupt 7 |
| 0x16 | EIC EXTINT8 | External Interrupt 8 |
| 0x17 | EIC EXTINT9 | External Interrupt 9 |
| 0x18 | EIC EXTINT10 | External Interrupt 10 |
| 0x19 | EIC EXTINT11 | External Interrupt 11 |
| 0x1A | EIC EXTINT12 | External Interrupt 12 |
| 0x1B | EIC EXTINT13 | External Interrupt 13 |
| 0x1C | EIC EXTINT14 | External Interrupt 14 |
| 0x1D | EIC EXTINT15 | External Interrupt 15 |
| 0x1E | TSENS WINMON | Window Monitor |
| 0x1F | DMAC CH0 | Channel 0 |
| 0x20 | DMAC CH1 | Channel 1 |
| 0x21 | DMAC CH2 | Channel 2 |
| 0x22 | DMAC CH3 | Channel 3 |
| 0x23 | TCC0 OVF | Overflow |
| 0x24 | TCC0 TRG | Trig |
| 0x25 | TCC0 CNT | Counter |
| 0x26 | TCC0 MC0 | Match/Capture 0 |
| 0x27 | TCC0 MC1 | Match/Capture 1 |
| 0x28 | TCC0 MC2 | Match/Capture 2 |
| 0x29 | TCC0 MC3 | Match/Capture 3 |
| 0x2A | TCC1 OVF | Overflow |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 29-2.** Event Generators (continued)

| Value | Event Generator | Description |
|---|---|---|
| 0x2B | TCC1 TRG | Trig |
| 0x2C | TCC1 CNT | Counter |
| 0x2D | TCC1 MC0 | Match/Capture 0 |
| 0x2E | TCC1 MC1 | Match/Capture 1 |
| 0x2F | TCC2 OVF | Overflow |
| 0x30 | TCC2 TRG | Trig |
| 0x31 | TCC2 CNT | Counter |
| 0x32 | TCC2 MC0 | Match/Capture 0 |
| 0x33 | TCC2 MC1 | Match/Capture 1 |
| 0x34 | TC0 OVF | Overflow/Underflow |
| 0x35 | TC0 MC0 | Match/Capture 0 |
| 0x36 | TC0 MC1 | Match/Capture 1 |
| 0x37 | TC1 OVF | Overflow/Underflow |
| 0x38 | TC1 MC0 | Match/Capture 0 |
| 0x39 | TC1 MC1 | Match/Capture 1 |
| 0x3A | TC2 OVF | Overflow/Underflow |
| 0x3B | TC2 MC0 | Match/Capture 0 |
| 0x3C | TC2 MC1 | Match/Capture 1 |
| 0x3D | TC3 OVF | Overflow/Underflow |
| 0x3E | TC3 MC0 | Match/Capture 0 |
| 0x3F | TC3 MC1 | Match/Capture 1 |
| 0x40 | TC4 OVF | Overflow/Underflow |
| 0x41 | TC4 MC0 | Match/Capture 0 |
| 0x42 | TC4 MC1 | Match/Capture 1 |
| 0x43 | ADC0 RESRDY | Result Ready |
| 0x44 | ADC0 WINMON | Window Monitor |
| 0x45 | ADC1 RESRDY | Result Ready |
| 0x46 | ADC1 WINMON | Window Monitor |
| 0x47 | SDADC RESRDY | Result Ready |
| 0x48 | SDADC WINMON | Window Monitor |
| 0x49 | AC COMP0 | Comparator 0 |
| 0x4A | AC COMP1 | Comparator 1 |
| 0x4B | AC COMP2 | Comparator 2 |
| 0x4C | AC COMP3 | Comparator 3 |
| 0x4D | AC WIN0 | Window 0 |
| 0x4E | AC WIN1 | Window 1 |
| 0x4F | DAC EMPTY | Data Buffer Empty |
| 0x50 | PTC EOC | End of Conversion |
| 0x51 | PTC WINCOMP | Window Comparator |
| 0x52 | CCL LUTOUT0 | CCL output |
| 0x53 | CCL LUTOUT1 | CCL output |
| 0x54 | CCL LUTOUT2 | CCL output |
| 0x55 | CCL LUT3 | CCL output |
| 0x56 | PAC ACCERR | Access Error |
| 0x57 | - | Reserved |
| 0x58 | TC5 OVF | Overflow/Underflow |
| 0x59 | TC5 MC0 | Match/Capture 0 |
| 0x5A | TC5 MC1 | Match/Capture 1 |
| 0x5B | TC6 OVF | Overflow/Underflow |
| 0x5C | TC6 MC0 | Match/Capture 0 |
| 0x5D | TC6 MC1 | Match/Capture 1 |
| 0x5E | TC7 OVF | Overflow/Underflow |
| 0x5F | TC7 MC0 | Match/Capture 0 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 29-2.** Event Generators (continued)

| Value | Event Generator | Description |
|---|---|---|
| 0x60 | TC7 MC1 | Match/Capture 1 |
| 0x61 - 0xFF | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 29.8.8. Event User m

**Name:** USERm
**Offset:** 0x80 + m*0x04 [m=0..49]
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CHANNEL[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – CHANNEL[7:0]** Channel Event Selection
These bits are used to select the channel to connect to the event user.
Note that to select channel m, the value (m+1) must be written to the USER.CHANNEL bit group.

| Value | Channel Number |
|---|---|
| 0x00 | No channel output selected |
| 0x01 | 0 |
| 0x02 | 1 |
| 0x03 | 2 |
| 0x04 | 3 |
| 0x05 | 4 |
| 0x06 | 5 |
| 0x07 | 6 |
| 0x08 | 7 |
| 0x09 | 8 |
| 0x0A | 9 |
| 0x0B | 10 |
| 0x0C | 11 |
| 0x0D-0xFF | Reserved |

**Table 29-3.** User Multiplexer Number

| USERm | User Multiplexer | Description | Path Type |
|---|---|---|---|
| m = 0 | TSENS START | Start measurement | Asynchronous, synchronous, and resynchronized paths |
| m = 1 | PORT EV0 | Event 0 | Asynchronous path only |
| m = 2 | PORT EV1 | Event 1 | Asynchronous path only |
| m = 3 | PORT EV2 | Event 2 | Asynchronous path only |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 29-3.** User Multiplexer Number (continued)

| USERm | User Multiplexer | Description | Path Type |
|---|---|---|---|
| m = 4 | PORT EV3 | Event 3 | Asynchronous path only |
| m = 5 | DMAC CH0 | Channel 0 | Asynchronous, synchronous, and resynchronized paths |
| m = 6 | DMAC CH1 | Channel 1 | Asynchronous, synchronous, and resynchronized paths |
| m = 7 | DMAC CH2 | Channel 2 | Asynchronous, synchronous, and resynchronized paths |
| m = 8 | DMAC CH3 | Channel 3 | Asynchronous, synchronous, and resynchronized paths |
| m = 9 | TCC0 EV0 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 10 | TCC0 EV1 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 11 | TCC0 MC0 | Match/Capture 0 | Asynchronous, synchronous, and resynchronized paths |
| m = 12 | TCC0 MC1 | Match/Capture 1 | Asynchronous, synchronous, and resynchronized paths |
| m = 13 | TCC0 MC2 | Match/Capture 2 | Asynchronous, synchronous, and resynchronized paths |
| m = 14 | TCC0 MC3 | Match/Capture 3 | Asynchronous, synchronous, and resynchronized paths |
| m = 15 | TCC1 EV0 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 16 | TCC1 EV1 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 17 | TCC1 MC0 | Match/Capture 0 | Asynchronous, synchronous, and resynchronized paths |
| m = 18 | TCC1 MC1 | Match/Capture 1 | Asynchronous, synchronous, and resynchronized paths |
| m = 19 | TCC2 EV0 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 20 | TCC2 EV1 | - | Asynchronous, synchronous, and resynchronized paths |
| m = 21 | TCC2 MC0 | Match/Capture 0 | Asynchronous, synchronous, and resynchronized paths |
| m = 22 | TCC2 MC1 | Match/Capture 1 | Asynchronous, synchronous, and resynchronized paths |
| m = 23 | TC0EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m = 24 | TC1EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m = 25 | TC2EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m = 26 | TC3EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m = 27 | TC4EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m = 28 | ADC0 START | ADC start conversion | Asynchronous path only |
| m = 29 | ADC0 SYNC | Flush ADC | Asynchronous path only |
| m = 30 | ADC1 START | ADC start conversion | Asynchronous path only |
| m = 31 | ADC1 SYNC | Flush ADC | Asynchronous path only |
| m = 32 | SDADC START | SADC start conversion | Asynchronous path only |
| m = 33 | SDADC FLUSH | Flush SADC | Asynchronous path only |
| m = 34 | AC SOC0 | Start comparator 0 | Asynchronous path only |
| m = 35 | AC SOC1 | Start comparator 1 | Asynchronous path only |
| m = 36 | AC SOC2 | Start comparator 2 | Asynchronous path only |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 29-3.** User Multiplexer Number (continued)

| USERm | User Multiplexer | Description | Path Type |
|---|---|---|---|
| m = 37 | AC SOC3 | Start comparator 3 | Asynchronous path only |
| m = 38 | DAC START | DAC start conversion | Asynchronous path only |
| m = 39 | PTC STCONV | PTC start conversion | Asynchronous path only |
| m = 40 | CCL LUTIN 0 | CCL input | Asynchronous path only |
| m = 41 | CCL LUTIN 1 | CCL input | Asynchronous path only |
| m = 42 | CCL LUTIN 2 | CCL input | Asynchronous path only |
| m = 43 | CCL LUTIN 3 | CCL input | Asynchronous path only |
| m=44 | Reserved | - | Reserved |
| m=45 | MTB_START | MTB Start | Asynchronous path only |
| m=46 | MTB_STOP | MTB Stop | Asynchronous path only |
| m=47(1) | TC5EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m=48(1) | TC6EVU | - | Asynchronous, synchronous, and resynchronized paths |
| m=49(1) | TC7EVU | - | Asynchronous, synchronous, and resynchronized paths |
| others | Reserved | - | Reserved |

**Note:**

1. These Users are only available for SAM C20N/C21N variants.


© 2026 Microchip Technology Inc. and its subsidiaries

# 30. SERCOM – Serial Communication Interface

## 30.1. Overview

There are up to eight instances of the serial communication interface (SERCOM) peripheral.

A SERCOM can be configured to support a number of modes: $I^2C$, SPI, and USART. When an instance of SERCOM is configured and enabled, all of the resources of that SERCOM instance will be dedicated to the selected mode.

The SERCOM serial engine consists of a transmitter and receiver, baud-rate generator and address matching functionality. It can use the internal generic clock or an external clock. Using an external clock allows the SERCOM to be operated in all Sleep modes.

**Related Links**

SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit

## 30.2. Features

- Interface for configuring into one of the following:
  - Inter-Integrated Circuit ($I^2C$) Two-wire Serial Interface
  - System Management Bus (SMBus™) compatible
  - Serial Peripheral Interface (SPI)
  - Universal Synchronous/Asynchronous Receiver/Transmitter (USART)
- Single transmit buffer and double receive buffer
- Baud-rate generator
- Address match/mask logic
- Operational in all Sleep modes with an external clock source
- Can be used with DMA

See the Related Links for full feature lists of the interface configurations.

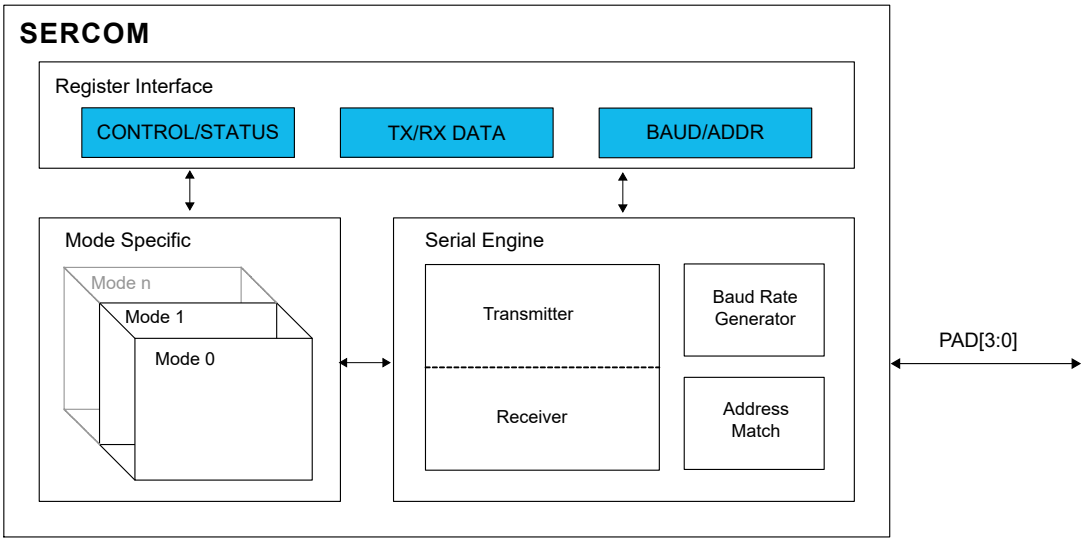
**Related Links**

SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit


© 2026 Microchip Technology Inc. and its subsidiaries

## 30.3. Block Diagram

**Figure 30-1.** SERCOM Block Diagram




## 30.4. Signal Description

See the respective SERCOM mode chapters for details.

**Related Links**

SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit

## 30.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 30.5.1. I/O Lines

Using the SERCOM I/O lines requires the I/O pins to be configured using port configuration (PORT).

The SERCOM has four internal pads, PAD[3:0], and the signals from $I^2C$, SPI and USART are routed through these SERCOM pads through a multiplexer. The configuration of the multiplexer is available from the different SERCOM modes. Refer to the mode specific chapters for additional information.

**Related Links**

SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit
PORT - I/O Pin Controller
Block Diagram

### 30.5.2. Power Management

The SERCOM can operate in any Sleep mode provided the selected clock source is running. SERCOM interrupts can be configured to wake the device from sleep modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 30.5.3. Clocks

The SERCOM bus clock (CLK_SERCOMx_APB) can be enabled and disabled in the Main Clock Controller. Refer to *Peripheral Clock Masking* for details and default status of this clock.

The SERCOM uses two generic clocks: GCLK_SERCOMx_CORE and GCLK_SERCOMx_SLOW. The core clock (GCLK_SERCOMx_CORE) is required to clock the SERCOM while working as a host. The slow clock (GCLK_SERCOMx_SLOW) is only required for certain functions. See specific mode chapters for details.

These clocks must be configured and enabled in the Generic Clock Controller (GCLK) before using the SERCOM.

The generic clocks are asynchronous to the user interface clock (CLK_SERCOMx_APB). Due to this asynchronicity, writing to certain registers will require synchronization between the clock domains. Refer to Synchronization for details.

**Related Links**

GCLK - Generic Clock Controller
MCLK – Main Clock

### 30.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). The DMAC must be configured before the SERCOM DMA requests are used.

**Related Links**

DMAC – Direct Memory Access Controller

### 30.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller (NVIC). The NVIC must be configured before the SERCOM interrupts are used.

**Related Links**

Nested Vector Interrupt Controller

### 30.5.6. Events

Not applicable.

### 30.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

### 30.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Interrupt Flag Clear and Status register (INTFLAG)
- Status register (STATUS)
- Data register (DATA)
- Address register (ADDR)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
PAC - Peripheral Access Controller

### 30.5.9. Analog Connections

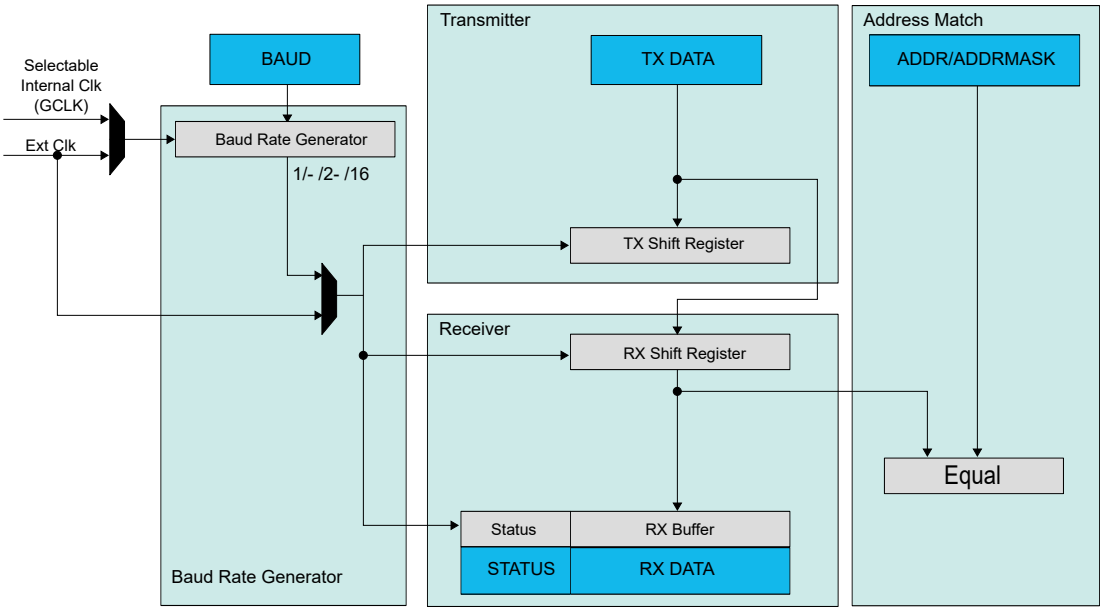
Not applicable.

## 30.6. Functional Description

### 30.6.1. Principle of Operation

The basic structure of the SERCOM serial engine is shown in Figure 30-2. Labels in capital letters are synchronous to the system clock and accessible by the CPU; labels in lowercase letters can be configured to run on the GCLK_SERCOMx_CORE clock or an external clock.

**Figure 30-2.** SERCOM Serial Engine




The transmitter consists of a single write buffer and a shift register.

The receiver consists of a one-level ($I^2C$), two-level (USART, SPI) receive buffer and a shift register.

The baud-rate generator is capable of running on the GCLK_SERCOMx_CORE clock or an external clock.

Address matching logic is included for SPI and $I^2C$ operation.

### 30.6.2. Basic Operation

#### 30.6.2.1. Initialization

The SERCOM must be configured to the desired mode by writing the Operating Mode bits in the Control A register (CTRLA.MODE) as shown in the table below.

**Table 30-1.** SERCOM Modes

| CTRLA.MODE | Description |
|---|---|
| 0x0 | USART with external clock |
| 0x1 | USART with internal clock |
| 0x2 | SPI in client operation |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 30-1.** SERCOM Modes (continued)

| CTRLA.MODE | Description |
|---|---|
| 0x3 | SPI in host operation |
| 0x4 | $I^2C$ client operation |
| 0x5 | $I^2C$ host operation |
| 0x6-0x7 | Reserved |

For further initialization information, see the respective SERCOM mode chapters:

**Related Links**

SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit

### 30.6.2.2. Enabling, Disabling, and Resetting

This peripheral is enabled by writing '1' to the Enable bit in the Control A register (CTRLA.ENABLE), and disabled by writing '0' to it.

Writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST) will reset all registers of this peripheral to their initial states, except the DBGCTRL register, and the peripheral is disabled.

Refer to the CTRLA register description for details.

### 30.6.2.3. Clock Generation – Baud-Rate Generator

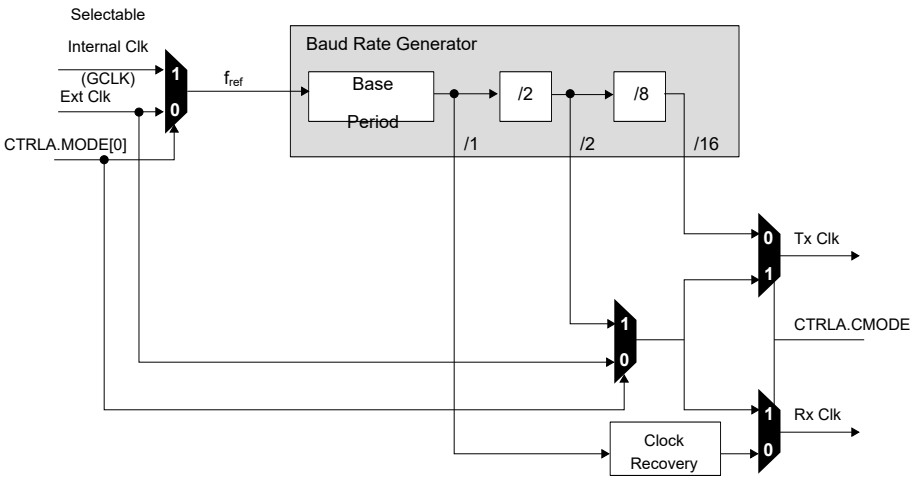
The baud-rate generator, as shown in the following figure, generates internal clocks for asynchronous and synchronous communication. The output frequency ($f_{BAUD}$) is determined by the Baud register (BAUD) setting and the baud reference frequency ($f_{ref}$). The baud reference clock is the serial engine clock, and it can be internal or external.

For asynchronous communication, the /16 (divide-by-16) output is used when transmitting, whereas the /1 (divide-by-1) output is used while receiving.

For synchronous communication, the /2 (divide-by-2) output is used.

This functionality is automatically configured, depending on the selected operating mode.

**Figure 30-3.** Baud Rate Generator




The following table contains equations for the baud rate (in bits per second) and the BAUD register value for each operating mode.

For asynchronous operation, there are two modes:


© 2026 Microchip Technology Inc. and its subsidiaries

- *Arithmetic mode*: the BAUD register value is 16 bits (0 to 65,535)
- *Fractional mode*: the BAUD register value is 13 bits, while the fractional adjustment is 3 bits. In this mode the BAUD setting must be greater than or equal to 1.

For synchronous operation, the BAUD register value is 8 bits (0 to 255).

**Table 30-2.** Baud Rate Equations

| Operating Mode | Condition | Baud Rate (Bits Per Second) | BAUD Register Value Calculation |
|---|---|---|---|
| Asynchronous Arithmetic | $f_{BAUD} \leq \frac{f_{ref}}{16}$ | $f_{BAUD} = \frac{f_{ref}}{16}\left(1 - \frac{BAUD}{65536}\right)$ | $BAUD = 65536 \cdot \left(1 - S \cdot \frac{f_{BAUD}}{f_{ref}}\right)$ |
| Asynchronous Fractional | $f_{BAUD} \leq \frac{f_{ref}}{S}$ | $f_{BAUD} = \frac{f_{ref}}{S \cdot \left(BAUD + \frac{FP}{8}\right)}$ | $BAUD = \frac{f_{ref}}{S \cdot f_{BAUD}} - \frac{FP}{8}$ |
| Synchronous | $f_{BAUD} \leq \frac{f_{ref}}{2}$ | $f_{BAUD} = \frac{f_{ref}}{2 \cdot (BAUD + 1)}$ | $BAUD = \frac{f_{ref}}{2 \cdot f_{BAUD}} - 1$ |

*S* - Number of samples per bit, which can be 16, 8, or 3.

The Asynchronous Fractional option is used for auto-baud detection.

The baud rate error is represented by the following formula:

$$\text{Error} = 1 - \left(\frac{\text{ExpectedBaudRate}}{\text{ActualBaudRate}}\right)$$

### 30.6.2.3.1. Asynchronous Arithmetic Mode BAUD Value Selection

The formula given for $f_{BAUD}$ calculates the average frequency over 65536 $f_{ref}$ cycles. Although the BAUD register can be set to any value between 0 and 65536, the actual average frequency of $f_{BAUD}$ over a single frame is more granular. The BAUD register values that will affect the average frequency over a single frame lead to an integer increase in the cycles per frame (CPF)

$$CPF = \frac{f_{ref}}{f_{BAUD}}(D + S)$$

where

- *D* represent the data bits per frame
- *S* represent the sum of start and first stop bits, if present.

Table 30-3 shows the BAUD register value versus baud frequency $f_{BAUD}$ at a serial engine frequency of 48MHz. This assumes a *D* value of 8 bits and an *S* value of 2 bits (10 bits, including start and stop bits).

**Table 30-3.** BAUD Register Value vs. Baud Frequency

| BAUD Register Value | Serial Engine CPF | $f_{BAUD}$ at 48MHz Serial Engine Frequency ($f_{REF}$) |
|---|---|---|
| 0 – 406 | 160 | 3MHz |
| 407 – 808 | 161 | 2.981MHz |
| 809 – 1205 | 162 | 2.963MHz |
| ... | ... | ... |
| 65206 | 31775 | 15.11kHz |
| 65207 | 31871 | 15.06kHz |
| 65208 | 31969 | 15.01kHz |


© 2026 Microchip Technology Inc. and its subsidiaries

## 30.6.3. Additional Features

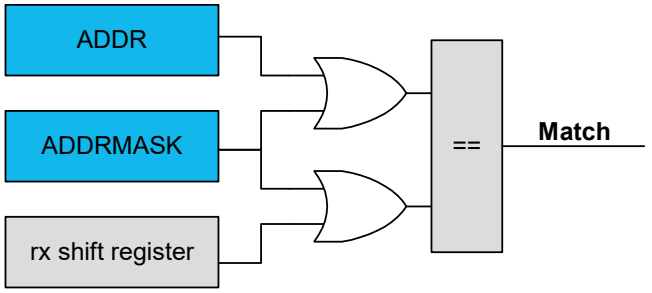
### 30.6.3.1. Address Match and Mask

The SERCOM address match and mask feature is capable of matching either one address, two unique addresses, or a range of addresses with a mask, based on the mode selected. The match uses seven or eight bits, depending on the mode.

#### 30.6.3.1.1. Address With Mask

An address written to the Address bits in the Address register (ADDR.ADDR), and a mask written to the Address Mask bits in the Address register (ADDR.ADDRMASK) will yield an address match. All bits that are masked are not included in the match. Note that writing the ADDR.ADDRMASK to 'all zeros' will match a single unique address, while writing ADDR.ADDRMASK to 'all ones' will result in all addresses being accepted.

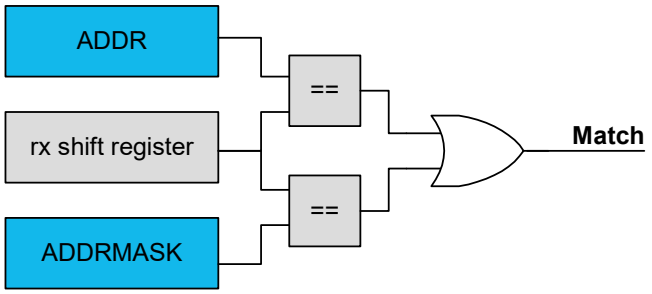
**Figure 30-4.** Address With Mask




#### 30.6.3.1.2. Two Unique Addresses

The two addresses written to ADDR and ADDRMASK will cause a match.

**Figure 30-5.** Two Unique Addresses




#### 30.6.3.1.3. Address Range

The range of addresses between and including ADDR.ADDR and ADDR.ADDRMASK will cause a match. ADDR.ADDR and ADDR.ADDRMASK can be set to any two addresses, with ADDR.ADDR acting as the upper limit and ADDR.ADDRMASK acting as the lower limit.

**Figure 30-6.** Address Range




## 30.6.4. DMA Operation

The available DMA interrupts and their depend on the operation mode of the SERCOM peripheral. Refer to the Functional Description sections of the respective SERCOM mode.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter
SERCOM SPI – SERCOM Serial Peripheral Interface
SERCOM I2C – Inter-Integrated Circuit

### 30.6.5. Interrupts

Interrupt sources are mode-specific. See the respective SERCOM mode chapters for details.

Each interrupt source has its own interrupt flag.

The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the interrupt condition is met.

Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR).

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled, or the SERCOM is reset. For details on clearing interrupt flags, refer to the INTFLAG register description.

The value of INTFLAG indicates which interrupt condition occurred. The user must read the INTFLAG register to determine which interrupt condition is present.

**Note:** Interrupts must be globally enabled for interrupt requests.

**Related Links**
Nested Vector Interrupt Controller

### 30.6.6. Events

Not applicable.

### 30.6.7. Sleep Mode Operation

The peripheral can operate in any sleep mode where the selected serial clock is running. This clock can be external or generated by the internal baud-rate generator.

The SERCOM interrupts can be used to wake up the device from sleep modes. Refer to the different SERCOM mode chapters for details.

### 30.6.8. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

**Related Links**
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

# 31. SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter

## 31.1. Overview

The Universal Synchronous and Asynchronous Receiver and Transmitter (USART) is one of the available modes in the Serial Communication Interface (SERCOM).

The USART uses the SERCOM transmitter and receiver, see Block Diagram. Labels in uppercase letters are synchronous to CLK_SERCOMx_APB and accessible for CPU. Labels in lowercase letters can be programmed to run on the internal generic clock or an external clock.

The transmitter consists of a single write buffer, a shift register, and control logic for different frame formats. The write buffer support data transmission without any delay between frames. The receiver consists of a two-level receive buffer and a shift register. Status information of the received data is available for error checking. Data and clock recovery units ensure robust synchronization and noise filtering during asynchronous data reception.

**Related Links**
SERCOM – Serial Communication Interface

## 31.2. USART Features

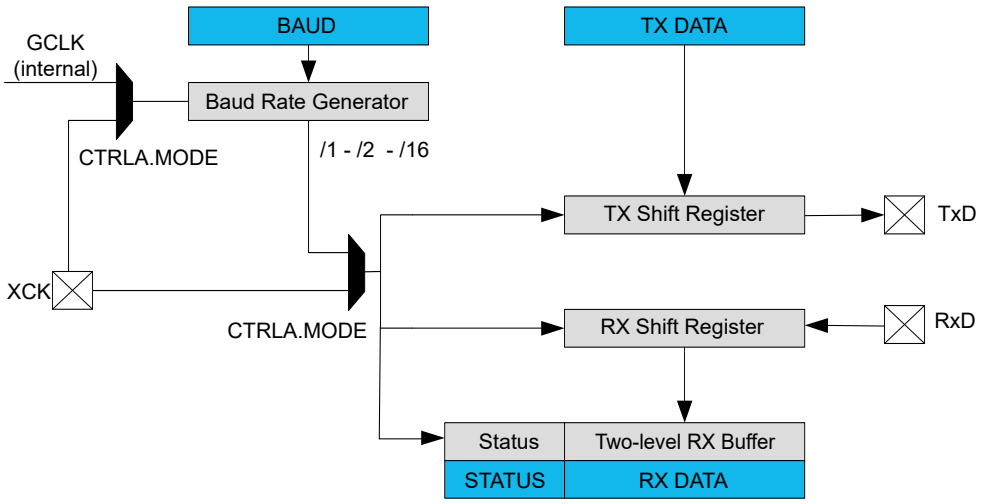
- Full-duplex operation
- Asynchronous (with clock reconstruction) or synchronous operation
- Internal or external clock source for asynchronous and synchronous operation
- Baud-rate generator
- Supports serial frames with 5, 6, 7, 8 or 9 data bits and 1 or 2 stop bits
- Odd or even parity generation and parity check
- Selectable LSB- or MSB-first data transfer
- Buffer overflow and frame error detection
- Noise filtering, including false start-bit detection and digital low-pass filter
- Collision detection
- Can operate in all sleep modes
- Operation at speeds up to half the system clock for internally generated clocks
- Operation at speeds up to the system clock for externally generated clocks
- RTS and CTS flow control
- IrDA modulation and demodulation up to 115.2kbps
- LIN Host support
- LIN Client support
  - Auto-baud and break character detection
- RS485 Support
- Start-of-frame detection
- Can work with DMA

**Related Links**
Features


© 2026 Microchip Technology Inc. and its subsidiaries

## 31.3. Block Diagram

**Figure 31-1.** USART Block Diagram




## 31.4. Signal Description

**Table 31-1.** SERCOM USART Signals

| Signal Name | Type | Description |
|---|---|---|
| PAD[3:0] | Digital I/O | General SERCOM pins |

One signal can be mapped to one of several pins.

**Related Links**
I/O Multiplexing and Considerations

## 31.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 31.5.1. I/O Lines

Using the USART's I/O lines requires the I/O pins to be configured using the I/O Pin Controller (PORT).

When the SERCOM is used in USART mode, the SERCOM controls the direction and value of the I/O pins according to the table below. If the receiver or transmitter is disabled, these pins can be used for other purposes.

PORT Control bit PINCFGn.DRVSTR is still effective for the SERCOM output pins. PORT Control bit PINCFGn.PULLEN is still effective for enabling/disabling a pull on the SERCOM input pins, but is limited to the enabling/disabling of a pull down only (it is not possible to enable/disable a pull up).

**Table 31-2.** USART Pin Configuration

| Pin | Pin Configuration |
|---|---|
| TxD | Output |
| RxD | Input |
| XCK | Output or input |


© 2026 Microchip Technology Inc. and its subsidiaries

The combined configuration of PORT and the Transmit Data Pinout and Receive Data Pinout bit fields in the Control A register (CTRLA.TXPO and CTRLA.RXPO, respectively) will define the physical position of the USART signals in the table above.

**Related Links**
PORT - I/O Pin Controller

### 31.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes.

**Related Links**
PM - Power Manager

### 31.5.3. Clocks

The SERCOM bus clock (CLK_SERCOMx_APB) can be enabled and disabled in the Main Clock Controller. Refer to *Peripheral Clock Masking* for details and default status of this clock.

A generic clock (GCLK_SERCOMx_CORE) is required to clock the SERCOMx_CORE. This clock must be configured and enabled in the Generic Clock Controller before using the SERCOMx_CORE. Refer to *GCLK - Generic Clock Controller* for details.

This generic clock is asynchronous to the bus clock (CLK_SERCOMx_APB). Therefore, writing to certain registers will require synchronization to the clock domains. Refer to *Synchronization* for further details.

**Related Links**
Peripheral Clock Masking
Synchronization
GCLK - Generic Clock Controller

### 31.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

**Related Links**
DMAC – Direct Memory Access Controller

### 31.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

### 31.5.6. Events

Not applicable.

### 31.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging - refer to the Debug Control (DBGCTRL) register for details.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**

DBGCTRL

Debug Control

### 31.5.8. Register Access Protection

Registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC).

PAC Write-Protection is not available for the following registers:

- Interrupt Flag Clear and Status register (INTFLAG)
- Status register (STATUS)
- Data register (DATA)

Optional PAC Write-Protection is denoted by the "PAC Write-Protection" property in each individual register description.

Write-protection does not apply to accesses through an external debugger.

**Related Links**

PAC - Peripheral Access Controller

### 31.5.9. Analog Connections

Not applicable.

## 31.6. Functional Description

### 31.6.1. Principle of Operation

The USART uses the following lines for data transfer:

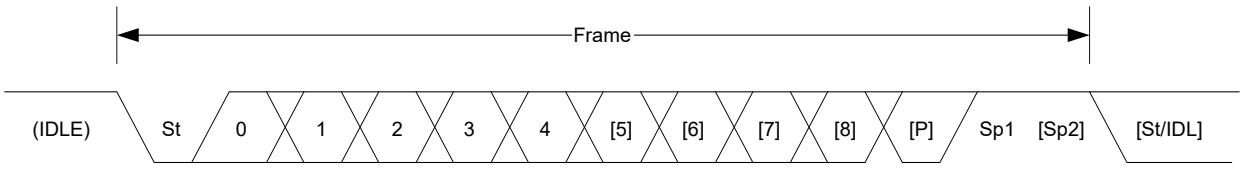
- RxD for receiving
- TxD for transmitting
- XCK for the transmission clock in synchronous operation

USART data transfer is frame based. A serial frame consists of:

- 1 start bit
- From 5 to 9 data bits (MSB or LSB first)
- No, even or odd parity bit
- 1 or 2 stop bits

A frame starts with the Start bit followed by one character of Data bits. If enabled, the parity bit is inserted after the Data bits and before the first Stop bit. After the stop bit(s) of a frame, either the next frame can follow immediately, or the communication line can return to the Idle (high) state. The figure below illustrates the possible frame formats. Values inside brackets ([x]) denote optional bits.

**Figure 31-2.** Frame Formats




**St** Start bit. Signal is always low.


© 2026 Microchip Technology Inc. and its subsidiaries

**n, [n]** Data bits. 0 to [5..9]

**[P]** Parity bit. Either odd or even.

**Sp, [Sp]** Stop bit. Signal is always high.

**IDLE** No frame is transferred on the communication line. Signal is always high in this state.

## 31.6.2. Basic Operation

### 31.6.2.1. Initialization

The following registers are enable-protected, meaning they can only be written when the USART is disabled (CTRL.ENABLE=0):

- Control A register (CTRLA), except the Enable (ENABLE) and Software Reset (SWRST) bits.
- Control B register (CTRLB), except the Receiver Enable (RXEN) and Transmitter Enable (TXEN) bits.
- Baud register (BAUD)

When the USART is enabled or is being enabled (CTRLA.ENABLE=1), any writing attempt to these registers will be discarded. If the peripheral is being disabled, writing to these registers will be executed after disabling is completed. Enable-protection is denoted by the "Enable-Protection" property in the register description.

Before the USART is enabled, it must be configured by these steps:

1. Select either external (0x0) or internal clock (0x1) by writing the Operating Mode value in the CTRLA register (CTRLA.MODE).
2. Select either asynchronous (0) or or synchronous (1) communication mode by writing the Communication Mode bit in the CTRLA register (CTRLA.CMODE).
3. Select pin for receive data by writing the Receive Data Pinout value in the CTRLA register (CTRLA.RXPO).
4. Select pads for the transmitter and external clock by writing the Transmit Data Pinout bit in the CTRLA register (CTRLA.TXPO).
5. Configure the Character Size field in the CTRLB register (CTRLB.CHSIZE) for character size.
6. Set the Data Order bit in the CTRLA register (CTRLA.DORD) to determine MSB- or LSB-first data transmission.
7. To use parity mode:
   a. Enable parity mode by writing 0x1 to the Frame Format field in the CTRLA register (CTRLA.FORM).
   b. Configure the Parity Mode bit in the CTRLB register (CTRLB.PMODE) for even or odd parity.
8. Configure the number of stop bits in the Stop Bit Mode bit in the CTRLB register (CTRLB.SBMODE).
9. When using an internal clock, write the Baud register (BAUD) to generate the desired baud rate.
10. Enable the transmitter and receiver by writing '1' to the Receiver Enable and Transmitter Enable bits in the CTRLB register (CTRLB.RXEN and CTRLB.TXEN).

### 31.6.2.2. Enabling, Disabling, and Resetting

This peripheral is enabled by writing '1' to the Enable bit in the Control A register (CTRLA.ENABLE), and disabled by writing '0' to it.

Writing ‘1’ to the Software Reset bit in the Control A register (CTRLA.SWRST) will reset all registers of this peripheral to their initial states, except the DBGCTRL register, and the peripheral is disabled.

Refer to the CTRLA register description for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.6.2.3. Clock Generation and Selection

For both Synchronous and Asynchronous modes, the clock used for shifting and sampling data can be generated internally by the SERCOM baud-rate generator or supplied externally through the XCK line.

The Synchronous mode is selected by writing a '1' to the Communication Mode bit in the Control A register (CTRLA.CMODE), the Asynchronous mode is selected by writing '0' to CTRLA.CMODE.

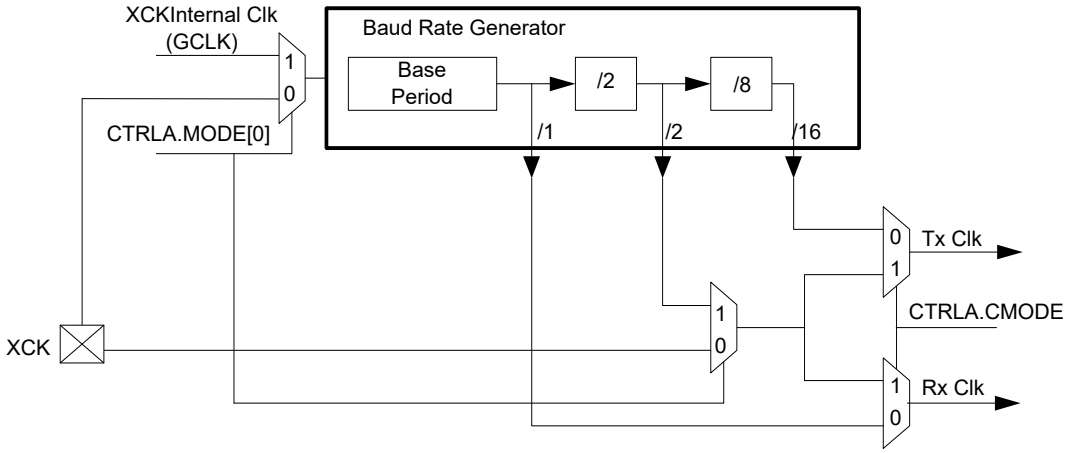
The internal clock source is selected by writing '1' to the Operation Mode bit field in the Control A register (CTRLA.MODE), the external clock source is selected by writing '0' to CTRLA.MODE.

The SERCOM baud-rate generator is configured as in the following figure.

In Asynchronous mode (CTRLA.CMODE=0), the 16-bit Baud register value is used.

In Synchronous mode (CTRLA.CMODE=1), the eight LSBs of the Baud register are used. Refer to *Clock Generation – Baud-Rate Generator* for details on configuring the baud rate.

**Figure 31-3.** Clock Generation




**Related Links**

Clock Generation – Baud-Rate Generator

Asynchronous Arithmetic Mode BAUD Value Selection

#### 31.6.2.3.1. Synchronous Clock Operation

In synchronous mode, the CTRLA.MODE bit field determines whether the transmission clock line (XCK) serves either as input or output. The dependency between clock edges, data sampling, and data change is the same for internal and external clocks. Data input on the RxD pin is sampled at the opposite XCK clock edge when data is driven on the TxD pin.

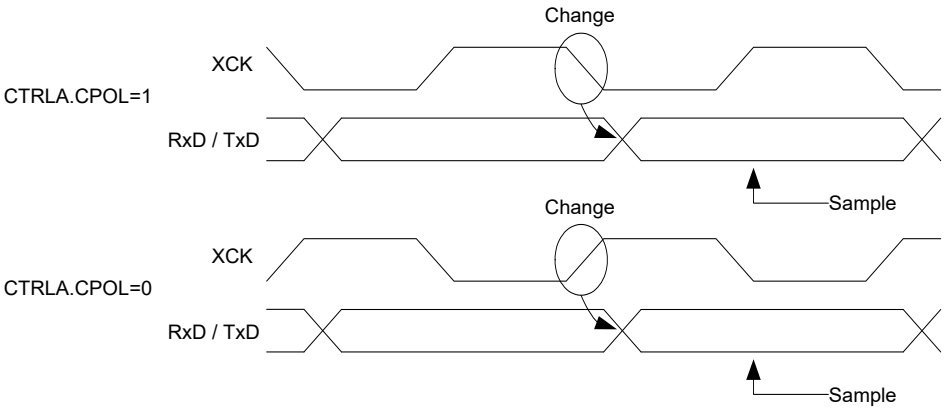
The Clock Polarity bit in the Control A register (CTRLA.CPOL) selects which XCK clock edge is used for RxD sampling, and which is used for TxD change:

When CTRLA.CPOL is '0', the data will be changed on the rising edge of XCK, and sampled on the falling edge of XCK.

When CTRLA.CPOL is '1', the data will be changed on the falling edge of XCK, and sampled on the rising edge of XCK.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 31-4.** Synchronous Mode XCK Timing




When the clock is provided through XCK (CTRLA.MODE=0x0), the shift registers operate directly on the XCK clock. This means that XCK is not synchronized with the system clock and, therefore, can operate at frequencies up to the system frequency.

#### 31.6.2.4. Data Register

The USART Transmit Data register (TxDATA) and USART Receive Data register (RxDATA) share the same I/O address, referred to as the Data register (DATA). Writing the DATA register will update the TxDATA register. Reading the DATA register will return the contents of the RxDATA register.

#### 31.6.2.5. Data Transmission

Data transmission is initiated by writing the data to be sent into the DATA register. Then, the data in TxDATA will be moved to the shift register when the shift register is empty and ready to send a new frame. After the shift register is loaded with data, the data frame will be transmitted.

When the entire data frame including stop bit has been transmitted (both the Tx buffer and the shift register are empty) and no new data was written to DATA register, the Transmit Complete interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.TXC) will be set, and the optional interrupt will be generated.

The Data Register Empty flag in the Interrupt Flag Status and Clear register (INTFLAG.DRE) indicates that the register is empty and ready for new data. The DATA register should only be written to when INTFLAG.DRE is set.

##### 31.6.2.5.1. Disabling the Transmitter

The transmitter is disabled by writing '0' to the Transmitter Enable bit in the CTRLB register (CTRLB.TXEN).

Disabling the transmitter will complete only after any ongoing and pending transmissions are completed, i.e., there is no data in the transmit shift register and TxDATA to transmit.

#### 31.6.2.6. Data Reception

The receiver accepts data when a valid start bit is detected. Each bit following the start bit will be sampled according to the baud rate or XCK clock, and shifted into the receive shift register until the first stop bit of a frame is received. The second stop bit will be ignored by the receiver.

When the first stop bit is received and a complete serial frame is present in the receive shift register, the contents of the shift register will be moved into the two-level receive buffer. Then, the Receive Complete interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.RXC) will be set, and the optional interrupt will be generated.

The received data can be read from the DATA register when the Receive Complete interrupt flag is set.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 31.6.2.6.1. Disabling the Receiver

Writing '0' to the Receiver Enable bit in the CTRLB register (CTRLB.RXEN) will disable the receiver, flush the two-level receive buffer, and data from ongoing receptions will be lost.

#### 31.6.2.6.2. Error Bits

The USART receiver has three error bits in the Status (STATUS) register: Frame Error (FERR), Buffer Overflow (BUFOVF), and Parity Error (PERR). Once an error happens, the corresponding error bit will be set until it is cleared by writing '1' to it. These bits are also cleared automatically when the receiver is disabled.

There are two methods for buffer overflow notification, selected by the Immediate Buffer Overflow Notification bit in the Control A register (CTRLA.IBON):

When CTRLA.IBON = 1, STATUS.BUFOVF is raised immediately upon buffer overflow. Software can then empty the two-level RX buffer by reading RxDATA, until the receiver complete interrupt flag (INTFLAG.RXC) is cleared.

When CTRLA.IBON = 0, the buffer overflow condition is attending data through the two-level RX buffer. After the received data is read, STATUS.BUFOVF will be set along with INTFLAG.RXC.

#### 31.6.2.6.3. Asynchronous Data Reception

The USART includes a clock recovery and data recovery unit for handling asynchronous data reception.

The clock recovery logic can synchronize the incoming asynchronous serial frames at the RxD pin to the internally generated baud-rate clock.

The data recovery logic samples and applies a low-pass filter to each incoming bit, thereby improving the noise immunity of the receiver.

#### 31.6.2.6.4. Asynchronous Operational Range

The operational range of the asynchronous reception depends on the accuracy of the internal baud-rate clock, the rate of the incoming frames, and the frame size (in number of bits). In addition, the operational range of the receiver is depending on the difference between the received bit rate and the internally generated baud rate. If the baud rate of an external transmitter is too high or too low compared to the internally generated baud rate, the receiver will not be able to synchronize the frames to the start bit.

There are two possible sources for a mismatch in baud rate: First, the reference clock will always have some minor instability. Second, the baud-rate generator cannot always do an exact division of the reference clock frequency to get the baud rate desired. In this case, the BAUD register value should be set to give the lowest possible error. Refer to *Clock Generation – Baud-Rate Generator* for details.

Recommended maximum receiver baud-rate errors for various character sizes are shown in the table below.

**Table 31-3.** Asynchronous Receiver Error for 16-fold Oversampling

| D (Data bits+Parity) | $R_{SLOW}$ [%] | $R_{FAST}$ [%] | Max. total error [%] | Recommended max. Rx error [%] |
|---|---|---|---|---|
| 5 | 94.12 | 107.69 | +5.88/-7.69 | ±2.5 |
| 6 | 94.92 | 106.67 | +5.08/-6.67 | ±2.0 |
| 7 | 95.52 | 105.88 | +4.48/-5.88 | ±2.0 |
| 8 | 96.00 | 105.26 | +4.00/-5.26 | ±2.0 |
| 9 | 96.39 | 104.76 | +3.61/-4.76 | ±1.5 |
| 10 | 96.70 | 104.35 | +3.30/-4.35 | ±1.5 |

The following equations calculate the ratio of the incoming data rate and internal receiver baud rate:

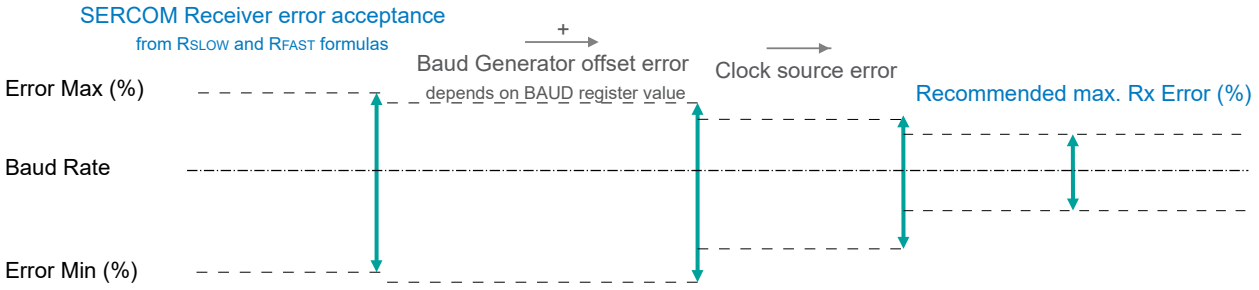

© 2026 Microchip Technology Inc. and its subsidiaries

$$R_{SLOW} = \frac{(D+1)S}{S-1+D\cdot S+S_F} \quad , \quad R_{FAST} = \frac{(D+2)S}{(D+1)S+S_M}$$

- $R_{SLOW}$ is the ratio of the slowest incoming data rate that can be accepted in relation to the receiver baud rate
- $R_{FAST}$ is the ratio of the fastest incoming data rate that can be accepted in relation to the receiver baud rate
- $D$ is the sum of character size and parity size ($D$ = 5 to 10 bits)
- $S$ is the number of samples per bit ($S$ = 16, 8 or 3)
- $S_F$ is the first sample number used for majority voting ($S_F$ = 7, 3, or 2) when CTRLA.SAMPA=0.
- $S_M$ is the middle sample number used for majority voting ($S_M$ = 8, 4, or 2) when CTRLA.SAMPA=0.

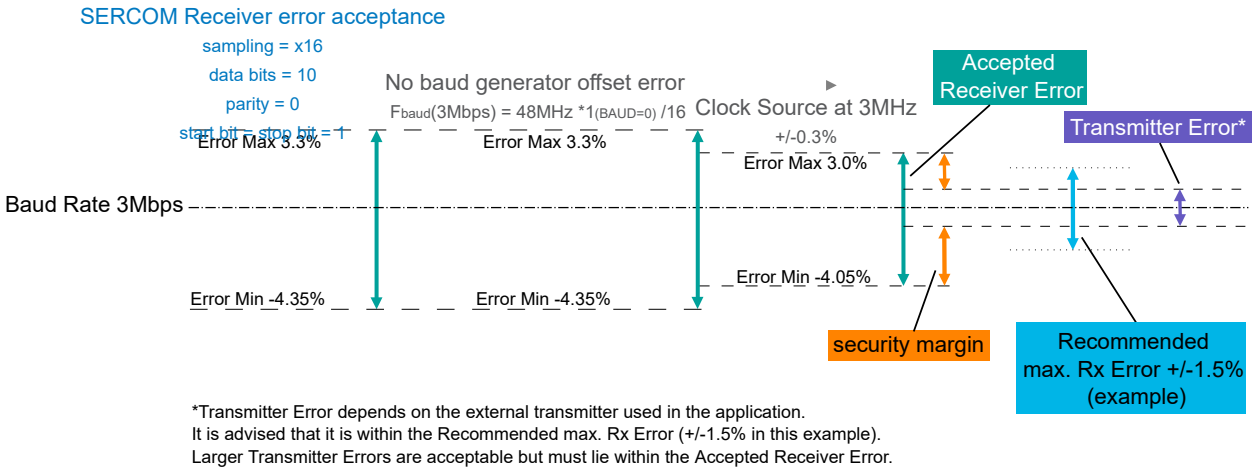
The recommended maximum Rx Error assumes that the receiver and transmitter equally divide the maximum total error. Its connection to the SERCOM Receiver error acceptance is depicted in this figure:

**Figure 31-5.** USART Rx Error Calculation




The recommendation values in the table above accommodate errors of the clock source and the baud generator. The following figure gives an example for a baud rate of 3Mbps:

**Figure 31-6.** USART Rx Error Calculation Example




*Transmitter Error depends on the external transmitter used in the application.
It is advised that it is within the Recommended max. Rx Error (+/-1.5% in this example).
Larger Transmitter Errors are acceptable but must lie within the Accepted Receiver Error.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Clock Generation – Baud-Rate Generator
Asynchronous Arithmetic Mode BAUD Value Selection

## 31.6.3. Additional Features

### 31.6.3.1. Parity

Even or odd parity can be selected for error checking by writing 0x1 to the Frame Format bit field in the Control A register (CTRLA.FORM).

If *even parity* is selected (CTRLB.PMODE=0), the parity bit of an outgoing frame is '1' if the data contains an odd number of bits that are '1', making the total number of '1' even.

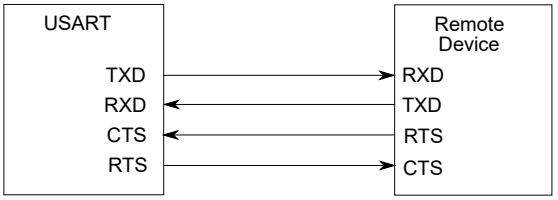
If *odd parity* is selected (CTRLB.PMODE=1), the parity bit of an outgoing frame is '1' if the data contains an even number of bits that are '0', making the total number of '1' odd.

When parity checking is enabled, the parity checker calculates the parity of the data bits in incoming frames and compares the result with the parity bit of the corresponding frame. If a parity error is detected, the Parity Error bit in the Status register (STATUS.PERR) is set.

### 31.6.3.2. Hardware Handshaking

The USART features an out-of-band hardware handshaking flow control mechanism, implemented by connecting the RTS and CTS pins with the remote device, as shown in the figure below.

**Figure 31-7.** Connection with a Remote Device for Hardware Handshaking

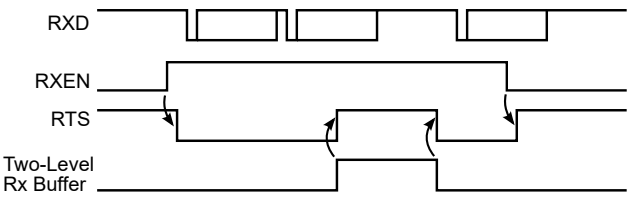



Hardware handshaking is only available in the following configuration:

- USART with internal clock (CTRLA.MODE = 1),
- Asynchronous mode (CTRLA.CMODE = 0),
- and Flow control pinout (CTRLA.TXPO = 2).

When the receiver is disabled or the two-level RX buffer is full, the receiver will drive the RTS pin high. This notifies the remote device to stop transfer after the ongoing transmission. Enabling and disabling the receiver by writing to CTRLB.RXEN will set/clear the RTS pin after a synchronization delay. When the two-level RX buffer goes full, RTS will be set immediately and the frame being received will be stored in the shift register until the two-level RX buffer is no longer full.

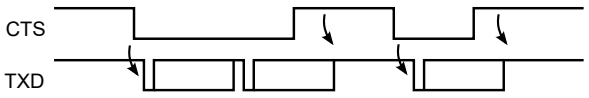
**Figure 31-8.** Receiver Behavior when Operating with Hardware Handshaking




The current CTS Status is in the STATUS register (STATUS.CTS). Character transmission will start only if STATUS.CTS=0. When CTS is set, the transmitter will complete the ongoing transmission and stop transmitting.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 31-9.** Transmitter Behavior when Operating with Hardware Handshaking



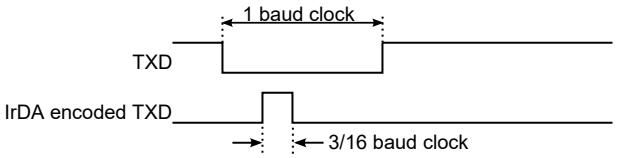

### 31.6.3.3. IrDA Modulation and Demodulation

Transmission and reception can be encoded IrDA compliant up to 115.2 kb/s. IrDA modulation and demodulation work in the following configuration:

- IrDA encoding enabled (CTRLB.ENC=1),
- Asynchronous mode (CTRLA.CMODE=0),
- and 16x sample rate (CTRLA.SAMPR[0]=0).

During transmission, each low bit is transmitted as a high pulse. The pulse width is 3/16 of the baud rate period, as illustrated in the figure below.

**Figure 31-10.** IrDA Transmit Encoding




The reception decoder has two main functions.

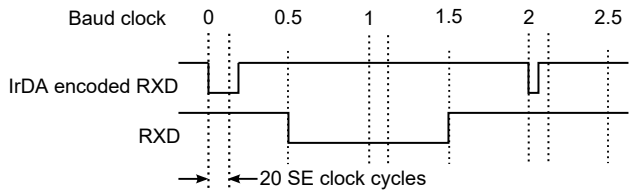
The first is to synchronize the incoming data to the IrDA baud rate counter. Synchronization is performed at the start of each zero pulse.

The second main function is to decode incoming Rx data. If a pulse width meets the minimum length set by configuration (RXPL.RXPL), it is accepted. When the baud rate counter reaches its middle value (1/2 bit length), it is transferred to the receiver.

**Note:** Note that the polarity of the transmitter and receiver are opposite: During transmission, a '0' bit is transmitted as a '1' pulse. During reception, an accepted '0' pulse is received as a '0' bit.

**Example:** The figure below illustrates reception where RXPL.RXPL is set to 19. This indicates that the pulse width should be at least 20 SE clock cycles. When using BAUD=0xE666 or 160 SE cycles per bit, this corresponds to 2/16 baud clock as minimum pulse width required. In this case the first bit is accepted as a '0', the second bit is a '1', and the third bit is also a '1'. A low pulse is rejected since it does not meet the minimum requirement of 2/16 baud clock.

**Figure 31-11.** IrDA Receive Decoding




### 31.6.3.4. Break Character Detection and Auto-Baud

Break character detection and auto-baud are available in this configuration:

- Auto-baud frame format (CTRLA.FORM = 0x04 or 0x05),


© 2026 Microchip Technology Inc. and its subsidiaries

- Asynchronous mode (CTRLA.CMODE = 0),
- and 16x sample rate using fractional baud rate generation (CTRLA.SAMPR = 1).

The USART uses a break detection threshold of greater than 11 nominal bit times at the configured baud rate. At any time, if more than 11 consecutive dominant bits are detected on the bus, the USART detects a Break Field. When a Break Field has been detected, the Receive Break interrupt flag (INTFLAG.RXBRK) is set and the USART expects the Sync Field character to be 0x55. This field is used to update the actual baud rate in order to stay synchronized. If the received Sync character is not 0x55, then the Inconsistent Sync Field error flag (STATUS.ISF) is set along with the Error interrupt flag (INTFLAG.ERROR), and the baud rate is unchanged.

After a break field is detected and the start bit of the Sync Field is detected, a counter is started. The counter is then incremented for the next 8 bit times of the Sync Field. At the end of these 8 bit times, the counter is stopped. At this moment, the 13 most significant bits of the counter (value divided by 8) give the new clock divider (BAUD.BAUD), and the 3 least significant bits of this value (the remainder) give the new Fractional Part (BAUD.FP).

When the Sync Field has been received, the clock divider (BAUD.BAUD) and the Fractional Part (BAUD.FP) are updated after a synchronization delay. After the Break and Sync Fields are received, multiple characters of data can be received.

#### 31.6.3.5. LIN Host

LIN Host is available with the following configuration:

- LIN Host format (CTRLA.FORM = 0x02)
- Asynchronous mode (CTRLA.CMODE = 0)
- 16x sample rate using fractional baud rate generation (CTRLA.SAMPR = 1)

LIN frames start with a header transmitted by the Host. The header consists of the break, sync, and identifier fields. After the Host transmits the header, the addressed Client will respond with 1-8 bytes of data plus checksum.

**Figure 31-12.** LIN Frame Format




Using the LIN command field (CTRLB.LINCMD), the complete header can be automatically transmitted, or software can control transmission of the various header components.

When CTRLB.LINCMD = 0x1, software controls transmission of the LIN header. In this case, software uses the following sequence.

- CTRLB.LINCMD is written to 0x1.
- The DATA register written to 0x00. This triggers transmission of the break field by hardware. Note that writing the DATA register with any other value will also result in the transmission of the break field by hardware.
- The DATA register written to 0x55. The 0x55 value (sync) is transmitted.
- The DATA register written to the identifier. The identifier is transmitted.

**Note:** It is recommended to use the Data Register Empty flag in the Interrupt Flag Status and Clear register (INTFLAG.DRE) to ensure no extra delay is added during the transmission of the data.

When CTRLB.LINCMD = 0x2, hardware controls transmission of the LIN header. In this case, software uses the following sequence.

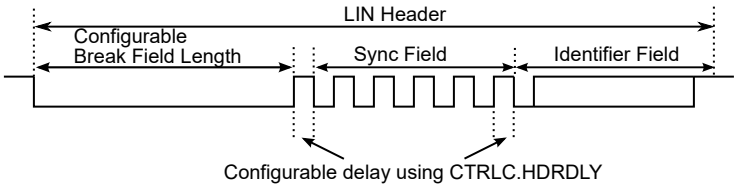
- CTRLB.LINCMD is written to 0x2.


© 2026 Microchip Technology Inc. and its subsidiaries

- The DATA register written to the identifier. This triggers transmission of the complete header by hardware. First the break field is transmitted. Next, the sync field is transmitted, and finally the identifier is transmitted.

In LIN Host mode, the length of the break field is programmable using the break length field (CTRLC.BRKLEN). When the LIN header command is used (CTRLB.LINCMD = 0x2), the delay between the break and sync fields, in addition to the delay between the sync and ID fields are configurable using the header delay field (CTRLC.HDRDLY). When manual transmission is used (CTRLB.LINCMD = 0x1), software controls the delay between break and sync.

**Figure 31-13.** LIN Header Generation




After header transmission is complete, the Client responds with 1-8 data bytes plus checksum.

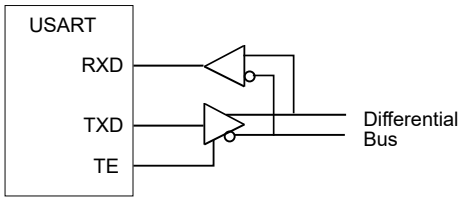
### 31.6.3.6. RS485

The RS485 feature is available with the following configuration:

- USART frame format (CTRLA.FORM = 0x00 or 0x01)
- RS485 pinout (CTRLA.TXPO = 0x3)

The RS485 feature enables control of an external line driver as shown in the figure below. While operating in RS485 mode, the transmit enable (TE) pin is driven high when the transmitter is active.

**Figure 31-14.** RS485 Bus Connection




The TE pin will remain high for the complete frame including stop bit. If a Guard Time is programmed in the Control C register (CTRLC.GTIME), the line will remain driven after the last character completion. The following figure shows a transfer with one stop bit and CTRLC.GTIME = 3.

**Figure 31-15.** Example of TE Drive with Guard Time




The Transmit Complete interrupt flag (INTFLAG.TXC) will be raised after the guard time is complete and TE goes low.

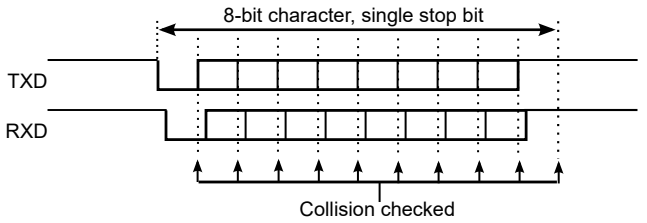
### 31.6.3.7. Collision Detection

When the receiver and transmitter are connected either through pin configuration or externally, transmit collision can be detected after selecting the Collision Detection Enable bit in the CTRLB register (CTRLB.COLDEN=1). To detect collision, the receiver and transmitter must be enabled (CTRLB.RXEN=1 and CTRLB.TXEN=1).


© 2026 Microchip Technology Inc. and its subsidiaries

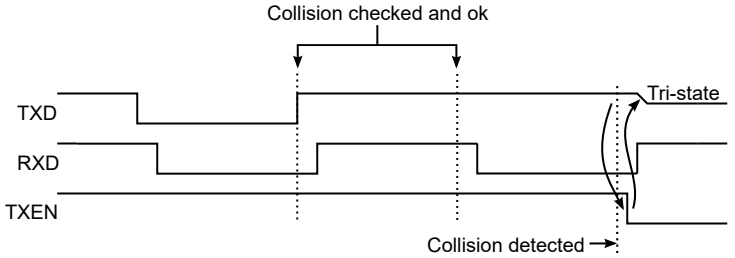
Collision detection is performed for each bit transmitted by comparing the received value with the transmit value, as shown in the figure below. While the transmitter is idle (no transmission in progress), characters can be received on RxD without triggering a collision.

**Figure 31-16.** Collision Checking




The next figure shows the conditions for a collision detection. In this case, the start bit and the first data bit are received with the same value as transmitted. The second received data bit is found to be different than the transmitted bit at the detection point, which indicates a collision.

**Figure 31-17.** Collision Detected




When a collision is detected, the USART follows this sequence:

1. Abort the current transfer.
2. Flush the transmit buffer.
3. Disable transmitter (CTRLB.TXEN=0)
   - This is done after a synchronization delay. The CTRLB Synchronization Busy bit (SYNCBUSY.CTRLB) will be set until this is complete.
   - After disabling, the TxD pin will be tri-stated.
4. Set the Collision Detected bit (STATUS.COLL) along with the Error interrupt flag (INTFLAG.ERROR).
5. Set the Transmit Complete interrupt flag (INTFLAG.TXC), since the transmit buffer no longer contains data.

After a collision, software must manually enable the transmitter again before continuing, after assuring that the CTRLB Synchronization Busy bit (SYNCBUSY.CTRLB) is not set.

### 31.6.3.8. Loop-Back Mode

For loop-back mode, configure the Receive Data Pinout (CTRLA.RXPO) and Transmit Data Pinout (CTRLA.TXPO) to use the same data pins for transmit and receive. The loop-back is through the pad, so the signal is also available externally.

### 31.6.3.9. Start-of-Frame Detection

The USART start-of-frame detector can wake up the CPU when it detects a start bit. In standby sleep mode, the internal fast startup oscillator must be selected as the GCLK_SERCOMx_CORE source.

When a 1-to-0 transition is detected on RxD, the 8MHz Internal Oscillator is powered up and the USART clock is enabled. After startup, the rest of the data frame can be received, provided that the baud rate is slow enough in relation to the fast startup internal oscillator start-up time. Refer to


© 2026 Microchip Technology Inc. and its subsidiaries

*Electrical Characteristics* for details. The start-up time of this oscillator varies with supply voltage and temperature.

The USART start-of-frame detection works both in asynchronous and synchronous modes. It is enabled by writing '1' to the Start of Frame Detection Enable bit in the Control B register (CTRLB.SFDE).

If the Receive Start Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.RXS) is set, the Receive Start interrupt is generated immediately when a start is detected.

When using start-of-frame detection without the Receive Start interrupt, start detection will force the 8MHz Internal Oscillator and USART clock active while the frame is being received. In this case, the CPU will not wake up until the Receive Complete interrupt is generated.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

#### 31.6.3.10. Sample Adjustment

In asynchronous mode (CTRLA.CMODE = 0), three samples in the middle are used to determine the value based on majority voting. The three samples used for voting can be selected using the Sample Adjustment bit field in the Control A register (CTRLA.SAMPA). When CTRLA.SAMPA = 0, samples 7-8-9 are used for 16x oversampling, and samples 3-4-5 are used for 8x oversampling.

**Note:** In full asynchronous mode, the start of frame may not occur at the UART clock reference rising edge meaning the counter can start incrementing from 0 to 1 in less than one UART clock reference period. The counter will then continue to increment at each positive edge of the UART clock reference regardless of the incoming bits.

### 31.6.4. DMA, Interrupts and Events

**Table 31-4.** Module Request for SERCOM USART

| Condition | Request | | |
|---|---|---|---|
| | DMA | Interrupt | Event |
| Data Register Empty (DRE) | Yes<br>(request cleared when data is written) | Yes | NA |
| Receive Complete (RXC) | Yes<br>(request cleared when data is read) | Yes | |
| Transmit Complete (TXC) | NA | Yes | |
| Receive Start (RXS) | NA | Yes | |
| Clear to Send Input Change (CTSIC) | NA | Yes | |
| Receive Break (RXBRK) | NA | Yes | |
| Error (ERROR) | NA | Yes | |

#### 31.6.4.1. DMA Operation

The USART generates the following DMA requests:

- Data received (RX): The request is set when data is available in the two-level RX buffer. The request is cleared when DATA is read.
- Data transmit (TX): The request is set when the transmit buffer (TX DATA) is empty. The request is cleared when DATA is written.

#### 31.6.4.2. Interrupts

The USART has the following interrupt sources. These are asynchronous interrupts, and can wake-up the device from any Sleep mode:

- Data Register Empty (DRE)


© 2026 Microchip Technology Inc. and its subsidiaries

- Receive Complete (RXC)
- Transmit Complete (TXC)
- Receive Start (RXS)
- Clear to Send Input Change (CTSIC)
- Received Break (RXBRK)
- Error (ERROR)

Each interrupt source has its own Interrupt flag. The Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the Interrupt condition is met. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the Interrupt flag is set and if the corresponding interrupt is enabled. The interrupt request remains active until either the Interrupt flag is cleared, the interrupt is disabled, or the USART is reset. For details on clearing Interrupt flags, refer to the INTFLAG register description.

The value of INTFLAG indicates which interrupt is executed. Note that interrupts must be globally enabled for interrupt requests. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

#### 31.6.4.3. Events

Not applicable.

### 31.6.5. Sleep Mode Operation

The behavior in sleep mode is depending on the clock source and the Run In Standby bit in the Control A register (CTRLA.RUNSTDBY):

- Internal clocking, CTRLA.RUNSTDBY=1: GCLK_SERCOMx_CORE can be enabled in all sleep modes. Any interrupt can wake up the device.
- External clocking, CTRLA.RUNSTDBY=1: The Receive Start and the Receive Complete interrupt(s) can wake up the device.
- Internal clocking, CTRLA.RUNSTDBY=0: Internal clock will be disabled, after any ongoing transfer was completed. The Receive Start and the Receive Complete interrupt(s) can wake up the device.
- External clocking, CTRLA.RUNSTDBY=0: External clock will be disconnected, after any ongoing transfer was completed. All reception will be dropped.

### 31.6.6. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in the CTRLA register (CTRLA.SWRST)
- Enable bit in the CTRLA register (CTRLA.ENABLE)
- Receiver Enable bit in the CTRLB register (CTRLB.RXEN)
- Transmitter Enable bit in the Control B register (CTRLB.TXEN)

**Note:** CTRLB.RXEN is write-synchronized somewhat differently. See also CTRLB for details.

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 31.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| | | 15:8 | SAMPR[2:0] | | | | | | | IBON |
| | | 23:16 | SAMPA[1:0] | | RXPO[1:0] | | | | TXPO[1:0] | |
| | | 31:24 | | DORD | CPOL | CMODE | FORM[3:0] | | | |
| 0x04 | CTRLB | 7:0 | | SBMODE | | | | CHSIZE[2:0] | | |
| | | 15:8 | | | PMODE | | | ENC | SFDE | COLDEN |
| | | 23:16 | | | | | | | RXEN | TXEN |
| | | 31:24 | | | | | | | LINCMD[1:0] | |
| 0x08 | CTRLC | 7:0 | | | | | | GTIME[2:0] | | |
| | | 15:8 | | | | | HDRDLY[1:0] | | BRKLEN[1:0] | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | BAUD | 7:0 | BAUD[7:0] | | | | | | | |
| | | 15:8 | BAUD[15:8] | | | | | | | |
| 0x0E | RXPL | 7:0 | RXPL[7:0] | | | | | | | |
| 0x0F ... 0x13 | Reserved | | | | | | | | | |
| 0x14 | INTENCLR | 7:0 | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| 0x15 | Reserved | | | | | | | | | |
| 0x16 | INTENSET | 7:0 | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| 0x17 | Reserved | | | | | | | | | |
| 0x18 | INTFLAG | 7:0 | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| 0x19 | Reserved | | | | | | | | | |
| 0x1A | STATUS | 7:0 | | TXE | COLL | ISF | CTS | BUFOVF | FERR | PERR |
| | | 15:8 | | | | | | | | |
| 0x1C | SYNCBUSY | 7:0 | | | | | | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 ... 0x27 | Reserved | | | | | | | | | |
| 0x28 | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | DATA[8] |
| 0x2A ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DBGCTRL | 7:0 | | | | | | | | DBGSTOP |

## 31.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | DORD | CPOL | CMODE | FORM[3:0] | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SAMPA[1:0] | | RXPO[1:0] | | | | TXPO[1:0] | |
| Access | R/W | R/W | R/W | R/W | | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SAMPR[2:0] | | | | | | | IBON |
| Access | R/W | R/W | R/W | | | | | R/W |
| Reset | 0 | 0 | 0 | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – DORD** Data Order
This bit selects the data order when a character is shifted out from the Data register.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | MSB is transmitted first. |
| 1 | LSB is transmitted first. |

**Bit 29 – CPOL** Clock Polarity
This bit selects the relationship between data output change and data input sampling in synchronous mode.
This bit is not synchronized.

| CPOL | TxD Change | RxD Sample |
|---|---|---|
| 0x0 | Rising XCK edge | Falling XCK edge |
| 0x1 | Falling XCK edge | Rising XCK edge |

**Bit 28 – CMODE** Communication Mode
This bit selects asynchronous or synchronous communication.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Asynchronous communication. |
| 1 | Synchronous communication. |

**Bits 27:24 – FORM[3:0]** Frame Format
These bits define the frame format.
These bits are not synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

| FORM[3:0] | Description |
|---|---|
| 0x0 | USART frame |
| 0x1 | USART frame with parity |
| 0x2 | LIN Host - Break and sync generation. See LIN Command (CTRLB.LINCMD). |
| 0x3 | Reserved |
| 0x4 | Auto-baud(LIN Client) - break detection and auto-baud. |
| 0x5 | Auto-baud - break detection and auto-baud with parity |
| 0x6-0xF | Reserved |

**Bits 23:22 – SAMPA[1:0]** Sample Adjustment
These bits define the sample adjustment.
These bits are not synchronized.

| SAMPA[1:0] | 16x Over-sampling (CTRLA.SAMPR=0 or 1) | 8x Over-sampling (CTRLA.SAMPR=2 or 3) |
|---|---|---|
| 0x0 | 7-8-9 | 3-4-5 |
| 0x1 | 9-10-11 | 4-5-6 |
| 0x2 | 11-12-13 | 5-6-7 |
| 0x3 | 13-14-15 | 6-7-8 |

**Bits 21:20 – RXPO[1:0]** Receive Data Pinout
These bits define the receive data (RxD) pin configuration.
These bits are not synchronized.

| RXPO[1:0] | Name | Description |
|---|---|---|
| 0x0 | PAD[0] | SERCOM PAD[0] is used for data reception |
| 0x1 | PAD[1] | SERCOM PAD[1] is used for data reception |
| 0x2 | PAD[2] | SERCOM PAD[2] is used for data reception |
| 0x3 | PAD[3] | SERCOM PAD[3] is used for data reception |

**Bits 17:16 – TXPO[1:0]** Transmit Data Pinout
These bits define the transmit data (TxD) and XCK pin configurations.
These bits are not synchronized.

| TXPO | TxD Pin Location | XCK Pin Location (When Applicable) | RTS/TE | CTS |
|---|---|---|---|---|
| 0x0 | SERCOM PAD[0] | SERCOM PAD[1] | N/A | N/A |
| 0x1 | SERCOM PAD[2] | SERCOM PAD[3] | N/A | N/A |
| 0x2 | SERCOM PAD[0] | N/A | SERCOM PAD[2] | SERCOM PAD[3] |
| 0x3 | SERCOM_PAD[0] | SERCOM_PAD[1] | SERCOM_PAD[2] | N/A |

**Bits 15:13 – SAMPR[2:0]** Sample Rate
These bits select the sample rate.
These bits are not synchronized.

| SAMPR[2:0] | Description |
|---|---|
| 0x0 | 16x over-sampling using arithmetic baud rate generation. |
| 0x1 | 16x over-sampling using fractional baud rate generation. |
| 0x2 | 8x over-sampling using arithmetic baud rate generation. |
| 0x3 | 8x over-sampling using fractional baud rate generation. |
| 0x4 | 3x over-sampling using arithmetic baud rate generation. |
| 0x5-0x7 | Reserved |

**Bit 8 – IBON** Immediate Buffer Overflow Notification
This bit controls when the buffer overflow status bit (STATUS.BUFOVF) is asserted when a buffer overflow occurs.

| Value | Description |
|---|---|
| 0 | STATUS.BUFOVF is asserted when it occurs in the data stream. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | STATUS.BUFOVF is asserted immediately upon buffer overflow. |

**Bit 7 – RUNSTDBY** Run In Standby

This bit defines the functionality in Standby sleep mode.
This bit is not synchronized.

| RUNSTDBY | External Clock | Internal Clock |
|---|---|---|
| 0x0 | External clock is disconnected when ongoing transfer is finished. All reception is dropped. | Generic clock is disabled when ongoing transfer is finished. The device will not wake up on either Receive Start or Transfer Complete interrupt unless the appropriate ONDEMAND bits are set in the clocking chain. |
| 0x1 | Wake on Receive Start or Receive Complete interrupt. | Generic clock is enabled in all sleep modes. Any interrupt can wake up the device. |

**Bits 4:2 – MODE[2:0]** Operating Mode

These bits select the USART serial communication interface of the SERCOM.
These bits are not synchronized.

| Value | Description |
|---|---|
| 0x0 | USART with external clock |
| 0x1 | USART with internal clock |

**Bit 1 – ENABLE** Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the Enable Synchronization Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE is cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled or being disabled. |
| 1 | The peripheral is enabled or being enabled. |

**Bit 0 – SWRST** Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.
Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.2. Control B

**Name:** CTRLB
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | LINCMD[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | RXEN | TXEN |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | PMODE | | | ENC | SFDE | COLDEN |
| Access | | | R/W | | | R/W | R/W | R/W |
| Reset | | | 0 | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | SBMODE | | | | CHSIZE[2:0] | | |
| Access | | R/W | | | | R/W | R/W | R/W |
| Reset | | 0 | | | | 0 | 0 | 0 |

**Bits 25:24 – LINCMD[1:0]** LIN Command

These bits define the LIN header transmission control. This field is only valid in LIN Host mode (CTRLA.FORM = LIN Host).
These are strobe bits and will always read back as zero.
These bits are not enable-protected.

| Value | Description |
|---|---|
| 0x0 | Normal USART transmission. |
| 0x1 | Break field is transmitted when DATA is written. |
| 0x2 | Break, sync, and identifier are automatically transmitted when DATA is written with the identifier. |
| 0x3 | Reserved |

**Bit 17 – RXEN** Receiver Enable

Writing '0' to this bit will disable the USART receiver. Disabling the receiver will flush the receive buffer and clear the FERR, PERR, and BUFOVF bits in the STATUS register.
Writing '1' to CTRLB.RXEN when the USART is disabled will set CTRLB.RXEN immediately. When the USART is enabled, CTRLB.RXEN will be cleared, and SYNCBUSY.CTRLB will be set and remain set until the receiver is enabled. When the receiver is enabled, CTRLB.RXEN will read back as '1'.
Writing '1' to CTRLB.RXEN when the USART is enabled will set SYNCBUSY.CTRLB, which will remain set until the receiver is enabled, and CTRLB.RXEN will read back as '1'.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The receiver is disabled or being enabled. |
| 1 | The receiver is enabled or will be enabled when the USART is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 16 – TXEN** Transmitter Enable

Writing '0' to this bit will disable the USART transmitter. Disabling the transmitter will not become effective until ongoing and pending transmissions are completed.
Writing '1' to CTRLB.TXEN when the USART is disabled will set CTRLB.TXEN immediately. When the USART is enabled, CTRLB.TXEN will be cleared, and SYNCBUSY.CTRLB will be set and remain set until the transmitter is enabled. When the transmitter is enabled, CTRLB.TXEN will read back as '1'.
Writing '1' to CTRLB.TXEN when the USART is enabled will set SYNCBUSY.CTRLB, which will remain set until the transmitter is enabled, and CTRLB.TXEN will read back as '1'.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The transmitter is disabled or being enabled. |
| 1 | The transmitter is enabled or will be enabled when the USART is enabled. |

**Bit 13 – PMODE** Parity Mode

This bit selects the type of parity used when parity is enabled (CTRLA.FORM is 0x1 or 0x5). The transmitter will automatically generate and send the parity of the transmitted data bits within each frame. The receiver will generate a parity value for the incoming data and parity bit, compare it to the parity mode and, if a mismatch is detected, STATUS.PERR will be set.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Even parity. |
| 1 | Odd parity. |

**Bit 10 – ENC** Encoding Format

This bit selects the data encoding format.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Data is not encoded. |
| 1 | Data is IrDA encoded. |

**Bit 9 – SFDE** Start of Frame Detection Enable

This bit controls whether the start-of-frame detector will wake up the device when a start bit is detected on the RxD line.
This bit is not synchronized.

| SFDE | INTENSET.RXS | INTENSET.RXC | Description |
|---|---|---|---|
| 0 | X | X | Start-of-frame detection disabled. |
| 1 | 0 | 0 | Reserved |
| 1 | 0 | 1 | Start-of-frame detection enabled. RXC wakes up the device from all sleep modes. |
| 1 | 1 | 0 | Start-of-frame detection enabled. RXS wakes up the device from all sleep modes. |
| 1 | 1 | 1 | Start-of-frame detection enabled. Both RXC and RXS wake up the device from all sleep modes. |

**Bit 8 – COLDEN** Collision Detection Enable

This bit enables collision detection.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Collision detection is not enabled. |
| 1 | Collision detection is enabled. |

**Bit 6 – SBMODE** Stop Bit Mode

This bit selects the number of stop bits transmitted.
This bit is not synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | One stop bit. |
| 1 | Two stop bits. |

**Bits 2:0 – CHSIZE[2:0]** Character Size
These bits select the number of bits in a character.
These bits are not synchronized.

| CHSIZE[2:0] | Description |
|---|---|
| 0x0 | 8 bits |
| 0x1 | 9 bits |
| 0x2-0x4 | Reserved |
| 0x5 | 5 bits |
| 0x6 | 6 bits |
| 0x7 | 7 bits |


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.3. Control C

**Name:** CTRLC
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | HDRDLY[1:0] | | BRKLEN[1:0] | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | GTIME[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 11:10 – HDRDLY[1:0]** LIN Host Header Delay
These bits define the delay between break and sync transmission in addition to the delay between the sync and identifier (ID) fields when in LIN Host mode (CTRLA.FORM = 0x2).
This field is only valid when using the LIN header command (CTRLB.LINCMD = 0x2).
**Note:** A correct setting for this bit field must be selected in order to follow the LIN specification.

| Value | Description |
|---|---|
| 0x0 | Delay between break and sync transmission is 1 bit time.<br>Delay between sync and ID transmission is 1 bit time. |
| 0x1 | Delay between break and sync transmission is 4 bit time.<br>Delay between sync and ID transmission is 4 bit time. |
| 0x2 | Delay between break and sync transmission is 8 bit time.<br>Delay between sync and ID transmission is 4 bit time. |
| 0x3 | Delay between break and sync transmission is 14 bit time.<br>Delay between sync and ID transmission is 4 bit time. |

**Bits 9:8 – BRKLEN[1:0]** LIN Host Break Length
These bits define the length of the break field transmitted when in LIN Host mode (CTRLA.FORM = 0x2).
**Note:** A correct setting for this bit field must be selected in order to follow the LIN specification.

| Value | Description |
|---|---|
| 0x0 | Break field transmission is 13 bit times |
| 0x1 | Break field transmission is 17 bit times |
| 0x2 | Break field transmission is 21 bit times |
| 0x3 | Break field transmission is 26 bit times |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 2:0 – GTIME[2:0]** Guard Time

These bits define the guard time when using RS485 mode (CTRLA.FORM = 0x0 or CTRLA.FORM = 0x1, and CTRLA.TXPO = 0x3).

For RS485 mode, the guard time is programmable from 0-7 bit times and defines the time that the transmit enable pin (TE) remains high after the last stop bit is transmitted and there is no remaining data to be transmitted.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.4. Baud

**Name:** BAUD
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** Enable-Protected, PAC Write-Protection

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | BAUD[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BAUD[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – BAUD[15:0]** Baud Value

Arithmetic Baud Rate Generation (`CTRLA.SAMPR[0]=0`):
These bits control the clock generation, as described in the SERCOM Baud Rate section.
If Fractional Baud Rate Generation (`CTRLA.SAMPR[0]=1`) bit positions 15 to 13 are replaced by FP[2:0] Fractional Part:

- **Bits 15:13 - FP[2:0]: Fractional Part**
  These bits control the clock generation, as described in the *SERCOM Clock Generation – Baud-Rate Generator* section.
- **Bits 12:0 - BAUD[12:0]: Baud Value**
  These bits control the clock generation, as described in the *SERCOM Clock Generation – Baud-Rate Generator* section.

**Related Links**
Clock Generation – Baud-Rate Generator
Asynchronous Arithmetic Mode BAUD Value Selection


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.5. Receive Pulse Length Register

**Name:** RXPL
**Offset:** 0x0E
**Reset:** 0x00
**Property:** Enable-Protected, PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RXPL[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – RXPL[7:0]** Receive Pulse Length
When the encoding format is set to IrDA (CTRLB.ENC=1), these bits control the minimum pulse length that is required for a pulse to be accepted by the IrDA receiver with regards to the serial engine clock period $SE_{per}$.
$PULSE \geq (\mathrm{RXPL} + 2) \cdot SE_{per}$


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.6. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| Access | R/W | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 5 – RXBRK** Receive Break Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Receive Break Interrupt Enable bit, which disables the Receive Break interrupt.

| Value | Description |
|---|---|
| 0 | Receive Break interrupt is disabled. |
| 1 | Receive Break interrupt is enabled. |

**Bit 4 – CTSIC** Clear to Send Input Change Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Clear To Send Input Change Interrupt Enable bit, which disables the Clear To Send Input Change interrupt.

| Value | Description |
|---|---|
| 0 | Clear To Send Input Change interrupt is disabled. |
| 1 | Clear To Send Input Change interrupt is enabled. |

**Bit 3 – RXS** Receive Start Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Receive Start Interrupt Enable bit, which disables the Receive Start interrupt.

| Value | Description |
|---|---|
| 0 | Receive Start interrupt is disabled. |
| 1 | Receive Start interrupt is enabled. |

**Bit 2 – RXC** Receive Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Receive Complete Interrupt Enable bit, which disables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Receive Complete interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Receive Complete interrupt is enabled. |

**Bit 1 – TXC** Transmit Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Transmit Complete Interrupt Enable bit, which disables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Transmit Complete interrupt is disabled. |
| 1 | Transmit Complete interrupt is enabled. |

**Bit 0 – DRE** Data Register Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Register Empty Interrupt Enable bit, which disables the Data Register Empty interrupt.

| Value | Description |
|---|---|
| 0 | Data Register Empty interrupt is disabled. |
| 1 | Data Register Empty interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.7. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x16
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| Access | R/W | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 5 – RXBRK** Receive Break Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Receive Break Interrupt Enable bit, which enables the Receive Break interrupt.

| Value | Description |
|---|---|
| 0 | Receive Break interrupt is disabled. |
| 1 | Receive Break interrupt is enabled. |

**Bit 4 – CTSIC** Clear to Send Input Change Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Clear To Send Input Change Interrupt Enable bit, which enables the Clear To Send Input Change interrupt.

| Value | Description |
|---|---|
| 0 | Clear To Send Input Change interrupt is disabled. |
| 1 | Clear To Send Input Change interrupt is enabled. |

**Bit 3 – RXS** Receive Start Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Receive Start Interrupt Enable bit, which enables the Receive Start interrupt.

| Value | Description |
|---|---|
| 0 | Receive Start interrupt is disabled. |
| 1 | Receive Start interrupt is enabled. |

**Bit 2 – RXC** Receive Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Receive Complete Interrupt Enable bit, which enables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Receive Complete interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Receive Complete interrupt is enabled. |

**Bit 1 – TXC** Transmit Complete Interrupt Enable

Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Transmit Complete Interrupt Enable bit, which enables the Transmit Complete interrupt.

| Value | Description |
|---|---|
| 0 | Transmit Complete interrupt is disabled. |
| 1 | Transmit Complete interrupt is enabled. |

**Bit 0 – DRE** Data Register Empty Interrupt Enable

Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Data Register Empty Interrupt Enable bit, which enables the Data Register Empty interrupt.

| Value | Description |
|---|---|
| 0 | Data Register Empty interrupt is disabled. |
| 1 | Data Register Empty interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.8. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x18
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | RXBRK | CTSIC | RXS | RXC | TXC | DRE |
| Access | R/W | | R/W | R/W | R/W | R | R/W | R |
| Reset | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error
This flag is cleared by writing '1' to it.
This bit is set when any error is detected. Errors that will set this flag have corresponding status flags in the STATUS register. Errors that will set this flag are COLL, ISF, BUFOVF, FERR, and PERR.Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 5 – RXBRK** Receive Break
This flag is cleared by writing '1' to it.
This flag is set when auto-baud is enabled (CTRLA.FORM) and a break character is received.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 4 – CTSIC** Clear to Send Input Change
This flag is cleared by writing a '1' to it.
This flag is set when a change is detected on the CTS pin.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 3 – RXS** Receive Start
This flag is cleared by writing '1' to it.
This flag is set when a start condition is detected on the RxD line and start-of-frame detection is enabled (CTRLB.SFDE is '1').
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Receive Start interrupt flag.

**Bit 2 – RXC** Receive Complete
This flag is cleared by reading the Data register (DATA) or by disabling the receiver.
This flag is set when there are unread data in DATA.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.

**Bit 1 – TXC** Transmit Complete
This flag is cleared by writing '1' to it or by writing new data to DATA.
This flag is set when the entire frame in the transmit shift register has been shifted out and there are no new data in DATA.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 0 – DRE** Data Register Empty
This flag is cleared by writing new data to DATA.
This flag is set when DATA is empty and ready to be written.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.9. Status

**Name:** STATUS
**Offset:** 0x1A
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | TXE | COLL | ISF | CTS | BUFOVF | FERR | PERR |
| Access | | R/W | R/W | R/W | R | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 6 – TXE** Transmitter Empty
When CTRLA.FORM is set to LIN Host mode, this bit is set when any ongoing transmission is complete and TxDATA is empty.
When CTRLA.FORM is not set to LIN Host mode, this bit will always read back as zero.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.

**Bit 5 – COLL** Collision Detected
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set when collision detection is enabled (CTRLB.COLDEN) and a collision is detected.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.

**Bit 4 – ISF** Inconsistent Sync Field
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set when the frame format is set to auto-baud (CTRLA.FORM) and a sync field not equal to 0x55 is received.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.

**Bit 3 – CTS** Clear to Send
This bit indicates the current level of the CTS pin when flow control is enabled (CTRLA.TXPO).
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.

**Bit 2 – BUFOVF** Buffer Overflow
Reading this bit before reading the Data register will indicate the error status of the next character to be read.
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set when a buffer overflow condition is detected. A buffer overflow occurs when the receive buffer is full, there is a new character waiting in the receive shift register and a new start bit is detected.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – FERR** Frame Error

Reading this bit before reading the Data register will indicate the error status of the next character to be read.
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set if the received character had a frame error, i.e., when the first stop bit is zero.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.

**Bit 0 – PERR** Parity Error

Reading this bit before reading the Data register will indicate the error status of the next character to be read.
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set if parity checking is enabled (CTRLA.FORM is 0x1, 0x5) and a parity error is detected.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.10. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CTRLB | ENABLE | SWRST |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CTRLB** CTRLB Synchronization Busy
Writing to the CTRLB register when the SERCOM is enabled requires synchronization. When writing to CTRLB the SYNCBUSY.CTRLB bit will be set until synchronization is complete. If CTRLB is written while SYNCBUSY.CTRLB is asserted, an APB error will be generated.

| Value | Description |
|---|---|
| 0 | CTRLB synchronization is not busy. |
| 1 | CTRLB synchronization is busy. |

**Bit 1 – ENABLE** SERCOM Enable Synchronization Busy
Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. When written, the SYNCBUSY.ENABLE bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | Enable synchronization is not busy. |
| 1 | Enable synchronization is busy. |

**Bit 0 – SWRST** Software Reset Synchronization Busy
Resetting the SERCOM (CTRLA.SWRST) requires synchronization. When written, the SYNCBUSY.SWRST bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | SWRST synchronization is not busy. |
| 1 | SWRST synchronization is busy. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.11. Data

**Name:** DATA
**Offset:** 0x28
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DATA[8] |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 8:0 – DATA[8:0]** Data

Reading these bits will return the contents of the Receive Data register. The register should be read only when the Receive Complete Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.RXC) is set. The status bits in STATUS should be read before reading the DATA value in order to get any corresponding error.

Writing these bits will write the Transmit Data register. This register should be written only when the Data Register Empty Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.DRE) is set.


© 2026 Microchip Technology Inc. and its subsidiaries

### 31.8.12. Debug Control

**Name:** DBGCTRL
**Offset:** 0x30
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGSTOP |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGSTOP** Debug Stop Mode
This bit controls the baud-rate generator functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The baud-rate generator continues normal operation when the CPU is halted by an external debugger. |
| 1 | The baud-rate generator is halted when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 32. SERCOM SPI – SERCOM Serial Peripheral Interface

## 32.1. Overview

The Serial Peripheral Interface (SPI) is one of the available modes in the Serial Communication Interface (SERCOM).

The SPI uses the SERCOM transmitter and receiver configured as shown in Block Diagram. Each side, host and client, depicts a separate SPI containing a Shift register, a transmit buffer and a two-level receive buffer. In addition, the SPI host uses the SERCOM baud-rate generator, while the SPI client can use the SERCOM address match logic. Labels in capital letters are synchronous to CLK_SERCOMx_APB and accessible by the CPU, while labels in lowercase letters are synchronous to the SCK clock.

**Related Links**

SERCOM – Serial Communication Interface

## 32.2. Features

SERCOM SPI includes the following features:

- Full-duplex, four-wire interface (MISO, MOSI, SCK, $\overline{SS}$)
- One-level transmit buffer, two-level receive buffer
- Supports all four SPI modes of operation
- Single data direction operation allows alternate function on MISO or MOSI pin
- Selectable LSB- or MSB-first data transfer
- Can be used with DMA
- Host operation:
  - Serial clock speed, $f_{SCK}=1/t_{SCK}$(1)
  - 8-bit clock generator
  - Hardware controlled $\overline{SS}$
- Client Operation:
  - Serial clock speed, $f_{SCK}=1/t_{SSCK}$(1)
  - Optional 8-bit address match operation
  - Operation in all sleep modes
  - Wake on $\overline{SS}$ transition

1. For $t_{SCK}$ and $t_{SSCK}$ values, refer to SPI Timing Characteristics.

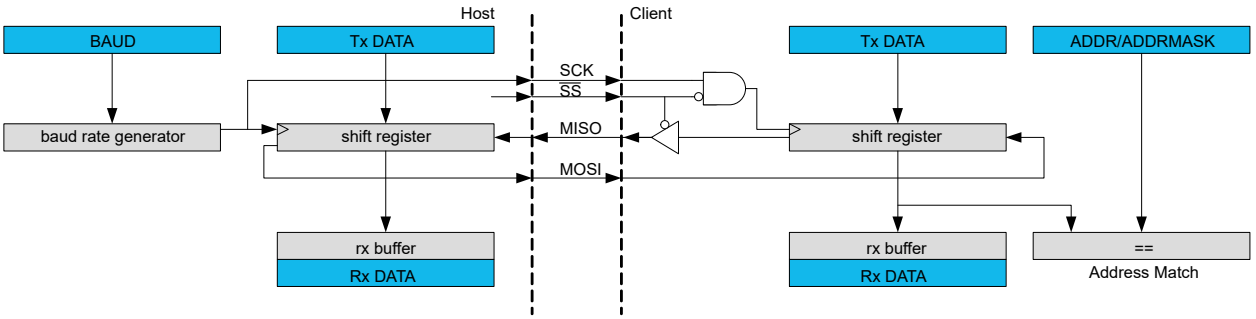
**Related Links**

SERCOM in SPI Mode Timing
SERCOM – Serial Communication Interface
Features


© 2026 Microchip Technology Inc. and its subsidiaries

## 32.3. Block Diagram

**Figure 32-1.** Full-Duplex SPI Host Client Interconnection




## 32.4. Signal Description

**Table 32-1.** SERCOM SPI Signals

| Signal Name | Type | Description |
|---|---|---|
| PAD[3:0] | Digital I/O | General SERCOM pins |

One signal can be mapped to one of several pins.

**Related Links**
I/O Multiplexing and Considerations

## 32.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 32.5.1. I/O Lines

In order to use the SERCOM's I/O lines, the I/O pins must be configured using the IO Pin Controller (PORT).

When the SERCOM is configured for SPI operation, the SERCOM controls the direction and value of the I/O pins according to the table below. If the receiver is disabled, the data input pin can be used for other purposes. In Host mode, the Client select line ($\overline{SS}$) is hardware controlled when the Host Client Select Enable bit in the Control B register (CTRLB.MSSEN) is '1'.

**Table 32-2.** SPI Pin Configuration

| Pin | Host SPI | Client SPI |
|---|---|---|
| MOSI | Output | Input |
| MISO | Input | Output |
| SCK | Output | Input |
| $\overline{SS}$ | Output (CTRLB.MSSEN=1) | Input |

The combined configuration of PORT, the Data In Pinout and the Data Out Pinout bit groups in the Control A register (CTRLA.DIPO and CTRLA.DOPO) define the physical position of the SPI signals in the table above.

**Related Links**
PORT - I/O Pin Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes.

**Related Links**
PM - Power Manager

### 32.5.3. Clocks

The SERCOM bus clock (CLK_SERCOMx_APB) can be enabled and disabled in the Main Clock Controller. Refer to *Peripheral Clock Masking* for details and default status of this clock.

A generic clock (GCLK_SERCOMx_CORE) is required to clock the SPI. This clock must be configured and enabled in the Generic Clock Controller before using the SPI.

This generic clock is asynchronous to the bus clock (CLK_SERCOMx_APB). Therefore, writes to certain registers will require synchronization to the clock domains.

**Related Links**
GCLK - Generic Clock Controller
Peripheral Clock Masking
Synchronization

### 32.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

**Related Links**
DMAC – Direct Memory Access Controller

### 32.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

### 32.5.6. Events

Not applicable.

### 32.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

### 32.5.8. Register Access Protection

Registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC).

PAC Write-Protection is not available for the following registers:

- Interrupt Flag Clear and Status register (INTFLAG)
- Status register (STATUS)


© 2026 Microchip Technology Inc. and its subsidiaries

- Data register (DATA)

Optional PAC Write-Protection is denoted by the "PAC Write-Protection" property in each individual register description.

Write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 32.5.9. Analog Connections

Not applicable.

## 32.6. Functional Description

### 32.6.1. Principle of Operation

The SPI is a high-speed synchronous data transfer interface. It allows high-speed communication between the device and peripheral devices.

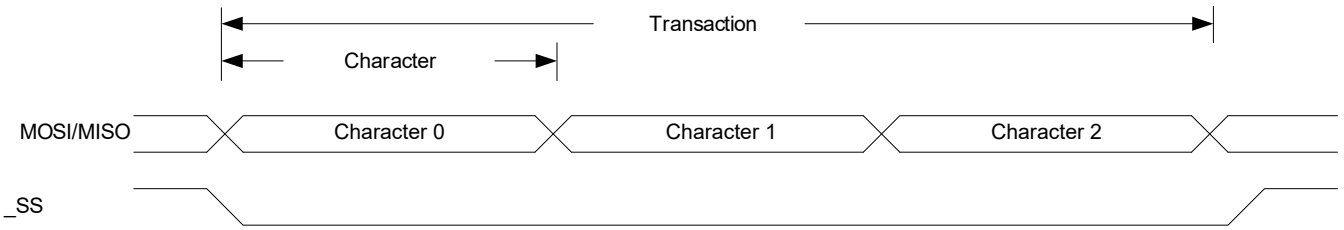
The SPI can operate as host or client. As host, the SPI initiates and controls all data transactions. The SPI is single buffered for transmitting and double buffered for receiving.

When transmitting data, the Data register can be loaded with the next character to be transmitted during the current transmission.

When receiving, the data is transferred to the two-level receive buffer, and the receiver is ready for a new character.

The SPI transaction format is shown in SPI Transaction Format. Each transaction can contain one or more characters. The character size is configurable, and can be either 8 or 9 bits.

**Figure 32-2.** SPI Transaction Format




The SPI host must pull the SPI select line ($\overline{SS}$) of the desired client low to initiate a transaction if multiple clients are connected to the bus. The SPI select line can be wired low if there is only one SPI client on the bus. The host and client prepare data to send via their respective Shift registers, and the host generates the serial clock on the SCK line.

Data is always shifted from host to client on the Host Output Client Input line (MOSI); data is shifted from client to host on the Host Input Client Output line (MISO).

Each time character is shifted out from the host, a character will be shifted out from the client simultaneously. To signal the end of a transaction, the host will pull the $\overline{SS}$ line high.

### 32.6.2. Basic Operation

#### 32.6.2.1. Initialization

The following registers are enable-protected, meaning that they can only be written when the SPI is disabled (CTRL.ENABLE=0):

- Control A register (CTRLA), except Enable (CTRLA.ENABLE) and Software Reset (CTRLA.SWRST)
- Control B register (CTRLB), except Receiver Enable (CTRLB.RXEN)


© 2026 Microchip Technology Inc. and its subsidiaries

- Baud register (BAUD)
- Address register (ADDR)

When the SPI is enabled or is being enabled (CTRLA.ENABLE=1), any writing to these registers will be discarded.

When the SPI is being disabled, writing to these registers will be completed after the disabling.

Enable-protection is denoted by the Enable-Protection property in the register description.

Initialize the SPI by following these steps:

1. Select SPI mode in host/client operation in the Operating Mode bit group in the CTRLA register (CTRLA.MODE= 0x2 or 0x3 ).
2. Select Transfer mode for the Clock Polarity bit and the Clock Phase bit in the CTRLA register (CTRLA.CPOL and CTRLA.CPHA) if desired.
3. Select the Frame Format value in the CTRLA register (CTRLA.FORM).
4. Configure the Data In Pinout field in the Control A register (CTRLA.DIPO) for SERCOM pads of the receiver.
5. Configure the Data Out Pinout bit group in the Control A register (CTRLA.DOPO) for SERCOM pads of the transmitter.
6. Select the Character Size value in the CTRLB register (CTRLB.CHSIZE).
7. Write the Data Order bit in the CTRLA register (CTRLA.DORD) for data direction.
8. If the SPI is used in Host mode:
   a. Select the desired baud rate by writing to the Baud register (BAUD).
   b. If Hardware $\overline{SS}$ control is required, write '1' to the Host SPI Select Enable bit in CTRLB register (CTRLB.MSSEN).
9. Enable the receiver by writing the Receiver Enable bit in the CTRLB register (CTRLB.RXEN=1).

#### 32.6.2.2. Enabling, Disabling, and Resetting

This peripheral is enabled by writing '1' to the Enable bit in the Control A register (CTRLA.ENABLE), and disabled by writing '0' to it.

Writing ‘1’ to the Software Reset bit in the Control A register (CTRLA.SWRST) will reset all registers of this peripheral to their initial states, except the DBGCTRL register, and the peripheral is disabled.

Refer to the CTRLA register description for details.

#### 32.6.2.3. Clock Generation

In SPI Host operation (CTRLA.MODE=0x3), the serial clock (SCK) is generated internally by the SERCOM baud-rate generator.

In SPI mode, the baud-rate generator is set to synchronous mode. The 8-bit Baud register (BAUD) value is used for generating SCK and clocking the shift register. Refer to *Clock Generation – Baud-Rate Generator* for more details.

In SPI Client operation (CTRLA.MODE is 0x2), the clock is provided by an external Host on the SCK pin. This clock is used to directly clock the SPI shift register.

**Related Links**

Clock Generation – Baud-Rate Generator

Asynchronous Arithmetic Mode BAUD Value Selection

#### 32.6.2.4. Data Register

The SPI Transmit Data register (TxDATA) and SPI Receive Data register (RxDATA) share the same I/O address, referred to as the SPI Data register (DATA). Writing DATA register will update the Transmit Data register. Reading the DATA register will return the contents of the Receive Data register.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 32.6.2.5. SPI Transfer Modes

There are four combinations of SCK phase and polarity to transfer serial data. The SPI data transfer modes are shown in SPI Transfer Modes (Table) and SPI Transfer Modes (Figure).

SCK phase is configured by the Clock Phase bit in the CTRLA register (CTRLA.CPHA). SCK polarity is programmed by the Clock Polarity bit in the CTRLA register (CTRLA.CPOL). Data bits are shifted out and latched in on opposite edges of the SCK signal. This ensures sufficient time for the data signals to stabilize.

**Table 32-3.** SPI Transfer Modes

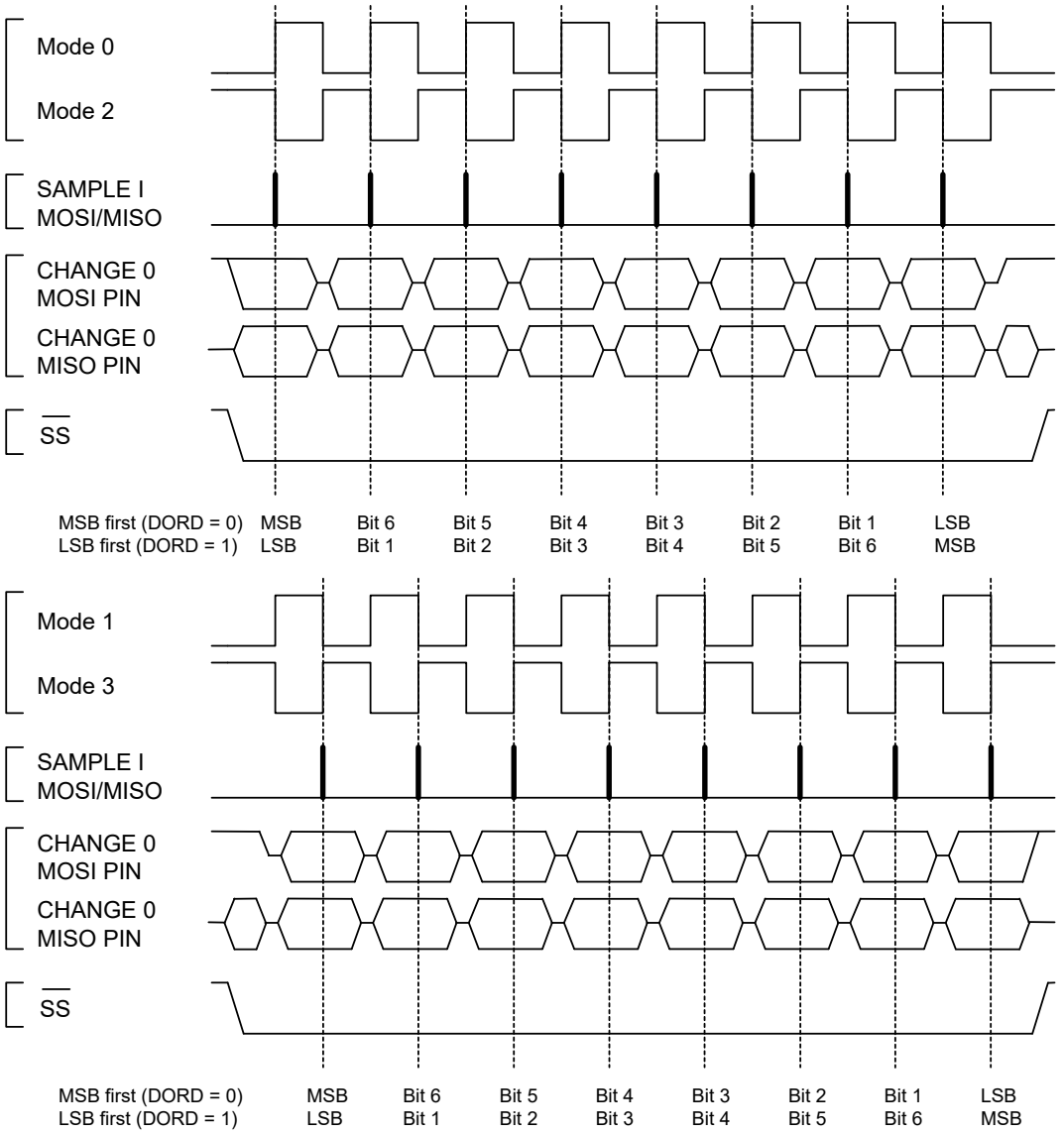
| Mode | CPOL | CPHA | Leading Edge | Trailing Edge |
|---|---|---|---|---|
| 0 | 0 | 0 | Rising, sample | Falling, setup |
| 1 | 0 | 1 | Rising, setup | Falling, sample |
| 2 | 1 | 0 | Falling, sample | Rising, setup |
| 3 | 1 | 1 | Falling, setup | Rising, sample |

**Note:**
Leading edge is the first clock edge in a clock cycle.

Trailing edge is the second clock edge in a clock cycle.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 32-3.** SPI Transfer Modes




### 32.6.2.6. Transferring Data

#### 32.6.2.6.1. Host

In Host mode (CTRLA.MODE=0x3), when Host Client Enable Select (CTRLB.MSSEN) is '1', hardware will control the $\overline{SS}$ line.

When Host Client Select Enable (CTRLB.MSSEN) is '0', the $\overline{SS}$ line must be configured as an output. $\overline{SS}$ can be assigned to any general purpose I/O pin. When the SPI is ready for a data transaction, software must pull the $\overline{SS}$ line low.

When writing a character to the Data register (DATA), the character will be transferred to the Shift register. Once the content of TxDATA has been transferred to the Shift register, the Data Register Empty flag in the Interrupt Flag Status and Clear register (INTFLAG.DRE) will be set, and a new character can be written to DATA.

Each time one character is shifted out from the Host, another character will be shifted in from the Client simultaneously. If the receiver is enabled (CTRLA.RXEN=1), the contents of the Shift register will be transferred to the two-level receive buffer. The transfer takes place in the same clock cycle as


© 2026 Microchip Technology Inc. and its subsidiaries

the last data bit is shifted in. Then the Receive Complete Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.RXC) will be set. The received data can be retrieved by reading DATA.

When the last character has been transmitted and there is no valid data in DATA, the Transmit Complete Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.TXC) will be set. When the transaction is finished, the Host must pull the $\overline{SS}$ line high to notify the Client. If Host Client Select Enable (CTRLB.MSSEN) is set to '0', the software must pull the $\overline{SS}$ line high.

#### 32.6.2.6.2. Client

In Client mode (CTRLA.MODE=0x2), the SPI interface will remain inactive with the MISO line tri-stated as long as the $\overline{SS}$ pin is pulled high. Software may update the contents of DATA at any time as long as the Data Register Empty flag in the Interrupt Status and Clear register (INTFLAG.DRE) is set.

When $\overline{SS}$ is pulled low and SCK is running, the client will sample and shift out data according to the Transaction mode set. When the content of TxDATA has been loaded into the Shift register, INTFLAG.DRE will be set, and new data can be written to DATA.

Similar to the host, the client will receive one character for each character transmitted. A character will be transferred into the two-level receive buffer within the same clock cycle its last data bit is received. The received character can be retrieved from DATA when the Receive Complete interrupt flag (INTFLAG.RXC) is set.

When the host pulls the $\overline{SS}$ line high, the transaction is done and the Transmit Complete Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.TXC) will be set.

After DATA is written it takes up to three SCK clock cycles until the content of DATA is ready to be loaded into the Shift register on the next character boundary. As a consequence, the first character transferred in a SPI transaction will not be the content of DATA. This can be avoided by using the preloading feature. Refer to Preloading of the Client Shift Register.

When transmitting several characters in one SPI transaction, the data has to be written into DATA register with at least three SCK clock cycles left in the current character transmission. If this criteria is not met, the previously received character will be transmitted.

Once the DATA register is empty, it takes three CLK_SERCOM_APB cycles for INTFLAG.DRE to be set.

#### 32.6.2.7. Receiver Error Bit

The SPI receiver has one error bit: the Buffer Overflow bit (BUFOVF), which can be read from the Status register (STATUS). Once an error happens, the bit will stay set until it is cleared by writing '1' to it. The bit is also automatically cleared when the receiver is disabled.

There are two methods for buffer overflow notification, selected by the immediate buffer overflow notification bit in the Control A register (CTRLA.IBON):

If CTRLA.IBON=1, STATUS.BUFOVF is raised immediately upon buffer overflow. Software can then empty the two-level RX buffer by reading RxDATA until the receiver complete interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.RXC) goes low.

If CTRLA.IBON=0, the buffer overflow condition travels with data through the two-level RX buffer. After the received data is read, STATUS.BUFOVF and INTFLAG.ERROR will be set along with INTFLAG.RXC, and RxDATA will be zero.

### 32.6.3. Additional Features

#### 32.6.3.1. Address Recognition

When the SPI is configured for client operation (CTRLA.MODE=0x2) with address recognition (CTRLA.FORM is 0x2), the SERCOM address recognition logic is enabled: the first character in a transaction is checked for an address match.

If there is a match, the Receive Complete Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.RXC) is set, the MISO output is enabled, and the transaction is processed. If the device is in Sleep mode, an address match can wake-up the device in order to process the transaction.


© 2026 Microchip Technology Inc. and its subsidiaries

If there is no match, the complete transaction is ignored.

If a 9-bit frame format is selected, only the lower 8 bits of the Shift register are checked against the Address register (ADDR).

Preload must be disabled (CTRLB.PLOADEN=0) in order to use this mode.

**Related Links**

Address Match and Mask

### 32.6.3.2. Preloading of the Client Shift Register

When starting a transaction, the client will first transmit the contents of the shift register before loading new data from DATA. The first character sent can be either the reset value of the shift register (if this is the first transmission since the last reset) or the last character in the previous transmission.

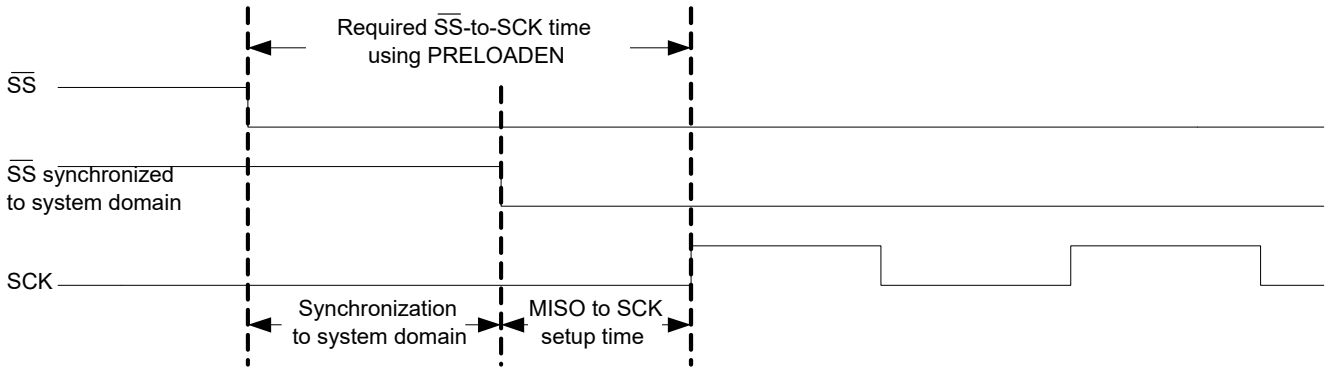
Preloading can be used to preload data into the shift register while $\overline{SS}$ is high: this eliminates sending a dummy character when starting a transaction. If the shift register is not preloaded, the current contents of the shift register will be shifted out.

Only one data character will be preloaded into the shift register while the synchronized $\overline{SS}$ signal is high. If the next character is written to DATA before $\overline{SS}$ is pulled low, the second character will be stored in DATA until transfer begins.

For proper preloading, sufficient time must elapse between $\overline{SS}$ going low and the first SCK sampling edge, as in Timing Using Preloading. See also the *Electrical Characteristics* chapters for timing details.

Preloading is enabled by writing '1' to the Client Data Preload Enable bit in the CTRLB register (CTRLB.PLOADEN).

**Figure 32-4.** Timing Using Preloading

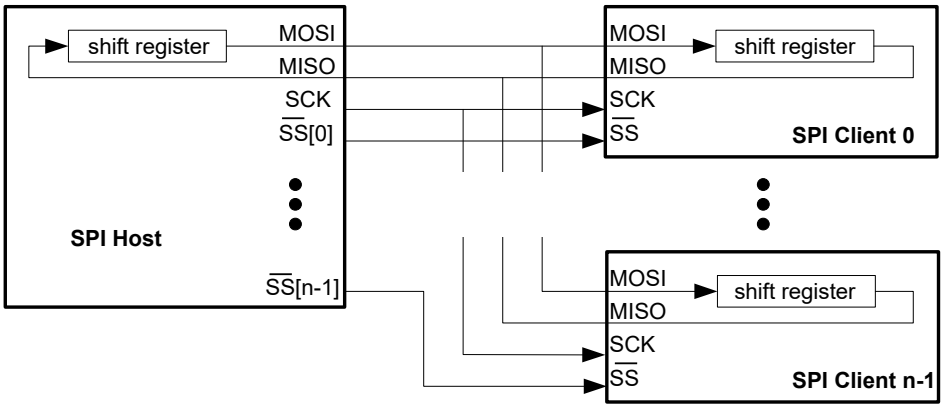



### 32.6.3.3. Host with Several Clients

Host with multiple clients in parallel is only available when Host SPI Select Enable (CTRLB.MSSEN) is set to zero and hardware $\overline{SS}$ control is disabled. If the bus consists of several SPI clients, a SPI host can use general purpose I/O pins to control the $\overline{SS}$ line to each of the clients on the bus, as shown in the following figure. In this configuration, the single selected SPI client will drive the tri-state MISO line.

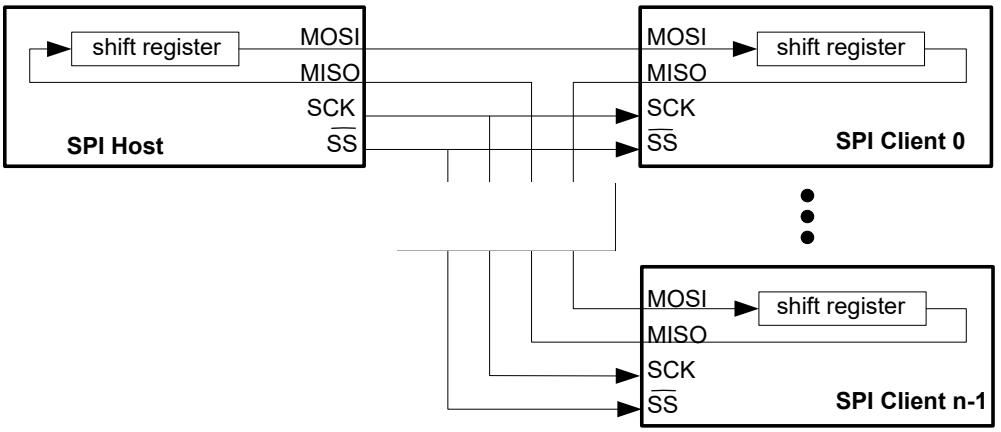

© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 32-5.** Multiple Clients in Parallel




Another configuration is multiple clients in series, as shown in the following figure. In this configuration, all n attached clients are connected in series. A common $\overline{SS}$ line is provided to all clients, enabling them simultaneously. The host must shift n characters for a complete transaction. Depending on the Host SPI Select Enable bit (CTRLB.MSSEN), the $\overline{SS}$ line can be controlled either by hardware or user software and normal GPIO.

**Figure 32-6.** Multiple Clients in Series




### 32.6.3.4. Loop-Back Mode

For loop-back mode, configure the Data In Pinout (CTRLA.DIPO) and Data Out Pinout (CTRLA.DOPO) to use the same data pins for transmit and receive. The loop-back is through the pad, so the signal is also available externally.

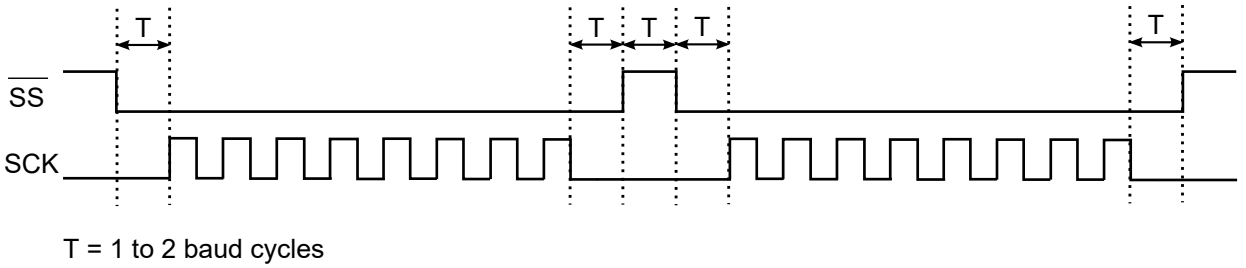
### 32.6.3.5. Hardware Controlled $\overline{SS}$

In Host mode, a single $\overline{SS}$ chip select can be controlled by hardware by writing the Host SPI Select Enable (CTRLB.MSSEN) bit to '1'. In this mode, the $\overline{SS}$ pin is driven low for a minimum of one baud cycle before transmission begins, and stays low for a minimum of one baud cycle after transmission completes. The $\overline{SS}$ pin will always be driven high for a minimum of one baud cycle between each data sent.

In Hardware Controlled $\overline{SS}$, the time T is between one and two baud cycles depending on the SPI Transfer mode.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 32-7.** Hardware Controlled $\overline{SS}$




When CTRLB.MSSEN=0, the $\overline{SS}$ pin(s) is/are controlled by user software and normal GPIO.

#### 32.6.3.6. SPI Select Low Detection

In Client mode, the SPI can wake the CPU when the SPI Select ($\overline{SS}$) goes low. When the SPI Select Low Detect is enabled (CTRLB.SSDE=1), a high-to-low transition will set the SPI Select Low Interrupt flag (INTFLAG.SSL) and the device will wake-up if applicable.

### 32.6.4. DMA, Interrupts, and Events

**Table 32-4.** Module Request for SERCOM SPI

| Condition | Request | | |
|---|---|---|---|
| | DMA | Interrupt | Event |
| Data Register Empty (DRE) | Yes<br>(request cleared when data is written) | Yes | NA |
| Receive Complete (RXC) | Yes<br>(request cleared when data is read) | Yes | |
| Transmit Complete (TXC) | NA | Yes | |
| SPI Select low (SSL) | NA | Yes | |
| Error (ERROR) | NA | Yes | |

#### 32.6.4.1. DMA Operation

The SPI generates the following DMA requests:

- Data received (RX): The request is set when data is available in the two-level RX buffer. The request is cleared when DATA is read.
- Data transmit (TX): The request is set when the transmit buffer (TX DATA) is empty. The request is cleared when DATA is written.

#### 32.6.4.2. Interrupts

The SPI has the following interrupt sources. These are asynchronous interrupts, and can wake-up the device from any Sleep mode:

- Data Register Empty (DRE)
- Receive Complete (RXC)
- Transmit Complete (TXC)
- SPI Select Low (SSL)
- Error (ERROR)

Each interrupt source has its own Interrupt flag. The Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the Interrupt condition is met. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.


© 2026 Microchip Technology Inc. and its subsidiaries

An interrupt request is generated when the Interrupt flag is set and if the corresponding interrupt is enabled. The interrupt request remains active until either the Interrupt flag is cleared, the interrupt is disabled, or the SPI is reset. For details on clearing Interrupt flags, refer to the INTFLAG register description.

The value of INTFLAG indicates which interrupt is executed. Note that interrupts must be globally enabled for interrupt requests. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

#### 32.6.4.3. Events

Not applicable.

### 32.6.5. Sleep Mode Operation

The behavior in sleep mode is depending on the Host/Client configuration and the Run In Standby bit in the Control A register (CTRLA.RUNSTDBY):

- **Host operation, CTRLA.RUNSTDBY=1:** The peripheral clock GCLK_SERCOM_CORE will continue to run in idle sleep mode and in standby sleep mode. Any interrupt can wake up the device.
- **Host operation, CTRLA.RUNSTDBY=0:** GLK_SERCOMx_CORE will be disabled after the ongoing transaction is finished. Any interrupt can wake up the device.
- **Client operation, CTRLA.RUNSTDBY=1:** The Receive Complete interrupt can wake up the device
- **Client operation, CTRLA.RUNSTDBY=0:** All reception will be dropped, including the ongoing transaction

### 32.6.6. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in the CTRLA register (CTRLA.SWRST)
- Enable bit in the CTRLA register (CTRLA.ENABLE)
- Receiver Enable bit in the CTRLB register (CTRLB.RXEN)

**Note:** CTRLB.RXEN is write-synchronized somewhat differently. See also *CTRLB* register for details.

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

**Related Links**
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 32.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | IBON |
| | | 23:16 | | | DIPO[1:0] | | | | DOPO[1:0] | |
| | | 31:24 | | DORD | CPOL | CPHA | FORM[3:0] | | | |
| 0x04 | CTRLB | 7:0 | | PLOADEN | | | | CHSIZE[2:0] | | |
| | | 15:8 | AMODE[1:0] | | MSSEN | | | | SSDE | |
| | | 23:16 | | | | | | | RXEN | |
| | | 31:24 | | | | | | | | |
| 0x08 ... 0x0B | Reserved | | | | | | | | | |
| 0x0C | BAUD | 7:0 | BAUD[7:0] | | | | | | | |
| 0x0D ... 0x13 | Reserved | | | | | | | | | |
| 0x14 | INTENCLR | 7:0 | ERROR | | | | SSL | RXC | TXC | DRE |
| 0x15 | Reserved | | | | | | | | | |
| 0x16 | INTENSET | 7:0 | ERROR | | | | SSL | RXC | TXC | DRE |
| 0x17 | Reserved | | | | | | | | | |
| 0x18 | INTFLAG | 7:0 | ERROR | | | | SSL | RXC | TXC | DRE |
| 0x19 | Reserved | | | | | | | | | |
| 0x1A | STATUS | 7:0 | | | | | | BUFOVF | | |
| | | 15:8 | | | | | | | | |
| 0x1C | SYNCBUSY | 7:0 | | | | | | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 ... 0x23 | Reserved | | | | | | | | | |
| 0x24 | ADDR | 7:0 | ADDR[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ADDRMASK[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x28 | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | DATA[8] |
| 0x2A ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DBGCTRL | 7:0 | | | | | | | | DBGSTOP |

## 32.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Refer to Synchronization

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Refer to Register Access Protection.


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | DORD | CPOL | CPHA | FORM[3:0] | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | DIPO[1:0] | | | | DOPO[1:0] | |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | IBON |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – DORD** Data Order
This bit selects the data order when a character is shifted out from the shift register.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | MSB is transferred first. |
| 1 | LSB is transferred first. |

**Bit 29 – CPOL** Clock Polarity
In combination with the Clock Phase bit (CPHA), this bit determines the SPI transfer mode.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | SCK is low when idle. The leading edge of a clock cycle is a rising edge, while the trailing edge is a falling edge. |
| 1 | SCK is high when idle. The leading edge of a clock cycle is a falling edge, while the trailing edge is a rising edge. |

**Bit 28 – CPHA** Clock Phase
In combination with the Clock Polarity bit (CPOL), this bit determines the SPI transfer mode.
This bit is not synchronized.

| Mode | CPOL | CPHA | Leading Edge | Trailing Edge |
|---|---|---|---|---|
| 0x0 | 0 | 0 | Rising, sample | Falling, change |
| 0x1 | 0 | 1 | Rising, change | Falling, sample |
| 0x2 | 1 | 0 | Falling, sample | Rising, change |
| 0x3 | 1 | 1 | Falling, change | Rising, sample |

| Value | Description |
|---|---|
| 0 | The data is sampled on a leading SCK edge and changed on a trailing SCK edge. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The data is sampled on a trailing SCK edge and changed on a leading SCK edge. |

**Bits 27:24 – FORM[3:0]** Frame Format

This bit field selects the various frame formats supported by the SPI in client mode. When the 'SPI frame with address' format is selected, the first byte received is checked against the ADDR register.

| FORM[3:0] | Name | Description |
|---|---|---|
| 0x0 | SPI | SPI frame |
| 0x1 | - | Reserved |
| 0x2 | SPI_ADDR | SPI frame with address |
| 0x3-0xF | - | Reserved |

**Bits 21:20 – DIPO[1:0]** Data In Pinout

These bits define the data in (DI) pad configurations.
In host operation, DI is MISO.
In client operation, DI is MOSI.
These bits are not synchronized.

| DIPO[1:0] | Name | Description |
|---|---|---|
| 0x0 | PAD[0] | SERCOM PAD[0] is used as data input |
| 0x1 | PAD[1] | SERCOM PAD[1] is used as data input |
| 0x2 | PAD[2] | SERCOM PAD[2] is used as data input |
| 0x3 | PAD[3] | SERCOM PAD[3] is used as data input |

**Bits 17:16 – DOPO[1:0]** Data Out Pinout

This bit defines the available pad configurations for data out (DO), the serial clock (SCK) and the SPI select ($\overline{SS}$). In Client operation, the SPI Select line ($\overline{SS}$) is controlled by DOPO. In host operation, the SPI Select line ($\overline{SS}$) is either controlled by DOPO when CTRLB.MSSEN = 1, or by a GPIO driven by the application when CTRLB.MSSEN = 0.
In host operation, DO is MOSI.
In client operation, DO is MISO.
These bits are not synchronized.

| DOPO | DO | SCK | Client $\overline{SS}$ | Host $\overline{SS}$ (MSSEN = 1) | Host $\overline{SS}$ (MSSEN = 0) |
|---|---|---|---|---|---|
| 0x0 | PAD[0] | PAD[1] | PAD[2] | PAD[2] | Any GPIO configured by the application |
| 0x1 | PAD[2] | PAD[3] | PAD[1] | PAD[1] | Any GPIO configured by the application |
| 0x2 | PAD[3] | PAD[1] | PAD[2] | PAD[2] | Any GPIO configured by the application |
| 0x3 | PAD[0] | PAD[3] | PAD[1] | PAD[1] | Any GPIO configured by the application |

**Bit 8 – IBON** Immediate Buffer Overflow Notification

This bit controls when the Buffer Overflow Status bit (STATUS.BUFOVF) is set when a buffer overflow occurs.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | STATUS.BUFOVF is set when it occurs in the data stream. |
| 1 | STATUS.BUFOVF is set immediately upon buffer overflow. |

**Bit 7 – RUNSTDBY** Run In Standby

This bit defines the functionality in Standby mode.
These bits are not synchronized.

| RUNSTDBY | Client | Host |
|---|---|---|
| 0x0 | Disabled. All reception is dropped, including the ongoing transaction. | Generic clock is disabled when ongoing transaction is finished. All interrupts can wake up the device. |


© 2026 Microchip Technology Inc. and its subsidiaries

| **RUNSTDBY** (continued) | | |
|---|---|---|
| **RUNSTDBY** | **Client** | **Host** |
| 0x1 | Ongoing transaction continues, wake on Receive Complete interrupt. | Generic clock is enabled while in sleep modes. All interrupts can wake up the device. |

**Bits 4:2 – MODE[2:0]** Operating Mode

These bits must be written to 0x2 or 0x3 to select the SPI serial communication interface of the SERCOM.
0x2: SPI client operation
0x3: SPI host operation
These bits are not synchronized.

**Bit 1 – ENABLE** Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the Synchronization Enable Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE is cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled or being disabled. |
| 1 | The peripheral is enabled or being enabled. |

**Bit 0 – SWRST** Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.
Writing ''1' to CTRL.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY. SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.2. Control B

**Name:** CTRLB
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | RXEN | |
| Access | | | | | | | R/W | |
| Reset | | | | | | | 0 | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | AMODE[1:0] | | MSSEN | | | | SSDE | |
| Access | R/W | R/W | R/W | | | | R/W | |
| Reset | 0 | 0 | 0 | | | | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | PLOADEN | | | | CHSIZE[2:0] | | |
| Access | | R/W | | | | R/W | R/W | R/W |
| Reset | | 0 | | | | 0 | 0 | 0 |

**Bit 17 – RXEN** Receiver Enable

Writing '0' to this bit will disable the SPI receiver immediately. The receive buffer will be flushed, data from ongoing receptions will be lost and STATUS.BUFOVF will be cleared.
Writing '1' to CTRLB.RXEN when the SPI is disabled will set CTRLB.RXEN immediately. When the SPI is enabled, CTRLB.RXEN will be cleared, SYNCBUSY.CTRLB will be set and remain set until the receiver is enabled. When the receiver is enabled CTRLB.RXEN will read back as '1'.
Writing '1' to CTRLB.RXEN when the SPI is enabled will set SYNCBUSY.CTRLB, which will remain set until the receiver is enabled, and CTRLB.RXEN will read back as '1'.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The receiver is disabled or being enabled. |
| 1 | The receiver is enabled or it will be enabled when SPI is enabled. |

**Bits 15:14 – AMODE[1:0]** Address Mode

These bits set the Client Addressing mode when the frame format (CTRLA.FORM) with address is used. They are unused in Host mode.
These bits are not synchronized.

| AMODE[1:0] | Name | Description |
|---|---|---|
| 0x0 | MASK | ADDRMASK is used as a mask to the ADDR register |
| 0x1 | 2_ADDRS | The client responds to the two unique addresses in ADDR and ADDRMASK |
| 0x2 | RANGE | The client responds to the range of addresses between and including ADDR and ADDRMASK. ADDR is the upper limit |
| 0x3 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 13 – MSSEN** Host SPI Select Enable

This bit enables hardware SPI Select ($\overline{SS}$) control.
This bit is not synchronized.
**Note:** When Hardware SPI Select Control is enabled (CTRLB.MSSEN = 1), the SPI Select (SS) pin goes high after each transferred word. Refer to the Hardware Controlled SS section for details.

| Value | Description |
|---|---|
| 0 | Hardware $\overline{SS}$ control is disabled. |
| 1 | Hardware $\overline{SS}$ control is enabled. |

**Bit 9 – SSDE** SPI Select Low Detect Enable

This bit enables wake-up when the SPI Select ($\overline{SS}$) pin transitions from high to low.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | $\overline{SS}$ low detector is disabled. |
| 1 | $\overline{SS}$ low detector is enabled. |

**Bit 6 – PLOADEN** Client Data Preload Enable

Setting this bit will enable preloading of the Client Shift register when there is no transfer in progress. If the $\overline{SS}$ line is high when DATA is written, it will be transferred immediately to the Shift register.
This bit is not synchronized.

**Bits 2:0 – CHSIZE[2:0]** Character Size

These bits are not synchronized.

| CHSIZE[2:0] | Name | Description |
|---|---|---|
| 0x0 | 8BIT | 8 bits |
| 0x1 | 9BIT | 9 bits |
| 0x2-0x7 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.3. Baud Rate

**Name:** BAUD
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BAUD[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – BAUD[7:0]** Baud Register
These bits control the clock generation, as described in the *SERCOM Clock Generation – Baud-Rate Generator*.

**Related Links**
Clock Generation – Baud-Rate Generator
Asynchronous Arithmetic Mode BAUD Value Selection


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | SSL | RXC | TXC | DRE |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 3 – SSL** SPI Select Low Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the SPI Select Low Interrupt Enable bit, which disables the SPI Select Low interrupt.

| Value | Description |
|---|---|
| 0 | SPI Select Low interrupt is disabled. |
| 1 | SPI Select Low interrupt is enabled. |

**Bit 2 – RXC** Receive Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Receive Complete Interrupt Enable bit, which disables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Receive Complete interrupt is disabled. |
| 1 | Receive Complete interrupt is enabled. |

**Bit 1 – TXC** Transmit Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Transmit Complete Interrupt Enable bit, which disable the Transmit Complete interrupt.

| Value | Description |
|---|---|
| 0 | Transmit Complete interrupt is disabled. |
| 1 | Transmit Complete interrupt is enabled. |

**Bit 0 – DRE** Data Register Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Register Empty Interrupt Enable bit, which disables the Data Register Empty interrupt.

| Value | Description |
|---|---|
| 0 | Data Register Empty interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Data Register Empty interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x16
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | SSL | RXC | TXC | DRE |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 3 – SSL** SPI Select Low Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the SPI Select Low Interrupt Enable bit, which enables the SPI Select Low interrupt.

| Value | Description |
|---|---|
| 0 | SPI Select Low interrupt is disabled. |
| 1 | SPI Select Low interrupt is enabled. |

**Bit 2 – RXC** Receive Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Receive Complete Interrupt Enable bit, which enables the Receive Complete interrupt.

| Value | Description |
|---|---|
| 0 | Receive Complete interrupt is disabled. |
| 1 | Receive Complete interrupt is enabled. |

**Bit 1 – TXC** Transmit Complete Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Transmit Complete Interrupt Enable bit, which enables the Transmit Complete interrupt.

| Value | Description |
|---|---|
| 0 | Transmit Complete interrupt is disabled. |
| 1 | Transmit Complete interrupt is enabled. |

**Bit 0 – DRE** Data Register Empty Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Data Register Empty Interrupt Enable bit, which enables the Data Register Empty interrupt.

| Value | Description |
|---|---|
| 0 | Data Register Empty interrupt is disabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Data Register Empty interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x18
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | SSL | RXC | TXC | DRE |
| Access | R/W | | | | R/W | R | R/W | R |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 7 – ERROR** Error
This flag is cleared by writing '1' to it.
This bit is set when any error is detected. Errors that will set this flag have corresponding Status flags in the STATUS register. The BUFOVF error will set this Interrupt flag.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 3 – SSL** SPI Select Low
This flag is cleared by writing '1' to it.
This bit is set when a high to low transition is detected on the $\overline{SS}$ pin in Client mode and SPI Select Low Detect (CTRLB.SSDE) is enabled.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 2 – RXC** Receive Complete
This flag is cleared by reading the Data (DATA) register or by disabling the receiver.
This flag is set when there are unread data in the receive buffer. If address matching is enabled, the first data received in a transaction will be an address.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.

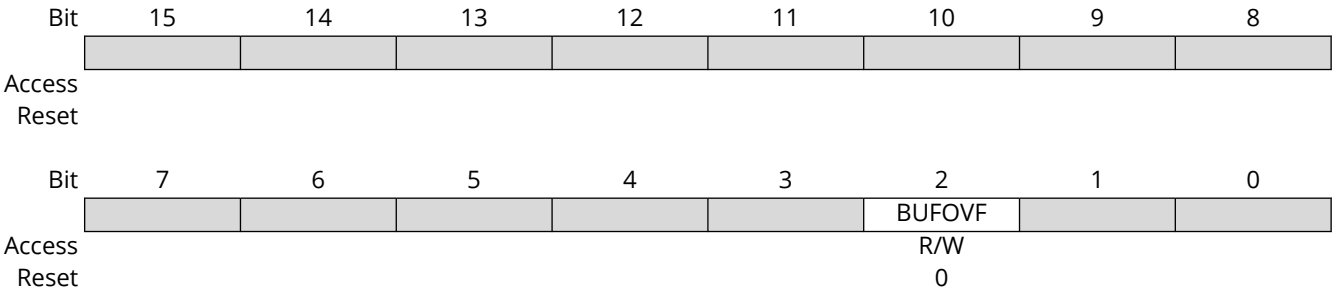
**Bit 1 – TXC** Transmit Complete
This flag is cleared by writing '1' to it or by writing new data to DATA.
In Host mode, this flag is set when the data have been shifted out and there are no new data in DATA.
In Client mode, this flag is set when the $\overline{SS}$ pin is pulled high. If address matching is enabled, this flag is only set if the transaction was initiated with an address match.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 0 – DRE** Data Register Empty
This flag is cleared by writing new data to DATA.
This flag is set when DATA is empty and ready for new data to transmit.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.7. Status

**Name:** STATUS
**Offset:** 0x1A
**Reset:** 0x0000
**Property:** –




**Bit 2 – BUFOVF** Buffer Overflow

Reading this bit before reading DATA will indicate the error status of the next character to be read.
This bit is cleared by writing '1' to the bit or by disabling the receiver.
This bit is set when a Buffer Overflow condition is detected. See also CTRLA.IBON for overflow handling.
When set, the corresponding RxDATA will be zero.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.

| Value | Description |
|---|---|
| 0 | No Buffer Overflow has occurred. |
| 1 | A Buffer Overflow has occurred. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.8. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CTRLB | ENABLE | SWRST |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – CTRLB** CTRLB Synchronization Busy

Writing to the CTRLB when the SERCOM is enabled requires synchronization. Ongoing synchronization is indicated by SYNCBUSY.CTRLB=1 until synchronization is complete. If CTRLB is written while SYNCBUSY.CTRLB=1, an APB error will be generated.

| Value | Description |
|---|---|
| 0 | CTRLB synchronization is not busy. |
| 1 | CTRLB synchronization is busy. |

**Bit 1 – ENABLE** SERCOM Enable Synchronization Busy

Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. Ongoing synchronization is indicated by SYNCBUSY.ENABLE=1 until synchronization is complete.

| Value | Description |
|---|---|
| 0 | Enable synchronization is not busy. |
| 1 | Enable synchronization is busy. |

**Bit 0 – SWRST** Software Reset Synchronization Busy

Resetting the SERCOM (CTRLA.SWRST) requires synchronization. Ongoing synchronization is indicated by SYNCBUSY.SWRST=1 until synchronization is complete.

| Value | Description |
|---|---|
| 0 | SWRST synchronization is not busy. |
| 1 | SWRST synchronization is busy. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.9. Address

**Name:** ADDR
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ADDRMASK[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:16 – ADDRMASK[7:0]** Address Mask
These bits hold the address mask when the transaction format with address is used (CTRLA.FORM, CTRLB.AMODE).

**Bits 7:0 – ADDR[7:0]** Address
These bits hold the address when the transaction format with address is used (CTRLA.FORM, CTRLB.AMODE).


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.10. Data

**Name:** DATA
**Offset:** 0x28
**Reset:** 0x0000
**Property:** –

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DATA[8] |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 8:0 – DATA[8:0]** Data

Reading these bits will return the contents of the receive data buffer. The register should be read only when the Receive Complete Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.RXC) is set.

Writing these bits will write the transmit data buffer. This register should be written only when the Data Register Empty Interrupt Flag bit in the Interrupt Flag Status and Clear register (INTFLAG.DRE) is set.


© 2026 Microchip Technology Inc. and its subsidiaries

### 32.8.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x30
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGSTOP |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGSTOP** Debug Stop Mode
This bit controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The baud-rate generator continues normal operation when the CPU is halted by an external debugger. |
| 1 | The baud-rate generator is halted when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 33. SERCOM I²C – Inter-Integrated Circuit

## 33.1. Overview

The inter-integrated circuit ( I²C) interface is one of the available modes in the serial communication interface (SERCOM).

The I²C interface uses the SERCOM transmitter and receiver configured as shown in Figure 33-1. Labels in capital letters are registers accessible by the CPU, while lowercase labels are internal to the SERCOM.

A SERCOM instance can be configured to be either an I²C host or an I²C client. Both host and client have an interface containing a shift register, a transmit buffer and a receive buffer. In addition, the I²C host uses the SERCOM baud-rate generator, while the I²C client uses the SERCOM address match logic.

**Related Links**

SERCOM – Serial Communication Interface

## 33.2. Features

SERCOM I²C includes the following features:

- Host or Client Operation
- Can be used with DMA
- Philips I²C Compatible
- SMBus Compatible
- PMBus™ Compatible
- Support of 100 kHz and 400 kHz, 1 MHz and 3.4 MHz I²C mode
- 4-Wire Operation Supported
- Physical interface includes:
  - Slew-rate limited outputs
  - Filtered inputs
- Client Operation:
  - Operation in all Sleep modes
  - Wake-up on address match
  - 7-bit and 10-bit Address match in hardware for:
  - 
    - Unique address and/or 7-bit general call address
    - Address range
    - Two unique addresses can be used with DMA

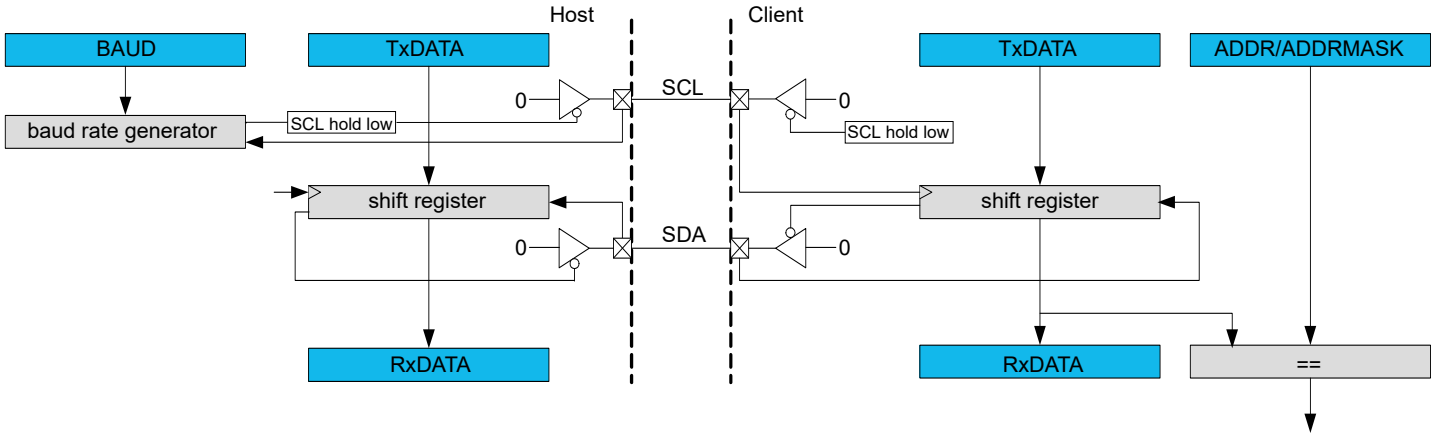
**Related Links**

Features


© 2026 Microchip Technology Inc. and its subsidiaries

## 33.3. Block Diagram

**Figure 33-1.** I²C Single-Host Single-Client Interconnection




## 33.4. Signal Description

| Signal Name | Type | Description |
|---|---|---|
| PAD[0] | Digital I/O | SDA |
| PAD[1] | Digital I/O | SCL |
| PAD[2] | Digital I/O | SDA_OUT (4-wire operation) |
| PAD[3] | Digital I/O | SCL_OUT (4-wire operation) |

One signal can be mapped on several pins.

Not all the pins are I²C pins. Refer to the SERCOM Pins Supporting I²C table for additional information.

Related Links:

Multiplexing and Considerations

4-Wire Mode

## 33.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 33.5.1. I/O Lines

In order to use the I/O lines of this peripheral, the I/O pins must be configured using the I/O Pin Controller (PORT).

When the SERCOM is used in I²C mode, the SERCOM controls the direction and value of the I/O pins. If the receiver or transmitter is disabled, these pins can be used for other purposes.

**Related Links**

PORT - I/O Pin Controller

### 33.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.5.3. Clocks

The SERCOM bus clock (CLK_SERCOMx_APB) can be enabled and disabled in the Main Clock Controller. Refer to *Peripheral Clock Masking* for details and default status of this clock.

Two generic clocks are used by SERCOM: GCLK_SERCOMx_CORE and GCLK_SERCOM_SLOW. The core clock (GCLK_SERCOMx_CORE) can clock the $I^2C$ when working as a host. The slow clock (GCLK_SERCOM_SLOW) is required only for certain functions, e.g. SMBus timing. These two clocks must be configured and enabled in the Generic Clock Controller (GCLK) before using the $I^2C$.

These generic clocks are asynchronous to the bus clock (CLK_SERCOMx_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
GCLK - Generic Clock Controller
Peripheral Clock Masking
PM - Power Manager

### 33.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

**Related Links**
DMAC – Direct Memory Access Controller

### 33.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

### 33.5.6. Events

Not applicable.

### 33.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

### 33.5.8. Register Access Protection

Registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC).

PAC Write-Protection is not available for the following registers:

- Interrupt Flag Clear and Status register (INTFLAG)
- Status register (STATUS)
- Data register (DATA)
- Address register (ADDR)

Optional PAC Write-Protection is denoted by the "PAC Write-Protection" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 33.5.9. Analog Connections

Not applicable.

## 33.6. Functional Description

### 33.6.1. Principle of Operation

The I²C interface uses two physical lines for communication:

- Serial Data Line (SDA) for data transfer
- Serial Clock Line (SCL) for the bus clock

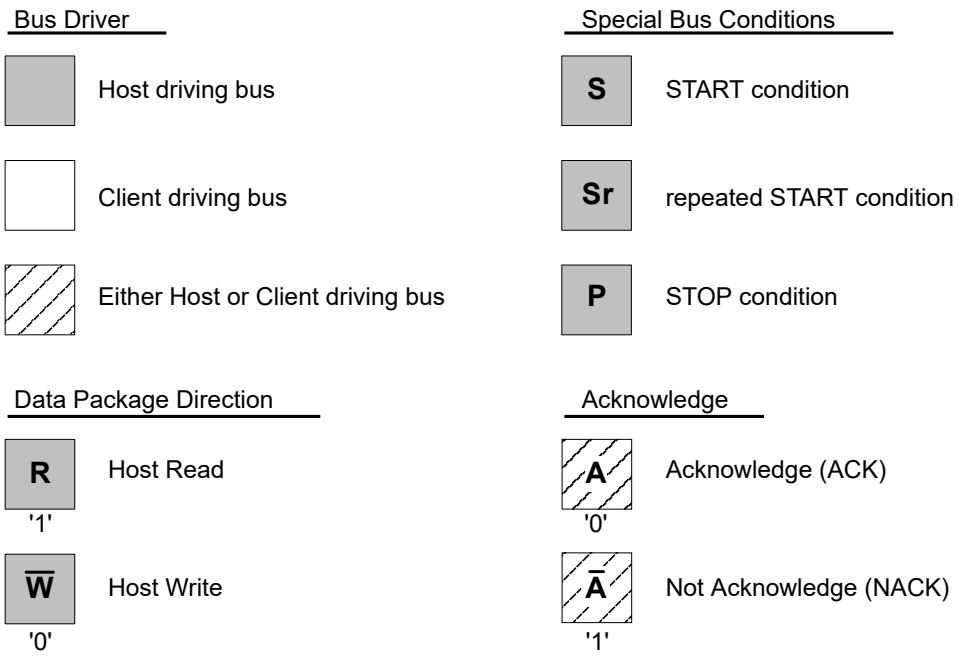
A transaction starts with the I²C host sending the Start condition, followed by a 7-bit address and a direction bit (read or write to/from the client).

The addressed I²C client will then Acknowledge (ACK) the address, and data packet transactions can begin. Every 9-bit data packet consists of 8 data bits followed by a one-bit reply indicating whether the data was acknowledged or not.

If a data packet is Not Acknowledged (NACK), whether by the I²C client or host, the I²C host takes action by either terminating the transaction by sending the Stop condition, or by sending a repeated start to transfer more data.

The figure below illustrates the possible transaction formats and Transaction Diagram Symbols explains the transaction symbols. These symbols will be used in the following descriptions.

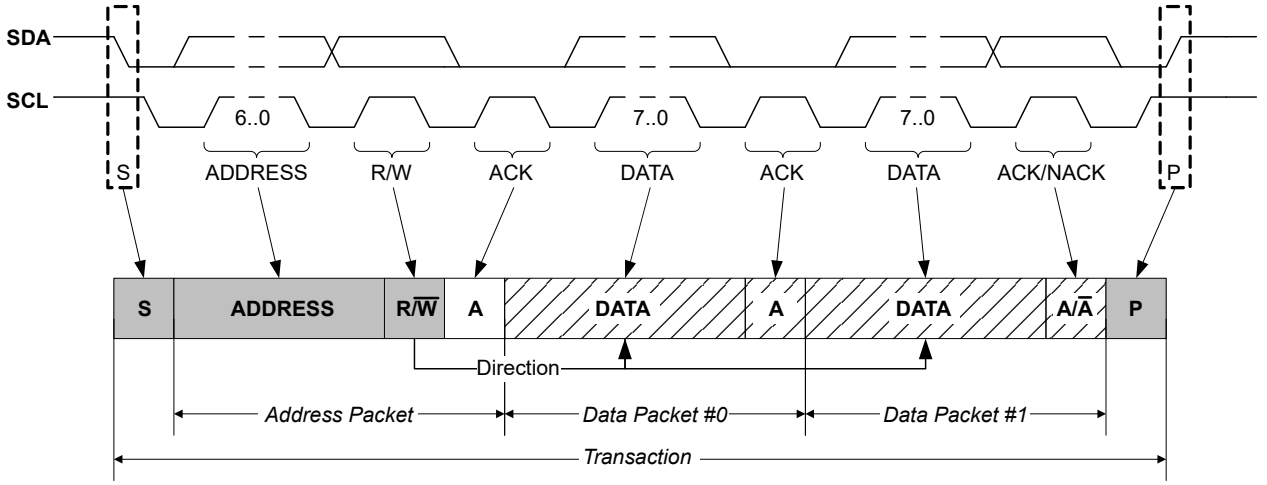
**Figure 33-2.** Transaction Diagram Symbols





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-3.** Basic I²C Transaction Diagram




## 33.6.2. Basic Operation

### 33.6.2.1. Initialization

The following registers are enable-protected, meaning they can be written only when the I²C interface is disabled (CTRLA.ENABLE is '0'):

- Control A register (CTRLA), except Enable (CTRLA.ENABLE) and Software Reset (CTRLA.SWRST) bits
- Control B register (CTRLB), except Acknowledge Action (CTRLB.ACKACT) and Command (CTRLB.CMD) bits
- Baud register (BAUD)
- Address register (ADDR) in client operation.

When the I²C is enabled or is being enabled (CTRLA.ENABLE=1), writing to these registers will be discarded. If the I²C is being disabled, writing to these registers will be completed after the disabling.

Enable-protection is denoted by the "Enable-Protection" property in the register description.

Before the I²C is enabled it must be configured as outlined by the following steps:

1. Select I²C Host or Client mode by writing 0x4 (Client mode) or 0x5 (Host mode) to the Operating Mode bits in the CTRLA register (CTRLA.MODE).
2. If desired, select the SDA Hold Time value in the CTRLA register (CTRLA.SDAHOLD).
3. If desired, enable smart operation by setting the Smart Mode Enable bit in the CTRLB register (CTRLB.SMEN).
4. If desired, enable SCL low time-out by setting the SCL Low Time-Out bit in the Control A register (CTRLA.LOWTOUTEN).
5. In Host mode:
   a. Select the inactive bus time-out in the Inactive Time-Out bit group in the CTRLA register (CTRLA.INACTOUT).
   b. Write the Baud Rate register (BAUD) to generate the desired baud rate.

   In Client mode:
   a. Configure the address match configuration by writing the Address Mode value in the CTRLB register (CTRLB.AMODE).
   b. Set the Address and Address Mask value in the Address register (ADDR.ADDR and ADDR.ADDRMASK) according to the address configuration.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.6.2.2. Enabling, Disabling, and Resetting

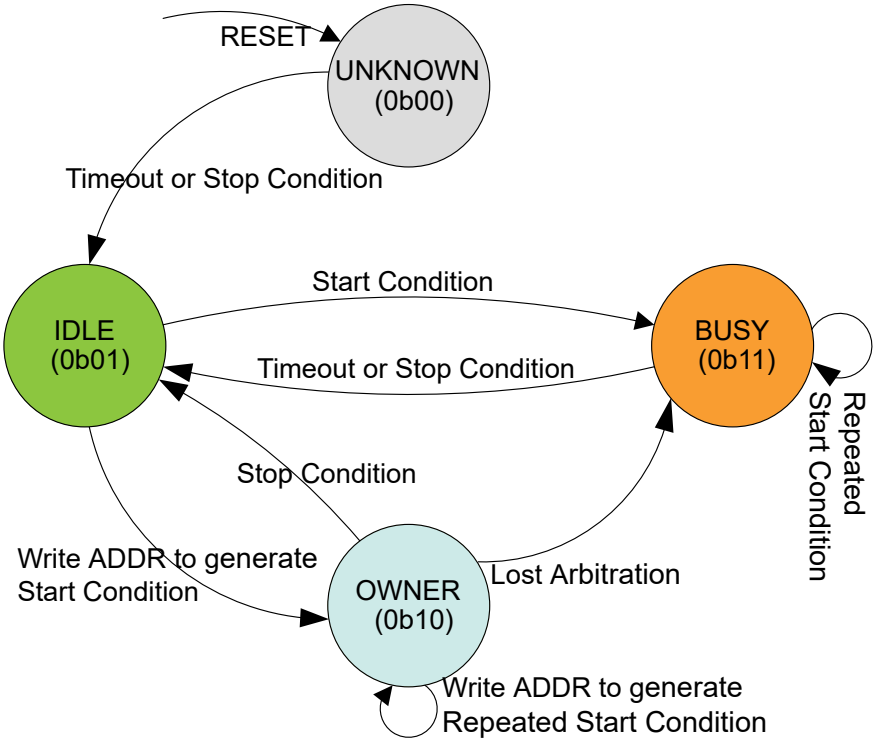
This peripheral is enabled by writing '1' to the Enable bit in the Control A register (CTRLA.ENABLE), and disabled by writing '0' to it.

Writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST) will reset all registers of this peripheral to their initial states, except the DBGCTRL register, and the peripheral is disabled.

### 33.6.2.3. I²C Bus State Logic

The Bus state logic includes several logic blocks that continuously monitor the activity on the I²C bus lines in all Sleep modes with running GCLK_SERCOM_x clocks. The start and stop detectors and the bit counter are all essential in the process of determining the current Bus state. The Bus state is determined according to Bus State Diagram. Software can get the current Bus state by reading the Host Bus State bits in the Status register (STATUS.BUSSTATE). The value of STATUS.BUSSTATE in the figure is shown in binary.

**Figure 33-4.** Bus State Diagram




The Bus state machine is active when the I²C host is enabled.

After the I²C host has been enabled, the Bus state is UNKNOWN (0b00). From the UNKNOWN state, the bus will transition to IDLE (0b01) by either:

- Forcing by writing 0b01 to STATUS.BUSSTATE
- A Stop condition is detected on the bus
- If the inactive bus time-out is configured for SMBus compatibility (CTRLA.INACTOUT) and a time-out occurs.

**Note:** Once a known Bus state is established, the Bus state logic will not re-enter the UNKNOWN state.

When the bus is IDLE it is ready for a new transaction. If a Start condition is issued on the bus by another I²C host in a multi-host setup, the bus becomes BUSY (0b11). The bus will re-enter IDLE either when a Stop condition is detected, or when a time-out occurs (inactive bus time-out needs to be configured).


© 2026 Microchip Technology Inc. and its subsidiaries

If a Start condition is generated internally by writing the Address bit group in the Address register (ADDR.ADDR) while IDLE, the OWNER state (0b10) is entered. If the complete transaction was performed without interference, i.e., arbitration was not lost, the I$^2$C host can issue a Stop condition, which will change the Bus state back to IDLE.

However, if a packet collision is detected while in OWNER state, the arbitration is assumed lost and the Bus state becomes BUSY until a Stop condition is detected. A repeated Start condition will change the Bus state only if arbitration is lost while issuing a repeated start.

**Note:** Violating the protocol may cause the I$^2$C to hang. If this happens it is possible to recover from this state by a software Reset (CTRLA.SWRST='1').

**Related Links**

CTRLA

Control A

#### 33.6.2.4. I$^2$C Host Operation

The I$^2$C host is byte-oriented and interrupt based. The number of interrupts generated is kept at a minimum by automatic handling of most incidents. The software driver complexity and code size are reduced by auto-triggering of operations, and a Special Smart mode, which can be enabled by the Smart Mode Enable bit in the Control B register (CTRLB.SMEN).

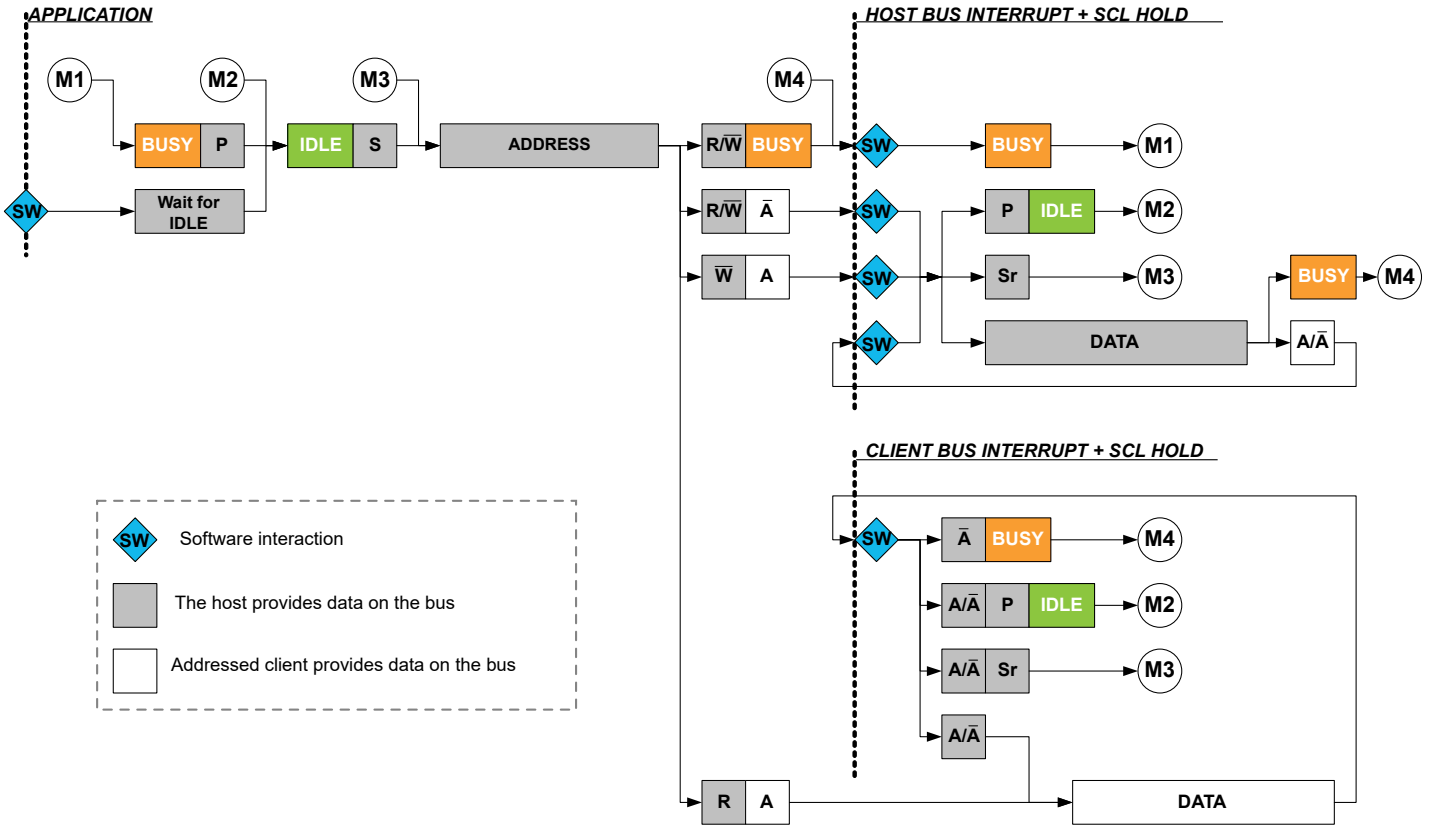
The I$^2$C host has two interrupt strategies.

When SCL Clock Stretch Mode (CTRLA.SCLSM) is '0', SCL is stretched before or after the Acknowledge bit . In this mode the I$^2$C host operates according to the following figure. The circles labeled "M*n*" (M1, M2..) indicate the nodes the bus logic can jump to, based on software or hardware interaction.

This diagram is used as reference for the description of the I$^2$C host operation throughout the document.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-5.** I²C Host Behavioral Diagram (SCLSM=0)

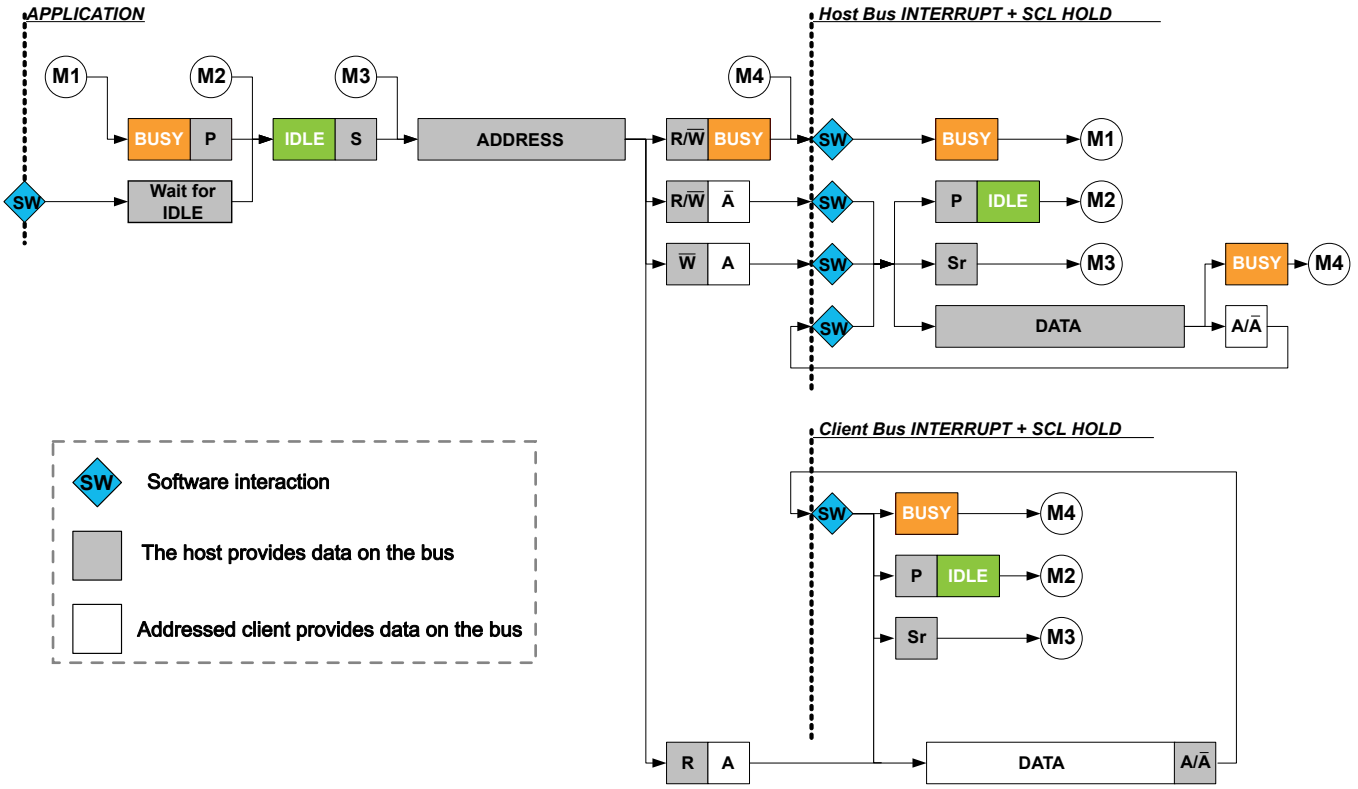



In the second strategy (CTRLA.SCLSM=1), interrupts only occur after the ACK bit, as shown in the following figure. This strategy can be used when it is not necessary to check DATA before acknowledging.

**Note:** I²C High-speed (*Hs*) mode requires CTRLA.SCLSM=1.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-6.** I²C Host Behavioral Diagram (SCLSM=1)




### 33.6.2.4.1. Host Clock Generation

The SERCOM peripheral supports several I²C bidirectional modes:

- Standard mode (*Sm*) up to 100 kHz
- Fast mode (*Fm*) up to 400 kHz
- Fast mode Plus (*Fm+*) up to 1 MHz
- High-speed mode (*Hs*) up to 3.4 MHz

The Host clock configuration for *Sm, Fm,* and *Fm+* are described in Clock Generation (Standard-Mode, Fast-Mode, and Fast-Mode Plus). For *Hs*, refer to Host Clock Generation (High-Speed Mode).

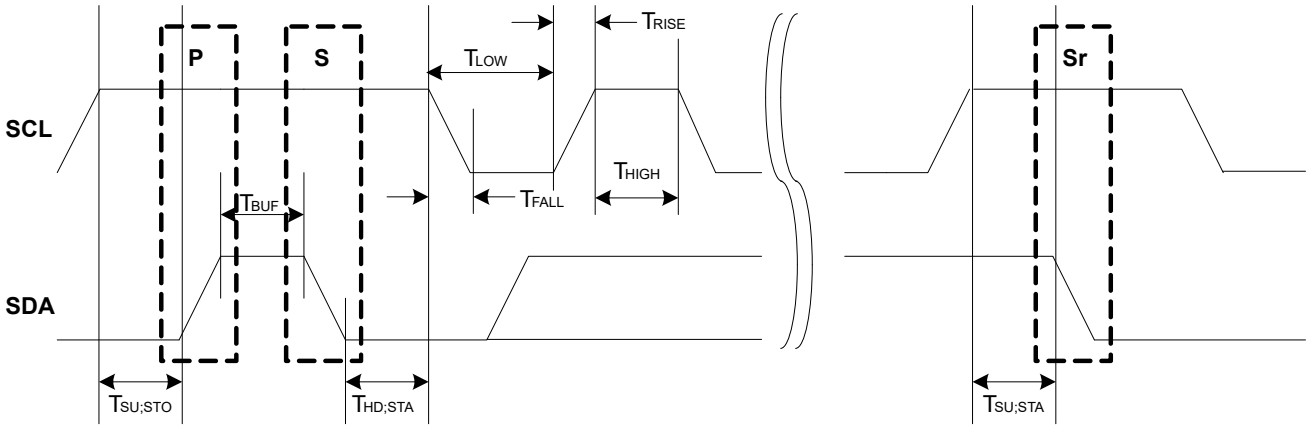
#### Clock Generation (Standard-Mode, Fast-Mode, and Fast-Mode Plus)

In I²C *Sm, Fm*, and *Fm+* mode, the Host clock (SCL) frequency is determined as described in this section:

The low ($T_{LOW}$) and high ($T_{HIGH}$) times are determined by the Baud Rate register (BAUD), while the rise ($T_{RISE}$) and fall ($T_{FALL}$) times are determined by the bus topology. Because of the wired-AND logic of the bus, $T_{FALL}$ will be considered as part of $T_{LOW}$. Likewise, $T_{RISE}$ will be in a state between $T_{LOW}$ and $T_{HIGH}$ until a high state has been detected.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-7.** SCL Timing




The following parameters are timed using the SCL low time period $T_{LOW}$. This comes from the Host Baud Rate Low bit group in the Baud Rate register (BAUD.BAUDLOW). When BAUD.BAUDLOW=0, or the Host Baud Rate bit group in the Baud Rate register (BAUD.BAUD) determines it.

- $T_{LOW}$ – Low period of SCL clock
- $T_{SU;STO}$ – Set-up time for stop condition
- $T_{BUF}$ – Bus free time between stop and start conditions
- $T_{HD;STA}$ – Hold time (repeated) start condition
- $T_{SU;STA}$ – Set-up time for repeated start condition
- $T_{HIGH}$ is timed using the SCL high time count from BAUD.BAUD
- $T_{RISE}$ is determined by the bus impedance; for internal pull-ups.
- $T_{FALL}$ is determined by the open-drain current limit and bus impedance; can typically be regarded as zero.

The SCL frequency is given by:

$$f_{\mathrm{SCL}} = \frac{1}{T_{\mathrm{LOW}} + T_{\mathrm{HIGH}} + T_{\mathrm{RISE}}}$$

When BAUD.BAUDLOW is zero, the BAUD.BAUD value is used to time both SCL high and SCL low. In this case the following formula will give the SCL frequency:

$$f_{\mathrm{SCL}} = \frac{f_{\mathrm{GCLK}}}{10 + 2BAUD + f_{\mathrm{GCLK}} \cdot T_{\mathrm{RISE}}}$$

When BAUD.BAUDLOW is non-zero, the following formula determines the SCL frequency:

$$f_{\mathrm{SCL}} = \frac{f_{\mathrm{GCLK}}}{10 + BAUD + BAUDLOW + f_{\mathrm{GCLK}} \cdot T_{\mathrm{RISE}}}$$

The following formulas can determine the SCL $T_{LOW}$ and $T_{HIGH}$ times:

$$T_{\mathrm{LOW}} = \frac{BAUDLOW + 5}{f_{\mathrm{GCLK}}}$$

$$T_{\mathrm{HIGH}} = \frac{BAUD + 5}{f_{\mathrm{GCLK}}}$$

**Note:** The I²C standard *Fm+* (Fast-mode plus) requires a nominal high to low SCL ratio of 1:2, and BAUD should be set accordingly. At a minimum, BAUD.BAUD and/or BAUD.BAUDLOW must be non-zero.


© 2026 Microchip Technology Inc. and its subsidiaries

**Startup Timing** The minimum time between SDA transition and SCL rising edge is 6 APB cycles when the DATA register is written in smart mode. If a greater startup time is required due to long rise times, the time between DATA write and IF clear must be controlled by software.
**Note:** When timing is controlled by user, the Smart Mode cannot be enabled.

**Related Links**
Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

**Host Clock Generation (High-Speed Mode)**
For I²C *Hs* transfers, there is no SCL synchronization. Instead, the SCL frequency is determined by the GCLK_SERCOMx_CORE frequency ($f_{GCLK}$) and the High-Speed Baud setting in the Baud register (BAUD.HSBAUD). When BAUD.HSBAUDLOW=0, the HSBAUD value will determine both SCL high and SCL low. In this case the following formula determines the SCL frequency.

$$f_{SCL} = \frac{f_{GCLK}}{2 + 2 \cdot HS\ BAUD}$$

When HSBAUDLOW is non-zero, the following formula determines the SCL frequency.

$$f_{SCL} = \frac{f_{GCLK}}{2 + HS\ BAUD + HSBAUDLOW}$$

**Note:** The I²C standard *Hs* (High-speed) requires a nominal high to low SCL ratio of 1:2, and HSBAUD should be set accordingly. At a minimum, BAUD.HSBAUD and/or BAUD.HSBAUDLOW must be non-zero.

### 33.6.2.4.2. Transmitting Address Packets

The I²C host starts a bus transaction by writing the I²C client address to ADDR.ADDR and the direction bit, as described in Principle of Operation. If the bus is busy, the I²C host will wait until the bus becomes idle before continuing the operation. When the bus is idle, the I²C host will issue a start condition on the bus. The I²C host will then transmit an address packet using the address written to ADDR.ADDR. After the address packet has been transmitted by the I²C host, one of four cases will arise according to arbitration and transfer direction.

**Case 1: Arbitration lost or bus error during address packet transmission**

If arbitration was lost during transmission of the address packet, the Host on Bus bit in the Interrupt Flag Status and Clear register (INTFLAG.MB) and the Arbitration Lost bit in the Status register (STATUS.ARBLOST) are both set. Serial data output to SDA is disabled, and the SCL is released, which disables clock stretching. In effect the I²C host is no longer allowed to execute any operation on the bus until the bus is idle again. A bus error will behave similarly to the Arbitration Lost condition. In this case, the MB Interrupt flag and Host Bus Error bit in the Status register (STATUS.BUSERR) are both set in addition to STATUS.ARBLOST.

The Host Received Not Acknowledge bit in the Status register (STATUS.RXNACK) will always contain the last successfully received acknowledge or not acknowledge indication.

In this case, software will typically inform the application code of the condition and then clear the Interrupt flag before exiting the interrupt routine. No other flags have to be cleared at this moment, because all flags will be cleared automatically the next time the ADDR.ADDR register is written.

**Case 2: Address packet transmit complete – No ACK received**

If there is no I²C client device responding to the address packet, then the INTFLAG.MB Interrupt flag and STATUS.RXNACK will be set. The clock hold is active at this point, preventing further activity on the bus.

The missing ACK response can indicate that the I²C client is busy with other tasks or sleeping. Therefore, it is not able to respond. In this event, the next step can be either issuing a Stop condition (recommended) or resending the address packet by a repeated Start condition. When using SMBus


© 2026 Microchip Technology Inc. and its subsidiaries

logic, the client must ACK the address. If there is no response, it means that the client is not available on the bus.

**Case 3: Address packet transmit complete – Write packet, Host on Bus set**

If the I²C host receives an acknowledge response from the I²C client, INTFLAG.MB will be set and STATUS.RXNACK will be cleared. The clock hold is active at this point, preventing further activity on the bus.

In this case, the software implementation becomes highly protocol dependent. Three possible actions can enable the I²C operation to continue:

- Initiate a data transmit operation by writing the data byte to be transmitted into DATA.DATA.
- Transmit a new address packet by writing ADDR.ADDR. A repeated Start condition will automatically be inserted before the address packet.
- Issue a Stop condition, consequently terminating the transaction.

**Case 4: Address packet transmit complete – Read packet, Client on Bus set**

If the I²C host receives an ACK from the I²C client, the I²C host proceeds to receive the next byte of data from the I²C client. When the first data byte is received, the Client on Bus bit in the Interrupt Flag register (INTFLAG.SB) will be set and STATUS.RXNACK will be cleared. The clock hold is active at this point, preventing further activity on the bus.

In this case, the software implementation becomes highly protocol dependent. Three possible actions can enable the I²C operation to continue:

- Let the I²C host continue to read data by acknowledging the data received. ACK can be sent by software, or automatically in Smart mode.
- Transmit a new address packet.
- Terminate the transaction by issuing a Stop condition.

**Note:** An ACK or NACK will be automatically transmitted if Smart mode is enabled. The Acknowledge Action bit in the Control B register (CTRLB.ACKACT) determines whether ACK or NACK should be sent.

#### 33.6.2.4.3. Transmitting Data Packets

When an address packet with direction Host Write (see Figure 33-3) was transmitted successfully , INTFLAG.MB will be set. The I²C host will start transmitting data via the I²C bus by writing to DATA.DATA, and monitor continuously for packet collisions.

If a collision is detected, the I²C host will lose arbitration and STATUS.ARBLOST will be set. If the transmit was successful, the I²C host will receive an ACK bit from the I²C client, and STATUS.RXNACK will be cleared. INTFLAG.MB will be set in both cases, regardless of arbitration outcome.

It is recommended to read STATUS.ARBLOST and handle the arbitration lost condition in the beginning of the I²C Host on Bus interrupt. This can be done as there is no difference between handling address and data packet arbitration.

STATUS.RXNACK must be checked for each data packet transmitted before the next data packet transmission can commence. The I²C host is not allowed to continue transmitting data packets if a NACK is received from the I²C client.

#### 33.6.2.4.4. Receiving Data Packets (SCLSM=0)

When INTFLAG.SB is set, the I²C host will already have received one data packet. The I²C host must respond by sending either an ACK or NACK. Sending a NACK may be unsuccessful when arbitration is lost during the transmission. In this case, a lost arbitration will prevent setting INTFLAG.SB. Instead, INTFLAG.MB will indicate a change in arbitration. Handling of lost arbitration is the same as for data bit transmission.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 33.6.2.4.5. Receiving Data Packets (SCLSM=1)

When INTFLAG.SB is set, the I²C host will already have received one data packet and transmitted an ACK or NACK, depending on CTRLB.ACKACT. At this point, CTRLB.ACKACT must be set to the correct value for the next ACK bit, and the transaction can continue by reading DATA and issuing a command if not in the Smart mode.

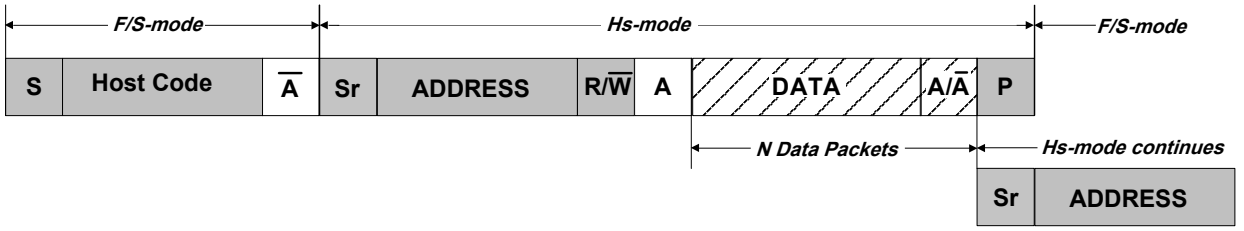
#### 33.6.2.4.6. High-Speed Mode

High-speed transfers are a multi-step process, see High Speed Transfer.

First, a host code (0b00001nnn, where 'nnn' is a unique host code) is transmitted in Full-speed mode, followed by a NACK since no client should acknowledge. Arbitration is performed only during the Full-speed Host Code phase. The host code is transmitted by writing the host code to the Address register (ADDR.ADDR) and writing the High-speed bit (ADDR.HS) to '0'.

After the host code and NACK have been transmitted, the host write interrupt will be asserted. In the meanwhile, the client address can be written to the ADDR.ADDR register together with ADDR.HS=1. Now in High-speed mode, the host will generate a repeated start, followed by the client address with RW-direction. The bus will remain in High-speed mode until a stop is generated. If a repeated start is desired, the ADDR.HS bit must again be written to '1', along with the new address ADDR.ADDR to be transmitted.

**Figure 33-8.** High Speed Transfer



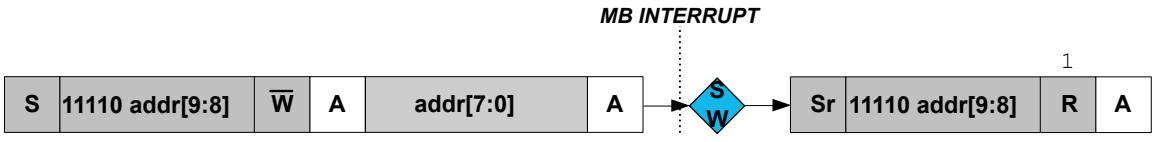

Transmitting in High-speed mode requires the I²C host to be configured in High-speed mode (CTRLA.SPEED=0x2) and the SCL Clock Stretch mode (CTRLA.SCLSM) bit set to '1'.

#### 33.6.2.4.7. 10-Bit Addressing

When 10-bit addressing is enabled by the Ten Bit Addressing Enable bit in the Address register (ADDR.TENBITEN=1) and the Address bit field ADDR.ADDR is written, the two address bytes will be transmitted, see 10-bit Address Transmission for a Read Transaction. The addressed client acknowledges the two address bytes, and the transaction continues. Regardless of whether the transaction is a read or write, the host must start by sending the 10-bit address with the direction bit (ADDR.ADDR[0]) being zero.

If the host receives a NACK after the first byte, the Write Interrupt flag will be raised and the STATUS.RXNACK bit will be set. If the first byte is acknowledged by one or more clients, then the host will proceed to transmit the second address byte and the host will first see the Write Interrupt flag after the second byte is transmitted. If the transaction direction is read-from-client, the 10-bit address transmission must be followed by a repeated start and the first 7 bits of the address with the read/write bit equal to '1'.

**Figure 33-9.** 10-bit Address Transmission for a Read Transaction




This implies the following procedure for a 10-bit read operation:


© 2026 Microchip Technology Inc. and its subsidiaries

1. Write the 10-bit address to ADDR.ADDR[10:1]. ADDR.TENBITEN must be '1', the direction bit (ADDR.ADDR[0]) must be '0' (can be written simultaneously with ADDR).
2. Once the Host on Bus interrupt is asserted, Write ADDR[7:0] register to '11110 `address[9:8]` 1'. ADDR.TENBITEN must be cleared (can be written simultaneously with ADDR).
3. Proceed to transmit data.

### 33.6.2.5. I²C Client Operation

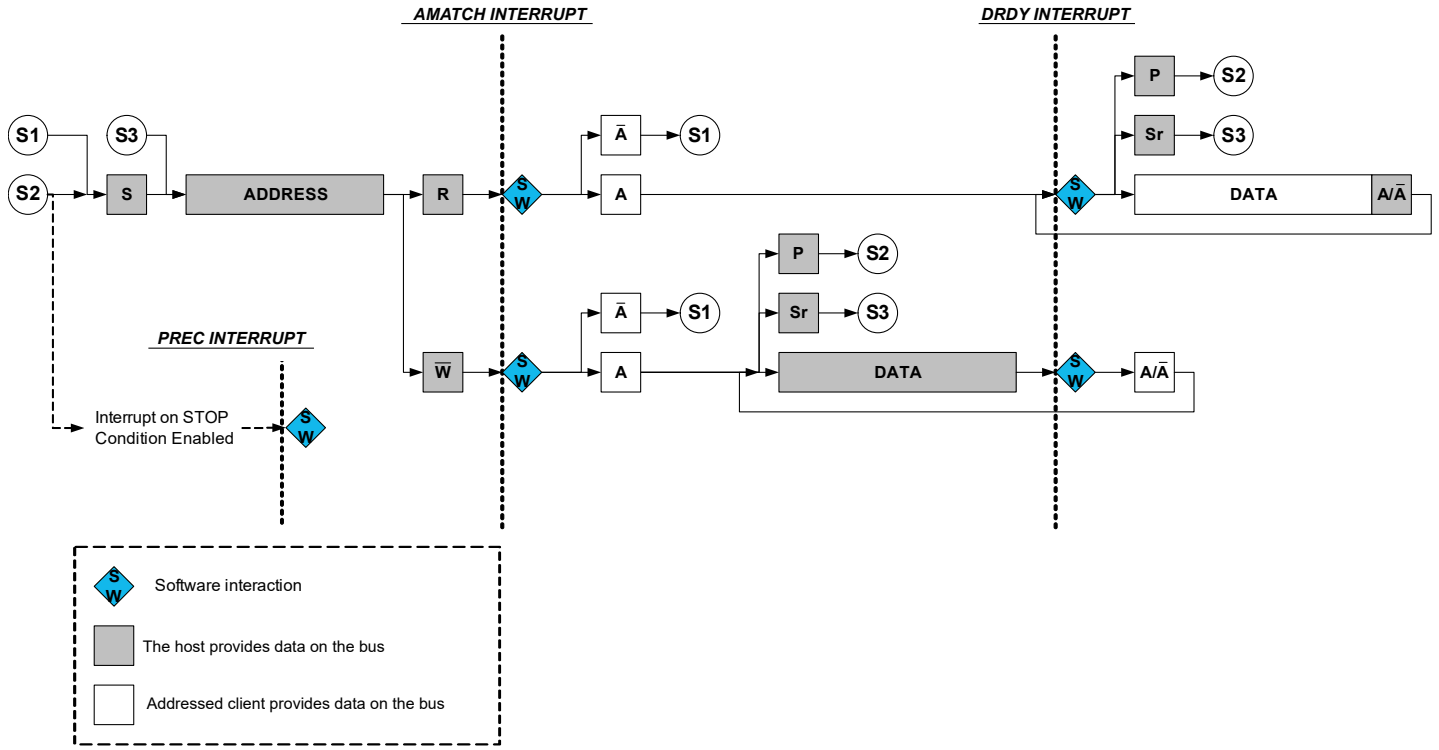
The I²C client is byte-oriented and interrupt-based. The number of interrupts generated is kept at a minimum by automatic handling of most events. The software driver complexity and code size are reduced by auto-triggering of operations, and a special smart mode, which can be enabled by the Smart Mode Enable bit in the Control B register (CTRLB.SMEN).

The I²C client has two interrupt strategies.

When SCL Stretch Mode bit (CTRLA.SCLSM) is '0', SCL is stretched before or after the acknowledge bit. In this mode, the I²C client operates according to the following figure. The circles labeled "S*n*" (S1, S2..) indicate the nodes the bus logic can jump to, based on software or hardware interaction.

This diagram is used as reference for the description of the I²C client operation throughout the document.

**Figure 33-10.** I²C Client Behavioral Diagram (SCLSM=0)

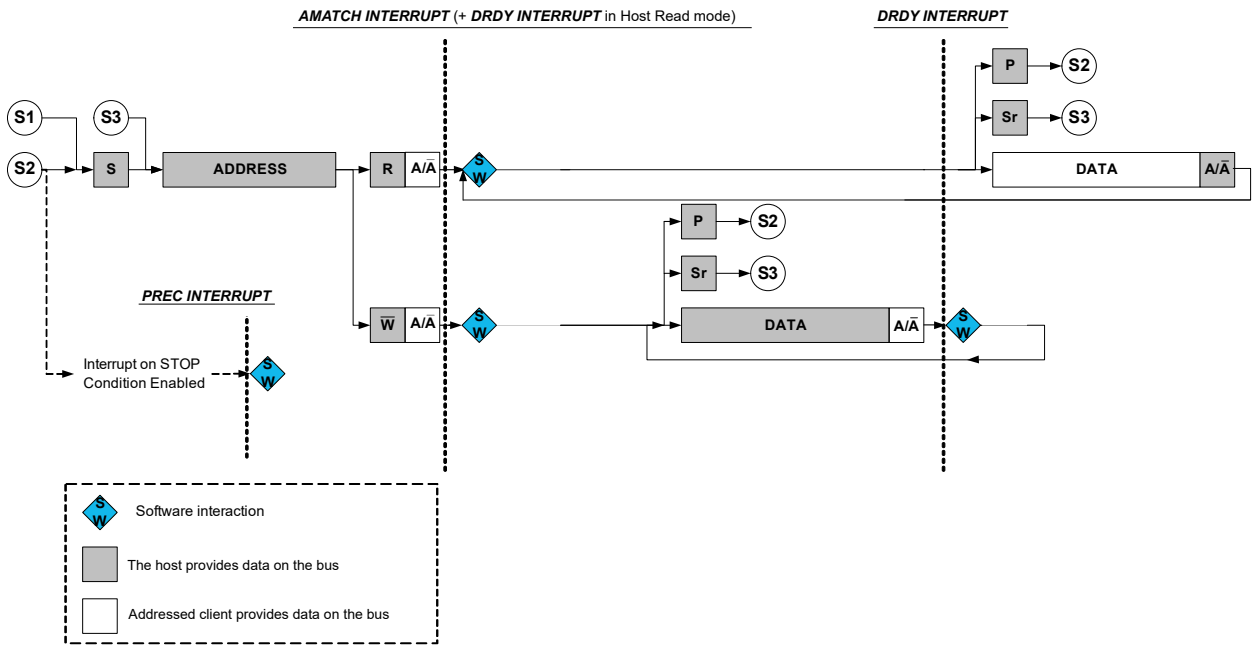



In the second strategy (CTRLA.SCLSM=1), interrupts only occur after the ACK bit is sent as shown in the following figure. This strategy can be used when it is not necessary to check DATA before acknowledging. For host reads, an address and data interrupt will be issued simultaneously after the address acknowledge. However, for host writes, the first data interrupt will be seen after the first data byte has been received by the client and the acknowledge bit has been sent to the host.

**Note:** For I²C High-speed mode (*Hs*), SCLSM=1 is required.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-11.** I²C Client Behavioral Diagram (SCLSM=1)




### 33.6.2.5.1. Receiving Address Packets (SCLSM=0)

When CTRLA.SCLSM=0, the I²C client stretches the SCL line according to Figure 33-10. When the I²C client is properly configured, it will wait for a Start condition.

When a Start condition is detected, the successive address packet will be received and checked by the address match logic. If the received address is not a match, the packet will be rejected, and the I²C client will wait for a new Start condition. If the received address is a match, the Address Match bit in the Interrupt Flag register (INTFLAG.AMATCH) will be set.

SCL will be stretched until the I²C client clears INTFLAG.AMATCH. As the I²C client holds the clock by forcing SCL low, the software has unlimited time to respond.

The direction of a transaction is determined by reading the Read/Write Direction bit in the Status register (STATUS.DIR). This bit will be updated only when a valid address packet is received.

If the Transmit Collision bit in the Status register (STATUS.COLL) is set, this indicates that the last packet addressed to the I²C client had a packet collision. A collision causes the SDA and SCL lines to be released without any notification to software. Therefore, the next AMATCH interrupt is the first indication of the previous packet's collision. Collisions are intended to follow the SMBus Address Resolution Protocol (ARP).

After the address packet has been received from the I²C host, one of two cases will arise based on transfer direction.

**Case 1: Address packet accepted – Read flag set**

The STATUS.DIR bit is '1', indicating an I²C host read operation. The SCL line is forced low, stretching the bus clock. If an ACK is sent, I²C client hardware will set the Data Ready bit in the Interrupt Flag register (INTFLAG.DRDY), indicating data are needed for transmit. If a NACK is sent, the I²C client will wait for a new Start condition and address match.

Typically, software will immediately acknowledge the address packet by sending an ACK/NACK bit. The I²C client Command bit field in the Control B register (CTRLB.CMD) can be written to '0x3' for both read and write operations as the command execution is dependent on the STATUS.DIR bit. Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.


© 2026 Microchip Technology Inc. and its subsidiaries

**Case 2: Address packet accepted – Write flag set**

The STATUS.DIR bit is cleared, indicating an I²C host write operation. The SCL line is forced low, stretching the bus clock. If an ACK is sent, the I²C client will wait for data to be received. Data, repeated start or stop can be received.

If a NACK is sent, the I²C client will wait for a new Start condition and address match. Typically, software will immediately acknowledge the address packet by sending an ACK/NACK. The I²C client command CTRLB.CMD = 3 can be used for both read and write operation as the command execution is dependent on STATUS.DIR.

Writing '1' to INTFLAG.AMATCH will also cause an ACK/NACK to be sent corresponding to the CTRLB.ACKACT bit.

#### 33.6.2.5.2. Receiving Address Packets (SCLSM=1)

When SCLSM=1, the I²C client will stretch the SCL line only after an ACK, see Client Behavioral Diagram (SCLSM=1). When the I²C client is properly configured, it will wait for a Start condition to be detected.

When a Start condition is detected, the successive address packet will be received and checked by the address match logic.

If the received address is not a match, the packet will be rejected and the I²C client will wait for a new Start condition.

If the address matches, the acknowledge action as configured by the Acknowledge Action bit Control B register (CTRLB.ACKACT) will be sent and the Address Match bit in the Interrupt Flag register (INTFLAG.AMATCH) is set. SCL will be stretched until the I²C client clears INTFLAG.AMATCH. As the I²C client holds the clock by forcing SCL low, the software is given unlimited time to respond to the address.

The direction of a transaction is determined by reading the Read/Write Direction bit in the Status register (STATUS.DIR). This bit will be updated only when a valid address packet is received.

If the Transmit Collision bit in the Status register (STATUS.COLL) is set, the last packet addressed to the I²C client had a packet collision. A collision causes the SDA and SCL lines to be released without any notification to software. The next AMATCH interrupt is, therefore, the first indication of the previous packet's collision. Collisions are intended to follow the SMBus Address Resolution Protocol (*ARP*).

After the address packet has been received from the I²C host, INTFLAG.AMATCH be set to '1' to clear it.

#### 33.6.2.5.3. Receiving and Transmitting Data Packets

After the I²C client has received an address packet, it will respond according to the direction either by waiting for the data packet to be received or by starting to send a data packet by writing to DATA.DATA. When a data packet is received or sent, INTFLAG.DRDY will be set. After receiving data, the I²C client will send an acknowledge according to CTRLB.ACKACT.

**Case 1: Data received**

INTFLAG.DRDY is set, and SCL is held low, pending for SW interaction.

**Case 2: Data sent**

When a byte transmission is successfully completed, the INTFLAG.DRDY Interrupt flag is set. If NACK is received, indicated by STATUS.RXNACK=1, the I²C client must expect a stop or a repeated start to be received. The I²C client must release the data line to allow the I²C host to generate a stop or repeated start. Upon detecting a Stop condition, the Stop Received bit in the Interrupt Flag register (INTFLAG.PREC) will be set and the I²C client will return to IDLE state.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 33.6.2.5.4. High-Speed Mode

When the I²C client is configured in High-speed mode (*Hs*, CTRLA.SPEED=0x2) and CTRLA.SCLSM=1, switching between Full-speed and High-speed modes is automatic. When the client recognizes a START followed by a host code transmission and a NACK, it automatically switches to High-speed mode and sets the High-speed status bit (STATUS.HS). The client will then remain in High-speed mode until a STOP is received.

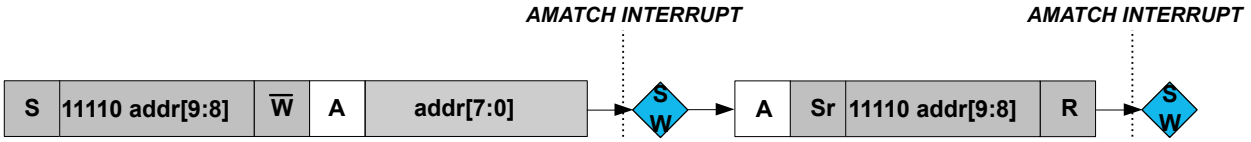
#### 33.6.2.5.5. 10-Bit Addressing

When 10-bit addressing is enabled (ADDR.TENBITEN=1), the two address bytes following a START will be checked against the 10-bit client address recognition. The first byte of the address will always be acknowledged, and the second byte will raise the address Interrupt flag, see 10-bit Addressing.

If the transaction is a write, then the 10-bit address will be followed by *N* data bytes.

If the operation is a read, the 10-bit address will be followed by a repeated START and reception of '11110 `ADDR[9:8]` 1', and the second address interrupt will be received with the DIR bit set. The client matches on the second address as it was addressed by the previous 10-bit address.

**Figure 33-12.** 10-bit Addressing




#### 33.6.2.5.6. PMBus Group Command

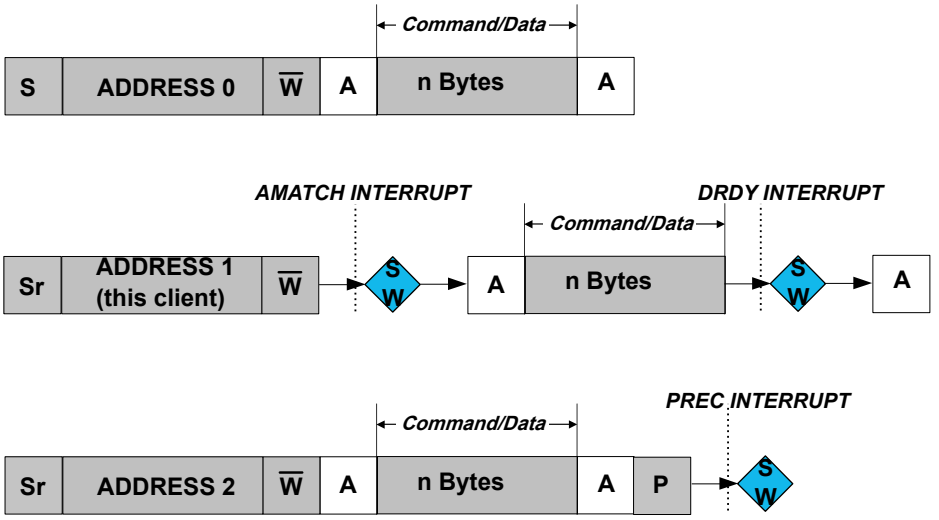
When the PMBus Group Command bit in the CTRLB register is set (CTRLB.GCMD=1) and 7-bit addressing is used, INTFLAG.PREC will be set if the client has been addressed since the last STOP condition. When CTRLB.GCMD=0, a STOP condition without address match will not be set INTFLAG.PREC.

The group command protocol is used to send commands to more than one device. The commands are sent in one continuous transmission with a single STOP condition at the end. When the STOP condition is detected by the clients addressed during the group command, they all begin executing the command they received.

The following figure shows an example where this client, bearing ADDRESS 1, is addressed after a repeated START condition. There can be multiple clients addressed before and after this client. Eventually, at the end of the group command, a single STOP is generated by the host. At this point a STOP interrupt is asserted.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-13.** PMBus Group Command Example




## 33.6.3. Additional Features

### 33.6.3.1. SMBus

The I²C includes three hardware SCL low time-outs which allow a time-out to occur for SMBus SCL low time-out, host extend time-out, and client extend time-out. This allows for SMBus functionality These time-outs are driven by the GCLK_SERCOM_SLOW clock. The GCLK_SERCOM_SLOW clock is used to accurately time the time-out and must be configured to use a 32.768 kHz oscillator. The I²C interface also allows for a SMBus compatible SDA hold time.

- $T_{TIMEOUT}$: SCL low time of 25..35ms – Measured for a single SCL low period. It is enabled by CTRLA.LOWTOUTEN.
- $T_{LOW:SEXT}$: Cumulative clock low extend time of 25 ms – Measured as the cumulative SCL low extend time by a client device in a single message from the initial START to the STOP. It is enabled by CTRLA.SEXTTOEN.
- $T_{LOW:MEXT}$: Cumulative clock low extend time of 10 ms – Measured as the cumulative SCL low extend time by the host device within a single byte from START-to-ACK, ACK-to-ACK, or ACK-to-STOP. It is enabled by CTRLA.MEXTTOEN.

### 33.6.3.2. Smart Mode

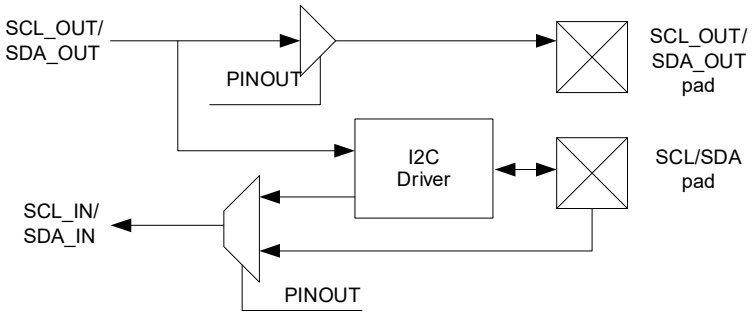
The I²C interface has a Smart mode that simplifies application code and minimizes the user interaction needed to adhere to the I²C protocol. The Smart mode accomplishes this by automatically issuing an ACK or NACK (based on the content of CTRLB.ACKACT) as soon as DATA.DATA is read.

### 33.6.3.3. 4-Wire Mode

Writing a '1' to the Pin Usage bit in the Control A register (CTRLA.PINOUT) will enable 4-Wire mode operation. In this mode, the internal I²C tri-state drivers are bypassed, and an external I²C compliant tri-state driver is needed when connecting to an I²C bus.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 33-14.** I²C Pad Interface




#### 33.6.3.4. Quick Command

Setting the Quick Command Enable bit in the Control B register (CTRLB.QCEN) enables quick command. When quick command is enabled, the corresponding Interrupt flag (INTFLAG.SB or INTFLAG.MB) is set immediately after the client acknowledges the address. At this point, the software can either issue a Stop command or a repeated start by writing CTRLB.CMD or ADDR.ADDR.

### 33.6.4. DMA, Interrupts and Events

Each interrupt source has its own interrupt flag. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the interrupt condition is meet. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request is active until the interrupt flag is cleared, the interrupt is disabled or the I²C is reset. See the INTFLAG (Client) or INTFLAG (Host) register for details on how to clear interrupt flags.

**Table 33-1.** Module Request for SERCOM I²C Client

| Condition | Request | | |
|---|---|---|---|
| | DMA | Interrupt | Event |
| Data needed for transmit (TX) (Client transmit mode) | Yes (request cleared when data is written) | | NA |
| Data received (RX) (Client receive mode) | Yes (request cleared when data is read) | | |
| Data Ready (DRDY) | | Yes | |
| Address Match (AMATCH) | | Yes | |
| Stop received (PREC) | | Yes | |
| Error (ERROR) | | Yes | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 33-2.** Module Request for SERCOM I²C Host

| Condition | Request | | |
|---|---|---|---|
| | DMA | Interrupt | Event |
| Data needed for transmit (TX) (Host transmit mode) | Yes (request cleared when data is written) | | NA |
| Data needed for transmit (RX) (Host transmit mode) | Yes (request cleared when data is read) | | |
| Host on Bus (MB) | | Yes | |
| Stop received (SB) | | Yes | |
| Error (ERROR) | | Yes | |

#### 33.6.4.1. DMA Operation

Smart mode must be enabled for DMA operation in the Control B register by writing CTRLB.SMEN=1.

##### 33.6.4.1.1. Client DMA

When using the I²C client with DMA, an address match will cause the address Interrupt flag (INTFLAG.ADDRMATCH) to be raised. After the interrupt has been serviced, data transfer will be performed through DMA.

The I²C client generates the following requests:

- Write data received (RX): The request is set when host write data is received. The request is cleared when DATA is read.
- Read data needed for transmit (TX): The request is set when data is needed for a host read operation. The request is cleared when DATA is written.

##### 33.6.4.1.2. Host DMA

When using the I²C host with DMA, the ADDR register must be written with the desired address (ADDR.ADDR), transaction length (ADDR.LEN), and transaction length enable (ADDR.LENEN). When ADDR.LENEN is written to 1 along with ADDR.ADDR, ADDR.LEN determines the number of data bytes in the transaction from 0 to 255. DMA is then used to transfer ADDR.LEN bytes followed by an automatically generated NACK (for host reads) and a STOP.

If a NACK is received by the client for a host write transaction before ADDR.LEN bytes, a STOP will be automatically generated and the length error (STATUS.LENERR) will be raised along with the INTFLAG.ERROR interrupt.

The I²C host generates the following requests:

- Read data received (RX): The request is set when host read data is received. The request is cleared when DATA is read.
- Write data needed for transmit (TX): The request is set when data is needed for a host write operation. The request is cleared when DATA is written.

#### 33.6.4.2. Interrupts

The I²C Client has the following interrupt sources. These are asynchronous interrupts. They can wake-up the device from any Sleep mode:

- Error (ERROR)
- Data Ready (DRDY)
- Address Match (AMATCH)
- Stop Received (PREC)

The I²C Host has the following interrupt sources. These are asynchronous interrupts. They can wake-up the device from any Sleep mode:


© 2026 Microchip Technology Inc. and its subsidiaries

- Error (ERROR)
- Client on Bus (SB)
- Host on Bus (MB)

Each interrupt source has its own Interrupt flag. The Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) will be set when the Interrupt condition is meet. Each interrupt can be individually enabled by writing '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR).

The status of enabled interrupts can be read from either INTENSET or INTENCLR. An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request active until the Interrupt flag is cleared, the interrupt is disabled or the $I^2C$ is reset. See the INTFLAG register for details on how to clear Interrupt flags.

The value of INTFLAG indicates which interrupt is executed. Note that interrupts must be globally enabled for interrupt requests. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

#### 33.6.4.3. Events

Not applicable.

### 33.6.5. Sleep Mode Operation

#### $I^2C$ Host Operation

The generic clock (GCLK_SERCOMx_CORE) will continue to run in idle sleep mode. If the Run In Standby bit in the Control A register (CTRLA.RUNSTDBY) is '1', the GLK_SERCOMx_CORE will also run in Standby Sleep mode. Any interrupt can wake-up the device.

If CTRLA.RUNSTDBY=0, the GLK_SERCOMx_CORE will be disabled after any ongoing transaction is finished. Any interrupt can wake-up the device.

#### $I^2C$ Client Operation

Writing CTRLA.RUNSTDBY=1 will allow the Address Match interrupt to wake-up the device.

When CTRLA.RUNSTDBY=0, all receptions will be dropped.

### 33.6.6. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in the CTRLA register (CTRLA.SWRST)
- Enable bit in the CTRLA register (CTRLA.ENABLE)
- Command bits in CTRLB register (CTRLB.CMD)
- Write to Bus State bits in the Status register (STATUS.BUSSTATE)
- Address bits in the Address register (ADDR.ADDR) when in host operation.

The following registers are synchronized when written:

- Data (DATA) when in host operation

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 33.7. Register Summary - I2C Client

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | SEXTTOEN | | SDAHOLD[1:0] | | | | | PINOUT |
| | | 31:24 | | LOWTOUTEN | | | SCLSM | | SPEED[1:0] | |
| 0x04 | CTRLB | 7:0 | | | | | | | | |
| | | 15:8 | AMODE[1:0] | | | | | AACKEN | GCMD | SMEN |
| | | 23:16 | | | | | | ACKACT | CMD[1:0] | |
| | | 31:24 | | | | | | | | |
| 0x08 ... 0x13 | Reserved | | | | | | | | | |
| 0x14 | INTENCLR | 7:0 | ERROR | | | | | DRDY | AMATCH | PREC |
| 0x15 | Reserved | | | | | | | | | |
| 0x16 | INTENSET | 7:0 | ERROR | | | | | DRDY | AMATCH | PREC |
| 0x17 | Reserved | | | | | | | | | |
| 0x18 | INTFLAG | 7:0 | ERROR | | | | | DRDY | AMATCH | PREC |
| 0x19 | Reserved | | | | | | | | | |
| 0x1A | STATUS | 7:0 | CLKHOLD | LOWTOUT | | SR | DIR | RXNACK | COLL | BUSERR |
| | | 15:8 | | | | | | HS | SEXTTOUT | |
| 0x1C | SYNCBUSY | 7:0 | | | | | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 ... 0x23 | Reserved | | | | | | | | | |
| 0x24 | ADDR | 7:0 | ADDR[6:0] | | | | | | | GENCEN |
| | | 15:8 | TENBITEN | | | | | ADDR[9:7] | | |
| | | 23:16 | ADDRMASK[6:0] | | | | | | | |
| | | 31:24 | | | | | | ADDRMASK[9:7] | | |
| 0x28 | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |

## 33.8. Register Description - I²C Client

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | LOWTOUTEN | | | SCLSM | | SPEED[1:0] | |
| Access | | R/W | | | R/W | | R/W | R/W |
| Reset | | 0 | | | 0 | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SEXTTOEN | | SDAHOLD[1:0] | | | | | PINOUT |
| Access | R/W | | R/W | R/W | | | | R/W |
| Reset | 0 | | 0 | 0 | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – LOWTOUTEN** SCL Low Time-Out
This bit enables the SCL low time-out. If SCL is held low for 25ms-35ms, the client will release its clock hold, if enabled, and reset the internal state machine. Any interrupt flags set at the time of time-out will remain set.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Time-out disabled. |
| 1 | Time-out enabled. |

**Bit 27 – SCLSM** SCL Clock Stretch Mode
This bit controls when SCL will be stretched for software interaction.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | SCL stretch according to Figure 33-10 |
| 1 | SCL stretch only after ACK bit according to Figure 33-11 |

**Bits 25:24 – SPEED[1:0]** Transfer Speed
These bits define bus speed.
These bits are not synchronized.

| Value | Description |
|---|---|
| 0x0 | Standard-mode (Sm) up to 100 kHz and Fast-mode (Fm) up to 400 kHz |
| 0x1 | Fast-mode Plus (Fm+) up to 1 MHz |
| 0x2 | High-speed mode (Hs-mode) up to 3.4 MHz |
| 0x3 | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 23 – SEXTTOEN** Client SCL Low Extend Time-Out

This bit enables the client SCL low extend time-out. If SCL is cumulatively held low for greater than 25ms from the initial START to a STOP, the client will release its clock hold if enabled and reset the internal state machine. Any interrupt flags set at the time of time-out will remain set. If the address was recognized, PREC will be set when a STOP is received.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Time-out disabled |
| 1 | Time-out enabled |

**Bits 21:20 – SDAHOLD[1:0]** SDA Hold Time

These bits define the SDA hold time with respect to the negative edge of SCL.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIS | Disabled |
| 0x1 | 75 | 50-100ns hold time |
| 0x2 | 450 | 300-600ns hold time |
| 0x3 | 600 | 400-800ns hold time |

**Bit 16 – PINOUT** Pin Usage

This bit sets the pin usage to either two- or four-wire operation:
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | 4-wire operation disabled |
| 1 | 4-wire operation enabled |

**Bit 7 – RUNSTDBY** Run in Standby

This bit defines the functionality in standby sleep mode.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Disabled – All reception is dropped. |
| 1 | Wake on address match, if enabled. |

**Bits 4:2 – MODE[2:0]** Operating Mode

These bits must be written to 0x04 to select the $I^2C$ client serial communication interface of the SERCOM.
These bits are not synchronized.

**Bit 1 – ENABLE** Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the Enable Synchronization Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled or being disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.2. Control B

**Name:** CTRLB
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected (unless indicated below)

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | ACKACT | CMD[1:0] | |
| Access | | | | | | R/W | W | W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | AMODE[1:0] | | | | | AACKEN | GCMD | SMEN |
| Access | R/W | R/W | | | | R/W | R/W | R/W |
| Reset | 0 | 0 | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 18 – ACKACT** Acknowledge Action

This bit defines the client's acknowledge behavior after an address or data byte is received from the host. The acknowledge action is executed when a command is written to the CMD bits. If smart mode is enabled (CTRLB.SMEN=1), the acknowledge action is performed when the DATA register is read.
ACKACT shall not be updated more than once between each peripheral interrupts request.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | Send ACK |
| 1 | Send NACK |

**Bits 17:16 – CMD[1:0]** Command

This bit field triggers the client operation as the below. The CMD bits are strobe bits, and always read as zero. The operation is dependent on the client interrupt flags, INTFLAG.DRDY and INTFLAG.AMATCH, in addition to STATUS.DIR.
All interrupt flags (INTFLAG.DRDY, INTFLAG.AMATCH and INTFLAG.PREC) are automatically cleared when a command is given.
This bit is not enable-protected.

**Table 33-3.** Command Description

| CMD[1:0] | DIR | Action |
|---|---|---|
| 0x0 | X | (No action) |
| 0x1 | X | (Reserved) |
| 0x2 | Used to complete a transaction in response to a data interrupt (DRDY) | |
| | 0 (Host write) | Execute acknowledge action succeeded by waiting for any start (S/Sr) condition |
| | 1 (Host read) | Wait for any start (S/Sr) condition |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 33-3.** Command Description (continued)

| CMD[1:0] | DIR | Action |
|---|---|---|
| 0x3 | Used in response to an address interrupt (AMATCH) | |
| | 0 (Host write) | Execute acknowledge action succeeded by reception of next byte |
| | 1 (Host read) | Execute acknowledge action succeeded by client data interrupt |
| | Used in response to a data interrupt (DRDY) | |
| | 0 (Host write) | Execute acknowledge action succeeded by reception of next byte |
| | 1 (Host read) | Execute a byte read operation followed by ACK/NACK reception |

**Bits 15:14 – AMODE[1:0]** Address Mode

These bits set the addressing mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | MASK | The client responds to the address written in ADDR.ADDR masked by the value in ADDR.ADDRMASK. See *SERCOM – Serial Communication Interface* for additional information. |
| 0x1 | 2_ADDRS | The client responds to the two unique addresses in ADDR.ADDR and ADDR.ADDRMASK. |
| 0x2 | RANGE | The client responds to the range of addresses between and including ADDR.ADDR and ADDR.ADDRMASK. ADDR.ADDR is the upper limit. |
| 0x3 | - | Reserved. |

**Bit 10 – AACKEN** Automatic Acknowledge Enable

This bit enables the address to be automatically acknowledged if there is an address match.

| Value | Description |
|---|---|
| 0 | Automatic acknowledge is disabled. |
| 1 | Automatic acknowledge is enabled. |

**Bit 9 – GCMD** PMBus Group Command

This bit enables PMBus group command support. When enabled, the Stop Received interrupt flag (INTFLAG.PREC) will be set when a STOP condition is detected if the client has been addressed since the last STOP condition on the bus.

| Value | Description |
|---|---|
| 0 | Group command is disabled. |
| 1 | Group command is enabled. |

**Bit 8 – SMEN** Smart Mode Enable

When smart mode is enabled, data is acknowledged automatically when DATA.DATA is read.

| Value | Description |
|---|---|
| 0 | Smart mode is disabled. |
| 1 | Smart mode is enabled. |

**Related Links**

SERCOM – Serial Communication Interface


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.3. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | DRDY | AMATCH | PREC |
| Access | R/W | | | | | R/W | R/W | R/W |
| Reset | 0 | | | | | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 2 – DRDY** Data Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Ready bit, which disables the Data Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Data Ready interrupt is disabled. |
| 1 | The Data Ready interrupt is enabled. |

**Bit 1 – AMATCH** Address Match Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Address Match Interrupt Enable bit, which disables the Address Match interrupt.

| Value | Description |
|---|---|
| 0 | The Address Match interrupt is disabled. |
| 1 | The Address Match interrupt is enabled. |

**Bit 0 – PREC** Stop Received Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Stop Received Interrupt Enable bit, which disables the Stop Received interrupt.

| Value | Description |
|---|---|
| 0 | The Stop Received interrupt is disabled. |
| 1 | The Stop Received interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.4. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x16
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | DRDY | AMATCH | PREC |
| Access | R/W | | | | | R/W | R/W | R/W |
| Reset | 0 | | | | | 0 | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 2 – DRDY** Data Ready Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Data Ready bit, which enables the Data Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Data Ready interrupt is disabled. |
| 1 | The Data Ready interrupt is enabled. |

**Bit 1 – AMATCH** Address Match Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Address Match Interrupt Enable bit, which enables the Address Match interrupt.

| Value | Description |
|---|---|
| 0 | The Address Match interrupt is disabled. |
| 1 | The Address Match interrupt is enabled. |

**Bit 0 – PREC** Stop Received Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Stop Received Interrupt Enable bit, which enables the Stop Received interrupt.

| Value | Description |
|---|---|
| 0 | The Stop Received interrupt is disabled. |
| 1 | The Stop Received interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.5. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x18
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | DRDY | AMATCH | PREC |
| Access | R/W | | | | | R/W | R/W | R/W |
| Reset | 0 | | | | | 0 | 0 | 0 |

**Bit 7 – ERROR** Error
This bit is set when any error is detected. Errors that will set this flag have corresponding status flags in the STATUS register. The corresponding bits in STATUS are SEXTTOUT, LOWTOUT, COLL, and BUSERR.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 2 – DRDY** Data Ready
This flag is set when a $I^2C$ client byte transmission or reception is successfully completed.
The flag is cleared by hardware when either:

- Writing to the DATA register.
- Reading the DATA register with smart mode enabled.
- Writing a valid command to the CMD register.

Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Data Ready interrupt flag.

**Bit 1 – AMATCH** Address Match
This flag is set when the $I^2C$ client address match logic detects that a valid address has been received.
The flag is cleared by hardware when CTRL.CMD is written.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Address Match interrupt flag. When cleared, an ACK/NACK will be sent according to CTRLB.ACKACT.

**Bit 0 – PREC** Stop Received
This flag is set when a stop condition is detected for a transaction being processed. A stop condition detected between a bus host and another client will not set this flag, unless the PMBus Group Command is enabled in the Control B register (CTRLB.GCMD=1).
This flag is cleared by hardware after a command is issued on the next address match.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Stop Received interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.6. Status

**Name:** STATUS
**Offset:** 0x1A
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | HS | SEXTTOUT | |
| Access | | | | | | R/W | R/W | |
| Reset | | | | | | 0 | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CLKHOLD | LOWTOUT | | SR | DIR | RXNACK | COLL | BUSERR |
| Access | R | R/W | | R | R | R | R/W | R/W |
| Reset | 0 | 0 | | 0 | 0 | 0 | 0 | 0 |

**Bit 10 – HS** High-speed
This bit is set if the client detects a START followed by a Host Code transmission.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status. However, this flag is automatically cleared when a STOP is received.

**Bit 9 – SEXTTOUT** Client SCL Low Extend Time-Out
This bit is set if a client SCL low extend time-out occurs.
This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

| Value | Description |
|---|---|
| 0 | No SCL low extend time-out has occurred. |
| 1 | SCL low extend time-out has occurred. |

**Bit 7 – CLKHOLD** Clock Hold
The client Clock Hold bit (STATUS.CLKHOLD) is set when the client is holding the SCL line low, stretching the $I^2C$ clock. Software should consider this bit a read-only status flag that is set when INTFLAG.DRDY or INTFLAG.AMATCH is set.
This bit is automatically cleared when the corresponding interrupt is also cleared.

**Bit 6 – LOWTOUT** SCL Low Time-out
This bit is set if an SCL low time-out occurs.
This bit is cleared automatically if responding to a new start condition with ACK or NACK (write 3 to CTRLB.CMD) or when INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

| Value | Description |
|---|---|
| 0 | No SCL low time-out has occurred. |
| 1 | SCL low time-out has occurred. |

**Bit 4 – SR** Repeated Start
When INTFLAG.AMATCH is raised due to an address match, SR indicates a repeated start or start condition.


© 2026 Microchip Technology Inc. and its subsidiaries

This flag is only valid while the INTFLAG.AMATCH flag is one.

| Value | Description |
|---|---|
| 0 | Start condition on last address match |
| 1 | Repeated start condition on last address match |

**Bit 3 – DIR** Read / Write Direction

The Read/Write Direction (STATUS.DIR) bit stores the direction of the last address packet received from a host.

| Value | Description |
|---|---|
| 0 | Host write operation is in progress. |
| 1 | Host read operation is in progress. |

**Bit 2 – RXNACK** Received Not Acknowledge

This bit indicates whether the last data packet sent was acknowledged or not.

| Value | Description |
|---|---|
| 0 | Host responded with ACK. |
| 1 | Host responded with NACK. |

**Bit 1 – COLL** Transmit Collision

If set, the $I^2C$ client was not able to transmit a high data or NACK bit, the $I^2C$ client will immediately release the SDA and SCL lines and wait for the next packet addressed to it.
This flag is intended for the SMBus address resolution protocol (ARP). A detected collision in non-ARP situations indicates that there has been a protocol violation, and should be treated as a bus error.
Note that this status will not trigger any interrupt, and should be checked by software to verify that the data were sent correctly. This bit is cleared automatically if responding to an address match with an ACK or a NACK (writing 0x3 to CTRLB.CMD), or INTFLAG.AMATCH is cleared.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the status.

| Value | Description |
|---|---|
| 0 | No collision detected on last data byte sent. |
| 1 | Collision detected on last data byte sent. |

**Bit 0 – BUSERR** Bus Error

The Bus Error bit (STATUS.BUSERR) indicates that an illegal bus condition has occurred on the bus, regardless of bus ownership. An illegal bus condition is detected if a protocol violating start, repeated start or stop is detected on the $I^2C$ bus lines. A start condition directly followed by a stop condition is one example of a protocol violation. If a time-out occurs during a frame, this is also considered a protocol violation, and will set STATUS.BUSERR.
This bit is cleared automatically if responding to an address match with an ACK or a NACK (writing 0x3 to CTRLB.CMD) or INTFLAG.AMATCH is cleared.
Writing a '1' to this bit will clear the status.
Writing a '0' to this bit has no effect.

| Value | Description |
|---|---|
| 0 | No bus error detected. |
| 1 | Bus error detected. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.7. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x1C
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** SERCOM Enable Synchronization Busy
Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. When written, the SYNCBUSY.ENABLE bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | Enable synchronization is not busy. |
| 1 | Enable synchronization is busy. |

**Bit 0 – SWRST** Software Reset Synchronization Busy
Resetting the SERCOM (CTRLA.SWRST) requires synchronization. When written, the SYNCBUSY.SWRST bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | SWRST synchronization is not busy. |
| 1 | SWRST synchronization is busy. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.8. Address

**Name:** ADDR
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | ADDRMASK[9:7] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ADDRMASK[6:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TENBITEN | | | | | ADDR[9:7] | | |
| Access | R/W | | | | | R/W | R/W | R/W |
| Reset | 0 | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[6:0] | | | | | | | GENCEN |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 26:17 – ADDRMASK[9:0]** Address Mask
These bits act as a second address match register, an address mask register or the lower limit of an address range, depending on the CTRLB.AMODE setting.

**Bit 15 – TENBITEN** Ten Bit Addressing Enable

| Value | Description |
|---|---|
| 0 | 10-bit address recognition disabled. |
| 1 | 10-bit address recognition enabled. |

**Bits 10:1 – ADDR[9:0]** Address
These bits contain the $I^2C$ client address used by the client address match logic to determine if a host has addressed the client.
When using 7-bit addressing, the client address is represented by ADDR[6:0].
When using 10-bit addressing (ADDR.TENBITEN=1), the client address is represented by ADDR[9:0]
When the address match logic detects a match, INTFLAG.AMATCH is set and STATUS.DIR is updated to indicate whether it is a read or a write transaction.

**Bit 0 – GENCEN** General Call Address Enable
A general call address is an address consisting of all-zeroes, including the direction bit (host write).

| Value | Description |
|---|---|
| 0 | General call address recognition disabled. |
| 1 | General call address recognition enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.8.9. Data

**Name:** DATA
**Offset:** 0x28
**Reset:** 0x0000
**Property:** Write-Synchronized, Read-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – DATA[7:0]** Data

The Client data register I/O location (DATA.DATA) provides access to the Host transmit and receive data buffers. Reading valid data or writing data to be transmitted can be successfully done only when SCL is held low by the Client (STATUS.CLKHOLD is set). An exception occurs when reading the last data byte after the stop condition has been received.
Accessing DATA.DATA auto-triggers $I^2C$ bus operations. The operation performed depends on the state of CTRLB.ACKACT, CTRLB.SMEN and the type of access (read/write).
Writing or reading DATA.DATA when not in smart mode does not require synchronization.


© 2026 Microchip Technology Inc. and its subsidiaries

## 33.9. Register Summary - I2C Host

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | SEXTTOEN | MEXTTOEN | SDAHOLD[1:0] | | | | | PINOUT |
| | | 31:24 | | LOWTOUTEN | INACTOUT[1:0] | | SCLSM | | SPEED[1:0] | |
| 0x04 | CTRLB | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | QCEN | SMEN |
| | | 23:16 | | | | | | ACKACT | CMD[1:0] | |
| | | 31:24 | | | | | | | | |
| 0x08 ... 0x0B | Reserved | | | | | | | | | |
| 0x0C | BAUD | 7:0 | BAUD[7:0] | | | | | | | |
| | | 15:8 | BAUDLOW[7:0] | | | | | | | |
| | | 23:16 | HSBAUD[7:0] | | | | | | | |
| | | 31:24 | HSBAUDLOW[7:0] | | | | | | | |
| 0x10 ... 0x13 | Reserved | | | | | | | | | |
| 0x14 | INTENCLR | 7:0 | ERROR | | | | | | SB | MB |
| 0x15 | Reserved | | | | | | | | | |
| 0x16 | INTENSET | 7:0 | ERROR | | | | | | SB | MB |
| 0x17 | Reserved | | | | | | | | | |
| 0x18 | INTFLAG | 7:0 | ERROR | | | | | | SB | MB |
| 0x19 | Reserved | | | | | | | | | |
| 0x1A | STATUS | 7:0 | CLKHOLD | LOWTOUT | BUSSTATE[1:0] | | | RXNACK | ARBLOST | BUSERR |
| | | 15:8 | | | | | | LENERR | SEXTTOUT | MEXTTOUT |
| 0x1C | SYNCBUSY | 7:0 | | | | | | SYSOP | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 ... 0x23 | Reserved | | | | | | | | | |
| 0x24 | ADDR | 7:0 | ADDR[7:0] | | | | | | | |
| | | 15:8 | TENBITEN | HS | LENEN | | | ADDR[10:8] | | |
| | | 23:16 | LEN[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x28 | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| 0x2A ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DBGCTRL | 7:0 | | | | | | | | DBGSTOP |

## 33.10. Register Description - I²C Host

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | LOWTOUTEN | INACTOUT[1:0] | | SCLSM | | SPEED[1:0] | |
| Access | | R/W | R/W | R/W | R/W | | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SEXTTOEN | MEXTTOEN | SDAHOLD[1:0] | | | | | PINOUT |
| Access | R/W | R/W | R/W | R/W | | | | R/W |
| Reset | 0 | 0 | 0 | 0 | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RUNSTDBY | | | MODE[2:0] | | | ENABLE | SWRST |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – LOWTOUTEN** SCL Low Time-Out

This bit enables the SCL low time-out. If SCL is held low for 25ms-35ms, the host will release its clock hold, if enabled, and complete the current transaction. A stop condition will automatically be transmitted.
INTFLAG.SB or INTFLAG.MB will be set as normal, but the clock hold will be released. The STATUS.LOWTOUT and STATUS.BUSERR status bits will be set.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Time-out disabled. |
| 1 | Time-out enabled. |

**Bits 29:28 – INACTOUT[1:0]** Inactive Time-Out

If the inactive bus time-out is enabled and the bus is inactive for longer than the time-out setting, the bus state logic will be set to idle. An inactive bus arise when either an $I^2C$ host or client is holding the SCL low.
Enabling this option is necessary for SMBus compatibility, but can also be used in a non-SMBus set-up.
Calculated time-out periods are based on a 100kHz baud rate.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIS | Disabled |
| 0x1 | 55US | 5-6 SCL cycle time-out (50-60µs) |
| 0x2 | 105US | 10-11 SCL cycle time-out (100-110µs) |
| 0x3 | 205US | 20-21 SCL cycle time-out (200-210µs) |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 27 – SCLSM** SCL Clock Stretch Mode

This bit controls when SCL will be stretched for software interaction.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | SCL stretch according to Figure 33-5. |
| 1 | SCL stretch only after ACK bit, Figure 33-6. |

**Bits 25:24 – SPEED[1:0]** Transfer Speed

These bits define bus speed.
These bits are not synchronized.

| Value | Description |
|---|---|
| 0x0 | Standard-mode (Sm) up to 100 kHz and Fast-mode (Fm) up to 400 kHz |
| 0x1 | Fast-mode Plus (Fm+) up to 1 MHz |
| 0x2 | High-speed mode (Hs-mode) up to 3.4 MHz |
| 0x3 | Reserved |

**Bit 23 – SEXTTOEN** Client SCL Low Extend Time-Out

This bit enables the client SCL low extend time-out. If SCL is cumulatively held low for greater than 25ms from the initial START to a STOP, the host will release its clock hold if enabled, and complete the current transaction. A STOP will automatically be transmitted.
SB or MB will be set as normal, but CLKHOLD will be release. The MEXTTOUT and BUSERR status bits will be set.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Time-out disabled |
| 1 | Time-out enabled |

**Bit 22 – MEXTTOEN** Host SCL Low Extend Time-Out

This bit enables the host SCL low extend time-out. If SCL is cumulatively held low for greater than 10ms from START-to-ACK, ACK-to-ACK, or ACK-to-STOP the host will release its clock hold if enabled, and complete the current transaction. A STOP will automatically be transmitted.
SB or MB will be set as normal, but CLKHOLD will be released. The MEXTTOUT and BUSERR status bits will be set.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | Time-out disabled |
| 1 | Time-out enabled |

**Bits 21:20 – SDAHOLD[1:0]** SDA Hold Time

These bits define the SDA hold time with respect to the negative edge of SCL.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIS | Disabled |
| 0x1 | 75NS | 50-100ns hold time |
| 0x2 | 450NS | 300-600ns hold time |
| 0x3 | 600NS | 400-800ns hold time |

**Bit 16 – PINOUT** Pin Usage

This bit set the pin usage to either two- or four-wire operation:
This bit is not synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | 4-wire operation disabled. |
| 1 | 4-wire operation enabled. |

**Bit 7 – RUNSTDBY** Run in Standby

This bit defines the functionality in standby sleep mode.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | GCLK_SERCOMx_CORE is disabled and the $I^2C$ host will not operate in standby sleep mode. |
| 1 | GCLK_SERCOMx_CORE is enabled in all sleep modes. |

**Bits 4:2 – MODE[2:0]** Operating Mode

These bits must be written to 0x5 to select the $I^2C$ host serial communication interface of the SERCOM.
These bits are not synchronized.

**Bit 1 – ENABLE** Enable

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the Synchronization Enable Busy bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled or being disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset

Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the SERCOM, except DBGCTRL, to their initial state, and the SERCOM will be disabled.
Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.2. Control B

**Name:** CTRLB
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | ACKACT | CMD[1:0] | |
| Access | | | | | | R/W | W | W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | QCEN | SMEN |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 18 – ACKACT** Acknowledge Action

This bit defines the $I^2C$ host's acknowledge behavior after a data byte is received from the $I^2C$ client. The acknowledge action is executed when a command is written to CTRLB.CMD, or if Smart mode is enabled (CTRLB.SMEN is written to one), when DATA.DATA is read.
This bit is not enable-protected.
This bit is not write-synchronized.

| Value | Description |
|---|---|
| 0 | Send ACK. |
| 1 | Send NACK. |

**Bits 17:16 – CMD[1:0]** Command

Writing these bits triggers a host operation as described below. The CMD bits are strobe bits, and always read as zero. The acknowledge action is only valid in Host Read mode. In Host Write mode, a command will only result in a repeated Start or Stop condition. The CTRLB.ACKACT bit and the CMD bits can be written at the same time, and then the acknowledge action will be updated before the command is triggered.
Commands can only be issued when either the Client on Bus Interrupt flag (INTFLAG.SB) or Host on Bus Interrupt flag (INTFLAG.MB) is '1'.
If CMD 0x1 is issued, a repeated start will be issued followed by the transmission of the current address in ADDR.ADDR. If another address is desired, ADDR.ADDR must be written instead of the CMD bits. This will trigger a repeated start followed by transmission of the new address.
Issuing a command will set the System Operation bit in the Synchronization Busy register (SYNCBUSY.SYSOP).


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 33-4.** Command Description

| CMD[1:0] | Direction | Action |
|---|---|---|
| 0x0 | X | (No action) |
| 0x1 | X | Execute acknowledge action succeeded by repeated Start |
| 0x2 | 0 (Write) | No operation |
| | 1 (Read) | Execute acknowledge action succeeded by a byte read operation |
| 0x3 | X | Execute acknowledge action succeeded by issuing a Stop condition |

These bits are not enable-protected.

**Bit 9 – QCEN** Quick Command Enable
This bit is not write-synchronized.

| Value | Description |
|---|---|
| 0 | Quick Command is disabled. |
| 1 | Quick Command is enabled. |

**Bit 8 – SMEN** Smart Mode Enable
When Smart mode is enabled, acknowledge action is sent when DATA.DATA is read.
This bit is not write-synchronized.

| Value | Description |
|---|---|
| 0 | Smart mode is disabled. |
| 1 | Smart mode is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.3. Baud Rate

**Name:** BAUD
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | HSBAUDLOW[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | HSBAUD[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | BAUDLOW[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BAUD[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:24 – HSBAUDLOW[7:0]** High Speed Host Baud Rate Low

HSBAUDLOW non-zero: HSBAUDLOW indicates the SCL low time in High-speed mode according to $\mathrm{HSBAUDLOW} = f_{\mathrm{GCLK}} \cdot T_{\mathrm{LOW}} - 1$

HSBAUDLOW equal to zero: The HSBAUD register is used to time $T_{LOW}$, $T_{HIGH}$, $T_{SU;STO}$, $T_{HD;STA}$ and $T_{SU;STA}$. $T_{BUF}$ is timed by the BAUD register.

**Bits 23:16 – HSBAUD[7:0]** High Speed Host Baud Rate

This bit field indicates the SCL high time in High-speed mode according to the following formula. When HSBAUDLOW is zero, $T_{LOW}$, $T_{HIGH}$, $T_{SU;STO}$, $T_{HD;STA}$ and $T_{SU;STA}$ are derived using this formula. $T_{BUF}$ is timed by the BAUD register.

$\mathrm{HSBAUD} = f_{\mathrm{GCLK}} \cdot T_{\mathrm{HIGH}} - 1$

**Bits 15:8 – BAUDLOW[7:0]** Host Baud Rate Low

If this bit field is non-zero, the SCL low time will be described by the value written.
For more information on how to calculate the frequency, see SERCOM Clock Generation – Baud-Rate Generator.

**Bits 7:0 – BAUD[7:0]** Host Baud Rate

This bit field is used to derive the SCL high time if BAUD.BAUDLOW is non-zero. If BAUD.BAUDLOW is zero, BAUD will be used to generate both high and low periods of the SCL.
For more information on how to calculate the frequency, see SERCOM Clock Generation – Baud-Rate Generator.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | | SB | MB |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 1 – SB** Client on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Client on Bus Interrupt Enable bit, which disables the Client on Bus interrupt.

| Value | Description |
|---|---|
| 0 | The Client on Bus interrupt is disabled. |
| 1 | The Client on Bus interrupt is enabled. |

**Bit 0 – MB** Host on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the Host on Bus Interrupt Enable bit, which disables the Host on Bus interrupt.

| Value | Description |
|---|---|
| 0 | The Host on Bus interrupt is disabled. |
| 1 | The Host on Bus interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x16
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | | SB | MB |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

**Bit 7 – ERROR** Error Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | Error interrupt is disabled. |
| 1 | Error interrupt is enabled. |

**Bit 1 – SB** Client on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Client on Bus Interrupt Enable bit, which enables the Client on Bus interrupt.

| Value | Description |
|---|---|
| 0 | The Client on Bus interrupt is disabled. |
| 1 | The Client on Bus interrupt is enabled. |

**Bit 0 – MB** Host on Bus Interrupt Enable
Writing '0' to this bit has no effect.
Writing '1' to this bit will set the Host on Bus Interrupt Enable bit, which enables the Host on Bus interrupt.

| Value | Description |
|---|---|
| 0 | The Host on Bus interrupt is disabled. |
| 1 | The Host on Bus interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x18
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ERROR | | | | | | SB | MB |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

**Bit 7 – ERROR** Error

This flag is cleared by writing '1' to it.
This bit is set when any error is detected. Errors that will set this flag have corresponding status bits in the STATUS register. These status bits are LENERR, SEXTTOUT, MEXTTOUT, LOWTOUT, ARBLOST, and BUSERR.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear the flag.

**Bit 1 – SB** Client on Bus

The Client on Bus flag (SB) is set when a byte is successfully received in Host Read mode, for example, no arbitration lost or bus error occurred during the operation. When this flag is set, the host forces the SCL line low, stretching the $I^2C$ clock period. The SCL line will be released and SB will be cleared on one of the following actions:

- Writing to ADDR.ADDR
- Writing to DATA.DATA
- Reading DATA.DATA when Smart mode is enabled (CTRLB.SMEN)
- Writing a valid command to CTRLB.CMD

Writing '1' to this bit location will clear the SB flag. The transaction will not continue or be terminated until one of the above actions is performed.
Writing '0' to this bit has no effect.

**Bit 0 – MB** Host on Bus

This flag is set when a byte is transmitted in Host Write mode. The flag is set regardless of the occurrence of a bus error or an Arbitration Lost condition. MB is also set when arbitration is lost during sending of NACK in Host Read mode, or when issuing a Start condition if the bus state is unknown. When this flag is set and arbitration is not lost, the host forces the SCL line low, stretching the $I^2C$ clock period. The SCL line will be released and MB will be cleared on one of the following actions:

- Writing to ADDR.ADDR
- Writing to DATA.DATA
- Reading DATA.DATA when Smart mode is enabled (CTRLB.SMEN)
- Writing a valid command to CTRLB.CMD

Writing '1' to this bit location will clear the MB flag. The transaction will not continue or be terminated until one of the above actions is performed.
Writing '0' to this bit has no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.7. Status

**Name:** STATUS
**Offset:** 0x1A
**Reset:** 0x0000
**Property:** Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | LENERR | SEXTTOUT | MEXTTOUT |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CLKHOLD | LOWTOUT | BUSSTATE[1:0] | | | RXNACK | ARBLOST | BUSERR |
| Access | R | R/W | R/W | R/W | | R | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bit 10 – LENERR** Transaction Length Error
This bit is set when automatic length is used for a DMA transaction and the client sends a NACK before ADDR.LEN bytes have been written by the host.
Writing '1' to this bit location will clear STATUS.LENERR. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

**Bit 9 – SEXTTOUT** Client SCL Low Extend Time-Out
This bit is set if a client SCL low extend time-out occurs.
This bit is automatically cleared when writing to the ADDR register.
Writing '1' to this bit location will clear SEXTTOUT. Normal use of the $I^2C$ interface does not require the SEXTTOUT flag to be cleared by this method.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

**Bit 8 – MEXTTOUT** Host SCL Low Extend Time-Out
This bit is set if a Host SCL low time-out occurs.
Writing '1' to this bit location will clear STATUS.MEXTTOUT. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.

**Bit 7 – CLKHOLD** Clock Hold
This bit is set when the host is holding the SCL line low, stretching the $I^2C$ clock. Software should consider this bit when INTFLAG.SB or INTFLAG.MB is set.
This bit is cleared when the corresponding Interrupt flag is cleared and the next operation is given.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.
This bit is not write-synchronized.

**Bit 6 – LOWTOUT** SCL Low Time-Out
This bit is set if an SCL low time-out occurs.
Writing '1' to this bit location will clear this bit. This flag is automatically cleared when writing to the ADDR register.
Writing '0' to this bit has no effect.
This bit is not write-synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 5:4 – BUSSTATE[1:0]** Bus State

These bits indicate the current I²C Bus state.
When in UNKNOWN state, writing 0x1 to BUSSTATE forces the bus state into the IDLE state. The bus state cannot be forced into any other state.
Writing BUSSTATE to idle will set SYNCBUSY.SYSOP.

| Value | Name | Description |
|---|---|---|
| 0x0 | UNKNOWN | The Bus state is unknown to the I²C host and will wait for a Stop condition to be detected or wait to be forced into an Idle state by software |
| 0x1 | IDLE | The Bus state is waiting for a transaction to be initialized |
| 0x2 | OWNER | The I²C host is the current owner of the bus |
| 0x3 | BUSY | Some other I²C host owns the bus |

**Bit 2 – RXNACK** Received Not Acknowledge

This bit indicates whether the last address or data packet sent was acknowledged or not.
Writing '0' to this bit has no effect.
Writing '1' to this bit has no effect.
This bit is not write-synchronized.

| Value | Description |
|---|---|
| 0 | Client responded with ACK. |
| 1 | Client responded with NACK. |

**Bit 1 – ARBLOST** Arbitration Lost

This bit is set if arbitration is lost while transmitting a high data bit or a NACK bit, or while issuing a Start or Repeated Start condition on the bus. The Host on Bus Interrupt flag (INTFLAG.MB) will be set when STATUS.ARBLOST is set.
Writing the ADDR.ADDR register will automatically clear STATUS.ARBLOST.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.
This bit is not write-synchronized.

**Bit 0 – BUSERR** Bus Error

This bit indicates that an illegal Bus condition has occurred on the bus, regardless of bus ownership. An illegal Bus condition is detected if a protocol violating start, repeated start or stop is detected on the I²C bus lines. A Start condition directly followed by a Stop condition is one example of a protocol violation. If a time-out occurs during a frame, this is also considered a protocol violation, and will set BUSERR.
If the I²C host is the bus owner at the time a bus error occurs, STATUS.ARBLOST and INTFLAG.MB will be set in addition to BUSERR.
Writing the ADDR.ADDR register will automatically clear the BUSERR flag.
Writing '0' to this bit has no effect.
Writing '1' to this bit will clear it.
This bit is not write-synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.8. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x1C
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SYSOP | ENABLE | SWRST |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – SYSOP** System Operation Synchronization Busy
Writing CTRLB.CMD, STATUS.BUSSTATE, ADDR, or DATA when the SERCOM is enabled requires synchronization. When written, the SYNCBUSY.SYSOP bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | System operation synchronization is not busy. |
| 1 | System operation synchronization is busy. |

**Bit 1 – ENABLE** SERCOM Enable Synchronization Busy
Enabling and disabling the SERCOM (CTRLA.ENABLE) requires synchronization. When written, the SYNCBUSY.ENABLE bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | Enable synchronization is not busy. |
| 1 | Enable synchronization is busy. |

**Bit 0 – SWRST** Software Reset Synchronization Busy
Resetting the SERCOM (CTRLA.SWRST) requires synchronization. When written, the SYNCBUSY.SWRST bit will be set until synchronization is complete.

| Value | Description |
|---|---|
| 0 | SWRST synchronization is not busy. |
| 1 | SWRST synchronization is busy. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.9. Address

**Name:** ADDR
**Offset:** 0x24
**Reset:** 0x0000
**Property:** Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LEN[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TENBITEN | HS | LENEN | | | ADDR[10:8] | | |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADDR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:16 – LEN[7:0]** Transaction Length
These bits define the transaction length of a DMA transaction from 0 to 255 bytes. The Transfer Length Enable (LENEN) bit must be written to '1' in order to use DMA.

**Bit 15 – TENBITEN** Ten Bit Addressing Enable
This bit enables 10-bit addressing. This bit can be written simultaneously with ADDR to indicate a 10-bit or 7-bit address transmission.

| Value | Description |
|---|---|
| 0 | 10-bit addressing disabled. |
| 1 | 10-bit addressing enabled. |

**Bit 14 – HS** High Speed
This bit enables High-speed mode for the current transfer from repeated START to STOP. This bit can be written simultaneously with ADDR for a high speed transfer.

| Value | Description |
|---|---|
| 0 | High-speed transfer disabled. |
| 1 | High-speed transfer enabled. |

**Bit 13 – LENEN** Transfer Length Enable

| Value | Description |
|---|---|
| 0 | Automatic transfer length disabled. |
| 1 | Automatic transfer length enabled. |

**Bits 10:0 – ADDR[10:0]** Address
When ADDR is written, the consecutive operation will depend on the bus state:


© 2026 Microchip Technology Inc. and its subsidiaries

UNKNOWN: INTFLAG.MB and STATUS.BUSERR are set, and the operation is terminated.
BUSY: The $I^2C$ host will await further operation until the bus becomes IDLE.
IDLE: The $I^2C$ host will issue a start condition followed by the address written in ADDR. If the address is acknowledged, SCL is forced and held low, and STATUS.CLKHOLD and INTFLAG.MB are set.
OWNER: A repeated start sequence will be performed. If the previous transaction was a read, the acknowledge action is sent before the repeated start bus condition is issued on the bus. Writing ADDR to issue a repeated start is performed while INTFLAG.MB or INTFLAG.SB is set. STATUS.BUSERR, STATUS.ARBLOST, INTFLAG.MB and INTFLAG.SB will be cleared when ADDR is written.
The ADDR register can be read at any time without interfering with ongoing bus activity, as a read access does not trigger the host logic to perform any bus protocol related operations.
The $I^2C$ host control logic uses bit 0 of ADDR as the bus protocol's read/write flag (R/W); 0 for write and 1 for read.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.10. Data

**Name:** DATA
**Offset:** 0x28
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – DATA[7:0]** Data

The Host data register I/O location (DATA) provides access to the Host transmit and receive data buffers. Reading valid data or writing data to be transmitted can be successfully done only when SCL is held low by the Host (STATUS.CLKHOLD is set). An exception is reading the last data byte after the stop condition has been sent.

Accessing DATA.DATA auto-triggers $I^2C$ bus operations. The operation performed depends on the state of CTRLB.ACKACT, CTRLB.SMEN and the type of access (read/write).

Writing or reading DATA.DATA when not in Smart mode does not require synchronization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 33.10.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x30
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGSTOP |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGSTOP** Debug Stop Mode
This bit controls functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The baud-rate generator continues normal operation when the CPU is halted by an external debugger. |
| 1 | The baud-rate generator is halted when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 34. CAN - Control Area Network (SAM C21 Only)

## 34.1. Overview

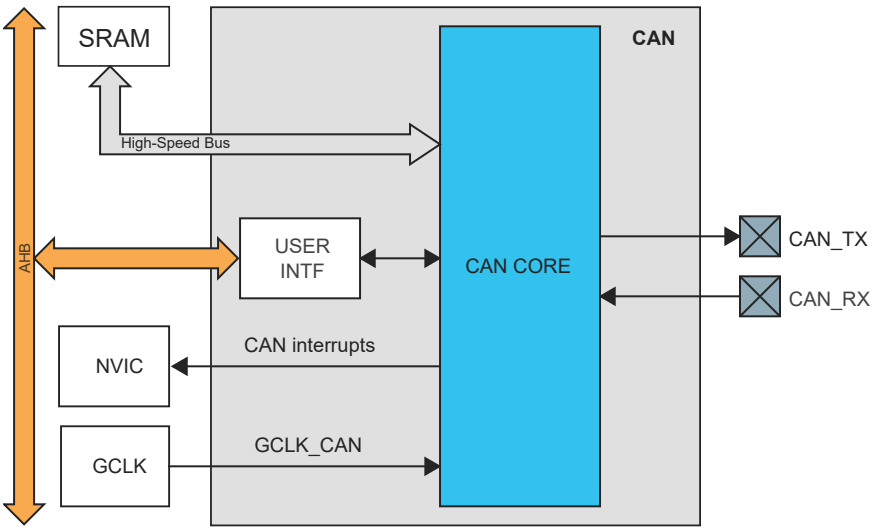
The Control Area Network (CAN) performs communication according to ISO 11898-1:2015 (Bosch CAN specification 2.0 part A,B, ISO CAN FD). Message storage is intended to be a single- or dual-ported Message RAM outside of the module.

## 34.2. Features

- Conform with CAN protocol version 2.0 part A, B and ISO 11898-1:2015
- Up to two Controller Area Network CAN interfaces
  - Supporting CAN2.0 A/B and CAN-FD (ISO 11898-1:2015)
- CAN FD with up to 64 data bytes supported
- CAN Error Logging
- AUTOSAR optimized
- SAE J1939 optimized
- Two configurable Receive FIFOs
- Separate signaling on reception of High-Priority Messages
- Up to 64 dedicated Receive Buffers and up to 32 dedicated Transmit Buffers
- Configurable Transmit FIFO, Transmit Queue, Transmit Event FIFO
- Direct Message RAM access for CPU
- Programmable Loop-Back Test mode
- Maskable module interrupts
- Power-down support; Debug on CAN support
- Transfer rates:
  - 1 Mb/s for CAN 2.0 mode
  - 10 Mb/s for CAN-FD mode

## 34.3. Block Diagram

**Figure 34-1.** CAN Block Diagram





© 2026 Microchip Technology Inc. and its subsidiaries

## 34.4. Signal Description

**Table 34-1.** Signal Description

| Signal | Description | Type |
|---|---|---|
| CAN_TX | CAN transmit | Digital output |
| CAN_RX | CAN receive | Digital input |

Refer to for details on the pin mapping for this peripheral. One signal can be mapped to one of several pins.

**Related Links**
I/O Multiplexing and Considerations

## 34.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 34.5.1. I/O Lines

Using the CAN's I/O lines requires the I/O pins to be configured.

**Related Links**
PORT - I/O Pin Controller

### 34.5.2. Power Management

The CAN continues to operate in idle0 Sleep mode, but cannot operate in idle2 and standby Sleep modes. Consequently, the CAN interrupts can only be used to wake up the device from idle0 Sleep mode. Refer to the Power Manager chapter for details on the different sleep modes.

The CAN module has its own Low-Power mode. The clock sources cannot be halted while the CAN is enabled unless this mode is used. Refer to the section "Sleep Mode Operation" for additional information.

**Related Links**
Sleep Mode Operation

### 34.5.3. Clocks

An AHB clock (CLK_CAN_AHB) is required to clock the CAN. This clock can be configured in the Main Clock peripheral (MCLK) before using the CAN, and the default state of CLK_CAN_AHB can be found in the MCLK.AHBMASK register.

A generic clock (GCLK_CAN) is required to clock the CAN. This clock must be configured and enabled in the generic clock controller before using the CAN.

This generic clock is asynchronous to the bus clock (CLK_CAN_AHB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains.

**Related Links**
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 34.5.4. DMA

The CAN has a built-in Direct Memory Access (DMA) and will read/write data to/from the system RAM when a CAN transaction takes place. No CPU or DMA Controller (DMAC) resources are required.

The DMAC can be used for debug messages functionality.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
DMAC – Direct Memory Access Controller

### 34.5.5. Interrupts

The interrupt request lines are connected to the interrupt controller. Using the CAN interrupts requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 34.5.6. Events

Not applicable.

### 34.5.7. Debug Operation

Not applicable.

### 34.5.8. Register Access Protection

Not applicable.

### 34.5.9. Analog Connections

No analog connections.

## 34.6. Functional Description

### 34.6.1. Principle of Operation

The CAN performs communication according to ISO 11898-1:2015 (identical to Bosch CAN protocol specification 2.0 part A,B, ISO CAN FD).

The Message storage is intended to be a single- or dual-ported Message RAM outside the module. It is connected to the CAN via AHB.

All functions concerning the handling of messages are implemented by the Rx Handler and the Tx Handler. The Rx Handler manages message acceptance filtering, the transfer of received messages from the CAN Core to the Message RAM as well as providing receive message status information. The Tx Handler is responsible for the transfer of transmit messages from the Message RAM to the CAN Core as well as providing transmit status information.

Acceptance filtering is implemented by a combination of up to 128 filter elements where each one can be configured as a range, as a bit mask, or as a dedicated ID filter.

### 34.6.2. Operating Modes

#### 34.6.2.1. Software Initialization

Software initialization is started by setting bit CCCR.INIT, either by software or by a hardware reset, when an uncorrected bit error was detected in the Message RAM, or by going Bus_Off. While CCCR.INIT is set, message transfer from and to the CAN bus is stopped, the status of the CAN bus output CAN_TX is ”recessive” (HIGH). The counters of the Error Management Logic EML are unchanged. Setting CCCR.INIT does not change any configuration register. Resetting CCCR.INIT finishes the software initialization. Afterwards the Bit Stream Processor BSP synchronizes itself to the data transfer on the CAN bus by waiting for the occurrence of a sequence of 11 consecutive ”recessive” bits (= Bus_Idle) before it can take part in bus activities and start the message transfer.

Access to the CAN configuration registers is only enabled when both bits CCCR.INIT and CCCR.CCE are set (protected write).


© 2026 Microchip Technology Inc. and its subsidiaries

CCCR.CCE can only be set/reset while CCCR.INIT = '1'. CCCR.CCE is automatically reset when CCCR.INIT is reset.

The following registers are reset when CCCR.CCE is set

- HPMS - High Priority Message Status
- RXF0S - Rx FIFO 0 Status
- RXF1S - Rx FIFO 1 Status
- TXFQS - Tx FIFO/Queue Status
- TXBRP - Tx Buffer Request Pending
- TXBTO - Tx Buffer Transmission Occurred
- TXBCF - Tx Buffer Cancellation Finished
- TXEFS - Tx Event FIFO Status

The Timeout Counter value TOCV.TOC is preset to the value configured by TOCC.TOP when CCCR.CCE is set.

In addition the state machines of the Tx Handler and Rx Handler are held in idle state while CCCR.CCE = '1'.

The following registers are only writable while CCCR.CCE = '0'

- TXBAR - Tx Buffer Add Request
- TXBCR - Tx Buffer Cancellation Request

CCCR.TEST and CCCR.MON can only be set by the CPU while CCCR.INIT = '1' and CCR.CCE = '1'. Both bits may be reset at any time. CCCR.DAR can only be set/reset while CCCR.INIT = '1' and CCCR.CCE = '1'.

#### 34.6.2.2. Normal Operation

Once the CAN is initialized and CCCR.INIT is reset to '0', the CAN synchronizes itself to the CAN bus and is ready for communication.

After passing the acceptance filtering, received messages including Message ID and DLC are stored into a dedicated Rx Buffer or into Rx FIFO0 or Rx FIFO1.

For messages to be transmitted dedicated Tx Buffers and/or a Tx FIFO or a Tx Queue can be initialized or updated. Automated transmission on reception of remote frames is not implemented.

#### 34.6.2.3. CAN FD Operation

There are two variants in the CAN FD frame format, first the CAN FD frame without bit rate switching where the data field of a CAN frame may be longer than 8 bytes. The second variant is the CAN FD frame where control field, data field, and CRC field of a CAN frame are transmitted with a higher bit rate than the beginning and the end of the frame.

The previously reserved bit in CAN frames with 11-bit identifiers and the first previously reserved bit in CAN frames with 29-bit identifiers will now be decoded as FDF bit. FDF = recessive signifies a CAN FD frame, FDF = dominant signifies a Classic CAN frame. In a CAN FD frame, the two bits following FDF, res and BRS, decide whether the bit rate inside of this CAN FD frame is switched. A CAN FD bit rate switch is signified by res = dominant and BRS = recessive. The coding of res = recessive is reserved for future expansion of the protocol. In case the CAN receives a frame with FDF = recessive and res = recessive, it will signal a Protocol Exception Event by setting bit PSR.PXE. When Protocol Exception Handling is enabled (CCCR.PXHD = '0'), this causes the operation state to change from Receiver (PSR.ACT = "10") to Integrating (PSR.ACT = "00") at the next sample point. In case Protocol Exception Handling is disabled (CCCR.PXHD = '1'), the CAN will treat a recessive res bit as a form error and will respond with an error frame.


© 2026 Microchip Technology Inc. and its subsidiaries

CAN FD operation is enabled by programming CCCR.FDOE. In case CCCR.FDOE = '1', transmission and reception of CAN FD frames is enabled. Transmission and reception of Classic CAN frames is always possible. Whether a CAN FD frame or a Classic CAN frame is transmitted can be configured via bit FDF in the respective Tx Buffer element. With CCCR.FDOE = '0', received frames are interpreted as Classic CAN frames, witch leads to the transmission of an error frame when receiving a CAN FD frame. When CAN FD operation is disabled, no CAN FD frames are transmitted even if bit FDF of a Tx Buffer element is set. CCCR.FDOE and CCCR.BRSE can only be changed while CCCR.INIT and CCCR.CCE are both set.

With CCCR.FDOE = '0', the setting of bits FDF and BRS is ignored and frames are transmitted in Classic CAN format. With CCCR.FDOE = '1' and CCCR.BRSE = '0', only bit FDF of a Tx Buffer element is evaluated. With CCCR.FDOE = '1' and CCCR.BRSE = '1', transmission of CAN FD frames with bit rate switching is enabled. All Tx Buffer elements with bits FDF and BRS set are transmitted in CAN FD format with bit rate switching.

A mode change during CAN operation is only recommended under the following conditions:

- The failure rate in the CAN FD data phase is significantly higher than in the CAN FD arbitration phase. In this case disable the CAN FD bit rate switching option for transmissions.
- During system startup all nodes are transmitting Classic CAN messages until it is verified that they are able to communicate in CAN FD format. If this is true, all nodes switch to CAN FD operation.
- Wake-up messages in CAN Partial Networking have to be transmitted in Classic CAN format.
- End-of-line programming in case not all nodes are CAN FD capable. Non CAN FD nodes are held in silent mode until programming has completed. Then all nodes switch back to Classic CAN communication.

In the CAN FD format, the coding of the DLC differs from the standard CAN format. The DLC codes 0 to 8 have the same coding as in standard CAN, the codes 9 to 15, which in standard CAN all code a data field of 8 bytes, are coded according to the table below.

**Table 34-2.** Coding of DLC in CAN FD

| DLC | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|
| Number of Data Bytes | 12 | 16 | 20 | 24 | 32 | 48 | 64 |

In CAN FD frames, the bit timing will be switched inside the frame, after the BRS (Bit Rate Switch) bit, if this bit is recessive. Before the BRS bit, in the CAN FD arbitration phase, the nominal CAN bit timing is used as defined by the Nominal Bit Timing & Prescaler Register NBTP. In the following CAN FD data phase, the fast CAN bit timing is used as defined by the Data Bit Timing & Prescaler Register DBTP. The bit timing is switched back from the fast timing at the CRC delimiter or when an error is detected, whichever occurs first.

The maximum configurable bit rate in the CAN FD data phase depends on the CAN clock frequency (GCLK_CAN). Example: with a CAN clock frequency of 20MHz and the shortest configurable bit time of 4 $t_q$, the bit rate in the data phase is 5 Mbit/s.

In both data frame formats, CAN FD long and CAN FD fast, the value of the bit ESI (Error Status Indicator) is determined by the transmitter's error state at the start of the transmission. If the transmitter is error passive, ESI is transmitted recessive, else it is transmitted dominant.

#### 34.6.2.4. Transceiver Delay Compensation

During the data phase of a CAN FD transmission only one node is transmitting, all others are receivers. The length of the bus line has no impact. When transmitting via pin CAN_TX the CAN receives the transmitted data from its local CAN transceiver via pin CAN_RX. The received data is delayed by the CAN transceiver's loop delay. In case this delay is greater than TSEG1 (time segment before sample point), a bit error is detected. In order to enable a data phase bit time that is even shorter than the transceiver loop delay, the delay compensation is introduced. Without transceiver


© 2026 Microchip Technology Inc. and its subsidiaries

delay compensation, the bit rate in the data phase of a CAN FD frame is limited by the transceivers loop delay.

Description

The CAN's protocol unit has implemented a delay compensation mechanism to compensate the transmitter delay, thereby enabling transmission with higher bit rates during the CAN FD data phase independent of the delay of a specific CAN transceiver.

To check for bit errors during the data phase of transmitting nodes, the delayed transmit data is compared against the received data at the Secondary Sample Point SSP. If a bit error is detected, the transmitter will react on this bit error at the next following regular sample point. During arbitration phase the delay compensation is always disabled.

The transmitter delay compensation enables configurations where the data bit time is shorter than the transmitter delay, it is described in detail in the new ISO11898-1. It is enabled by setting bit DBTP.TDC.

The received bit is compared against the transmitted bit at the SSP. The SSP position is defined as the sum of the measured delay from the CAN's transmit output CAN_TX through the transceiver to the receive input CAN_RX plus the transmitter delay compensation offset as configured by TDCR.TDCO. The transmitter delay compensation offset is used to adjust the position of the SSP inside the received bit (e.g. half of the bit time in the data phase). The position of the secondary sample point is rounded down to the next integer number of mtq.

PSR.TDCV shows the actual transmitter delay compensation value. PSR.TDCV is cleared when CCCR.INIT is set and is updated at each transmission of an FD frame while DBTP.TDC is set.

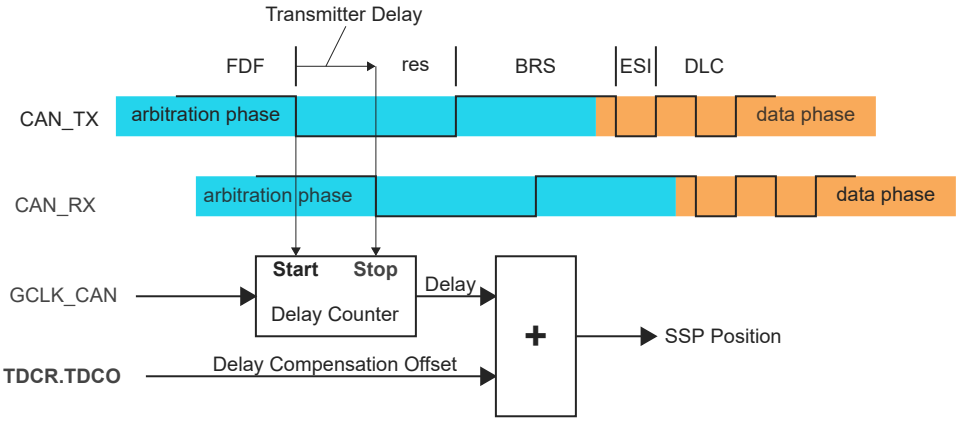
The following boundary conditions have to be considered for the transmitter delay compensation implemented in the CAN:

- The sum of the measured delay from CAN_TX to CAN_RX and the configured transceiver delay compensation offset FBTP.TDCO has to be less than 6 bit times in the data phase.
- The sum of the measured delay from CAN_TX to CAN_RX and the configured transceiver delay compensation offset FBTP.TDCO has to be less or equal to 127 mtq. In case this sum exceeds 127 mtq, the maximum value of 127 mtq is used for transceiver delay compensation.
- The data phase ends at the sample point of the CRC delimiter, that stops checking of receive bits at the SSPs.
  Transmitter Delay Compensation Measurement

If transmitter delay compensation is enabled by programming DBTP.TDC = '1', the measurement is started within each transmitted CAN FD frame at the falling edge of bit FDF to bit res. The measurement is stopped when this edge is seen at the receive input CAN_TX of the transmitter. The resolution of this measurement is one mtq.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-2.** Transceiver delay measurement




To avoid that a dominant glitch inside the received FDF bit ends the delay compensation measurement before the falling edge of the received res bit, resulting in a too early SSP position, the use of a transmitter delay compensation filter window can be enabled by programming TDCR.TDCF. This defines a minimum value for the SSP position. Dominant edges of CAN_RX, that would result in an earlier SSP position are ignored for transmitter delay measurement. The measurement is stopped when the SSP position is at least TDCR.TDCF AND CAN _RX is low.

### 34.6.2.5. Restricted Operation Mode

In Restricted Operation Mode the node is able to receive data and remote frames and to give acknowledge to valid frames, but it does not send data frames, remote frames, active error frames, or overload frames. In case of an error condition or overload condition, it does not send dominant bits, instead it waits for the occurrence of bus idle condition to resynchronize itself to the CAN communication. The error counters (ECR.REC, ECR.TEC) are frozen while Error Logging (ECR.CEL) is still incremented. The CPU can set the CAN into Restricted Operation mode by setting bit CCCR.ASM. The bit can only be set by the CPU when both CCCR.CCE and CCCR.INIT are set to '1'. The bit can be reset by the CPU at any time.

Restricted Operation Mode is automatically entered when the Tx Handler was not able to read data from the Message RAM in time. To leave Restricted Operation Mode, the CPU has to reset CCCR.ASM.

The Restricted Operation Mode can be used in applications that adapt themselves to different CAN bit rates. In this case the application tests different bit rates and leaves the Restricted Operation Mode after it has received a valid frame.

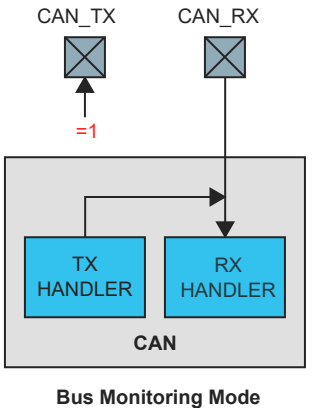
### 34.6.2.6. Bus Monitoring Mode

The CAN is set in Bus Monitoring Mode by programming CCCR.MON to '1'. In Bus Monitoring Mode (see ISO 11898-1, 10.12 Bus monitoring), the CAN is able to receive valid data frames and valid remote frames, but cannot start a transmission. In this mode, it sends only recessive bits on the CAN bus. If the CAN is required to send a dominant bit (ACK bit, overload flag, active error flag), the bit is rerouted internally so that the CAN monitors this dominant bit, although the CAN bus may remain in recessive state. In Bus Monitoring Mode register TXBRP is held in reset state.

The Bus Monitoring Mode can be used to analyze the traffic on a CAN bus without affecting it by the transmission of dominant bits. The figure below shows the connection of signals CAN_TX and CAN_RX to the CAN in Bus Monitoring Mode.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-3.** Pin Control in Bus Monitoring Mode




### 34.6.2.7. Disabled Automatic Retransmission

According to the CAN Specification (see ISO 11898-1, 6.3.3 Recovery Management), the CAN provides means for automatic retransmission of frames that have lost arbitration or that have been disturbed by errors during transmission. By default automatic retransmission is enabled. To support time-triggered communication as described in ISO 11898-1, chapter 9.2, the automatic retransmission may be disabled via CCCR.DAR.

Frame Transmission in DAR Mode

In DAR mode all transmissions are automatically cancelled after they started on the CAN bus. A Tx Buffer's Tx Request Pending bit TXBRP.TRPx is reset after successful transmission, when a transmission has not yet been started at the point of cancellation, has been aborted due to lost arbitration, or when an error occurred during frame transmission.

- Successful transmission:
  - Corresponding Tx Buffer Transmission Occurred bit TXBTO.TOx set
  - Corresponding Tx Buffer Cancellation Finished bit TXBCF.CFx not set
- Successful transmission in spite of cancellation:
  - Corresponding Tx Buffer Transmission Occurred bit TXBTO.TOx set
  - Corresponding Tx Buffer Cancellation Finished bit TXBCF.CFx set
- Arbitration lost or frame transmission disturbed:
  - Corresponding Tx Buffer Transmission Occurred bit TXBTO.TOx not set
  - Corresponding Tx Buffer Cancellation Finished bit TXBCF.CFx set

In case of a successful frame transmission, and if storage of Tx events is enabled, a Tx Event FIFO element is written with Event Type ET = "10" (transmission in spite of cancellation).

### 34.6.2.8. Test Modes

To enable write access to register TEST, bit CCCR.TEST has to be set to '1'. This allows the configuration of the test modes and test functions.

Four output functions are available for the CAN transmit pin CAN_TX by programming TEST.TX. Additionally to its default function – the serial data output – it can drive the CAN Sample Point signal to monitor the CAN's bit timing and it can drive constant dominant or recessive values. The actual value at pin CAN_RX can be read from TEST.RX. Both functions can be used to check the CAN bus' physical layer.

Due to the synchronization mechanism between GCLK_CAN and GCLK_CAN_APB domains, there may be a delay of several GCLK_CAN_APB periods between writing to TEST.TX until the new


© 2026 Microchip Technology Inc. and its subsidiaries

configuration is visible at output pin CAN_TX. This applies also when reading input pin CAN_RX via TEST.RX.

Note: Test modes should be used for production tests or self test only. The software control for pin CAN_TX interferes with all CAN protocol functions. It is not recommended to use test modes for application.

### External Loop Back Mode

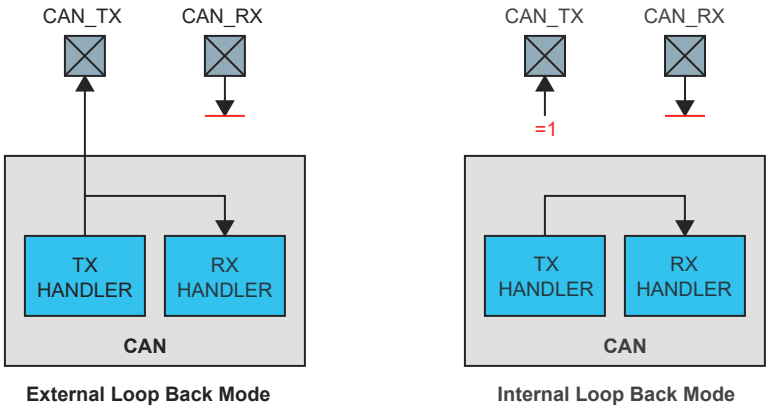
The CAN can be set in External Loop Back Mode by programming TEST.LBCK to '1'. In Loop Back Mode, the CAN treats its own transmitted messages as received messages and stores them (if they pass acceptance filtering) into an Rx Buffer or an Rx FIFO. The figure below shows the connection of signals CAN_TX and CAN_RX to the CAN in External Loop Back Mode.

This mode is provided for hardware self-test. To be independent from external stimulation, the CAN ignores acknowledge errors (recessive bit sampled in the acknowledge slot of a data/remote frame) in Loop Back Mode. In this mode the CAN performs an internal feedback from its Tx output to its Rx input. The actual value of the CAN_RX input pin is disregarded by the CAN. The transmitted messages can be monitored at the CAN_TX pin.

### Internal Loop Back Mode

Internal Loop Back Mode is entered by programming bits TEST.LBCK and CCCR.MON to '1'. This mode can be used for a "Hot Selftest", meaning the CAN can be tested without affecting a running CAN system connected to the pins CAN_TX and CAN_RX. In this mode pin CAN_RX is disconnected from the CAN and pin CAN_TX is held recessive. The figure below shows the connection of CAN_TX and CAN_RX to the CAN in case of Internal Loop Back Mode.

**Figure 34-4.** Pin Control in Loop Back Modes




## 34.6.3. Timestamp Generation

For timestamp generation the CAN supplies a 16-bit wrap-around counter. A prescaler TSCC.TCP can be configured to clock the counter in multiples of CAN bit times (1...16). The counter is readable via TSCV.TSC. A write access to register TSCV resets the counter to zero. When the timestamp counter wraps around interrupt flag IR.TSW is set.

On start of frame reception / transmission the counter value is captured and stored into the timestamp section of an Rx Buffer / Rx FIFO (RXTS[15:0]) or Tx Event FIFO (TXTS[15:0]) element.

## 34.6.4. Timeout Counter

To signal timeout conditions for Rx FIFO 0, Rx FIFO 1, and the Tx Event FIFO the CAN supplies a 16-bit Timeout Counter. It operates as down-counter and uses the same prescaler controlled by TSCC.TCP as the Timestamp Counter. The Timeout Counter is configured via register TOCC. The actual counter


© 2026 Microchip Technology Inc. and its subsidiaries

value can be read from TOCV.TOC. The Timeout Counter can only be started while CCCR.INIT = '0'. It is stopped when CCCR.INIT = '1', e.g. when the CAN enters Bus_Off state.

The operation mode is selected by TOCC.TOS. When operating in Continuous Mode, the counter starts when CCCR.INIT is reset. A write to TOCV presets the counter to the value configured by TOCC.TOP and continues down-counting.

When the Timeout Counter is controlled by one of the FIFOs, an empty FIFO presets the counter to the value configured by TOCC.TOP. Down-counting is started when the first FIFO element is stored. Writing to TOCV has no effect.

When the counter reaches zero, interrupt flag IR.TOO is set. In Continuous Mode, the counter is immediately restarted at TOCC.TOP.

Note: The clock signal for the Timeout Counter is derived from the CAN Core's sample point signal. Therefore the point in time where the Timeout Counter is decremented may vary due to the synchronization / re-synchronization mechanism of the CAN Core. If the baud rate switch feature in CAN FD is used, the timeout counter is clocked differently in arbitration and data field.

### 34.6.5. Rx Handling

The Rx Handler controls the acceptance filtering, the transfer of received messages to the Rx Buffers or to one of the two Rx FIFOs, as well as the Rx FIFO's Put and Get Indices.

#### 34.6.5.1. Acceptance Filtering

The CAN offers the possibility to configure two sets of acceptance filters, one for standard identifiers and one for extended identifiers. These filters can be assigned to an Rx Buffer or to Rx FIFO 0,1. For acceptance filtering each list of filters is executed from element #0 until the first matching element. Acceptance filtering stops at the first matching element. The following filter elements are not evaluated for this message.

The main features are:

- Each filter element can be configured as
  - range filter (from - to)
  - filter for one or two dedicated IDs
  - classic bit mask filter
- Each filter element is configurable for acceptance or rejection filtering
- Each filter element can be enabled / disabled individually
- Filters are checked sequentially, execution stops with the first matching filter element

Related configuration registers are:

- Global Filter Configuration GFC
- Standard ID Filter Configuration SIDFC
- Extended ID Filter Configuration XIDFC
- Extended ID AND Mask XIDAM

Depending on the configuration of the filter element (SFEC/EFEC) a match triggers one of the following actions:

- Store received frame in FIFO 0 or FIFO 1
- Store received frame in Rx Buffer
- Store received frame in Rx Buffer and generate pulse at filter event pin
- Reject received frame
- Set High Priority Message interrupt flag IR.HPM


© 2026 Microchip Technology Inc. and its subsidiaries

- Set High Priority Message interrupt flag IR.HPM and store received frame in FIFO 0 or FIFO 1

Acceptance filtering is started after the complete identifier has been received. After acceptance filtering has completed, and if a matching Rx Buffer or Rx FIFO has been found, the Message Handler starts writing the received message data in portions of 32 bit to the matching Rx Buffer or Rx FIFO. If the CAN protocol controller has detected an error condition (e.g. CRC error), this message is discarded with the following impact on the affected Rx Buffer or Rx FIFO:

Rx Buffer
New Data flag of matching Rx Buffer is not set, but Rx Buffer (partly) overwritten with received data. For error type see PSR.LEC respectively PSR.FLEC.

Rx FIFO
Put index of matching Rx FIFO is not updated, but related Rx FIFO element (partly) overwritten with received data. For error type see PSR.LEC respectively PSR.FLEC. In case the matching Rx FIFO is operated in overwrite mode, the boundary conditions described in Rx FIFO Overwrite Mode have to be considered.

Note: When an accepted message is written to one of the two Rx FIFOs, or into an Rx Buffer, the unmodified received identifier is stored independent of the filter(s) used. The result of the acceptance filter process is strongly depending on the sequence of configured filter elements.

**Range Filter**

The filter matches for all received frames with Message IDs in the range defined by SF1ID/SF2ID for standard frames or EF1ID/EF2ID for extended frames.

There are two possibilities when range filtering is used together with extended frames:

| | |
|---|---|
| **EFT = "00"** | The Message ID of received frames is AND'ed with the Extended ID AND Mask (XIDAM) before the range filter is applied |
| **EFT = "11"** | The Extended ID AND Mask (XIDAM) is not used for range filtering |

**Filter for specific IDs**

A filter element can be configured to filter for one or two specific Message IDs. To filter for one specific Message ID, the filter element has to be configured with SF1ID = SF2ID resp. EF1ID = EF2ID.

**Classic Bit Mask Filter**

Classic bit mask filtering is intended to filter groups of Message IDs by masking single bits of a received Message ID. With classic bit mask filtering SF1ID/EF1ID is used as Message ID filter, while SF2ID/EF2ID is used as filter mask.

A zero bit at the filter mask will mask out the corresponding bit position of the configured ID filter, e.g. the value of the received Message ID at that bit position is not relevant for acceptance filtering. Only those bits of the received Message ID where the corresponding mask bits are one are relevant for acceptance filtering.

In case all mask bits are one, a match occurs only when the received Message ID and the Message ID filter are identical. If all mask bits are zero, all Message IDs match.

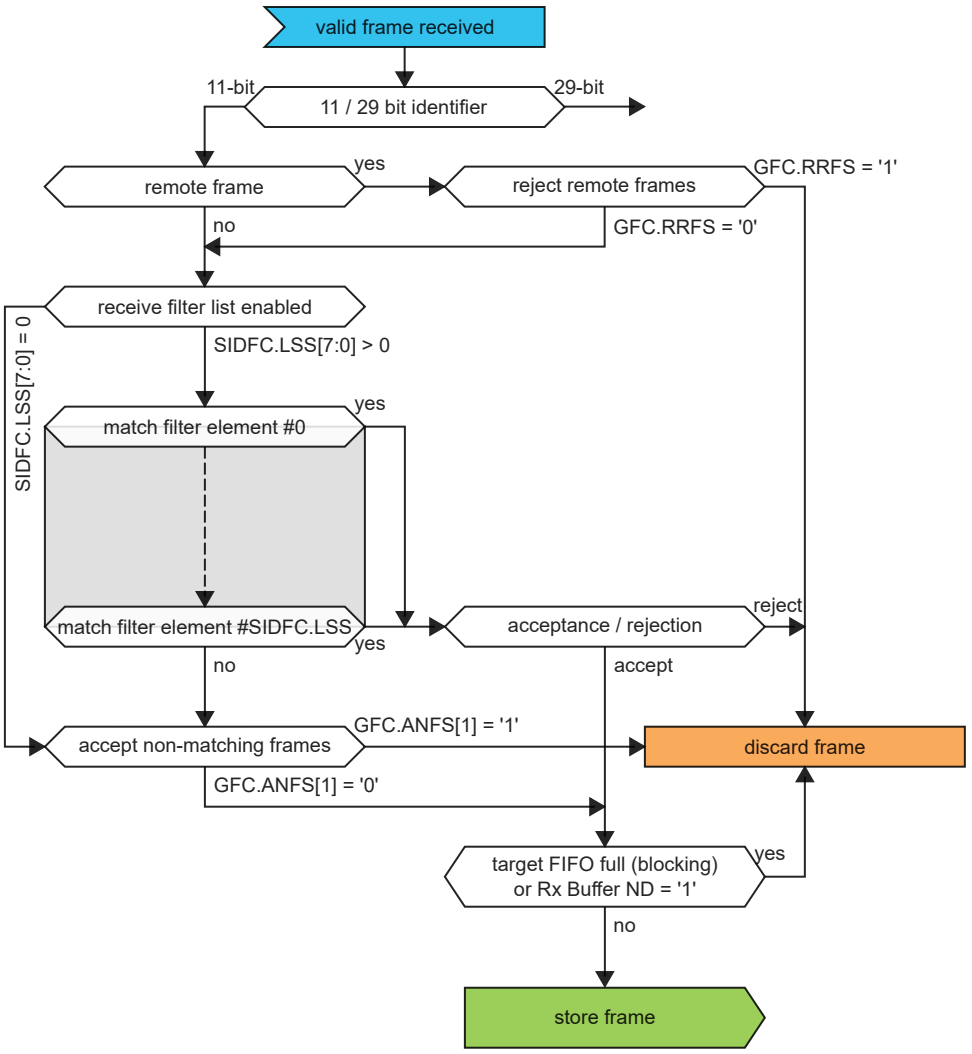
**Standard Message ID Filtering**

The figure below shows the flow for standard Message ID (11-bit Identifier) filtering. The Standard Message ID Filter element is described in Standard Message ID Filter Element.

Controlled by the Global Filter Configuration GFC and the Standard ID Filter Configuration SIDFC Message ID, Remote Transmission Request bit (RTR), and the Identifier Extension bit (IDE) of received frames are compared against the list of configured filter elements.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-5.** Standard Message ID Filtering




### Extended Message ID Filtering

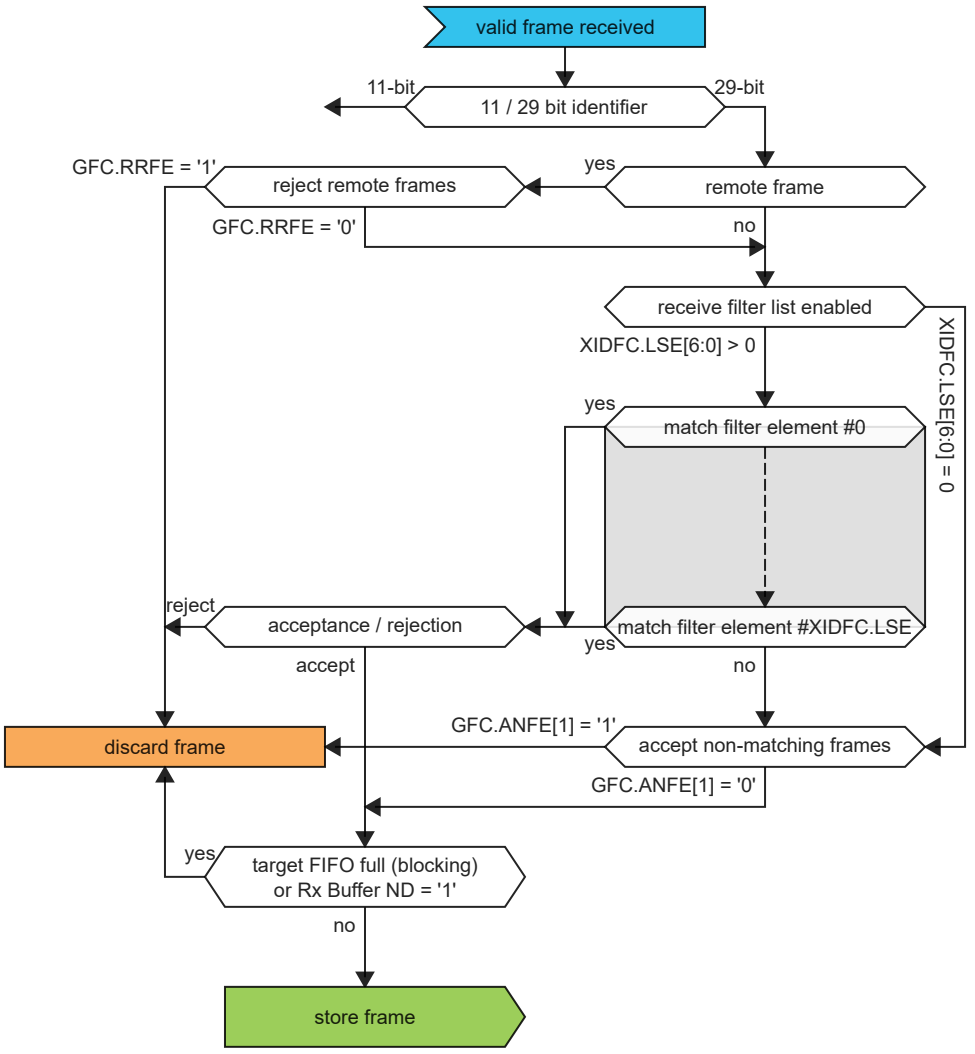
The figure below shows the flow for extended Message ID (29-bit Identifier) filtering. The Extended Message ID Filter element is described in Extended Message ID Filter Element.

Controlled by the Global Filter Configuration GFC and the Extended ID Filter Configuration XIDFC Message ID, Remote Transmission Request bit (RTR), and the Identifier Extension bit (IDE) of received frames are compared against the list of configured filter elements.

The Extended ID AND Mask XIDAM is AND'ed with the received identifier before the filter list is executed.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-6.** Extended Message ID Filtering




### 34.6.5.2. Rx FIFOs

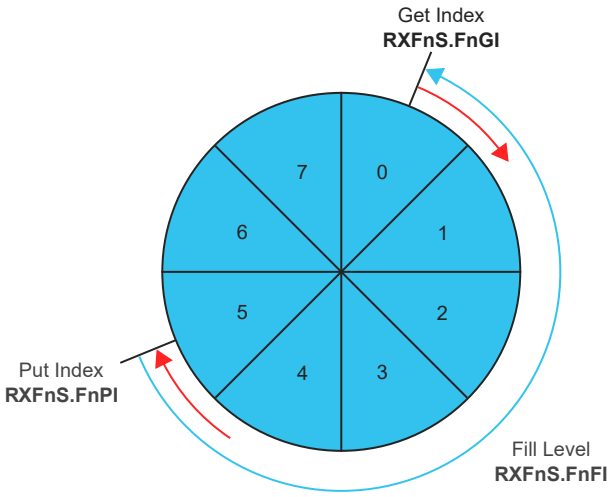
Rx FIFO 0 and Rx FIFO 1 can be configured to hold up to 64 elements each. Configuration of the two Rx FIFOs is done via registers RXF0C and RXF1C.

Received messages that passed acceptance filtering are transferred to the Rx FIFO as configured by the matching filter element. For a description of the filter mechanisms available for Rx FIFO 0 and Rx FIFO 1 see Acceptance Filtering. The Rx FIFO element is described in Rx Buffer and FIFO Element.

To avoid an Rx FIFO overflow, the Rx FIFO watermark can be used. When the Rx FIFO fill level reaches the Rx FIFO watermark configured by RXFnC.FnWM, interrupt flag IR.RFnW is set. When the Rx FIFO Put Index reaches the Rx FIFO Get Index an Rx FIFO Full condition is signalled by RXFnS.FnF. In addition interrupt flag IR.RFnF is set.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-7.** Rx FIFO Status




When reading from an Rx FIFO, Rx FIFO Get Index RXFnS.FnGI • FIFO Element Size has to be added to the corresponding Rx FIFO start address RXFnC.FnSA.

**Table 34-3.** Rx Buffer / FIFO Element Size

| RXESC.RBDS[2:0]<br>RXESC.FnDS[2:0] | Data Field<br>[bytes] | FIFO Element Size<br>[RAM words] |
|---|---|---|
| 000 | 8 | 4 |
| 001 | 12 | 5 |
| 010 | 16 | 6 |
| 011 | 20 | 7 |
| 100 | 24 | 8 |
| 101 | 32 | 10 |
| 110 | 48 | 14 |
| 111 | 64 | 18 |

### Rx FIFO Blocking Mode

The Rx FIFO blocking mode is configured by RXFnC.FnOM = '0'. This is the default operation mode for the Rx FIFOs.

When an Rx FIFO full condition is reached (RXFnS.FnPI = RXFnS.FnGI), no further messages are written to the corresponding Rx FIFO until at least one message has been read out and the Rx FIFO Get Index has been incremented. An Rx FIFO full condition is signaled by RXFnS.FnF = '1'. In addition interrupt flag IR.RFnF is set.

In case a message is received while the corresponding Rx FIFO is full, this message is discarded and the message lost condition is signalled by RXFnS.RFnL = '1'. In addition interrupt flag IR.RFnL is set.

### Rx FIFO Overwrite Mode

The Rx FIFO overwrite mode is configured by RXFnC.FnOM = '1'.

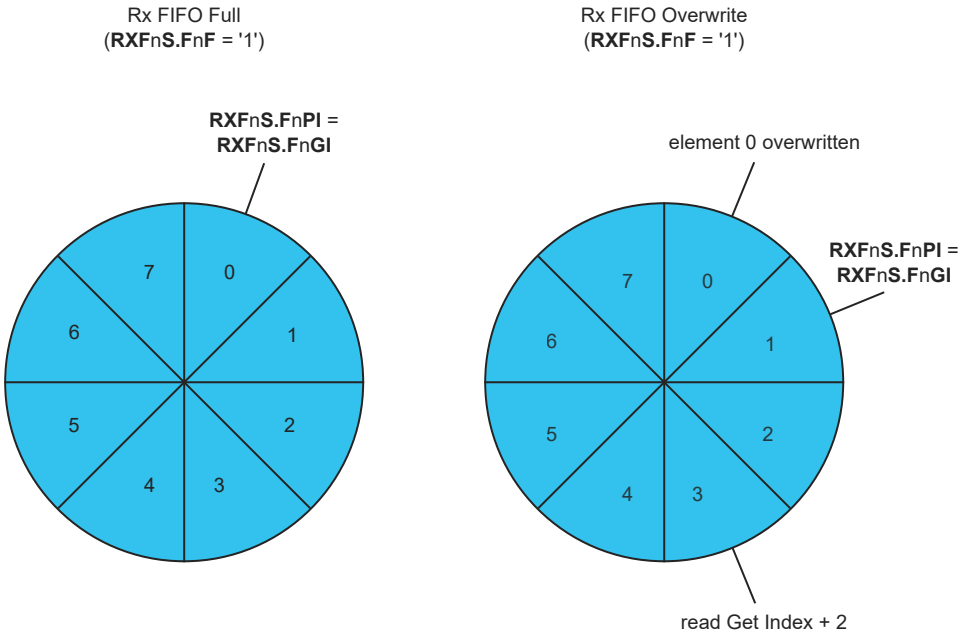
When an Rx FIFO full condition (RXFnS.FnPI = RXFnS.FnGI) is signaled by RXFnS.FnF = '1', the next message accepted for the FIFO will overwrite the oldest FIFO message. Put and get index are both incremented by one.

When an Rx FIFO is operated in overwrite mode and an Rx FIFO full condition is signaled, reading of the Rx FIFO elements should start at least at get index + 1. The reason for that is, that it might


© 2026 Microchip Technology Inc. and its subsidiaries

happen, that a received message is written to the Message RAM (put index) while the CPU is reading from the Message RAM (get index). In this case inconsistent data may be read from the respective Rx FIFO element. Adding an offset to the get index when reading from the Rx FIFO avoids this problem. The offset depends on how fast the CPU accesses the Rx FIFO. The figure below shows an offset of two with respect to the get index when reading the Rx FIFO. In this case the two messages stored in element 1 and 2 are lost.

**Figure 34-8.** Rx FIFO Overflow Handling




After reading from the Rx FIFO, the number of the last element read has to be written to the Rx FIFO Acknowledge Index RXFnA.FnA. This increments the get index to that element number. In case the put index has not been incremented to this Rx FIFO element, the Rx FIFO full condition is reset (RXFnS.FnF = '0').

### 34.6.5.3. Dedicated Rx Buffers

The CAN supports up to 64 dedicated Rx Buffers. The start address of the dedicated Rx Buffer section is configured via RXBC.RBSA.

For each Rx Buffer a Standard or Extended Message ID Filter Element with SFEC / EFEC = "111" and SFID2 / EFID2[10:9] = "00" has to be configured (see Standard Message ID Filter Element and Extended Message ID Filter Element).

After a received message has been accepted by a filter element, the message is stored into the Rx Buffer in the Message RAM referenced by the filter element. The format is the same as for an Rx FIFO element. In addition the flag IR.DRX (Message stored in Dedicated Rx Buffer) in the interrupt register is set.

**Table 34-4.** Example Filter Configuration for Rx Buffers

| Filter Element | SFID1[10:0] / EFID1[28:0] | SFID2[10:9] / EFID2[10:9] | SFID2[5:0] / EFID2[5:0] |
|---|---|---|---|
| 0 | ID message 1 | 00 | 00 0000 |
| 1 | ID message 2 | 00 | 00 0001 |
| 2 | ID message 3 | 00 | 00 0010 |

After the last word of a matching received message has been written to the Message RAM, the respective New Data flag in register NDAT1, NDAT2 is set. As long as the New Data flag is set, the


© 2026 Microchip Technology Inc. and its subsidiaries

respective Rx Buffer is locked against updates from received matching frames. The New Data flags have to be reset by the CPU by writing a '1' to the respective bit position.

While an Rx Buffer's New Data flag is set, a Message ID Filter Element referencing this specific Rx Buffer will not match, causing the acceptance filtering to continue. Following Message ID Filter Elements may cause the received message to be stored into another Rx Buffer, or into an Rx FIFO, or the message may be rejected, depending on filter configuration.

**Rx Buffer Handling**

- Reset interrupt flag IR.DRX
- Read New Data registers
- Read messages from Message RAM
- Reset New Data flags of processed messages

### 34.6.5.4. Debug on CAN Support

Debug messages are stored into Rx Buffers. For debug handling three consecutive Rx buffers (e.g. #61, #62, #63) have to be used for storage of debug messages A, B, and C. The format is the same as for an Rx Buffer or an Rx FIFO element (see Rx Buffer and FIFO Element ).

Advantage: Fixed start address for the DMA transfers (relative to RXBC.RBSA), no additional configuration required.

For filtering of debug messages Standard / Extended Filter Elements with SFEC / EFEC = "111" have to be set up. Messages matching these filter elements are stored into the Rx Buffers addressed by SFID2 / EFID2[5:0].

After message C has been stored, the DMA request output is activated and the three messages can be read from the Message RAM under DMA control. The RAM words holding the debug messages will not be changed by the CAN while DMA request is activated. The behavior is similar to that of an Rx Buffers with its New Data flag set.

After the DMA has completed the DMA unit sets the DMA acknowledge. This resets DMA request. Now the CAN is prepared to receive the next set of debug messages.

**Filtering for Debug Messages**

Filtering for debug messages is done by configuring one Standard / Extended Message ID Filter Element for each of the three debug messages. To enable a filter element to filter for debug messages SFEC / EFEC has to be programmed to "111". In this case fields SFID1 / SFID2 and EFID1 / EFID2 have a different meaning (see Standard Message ID Filter Element and Extended Message ID Filter Element). While SFID2 / EFID2[10:9] controls the debug message handling state machine, SFID2 / EFID2[5:0] controls the location for storage of a received debug message.

When a debug message is stored, neither the respective New Data flag nor IR.DRX are set. The reception of debug messages can be monitored via RXF1S.DMS.

**Table 34-5.** Example Filter Configuration for Debug Messages

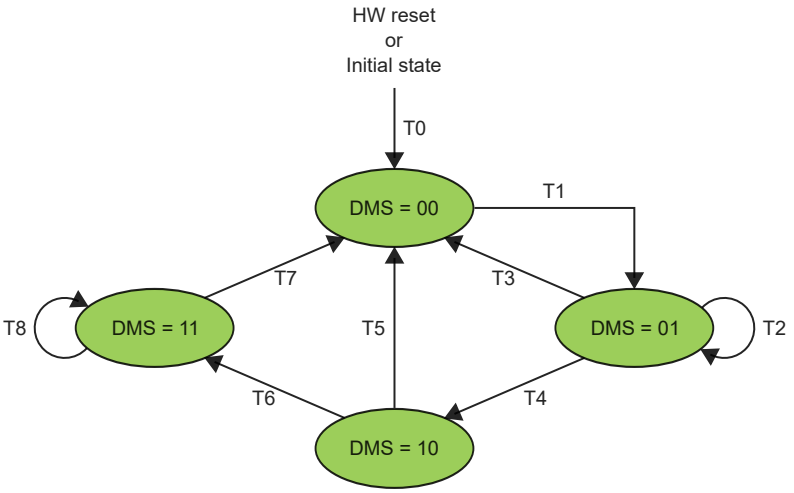
| Filter Element | SFID1[10:0] / EFID1[28:0] | SFID2[10:9] / EFID2[10:9] | SFID2[5:0] / EFID2[5:0] |
|---|---|---|---|
| 0 | ID debug message A | 01 | 11 1101 |
| 1 | ID debug message B | 10 | 11 1110 |
| 2 | ID debug message C | 11 | 11 1111 |

**Debug Message Handling**

The debug message handling state machine assures that debug messages are stored to three consecutive Rx Buffers in correct order. In case of missing messages the process is restarted. The DMA request is activated only when all three debug messages A, B, C have been received in correct order.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-9.** Debug Message Handling State Machine




T0: Reset DMA request output, enable reception of debug message A, B, and C
T1: Reception of debug message A
T2: Reception of debug message A
T3: Reception of debug message C
T4: Reception of debug message B
T5: Reception of debug message A, B
T6: Reception of debug message C
T7: DMA transfer completed
T8: Reception of debug message A, B, C (message rejected)

### 34.6.6. Tx Handling

The Tx Handler handles transmission requests for the dedicated Tx Buffers, the Tx FIFO, and the Tx Queue. It controls the transfer of transmit messages to the CAN Core, the Put and Get Indices, and the Tx Event FIFO. Up to 32 Tx Buffers can be set up for message transmission. The CAN mode for transmission (Classic CAN or CAN FD) can be configured separately for each Tx Buffer element. The Tx Buffer element is described in Tx Buffer Element. The table below describes the possible configurations for frame transmission.

**Table 34-6.** Possible Configurations for Frame Transmission

| CCCR | | Tx Buffer Element | | Frame Transmission |
|---|---|---|---|---|
| BRSE | FDOE | FDF | BRS | |
| ignored | 0 | ignored | ignored | Classic CAN |
| 0 | 1 | 0 | ignored | Classic CAN |
| 0 | 1 | 1 | ignored | FD without bit rate switching |
| 1 | 1 | 0 | ignored | Classic CAN |
| 1 | 1 | 1 | 0 | FD without bit rate switching |
| 1 | 1 | 1 | 1 | FD with bit rate switching |

Note: AUTOSAR requires at least three Tx Queue Buffers and support of transmit cancellation

The Tx Handler starts a Tx scan to check for the highest priority pending Tx request (Tx Buffer with lowest Message ID) when the Tx Buffer Request Pending register TXBRP is updated, or when a transmission has been started.

#### 34.6.6.1. Transmit Pause

The transmit pause feature is intended for use in CAN systems where the CAN message identifiers are (permanently) specified to specific values and cannot easily be changed. These message identifiers may have a higher CAN arbitration priority than other defined messages, while in a


© 2026 Microchip Technology Inc. and its subsidiaries

specific application their relative arbitration priority should be inverse. This may lead to a case where one ECU sends a burst of CAN messages that cause another ECU's CAN messages to be delayed because that other messages have a lower CAN arbitration priority.

If e.g. CAN ECU-1 has the transmit pause feature enabled and is requested by its application software to transmit four messages, it will, after the first successful message transmission, wait for two CAN bit times of bus idle before it is allowed to start the next requested message. If there are other ECUs with pending messages, those messages are started in the idle time, they would not need to arbitrate with the next message of ECU-1. After having received a message, ECU-1 is allowed to start its next transmission as soon as the received message releases the CAN bus.

The transmit pause feature is controlled by bit CCCR.TXP. If the bit is set, the CAN will, each time it has successfully transmitted a message, pause for two CAN bit times before starting the next transmission. This enables other CAN nodes in the network to transmit messages even if their messages have lower prior identifiers. Default is transmit pause disabled (CCCR.TXP = '0').

This feature looses up burst transmissions coming from a single node and it protects against "babbling idiot" scenarios where the application program erroneously requests too many transmissions.

#### 34.6.6.2. Dedicated Tx Buffers

Dedicated Tx Buffers are intended for message transmission under complete control of the CPU. Each Dedicated Tx Buffer is configured with a specific Message ID. In case that multiple Tx Buffers are configured with the same Message ID, the Tx Buffer with the lowest buffer number is transmitted first.

If the data section has been updated, a transmission is requested by an Add Request via TXBAR.ARn. The requested messages arbitrate internally with messages from an optional Tx FIFO or Tx Queue and externally with messages on the CAN bus, and are sent out according to their Message ID.

A Dedicated Tx Buffer allocates Element Size 32-bit words in the Message RAM (refer to table below). Therefore the start address of a dedicated Tx Buffer in the Message RAM is calculated by adding transmit buffer index (0...31) • Element Size to the Tx Buffer Start Address TXBC.TBSA.

**Table 34-7.** Tx Buffer / FIFO / Queue Element Size

| TXESC.TBDS[2:0] | Data Field [bytes] | Element Size [RAM words] |
|---|---|---|
| 000 | 8 | 4 |
| 001 | 12 | 5 |
| 010 | 16 | 6 |
| 011 | 20 | 7 |
| 100 | 24 | 8 |
| 101 | 32 | 10 |
| 110 | 48 | 14 |
| 111 | 64 | 18 |

#### 34.6.6.3. Tx FIFO

Tx FIFO operation is configured by programming TXBC.TFQM to '0'. Messages stored in the Tx FIFO are transmitted starting with the message referenced by the Get Index TXFQS.TFGI. After each transmission the Get Index is incremented cyclically until the Tx FIFO is empty. The Tx FIFO enables transmission of messages with the same Message ID from different Tx Buffers in the order these messages have been written to the Tx FIFO. The CAN calculates the Tx FIFO Free Level TXFQS.TFFL as difference between Get and Put Index. It indicates the number of available (free) Tx FIFO elements.

New transmit messages have to be written to the Tx FIFO starting with the Tx Buffer referenced by the Put Index TXFQS.TFQPI. An Add Request increments the Put Index to the next free Tx FIFO element. When the Put Index reaches the Get Index, Tx FIFO Full (TXFQS.TFQF = '1') is signaled. In


© 2026 Microchip Technology Inc. and its subsidiaries

this case no further messages should be written to the Tx FIFO until the next message has been transmitted and the Get Index has been incremented.

When a single message is added to the Tx FIFO, the transmission is requested by writing a '1' to the TXBAR bit related to the Tx Buffer referenced by the Tx FIFO's Put Index.

When multiple (n) messages are added to the Tx FIFO, they are written to n consecutive Tx Buffers starting with the Put Index. The transmissions are then requested via TXBAR. The Put Index is then cyclically incremented by n. The number of requested Tx buffers should not exceed the number of free Tx Buffers as indicated by the Tx FIFO Free Level.

When a transmission request for the Tx Buffer referenced by the Get Index is canceled, the Get Index is incremented to the next Tx Buffer with pending transmission request and the Tx FIFO Free Level is recalculated. When transmission cancellation is applied to any other Tx Buffer, the Get Index and the FIFO Free Level remain unchanged.

A Tx FIFO element allocates Element Size 32-bit words in the Message RAM (refer to Table 34-7). Therefore the start address of the next available (free) Tx FIFO Buffer is calculated by adding Tx FIFO/Queue Put Index TXFQS.TFQPI (0...31) • Element Size to the Tx Buffer Start Address TXBC.TBSA.

#### 34.6.6.4. Tx Queue

Tx Queue operation is configured by programming TXBC.TFQM to '1'. Messages stored in the Tx Queue are transmitted starting with the message with the lowest Message ID (highest priority). In case that multiple Queue Buffers are configured with the same Message ID, the Queue Buffer with the lowest buffer number is transmitted first.

New messages have to be written to the Tx Buffer referenced by the Put Index TXFQS.TFQPI. An Add Request cyclically increments the Put Index to the next free Tx Buffer. In case that the Tx Queue is full (TXFQS.TFQF = '1'), the Put Index is not valid and no further message should be written to the Tx Queue until at least one of the requested messages has been sent out or a pending transmission request has been canceled.

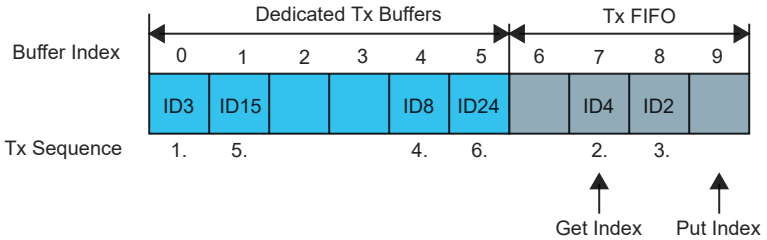
The application may use register TXBRP instead of the Put Index and may place messages to any Tx Buffer without pending transmission request.

A Tx Queue Buffer allocates Element Size 32-bit words in the Message RAM (refer to Table 34-7). Therefore the start address of the next available (free) Tx Queue Buffer is calculated by adding Tx FIFO/Queue Put Index TXFQS.TFQPI (0...31) • Element Size to the Tx Buffer Start Address TXBC.TBSA.

#### 34.6.6.5. Mixed Dedicated Tx Buffers / Tx FIFO

In this case the Tx Buffers section in the Message RAM is subdivided into a set of Dedicated Tx Buffers and a Tx FIFO. The number of Dedicated Tx Buffers is configured by TXBC.NDTB. The number of Tx Buffers assigned to the Tx FIFO is configured by TXBC.TFQS. In case TXBC.TFQS is programmed to zero, only Dedicated Tx Buffers are used.

**Figure 34-10.** Example of mixed Configuration Dedicated Tx Buffers / Tx FIFO




Tx prioritization:

- Scan Dedicated Tx Buffers and oldest pending Tx FIFO Buffer (referenced by TXFS.TFGI)

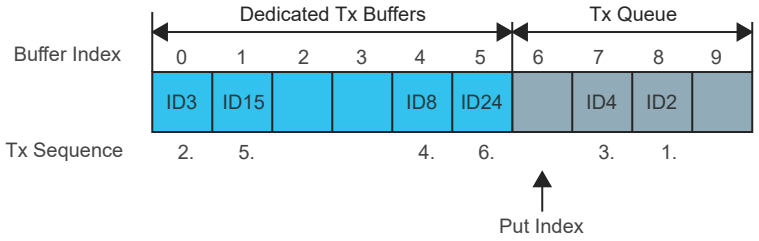

© 2026 Microchip Technology Inc. and its subsidiaries

- Buffer with lowest Message ID gets highest priority and is transmitted next

#### 34.6.6.6. Mixed Dedicated Tx Buffers / Tx Queue

In this case the Tx Buffers section in the Message RAM is subdivided into a set of Dedicated Tx Buffers and a Tx Queue. The number of Dedicated Tx Buffers is configured by TXBC.NDTB. The number of Tx Queue Buffers is configured by TXBC.TFQS. In case TXBC.TFQS is programmed to zero, only Dedicated Tx Buffers are used.

**Figure 34-11.** Example of mixed Configuration Dedicated Tx Buffers / Tx Queue




Tx prioritization:

- Scan all Tx Buffers with activated transmission request
- Tx Buffer with lowest Message ID gets highest priority and is transmitted next

#### 34.6.6.7. Transmit Cancellation

The CAN supports transmit cancellation. This feature is especially intended for gateway applications and AUTOSAR based applications. To cancel a requested transmission from a dedicated Tx Buffer or a Tx Queue Buffer the CPU has to write a '1' to the corresponding bit position (=number of Tx Buffer) of register TXBCR. Transmit cancellation is not intended for Tx FIFO operation.

Successful cancellation is signaled by setting the corresponding bit of register TXBCF to '1'.

In case a transmit cancellation is requested while a transmission from a Tx Buffer is already ongoing, the corresponding TXBRP bit remains set as long as the transmission is in progress. If the transmission was successful, the corresponding TXBTO and TXBCF bits are set. If the transmission was not successful, it is not repeated and only the corresponding TXBCF bit is set.

**Note:** In case a pending transmission is canceled immediately before this transmission could have been started, there follows a short time window where no transmission is started even if another message is also pending in this node. This may enable another node to transmit a message which may have a lower priority than the second message in this node.

#### 34.6.6.8. Tx Event Handling

To support Tx event handling the CAN has implemented a Tx Event FIFO. After the CAN has transmitted a message on the CAN bus, Message ID and timestamp are stored in a Tx Event FIFO element. To link a Tx event to a Tx Event FIFO element, the Message Marker from the transmitted Tx Buffer is copied into the Tx Event FIFO element.

The Tx Event FIFO can be configured to a maximum of 32 elements. The Tx Event FIFO element is described in Tx Event FIFO Element.

When a Tx Event FIFO full condition is signaled by IR.TEFF, no further elements are written to the Tx Event FIFO until at least one element has been read out and the Tx Event FIFO Get Index has been incremented. In case a Tx event occurs while the Tx Event FIFO is full, this event is discarded and interrupt flag IR.TEFL is set.

To avoid a Tx Event FIFO overflow, the Tx Event FIFO watermark can be used. When the Tx Event FIFO fill level reaches the Tx Event FIFO watermark configured by TXEFC.EFWM, interrupt flag IR.TEFW is set.


© 2026 Microchip Technology Inc. and its subsidiaries

When reading from the Tx Event FIFO, two times the Tx Event FIFO Get Index TXEFS.EFGI has to be added to the Tx Event FIFO start address TXEFC.EFSA.

### 34.6.7. FIFO Acknowledge Handling

The Get Indexes of Rx FIFO 0, Rx FIFO 1 and the Tx Event FIFO are controlled by writing to the corresponding FIFO Acknowledge Index (refer to RXF0A, RXF1A and TXEFA). Writing to the FIFO Acknowledge Index will set the FIFO Get Index to the FIFO Acknowledge Index plus one and thereby updates the FIFO Fill Level. There are two use cases:

When only a single element has been read from the FIFO (the one being pointed to by the Get Index), this Get Index value is written to the FIFO Acknowledge Index.

When a sequence of elements has been read from the FIFO, it is sufficient to write the FIFO Acknowledge Index only once at the end of that read sequence (value: Index of the last element read), to update the FIFO's Get Index.

Due to the fact that the CPU has free access to the CAN's Message RAM, special care has to be taken when reading FIFO elements in an arbitrary order (Get Index not considered). This might be useful when reading a High Priority Message from one of the two Rx FIFOs. In this case the FIFO's Acknowledge Index should not be written because this would set the Get Index to a wrong position and also alters the FIFO's Fill Level. In this case some of the older FIFO elements would be lost.

**Note:** The application has to ensure that a valid value is written to the FIFO Acknowledge Index. The CAN does not check for erroneous values.

### 34.6.8. Interrupts

The CAN has the following interrupt sources:

- Access to Reserved Address
- Protocol Errors (Data Phase/Arbitration Phase)
- Watchdog Interrupt
- Bus_Off Status
- Error Warning and Passive
- Error Logging Overflow
- Message RAM Bit Errors (Uncorrected/Corrected)
- Message stored to Dedicated Rx Buffer
- Timeout Occurred
- Message RAM Access Failure
- Timestamp Wraparound
- Tx Event FIFO statuses (Element Lost/Full/Watermark Reached/New Entry)
- Tx FIFO Empty
- Transmission Cancellation Finished
- Timestamp Completed
- High Priority Message
- Rx FIFO 1 Statuses (Message Lost/Full/Watermark Reached/New Message)
- Rx FIFO 0 Statuses (Message Lost/Full/Watermark Reached/New Message)

Each interrupt source has an interrupt flag associated with it. The interrupt flag register (IR) is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing '1' or disabled by writing '0' to the corresponding bit in the interrupt enable register (IE). Each interrupt flag can be assigned to one of two interrupt service lines.


© 2026 Microchip Technology Inc. and its subsidiaries

An interrupt request is generated when an interrupt flag is set, the corresponding interrupt enable is set, and the corresponding service line enable assigned to the interrupt is set. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, the service line is disabled, or the CAN is reset. Refer to IR for details on how to clear interrupt flags. All interrupt requests from the peripheral are sent to the NVIC. The user must read the IR register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**

Nested Vector Interrupt Controller

### 34.6.9. Sleep Mode Operation

The CAN can operate in IDLE0 sleep mode, but cannot operate in IDLE2 and Standby sleep modes.

The CAN has its own low-power mode that may be used at any time without disabling the CAN. It is also mandatory to allow the CAN to complete all pending transactions before entering stand-by by activating this Low-Power mode. This is performed by writing one to the Clock Stop Request bit in the CC Control register (CCCR.CSR = 1). Once all pending transactions are completed and the idle bus state is detected, the CAN will automatically set the Clock Stop Acknowledge bit (CCCR.CSA = 1). The CAN then reverts to its initial state (CCCR.INIT = 1), blocking further transfers, and it is now safe for CLK_CANx_AHB and GCLK_CANx to be switched off and the system may go to standby.

To leave Low-Power mode, CLK_CANx_AHB and GCLK_CANx must be active before writing CCCR.CSR to '0'. The CAN will acknowledge this by resetting CCCR.CSA = 0. Afterwards, the application can restart CAN communication by resetting the CCCR.INIT bit.

### 34.6.10. Synchronization

Due to the asynchronicity between the main clock domain (CLK_CAN_AHB) and the peripheral clock domain (GCLK_CAN) some registers are synchronized when written. When a write-synchronized register is written, the read back value will not be updated until the register has completed synchronization.

The following bits and registers are write-synchronized: The lNIT Initialization bit in the CC Control register (CCCR.INIT).


© 2026 Microchip Technology Inc. and its subsidiaries

## 34.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CREL | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | SUBSTEP[3:0] | | | | | | | |
| | | 31:24 | REL[3:0] | | | | STEP[3:0] | | | |
| 0x04 | ENDN | 7:0 | ETV[7:0] | | | | | | | |
| | | 15:8 | ETV[15:8] | | | | | | | |
| | | 23:16 | ETV[23:16] | | | | | | | |
| | | 31:24 | ETV[31:24] | | | | | | | |
| 0x08 | MRCFG | 7:0 | | | | | | | DQOS[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | DBTP | 7:0 | DTSEG2[3:0] | | | | DSJW[3:0] | | | |
| | | 15:8 | | | | DTSEG1[4:0] | | | | |
| | | 23:16 | TDC | | | DBRP[4:0] | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | TEST | 7:0 | RX | TX[1:0] | | LBCK | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | RWD | 7:0 | WDC[7:0] | | | | | | | |
| | | 15:8 | WDV[7:0] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | CCCR | 7:0 | TEST | DAR | MON | CSR | CSA | ASM | CCE | INIT |
| | | 15:8 | NISO | TXP | EFBI | PXHD | | | BRSE | FDOE |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | NBTP | 7:0 | | NTSEG2[6:0] | | | | | | |
| | | 15:8 | NTSEG1[7:0] | | | | | | | |
| | | 23:16 | NBRP[7:0] | | | | | | | |
| | | 31:24 | NSJW[6:0] | | | | | | | NBRP[8] |
| 0x20 | TSCC | 7:0 | | | | | | | TSS[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | TCP[3:0] | | | |
| | | 31:24 | | | | | | | | |
| 0x24 | TSCV | 7:0 | TSC[7:0] | | | | | | | |
| | | 15:8 | TSC[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x28 | TOCC | 7:0 | | | | | | TOS[1:0] | | ETOC |
| | | 15:8 | | | | | | | | |
| | | 23:16 | TOP[7:0] | | | | | | | |
| | | 31:24 | TOP[15:8] | | | | | | | |
| 0x2C | TOCV | 7:0 | TOC[7:0] | | | | | | | |
| | | 15:8 | TOC[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x30 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | ECR | 7:0 | TEC[7:0] | | | | | | | |
| | | 15:8 | RP | REC[6:0] | | | | | | |
| | | 23:16 | CEL[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x44 | PSR | 7:0 | BO | EW | EP | ACT[1:0] | | LEC[2:0] | | |
| | | 15:8 | | PXE | RFDF | RBRS | RESI | DLEC[2:0] | | |
| | | 23:16 | | TDCV[6:0] | | | | | | |
| | | 31:24 | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x48 | TDCR | 7:0 | | | | | TDCF[6:0] | | | |
| | | 15:8 | | | | | TDCO[6:0] | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x4C ... 0x4F | Reserved | | | | | | | | | |
| 0x50 | IR | 7:0 | RF1L | RF1F | RF1W | RF1N | RF0L | RF0F | RF0W | RF0N |
| | | 15:8 | TEFL | TEFF | TEFW | TEFN | TFE | TCF | TC | HPM |
| | | 23:16 | EP | ELO | BEU | BEC | DRX | TOO | MRAF | TSW |
| | | 31:24 | | | ARA | PED | PEA | WDI | BO | EW |
| 0x54 | IE | 7:0 | RF1LE | RF1FE | RF1WE | RF1NE | RF0LE | RF0FE | RF0WE | RF0NE |
| | | 15:8 | TEFLE | TEFFE | TEFWE | TEFNE | TFEE | TCFE | TCE | HPME |
| | | 23:16 | EPE | ELOE | BEUE | BECE | DRXE | TOOE | MRAFE | TSWE |
| | | 31:24 | | | ARAE | PEDE | PEAE | WDIE | BOE | EWE |
| 0x58 | ILS | 7:0 | RF1LL | RF1FL | RF1WL | RF1NL | RF0LL | RF0FL | RF0WL | RF0NL |
| | | 15:8 | TEFLL | TEFFL | TEFWL | TEFNL | TFEL | TCFL | TCL | HPML |
| | | 23:16 | EPL | ELOL | BEUL | BECL | DRXL | TOOL | MRAFL | TSWL |
| | | 31:24 | | | ARAL | PEDL | PEAL | WDIL | BOL | EWL |
| 0x5C | ILE | 7:0 | | | | | | | EINT1 | EINT0 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x60 ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | GFC | 7:0 | | | ANFS[1:0] | | ANFE[1:0] | | RRFS | RRFE |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x84 | SIDFC | 7:0 | FLSSA[7:0] | | | | | | | |
| | | 15:8 | FLSSA[15:8] | | | | | | | |
| | | 23:16 | LSS[7:0] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x88 | XIDFC | 7:0 | FLESA[7:0] | | | | | | | |
| | | 15:8 | FLESA[15:8] | | | | | | | |
| | | 23:16 | | LSE[6:0] | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x8C ... 0x8F | Reserved | | | | | | | | | |
| 0x90 | XIDAM | 7:0 | EIDM[7:0] | | | | | | | |
| | | 15:8 | EIDM[15:8] | | | | | | | |
| | | 23:16 | EIDM[23:16] | | | | | | | |
| | | 31:24 | | | | EIDM[28:24] | | | | |
| 0x94 | HPMS | 7:0 | MSI[1:0] | | BIDX[5:0] | | | | | |
| | | 15:8 | FLST | FIDX[6:0] | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x98 | NDAT1 | 7:0 | NDn[7:0] | | | | | | | |
| | | 15:8 | NDn[15:8] | | | | | | | |
| | | 23:16 | NDn[23:16] | | | | | | | |
| | | 31:24 | NDn[31:24] | | | | | | | |
| 0x9C | NDAT2 | 7:0 | NDn[7:0] | | | | | | | |
| | | 15:8 | NDn[15:8] | | | | | | | |
| | | 23:16 | NDn[23:16] | | | | | | | |
| | | 31:24 | NDn[31:24] | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0xA0 | RXF0C | 7:0 | F0SA[7:0] | | | | | | | |
| | | 15:8 | F0SA[15:8] | | | | | | | |
| | | 23:16 | | F0S[6:0] | | | | | | |
| | | 31:24 | F0OM | F0WM[6:0] | | | | | | |
| 0xA4 | RXF0S | 7:0 | | F0FL[6:0] | | | | | | |
| | | 15:8 | | | F0GI[5:0] | | | | | |
| | | 23:16 | | | F0PI[5:0] | | | | | |
| | | 31:24 | | | | | | | RF0L | F0F |
| 0xA8 | RXF0A | 7:0 | | | F0AI[5:0] | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xAC | RXBC | 7:0 | RBSA[7:0] | | | | | | | |
| | | 15:8 | RBSA[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xB0 | RXF1C | 7:0 | F1SA[7:0] | | | | | | | |
| | | 15:8 | F1SA[15:8] | | | | | | | |
| | | 23:16 | | F1S[6:0] | | | | | | |
| | | 31:24 | F1OM | F1WM[6:0] | | | | | | |
| 0xB4 | RXF1S | 7:0 | | F1FL[6:0] | | | | | | |
| | | 15:8 | | | F1GI[5:0] | | | | | |
| | | 23:16 | | | F1PI[5:0] | | | | | |
| | | 31:24 | DMS[1:0] | | | | | | RF1L | F1F |
| 0xB8 | RXF1A | 7:0 | | | F1AI[5:0] | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xBC | RXESC | 7:0 | | F1DS[2:0] | | | | F0DS[2:0] | | |
| | | 15:8 | | | | | | RBDS[2:0] | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xC0 | TXBC | 7:0 | TBSA[7:0] | | | | | | | |
| | | 15:8 | TBSA[15:8] | | | | | | | |
| | | 23:16 | | | NDTB[5:0] | | | | | |
| | | 31:24 | | TFQM | TFQS[5:0] | | | | | |
| 0xC4 | TXFQS | 7:0 | | | TFFL[5:0] | | | | | |
| | | 15:8 | | | | TFGI[4:0] | | | | |
| | | 23:16 | | | TFQF | TFQPI[4:0] | | | | |
| | | 31:24 | | | | | | | | |
| 0xC8 | TXESC | 7:0 | | | | | | TBDS[2:0] | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xCC | TXBRP | 7:0 | TRPn[7:0] | | | | | | | |
| | | 15:8 | TRPn[15:8] | | | | | | | |
| | | 23:16 | TRPn[23:16] | | | | | | | |
| | | 31:24 | TRPn[31:24] | | | | | | | |
| 0xD0 | TXBAR | 7:0 | ARn[7:0] | | | | | | | |
| | | 15:8 | ARn[15:8] | | | | | | | |
| | | 23:16 | ARn[23:16] | | | | | | | |
| | | 31:24 | ARn[31:24] | | | | | | | |
| 0xD4 | TXBCR | 7:0 | CRn[7:0] | | | | | | | |
| | | 15:8 | CRn[15:8] | | | | | | | |
| | | 23:16 | CRn[23:16] | | | | | | | |
| | | 31:24 | CRn[31:24] | | | | | | | |
| 0xD8 | TXBTO | 7:0 | TOn[7:0] | | | | | | | |
| | | 15:8 | TOn[15:8] | | | | | | | |
| | | 23:16 | TOn[23:16] | | | | | | | |
| | | 31:24 | TOn[31:24] | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0xDC | TXBCF | 7:0 | CFn[7:0] | | | | | | | |
| | | 15:8 | CFn[15:8] | | | | | | | |
| | | 23:16 | CFn[23:16] | | | | | | | |
| | | 31:24 | CFn[31:24] | | | | | | | |
| 0xE0 | TXBTIE | 7:0 | TIEn[7:0] | | | | | | | |
| | | 15:8 | TIEn[15:8] | | | | | | | |
| | | 23:16 | TIEn[23:16] | | | | | | | |
| | | 31:24 | TIEn[31:24] | | | | | | | |
| 0xE4 | TXBCIE | 7:0 | CFIEn[7:0] | | | | | | | |
| | | 15:8 | CFIEn[15:8] | | | | | | | |
| | | 23:16 | CFIEn[23:16] | | | | | | | |
| | | 31:24 | CFIEn[31:24] | | | | | | | |
| 0xE8 ... 0xEF | Reserved | | | | | | | | | |
| 0xF0 | TXEFC | 7:0 | EFSA[7:0] | | | | | | | |
| | | 15:8 | EFSA[15:8] | | | | | | | |
| | | 23:16 | | | EFS[5:0] | | | | | |
| | | 31:24 | | | EFWM[5:0] | | | | | |
| 0xF4 | TXEFS | 7:0 | | | EFFL[5:0] | | | | | |
| | | 15:8 | | | | EFGI[4:0] | | | | |
| | | 23:16 | | | | EFPI[4:0] | | | | |
| | | 31:24 | | | | | | | TEFL | EFF |
| 0xF8 | TXEFA | 7:0 | | | | EFAI[4:0] | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 34.8. Register Description

Registers are 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.1. Core Release

**Name:** CREL
**Offset:** 0x00
**Reset:** 0x32100000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | REL[3:0] | | | | STEP[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SUBSTEP[3:0] | | | | | | | |
| Access | R | R | R | R | | | | |
| Reset | 0 | 0 | 0 | 1 | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bits 31:28 – REL[3:0]** Core Release
One digit, BCD-coded.

**Bits 27:24 – STEP[3:0]** Step of Core Release
One digit, BCD-coded.

**Bits 23:20 – SUBSTEP[3:0]** Sub-step of Core Release
One digit, BCD-coded.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.2. Endian

**Name:** ENDN
**Offset:** 0x04
**Reset:** 0x87654321
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ETV[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ETV[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ETV[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ETV[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |

**Bits 31:0 – ETV[31:0]** Endianness Test Value
The endianness test value is 0x87654321


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.3. Message RAM Configuration

**Name:** MRCFG
**Offset:** 0x08
**Reset:** 0x00000002
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DQOS[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 1 | 0 |

**Bits 1:0 – DQOS[1:0]** Data Quality of Service
This field defines the memory priority access during the Message RAM read/write data operation.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Background (no sensitive operation) |
| 0x1 | LOW | Sensitive bandwidth |
| 0x2 | MEDIUM | Sensitive latency |
| 0x3 | HIGH | Critical latency |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.4. Data Bit Timing and Prescaler

**Name:** DBTP
**Offset:** 0x0C
**Reset:** 0x00000A33
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

The CAN bit time may be programmed in the range of 4 to 49 time quanta. The CAN time quantum may be programmed in the range of 1 to 32 GCLK_CAN periods. $t_q$ = (DBRP + 1) mtq.

**Note:**
With a GCLK_CAN of 8MHz, the reset value 0x00000A33 configures the CAN for a fast bit rate of 500 kBits/s.

The bit rate configured for the CAN FD data phase via DBTP must be higher or equal to the bit rate configured for the arbitration phase via NBTP.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TDC | | | DBRP[4:0] | | | | |
| Access | R/W | | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | DTSEG1[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 1 | 0 | 1 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DTSEG2[3:0] | | | | DSJW[3:0] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

**Bit 23 – TDC** Transceiver Delay Compensation

| Value | Description |
|---|---|
| 0 | Transceiver Delay Compensation disabled. |
| 1 | Transceiver Delay Compensation enabled. |

**Bits 20:16 – DBRP[4:0]** Data Baud Rate Prescaler

| Value | Description |
|---|---|
| 0x00 – 0x1F | The value by which the oscillator frequency is divided for generating the bit time quanta. The bit time is built up from a multiple of this quanta. Valid values for the Baud Rate Prescaler are 0 to 31. The actual interpretation by the hardware of this value is such that one more than the value programmed here is used. |

**Bits 12:8 – DTSEG1[4:0]** Fast time segment before sample point

| Value | Description |
|---|---|
| 0x00 – 0x1F | Valid values are 0 to 31. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. DTSEG1 is the sum of Prop_Seg and Phase_Seg1. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 7:4 – DTSEG2[3:0]** Data time segment after sample point

| Value | Description |
|---|---|
| 0x0 - 0xF | Valid values are 0 to 15. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. DTSEG2 is Phase_Seg2. |

**Bits 3:0 – DSJW[3:0]** Data (Re)Syncronization Jump Width

| Value | Description |
|---|---|
| 0x0 - 0xF | Valid values are 0 to 15. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.5. Test

**Name:** TEST
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** Read-only, Write-restricted

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RX | TX[1:0] | | LBCK | | | | |
| Access | R | R/W | R/W | R/W | | | | |
| Reset | 0 | 0 | 0 | 0 | | | | |

**Bit 7 – RX** Receive Pin
Monitors the actual value of pin CAN_RX

| Value | Description |
|---|---|
| 0 | The CAN bus is dominant (CAN_RX = 0). |
| 1 | The CAN bus is recessive (CAN_RX = 1). |

**Bits 6:5 – TX[1:0]** Control of Transmit Pin
This field defines the control of the transmit pin.

| Value | Name | Description |
|---|---|---|
| 0x0 | CORE | Reset value, CAN_TX controlled by CAN core, updated at the end of CAN bit time. |
| 0x1 | SAMPLE | Sample Point can be monitored at pin CAN_TX. |
| 0x2 | DOMINANT | Dominant (‘0’) level at pin CAN_TX. |
| 0x3 | RECESSIVE | Recessive (‘1’) level at pin CAN_TX. |

**Bit 4 – LBCK** Loop Back Mode

| Value | Description |
|---|---|
| 0 | Loop Back Mode is disabled. |
| 1 | Loop Back Mode is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.6. RAM Watchdog

**Name:** RWD
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** Read-only, Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

The RAM Watchdog monitors the READY output of the Message RAM. A Message RAM access via the CAN's AHB Host Interface starts the Message RAM Watchdog Counter with the value configured by RWD.WDC. The counter is reloaded with RWD.WDC when the Message RAM signals successful completion by activating its READY output. In case there is no response from the Message RAM until the counter has counted down to zero, the counter stops and interrupt IR.WDI is set.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WDV[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WDC[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:8 – WDV[7:0]** Watchdog Value
Actual Message RAM Watchdog Counter Value.

**Bits 7:0 – WDC[7:0]** Watchdog Configuration
Start value of the Message RAM Watchdog Counter. With the reset value of 0x00 the counter is disabled.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.7. CC Control

**Name:** CCCR
**Offset:** 0x18
**Reset:** 0x00000001
**Property:** Read-only, Write-restricted

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NISO | TXP | EFBI | PXHD | | | BRSE | FDOE |
| Access | R/W | R/W | R/W | R/W | | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TEST | DAR | MON | CSR | CSA | ASM | CCE | INIT |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bit 15 – NISO** Non ISO Operation
If this bit is set, the CAN uses the CAN FD frame format as specified by the Bosch CAN FD Specification V1.0.

| Value | Description |
|---|---|
| 0 | CAN FD frame format according to ISO 11898-1:2015 |
| 1 | CAN FD frame format according to Bosch CAN FD Specification V1.0 |

**Bit 14 – TXP** Transmit Pause
This bit field is write-restricted and only writable if bit fields CCE = 1 and INIT = 1.

| Value | Description |
|---|---|
| 0 | Transmit pause disabled. |
| 1 | Transmit pause enabled. The CAN pauses for two CAN bit times before starting the next transmission after itself has successfully transmitted a frame. |

**Bit 13 – EFBI** Edge Filtering during Bus Integration

| Value | Description |
|---|---|
| 0 | Edge filtering is disabled. |
| 1 | Two consecutive dominant tq required to detect an edge for hard synchronization. |

**Bit 12 – PXHD** Protocol Exception Handling Disable
**Note:** When protocol exception handling is disabled, the CAN will transmit an error frame when it detects a protocol exception condition.

| Value | Description |
|---|---|
| 0 | Protocol exception handling enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Protocol exception handling disabled. |

**Bit 9 – BRSE** Bit Rate Switch Enable

**Note:** When CAN FD operation is disabled FDOE = 0, BRSE is not evaluated.

| Value | Description |
|---|---|
| 0 | Bit rate switching for transmissions disabled. |
| 1 | Bit rate switching for transmissions enabled. |

**Bit 8 – FDOE** FD Operation Enable

| Value | Description |
|---|---|
| 0 | FD operation disabled. |
| 1 | FD operation enabled. |

**Bit 7 – TEST** Test Mode Enable

This bit field is write-restricted.
Writing a 0 to this field is always allowed.
Writing a 1 to this field is only allowed if bit fields CCE = 1 and INIT = 1.

| Value | Description |
|---|---|
| 0 | Normal operation. Register TEST holds reset values. |
| 1 | Test Mode, write access to register TEST enabled. |

**Bit 6 – DAR** Disable Automatic Retransmission

This bit field is write-restricted and only writable if bit fields CCE = 1 and INIT = 1.

| Value | Description |
|---|---|
| 0 | Automatic retransmission of messages not transmitted successfully enabled. |
| 1 | Automatic retransmission disabled. |

**Bit 5 – MON** Bus Monitoring Mode

This bit field is write-restricted.
Writing a 0 to this field is always allowed.
Writing a 1 to this field is only allowed if bit fields CCE = 1 and INIT = 1.

| Value | Description |
|---|---|
| 0 | Bus Monitoring Mode is disabled. |
| 1 | Bus Monitoring Mode is enabled. |

**Bit 4 – CSR** Clock Stop Request

| Value | Description |
|---|---|
| 0 | No clock stop is requested. |
| 1 | Clock stop requested. When clock stop is requested, first INIT and then CSA will be set after all pending transfer requests have been completed and the CAN bus reached idle. |

**Bit 3 – CSA** Clock Stop Acknowledge

| Value | Description |
|---|---|
| 0 | No clock stop acknowledged. |
| 1 | CAN may be set in power down by stopping CLK_CAN_AHB and GCLK_CAN. |

**Bit 2 – ASM** Restricted Operation Mode

This bit field is write-restricted.
Writing a 0 to this field is always allowed.
Writing a 1 to this field is only allowed if bit fields CCE = 1 and INIT = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Normal CAN operation. |
| 1 | Restricted Operation Mode active. |

**Bit 1 – CCE** Configuration Change Enable

This bit field is write-restricted and only writable if bit field INIT = 1.

| Value | Description |
|---|---|
| 0 | The CPU has no write access to the protected configuration registers. |
| 1 | The CPU has write access to the protected configuration registers (while CCCR.INIT = 1). |

**Bit 0 – INIT** Initialization

Due to the synchronization mechanism between the two clock domains, there may be a delay until the value written to INIT can be read back. The programmer has to assure that the previous value written to INIT has been accepted by reading INIT before setting INIT to a new value.

| Value | Description |
|---|---|
| 0 | Normal Operation. |
| 1 | Initialization is started. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.8. Nominal Bit Timing and Prescaler

**Name:** NBTP
**Offset:** 0x1C
**Reset:** 0x06000A03
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

The CAN bit time may be programmed in the range of 4 to 385 time quanta. The CAN time quantum may be programmed in the range of 1 to 512 GCLK_CAN periods. $t_q$ = (NBRP + 1) mtq.

NTSEG1 is the sum of Prop_Seg and Phase_Seg1. NTSEG2 is Phase_Seg2. Therefore the length of the bit time is (programmed values) [NTSEG1 + NTSEG2 + 3] tq or (functional values) [Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq.

The Information Processing Time (IPT) is zero, meaning the data for the next bit is available at the first clock edge after the sample point.

**Note:** With a CAN clock (GCLK_CAN) of 8 MHz, the reset value 0x06000A03 configures the CAN for a bit rate of 500 kBits/s.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NSJW[6:0] | | | | | | | NBRP[8] |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | NBRP[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NTSEG1[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | NTSEG2[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

**Bits 31:25 – NSJW[6:0]** Nominal (Re)Syncronization Jump Width

| Value | Description |
|---|---|
| 0x00 – 0x7F | Valid values are 0 to 127. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. |

**Bits 24:16 – NBRP[8:0]** Nominal Baud Rate Prescaler

| Value | Description |
|---|---|
| 0x000 – 0x1FF | The value by which the oscillator frequency is divided for generating the bit time quanta. The bit time is built up from a multiple of this quanta. Valid values for the Baud Rate Prescaler are 0 to 511. The actual interpretation by the hardware of this value is such that one more than the value programmed here is used. |

**Bits 15:8 – NTSEG1[7:0]** Nominal Time segment before sample point


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0x00 - 0x7F | Valid values are 1 to 255. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. NTSEG1 is the sum of Prop_Seg and Phase_Seg1. |

**Bits 6:0 – NTSEG2[6:0]** Time segment after sample point

| Value | Description |
|---|---|
| 0x00 - 0x7F | Valid values are 0 to 127. The actual interpretation by the hardware of this value is such that one more than the programmed value is used. NTSEG2 is Phase_Seg2. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.9. Timestamp Counter Configuration

**Name:** TSCC
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TCP[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | TSS[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 19:16 – TCP[3:0]** Timestamp Counter Prescaler

| Value | Description |
|---|---|
| 0x0 – 0xF | Configures the timestamp and timeout counters time unit in multiples of CAN bit times [1...16]. The actual interpretation by the hardware of this value is such that one more than the value programmed here is used. |

**Bits 1:0 – TSS[1:0]** Timestamp Select
This field defines the timestamp counter selection.

| Value | Name | Description |
|---|---|---|
| 0x0 or 0x3 | ZERO | Timestamp counter value always 0x0000. |
| 0x1 | INC | Timestamp counter value incremented by TCP. |
| 0x2 | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.10. Timestamp Counter Value

**Name:** TSCV
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** Read-only

**Notes:**

1. A write access to TSCV while in internal mode clears the Timestamp Counter value. A write access to TSCV while in external mode has no impact.
2. A “wrap around” is a change of the Timestamp Counter value from non-zero to zero not caused by the write access to TSCV.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TSC[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TSC[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – TSC[15:0]** Timestamp Counter
The internal Timestamp Counter value is captured on start of frame (both Rx and Tx). When TSCC.TSS = 0x1, the Timestamp Counter is incremented in multiples of CAN bit times [1...16] depending on the configuration of TSCC.TCP. A wrap around sets interrupt flag IR.TSW.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.11. Timeout Counter Configuration

**Name:** TOCC
**Offset:** 0x28
**Reset:** 0xFFFF0000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TOP[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TOP[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | TOS[1:0] | | ETOC |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 31:16 – TOP[15:0]** Timeout Period
Start value of the Timeout Counter (down-counter). Configures the Timeout Period.

**Bits 2:1 – TOS[1:0]** Timeout Select
When operating in Continuous mode, a write to TOCV presets the counter to the value configured by TOCC.TOP and continues down-counting. When the Timeout Counter is controlled by one of the FIFOs, an empty FIFO presets the counter to the value configured by TOCC.TOP. Down-counting is started when the first FIFO element is stored.

| Value | Name | Description |
|---|---|---|
| 0x0 | CONT | Continuous operation. |
| 0x1 | TXEF | Timeout controlled by TX Event FIFO. |
| 0x2 | RXF0 | Timeout controlled by Rx FIFO 0. |
| 0x3 | RXF1 | Timeout controlled by Rx FIFO 1. |

**Bit 0 – ETOC** Enable Timeout Counter

| Value | Description |
|---|---|
| 0 | Timeout Counter disabled. |
| 1 | Timeout Counter enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.12. Timeout Counter Value

**Name:** TOCV
**Offset:** 0x2C
**Reset:** 0x0000FFFF
**Property:** Read-only

**Note:** A write access to TOCV reloads the Timeout Counter with the value of TOCV.TOP.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TOC[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TOC[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 15:0 – TOC[15:0]** Timeout Counter
The Timeout Counter is decremented in multiples of CAN bit times [1...16] depending on the configuration of TSCC.TCP. When decremented to zero, interrupt flag IR.TOO is set and the Timeout Counter is stopped. Start and reset/restart conditions are configured via TOCC.TOS.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.13. Error Counter

**Name:** ECR
**Offset:** 0x40
**Reset:** 0x00000000
**Property:** Read-only

**Note:** When CCCR.ASM is set, the CAN protocol controller does not increment TECand REC when a CAN protocol error is detected, but CEL is still incremented.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CEL[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RP | REC[6:0] | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TEC[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:16 – CEL[7:0]** CAN Error Logging
The counter is incremented each time when a CAN protocol error causes the Transmit Error Counter or Receive Error Counter to be incremented. It is reset by read access to CEL. The counter stops at 0xFF; the next increment of TEC or REC sets interrupt flag IR.ELO.

**Bit 15 – RP** Receive Error Passive

**Bits 14:8 – REC[6:0]** Receive Error Counter
Actual state of the Receive Error Counter, values between 0 and 127.

**Bits 7:0 – TEC[7:0]** Transmit Error Counter
Actual state of the Transmit Error Counter, values between 0 and 255.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.14. Protocol Status

**Name:** PSR
**Offset:** 0x44
**Reset:** 0x00000707
**Property:** Read-only

**Notes:**

1. When a frame in CAN FD format has reached the data phase with BRS flag set, the next CAN event (error or valid frame) will be shown in FLEC instead of LEC. An error in a fixed stuff bit of a CAN FD CRC sequence will be shown as a Form Error, not Stuff Error.
2. The Bus_Off recovery sequence (see CAN Specification Rev. 2.0 or ISO 11898-1) cannot be shortened by setting or resetting CCCR.INIT. If the device goes Bus_Off, it will set CCCR.INIT of its own accord, stopping all bus activities. Once CCCR.INIT has been cleared by the CPU, the device will then wait for 129 occurrences of Bus Idle (129 * 11 consecutive recessive bits) before resuming normal operation. At the end of the Bus_Off recovery sequence, the Error Management Counters will be reset. During the waiting time after the resetting of CCCR.INIT, each time a sequence of 11 recessive bits has been monitored, a Bit0 Error code is written to PSR.LEC, enabling the CPU to readily check up whether the CAN bus is stuck at dominant or continuously disturbed and to monitor the Bus_Off recovery sequence. ECR.REC is used to count these sequences.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | TDCV[6:0] | | | | | | |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | PXE | RFDF | RBRS | RESI | DLEC[2:0] | | |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BO | EW | EP | ACT[1:0] | | LEC[2:0] | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

**Bits 22:16 – TDCV[6:0]** Transmitter Delay Compensation Value

| Value | Description |
|---|---|
| 0x00 – 0x7F | Position of the secondary sample point, defined by the sum of the measured delay from CAN_TX to CAN_RX and TDCR.TDCO. The SSP position is, in the data phase, the number of mtq between the start of the transmitted bit and the secondary sample point. Valid values are 0 to 127 mtq. |

**Bit 14 – PXE** Protocol Exception Event
This field is cleared on read access.

| Value | Description |
|---|---|
| 0 | No protocol exception event occurred since last read access. |
| 1 | Protocol exception event occurred. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 13 – RFDF** Received a CAN FD Message
This field is cleared on read access.

| Value | Description |
|---|---|
| 0 | Since this bit was reset by the CPU, no CAN FD message has been received. |
| 1 | Message in CAN FD format with FDF flag set has been received. This bit is set independent of acceptance filtering. |

**Bit 12 – RBRS** BRS flag of last received CAN FD Message
This field is cleared on read access.

| Value | Description |
|---|---|
| 0 | Last received CAN FD message did not have its BRS flag set. |
| 1 | Last received CAN FD message had its BRS flag set. This bit is set together with RFDF, independent of acceptance filtering. |

**Bit 11 – RESI** ESI flag of last received CAN FD Message
This field is cleared on read access.

| Value | Description |
|---|---|
| 0 | Last received CAN FD message did not have its ESI flag set. |
| 1 | Last received CAN FD message had its ESI flag set. |

**Bits 10:8 – DLEC[2:0]** Data Last Error Code
Type of last error that occurred in the data phase of a CAN FD format frame with its BRS flag set. Coding is the same as for LEC. This field will be cleared to zero when a CAN FD format frame with its BRS flag set has been transferred (reception or transmission) without error.

**Bit 7 – BO** Bus_Off Status

| Value | Description |
|---|---|
| 0 | The CAN is not Bus_Off. |
| 1 | The CAN is in Bus_Off state. |

**Bit 6 – EW** Error Warning Status

| Value | Description |
|---|---|
| 0 | Both error counters are below the Error_Warning limit of 96. |
| 1 | At least one of the error counter has reached the Error_Warning limit of 96. |

**Bit 5 – EP** Error Passive

| Value | Description |
|---|---|
| 0 | The CAN is in the Error_Active state. It normally takes part in bus communication and sends an active error flag when an error has been detected. |
| 1 | The CAN is in the Error_Passive state. |

**Bits 4:3 – ACT[1:0]** Activity
Monitors the module's CAN communication state.

| Value | Name | Description |
|---|---|---|
| 0x0 | SYNC | Node is synchronizing on CAN communication. |
| 0x1 | IDLE | Node is neither receiver nor transmitter. |
| 0x2 | RX | Node is operating as receiver. |
| 0x3 | TX | Node is operating as transmitter. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 2:0 – LEC[2:0]** Last Error Code

The LEC indicates the type of the last error to occur on the CAN bus. This field will be cleared to '0' when a message has been transferred (reception or transmission) without error.
This field is set on read access.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No Error: No error occurred since LEC has been reset by successful reception or transmission. |
| 0x1 | STUFF | Stuff Error: More than 5 equal bits in a sequence have occurred in a part of a received message where this is not allowed. |
| 0x2 | FORM | Form Error: A fixed format part of a received frame has the wrong format. |
| 0x3 | ACK | Ack Error: The message transmitted by the CAN was not acknowledged by another node. |
| 0x4 | BIT1 | Bit1 Error: During the transmission of a message (with the exception of the arbitration field), the device wanted to send a recessive level (bit of logical value '1'), but the monitored bus was dominant. |
| 0x5 | BIT0 | Bit0 Error: During the transmission of a message (or acknowledge bit, or active error flag, or overload flag), the device wanted to send a dominant level (data or identifier bit logical value '0'), but the monitored bus value was recessive. During Bus_Off recovery this status is set each time a sequence of 11 recessive bits have been monitored. This enables the CPU to monitor the proceeding of the Bus_Off recovery sequence (indicating the bus is not stuck at dominant or continuously disturbed). |
| 0x6 | CRC | CRC Error: The CRC checksum of a received message was incorrect. The CRC of an incoming message does not match with the CRC calculated from the received data. |
| 0x7 | NC | No Change: Any read access to the Protocol Status Register re-initializes the LEC to '7'. When the LEC shows the value '7', no CAN bus event was detected since the last CPU read access to the Protocol Status Register. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.15. Transmitter Delay Compensation

**Name:** TDCR
**Offset:** 0x48
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | TDCO[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | TDCF[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 14:8 – TDCO[6:0]** Transmitter Delay Compensation Offset

| Value | Description |
|---|---|
| 0x00 – 0x7F | Offset value defining the distance between the measured delay from CAN_TX to CAN_RX and the secondary sample point. Valid values are 0 to 127 mtq. |

**Bits 6:0 – TDCF[6:0]** Transmitter Delay Compensation Filter Window Length

| Value | Description |
|---|---|
| 0x00 – 0x7F | Defines the minimum value for the SSP position, dominant edges on CAN_RX that would result in an earlier SSP position are ignored for transmitter delay measurement. The feature is enabled when TDCF is configured to a value greater than TDCO. Valid values are 0 to 127 mtq. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.16. Interrupt

**Name:** IR
**Offset:** 0x50
**Reset:** 0x00000000
**Property:** -

The flags are set when one of the listed conditions is detected (edge-sensitive). A flag is cleared by writing a 1 to the corresponding bit field. Writing a 0 has no effect. A hard reset will clear the register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | ARA | PED | PEA | WDI | BO | EW |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EP | ELO | BEU | BEC | DRX | TOO | MRAF | TSW |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TEFL | TEFF | TEFW | TEFN | TFE | TCF | TC | HPM |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RF1L | RF1F | RF1W | RF1N | RF0L | RF0F | RF0W | RF0N |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 29 – ARA** Access to Reserved Address

| Value | Description |
|---|---|
| 0 | No access to reserved address occurred. |
| 1 | Access to reserved address occurred. |

**Bit 28 – PED** Protocol Error in Data Phase

| Value | Description |
|---|---|
| 0 | No protocol error in data phase. |
| 1 | Protocol error in data phase detected (PSR.DLEC != 0,7). |

**Bit 27 – PEA** Protocol Error in Arbitration Phase

| Value | Description |
|---|---|
| 0 | No protocol error in arbitration phase. |
| 1 | Protocol error in arbitration phase detected (PSR.LEC != 0,7). |

**Bit 26 – WDI** Watchdog Interrupt

| Value | Description |
|---|---|
| 0 | No Message RAM Watchdog event occurred. |
| 1 | Message RAM Watchdog event due to missing READY. |

**Bit 25 – BO** Bus_Off Status


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Bus_Off status unchanged. |
| 1 | Bus_Off status changed. |

**Bit 24 – EW** Error Warning Status

| Value | Description |
|---|---|
| 0 | Error_Warning status unchanged. |
| 1 | Error_Warning status changed. |

**Bit 23 – EP** Error Passive

| Value | Description |
|---|---|
| 0 | Error_Passive status unchanged. |
| 1 | Error_Passive status changed. |

**Bit 22 – ELO** Error Logging Overflow

| Value | Description |
|---|---|
| 0 | CAN Error Logging Counter did not overflow. |
| 1 | Overflow of CAN Error Logging Counter occurred. |

**Bit 21 – BEU** Bit Error Uncorrected

Message RAM bit error detected, uncorrected. Generated by an optional external parity / ECC logic attached to the Message RAM. An uncorrected Message RAM bit sets CCCR.INIT to 1. This is done to avoid transmission of corrupted data.

| Value | Description |
|---|---|
| 0 | Not bit error detected when reading from Message RAM. |
| 1 | Bit error detected, uncorrected (e.g. parity logic). |

**Bit 20 – BEC** Bit Error Corrected

Message RAM bit error detected and corrected. Generated by an optional external parity / ECC logic attached to the Message RAM.

| Value | Description |
|---|---|
| 0 | Not bit error detected when reading from Message RAM. |
| 1 | Bit error detected and corrected (e.g. ECC). |

**Bit 19 – DRX** Message stored to Dedicated Rx Buffer

The flag is set whenever a received message has been stored into a dedicated Rx Buffer.

| Value | Description |
|---|---|
| 0 | No Rx Buffer updated. |
| 1 | At least one received message stored into a Rx Buffer. |

**Bit 18 – TOO** Timeout Occurred

| Value | Description |
|---|---|
| 0 | No timeout. |
| 1 | Timeout reached. |

**Bit 17 – MRAF** Message RAM Access Failure

The flag is set, when the Rx Handler

- has not completed acceptance filtering or storage of an accepted message until the arbitration field of the following message has been received. In this case acceptance filtering or message storage is aborted and the Rx Handler starts processing of the following message.


© 2026 Microchip Technology Inc. and its subsidiaries

- was not able to write a message to the Message RAM. In this case message storage is aborted.

In both cases the FIFO put index is not updated resp. the New Data flag for a dedicated Rx Buffer is not set, a partly stored message is overwritten when the next message is stored to this location. The flag is also set when the Tx Handler was not able to read a message from the Message RAM in time. In this case message transmission is aborted. In case of a Tx Handler access failure the CAN is switched into Restricted Operation Mode. To leave Restricted Operation Mode, the Host CPU has to reset CCCR.ASM.

| Value | Description |
|---|---|
| 0 | No Message RAM access failure occurred. |
| 1 | Message RAM access failure occurred. |

**Bit 16 – TSW** Timestamp Wraparound

| Value | Description |
|---|---|
| 0 | No timestamp counter wrap-around. |
| 1 | Timestamp counter wrapped around. |

**Bit 15 – TEFL** Tx Event FIFO Element Lost

| Value | Description |
|---|---|
| 0 | No Tx Event FIFO element lost. |
| 1 | Tx Event FIFO element lost, also set after write attempt to Tx Event FIFO of size zero. |

**Bit 14 – TEFF** Tx Event FIFO Full

| Value | Description |
|---|---|
| 0 | Tx Event FIFO not full. |
| 1 | Tx Event FIFO full. |

**Bit 13 – TEFW** Tx Event FIFO Watermark Reached

| Value | Description |
|---|---|
| 0 | Tx Event FIFO fill level below watermark. |
| 1 | Tx Event FIFO fill level reached watermark. |

**Bit 12 – TEFN** Tx Event FIFO New Entry

| Value | Description |
|---|---|
| 0 | Tx Event FIFO unchanged. |
| 1 | Tx Handler wrote Tx Event FIFO element. |

**Bit 11 – TFE** Tx FIFO Empty

| Value | Description |
|---|---|
| 0 | Tx FIFO non-empty. |
| 1 | Tx FIFO empty. |

**Bit 10 – TCF** Transmission Cancellation Finished

| Value | Description |
|---|---|
| 0 | No transmission cancellation finished. |
| 1 | Transmission cancellation finished. |

**Bit 9 – TC** Timestamp Completed

| Value | Description |
|---|---|
| 0 | Timestamp not completed. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Timestamp completed. |

**Bit 8 – HPM** High Priority Message

| Value | Description |
|---|---|
| 0 | No high priority message received. |
| 1 | High priority message received. |

**Bit 7 – RF1L** Rx FIFO 1 Message Lost

| Value | Description |
|---|---|
| 0 | No Rx FIFO 1 message lost. |
| 1 | Rx FIFO 1 message lost. also set after write attempt to Rx FIFO 1 of size zero. |

**Bit 6 – RF1F** Rx FIFO 1 Full

| Value | Description |
|---|---|
| 0 | Rx FIFO 1 not full. |
| 1 | Rx FIFO 1 full. |

**Bit 5 – RF1W** Rx FIFO 1 Watermark Reached

| Value | Description |
|---|---|
| 0 | Rx FIFO 1 fill level below watermark. |
| 1 | Rx FIFO 1 fill level reached watermark. |

**Bit 4 – RF1N** Rx FIFO 1 New Message

| Value | Description |
|---|---|
| 0 | No new message written to Rx FIFO 1. |
| 1 | New message written to Rx FIFO 1. |

**Bit 3 – RF0L** Rx FIFO 0 Message Lost

| Value | Description |
|---|---|
| 0 | No Rx FIFO 0 message lost. |
| 1 | Rx FIFO 0 message lost. also set after write attempt to Rx FIFO 0 of size zero. |

**Bit 2 – RF0F** Rx FIFO 0 Full

| Value | Description |
|---|---|
| 0 | Rx FIFO 0 not full. |
| 1 | Rx FIFO 0 full. |

**Bit 1 – RF0W** Rx FIFO 0 Watermark Reached

| Value | Description |
|---|---|
| 0 | Rx FIFO 0 fill level below watermark. |
| 1 | Rx FIFO 0 fill level reached watermark. |

**Bit 0 – RF0N** Rx FIFO 0 New Message

| Value | Description |
|---|---|
| 0 | No new message written to Rx FIFO 0. |
| 1 | New message written to Rx FIFO 0. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.17. Interrupt Enable

**Name:** IE
**Offset:** 0x54
**Reset:** 0x00000000
**Property:** -

The settings in the Interrupt Enable register determine which status changes in the Interrupt Register will be signalled on an interrupt line.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | ARAE | PEDE | PEAE | WDIE | BOE | EWE |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EPE | ELOE | BEUE | BECE | DRXE | TOOE | MRAFE | TSWE |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TEFLE | TEFFE | TEFWE | TEFNE | TFEE | TCFE | TCE | HPME |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RF1LE | RF1FE | RF1WE | RF1NE | RF0LE | RF0FE | RF0WE | RF0NE |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 29 – ARAE** Access to Reserved Address Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 28 – PEDE** Protocol Error in Data Phase Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 27 – PEAE** Protocol Error in Arbitration Phase Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 26 – WDIE** Watchdog Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 25 – BOE** Bus_Off Status Interrupt Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 24 – EWE** Error Warning Status Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 23 – EPE** Error Passive Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 22 – ELOE** Error Logging Overflow Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 21 – BEUE** Bit Error Uncorrected Interrupt Enable.

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 20 – BECE** Bit Error Corrected Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 19 – DRXE** Message stored to Dedicated Rx Buffer Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 18 – TOOE** Timeout Occurred Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 17 – MRAFE** Message RAM Access Failure Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 16 – TSWE** Timestamp Wraparound Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 15 – TEFLE** Tx Event FIFO Event Lost Interrupt Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 14 – TEFFE** Tx Event FIFO Full Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 13 – TEFWE** Tx Event FIFO Watermark Reached Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 12 – TEFNE** Tx Event FIFO New Entry Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 11 – TFEE** Tx FIFO Empty Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 10 – TCFE** Transmission Cancellation Finished Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 9 – TCE** Transmission Completed Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 8 – HPME** High Priority Message Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 7 – RF1LE** Rx FIFO 1 Message Lost Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 6 – RF1FE** Rx FIFO 1 Full Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 5 – RF1WE** Rx FIFO 1 Watermark Reached Interrupt Enable


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 4 – RF1NE** Rx FIFO 1 New Message Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 3 – RF0LE** Rx FIFO 0 Message Lost Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 2 – RF0FE** Rx FIFO 0 Full Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 1 – RF0WE** Rx FIFO 0 Watermark Reached Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 0 – RF0NE** Rx FIFO 0 New Message Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.18. Interrupt Line Select

**Name:** ILS
**Offset:** 0x58
**Reset:** 0x00000000
**Property:** -

The Interrupt Line Select register assigns an interrupt generated by a specific interrupt flag from IR to one of the two module interrupt lines.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | ARAL | PEDL | PEAL | WDIL | BOL | EWL |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EPL | ELOL | BEUL | BECL | DRXL | TOOL | MRAFL | TSWL |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TEFLL | TEFFL | TEFWL | TEFNL | TFEL | TCFL | TCL | HPML |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RF1LL | RF1FL | RF1WL | RF1NL | RF0LL | RF0FL | RF0WL | RF0NL |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 29 – ARAL** Access to Reserved Address Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 28 – PEDL** Protocol Error in Data Phase Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 27 – PEAL** Protocol Error in Arbitration Phase Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 26 – WDIL** Watchdog Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 25 – BOL** Bus_Off Status Interrupt Line


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 24 – EWL** Error Warning Status Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 23 – EPL** Error Passive Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 22 – ELOL** Error Logging Overflow Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 21 – BEUL** Bit Error Uncorrected Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 20 – BECL** Bit Error Corrected Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 19 – DRXL** Message stored to Dedicated Rx Buffer Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 18 – TOOL** Timeout Occurred Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 17 – MRAFL** Message RAM Access Failure Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 16 – TSWL** Timestamp Wraparound Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 15 – TEFLL** Tx Event FIFO Event Lost Interrupt Line


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 14 – TEFFL** Tx Event FIFO Full Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 13 – TEFWL** Tx Event FIFO Watermark Reached Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 12 – TEFNL** Tx Event FIFO New Entry Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 11 – TFEL** Tx FIFO Empty Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 10 – TCFL** Transmission Cancellation Finished Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 9 – TCL** Transmission Completed Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 8 – HPML** High Priority Message Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 7 – RF1LL** Rx FIFO 1 Message Lost Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 6 – RF1FL** Rx FIFO 1 Full Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 5 – RF1WL** Rx FIFO 1 Watermark Reached Interrupt Line


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 4 – RF1NL** Rx FIFO 1 New Message Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 3 – RF0LL** Rx FIFO 0 Message Lost Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 2 – RF0FL** Rx FIFO 0 Full Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 1 – RF0WL** Rx FIFO 0 Watermark Reached Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Bit 0 – RF0NL** Rx FIFO 0 New Message Interrupt Line

| Value | Description |
|---|---|
| 0 | Interrupt assigned to CAN interrupt line 0. |
| 1 | Interrupt assigned to CAN interrupt line 1. |

**Note:**

1. For SAM C20/C21 devices, CAN Interrupt lines 0 and 1 are mapped to the same NVIC interrupt; therefore, the preferred configuration is ILS = 0x00000000.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.19. Interrupt Line Enable

**Name:** ILE
**Offset:** 0x5C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | EINT1 | EINT0 |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0, 1 – EINTn** Enable Interrupt Line n [n = 1,0](1)

| Value | Description |
|---|---|
| 0 | CAN interrupt line n disabled. |
| 1 | CAN interrupt line n enabled. |

**Note:**

1. Interrupt lines 0 and 1 are mapped to the same NVIC interrupt on SAM C20/C21 devices. The preferred configuration is ILE.EINT0 = 1 and ILE.EINT1 = 0.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.20. Global Filter Configuration

**Name:** GFC
**Offset:** 0x80
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | ANFS[1:0] | | ANFE[1:0] | | RRFS | RRFE |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 5:4 – ANFS[1:0]** Accept Non-matching Frames Standard
Defines how received messages with 11-bit IDs that do not match any element of the filter list are treated.

| Value | Name | Description |
|---|---|---|
| 0x0 | RXF0 | Accept in Rx FIFO 0. |
| 0x1 | RXF1 | Accept in Rx FIFO 1. |
| 0x2 or 0x3 | REJECT | Reject |

**Bits 3:2 – ANFE[1:0]** Accept Non-matching Frames Extended
Defines how received messages with 29-bit IDs that do not match any element of the filter list are treated.

| Value | Name | Description |
|---|---|---|
| 0x0 | RXF0 | Accept in Rx FIFO 0. |
| 0x1 | RXF1 | Accept in Rx FIFO 1. |
| 0x2 or 0x3 | REJECT | Reject |

**Bit 1 – RRFS** Reject Remote Frames Standard

| Value | Description |
|---|---|
| 0 | Filter remote frames with 11-bit standard IDs. |
| 1 | Reject all remote frames with 11-bit standard IDs. |

**Bit 0 – RRFE** Reject Remote Frames Extended


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Filter remote frames with 29-bit extended IDs. |
| 1 | Reject all remote frames with 29-bit extended IDS. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.21. Standard ID Filter Configuration

**Name:** SIDFC
**Offset:** 0x84
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LSS[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FLSSA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | FLSSA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:16 – LSS[7:0]** List Size Standard

| Value | Description |
|---|---|
| 0 | No standard Message ID filter. |
| 1 - 128 | Number of standard Message ID filter elements. |
| > 128 | Values greater than 128 are interpreted as 128. |

**Bits 15:0 – FLSSA[15:0]** Filter List Standard Start Address
Start address of standard Message ID filter list. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.22. Extended ID Filter Configuration

**Name:** XIDFC
**Offset:** 0x88
**Reset:** 0x00000000
**Property:** Write-restricted

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | LSE[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FLESA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | FLESA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 22:16 – LSE[6:0]** List Size Extended

| Value | Description |
|---|---|
| 0 | No extended Message ID filter. |
| 1 – 64 | Number of Extended Message ID filter elements. |
| > 64 | Values greater than 64 are interpreted as 64. |

**Bits 15:0 – FLESA[15:0]** Filter List Extended Start Address
Start address of extended Message ID filter list. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.23. Extended ID AND Mask

**Name:** XIDAM
**Offset:** 0x90
**Reset:** 0x1FFFFFFF
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | EIDM[28:24] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EIDM[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EIDM[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EIDM[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 28:0 – EIDM[28:0]** Extended ID Mask
For acceptance filtering of extended frames the Extended ID AND Mask is ANDed with the Message ID of a received frame. Intended for masking of 29-bit IDs in SAE J1939. With the reset value of all bits set to one the mask is not active.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.24. High Priority Message Status

**Name:** HPMS
**Offset:** 0x94
**Reset:** 0x00000000
**Property:** Read-only

This register is updated every time a Message ID filter element configured to generate a priority event matches. This can be used to monitor the status of incoming high priority messages and to enable fast access to these messages.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FLST | FIDX[6:0] | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | MSI[1:0] | | BIDX[5:0] | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – FLST** Filter List
Indicates the filter list of the matching filter element.

| Value | Description |
|---|---|
| 0 | Standard Filter List. |
| 1 | Extended Filter List. |

**Bits 14:8 – FIDX[6:0]** Filter Index
Index of matching filter element. Range is 0 to SIDFC.LSS - 1 (standard) or XIDFC.LSE - 1 (extended).

**Bits 7:6 – MSI[1:0]** Message Storage Indicator
This field defines the message storage information to a FIFO.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No FIFO selected. |
| 0x1 | LOST | FIFO message lost. |
| 0x2 | FIFO0 | Message stored in FIFO 0. |
| 0x3 | FIFO1 | Message stored in FIFO 1. |

**Bits 5:0 – BIDX[5:0]** Buffer Index
Index of Rx FIFO element to which the message was stored. Only valid when MSI[1] = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.25. New Data 1

**Name:** NDAT1
**Offset:** 0x98
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NDn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | NDn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NDn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | NDn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – NDn[31:0]** New Data n [n = 0..31]
The register holds the New Data flags of Rx Buffers 0 to 31. The flags are set when the respective Rx Buffer has been updated from a received frame. The flags remain set until the Host clears them. A flag is cleared by writing 1 to the corresponding bit position. Writing a 0 has no effect. A hard reset will clear the register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.26. New Data 2

**Name:** NDAT2
**Offset:** 0x9C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NDn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | NDn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NDn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | NDn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – NDn[31:0]** New Data [n = 32..63]
The register holds the New Data flags of Rx Buffers 32 to 63. The flags are set when the respective Rx Buffer has been updated from a received frame. The flags remain set until the Host clears them. A flag is cleared by writing 1 to the corresponding bit position. Writing a 0 has no effect. A hard reset will clear the register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.27. Rx FIFO 0 Configuration

**Name:** RXF0C
**Offset:** 0xA0
**Reset:** 0x00000000
**Property:** Write-restricted

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | F0OM | F0WM[6:0] | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | | F0S[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| | F0SA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | F0SA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 31 – F0OM** FIFO 0 Operation Mode
FIFO 0 can be operated in blocking or in overwrite mode.

| Value | Description |
|---|---|
| 0 | FIFO 0 blocking mode. |
| 1 | FIFO 0 overwrite mode. |

**Bits 30:24 – F0WM[6:0]** Rx FIFO 0 Watermark

| Value | Description |
|---|---|
| 0 | Watermark interrupt disabled. |
| 1 – 64 | Level for Rx FIFO 0 watermark interrupt (IR.RF0W). |
| >64 | Watermark interrupt disabled. |

**Bits 22:16 – F0S[6:0]** Rx FIFO 0 Size
The Rx FIFO 0 elements are indexed from 0 to F0S - 1.

| Value | Description |
|---|---|
| 0 | No Rx FIFO 0 |
| 1 – 64 | Number of Rx FIFO 0 elements. |
| >64 | Values greater than 64 are interpreted as 64. |

**Bits 15:0 – F0SA[15:0]** Rx FIFO 0 Start Address
Start address of Rx FIFO 0 in Message RAM. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.28. Rx FIFO 0 Status

**Name:** RXF0S
**Offset:** 0xA4
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | RF0L | F0F |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | F0PI[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | F0GI[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | F0FL[6:0] | | | | | | |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 25 – RF0L** Rx FIFO 0 Message Lost
This bit is a copy of interrupt flag IR.RF0L. When IR.RF0L is reset, this bit is also reset.
Overwriting the oldest message when RXF0C.F0OM = '1' will not set this flag.

| Value | Description |
|---|---|
| 0 | No Rx FIFO 0 message lost. |
| 1 | Rx FIFO 0 message lost, also set after write attempt to Rx FIFO 0 of size zero. |

**Bit 24 – F0F** Rx FIFO 0 Full

| Value | Description |
|---|---|
| 0 | Rx FIFO 0 not full. |
| 1 | Rx FIFO 0 full. |

**Bits 21:16 – F0PI[5:0]** Rx FIFO 0 Put Index
Rx FIFO 0 write index pointer, range 0 to 63.

**Bits 13:8 – F0GI[5:0]** Rx FIFO 0 Get Index
Rx FIFO 0 read index pointer, range 0 to 63.

**Bits 6:0 – F0FL[6:0]** Rx FIFO 0 Fill Level
Number of elements stored in Rx FIFO 0, range 0 to 64.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.29. Rx FIFO 0 Acknowledge

**Name:** RXF0A
**Offset:** 0xA8
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | F0AI[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 5:0 – F0AI[5:0]** Rx FIFO 0 Acknowledge Index
After the Host has read a message or a sequence of messages from Rx FIFO 0 it has to write the buffer index of the last element read from Rx FIFO 0 to F0AI. This will set the Rx FIFO 0 Get Index RXF0S.F0GI to F0AI + 1 and update the FIFO 0 Fill Level RXF0S.F0FL.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.30. Rx Buffer Configuration

**Name:** RXBC
**Offset:** 0xAC
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RBSA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RBSA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – RBSA[15:0]** Rx Buffer Start Address
Configures the start address of the Rx Buffers section in the Message RAM. Also used to reference debug message A,B,C. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.31. Rx FIFO 1 Configuration

**Name:** RXF1C
**Offset:** 0xB0
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | F1OM | F1WM[6:0] | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| | | F1S[6:0] | | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
| | F1SA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| | F1SA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 31 – F1OM** FIFO 1 Operation Mode
FIFO 1 can be operated in blocking or in overwrite mode.

| Value | Description |
|---|---|
| 0 | FIFO 1 blocking mode. |
| 1 | FIFO 1 overwrite mode. |

**Bits 30:24 – F1WM[6:0]** Rx FIFO 1 Watermark

| Value | Description |
|---|---|
| 0 | Watermark interrupt disabled. |
| 1 – 64 | Level for Rx FIFO 1 watermark interrupt (IR.RF1W). |
| >64 | Watermark interrupt disabled. |

**Bits 22:16 – F1S[6:0]** Rx FIFO 1 Size
The Rx FIFO 1 elements are indexed from 0 to F1S - 1.

| Value | Description |
|---|---|
| 0 | No Rx FIFO 1 |
| 1 – 64 | Number of Rx FIFO 1 elements. |
| >64 | Values greater than 64 are interpreted as 64. |

**Bits 15:0 – F1SA[15:0]** Rx FIFO 1 Start Address
Start address of Rx FIFO 1 in Message RAM. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.32. Rx FIFO 1 Status

**Name:** RXF1S
**Offset:** 0xB4
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DMS[1:0] | | | | | | RF1L | F1F |
| Access | R | R | | | | | R | R |
| Reset | 0 | 0 | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | F1PI[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | F1GI[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | F1FL[6:0] | | | | | | |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:30 – DMS[1:0]** Debug Message Status
This field defines the debug message status.

| Value | Name | Description |
|---|---|---|
| 0x0 | IDLE | Idle state, wait for reception of debug messages, DMA request is cleared. |
| 0x1 | DBGA | Debug message A received. |
| 0x2 | DBGB | Debug message A, B received. |
| 0x3 | DBGC | Debug message A, B, C received, DMA request is set. |

**Bit 25 – RF1L** Rx FIFO 1 Message Lost
This bit is a copy of interrupt flag IR.RF1L. When IR.RF1L is reset, this bit is also reset.
Overwriting the oldest message when RXF1C.F1OM = '1' will not set this flag.

| Value | Description |
|---|---|
| 0 | No Rx FIFO 1 message lost. |
| 1 | Rx FIFO 1 message lost, also set after write attempt to Rx FIFO 1 of size zero. |

**Bit 24 – F1F** Rx FIFO 1 Full

| Value | Description |
|---|---|
| 0 | Rx FIFO 1 not full. |
| 1 | Rx FIFO 1 full. |

**Bits 21:16 – F1PI[5:0]** Rx FIFO 1 Put Index
Rx FIFO 1 write index pointer, range 0 to 63.

**Bits 13:8 – F1GI[5:0]** Rx FIFO 1 Get Index
Rx FIFO 1 read index pointer, range 0 to 63.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 6:0 – F1FL[6:0]** Rx FIFO 1 Fill Level
Number of elements stored in Rx FIFO 1, range 0 to 64.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.33. Rx FIFO 1 Acknowledge

**Name:** RXF1A
**Offset:** 0xB8
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | F1AI[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 5:0 – F1AI[5:0]** Rx FIFO 1 Acknowledge Index
After the Host has read a message or a sequence of messages from Rx FIFO 1 it has to write the buffer index of the last element read from Rx FIFO 1 to F1AI. This will set the Rx FIFO 1 Get Index RXF1S.F0GI to F1AI + 1 and update the FIFO 1 Fill Level RXF1S.F1FL.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.34. Rx Buffer / FIFO Element Size Configuration

**Name:** RXESC
**Offset:** 0xBC
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

Configures the number of data bytes belonging to an Rx Buffer / Rx FIFO element. Data field sizes >8 bytes are intended for CAN FD operation only.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | RBDS[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | F1DS[2:0] | | | | F0DS[2:0] | | |
| Access | | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bits 10:8 – RBDS[2:0]** Rx Buffer Data Field Size
In case the data field size of an accepted CAN frame exceeds the data field size configured for the matching Rx Buffer, only the number of bytes as configured by RXESC are stored to the Rx Buffer element. The rest of the frame's data field is ignored.

| Value | Name | Description |
|---|---|---|
| 0x0 | DATA8 | 8 byte data field. |
| 0x1 | DATA12 | 12 byte data field. |
| 0x2 | DATA16 | 16 byte data field. |
| 0x3 | DATA20 | 20 byte data field. |
| 0x4 | DATA24 | 24 byte data field. |
| 0x5 | DATA32 | 32 byte data field. |
| 0x6 | DATA48 | 48 byte data field. |
| 0x7 | DATA64 | 64 byte data field. |

**Bits 6:4 – F1DS[2:0]** Rx FIFO 1 Data Field Size
In case the data field size of an accepted CAN frame exceeds the data field size configured for the matching Rx FIFO 1, only the number of bytes as configured by RXESC are stored to the Rx FIFO 1 element. The rest of the frame's data field is ignored.

| Value | Name | Description |
|---|---|---|
| 0x0 | DATA8 | 8 byte data field. |
| 0x1 | DATA12 | 12 byte data field. |
| 0x2 | DATA16 | 16 byte data field. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x3 | DATA20 | 20 byte data field. |
| 0x4 | DATA24 | 24 byte data field. |
| 0x5 | DATA32 | 32 byte data field. |
| 0x6 | DATA48 | 48 byte data field. |
| 0x7 | DATA64 | 64 byte data field. |

**Bits 2:0 – F0DS[2:0]** Rx FIFO 0 Data Field Size
In case the data field size of an accepted CAN frame exceeds the data field size configured for the matching Rx FIFO 0, only the number of bytes as configured by RXESC are stored to the Rx FIFO 0 element. The rest of the frame's data field is ignored.

| Value | Name | Description |
|---|---|---|
| 0x0 | DATA8 | 8 byte data field. |
| 0x1 | DATA12 | 12 byte data field. |
| 0x2 | DATA16 | 16 byte data field. |
| 0x3 | DATA20 | 20 byte data field. |
| 0x4 | DATA24 | 24 byte data field. |
| 0x5 | DATA32 | 32 byte data field. |
| 0x6 | DATA48 | 48 byte data field. |
| 0x7 | DATA64 | 64 byte data field. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.35. Tx Buffer Configuration

**Name:** TXBC
**Offset:** 0xC0
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

**Note:** Be aware that the sum of TFQS and NDTB may not be greater than 32. There is no check for erroneous configurations. The Tx Buffers section in the Message RAM starts with the dedicated Tx Buffers.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | TFQM | TFQS[5:0] | | | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | NDTB[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TBSA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TBSA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 30 – TFQM** Tx FIFO/Queue Mode

| Value | Description |
|---|---|
| 0 | Tx FIFO operation. |
| 1 | Tx Queue operation. |

**Bits 29:24 – TFQS[5:0]** Transmit FIFO/Queue Size

| Value | Description |
|---|---|
| 0 | No Tx FIFO/Queue. |
| 1 – 32 | Number of Tx Buffers used for Tx FIFO/Queue. |
| >32 | Values greater than 32 are interpreted as 32. |

**Bits 21:16 – NDTB[5:0]** Number of Dedicated Transmit Buffers

| Value | Description |
|---|---|
| 0 | No Tx FIFO/Queue. |
| 1 – 32 | Number of Tx Buffers used for Tx FIFO/Queue. |
| >32 | Values greater than 32 are interpreted as 32. |

**Bits 15:0 – TBSA[15:0]** Tx Buffers Start Address
Start address of Tx Buffers section in Message RAM. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word


© 2026 Microchip Technology Inc. and its subsidiaries

addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.36. Tx FIFO/Queue Status

**Name:** TXFQS
**Offset:** 0xC4
**Reset:** 0x00000000
**Property:** Read-only

**Note:** In case of mixed configurations where dedicated Tx Buffers are combined with a Tx FIFO or a Tx Queue, the Put and Get Indexes indicate the number of the Tx Buffer starting with the first dedicated Tx Buffers. Example: For a configuration of 12 dedicated Tx Buffers and a Tx FIFO of 20 Buffers a Put Index of 15 points to the fourth buffer of the Tx FIFO.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | TFQF | TFQPI[4:0] | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | TFGI[4:0] | | | | |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TFFL[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 21 – TFQF** Tx FIFO/Queue Full

| Value | Description |
|---|---|
| 0 | Tx FIFO/Queue not full. |
| 1 | Tx FIFO/Queue full. |

**Bits 20:16 – TFQPI[4:0]** Tx FIFO/Queue Put Index
Tx FIFO/Queue write index pointer, range 0 to 31.

**Bits 12:8 – TFGI[4:0]** Tx FIFO/Queue Get Index
Tx FIFO read index pointer, range 0 to 31. Read as zero when Tx Queue operation is configured (TXBC.TFQM = '1').

**Bits 5:0 – TFFL[5:0]** Tx FIFO Free Level
Number of consecutive free Tx FIFO elements starting from TFGI, range 0 to 32. Read as zero when Tx Queue operation is configured (TXBC.TFQM = '1').


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.37. Tx Buffer Element Size Configuration

**Name:** TXESC
**Offset:** 0xC8
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

Configures the number of data bytes belonging to a Tx Buffer element. Data field sizes >8 bytes are intended for CAN FD operation only.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | TBDS[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – TBDS[2:0]** Tx Buffer Data Field Size
In case the data length code DLC of a Tx Buffer element is configured to a value higher than the Tx Buffer data field size TXESC.TBDS, the bytes not defined by the Tx Buffer are transmitted as “0xCC” (padding bytes).

| Value | Name | Description |
|---|---|---|
| 0x0 | DATA8 | 8 byte data field. |
| 0x1 | DATA12 | 12 byte data field. |
| 0x2 | DATA16 | 16 byte data field. |
| 0x3 | DATA20 | 20 byte data field. |
| 0x4 | DATA24 | 24 byte data field. |
| 0x5 | DATA32 | 32 byte data field. |
| 0x6 | DATA48 | 48 byte data field. |
| 0x7 | DATA64 | 64 byte data field. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.38. Tx Buffer Request Pending

**Name:** TXBRP
**Offset:** 0xCC
**Reset:** 0x00000000
**Property:** Read-only

**Note:** TXBRP bits which are set while a Tx scan is in progress are not considered during this particular Tx scan. In case a cancellation is requested for such a Tx Buffer, this Add Request is canceled immediately, the corresponding TXBRP bit is reset.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TRPn[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TRPn[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TRPn[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TRPn[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – TRPn[31:0]** Transmission Request Pending
Each Tx Buffer has its own Transmission Request Pending bit.
The bits are reset after a requested transmission has completed or has been cancelled via register TXBCR.
TXBRP bits are set only for those Tx Buffers configured via TXBC. After a TXBRP bit has been set, a Tx scan is started to check for the pending Tx request with the highest priority (Tx Buffer with lowest Message ID).
A cancellation request resets the corresponding transmission request pending bit of register TXBRP. In case a transmission has already been started when a cancellation is requested, this is done at the end of the transmission, regardless whether the transmission was successful or not. The cancellation request bits are reset directly after the corresponding TXBRP bit has been reset. After a cancellation has been requested, a finished cancellation is signaled via TXBCF

- after successful transmission together with the corresponding TXBTO bit
- when the transmission has not yet been started at the point of cancellation
- when the transmission has been aborted due to lost arbitration
- when an error occurred during frame transmission

In DAR mode all transmissions are automatically canceled if they are not successful. The corresponding TXBCF bit is set for all unsuccessful transmissions.

| Value | Description |
|---|---|
| 0 | No transmission request pending. |
| 1 | Transmission request pending. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.39. Tx Buffer Add Request

**Name:** TXBAR
**Offset:** 0xD0
**Reset:** 0x00000000
**Property:** -

**Note:** If an add request is applied for a Tx Buffer with pending transmission request (corresponding TXBRP bit is already set), this add request is ignored.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ARn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | ARn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ARn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ARn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – ARn[31:0]** Add Request
Each Tx Buffer has its own Add Request bit.
Writing a '1' will set the corresponding Add Request bit; writing a '0' has no impact. This enables the Host to set transmission requests for multiple Tx Buffers with one write to TXBAR. TXBAR bits are set only for those Tx Buffers configured via TXBC. When no Tx scan is running, the bits are reset immediately, else the bits remain set until the Tx scan process has completed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.40. Tx Buffer Cancellation Request

**Name:** TXBCR
**Offset:** 0xD4
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CRn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CRn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CRn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CRn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CRn[31:0]** Cancellation Request

Each Tx Buffer has its own Cancellation Request bit.
Writing a '1' will set the corresponding Cancellation Request bit; writing a '0' has no impact. This enables the Host to set cancellation requests for multiple Tx Buffers with one write to TXBCR. TXBCR bits are set only for those Tx Buffers configured via TXBC. The bits remain set until the corresponding bit of TXBRP is reset.

| Value | Description |
|---|---|
| 0 | No cancellation pending. |
| 1 | Cancellation pending. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.41. Tx Buffer Transmission Occurred

**Name:** TXBTO
**Offset:** 0xD8
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TOn[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TOn[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TOn[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TOn[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – TOn[31:0]** Transmission Occurred
Each Tx Buffer has its own Transmission Occurred bit.
The bits are set when the corresponding TXBRP bit is cleared after a successful transmission.
The bits are reset when a new transmission is requested by writing '1' to the corresponding bit of register TXBAR.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.42. Tx Buffer Cancellation Finished

**Name:** TXBCF
**Offset:** 0xDC
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CFn[31:24] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CFn[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CFn[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CFn[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CFn[31:0]** Cancellation Finished

Each Tx Buffer has its own Cancellation Finished bit.
The bits are set when the corresponding TXBRP bit is cleared after a cancellation was requested via TXBCR. In case the corresponding TXBRP bit was not set at the point of cancellation, CF is set immediately.
The bits are reset when a new transmission is requested by writing '1' to the corresponding bit of register TXBAR.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.43. Tx Buffer Transmission Interrupt Enable

**Name:** TXBTIE
**Offset:** 0xE0
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TIEn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TIEn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TIEn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TIEn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – TIEn[31:0]** Transmission Interrupt Enable
Each Tx Buffer has its own Transmission Interrupt Enable bit.

| Value | Description |
|---|---|
| 0 | Transmission interrupt disabled. |
| 1 | Transmission interrupt enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.44. Tx Buffer Cancellation Finished Interrupt Enable

**Name:** TXBCIE
**Offset:** 0xE4
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CFIEn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CFIEn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CFIEn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CFIEn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CFIEn[31:0]** Cancellation Finished Interrupt Enable
Each Tx Buffer has its own Cancellation Finished Interrupt Enable bit.

| Value | Description |
|---|---|
| 0 | Cancellation finished interrupt disabled. |
| 1 | Cancellation finished interrupt enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.45. Tx Event FIFO Configuration

**Name:** TXEFC
**Offset:** 0xF0
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | EFWM[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | EFS[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EFSA[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EFSA[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 29:24 – EFWM[5:0]** Event FIFO Watermark

| Value | Description |
|---|---|
| 0 | Watermark interrupt disabled. |
| 1 – 32 | Level for Tx Event FIFO watermark interrupt (IR.TEFW). |
| >32 | Watermark interrupt disabled. |

**Bits 21:16 – EFS[5:0]** Event FIFO Size
The Tx Event FIFO elements are indexed from 0 to EFS - 1.

| Value | Description |
|---|---|
| 0 | Tx Event FIFO disabled |
| 1 – 32 | Number of Tx Event FIFO elements. |
| >32 | Values greater than 32 are interpreted as 32. |

**Bits 15:0 – EFSA[15:0]** Event FIFO Start Address
Start address of Tx Event FIFO in Message RAM. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.46. Tx Event FIFO Status

**Name:** TXEFS
**Offset:** 0xF4
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | TEFL | EFF |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | EFPI[4:0] | | | | |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | EFGI[4:0] | | | | |
| Access | | | | R | R | R | R | R |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | EFFL[5:0] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 25 – TEFL** Tx Event FIFO Element Lost
This bit is a copy of interrupt flag IR.TEFL. When IR.TEFL is reset, this bit is also reset.

| Value | Description |
|---|---|
| 0 | No Tx Event FIFO element lost. |
| 1 | Tx Event FIFO element lost, also set after write attempt to Tx Event FIFO of size zero. |

**Bit 24 – EFF** Event FIFO Full

| Value | Description |
|---|---|
| 0 | Tx Event FIFO not full. |
| 1 | Tx Event FIFO full. |

**Bits 20:16 – EFPI[4:0]** Event FIFO Put Index
Tx Event FIFO write index pointer, range 0 to 31.

**Bits 12:8 – EFGI[4:0]** Event FIFO Get Index
Tx Event FIFO read index pointer, range 0 to 31.

**Bits 5:0 – EFFL[5:0]** Event FIFO Fill Level
Number of elements stored in Tx Event FIFO, range 0 to 32.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.8.47. Tx Event FIFO Acknowledge

**Name:** TXEFA
**Offset:** 0xF8
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | EFAI[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 4:0 – EFAI[4:0]** Event FIFO Acknowledge Index
After the Host has read an element or a sequence of elements from the Tx Event FIFO it has to write the index of the last element read from Tx Event FIFO to EFAI. This will set the Tx Event FIFO Get Index TXEFS.EFGI to EFAI + 1 and update the FIFO 0 Fill Level TXEFS.EFFL.

## 34.9. Message RAM

For storage of Rx/Tx messages and for storage of the filter configuration a single- or dual-ported Message RAM has to be connected to the CAN module.

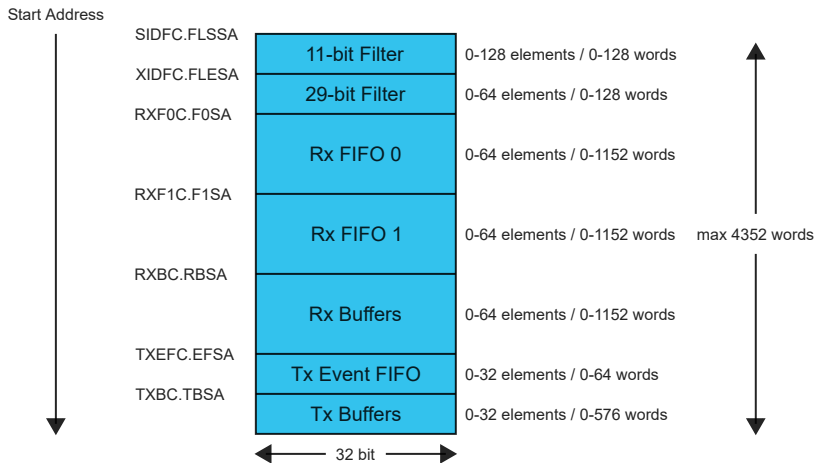
### 34.9.1. Message RAM Configuration

The Message RAM has a width of 32 bits. In case parity checking or ECC is used a respective number of bits has to be added to each word. The CAN module can be configured to allocate up to 4352 words in the Message RAM. It is not necessary to configure each of the sections listed in the figure below, nor is there any restriction with respect to the sequence of the sections.

When operated in CAN FD mode the required Message RAM size strongly depends on the element size configured for Rx FIFO 0, Rx FIFO 1, Rx Buffers, and Tx Buffers via RXESC.F0DS, RXESC.F1DS, RXESC.RBDS, and TXESC.TBDS.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 34-12.** Message RAM Configuration




When the CAN addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses (i.e. only bits 15 to 2 are evaluated and the two LSBs are ignored).

**WARNING** The CAN does not check for erroneous configuration of the Message RAM. Especially the configuration of the start addresses of the different sections and the number of elements of each section has to be done carefully to avoid falsification or loss of data.

## 34.9.2. Rx Buffer and FIFO Element

Up to 64 Rx Buffers and two Rx FIFOs can be configured in the Message RAM. Each Rx FIFO section can be configured to store up to 64 received messages. The structure of a Rx Buffer / FIFO element is shown in the table below. The element size can be configured for storage of CAN FD messages with up to 64 bytes data field via register RXESC.

**Table 34-8.** Rx Buffer and FIFO Element

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R0 | ESI | XTD | RTR | ID[28:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| R1 | ANMF | FIDX[6:0] | | | | | | | | | FDF | BRS | DLC[3:0] | | | | RXTS[15:0] | | | | | | | | | | | | | | | |
| R2 | DB3[7:0] | | | | | | | DB2[7:0] | | | | | | | | | DB1[7:0] | | | | | | | | DB0[7:0] | | | | | | | |
| R3 | DB7[7:0] | | | | | | | DB6[7:0] | | | | | | | | | DB5[7:0] | | | | | | | | DB4[7:0] | | | | | | | |
| ... | ... | | | | | | | ... | | | | | | | | | ... | | | | | | | | ... | | | | | | | |
| Rn | DBm[7:0] | | | | | | | DBm-1[7:0] | | | | | | | | | DBm-2[7:0] | | | | | | | | DBm-3[7:0] | | | | | | | |

- R0 Bit 31 - ESI: Error State Indicator
  0 : Transmitting node is error active.
  1 : Transmitting node is error passive.
- R0 Bit 30 - XTD: Extended Identifier


© 2026 Microchip Technology Inc. and its subsidiaries

Signals to the Host whether the received frame has a standard or extended identifier.

0 : 11-bit standard identifier.

1 : 29-bit extended identifier.

- R0 Bit 29 - RTR: Remote Transmission Request

  Signals to the Host whether the received frame is a data frame or a remote frame.

  0 : Received frame is a data frame.

  1 : Received frame is a remote frame.

  **Note:** There are no remote frames in CAN FD format. In case a CAN FD frame was received (EDL = '1'), bit RTR reflects the state of the reserved bit r1.

- R0 Bits 28:0 - ID[28:0]: Identifier

  Standard or extended identifier depending on bit XTD. A standard identifier is stored into ID[28:18].

- R1 Bit 31 - ANMF: Accepted Non-matching Frame

  Acceptance of non-matching frames may be enabled via GFC.ANFS and GFC.ANFE.

  0 : Received frame matching filter index FIDX.

  1 : Received frame did not match any Rx filter element.

- R1 Bits 30:24 - FIDX[6:0]: Filter Index

  0-127 : Index of matching Rx acceptance filter element (invalid if ANMF = '1').

  **Note:** Range is 0 to SIDFC.LSS-1 for standard and 0 to XIDFC.LSE-1 for extended.

- R1 Bits 23:22 - Reserved
- R1 Bit 21 - FDF: FD Format

  0 : Standard frame format.

  1 : CAN FD frame format (new DLC-coding and CRC).

- R1 Bit 20 - BRS: Bit Rate Search

  0 : Frame received without bit rate switching.

  1 : Frame received with bit rate switching.

- R1 Bits 19:16 - DLC[3:0]: Data Length Code

  0-8 : CAN + CAN FD: received frame has 0-8 data bytes.

  9-15 : CAN: received frame has 8 data bytes.

  9-15 : CAN FD: received frame has 12/16/20/24/32/48/64 data bytes.

- R1 Bits 15:0 - RXTS[15:0]: Rx Timestamp

  Timestamp Counter value captured on start of frame reception. Resolution depending on configuration of the Timestamp Counter Prescaler TSCC.TCP.

- R2 Bits 31:24 - DB3[7:0]: Data Byte 3
- R2 Bits 23:16 - DB2[7:0]: Data Byte 2
- R2 Bits 15:8 - DB1[7:0]: Data Byte 1
- R2 Bits 7:0 - DB0[7:0]: Data Byte 0
- R3 Bits 31:24 - DB7[7:0]: Data Byte 7
- R3 Bits 23:16 - DB6[7:0]: Data Byte 6
- R3 Bits 15:8 - DB5[7:0]: Data Byte 5
- R3 Bits 7:0 - DB4[7:0]: Data Byte 4

...


© 2026 Microchip Technology Inc. and its subsidiaries

- Rn Bits 31:24 - DBm[7:0]: Data Byte m
- Rn Bits 23:16 - DBm-1[7:0]: Data Byte m-1
- Rn Bits 15:8 - DBm-2[7:0]: Data Byte m-2
- Rn Bits 7:0 - DBm-3[7:0]: Data Byte m-3

**WARNING** Depending on the configuration of RXESC, between two and sixteen 32-bit words (Rn = 3 ... 17) are used for storage of a CAN message's data field.

### 34.9.3. Tx Buffer Element

The Tx Buffers section can be configured to hold dedicated Tx Buffers as well as a Tx FIFO / Tx Queue. In case that the Tx Buffers section is shared by dedicated Tx buffers and a Tx FIFO / Tx Queue, the dedicated Tx Buffers start at the beginning of the Tx Buffers section followed by the buffers assigned to the Tx FIFO or Tx Queue. The Tx Handler distinguishes between dedicated Tx Buffers and Tx FIFO / Tx Queue by evaluating the Tx Buffer configuration TXBC.TFQS and TXBC.NDTB. The element size can be configured for storage of CAN FD messages with up to 64 bytes data field via register TXESC.

**Table 34-9.** Tx Buffer Element

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T0 | ESI | XTD | RTR | ID[28:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| T1 | MM[7:0] | | | | | | | | EFC | | FDF | BRS | DLC[3:0] | | | | | | | | | | | | | | | | | | | |
| T2 | DB3[7:0] | | | | | | | | DB2[7:0] | | | | | | | | DB1[7:0] | | | | | | | | DB0[7:0] | | | | | | | |
| T3 | DB7[7:0] | | | | | | | | DB6[7:0] | | | | | | | | DB5[7:0] | | | | | | | | DB4[7:0] | | | | | | | |
| ... | ... | | | | | | | | ... | | | | | | | | ... | | | | | | | | ... | | | | | | | |
| Tn | DBm[7:0] | | | | | | | | DBm-1[7:0] | | | | | | | | DBm-2[7:0] | | | | | | | | DBm-3[7:0] | | | | | | | |

- T0 Bit 31 - ESI: Error State Indicator

  0 : ESI bit in CAN FD format depends only on error passive flag.

  1 : ESI bit in CAN FD format transmitted recessive.

  **Note:** The ESI bit of the transmit buffer is OR'ed with the error passive flag to decide the value of the ESI bit in the transmitted FD frame. As required by the CAN FD protocol specification, an error active node may optionally transmit the ESI bit recessive, but an error passive node will always transmit the ESI bit recessive.
- T0 Bit 30 - XTD: Extended Identifier

  0 : 11-bit standard identifier.

  1 : 29-bit extended identifier.
- T0 Bit 29 - RTR: Remote Transmission Request

  0 : Transmit data frame.

  1 : Transmit remote frame.

  **Note:** When RTR = '1', the CAN transmits a remote frame according to ISO 11898-1, even if CCCR.CME enables the transmission in CAN FD format.
- T0 Bits 28:0 - ID[28:0]: Identifier

  Standard or extended identifier depending on bit XTD. A standard identifier is stored into ID[28:18].


© 2026 Microchip Technology Inc. and its subsidiaries

- T1 Bits 31:24 - MM[7:0]: Message Marker
  Written by CPU during Tx Buffer configuration. Copied into Tx Event FIFO element for identification of Tx message status.
- T1 Bit 23 - EFC: Event FIFO Control
  0 : Don't store Tx events.
  1 : Store Tx events.
- T1 Bit 22 - Reserved
- TR1 Bit 21 - FDF: FD Format
  0 : Frame transmitted in Classic CAN format.
  1 : Frame transmitted in CAN FD format.
- T1 Bit 20 - BRS: Bit Rate Search
  0 : CAN FD frames transmitted without bit rate switching.
  1 : CAN FD frames transmitted with bit rate switching.
  **Note:** Bits ESI, FDF, and BRS are only evaluated when CAN FD operation is enabled CCCR.FDOE = '1'. Bit BRS is only evaluated when in addition CCCR.BRSE = '1'.
- T1 Bits 19:16 - DLC[3:0]: Data Length Code
  0-8 : CAN + CAN FD: received frame has 0-8 data bytes.
  9-15 : CAN: received frame has 8 data bytes.
  9-15 : CAN FD: received frame has 12/16/20/24/32/48/64 data bytes.
- T1 Bits 15:0 - Reserved
- T2 Bits 31:24 - DB3[7:0]: Data Byte 3
- T2 Bits 23:16 - DB2[7:0]: Data Byte 2
- T2 Bits 15:8 - DB1[7:0]: Data Byte 1
- T2 Bits 7:0 - DB0[7:0]: Data Byte 0
- T3 Bits 31:24 - DB7[7:0]: Data Byte 7
- T3 Bits 23:16 - DB6[7:0]: Data Byte 6
- T3 Bits 15:8 - DB5[7:0]: Data Byte 5
- T3 Bits 7:0 - DB4[7:0]: Data Byte 4
  ...
- Tn Bits 31:24 - DBm[7:0]: Data Byte m
- Tn Bits 23:16 - DBm-1[7:0]: Data Byte m-1
- Tn Bits 15:8 - DBm-2[7:0]: Data Byte m-2
- Tn Bits 7:0 - DBm-3[7:0]: Data Byte m-3

**Note:** Depending on the configuration of TXESC, between two and sixteen 32-bit words (Tn = 3 ... 17) are used for storage of a CAN message's data field.

### 34.9.4. Tx Event FIFO Element

Each element stores information about transmitted messages. By reading the Tx Event FIFO the Host CPU gets this information in the order the messages were transmitted. Status information about the Tx Event FIFO can be obtained from register TXEFS.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 34-10.** Tx Event FIFO Element

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| E0 | ESI | XTD | RTR | ID[28:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E1 | MM[7:0] | | | | | | | | ET[1:0] | | FDF | BRS | DLC[3:0] | | | | TXTS[15:0] | | | | | | | | | | | | | | | |

- E0 Bit 31 - ESI: Error State Indicator
  0 : Transmitting node is error active.
  1 : Transmitting node is error passive.
- E0 Bit 30 - XTD: Extended Identifier
  0 : 11-bit standard identifier.
  1 : 29-bit extended identifier.
- E0 Bit 29 - RTR: Remote Transmission Request
  0 : Received frame is a data frame.
  1 : Received frame is a remote frame.
- E0 Bits 28:0 - ID[28:0]: Identifier
  Standard or extended identifier depending on bit XTD. A standard identifier is stored into ID[28:18].
- E1 Bits 31:24 - MM[7:0]: Message Marker
  Copied from Tx Buffer into Tx Event FIFO element for identification of Tx message status.
- E1 Bits 23:22 - ET[1:0]: Event Type
  This field defines the event type.

**Table 34-11.** Event Type

| Value | Name | Description |
|---|---|---|
| 0x0 or 0x3 | RES | Reserved |
| 0x1 | TXE | Tx event |
| 0x2 | TXC | Transmission in spite of cancellation (always set for transmission in DAR mode) |

- E1 Bit 21 - FDF: FD Format
  0 : Standard frame format.
  1 : CAN FD frame format (new DLC-coding and CRC).
- E1 Bit 20 - BRS: Bit Rate Search
  0 : Frame received without bit rate switching.
  1 : Frame received with bit rate switching.
- E1 Bits 19:16 - DLC[3:0]: Data Length Code
  0-8 : CAN + CAN FD: received frame has 0-8 data bytes.
  9-15 : CAN: received frame has 8 data bytes.
  9-15 : CAN FD: received frame has 12/16/20/24/32/48/64 data bytes.
- E1 Bits 15:0 - TXTS[15:0]: Tx Timestamp
  Timestamp Counter value captured on start of frame transmission. Resolution depending on configuration of the Timestamp Counter Prescaler TSCC.TCP.


© 2026 Microchip Technology Inc. and its subsidiaries

### 34.9.5. Standard Message ID Filter Element

Up to 128 filter elements can be configured for 11-bit standard IDs. When accessing a Standard Message ID Filter element, its address is the Filter List Standard Start Address SIDFC.FLSSA plus the index of the filter element (0 ... 127).

**Table 34-12.** Standard Message ID Filter Element

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| S0 | SFT[1:0] | | SFEC[2:0] | | | SFID1[10:0] | | | | | | | | | | | | | | | | SFID2[10:0] | | | | | | | | | | |

- Bits 31:30 - SFT[1:0]: Standard Filter Type
  This field defines the standard filter type.

**Table 34-13.** Standard Filter Type

| Value | Name | Description |
|---|---|---|
| 0x0 | RANGE | Range filter from SFID1 to SFID2 (SFID2 >= SFID1) |
| 0x1 | DUAL | Dual ID filter for SFID1 or SFID2 |
| 0x2 | CLASSIC | Classic filter: SFID1 = filter, SFID2 = mask |
| 0x3 | RES | Reserved |

- Bits 29:27 - SFEC[2:0]: Standard Filter Element Configuration
  All enabled filter elements are used for acceptance filtering of standard frames. Acceptance filtering stops at the first matching enabled filter element or when the end of the filter list is reached. If SFEC = “100”, “101”, or “110” a match sets interrupt flag IR.HPM and, if enabled, an interrupt is generated. In this case register HPMS is updated with the status of the priority match.

**Table 34-14.** Standard Filter Element Configuration

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Disable filter element |
| 0x1 | STF0M | Store in Rx FIFO 0 if filter matches |
| 0x2 | STF1M | Store in Rx FIFO 1 if filter matches |
| 0x3 | REJECT | Reject ID if filter matches |
| 0x4 | PRIORITY | Set priority if filter matches. |
| 0x5 | PRIF0M | Set priority and store in FIFO 0 if filter matches. |
| 0x6 | PRIF1M | Set priority and store in FIFO 1 if filter matches. |
| 0x7 | STRXBUF | Store into Rx Buffer or as debug message, configuration of SFT[1:0] ignored. |

- Bits 26:16 - SFID1[10:0]: Standard Filter ID 1
  First ID of standard ID filter element.
  When filtering for Rx Buffers or for debug messages this field defines the ID of a standard mesage to be stored. The received identifiers must match exactly, no masking mechanism is used.
- Bits 15:11 - Reserved
- Bits 10:0 - SFID2[10:0]: Standard Filter ID 2
  This bit field has a different meaning depending on the configuration of SFEC.
  a. SFEC = “001” ... “110”: Second ID of standard ID filter element.
  b. SFEC = “111”: Filter for Rx Buffers or for debug messages.

  SFID2[10:9] decides whether the received message is stored into an Rx Buffer or treated as message A, B, or C of the debug message sequence.
  00 = Store message into an Rx Buffer


© 2026 Microchip Technology Inc. and its subsidiaries

01 = Debug Message A
10 = Debug Message B
11 = Debug Message C

SFID2[8:6] - Reserved

SFID2[5:0] defines the offset to the Rx Buffer Start Address RXBC.RBSA for storage of a matching message.

### 34.9.6. Extended Message ID Filter Element

Up to 64 filter elements can be configured for 29-bit extended IDs. When accessing an Extended Message ID Filter element, its address is the Filter List Extended Start Address XIDFC.FLESA plus two times the index of the filter element (0...63).

**Table 34-15.** Extended Message ID Filter Element

| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| F0 | EFEC [2:0] | | | EFID1[28:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| F1 | EFT [1:0] | | | EFID2[28:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- F0 Bits 31:29 - EFEC[2:0]: Extended Filter Element Configuration

  All enabled filter elements are used for acceptance filtering of extended frames. Acceptance filtering stops at the first matching enabled filter element or when the end of the filter list is reached. If EFEC = “100”, “101”, or “110” a match sets interrupt flag IR.HPM and, if enabled, an interrupt is generated. In this case register HPMS is updated with the status of the priority match.

**Table 34-16.** Extended Filter Element Configuration

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Disable filter element. |
| 0x1 | STF0M | Store in Rx FIFO 0 if filter matches. |
| 0x2 | STF1M | Store in Rx FIFO 1 if filter matches. |
| 0x3 | REJECT | Reject ID if filter matches. |
| 0x4 | PRIORITY | Set priority if filter matches. |
| 0x5 | PRIF0M | Set priority and store in FIFO 0 if filter matches. |
| 0x6 | PRIF1M | Set priority and store in FIFO 1 if filter matches. |
| 0x7 | STRXBUF | Store into Rx Buffer or as debug message, configuration of EFT[1:0] ignored. |

- F0 Bits 28:0 - EFID1[28:0]: Extended Filter ID 1

  First ID of extended ID filter element.

  When filtering for Rx Buffers or for debug messages this field defines the ID of a extended mesage to be stored. The received identifiers must match exactly, only XIDAM masking mechanism is used.

- F1 Bits 31:30 - EFT[1:0]: Extended Filter Type

  This field defines the extended filter type.

**Table 34-17.** Extended Filter Type

| Value | Name | Description |
|---|---|---|
| 0x0 | RANGEM | Range filter from EFID1 to EFID2 (EFID2 >= EFID1). |
| 0x1 | DUAL | Dual ID filter for EFID1 or EFID2. |
| 0x2 | CLASSIC | Classic filter: EFID1 = filter, EFID2 = mask. |
| 0x3 | RANGE | Range filter from EFID1 to EFID2 (EFID2 >= EFID1), XIDAM mask not applied. |

- F1 Bits 28:0 - EFID2[28:0]: Extended Filter ID 2


© 2026 Microchip Technology Inc. and its subsidiaries

This bit field has a different meaning depending on the configuration of EFEC.
1) EFEC = “001” ... “110” Second ID of standard ID filter element.
2) EFEC = “111” Filter for Rx Buffers or for debug messages.

EFID2[10:9] decides whether the received message is stored into an Rx Buffer or treated as message A, B, or C of the debug message sequence.

00 = Store message into an Rx Buffer

01 = Debug Message A

10 = Debug Message B

11 = Debug Message C

EFID2[8:6] - Reserved

EFID2[5:0] defines the offset to the Rx Buffer Start Address RXBC.RBSA for storage of a matching message.


© 2026 Microchip Technology Inc. and its subsidiaries

# 35. Timer/Counter (TC)

## 35.1. Overview

There are up to eight TC peripheral instances.

Each TC consists of a counter, a prescaler, compare/capture channels and control logic. The counter can be set to count events, or clock pulses. The counter, together with the compare/capture channels, can be configured to timestamp input events or IO pin edges, allowing for capturing of frequency and/or pulse width.

A TC can also perform waveform generation, such as frequency generation and pulse-width modulation.

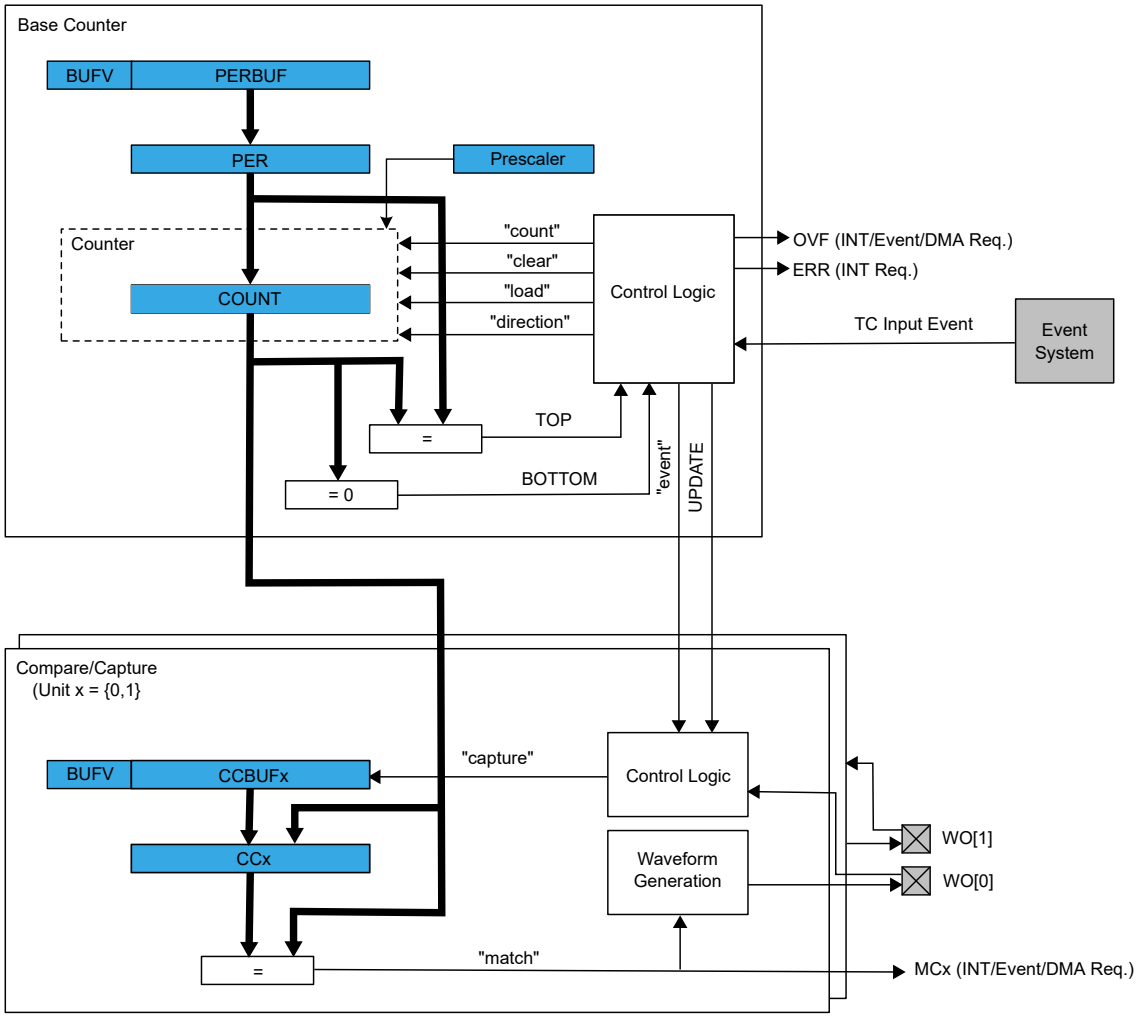
## 35.2. Features

- Selectable configuration
  - 8-, 16- or 32-bit TC operation, with compare/capture channels
- 2 compare/capture channels (CC) with:
  - Double buffered timer period setting (in 8-bit mode only)
  - Double buffered compare channel
- Waveform generation
  - Frequency generation
  - Single-slope pulse-width modulation
- Input capture
  - Event / IO pin edge capture
  - Frequency capture
  - Pulse-width capture
  - Time-stamp capture
  - Minimum and maximum capture (only available on SAM C20/C21 N variants)
- One input event
- Interrupts/output events on:
  - Counter overflow/underflow
  - Compare match or capture
- Internal prescaler
- DMA support


© 2026 Microchip Technology Inc. and its subsidiaries

## 35.3. Block Diagram

**Figure 35-1.** Timer/Counter Block Diagram




## 35.4. Signal Description

**Table 35-1.** Signal Description for TC.

| Signal Name | Type | Description |
|---|---|---|
| WO[1:0] | Digital output | Waveform output |
| | Digital input | Capture input |

Refer to *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**

I/O Multiplexing and Considerations


© 2026 Microchip Technology Inc. and its subsidiaries

## 35.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 35.5.1. I/O Lines

In order to use the I/O lines of this peripheral, the I/O pins must be configured using the I/O Pin Controller (PORT).

**Related Links**

PORT - I/O Pin Controller

### 35.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

**Related Links**

PM - Power Manager

### 35.5.3. Clocks

The TC bus clocks (CLK_TCx_APB) can be enabled and disabled in the Main Clock Module. The default state of CLK_TCx_APB can be found in the *Peripheral Clock Masking*.

The generic clocks (GCLK_TCx) are asynchronous to the user interface clock (CLK_TCx_APB). Due to this asynchronicity, accessing certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Note:** Two instances of the TC may share a peripheral clock channel. In this case, they cannot be set to different clock frequencies. Refer to the peripheral clock channel mapping of the Generic Clock Controller (GCLK.PCHTRLm) to identify shared peripheral clocks.

**Related Links**

PCHCTRLm

Peripheral Channel Control

Peripheral Clock Masking

### 35.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

**Related Links**

DMAC – Direct Memory Access Controller

### 35.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

Nested Vector Interrupt Controller

### 35.5.6. Events

The events of this peripheral are connected to the Event System.

**Related Links**

Event System (EVSYS)


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will halt normal operation. This peripheral can be forced to continue operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

**Related Links**

DBGCTRL

Debug Control

### 35.5.8. Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except for the following:

- Interrupt Flag Status and Clear register (INTFLAG)
- Status register (STATUS)
- Count register (COUNT)
- Period and Period Buffer registers (PER, PERBUF)
- Compare/Capture Value registers and Compare/Capture Value Buffer registers (CCx, CCBUFx)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 35.5.9. Analog Connections

Not applicable.

## 35.6. Functional Description

### 35.6.1. Principle of Operation

The following definitions are used throughout the documentation:

**Table 35-2.** Timer/Counter Definitions

| Name | Description |
|---|---|
| TOP | The counter reaches TOP when it becomes equal to the highest value in the count sequence. The TOP value can be the same as Period (PER) or the Compare Channel 0 (CC0) register value depending on the waveform generator mode in Waveform Output Operations. |
| ZERO | The counter is ZERO when it contains all zeroes |
| MAX | The counter reaches MAX when it contains all ones |
| UPDATE | The timer/counter signals an update when it reaches ZERO or TOP, depending on the direction settings. |
| Timer | The timer/counter clock control is handled by an internal source |
| Counter | The clock control is handled externally (e.g. counting external events) |
| CC | For compare operations, the CC are referred to as “compare channels”<br>For capture operations, the CC are referred to as “capture channels.” |

Each TC instance has up to two compare/capture channels (CC0 and CC1).

The counter in the TC can either count events from the Event System, or clock ticks of the GCLK_TCx clock, which may be divided by the prescaler.

The counter value is passed to the CCx where it can be either compared to user-defined values or captured.


© 2026 Microchip Technology Inc. and its subsidiaries

For optimized timing the CCx and CCBUFx registers share a common resource. When writing into CCBUFx, lock the access to the corresponding CCx register (SYNCBUSY.CCX = 1) till the CCBUFx register value is not loaded into the CCx register (BUFVx == 1). Each buffer register has a buffer valid (BUFV) flag that indicates when the buffer contains a new value.

The Counter register (COUNT) and the Compare and Capture registers with buffers (CCx and CCBUFx) can be configured as 8-, 16- or 32-bit registers, with according MAX values. Mode settings (CTRLA.MODE) determine the maximum range of the Counter register.

In 8-bit mode, a Period Value (PER) register and its Period Buffer Value (PERBUF) register are also available. The counter range and the operating frequency determine the maximum time resolution achievable with the TC peripheral.

The TC can be set to count up or down. Under normal operation, the counter value is continuously compared to the TOP or ZERO value to determine whether the counter has reached that value. On a comparison match the TC can request DMA transactions, or generate interrupts or events for the Event System.

In compare operation, the counter value is continuously compared to the values in the CCx registers. In case of a match the TC can request DMA transactions, or generate interrupts or events for the Event System. In waveform generator mode, these comparisons are used to set the waveform period or pulse width.

Capture operation can be enabled to perform input signal period and pulse width measurements, or to capture selectable edges from an IO pin or internal event from Event System.

## 35.6.2. Basic Operation

### 35.6.2.1. Initialization

The following registers are enable-protected, meaning that they can only be written when the TC is disabled (CTRLA.ENABLE =0):

- Control A register (CTRLA), except the Enable (ENABLE) and Software Reset (SWRST) bits
- Drive Control register (DRVCTRL)
- Wave register (WAVE)
- Event Control register (EVCTRL)

Writing to Enable-Protected bits and setting the CTRLA.ENABLE bit can be performed in a single 32-bit access of the CTRLA register. Writing to Enable-Protected bits and clearing the CTRLA.ENABLE bit cannot be performed in a single 32-bit access.

Before enabling the TC, the peripheral must be configured by the following steps:

1. Enable the TC bus clock (CLK_TCx_APB).
2. Select 8-, 16- or 32-bit counter mode via the TC Mode bit group in the Control A register (CTRLA.MODE). The default mode is 16-bit.
3. Select one wave generation operation in the Waveform Generation Operation bit group in the WAVE register (WAVE.WAVEGEN).
4. If desired, the GCLK_TCx clock can be prescaled via the Prescaler bit group in the Control A register (CTRLA.PRESCALER).
   - If the prescaler is used, select a prescaler synchronization operation via the Prescaler and Counter Synchronization bit group in the Control A register (CTRLA.PRESYNC).
5. If desired, select one-shot operation by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT).
6. If desired, configure the counting direction 'down' (starting from the TOP value) by writing a '1' to the Counter Direction bit in the Control B register (CTRLBSET.DIR).


© 2026 Microchip Technology Inc. and its subsidiaries

7. For capture operation, enable the individual channels to capture in the Capture Channel x Enable bit group in the Control A register (CTRLA.CAPTEN).
8. If desired, enable inversion of the waveform output or IO pin input signal for individual channels via the Invert Enable bit group in the Drive Control register (DRVCTRL.INVEN).

### 35.6.2.2. Enabling, Disabling, and Resetting

The TC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The TC is disabled by writing a zero to CTRLA.ENABLE.

The TC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TC, except DBGCTRL, will be reset to their initial state. Refer to the CTRLA register for details.

The TC should be disabled before the TC is reset in order to avoid undefined behavior.

### 35.6.2.3. Prescaler Selection

The GCLK_TCx is fed into the internal prescaler.

The prescaler consists of a counter that counts up to the selected prescaler value, whereupon the output of the prescaler toggles.

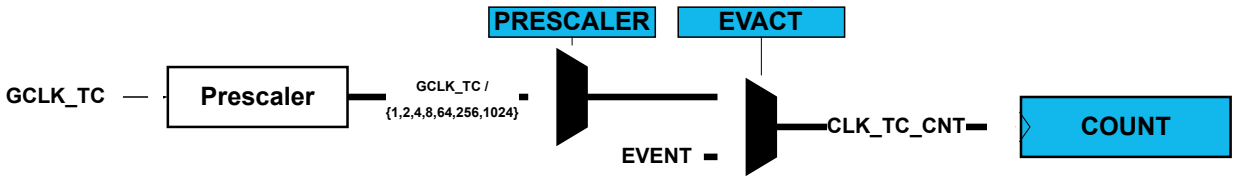
If the prescaler value is higher than one, the counter update condition can be optionally executed on the next GCLK_TCx clock pulse or the next prescaled clock pulse. For further details, refer to Prescaler (CTRLA.PRESCALER) and Counter Synchronization (CTRLA.PRESYNC) description.

Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).

**Note:** When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TC_CNT.

**Figure 35-2.** Prescaler




### 35.6.2.4. Counter Mode

The counter mode is selected by the Mode bit group in the Control A register (CTRLA.MODE). By default, the counter is enabled in the 16-bit counter resolution. Three counter resolutions are available:

- **COUNT8:** The 8-bit TC has its own Period Value and Period Buffer Value registers (PER and PERBUF).
- **COUNT16:** 16-bit is the default counter mode. There is no dedicated period register in this mode.
- **COUNT32:** 32-bit mode is achieved by pairing two 16-bit TC peripherals. TC0 is paired with TC1, TC2 is paired with TC3. TC4, TC5, TC6 and TC7 cannot be paired.
  When paired, the TC peripherals are configured using the registers of the even-numbered TC. The TC bus clocks (CLK_TCx_APB) for both Host and Client TCs need to be enabled.

  The odd-numbered partner will act as a Client, and the Client bit in the Status register (STATUS.Client) will be set. The register values of a Client will not reflect the registers of the 32-bit counter. Writing to any of the Client registers will not affect the 32-bit counter. Normal access to the Client COUNT and CCx registers is not allowed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.6.2.5. Counter Operations

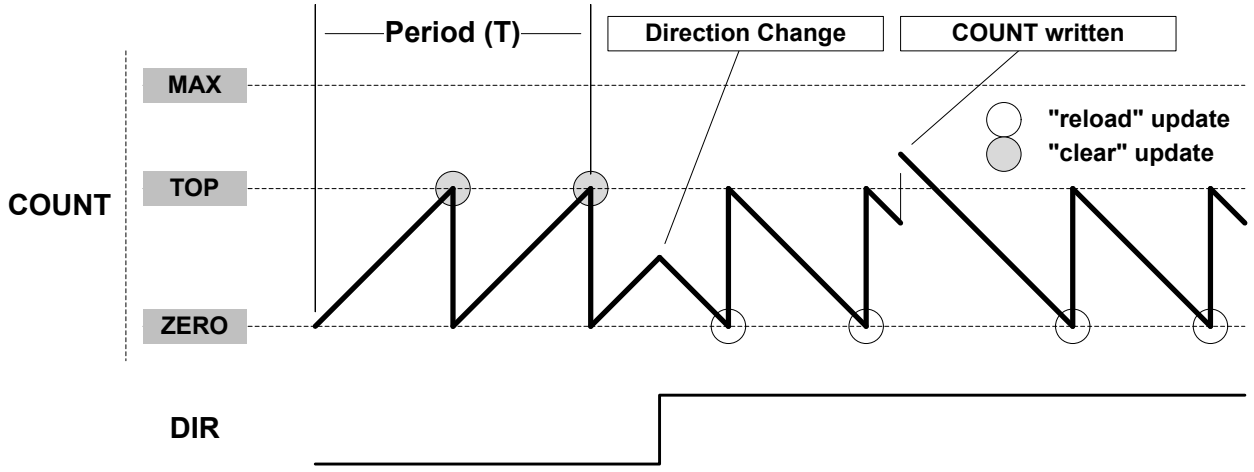
Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TC clock input (CLK_TC_CNT). A counter clear or reload marks the end of the current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If this bit is zero the counter is counting up, and counting down if CTRLB.DIR=1. The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it is counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When it is counting down, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

INTFLAG.OVF can be used to trigger an interrupt, a DMA request, or an event. An overflow/underflow occurrence (i.e., a compare match with TOP/ZERO) will stop counting if the One-Shot bit in the Control B register is set (CTRLBSET.ONESHOT).

It is possible to change the counter value (by writing directly in the COUNT register) even when the counter is running. When starting the TC, the COUNT value will be either ZERO or TOP (depending on the counting direction set by CTRLBSET.DIR or CTRLBCLR.DIR), unless a different value has been written to it, or the TC has been stopped at a value other than ZERO. The write access has higher priority than count, clear, or reload. The direction of the counter can also be changed when the counter is running. See also the following figure.

**Figure 35-3.** Counter Operation




Due to asynchronous clock domains, the internal counter settings are written when the synchronization is complete. Normal operation must be used when using the counter as timer base for the capture channels.

#### 35.6.2.5.1. Stop Command and Event Action

A Stop command can be issued from software by using Command bits in the Control B Set register (CTRLBSET.CMD = 0x2, STOP). When a Stop is detected while the counter is running, the counter will retain its current value. All waveforms are cleared and the Stop bit in the Status register is set (STATUS.STOP).

#### 35.6.2.5.2. Re-Trigger Command and Event Action

A re-trigger command can be issued from software by writing the Command bits in the Control B Set register (CTRLBSET.CMD = 0x1, RETRIGGER), or from event when a re-trigger event action is configured in the Event Control register (EVCTRL.EVACT = 0x1, RETRIGGER).

When the command is detected during counting operation, the counter will be reloaded or cleared, depending on the counting direction (CTRLBSET.DIR or CTRLBCLR.DIR). When the re-trigger


© 2026 Microchip Technology Inc. and its subsidiaries

command is detected while the counter is stopped, the counter will resume counting from the current value in the COUNT register.

**Note:** When a re-trigger event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACT=0x1, RETRIGGER), enabling the counter will not start the counter. The counter will start on the next incoming event and restart on corresponding following event.

#### 35.6.2.5.3. Count Event Action

The TC can count events. When an event is received, the counter increases or decreases the value, depending on direction settings (CTRLBSET.DIR or CTRLBCLR.DIR). The count event action can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT=0x2, COUNT).

**Note:** If this operation mode is selected, PWM generation is not supported.

#### 35.6.2.5.4. Start Event Action

The TC can start counting operation on an event when previously stopped. In this configuration, the event has no effect if the counter is already counting. When the peripheral is enabled, the counter operation starts when the event is received or when a re-trigger software command is applied.

The Start TC on Event action can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT=0x3, START).

### 35.6.2.6. Compare Operations

By default, the Compare/Capture channel is configured for compare operations.

When using the TC and the Compare/Capture Value registers (CCx) for compare operations, the counter value is continuously compared to the values in the CCx registers. This can be used for timer or for waveform operation.

The Channel x Compare Buffer (CCBUFx) registers provide double buffer capability. The double buffering synchronizes the update of the CCx register with the buffer value at the UPDATE condition or a forced update command (CTRLBSET.CMD=UPDATE). For further details, refer to Double Buffering. The synchronization prevents the occurrence of odd-length, non-symmetrical pulses and ensures glitch-free output.

#### 35.6.2.6.1. Waveform Output Operations

The compare channels can be used for waveform generation on output port pins. To make the waveform available on the connected pin, the following requirements must be fulfilled:

1. Choose a Waveform Generation mode in the Waveform Generation Operation bit in Waveform register (WAVE.WAVEGEN).
2. Optionally invert the waveform output WO[x] by writing the corresponding Output Waveform x Invert Enable bit in the Driver Control register (DRVCTRL.INVENx).
3. Configure the pins with the I/O Pin Controller. Refer to *PORT - I/O Pin Controller* for details.
   **Note:** Event must not be used when the compare channel is set in waveform output operating mode.

The counter value is continuously compared with each CCx value. On a comparison match, the Match or Capture Channel x bit in the Interrupt Flag Status and Clear register (INTFLAG.MCx) will be set on the next zero-to-one transition of CLK_TC_CNT (see Normal Frequency Operation). An interrupt/and or event can be generated on comparison match if enabled. The same condition generates a DMA request.

There are four waveform configurations for the Waveform Generation Operation bit group in the Waveform register (WAVE.WAVEGEN). This will influence how the waveform is generated and impose restrictions on the top value. The configurations are:

- Normal frequency (NFRQ)
- Match frequency (MFRQ)
- Normal pulse-width modulation (NPWM)


© 2026 Microchip Technology Inc. and its subsidiaries

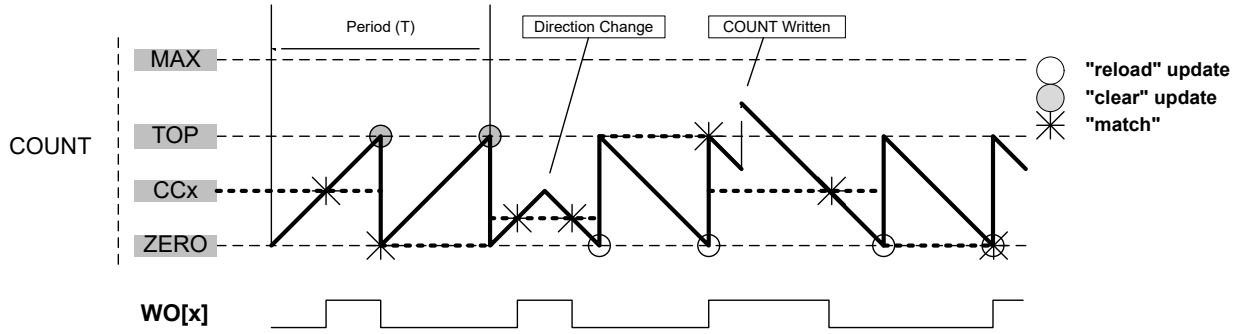
- Match pulse-width modulation (MPWM)

When using NPWM or NFRQ configuration, the TOP will be determined by the counter resolution. In 8-bit Counter mode, the Period register (PER) is used as TOP, and the TOP can be changed by writing to the PER register. In 16- and 32-bit Counter mode, TOP is fixed to the maximum (MAX) value of the counter.

### Normal Frequency Generation (NFRQ)

For Normal Frequency Generation, the period time (T) is controlled by the period register (PER) for 8-bit Counter mode and MAX for 16- and 32-bit mode. The waveform generation output (WO[x]) is toggled on each compare match between COUNT and CCx, and the corresponding Match or Capture Channel x Interrupt Flag (INTFLAG.MCx) will be set.

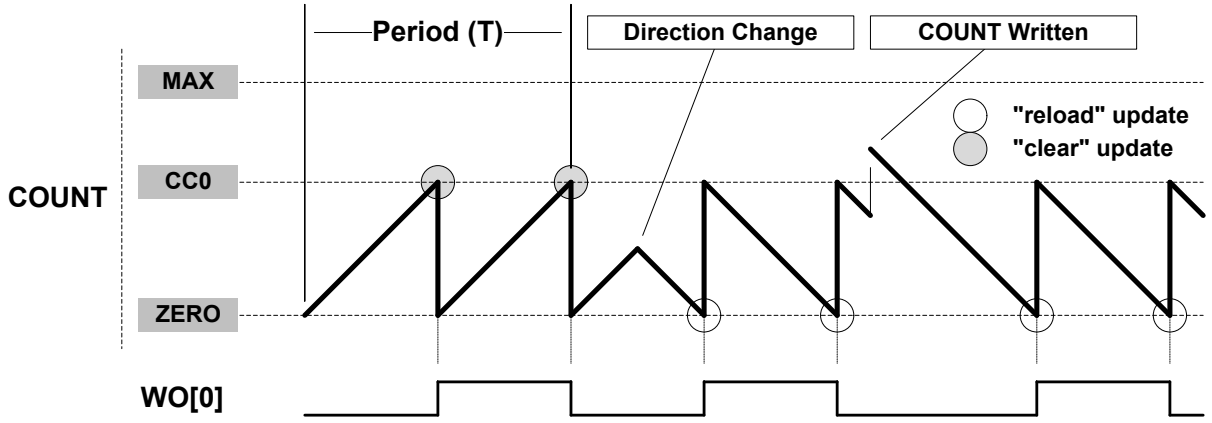
**Figure 35-4.** Normal Frequency Operation




### Match Frequency Generation (MFRQ)

For Match Frequency Generation, the period time (T) is controlled by the CC0 register instead of PER or MAX. WO[0] toggles on each Update condition.

**Figure 35-5.** Match Frequency Operation




### Normal Pulse-Width Modulation Operation (NPWM)

NPWM uses single-slope PWM generation.

For single-slope PWM generation, the period time (T) is controlled by the TOP value, and CCx controls the duty cycle of the generated waveform output. When up-counting, the WO[x] is set at start or compare match between the COUNT and TOP values, and cleared on compare match between COUNT and CCx register values. When down-counting, the WO[x] is cleared at start or compare match between the COUNT and ZERO values, and set on compare match between COUNT and CCx register values.

The following equation calculates the exact resolution for a single-slope PWM ($R_{PWM_SS}$) waveform:


© 2026 Microchip Technology Inc. and its subsidiaries

$$R_{PWM_SS} = \frac{\log(TOP+1)}{\log(2)}$$

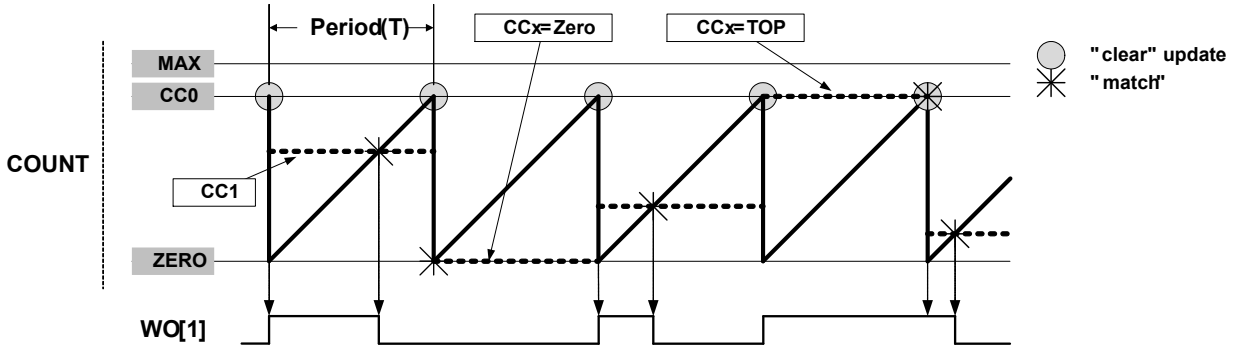
The PWM frequency ($f_{PWM_SS}$) depends on TOP value and the peripheral clock frequency ($f_{GCLK_TC}$), and can be calculated by the following equation:

$$f_{PWM_SS} = \frac{f_{GCLK_TC}}{N(TOP+1)}$$

Where N represents the prescaler divider used (1, 2, 4, 8, 16, 64, 256, 1024).

**Match Pulse-Width Modulation Operation (MPWM)**
In MPWM, the output of WO[1] is depending on CC1 as shown in the figure below. On every overflow/underflow, a one-TC-clock-cycle negative pulse is put out on WO[0] (not shown in the figure).

**Figure 35-6.** Match PWM Operation




The table below shows the Update Counter and Overflow Event/Interrupt Generation conditions in different operation modes.

**Table 35-3.** Counter Update and Overflow Event/interrupt Conditions in TC

| Name | Operation | TOP | Update | Output Waveform | | OVFIF/Event | |
|---|---|---|---|---|---|---|---|
| | | | | On Match | On Update | Up | Down |
| NFRQ | Normal Frequency | PER | TOP/ ZERO | Toggle | Stable | TOP | ZERO |
| MFRQ | Match Frequency | CC0 | TOP/ ZERO | Toggle | Stable | TOP | ZERO |
| NPWM | Single-slope PWM | PER | TOP/ ZERO | See description above. | | TOP | ZERO |
| MPWM | Single-slope PWM | CC0 | TOP/ ZERO | Toggle | Toggle | TOP | ZERO |

**Related Links**
PORT - I/O Pin Controller

### 35.6.2.7. Double Buffering

The Compare Channels (CCx) registers, and the Period (PER) register in 8-bit mode are double buffered. Each buffer register has a buffer valid bit (CCBUFVx or PERBUFV) in the STATUS register, which indicates that the buffer register contains a new valid value that can be copied into the corresponding register. As long as the respective buffer valid status flag (PERBUFV or CCBUFVx) are set to '1', related syncbusy bits are set (SYNCBUSY.PER or SYNCBUSY.CCx), a write to the respective PER/PERBUF or CCx/CCBUFx registers will generate a PAC error, and access to the respective PER or CCx register is invalid.

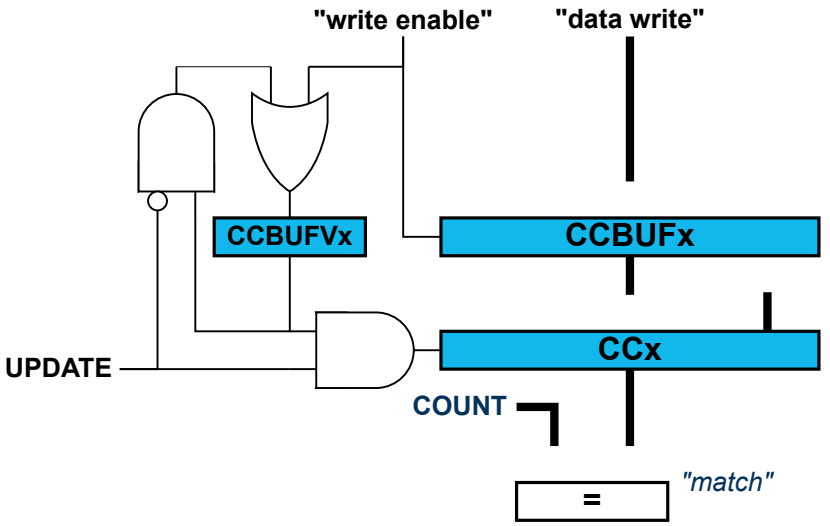
When the buffer valid flag bit in the STATUS register is '1' and the Lock Update bit in the CTRLB register is set to '0', (writing CTRLBCLR.LUPD to '1'), double buffering is enabled: the data from buffer registers will be copied into the corresponding register under hardware UPDATE conditions, then the buffer valid flags bit in the STATUS register are automatically cleared by hardware.


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** The software update command (CTRLBSET.CMD=0x3) is acting independently of the LUPD value.

A compare register is double buffered as in the following figure.

**Figure 35-7.** Compare Channel Double Buffering



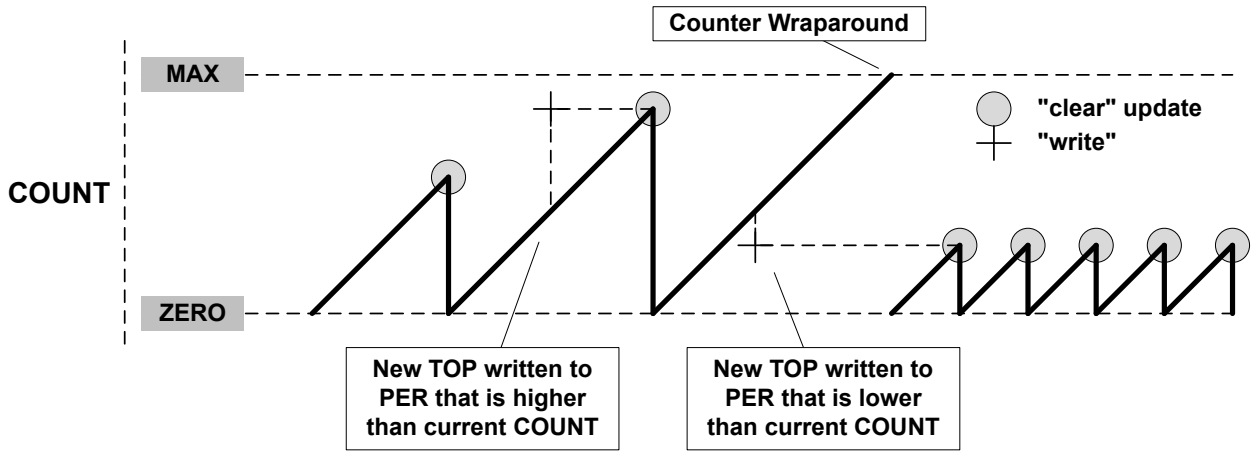

Both the registers (PER/CCx) and corresponding buffer registers (PERBUF/CCBUFx) are available in the I/O register map, and the double buffering feature is not mandatory. The double buffering is disabled by writing a '1' to CTRLBSET.LUPD.

**Note:** In NFRQ, MFRQ or PWM down-counting counter mode (CTRLBSET.DIR=1), when double buffering is enabled (CTRLBCLR.LUPD=1), PERBUF register is continously copied into the PER independently of update conditions.

### Changing the Period

The counter period can be changed by writing a new TOP value to the Period register (PER or CC0, depending on the waveform generation mode), which is available in 8-bit mode. Any period update on registers (PER or CCx) is effective after the synchronization delay.

**Figure 35-8.** Unbuffered Single-Slope Up-Counting Operation



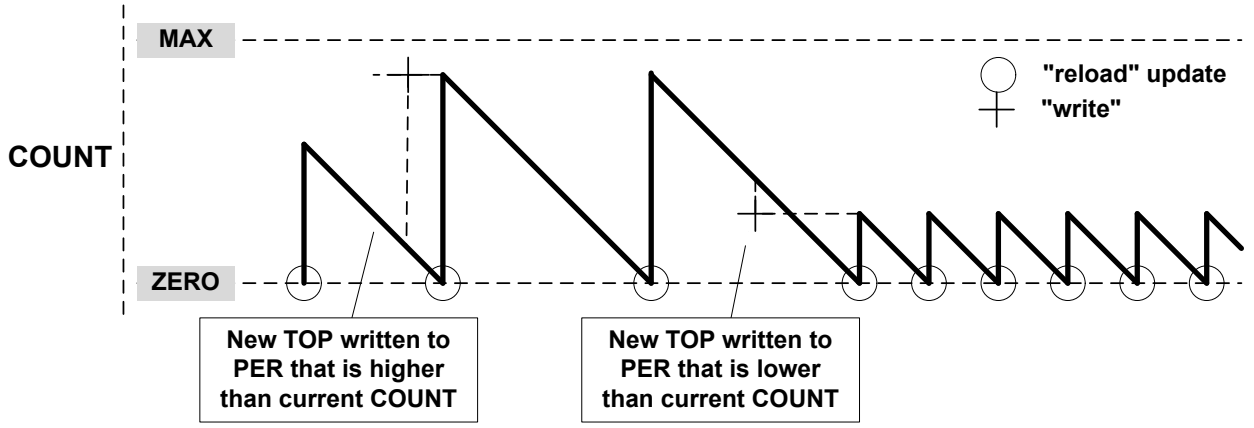


© 2026 Microchip Technology Inc. and its subsidiaries

A counter wraparound can occur in any operation mode when up-counting without buffering, see Figure 35-8.

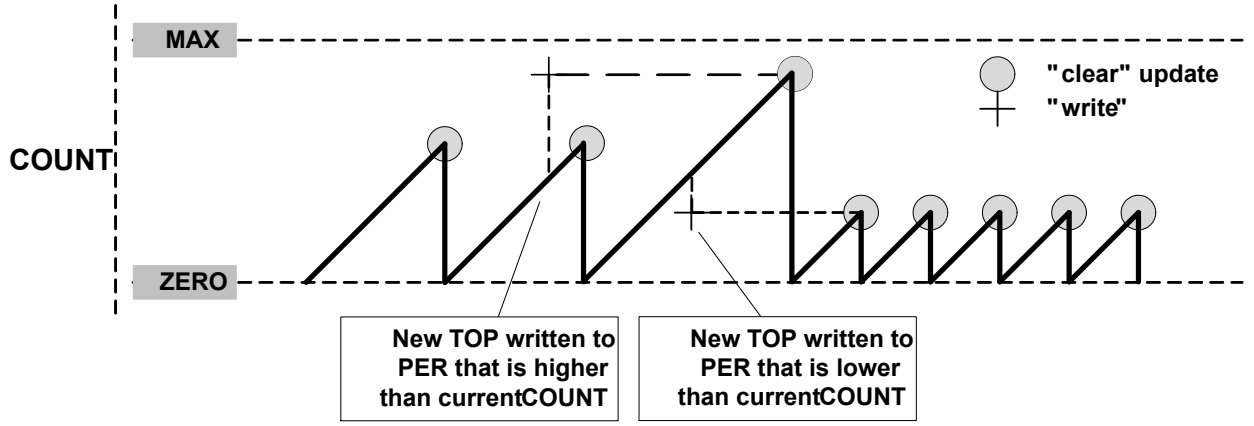
COUNT and TOP are continuously compared, so when a new TOP value that is lower than current COUNT is written to TOP, COUNT will wrap before a compare match.

**Figure 35-9.** Unbuffered Single-Slope Down-Counting Operation




When double buffering is used, the buffer can be written at any time and the counter will still maintain correct operation. The period register is always updated on the update condition, as shown in Figure 35-10. This prevents wraparound and the generation of odd waveforms.

**Figure 35-10.** Changing the Period Using Buffering




### 35.6.2.8. Capture Operations

To enable and use capture operations, the corresponding Capture Channel x Enable bit in the Control A register (CTRLA.CAPTENx) must be written to '1'.

A capture trigger can be provided by input event line TC_EV or by asynchronous I/O pin WO[x] for each capture channel or by a TC event. To enable the capture from input event line, Event Input Enable bit in the Event Control register (EVCTRL.TCEI) must be written to '1'. To enable the capture from the I/O pin, the Capture On Pin x Enable bit in the CTRLA register (CTRLA.COPENx) must be written to '1'.


© 2026 Microchip Technology Inc. and its subsidiaries

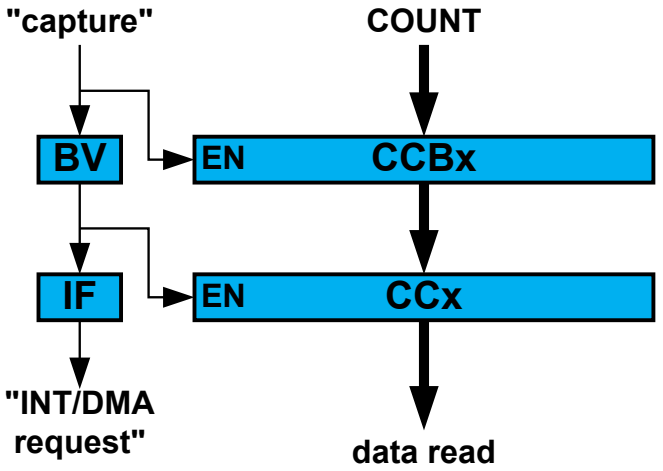
**Notes:**

1. Capture on I/Os is only possible in 'Event' and 'Time-Stamp' capture action modes. Other modes can only use internal events (if I/Os toggling is needed in other modes, then the I/Os edge must be configured for generating internal events).
2. Capture on an event from the Event System is possible in 'Event', 'PPW/PWP/PW' and 'Time-Stamp' capture modes. In this case, the event system channels must be configured to operate in Asynchronous mode of operation.
3. Depending on CTRLA.COPENx, channelx can be configured for I/Os or internal event capture (both are mutually exclusive). One channel can be configured for I/Os capture while the other uses internal event capture.
4. Capture of the period and duty cycle on I/Os using PPW/PWP mode is possible using EIC and Event System.

By default, a capture operation is done when a rising edge is detected on the input signal. Capture on falling edge is available, its activation is depending on the input source:

- When the channel is used with a I/O pin, write a '1' to the corresponding Invert Enable bit in the Drive Control register (DRVCTRL.INVENx).
- When the channel is counting events from the Event System, write a '1' to the TC Event Input Invert Enable bit in Event Control register (EVCTRL.TCINV).

**Figure 35-11.** Capture Double Buffering



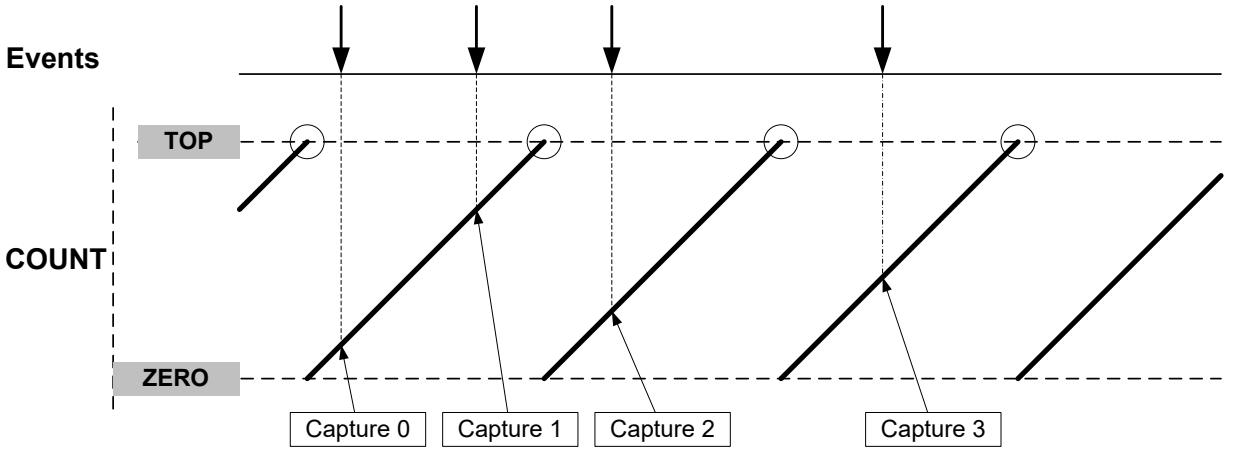

For input capture, the buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The buffer valid flag is passed to set the CCx interrupt flag (IF) and generate the optional interrupt, event, or DMA request. The CCBUFx register value can't be read, all captured data must be read from CCx register.

#### 35.6.2.8.1. Event Capture Action on Events or I/Os

The compare and capture channels can be used as input capture channels to capture events from the Event System or I/O pins and give them a timestamp. This mode is selected when EVTCTRL.EVACT is configured either as OFF, RETRIGGER, COUNT or START. The following figure shows four capture events for one capture channel.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 35-12.** Input Capture Timing




The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

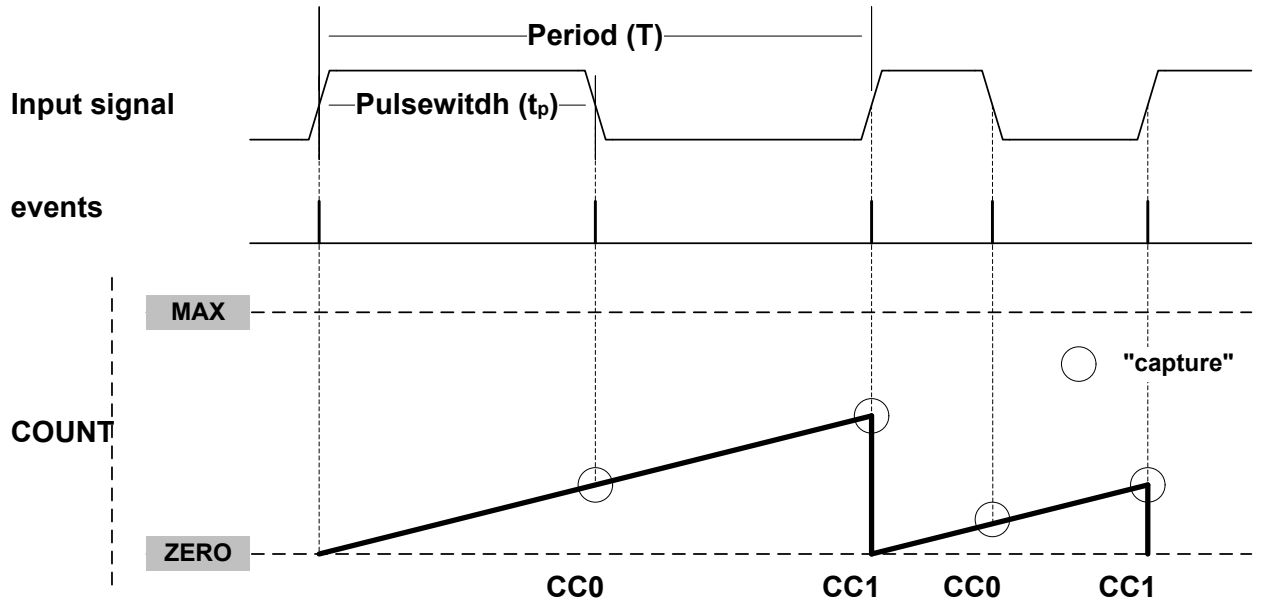
#### 35.6.2.8.2. Period and Pulse-Width (PPW) Capture Action on Events

The TC can perform two input captures and restart the counter on one of the edges. This enables the TC to measure the pulse width and period and to characterize the frequency *f* and duty cycle of an input signal:

$$f = \frac{1}{T}$$

$$\text{dutyCycle} = \frac{t_p}{T}$$

**Figure 35-13.** PWP Capture




Selecting PWP in the Event Action bit group in the Event Control register (EVCTRL.EVACT) enables the TC to perform one capture action on the rising edge and another one on the falling edge. The


© 2026 Microchip Technology Inc. and its subsidiaries

period *T* will be captured into CC1 and the pulse width $t_p$ in CC0. EVCTRL.EVACT = PPW (period and pulse-width) offers identical functionality, but will capture *T* into CC0 and $t_p$ into CC1.

The TC Event Input Invert Enable bit in the Event Control register (EVCTRL.TCINV) is used to select whether the wraparound must occur on the rising edge or the falling edge. If EVCTRL.TCINV = 1, the wraparound will happen on the falling edge.

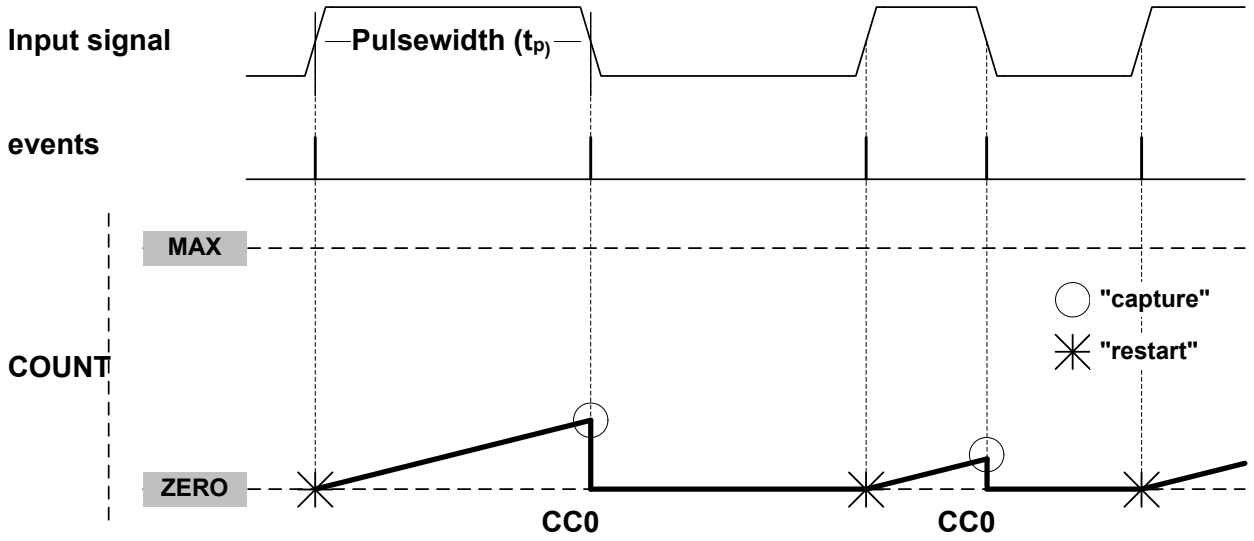
The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

**Note:** The corresponding capture is working only if the channel is enabled in capture mode (CTRLA.CAPTENx = 1). Consequently, both channels must be enabled to fully characterize the input.

#### 35.6.2.8.3. Pulse-Width Capture Action on Events

The TC performs the input capture on the falling edge of the input signal. When the edge is detected, the counter value is cleared and the TC stops counting. When a rising edge is detected on the input signal, the counter restarts the counting operation. To enable the operation on opposite edges, the input signal to capture must be inverted (refer to EVCTRL.TCEINV).

**Figure 35-14.** Pulse-Width Capture on Channel 0




The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

### 35.6.3. Additional Features

#### 35.6.3.1. One-Shot Operation

When one-shot is enabled, the counter automatically stops on the next counter overflow or underflow condition. When the counter is stopped, the Stop bit in the Status register (STATUS.STOP) is automatically set and the waveform outputs are set to zero.

One-shot operation is enabled by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT), and disabled by writing a '1' to CTRLBCLR.ONESHOT. When enabled, the TC will count until an overflow or underflow occurs and stops counting operation. The one-shot operation can be restarted by a re-trigger software command, a re-trigger event, or a start event. When the counter restarts its operation, STATUS.STOP is automatically cleared.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.6.3.2. Time-Stamp Capture on Events or I/Os

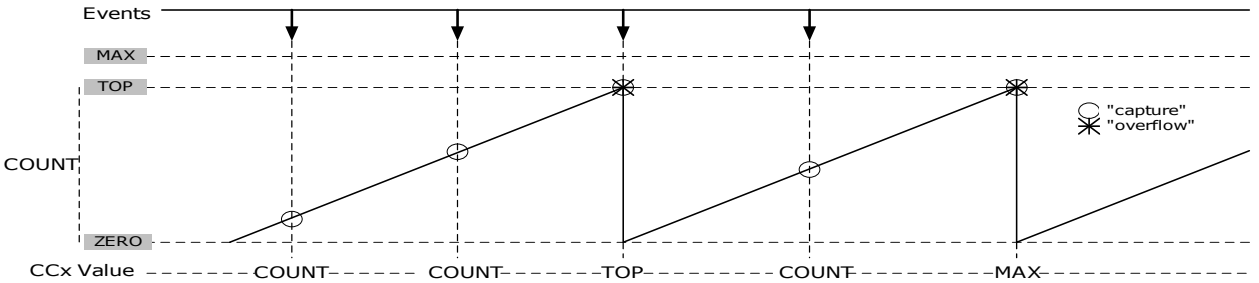
This feature is enabled when the Capture Time Stamp (STAMP) Event Action in Event Control register (EVCTRL.EVACT) is selected. The counter TOP value must be smaller than MAX.

When a capture event from the Event System or the I/O pin is detected, the COUNT value is copied into the corresponding Channel x Compare/Capture Value (CCx) register. In case of an overflow, the MAX value is copied into the corresponding CCx register.

When a valid captured value is present in the capture channel register, the corresponding Capture Channel x Interrupt Flag (INTFLAG.MCx) is set.

The timer/counter can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Channel interrupt flag (INTFLAG.MCx) is still set, the new time-stamp will not be stored and INTFLAG.ERR will be set.

**Figure 35-15.** Time Stamp




### 35.6.3.3. Minimum Capture (SAM C20/C21 N only)

The minimum capture is enabled by writing the CAPTMIN mode in the Channel n Capture Mode bits in the Control A register (CTRLA.CAPTMODEn = CAPTMIN).

*CCx Content:*

In CAPTMIN operations, CCx keeps the Minimum captured values. Before enabling this mode of capture, the user must initialize the corresponding CCx register value to a value different from zero. If the CCx register initial value is zero, no captures will be performed using the corresponding channel.

*MCx Behaviour:*

In CAPTMIN operation, capture is performed only when on capture event time, the counter value is lower than the last captured value. The MCx interrupt flag is set only when on capture event time, the counter value is upper or equal to the value captured on the previous event. So interrupt flag is set when a new absolute local Minimum value has been detected.

### 35.6.3.4. Maximum Capture (SAM C20/C21 N only)

The maximum capture is enabled by writing the CAPTMAX mode in the Channel n Capture Mode bits in the Control A register (CTRLA.CAPTMODEn = CAPTMAX).

*CCx Content:*

In CAPTMAX operations, CCx keeps the Maximum captured values. Before enabling this mode of capture, the user must initialize the corresponding CCx register value to a value different from TOP. If the CCx register initial value is TOP, no captures will be performed using the corresponding channel.

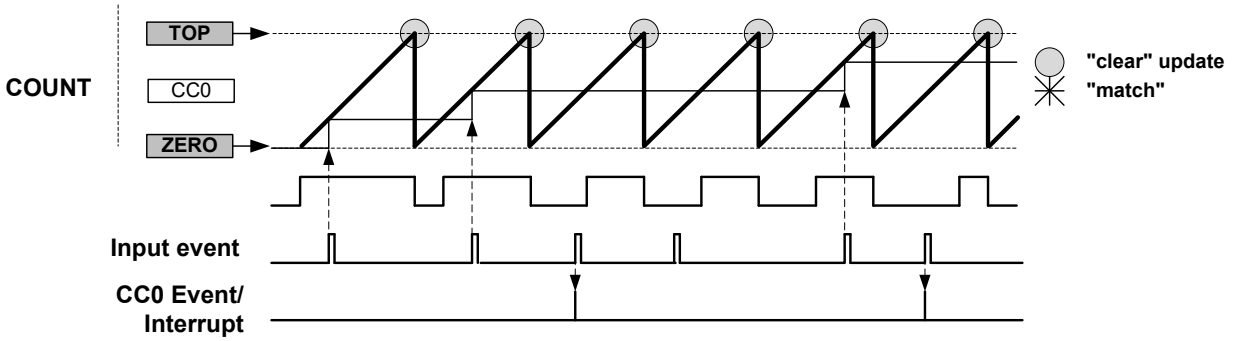
*MCx Behaviour:*

In CAPTMAX operation, capture is performed only when on capture event time, the counter value is upper than the last captured value. The MCx interrupt flag is set only when on capture event time,


© 2026 Microchip Technology Inc. and its subsidiaries

the counter value is lower or equal to the value captured on the previous event. So interrupt flag is set when a new absolute local Maximum value has been detected.

**Figure 35-16.** Maximum Capture Operation with CC0 Initialized with ZERO Value




### 35.6.4. DMA Operation

The TC can generate the following DMA requests:

- Overflow (OVF): the request is set when an update condition (overflow, underflow or re-trigger) is detected, the request is cleared by hardware on DMA acknowledge.
- Match or Capture Channel x (MCx): for a compare channel, the request is set on each compare match detection, the request is cleared by hardware on DMA acknowledge. For a capture channel, the request is set when valid data is present in the CCx register, and cleared when CCx register is read.

### 35.6.5. Interrupts

The TC has the following interrupt sources:

- Overflow/Underflow (OVF)
- Match or Capture Channel x (MCx)
- Capture Overflow Error (ERR)

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs.

Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until either the interrupt flag is cleared, the interrupt is disabled, or the TC is reset. See INTFLAG for details on how to clear interrupt flags.

The TC has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

Nested Vector Interrupt Controller

### 35.6.6. Events

The TC can generate the following output events:

- Overflow/Underflow (OVF)


© 2026 Microchip Technology Inc. and its subsidiaries

- Match or Capture Channel x (MCx)

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.MCEOx) enables the corresponding output event. The output event is disabled by writing EVCTRL.MCEOx=0.

One of the following event actions can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT):

- Disable event action (OFF)
- Start TC (START)
- Re-trigger TC (RETRIGGER)
- Count on event (COUNT)
- Capture time stamp (STAMP)
- Capture Period (PPW and PWP)
- Capture Pulse Width (PW)

Writing a '1' to the TC Event Input bit in the Event Control register (EVCTRL.TCEI) enables input events to the TC. Writing a '0' to this bit disables input events to the TC. The TC requires only asynchronous event inputs. For further details on how configuring the asynchronous events, refer to *EVSYS - Event System*.

**Related Links**
Event System (EVSYS)

### 35.6.7. Sleep Mode Operation

The TC can be configured to operate in any sleep mode. To be able to run in standby, the RUNSTDBY bit in the Control A register (CTRLA.RUNSTDBY) must be '1'. This peripheral can wake up the device from any sleep mode using interrupts or perform actions through the Event System.

If the On Demand bit in the Control A register (CTRLA.ONDEMAND) is written to '1', the module stops requesting its peripheral clock when the STOP bit in STATUS register (STATUS.STOP) is set to '1'. When a re-trigger or start condition is detected, the TC requests the clock before the operation starts.

### 35.6.8. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset and Enable bits in Control A register (CTRLA.SWRST and CTRLA.ENABLE)
- Capture Channel Buffer Valid bit in STATUS register (STATUS.CCBUFVx)

The following registers are synchronized when written:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Count Value register (COUNT)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Channel x Compare/Capture Value and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD).

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

## 35.7. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1. Register Summary - 8-bit Mode

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | | | | | ALOCK | PRESCALER[2:0] | | |
| | | 23:16 | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| | | 31:24 | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| 0x04 | CTRLBCLR | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x05 | CTRLBSET | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x06 | EVCTRL | 7:0 | | | TCEI | TCINV | | EVACT[2:0] | | |
| | | 15:8 | | | MCEO1 | MCEO0 | | | | OVFEO |
| 0x08 | INTENCLR | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x09 | INTENSET | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0A | INTFLAG | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0B | STATUS | 7:0 | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| 0x0C | WAVE | 7:0 | | | | | | | WAVEGEN[1:0] | |
| 0x0D | DRVCTRL | 7:0 | | | | | | | INVEN1 | INVEN0 |
| 0x0E | Reserved | | | | | | | | | |
| 0x0F | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x10 | SYNCBUSY | 7:0 | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| 0x15 ... 0x1A | Reserved | | | | | | | | | |
| 0x1B | PER | 7:0 | PER[7:0] | | | | | | | |
| 0x1C | CC0 | 7:0 | CC[7:0] | | | | | | | |
| 0x1D | CC1 | 7:0 | CC[7:0] | | | | | | | |
| 0x1E ... 0x2E | Reserved | | | | | | | | | |
| 0x2F | PERBUF | 7:0 | PERBUF[7:0] | | | | | | | |
| 0x30 | CCBUF0 | 7:0 | CCBUF[7:0] | | | | | | | |
| 0x31 | CCBUF1 | 7:0 | CCBUF[7:0] | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized, Enable-Protected Bits

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| Access | | | | R/W | R/W | | R/W | R/W |
| Reset | | | | 0 | 0 | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ALOCK | PRESCALER[2:0] | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 28:27 – CAPTMODE1[1:0]** Capture mode Channel 1
These bits select the channel 1 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 25:24 – CAPTMODE0[1:0]** Capture mode Channel 0
These bits select the channel 0 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 20, 21 – COPENx** Capture On Pin x Enable
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | Event from Event System is selected as trigger source for capture operation on channel x. |
| 1 | I/O pin is selected as trigger source for capture operation on channel x. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 16, 17 – CAPTENx** Capture Channel x Enable

Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | CAPTEN disables capture on channel x. |
| 1 | CAPTEN enables capture on channel x. |

**Bit 11 – ALOCK** Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The LUPD bit is not affected on overflow/underflow, and re-trigger event. |
| 1 | The LUPD bit is set on each overflow/underflow or re-trigger event. |

**Bits 10:8 – PRESCALER[2:0]** Prescaler

These bits select the counter prescaler factor.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV1 | Prescaler: GCLK_TC |
| 0x1 | DIV2 | Prescaler: GCLK_TC/2 |
| 0x2 | DIV4 | Prescaler: GCLK_TC/4 |
| 0x3 | DIV8 | Prescaler: GCLK_TC/8 |
| 0x4 | DIV16 | Prescaler: GCLK_TC/16 |
| 0x5 | DIV64 | Prescaler: GCLK_TC/64 |
| 0x6 | DIV256 | Prescaler: GCLK_TC/256 |
| 0x7 | DIV1024 | Prescaler: GCLK_TC/1024 |

**Bit 7 – ONDEMAND** Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1). |
| 1 | The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit is used to keep the TC running in standby mode.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TC is halted in standby. |
| 1 | The TC continues to run in standby. |

**Bits 5:4 – PRESCSYNC[1:0]** Prescaler and Counter Synchronization

These bits select whether the counter should wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | GCLK | Reload or reset the counter on next generic clock |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x1 | PRESC | Reload or reset the counter on next prescaler clock |
| 0x2 | RESYNC | Reload or reset the counter on next generic clock. Reset the prescaler counter |
| 0x3 | - | Reserved |

**Bits 3:2 – MODE[1:0]** Timer Counter Mode
These bits select the counter mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT16 | Counter in 16-bit mode |
| 0x1 | COUNT8 | Counter in 8-bit mode |
| 0x2 | COUNT32 | Counter in 32-bit mode |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** Enable
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.ENABLE must be checked to ensure the CTRLA.ENABLE synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the TC, except DBGCTRL, to their initial state, and the TC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.SWRST must be checked to ensure the CTRLA.SWRST synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.2. Control B Clear

**Name:** CTRLBCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-Synchronized, Write-Synchronized

This register allows the user to clear bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set register (CTRLBSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TCx clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will disable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
This bit has no effect when input capture operation is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the LUPD bit.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.3. Control B Set

**Name:** CTRLBSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-synchronized, Write-Synchronized

This register allows the user to set bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Clear register (CTRLBCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TC clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a value different from 0x0 to these bits will issue a command for execution.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RETRIGGER | Force a start, restart or retrigger |
| 0x2 | STOP | Force a stop |
| 0x3 | UPDATE | Force update of double buffered registers |
| 0x4 | READSYNC | Force a read synchronization of COUNT |

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will enable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.4. Event Control

**Name:** EVCTRL
**Offset:** 0x06
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | MCEO1 | MCEO0 | | | | OVFEO |
| Access | | | R/W | R/W | | | | R/W |
| Reset | | | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TCEI | TCINV | | EVACT[2:0] | | |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 0 | 0 | | 0 | 0 | 0 |

**Bit 13 – MCEO1** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 12 – MCEO0** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 8 – OVFEO** Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

| Value | Description |
|---|---|
| 0 | Overflow/Underflow event is disabled and will not be generated. |
| 1 | Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow. |

**Bit 5 – TCEI** TC Event Enable
This bit is used to enable asynchronous input events to the TC.

| Value | Description |
|---|---|
| 0 | Incoming events are disabled. |
| 1 | Incoming events are enabled. |

**Bit 4 – TCINV** TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

| Value | Description |
|---|---|
| 0 | Input event source is not inverted. |
| 1 | Input event source is inverted. |

**Bits 2:0 – EVACT[2:0]** Event Action
These bits define the event action the TC will perform on an event.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Event action disabled |
| 0x1 | RETRIGGER | Start, restart or retrigger TC on event |
| 0x2 | COUNT | Count on event |
| 0x3 | START | Start TC on event |
| 0x4 | STAMP | Time stamp capture |
| 0x5 | PPW | Period captured in CC0, pulse width in CC1 |
| 0x6 | PWP | Period captured in CC1, pulse width in CC0 |
| 0x7 | PW | Pulse width capture |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x0A
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 4 – MC0** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 1 – ERR** Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is nowhere to store the new capture.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

**Bit 0 – OVF** Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.8. Status

**Name:** STATUS
**Offset:** 0x0B
**Reset:** 0x01
**Property:** Read-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| Access | | | R/W | R/W | R/W | | R | R |
| Reset | | | 0 | 0 | 0 | | 0 | 1 |

**Bits 4, 5 – CCBUFV** Channel x Compare or Capture Buffer Valid
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register.
The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

**Bit 3 – PERBUFV** Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

**Bit 1 – SLAVE** Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1 and/or TC3). The bit is set when the associated Host TC (TC0 and TC2, respectively) is set to run in 32-bit mode.

**Bit 0 – STOP** Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

| Value | Description |
|---|---|
| 0 | Counter is running. |
| 1 | Counter is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.9. Waveform Generation Control

**Name:** WAVE
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | WAVEGEN[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 1:0 – WAVEGEN[1:0]** Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation should be used. The waveform generation operations are explained in Waveform Output Operations.
These bits are not synchronized.

| Value | Name | Operation | Top Value | Output Waveform on Match | Output Waveform on Wraparound |
|---|---|---|---|---|---|
| 0x0 | NFRQ | Normal frequency | PER[1] / Max | Toggle | No action |
| 0x1 | MFRQ | Match frequency | CC0 | Toggle | No action |
| 0x2 | NPWM | Normal PWM | PER[1] / Max | Set | Clear |
| 0x3 | MPWM | Match PWM | CC0 | Set | Clear |

1) This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode it is the respective MAX value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.10. Driver Control

**Name:** DRVCTRL
**Offset:** 0x0D
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | INVEN1 | INVEN0 |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0, 1 – INVENx** Output Waveform x Invert Enable
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

| Value | Description |
|---|---|
| 0 | Disable inversion of the WO[x] output and IO input pin. |
| 1 | Enable inversion of the WO[x] output and IO input pin. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0F
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Run in Debug Mode
This bit is not affected by a software Reset, and should not be changed by software while the TC is enabled.

| Value | Description |
|---|---|
| 0 | The TC is halted when the device is halted in debug mode. |
| 1 | The TC continues normal operation when the device is halted in debug mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.12. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 6, 7 – CCx** Compare/Capture Channel x Synchronization Busy
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

**Bit 5 – PER** PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PERBUF is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

**Bit 4 – COUNT** COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

**Bit 3 – STATUS** STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

**Bit 2 – CTRLB** CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE bit between the clock domains is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

This bit is set when the synchronization of ENABLE bit between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.13. Counter Value, 8-bit Mode

**Name:** COUNT
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

**Note:** Prior to any read access, this register must be synchronized by user by writing the according TC Command value to the Control B Set register (CTRLBSET.CMD=READSYNC).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – COUNT[7:0]** Counter Value
These bits contain the current counter value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.14. Period Value, 8-bit Mode

**Name:** PER
**Offset:** 0x1B
**Reset:** 0xFF
**Property:** Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.PER must be checked to ensure the PER register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 7:0 – PER[7:0]** Period Value
These bits hold the value of the TC period count.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.15. Channel x Compare/Capture Value, 8-bit Mode

**Name:** CCx
**Offset:** 0x1C + x*0x01 [x=0..1]
**Reset:** 0x00
**Property:** Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – CC[7:0]** Channel x Compare/Capture Value
These bits contain the compare/capture value in 8-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (WAVE.WAVEGEN), the CC0 register is used as a period register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.16. Period Buffer Value, 8-bit Mode

**Name:** PERBUF
**Offset:** 0x2F
**Reset:** 0xFF
**Property:** Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PERBUF[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 7:0 – PERBUF[7:0]** Period Buffer Value
These bits hold the value of the period buffer register. The value is copied to PER register on UPDATE condition.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.1.17. Channel x Compare Buffer Value, 8-bit Mode

**Name:** CCBUFx
**Offset:** 0x30 + x*0x01 [x=0..1]
**Reset:** 0x00
**Property:** Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – CCBUF[7:0]** Channel x Compare Buffer Value
These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2. Register Summary - 16-bit Mode

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | | | | | ALOCK | PRESCALER[2:0] | | |
| | | 23:16 | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| | | 31:24 | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| 0x04 | CTRLBCLR | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x05 | CTRLBSET | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x06 | EVCTRL | 7:0 | | | TCEI | TCINV | | EVACT[2:0] | | |
| | | 15:8 | | | MCEO1 | MCEO0 | | | | OVFEO |
| 0x08 | INTENCLR | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x09 | INTENSET | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0A | INTFLAG | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0B | STATUS | 7:0 | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| 0x0C | WAVE | 7:0 | | | | | | | WAVEGEN[1:0] | |
| 0x0D | DRVCTRL | 7:0 | | | | | | | INVEN1 | INVEN0 |
| 0x0E | Reserved | | | | | | | | | |
| 0x0F | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x10 | SYNCBUSY | 7:0 | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| 0x16 ... 0x1B | Reserved | | | | | | | | | |
| 0x1C | CC0 | 7:0 | CC[7:0] | | | | | | | |
| | | 15:8 | CC[15:8] | | | | | | | |
| 0x1E | CC1 | 7:0 | CC[7:0] | | | | | | | |
| | | 15:8 | CC[15:8] | | | | | | | |
| 0x20 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | CCBUF0 | 7:0 | CCBUF[7:0] | | | | | | | |
| | | 15:8 | CCBUF[15:8] | | | | | | | |
| 0x32 | CCBUF1 | 7:0 | CCBUF[7:0] | | | | | | | |
| | | 15:8 | CCBUF[15:8] | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized, Enable-Protected Bits

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| Access | | | | R/W | R/W | | R/W | R/W |
| Reset | | | | 0 | 0 | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ALOCK | PRESCALER[2:0] | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 28:27 – CAPTMODE1[1:0]** Capture mode Channel 1
These bits select the channel 1 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 25:24 – CAPTMODE0[1:0]** Capture mode Channel 0
These bits select the channel 0 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 20, 21 – COPENx** Capture On Pin x Enable
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | Event from Event System is selected as trigger source for capture operation on channel x. |
| 1 | I/O pin is selected as trigger source for capture operation on channel x. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 16, 17 – CAPTENx** Capture Channel x Enable

Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | CAPTEN disables capture on channel x. |
| 1 | CAPTEN enables capture on channel x. |

**Bit 11 – ALOCK** Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The LUPD bit is not affected on overflow/underflow, and re-trigger event. |
| 1 | The LUPD bit is set on each overflow/underflow or re-trigger event. |

**Bits 10:8 – PRESCALER[2:0]** Prescaler

These bits select the counter prescaler factor.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV1 | Prescaler: GCLK_TC |
| 0x1 | DIV2 | Prescaler: GCLK_TC/2 |
| 0x2 | DIV4 | Prescaler: GCLK_TC/4 |
| 0x3 | DIV8 | Prescaler: GCLK_TC/8 |
| 0x4 | DIV16 | Prescaler: GCLK_TC/16 |
| 0x5 | DIV64 | Prescaler: GCLK_TC/64 |
| 0x6 | DIV256 | Prescaler: GCLK_TC/256 |
| 0x7 | DIV1024 | Prescaler: GCLK_TC/1024 |

**Bit 7 – ONDEMAND** Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1). |
| 1 | The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit is used to keep the TC running in standby mode.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TC is halted in standby. |
| 1 | The TC continues to run in standby. |

**Bits 5:4 – PRESCSYNC[1:0]** Prescaler and Counter Synchronization

These bits select whether the counter should wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | GCLK | Reload or reset the counter on next generic clock |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x1 | PRESC | Reload or reset the counter on next prescaler clock |
| 0x2 | RESYNC | Reload or reset the counter on next generic clock. Reset the prescaler counter |
| 0x3 | - | Reserved |

**Bits 3:2 – MODE[1:0]** Timer Counter Mode
These bits select the counter mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT16 | Counter in 16-bit mode |
| 0x1 | COUNT8 | Counter in 8-bit mode |
| 0x2 | COUNT32 | Counter in 32-bit mode |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** Enable
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.ENABLE must be checked to ensure the CTRLA.ENABLE synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the TC, except DBGCTRL, to their initial state, and the TC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.SWRST must be checked to ensure the CTRLA.SWRST synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.2. Control B Clear

**Name:** CTRLBCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-Synchronized, Write-Synchronized

This register allows the user to clear bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set register (CTRLBSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TCx clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will disable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
This bit has no effect when input capture operation is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the LUPD bit.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.3. Control B Set

**Name:** CTRLBSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-synchronized, Write-Synchronized

This register allows the user to set bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Clear register (CTRLBCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TC clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a value different from 0x0 to these bits will issue a command for execution.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RETRIGGER | Force a start, restart or retrigger |
| 0x2 | STOP | Force a stop |
| 0x3 | UPDATE | Force update of double buffered registers |
| 0x4 | READSYNC | Force a read synchronization of COUNT |

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will enable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.4. Event Control

**Name:** EVCTRL
**Offset:** 0x06
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | MCEO1 | MCEO0 | | | | OVFEO |
| Access | | | R/W | R/W | | | | R/W |
| Reset | | | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TCEI | TCINV | | EVACT[2:0] | | |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 0 | 0 | | 0 | 0 | 0 |

**Bit 13 – MCEO1** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 12 – MCEO0** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 8 – OVFEO** Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

| Value | Description |
|---|---|
| 0 | Overflow/Underflow event is disabled and will not be generated. |
| 1 | Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow. |

**Bit 5 – TCEI** TC Event Enable
This bit is used to enable asynchronous input events to the TC.

| Value | Description |
|---|---|
| 0 | Incoming events are disabled. |
| 1 | Incoming events are enabled. |

**Bit 4 – TCINV** TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

| Value | Description |
|---|---|
| 0 | Input event source is not inverted. |
| 1 | Input event source is inverted. |

**Bits 2:0 – EVACT[2:0]** Event Action
These bits define the event action the TC will perform on an event.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Event action disabled |
| 0x1 | RETRIGGER | Start, restart or retrigger TC on event |
| 0x2 | COUNT | Count on event |
| 0x3 | START | Start TC on event |
| 0x4 | STAMP | Time stamp capture |
| 0x5 | PPW | Period captured in CC0, pulse width in CC1 |
| 0x6 | PWP | Period captured in CC1, pulse width in CC0 |
| 0x7 | PW | Pulse width capture |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x0A
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 4 – MC0** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 1 – ERR** Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is nowhere to store the new capture.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

**Bit 0 – OVF** Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.8. Status

**Name:** STATUS
**Offset:** 0x0B
**Reset:** 0x01
**Property:** Read-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| Access | | | R/W | R/W | R/W | | R | R |
| Reset | | | 0 | 0 | 0 | | 0 | 1 |

**Bits 4, 5 – CCBUFV** Channel x Compare or Capture Buffer Valid
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register.
The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

**Bit 3 – PERBUFV** Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

**Bit 1 – SLAVE** Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1 and/or TC3). The bit is set when the associated Host TC (TC0 and TC2, respectively) is set to run in 32-bit mode.

**Bit 0 – STOP** Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

| Value | Description |
|---|---|
| 0 | Counter is running. |
| 1 | Counter is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.9. Waveform Generation Control

**Name:** WAVE
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | WAVEGEN[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 1:0 – WAVEGEN[1:0]** Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation should be used. The waveform generation operations are explained in Waveform Output Operations.
These bits are not synchronized.

| Value | Name | Operation | Top Value | Output Waveform on Match | Output Waveform on Wraparound |
|---|---|---|---|---|---|
| 0x0 | NFRQ | Normal frequency | PER[1] / Max | Toggle | No action |
| 0x1 | MFRQ | Match frequency | CC0 | Toggle | No action |
| 0x2 | NPWM | Normal PWM | PER[1] / Max | Set | Clear |
| 0x3 | MPWM | Match PWM | CC0 | Set | Clear |

1) This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode it is the respective MAX value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.10. Driver Control

**Name:** DRVCTRL
**Offset:** 0x0D
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | INVEN1 | INVEN0 |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0, 1 – INVENx** Output Waveform x Invert Enable
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

| Value | Description |
|---|---|
| 0 | Disable inversion of the WO[x] output and IO input pin. |
| 1 | Enable inversion of the WO[x] output and IO input pin. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0F
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Run in Debug Mode
This bit is not affected by a software Reset, and should not be changed by software while the TC is enabled.

| Value | Description |
|---|---|
| 0 | The TC is halted when the device is halted in debug mode. |
| 1 | The TC continues normal operation when the device is halted in debug mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.12. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 6, 7 – CCx** Compare/Capture Channel x Synchronization Busy
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

**Bit 5 – PER** PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PERBUF is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

**Bit 4 – COUNT** COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

**Bit 3 – STATUS** STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

**Bit 2 – CTRLB** CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE bit between the clock domains is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

This bit is set when the synchronization of ENABLE bit between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.13. Counter Value, 16-bit Mode

**Name:** COUNT
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

**Note:** Prior to any read access, this register must be synchronized by user by writing the according TC Command value to the Control B Set register (CTRLBSET.CMD=READSYNC).

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – COUNT[15:0]** Counter Value
These bits contain the current counter value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.14. Channel x Compare/Capture Value, 16-bit Mode

**Name:** CCx
**Offset:** 0x1C + x*0x02 [x=0..1]
**Reset:** 0x0000
**Property:** Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CC[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – CC[15:0]** Channel x Compare/Capture Value
These bits contain the compare/capture value in 16-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (WAVE.WAVEGEN), the CC0 register is used as a period register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.2.15. Channel x Compare Buffer Value, 16-bit Mode

**Name:** CCBUFx
**Offset:** 0x30 + x*0x02 [x=0..1]
**Reset:** 0x0000
**Property:** Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – CCBUF[15:0]** Channel x Compare Buffer Value

These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3. Register Summary - 32-bit Mode

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | | | | | ALOCK | PRESCALER[2:0] | | |
| | | 23:16 | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| | | 31:24 | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| 0x04 | CTRLBCLR | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x05 | CTRLBSET | 7:0 | | CMD[2:0] | | | | ONESHOT | LUPD | DIR |
| 0x06 | EVCTRL | 7:0 | | | TCEI | TCINV | | EVACT[2:0] | | |
| | | 15:8 | | | MCEO1 | MCEO0 | | | | OVFEO |
| 0x08 | INTENCLR | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x09 | INTENSET | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0A | INTFLAG | 7:0 | | | MC1 | MC0 | | | ERR | OVF |
| 0x0B | STATUS | 7:0 | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| 0x0C | WAVE | 7:0 | | | | | | | WAVEGEN[1:0] | |
| 0x0D | DRVCTRL | 7:0 | | | | | | | INVEN1 | INVEN0 |
| 0x0E | Reserved | | | | | | | | | |
| 0x0F | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x10 | SYNCBUSY | 7:0 | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| | | 23:16 | COUNT[23:16] | | | | | | | |
| | | 31:24 | COUNT[31:24] | | | | | | | |
| 0x18 ... 0x1B | Reserved | | | | | | | | | |
| 0x1C | CC0 | 7:0 | CC[7:0] | | | | | | | |
| | | 15:8 | CC[15:8] | | | | | | | |
| | | 23:16 | CC[23:16] | | | | | | | |
| | | 31:24 | CC[31:24] | | | | | | | |
| 0x20 | CC1 | 7:0 | CC[7:0] | | | | | | | |
| | | 15:8 | CC[15:8] | | | | | | | |
| | | 23:16 | CC[23:16] | | | | | | | |
| | | 31:24 | CC[31:24] | | | | | | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | CCBUF0 | 7:0 | CCBUF[7:0] | | | | | | | |
| | | 15:8 | CCBUF[15:8] | | | | | | | |
| | | 23:16 | CCBUF[23:16] | | | | | | | |
| | | 31:24 | CCBUF[31:24] | | | | | | | |
| 0x34 | CCBUF1 | 7:0 | CCBUF[7:0] | | | | | | | |
| | | 15:8 | CCBUF[15:8] | | | | | | | |
| | | 23:16 | CCBUF[23:16] | | | | | | | |
| | | 31:24 | CCBUF[31:24] | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized, Enable-Protected Bits

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | CAPTMODE1[1:0] | | | CAPTMODE0[1:0] | |
| Access | | | | R/W | R/W | | R/W | R/W |
| Reset | | | | 0 | 0 | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | COPEN1 | COPEN0 | | | CAPTEN1 | CAPTEN0 |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ALOCK | PRESCALER[2:0] | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | PRESCSYNC[1:0] | | MODE[1:0] | | ENABLE | SWRST |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 28:27 – CAPTMODE1[1:0]** Capture mode Channel 1
These bits select the channel 1 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 25:24 – CAPTMODE0[1:0]** Capture mode Channel 0
These bits select the channel 0 capture mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DEFAULT | Default capture |
| 0x1 | CAPTMIN | Minimum capture |
| 0x2 | CAPTMAX | Maximum capture |
| 0x3 | | Reserved |

**Bits 20, 21 – COPENx** Capture On Pin x Enable
Bit x of COPEN[1:0] selects the trigger source for capture operation, either events or I/O pin input.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | Event from Event System is selected as trigger source for capture operation on channel x. |
| 1 | I/O pin is selected as trigger source for capture operation on channel x. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 16, 17 – CAPTENx** Capture Channel x Enable

Bit x of CAPTEN[1:0] selects whether channel x is a capture or a compare channel.
**Note:** These bits are enable-protected. These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | CAPTEN disables capture on channel x. |
| 1 | CAPTEN enables capture on channel x. |

**Bit 11 – ALOCK** Auto Lock

When this bit is set, Lock bit update (LUPD) is set to '1' on each overflow/underflow or re-trigger event.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The LUPD bit is not affected on overflow/underflow, and re-trigger event. |
| 1 | The LUPD bit is set on each overflow/underflow or re-trigger event. |

**Bits 10:8 – PRESCALER[2:0]** Prescaler

These bits select the counter prescaler factor.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV1 | Prescaler: GCLK_TC |
| 0x1 | DIV2 | Prescaler: GCLK_TC/2 |
| 0x2 | DIV4 | Prescaler: GCLK_TC/4 |
| 0x3 | DIV8 | Prescaler: GCLK_TC/8 |
| 0x4 | DIV16 | Prescaler: GCLK_TC/16 |
| 0x5 | DIV64 | Prescaler: GCLK_TC/64 |
| 0x6 | DIV256 | Prescaler: GCLK_TC/256 |
| 0x7 | DIV1024 | Prescaler: GCLK_TC/1024 |

**Bit 7 – ONDEMAND** Clock On Demand

This bit selects the clock requirements when the TC is stopped.
In standby mode, if the Run in Standby bit (CTRLA.RUNSTDBY) is '0', ONDEMAND is forced to '0'.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The On Demand is disabled. If On Demand is disabled, the TC will continue to request the clock when its operation is stopped (STATUS.STOP=1). |
| 1 | The On Demand is enabled. When On Demand is enabled, the stopped TC will not request the clock. The clock is requested when a software re-trigger command is applied or when an event with start/re-trigger action is detected. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit is used to keep the TC running in standby mode.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TC is halted in standby. |
| 1 | The TC continues to run in standby. |

**Bits 5:4 – PRESCSYNC[1:0]** Prescaler and Counter Synchronization

These bits select whether the counter should wrap around on the next GCLK_TCx clock or the next prescaled GCLK_TCx clock. It also makes it possible to reset the prescaler.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | GCLK | Reload or reset the counter on next generic clock |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x1 | PRESC | Reload or reset the counter on next prescaler clock |
| 0x2 | RESYNC | Reload or reset the counter on next generic clock. Reset the prescaler counter |
| 0x3 | - | Reserved |

**Bits 3:2 – MODE[1:0]** Timer Counter Mode
These bits select the counter mode.
**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | COUNT16 | Counter in 16-bit mode |
| 0x1 | COUNT8 | Counter in 8-bit mode |
| 0x2 | COUNT32 | Counter in 32-bit mode |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** Enable
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.ENABLE must be checked to ensure the CTRLA.ENABLE synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the TC, except DBGCTRL, to their initial state, and the TC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.
**Notes:**

1. This bit is write-synchronized: SYNCBUSY.SWRST must be checked to ensure the CTRLA.SWRST synchronization is complete.
2. This bit is not enable protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.2. Control B Clear

**Name:** CTRLBCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-Synchronized, Write-Synchronized

This register allows the user to clear bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set register (CTRLBSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TCx clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will disable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
This bit has no effect when input capture operation is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the LUPD bit.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.3. Control B Set

**Name:** CTRLBSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection, Read-synchronized, Write-Synchronized

This register allows the user to set bits in the CTRLB register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Clear register (CTRLBCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** Command
These bits are used for software control of the TC. The commands are executed on the next prescaled GCLK_TC clock cycle. When a command has been executed, the CMD bit group will be read back as zero.
Writing 0x0 to these bits has no effect.
Writing a value different from 0x0 to these bits will issue a command for execution.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RETRIGGER | Force a start, restart or retrigger |
| 0x2 | STOP | Force a stop |
| 0x3 | UPDATE | Force update of double buffered registers |
| 0x4 | READSYNC | Force a read synchronization of COUNT |

**Bit 2 – ONESHOT** One-Shot on Counter
This bit controls one-shot operation of the TC.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will enable one-shot operation.

| Value | Description |
|---|---|
| 0 | The TC will wrap around and continue counting on an overflow/underflow condition. |
| 1 | The TC will wrap around and stop on the next underflow/overflow condition. |

**Bit 1 – LUPD** Lock Update
This bit controls the update operation of the TC buffered registers.
When CTRLB.LUPD is set, the CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. Locking the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the LUPD bit.
This bit has no effect when input capture operation is enabled.

| Value | Description |
|---|---|
| 0 | The CCBUFx and PERBUF buffer registers value are copied into CCx and PER registers on hardware update condition. |
| 1 | The CCBUFx and PERBUF buffer registers value are not copied into CCx and PER registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction
This bit is used to change the direction of the counter.


© 2026 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.4. Event Control

**Name:** EVCTRL
**Offset:** 0x06
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | MCEO1 | MCEO0 | | | | OVFEO |
| Access | | | R/W | R/W | | | | R/W |
| Reset | | | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TCEI | TCINV | | EVACT[2:0] | | |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 0 | 0 | | 0 | 0 | 0 |

**Bit 13 – MCEO1** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 12 – MCEO0** Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

| Value | Description |
|---|---|
| 0 | Match/Capture event on channel x is disabled and will not be generated. |
| 1 | Match/Capture event on channel x is enabled and will be generated for every compare/capture. |

**Bit 8 – OVFEO** Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

| Value | Description |
|---|---|
| 0 | Overflow/Underflow event is disabled and will not be generated. |
| 1 | Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow. |

**Bit 5 – TCEI** TC Event Enable
This bit is used to enable asynchronous input events to the TC.

| Value | Description |
|---|---|
| 0 | Incoming events are disabled. |
| 1 | Incoming events are enabled. |

**Bit 4 – TCINV** TC Inverted Event Input Polarity
This bit inverts the asynchronous input event source.

| Value | Description |
|---|---|
| 0 | Input event source is not inverted. |
| 1 | Input event source is inverted. |

**Bits 2:0 – EVACT[2:0]** Event Action
These bits define the event action the TC will perform on an event.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Event action disabled |
| 0x1 | RETRIGGER | Start, restart or retrigger TC on event |
| 0x2 | COUNT | Count on event |
| 0x3 | START | Start TC on event |
| 0x4 | STAMP | Time stamp capture |
| 0x5 | PPW | Period captured in CC0, pulse width in CC1 |
| 0x6 | PWP | Period captured in CC1, pulse width in CC0 |
| 0x7 | PW | Pulse width capture |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will clear the corresponding Match or Capture Channel x Interrupt Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Error Interrupt Enable bit, which disables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Disable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overflow Interrupt Enable bit, which disables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 4 – MC0** Match or Capture Channel x Interrupt Enable
Writing a '0' to these bits has no effect.
Writing a '1' to MCx will set the corresponding Match or Capture Channel x Interrupt Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Error Interrupt Enable bit, which enables the Error interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Enable bit, which enables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x0A
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | | | ERR | OVF |
| Access | | | R/W | R/W | | | R/W | R/W |
| Reset | | | 0 | 0 | | | 0 | 0 |

**Bit 5 – MC1** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 4 – MC0** Match or Capture Channel x
This flag is set on a comparison match, or when the corresponding CCx register contains a valid capture value. This flag is set on the next CLK_TC_CNT cycle, and will generate an interrupt request if the corresponding Match or Capture Channel x Interrupt Enable bit in the Interrupt Enable Set register (INTENSET.MCx) is '1'.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In capture operation, this flag is automatically cleared when CCx register is read.

**Bit 1 – ERR** Error Interrupt Flag
This flag is set when a new capture occurs on a channel while the corresponding Match or Capture Channel x interrupt flag is set, in which case there is nowhere to store the new capture.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Error interrupt flag.

**Bit 0 – OVF** Overflow Interrupt Flag
This flag is set on the next CLK_TC_CNT cycle after an overflow condition occurs, and will generate an interrupt request if INTENCLR.OVF or INTENSET.OVF is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.8. Status

**Name:** STATUS
**Offset:** 0x0B
**Reset:** 0x01
**Property:** Read-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | CCBUFV1 | CCBUFV0 | PERBUFV | | SLAVE | STOP |
| Access | | | R/W | R/W | R/W | | R | R |
| Reset | | | 0 | 0 | 0 | | 0 | 1 |

**Bits 4, 5 – CCBUFV** Channel x Compare or Capture Buffer Valid
For a compare channel x, the bit x is set when a new value is written to the corresponding CCBUFx register.
The bit x is cleared by writing a '1' to it when CTRLB.LUPD is set, or it is cleared automatically by hardware on UPDATE condition.
For a capture channel x, the bit x is set when a valid capture value is stored in the CCBUFx register. The bit x is cleared automatically when the CCx register is read.

**Bit 3 – PERBUFV** Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. The bit is cleared by writing '1' to the corresponding location when CTRLB.LUPD is set, or automatically cleared by hardware on UPDATE condition. This bit is available only in 8-bit mode and will always read zero in 16- and 32-bit modes.

**Bit 1 – SLAVE** Client Status Flag
This bit is only available in 32-bit mode on the Client TC (i.e., TC1 and/or TC3). The bit is set when the associated Host TC (TC0 and TC2, respectively) is set to run in 32-bit mode.

**Bit 0 – STOP** Stop Status Flag
This bit is set when the TC is disabled, on a Stop command, or on an overflow/underflow condition when the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) is '1'.

| Value | Description |
|---|---|
| 0 | Counter is running. |
| 1 | Counter is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.9. Waveform Generation Control

**Name:** WAVE
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | WAVEGEN[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 1:0 – WAVEGEN[1:0]** Waveform Generation Mode
These bits select the waveform generation operation. They affect the top value, as shown in Waveform Output Operations. They also control whether frequency or PWM waveform generation should be used. The waveform generation operations are explained in Waveform Output Operations.
These bits are not synchronized.

| Value | Name | Operation | Top Value | Output Waveform on Match | Output Waveform on Wraparound |
|---|---|---|---|---|---|
| 0x0 | NFRQ | Normal frequency | PER[1] / Max | Toggle | No action |
| 0x1 | MFRQ | Match frequency | CC0 | Toggle | No action |
| 0x2 | NPWM | Normal PWM | PER[1] / Max | Set | Clear |
| 0x3 | MPWM | Match PWM | CC0 | Set | Clear |

1) This depends on the TC mode: In 8-bit mode, the top value is the Period Value register (PER). In 16- and 32-bit mode it is the respective MAX value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.10. Driver Control

**Name:** DRVCTRL
**Offset:** 0x0D
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | INVEN1 | INVEN0 |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0, 1 – INVENx** Output Waveform x Invert Enable
Bit x of INVEN[1:0] selects inversion of the output or capture trigger input of channel x.

| Value | Description |
|---|---|
| 0 | Disable inversion of the WO[x] output and IO input pin. |
| 1 | Enable inversion of the WO[x] output and IO input pin. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0F
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Run in Debug Mode
This bit is not affected by a software Reset, and should not be changed by software while the TC is enabled.

| Value | Description |
|---|---|
| 0 | The TC is halted when the device is halted in debug mode. |
| 1 | The TC continues normal operation when the device is halted in debug mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.12. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC1 | CC0 | PER | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 6, 7 – CCx** Compare/Capture Channel x Synchronization Busy
For details on CC channels number, refer to each TC feature list.
This bit is set when the synchronization of CCx between clock domains is started.
This bit is also set when the CCBUFx is written, and cleared on update condition. The bit is automatically cleared when the STATUS.CCBUFx bit is cleared.

**Bit 5 – PER** PER Synchronization Busy
This bit is cleared when the synchronization of PER between the clock domains is complete.
This bit is set when the synchronization of PER between clock domains is started.
This bit is also set when the PERBUF is written, and cleared on update condition. The bit is automatically cleared when the STATUS.PERBUF bit is cleared.

**Bit 4 – COUNT** COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT between the clock domains is complete.
This bit is set when the synchronization of COUNT between clock domains is started.

**Bit 3 – STATUS** STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS between the clock domains is complete.
This bit is set when a '1' is written to the Capture Channel Buffer Valid status flags (STATUS.CCBUFVx) and the synchronization of STATUS between clock domains is started.

**Bit 2 – CTRLB** CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB between the clock domains is complete.
This bit is set when the synchronization of CTRLB between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE bit between the clock domains is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

This bit is set when the synchronization of ENABLE bit between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.13. Counter Value, 32-bit Mode

**Name:** COUNT
**Offset:** 0x14
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

**Note:** Prior to any read access, this register must be synchronized by user by writing the according TC Command value to the Control B Set register (CTRLBSET.CMD=READSYNC).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – COUNT[31:0]** Counter Value
These bits contain the current counter value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.14. Channel x Compare/Capture Value, 32-bit Mode

**Name:** CCx
**Offset:** 0x1C + x*0x04 [x=0..1]
**Reset:** 0x00000000
**Property:** Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CC[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CC[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CC[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CC[31:0]** Channel x Compare/Capture Value
These bits contain the compare/capture value in 32-bit TC mode. In Match frequency (MFRQ) or Match PWM (MPWM) waveform operation (WAVE.WAVEGEN), the CC0 register is used as a period register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 35.7.3.15. Channel x Compare Buffer Value, 32-bit Mode

**Name:** CCBUFx
**Offset:** 0x30 + x*0x04 [x=0..1]
**Reset:** 0x00000000
**Property:** Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – CCBUF[31:0]** Channel x Compare Buffer Value

These bits hold the value of the Channel x Compare Buffer Value. When the buffer valid flag is '1' and double buffering is enabled (CTRLBCLR.LUPD=1), the data from buffer registers will be copied into the corresponding CCx register under UPDATE condition (CTRLBSET.CMD=0x3), including the software update command.


© 2026 Microchip Technology Inc. and its subsidiaries

# 36. Timer/Counter for Control Applications (TCC)

## 36.1. Overview

The device provides three instances of the Timer/Counter for Control applications (TCC) peripheral, TCC[2:0].

Each TCC instance consists of a counter, a prescaler, compare/capture channels and control logic. The counter can be set to count events or clock pulses. The counter together with the compare/capture channels can be configured to time stamp input events, allowing capture of frequency and pulse-width. It can also perform waveform generation, such as frequency generation and pulse-width modulation.

Waveform extensions are featured for motor control, ballast, LED, H-bridge, power converters, and other types of power control applications. They allow for low-side and high-side output with optional dead-time insertion. Waveform extensions can also generate a synchronized bit pattern across the waveform output pins. The fault options enable fault protection for safe and deterministic handling, disabling and/or shut down of external drivers.

Figure 36-1 shows all features in TCC.

**Note:** The TCC configurations, such as channel numbers and features, may be reduced for some of the TCC instances.

**Related Links**

TCC Configurations

## 36.2. Features

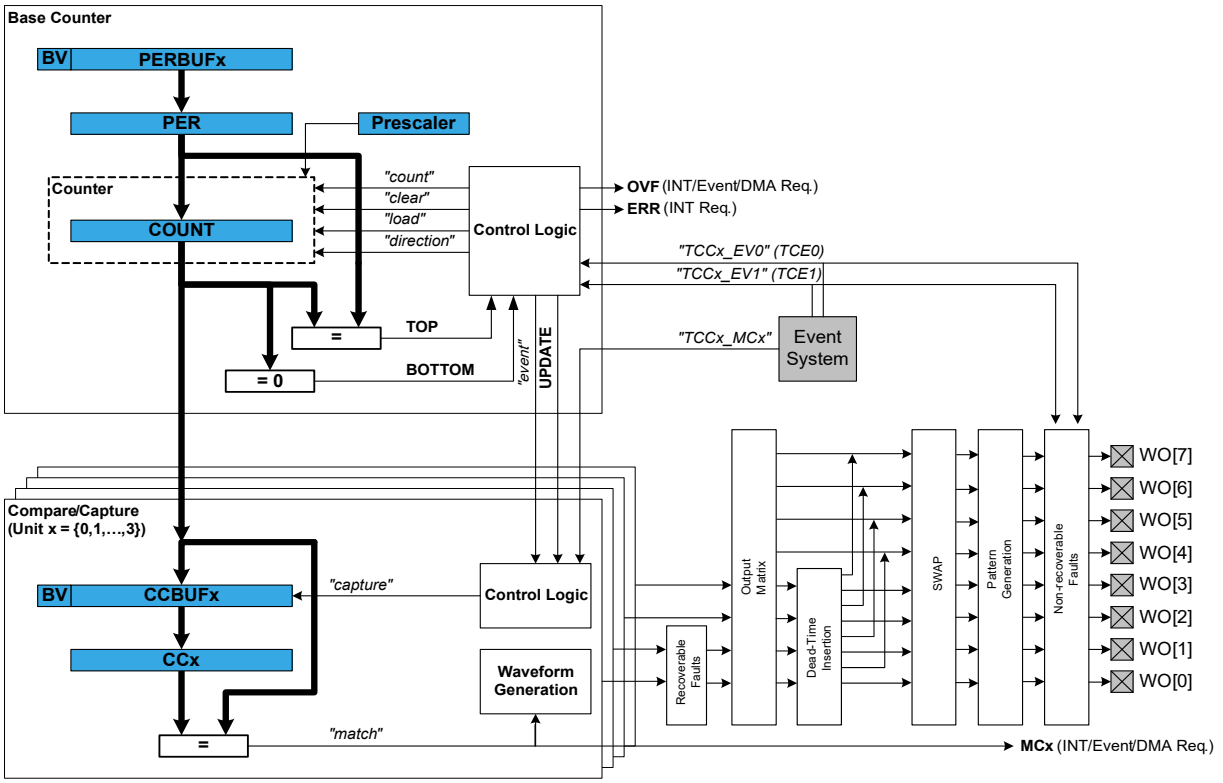
- Up to four Compare/Capture Channels (CC) with:
  - Double buffered period setting
  - Double buffered compare or capture channel
  - Circular buffer on period and compare channel registers
- Waveform Generation:
  - Frequency generation
  - Single-slope pulse-width modulation (PWM)
  - Dual-slope PWM with half-cycle reload capability
- Input Capture:
  - Event capture
  - Frequency capture
  - Pulse-width capture
- Waveform Extensions:
  - Configurable distribution of compare channels outputs across port pins
  - Low-side and high-side output with programmable dead-time insertion
  - Waveform swap option with double buffer support
  - Pattern generation with double buffer support
  - Dithering support
- Fault Protection for Safe Disabling of Drivers:
  - Two recoverable fault sources
  - Two non-recoverable fault sources


© 2026 Microchip Technology Inc. and its subsidiaries

  - Debugger can be a source of non-recoverable fault
- Input Events:
  - Two input events (EVx) for counter
  - One input event (MCx) for each channel
- Output Events:
  - Three output events (Count, re-trigger and overflow) are available for counter
  - One compare match/input capture event output for each channel
- Interrupts:
  - Overflow and re-trigger interrupt
  - Compare match/input capture interrupt
  - Interrupt on fault detection
- Can be Used with DMA and can Trigger DMA Transactions

## 36.3. Block Diagram

**Figure 36-1.** Timer/Counter for Control Applications - Block Diagram




## 36.4. Signal Description

| Pin Name | Type | Description |
|---|---|---|
| TCCx/WO[0] | Digital output | Compare channel 0 waveform output |
| TCCx/WO[1] | Digital output | Compare channel 1 waveform output |
| ... | ... | ... |
| TCCx/WO[WO_NUM-1] | Digital output | Compare channel n waveform output |


© 2026 Microchip Technology Inc. and its subsidiaries

Refer to *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**
I/O Multiplexing and Considerations

## 36.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 36.5.1. I/O Lines

In order to use the I/O lines of this peripheral, the I/O pins must be configured using the I/O Pin Controller (PORT).

**Related Links**
PORT - I/O Pin Controller

### 36.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. The interrupts can wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

### 36.5.3. Clocks

The TCC bus clocks (CLK_TCCx_APB) can be enabled and disabled in the Main Clock module. The default state of CLK_TCCx_APB can be found in the Peripheral Clock Masking section (see the Related Links below).

A generic clock (GCLK_TCCx) is required to clock the TCC. This clock must be configured and enabled in the generic clock controller before using the TCC. Note that TCC0 and TCC1 share a peripheral clock generator.

The generic clocks (GCLK_TCCx) are asynchronous to the bus clock (CLK_TCCx_APB). Due to this asynchronicity, writing certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 36.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). In order to use DMA requests with this peripheral the DMAC must be configured first. Refer to *DMAC – Direct Memory Access Controller* for details.

**Related Links**
DMAC – Direct Memory Access Controller

### 36.5.5. Interrupts

The interrupt request line is connected to the Interrupt Controller. In order to use interrupt requests of this peripheral, the Interrupt Controller (NVIC) must be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller

### 36.5.6. Events

The events of this peripheral are connected to the Event System.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

### 36.5.7. Debug Operation

When the CPU is halted in debug mode, this peripheral will halt normal operation. This peripheral can be forced to continue operation during debugging - refer to the Debug Control (DBGCTRL) register for details.

Refer to DBGCTRL register for details.

### 36.5.8. Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except for the following:

- Interrupt Flag register (INTFLAG)
- Status register (STATUS)
- Period and Period Buffer registers (PER, PERBUF)
- Compare/Capture and Compare/Capture Buffer registers (CCx, CCBUFx)
- Control Waveform register (WAVE)
- Control Waveform Buffer register (WAVEBUF)
- Pattern Generation Value and Pattern Generation Value Buffer registers (PATT, PATTBUF)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 36.5.9. Analog Connections

Not applicable.

## 36.6. Functional Description

### 36.6.1. Principle of Operation

The following definitions are used throughout the documentation:

**Table 36-1.** Timer/Counter for Control Applications - Definitions

| Name | Description |
|---|---|
| TOP | The counter reaches TOP when it becomes equal to the highest value in the count sequence. The TOP value can be the same as Period (PER) or the Compare Channel 0 (CC0) register value depending on the waveform generator mode in Waveform Output Generation Operations. |
| ZERO | The counter reaches ZERO when it contains all zeroes. |
| MAX | The counter reaches maximum when it contains all ones. |
| UPDATE | The timer/counter signals an update when it reaches ZERO or TOP, depending on the direction settings. |
| Timer | The timer/counter clock control is handled by an internal source. |
| Counter | The clock control is handled externally (e.g., counting external events). |
| CC | For compare operations, the CC are referred to as "compare channels."<br>For capture operations, the CC are referred to as "capture channels." |

Each TCC instance has up to four compare/capture channels (CCx).

The counter register (COUNT), period registers with buffer (PER and PERBUF), and compare and capture registers with buffers (CCx and CCBUFx) are 16- or 24-bit registers, depending on each TCC


© 2026 Microchip Technology Inc. and its subsidiaries

instance. Each buffer register has a buffer valid (BUFV) flag that indicates when the buffer contains a new value.

Under normal operation, the counter value is continuously compared to the TOP or ZERO value to determine whether the counter has reached TOP or ZERO. In either case, the TCC can generate interrupt requests, request DMA transactions, or generate events for the Event System. In waveform generator mode, these comparisons are used to set the waveform period or pulse width.

A prescaled generic clock (GCLK_TCCx) and events from the event system can be used to control the counter. The event system is also used as a source to the input capture.

The Recoverable Fault Unit enables event controlled waveforms by acting directly on the generated waveforms of the TCC compare channels output. These events can restart, halt the timer/counter period, shorten the output pulse active time, or disable waveform output as long as the fault condition is present. This can typically be used for current sensing regulation, and zero-crossing and demagnetization re-triggering.

The MCE0 and MCE1 asynchronous event sources are shared with the Recoverable Fault Unit. Only asynchronous events are used internally when fault unit extension is enabled. For further details on how to configure asynchronous events routing, refer to *EVSYS – Event System*.

Recoverable fault sources can be filtered and/or windowed to avoid false triggering, for example from I/O pin glitches, by using digital filtering, input blanking, and qualification options. See also Recoverable Faults.

In order to support applications with different types of motor control, ballast, LED, H-bridge, power converter, and other types of power switching applications, the following independent units are implemented in some of the TCC instances as optional and successive units:

- Recoverable faults and non-recoverable faults
- Output matrix
- Dead-time insertion
- Swap
- Pattern generation

See also Figure 36-1.

The output matrix (OTMX) can distribute and route out the TCC waveform outputs across the port pins in different configurations, each optimized for different application types. The Dead-Time Insertion (DTI) unit splits the four lower OTMX outputs into two non-overlapping signals: the non-inverted low side (LS) and inverted high side (HS) of the waveform output with optional dead-time insertion between LS and HS switching. The SWAP unit can swap the LS and HS pin outputs, and can be used for fast decay motor control.

The pattern generation unit can be used to generate synchronized waveforms with constant logic level on TCC UPDATE conditions. This is useful for easy stepper motor and full bridge control.

The non-recoverable fault module enables event controlled fault protection by acting directly on the generated waveforms of the timer/counter compare channel outputs. When a non-recoverable fault condition is detected, the output waveforms are forced to a preconfigured value that is safe for the application. This is typically used for instant and predictable shut down and disabling high current or voltage drives.

The count event sources (TCE0 and TCE1) are shared with the non-recoverable fault extension. The events can be optionally filtered. If the filter options are not used, the non-recoverable faults provide an immediate asynchronous action on waveform output, even for cases where the clock is not present. For further details on how to configure asynchronous events routing, refer to section *EVSYS – Event System*.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

## 36.6.2. Basic Operation

### 36.6.2.1. Initialization

The following registers are enable-protected, meaning that they can only be written when the TCC is disabled(CTRLA.ENABLE=0):

- Control A (CTRLA) register, except Run Standby (RUNSTDBY), Enable (ENABLE) and Software Reset (SWRST) bits
- Recoverable Fault n Control registers (FCTRLA and FCTRLB)
- Waveform Extension Control register (WEXCTRL)
- Drive Control register (DRVCTRL)
- Event Control register (EVCTRL)

Enable-protected bits in the CTRLA register can be written at the same time as CTRLA.ENABLE is written to '1', but not at the same time as CTRLA.ENABLE is written to '0'. Enable-protection is denoted by the “Enable-Protected” property in the register description.

Before the TCC is enabled, it must be configured as outlined by the following steps:

1. Enable the TCC bus clock (CLK_TCCx_APB).
2. If Capture mode is required, enable the channel in Capture mode by writing a '1' to the Capture Enable bit in the Control A register (CTRLA.CPTEN).

Optionally, the following configurations can be set before enabling TCC:

1. Select PRESCALER setting in the Control A register (CTRLA.PRESCALER).
2. Select Prescaler Synchronization setting in Control A register (CTRLA.PRESCSYNC).
3. If down-counting operation is desired, write the Counter Direction bit in the Control B Set register (CTRLBSET.DIR) to '1'.
4. Select the Waveform Generation operation in the WAVE register (WAVE.WAVEGEN).
5. Select the Waveform Output Polarity in the WAVE register (WAVE.POL).
6. The waveform output can be inverted for the individual channels using the Waveform Output Invert Enable bit group in the Driver register (DRVCTRL.INVEN).

### 36.6.2.2. Enabling, Disabling, and Resetting

The TCC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The TCC is disabled by writing a zero to CTRLA.ENABLE.

The TCC is reset by writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TCC, except DBGCTRL, will be reset to their initial state, and the TCC will be disabled. Refer to Control A (CTRLA) register for details.

The TCC should be disabled before the TCC is reset to avoid undefined behavior.

### 36.6.2.3. Prescaler Selection

The GCLK_TCCx clock is fed into the internal prescaler.

The prescaler consists of a counter that counts up to the selected prescaler value, whereupon the output of the prescaler toggles.

If the prescaler value is higher than one, the counter update condition can be optionally executed on the next GCLK_TCC clock pulse or the next prescaled clock pulse. For further details, refer to the Prescaler (CTRLA.PRESCALER) and Counter Synchronization (CTRLA.PRESYNC) descriptions.

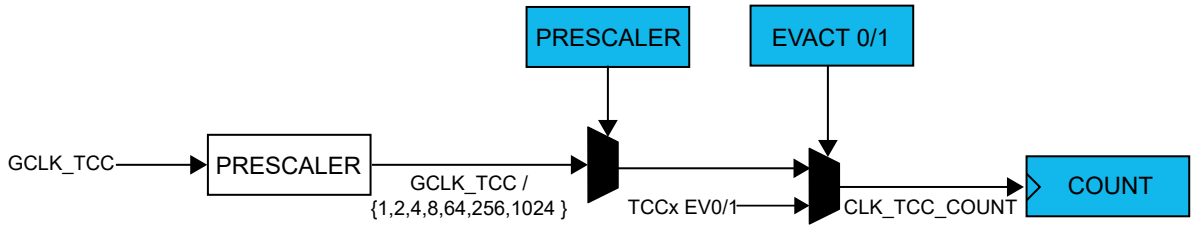
Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TCC_COUNT.

**Figure 36-2.** Prescaler




### 36.6.2.4. Counter Operation

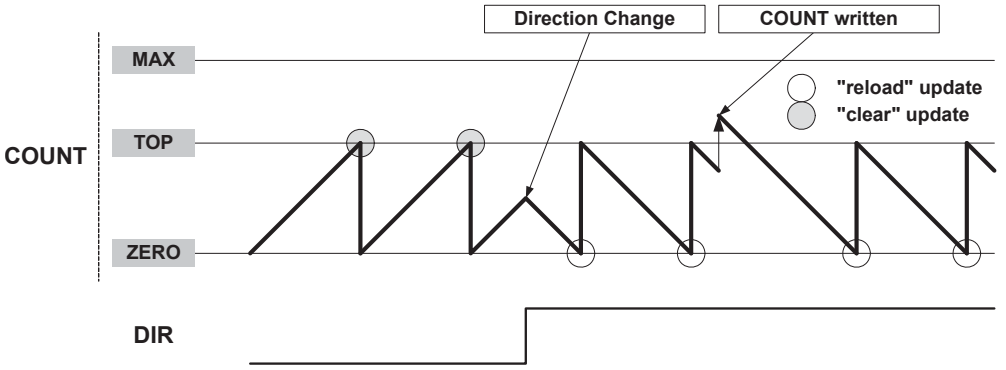
Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TCC clock input (CLK_TCC_COUNT). A counter clear or reload mark the end of current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If the bit is zero, it's counting up and one if counting down.

The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it's counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When down-counting, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

INTFLAG.OVF can be used to trigger an interrupt, a DMA request, or an event. An overflow/underflow occurrence (i.e. a compare match with TOP/ZERO) will stop counting if the One-Shot bit in the Control B register is set (CTRLBSET.ONESHOT). The One-Shot feature is explained in the Additional Features section.

**Figure 36-3.** Counter Operation




It is possible to change the counter value (by writing directly in the COUNT register) even when the counter is running. The COUNT value will always be ZERO or TOP, depending on direction set by CTRLBSET.DIR or CTRLBCLR.DIR, when starting the TCC, unless a different value has been written to it, or the TCC has been stopped at a value other than ZERO. The write access has higher priority than count, clear, or reload. The direction of the counter can also be changed during normal operation. See also Figure 36-3.

#### Stop Command

A stop command can be issued from software by using TCC Command bits in Control B Set register (CTRLBSET.CMD=0x2, STOP).


© 2026 Microchip Technology Inc. and its subsidiaries

### Pause Event Action

A pause command can be issued when the stop event action is configured in the Input Event Action 1 bits in Event Control register (EVCTRL.EVACT1=0x3, STOP).

### Re-Trigger Command and Event Action

A re-trigger command can be issued from software by using TCC Command bits in Control B Set register (CTRLBSET.CMD=0x1, RETRIGGER), or from event when the re-trigger event action is configured in the Input Event 0/1 Action bits in Event Control register (EVCTRL.EVACTn=0x1, RETRIGGER).

When the command is detected during counting operation, the counter will be reloaded or cleared, depending on the counting direction (CTRLBSET.DIR or CTRLBCLR.DIR). The Re-Trigger bit in the Interrupt Flag Status and Clear register will be set (INTFLAG.TRG). It is also possible to generate an event by writing a '1' to the Re-Trigger Event Output Enable bit in the Event Control register (EVCTRL.TRGEO). If the re-trigger command is detected when the counter is stopped, the counter will resume counting operation from the value in COUNT.

**Note:**
When a re-trigger event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACTn=0x1, RETRIGGER), enabling the counter will not start the counter. The counter will start on the next incoming event and restart on corresponding following event.

### Start Event Action

The start action can be selected in the Event Control register (EVCTRL.EVACT0=0x3, START) and can start the counting operation when previously stopped. The event has no effect if the counter is already counting. When the module is enabled, the counter operation starts when the event is received or when a re-trigger software command is applied.

**Note:**
When a start event action is configured in the Event Action bits in the Event Control register (EVCTRL.EVACT0=0x3, START), enabling the counter will not start the counter. The counter will start on the next incoming event, but it will not restart on subsequent events.

### Count Event Action

The TCC can count events. When an event is received, the counter increases or decreases the value, depending on direction settings (CTRLBSET.DIR or CTRLBCLR.DIR).

The count event action is selected by the Event Action 0 bit group in the Event Control register (EVCTRL.EVACT0=0x5, COUNT).

### Direction Event Action

The direction event action can be selected in the Event Control register (EVCTRL.EVACT1=0x2, DIR). When this event is used, the asynchronous event path specified in the event system must be configured or selected. The direction event action can be used to control the direction of the counter operation, depending on external events level. When received, the event level overrides the Direction settings (CTRLBSET.DIR or CTRLBCLR.DIR) and the direction bit value is updated accordingly.

### Increment Event Action

The increment event action can be selected in the Event Control register (EVCTRL.EVACT0=0x4, INC) and can change the counter state when an event is received. When the TCE0 event (TCCx_EV0) is received, the counter increments, whatever the direction setting (CTRLBSET.DIR or CTRLBCLR.DIR) is.

### Decrement Event Action

The decrement event action can be selected in the Event Control register (EVCTRL.EVACT1=0x4, DEC) and can change the counter state when an event is received. When the TCE1 (TCCx_EV1) event is


© 2026 Microchip Technology Inc. and its subsidiaries

received, the counter decrements, whatever the direction setting (CTRLBSET.DIR or CTRLBCLR.DIR) is.

**Non-recoverable Fault Event Action**

Non-recoverable fault actions can be selected in the Event Control register (EVCTRL.EVACTn=0x7, FAULT). When received, the counter will be stopped and the output of the compare channels is overridden according to the Driver Control register settings (DRVCTRL.NREx and DRVCTRL.NRVx). TCE0 and TCE1 must be configured as asynchronous events.

**Event Action Off**

If the event action is disabled (EVCTRL.EVACTn=0x0, OFF), enabling the counter will also start the counter.

**Related Links**

One-Shot Operation

#### 36.6.2.5. Compare Operations

By default, the Compare/Capture channel is configured for compare operations. To perform capture operations, it must be re-configured.

When using the TCC with the Compare/Capture Value registers (CCx) for compare operations, the counter value is continuously compared to the values in the CCx registers. This can be used for timer or for waveform operation.

The Channel x Compare/Capture Buffer Value (CCBUFx) registers provide double buffer capability. The double buffering synchronizes the update of the CCx register with the buffer value at the UPDATE condition or a force update command (CTRLBSET.CMD=0x3, UPDATE). For further details, refer to Double Buffering. The synchronization prevents the occurrence of odd-length, non-symmetrical pulses and ensures glitch-free output.

##### 36.6.2.5.1. Waveform Output Generation Operations

The compare channels can be used for waveform generation on output port pins. To make the waveform available on the connected pin, the following requirements must be fulfilled:

1. Choose a waveform generation mode in the Waveform Generation Operation bit in Waveform register (WAVE.WAVEGEN).
2. Optionally invert the waveform output WO[x] by writing the corresponding Waveform Output x Inversion bit in the Driver Control register (DRVCTRL.INVENx).
3. Configure the pins with the I/O Pin Controller. Refer to *PORT - I/O Pin Controller* for details.

The counter value is continuously compared with each CCx value. On a comparison match, the Match or Capture Channel x bit in the Interrupt Flag Status and Clear register (INTFLAG.MCx) will be set on the next zero-to-one transition of CLK_TCC_COUNT (see Normal Frequency Operation). An interrupt and/or event can be generated on the same condition if Match/Capture occurs, i.e. INTENSET.MCx and/or EVCTRL.MCEOx is '1'. Both interrupt and event can be generated simultaneously. The same condition generates a DMA request.

There are seven waveform configurations for the Waveform Generation Operation bit group in the Waveform register (WAVE.WAVEGEN). This will influence how the waveform is generated and impose restrictions on the top value. The configurations are:

- Normal Frequency (NFRQ)
- Match Frequency (MFRQ)
- Normal Pulse-Width Modulation (NPWM)
- Dual-slope, interrupt/event at TOP (DSTOP)
- Dual-slope, interrupt/event at ZERO (DSBOTTOM)
- Dual-slope, interrupt/event at Top and ZERO (DSBOTH)


© 2026 Microchip Technology Inc. and its subsidiaries

- Dual-slope, critical interrupt/event at ZERO (DSCRITICAL)

When using MFRQ configuration, the TOP value is defined by the CC0 register value. For the other waveform operations, the TOP value is defined by the Period (PER) register value.

For dual-slope waveform operations, the update time occurs when the counter reaches ZERO. For the other waveforms generation modes, the update time occurs on counter wraparound, on overflow, underflow, or re-trigger.

The table below shows the update counter and overflow event/interrupt generation conditions in different operation modes.

**Table 36-2.** Counter Update and Overflow Event/interrupt Conditions

| Name | Operation | TOP | Update | Output Waveform | | OVFIF/Event | |
|---|---|---|---|---|---|---|---|
| | | | | On Match | On Update | Up | Down |
| NFRQ | Normal Frequency | PER | TOP/ ZERO | Toggle | Stable | TOP | ZERO |
| MFRQ | Match Frequency | CC0 | TOP/ ZERO | Toggle | Stable | TOP | ZERO |
| NPWM | Single-slope PWM | PER | TOP/ ZERO | See section 'Output Polarity' below | | TOP | ZERO |
| DSCRITICAL | Dual-slope PWM | PER | ZERO | | | - | ZERO |
| DSBOTTOM | Dual-slope PWM | PER | ZERO | | | - | ZERO |
| DSBOTH | Dual-slope PWM | PER | TOP(1) & ZERO | | | TOP | ZERO |
| DSTOP | Dual-slope PWM | PER | ZERO | | | TOP | – |

1. The UPDATE condition on TOP only will occur when circular buffer is enabled for the channel.

**Related Links**

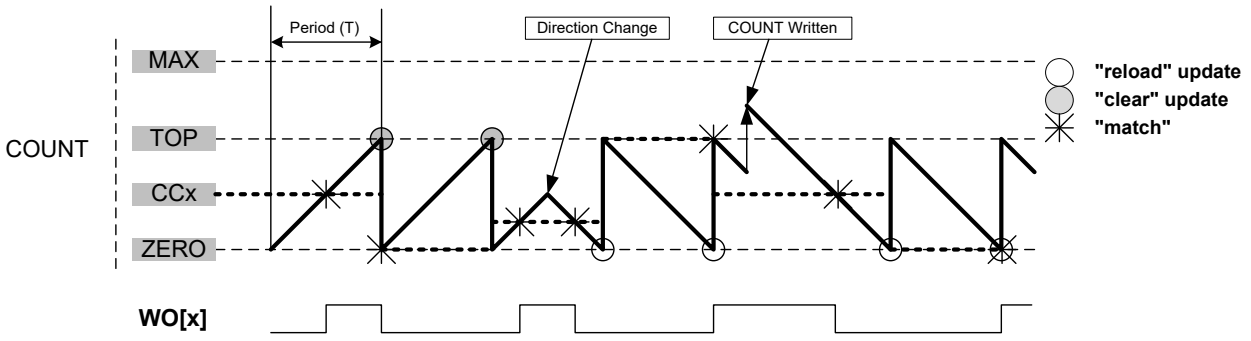
Circular Buffer

PORT - I/O Pin Controller

### 36.6.2.5.2. Normal Frequency (NFRQ)

For Normal Frequency generation, the period time (T) is controlled by the period register (PER). The waveform generation output (WO[x]) is toggled on each compare match between COUNT and CCx, and the corresponding Match or Capture Channel x Interrupt Flag (EVCTRL.MCEOx) will be set.

**Figure 36-4.** Normal Frequency Operation

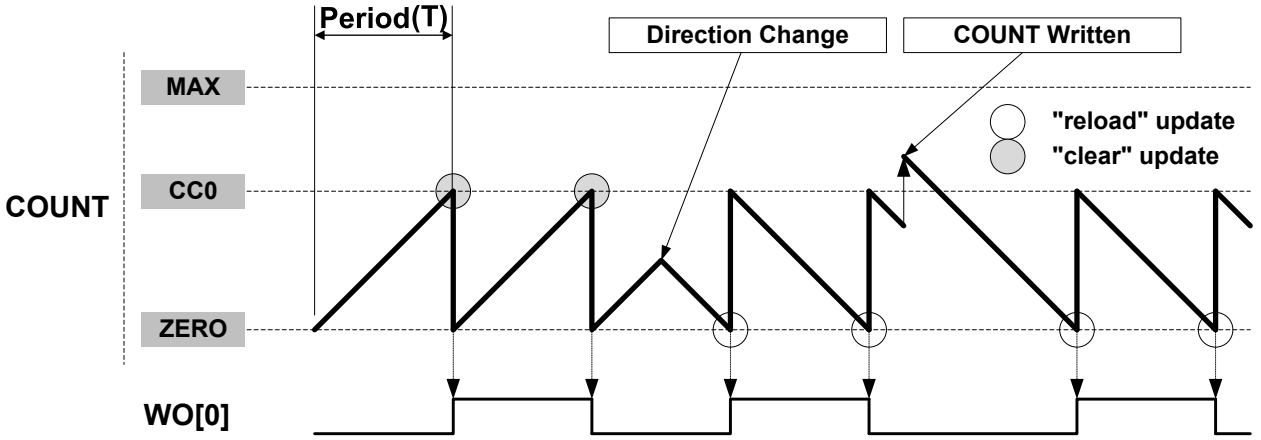



### 36.6.2.5.3. Match Frequency (MFRQ)

For Match Frequency generation, the period time (T) is controlled by CC0 register instead of PER. WO[0] toggles on each update condition.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-5.** Match Frequency Operation



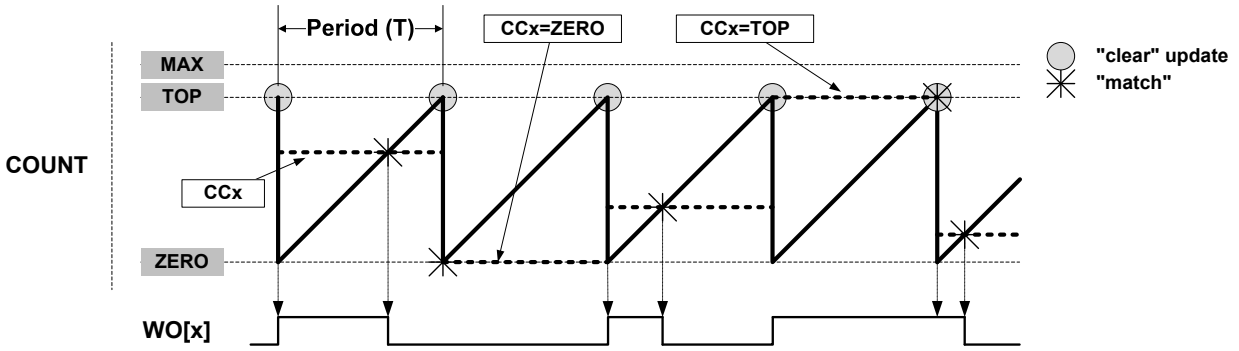

### 36.6.2.5.4. Normal Pulse-Width Modulation (NPWM)

NPWM uses single-slope PWM generation.

### 36.6.2.5.5. Single-Slope PWM Operation

For single-slope PWM generation, the period time (T) is controlled by Top value, and CCx controls the duty cycle of the generated waveform output. When up-counting, the WO[x] is set at start or compare match between the COUNT and TOP values, and cleared on compare match between COUNT and CCx register values. When down-counting, the WO[x] is cleared at start or compare match between the COUNT and ZERO values, and set on compare match between COUNT and CCx register values.

**Figure 36-6.** Single-Slope PWM Operation




The following equation calculates the exact resolution for a single-slope PWM ($R_{PWM_SS}$) waveform:

$$R_{PWM_SS} = \frac{\log(TOP+1)}{\log(2)}$$

The PWM frequency depends on the Period register value (PER) and the peripheral clock frequency ($f_{GCLK_TCC}$), and can be calculated by the following equation:

$$f_{PWM_SS} = \frac{f_{GCLK_TCC}}{N(TOP+1)}$$

Where N represents the prescaler divider used (1, 2, 4, 8, 16, 64, 256, 1024).

### 36.6.2.5.6. Dual-Slope PWM Generation

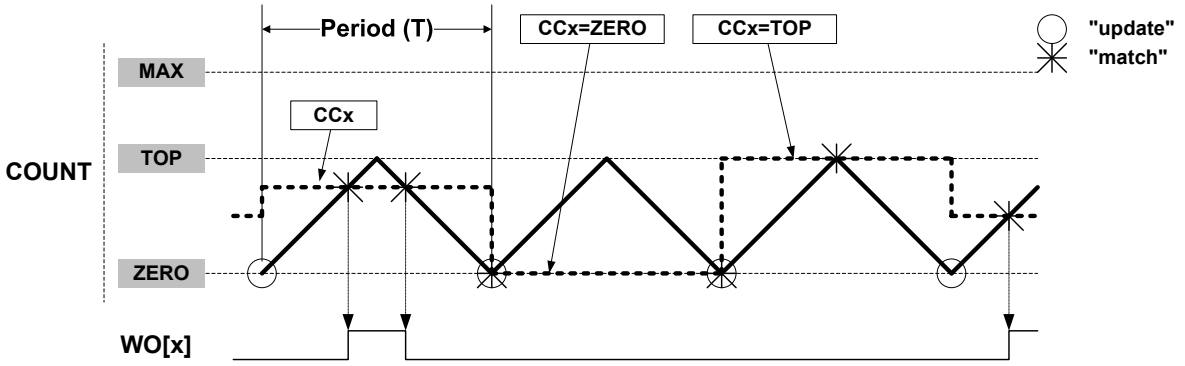
For dual-slope PWM generation, the period setting (TOP) is controlled by PER, while CCx control the duty cycle of the generated waveform output. The figure below shows how the counter repeatedly counts from ZERO to PER and then from PER to ZERO. The waveform generator output is set


© 2026 Microchip Technology Inc. and its subsidiaries

on compare match when up-counting, and cleared on compare match when down-counting. An interrupt and/or event is generated on TOP (when counting upwards) and/or ZERO (when counting up or down).

In DSBOTH operation, the circular buffer must be enabled to enable the update condition on TOP.

**Figure 36-7.** Dual-Slope Pulse Width Modulation




Using dual-slope PWM results in a lower maximum operation frequency compared to single-slope PWM generation. The period (TOP) defines the PWM resolution. The minimum resolution is 1 bit (TOP=0x00000001).

The following equation calculates the exact resolution for dual-slope PWM ($R_{PWM_DS}$):

$$R_{PWM_DS} = \frac{\log(PER+1)}{\log(2)}.$$

The PWM frequency $f_{PWM_DS}$ depends on the period setting (TOP) and the peripheral clock frequency $f_{GCLK_TCC}$, and can be calculated by the following equation:

$$f_{PWM_DS} = \frac{f_{GCLK_TCC}}{2N \cdot PER}$$

*N* represents the prescaler divider used. The waveform generated will have a maximum frequency of half of the TCC clock frequency ($f_{GCLK_TCC}$) when TOP is set to 0x00000001 and no prescaling is used.

The pulse width ($P_{PWM_DS}$) depends on the compare channel (CCx) register value and the peripheral clock frequency ($f_{GCLK_TCC}$), and can be calculated by the following equation:

$$P_{PWM_DS} = \frac{2N \cdot (TOP - CCx)}{f_{GCLK_TCC}}$$

*N* represents the prescaler divider used.

**Note:** In DSTOP, DSBOTTOM and DSBOTH operation, when TOP is lower than MAX/2, the CCx MSB bit defines the ramp on which the CCx Match interrupt or event is generated. (Rising if CCx[MSB] = 0, falling if CCx[MSB] = 1.)

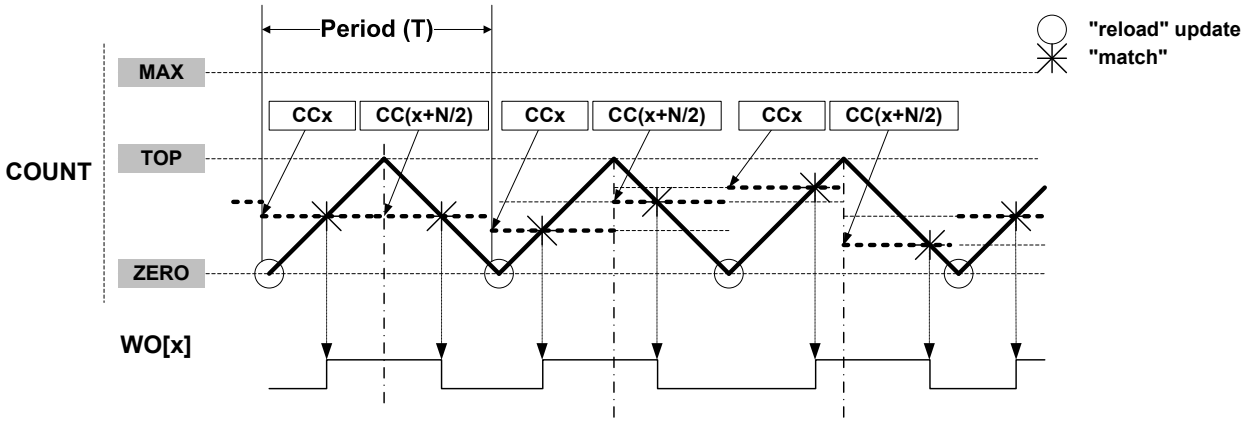
**Related Links**
Circular Buffer

### 36.6.2.5.7. Dual-Slope Critical PWM Generation

Critical mode generation allows generation of non-aligned centered pulses. In this mode, the period time is controlled by PER while CCx control the generated waveform output edge during up-counting and CC(x+CC_NUM/2) control the generated waveform output edge during down-counting.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-8.** Dual-Slope Critical Pulse Width Modulation (N=CC_NUM)




### 36.6.2.5.8. Output Polarity

The polarity (WAVE.POLx) is available in all waveform output generation. In single-slope and dual-slope PWM operation, it is possible to invert the pulse edge alignment individually on start or end of a PWM cycle for each compare channels. The table below shows the waveform output set/clear conditions, depending on the settings of timer/counter, direction, and polarity.

**Table 36-3.** Waveform Generation Set/Clear Conditions

| Waveform Generation operation | DIR | POLx | Waveform Generation Output Update | |
|---|---|---|---|---|
| | | | Set | Clear |
| Single-Slope PWM | 0 | 0 | Timer/counter matches TOP | Timer/counter matches CCx |
| | | 1 | Timer/counter matches CC | Timer/counter matches TOP |
| | 1 | 0 | Timer/counter matches CC | Timer/counter matches ZERO |
| | | 1 | Timer/counter matches ZERO | Timer/counter matches CC |
| Dual-Slope PWM | x | 0 | Timer/counter matches CC when counting up | Timer/counter matches CC when counting down |
| | | 1 | Timer/counter matches CC when counting down | Timer/counter matches CC when counting up |

In Normal and Match Frequency, the WAVE.POLx value represents the initial state of the waveform output.

### 36.6.2.6. Double Buffering

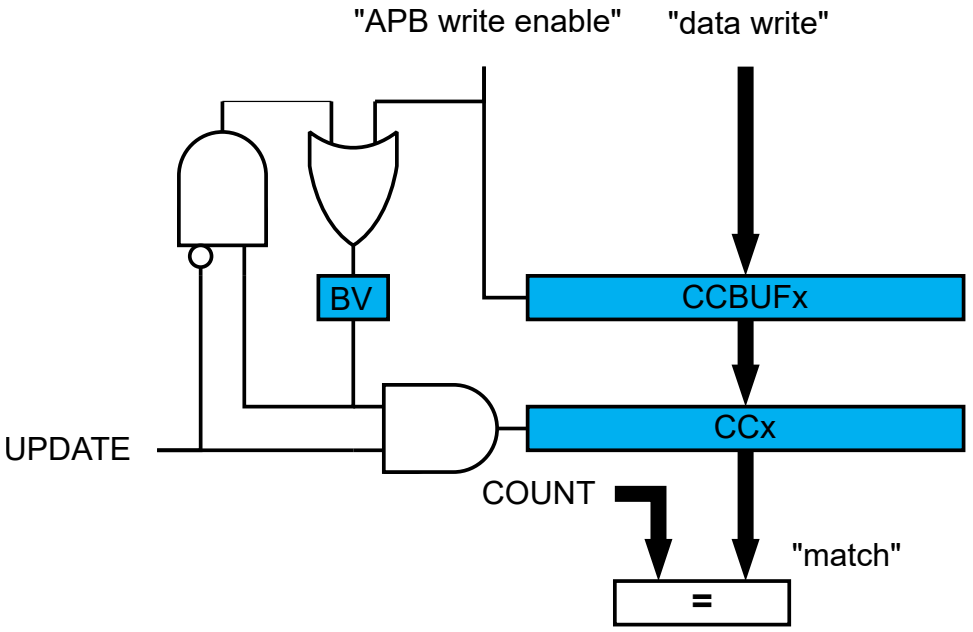
The Pattern (PATT), Waveform (WAVE), Period (PER) and Compare Channels (CCx) registers are all double buffered. Each buffer register has a buffer valid (PATTBUFV, WAVEBUFV, PERBUFV and CCBUFVx) bit in the STATUS register, which indicates that the buffer register contains a valid value that can be copied into the corresponding register. .

When the buffer valid flag bit in the STATUS register is '1' and the Lock Update bit in the CTRLB register is set to '0', (writing CTRLBCLR.LUPD to '1'), double buffering is enabled: the data from buffer registers will be copied into the corresponding register under hardware UPDATE conditions, then the buffer valid flags bit in the STATUS register are automatically cleared by hardware.
**Note:** Software update command (CTRLBSET.CMD=0x3) act independently of LUPD value.

A compare register is double buffered as in the following figure.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-9.** Compare Channel Double Buffering



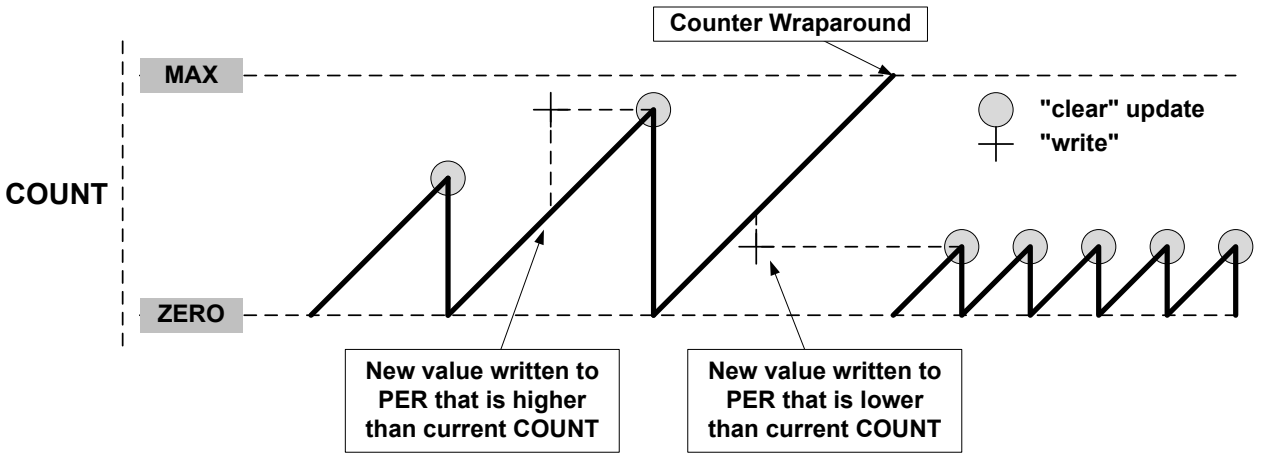

Both the registers (PATT/WAVE/PER/CCx) and corresponding buffer registers (PATTBUF/WAVEBUFV/PERBUF/CCBUFx) are available in the I/O register map, and the double buffering feature is not mandatory. The double buffering is disabled by writing a '1' to CTRLSET.LUPD.

**Note:** In NFRQ, MFRQ or PWM down-counting counter mode (CTRLBSET.DIR=1), when double buffering is enabled (CTRLBCLR.LUPD=1), PERBUF register is continuously copied into the PER independently of update conditions.

### Changing the Period

The counter period can be changed by writing a new Top value to the Period register (PER or CC0, depending on the waveform generation mode), any period update on registers (PER or CCx) is effective after the synchronization delay, whatever double buffering enabling is.

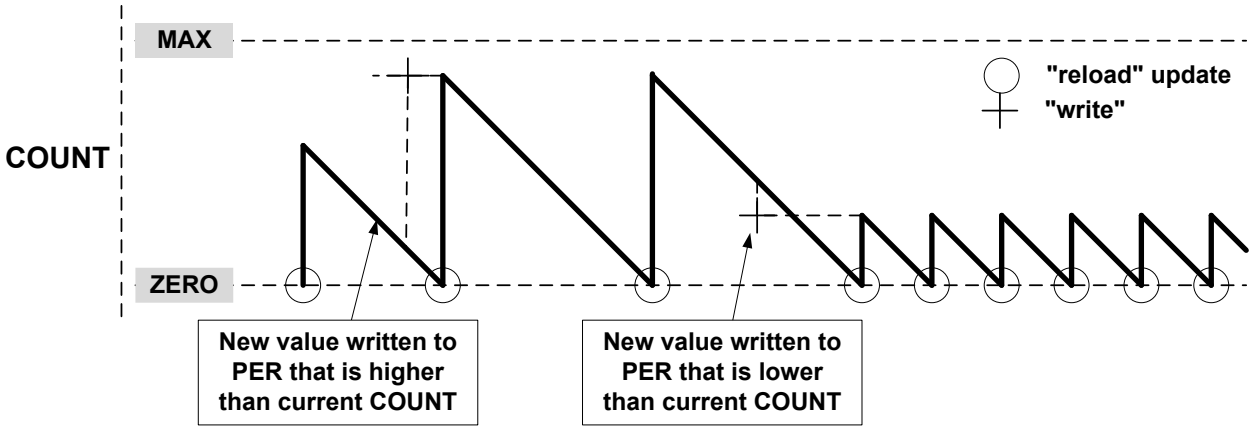
**Figure 36-10.** Unbuffered Single-Slope Up-Counting Operation





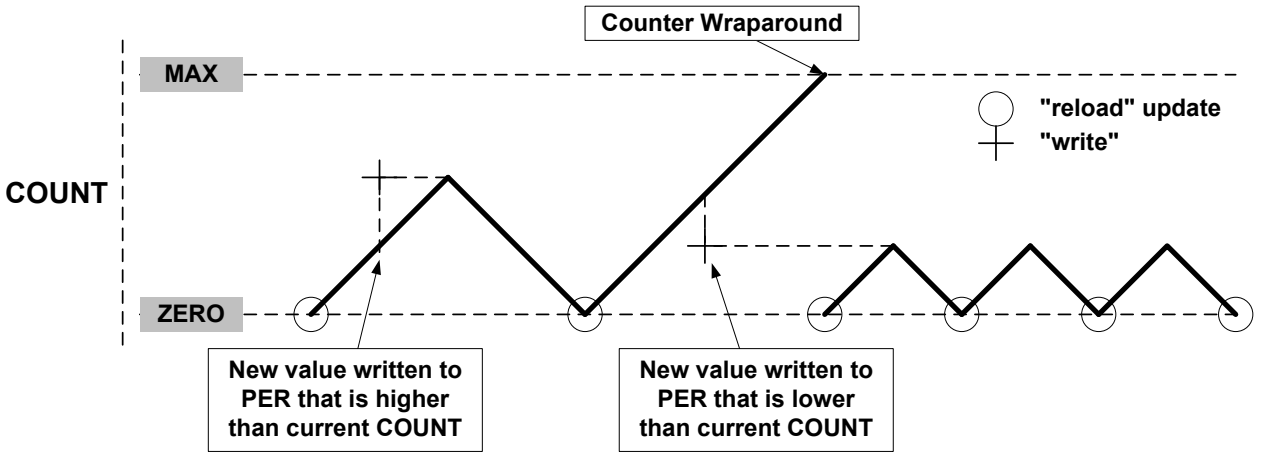
© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-11.** Unbuffered Single-Slope Down-Counting Operation



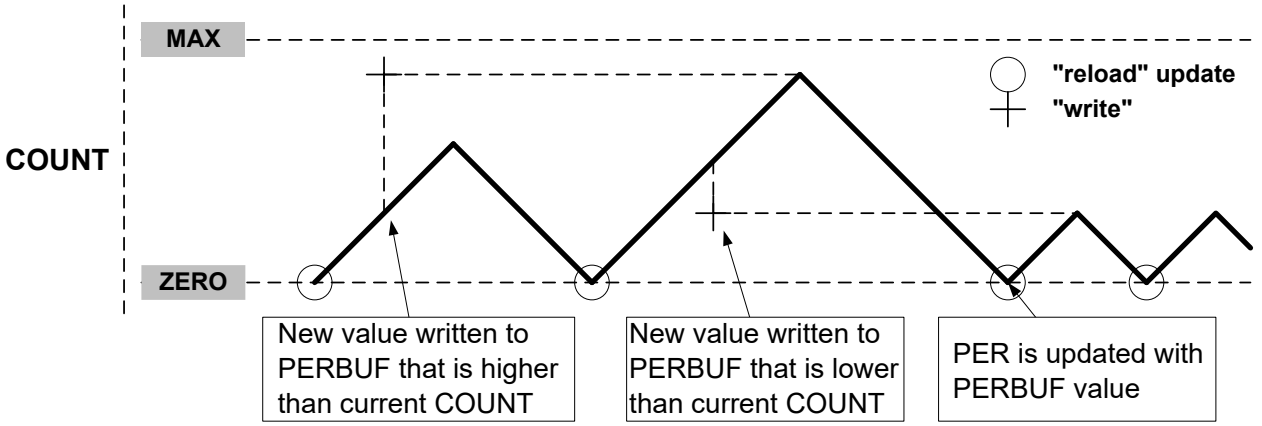

A counter wraparound can occur in any operation mode when up-counting without buffering, see Figure 36-10. COUNT and TOP are continuously compared, so when a new value that is lower than the current COUNT is written to TOP, COUNT will wrap before a compare match.

**Figure 36-12.** Unbuffered Dual-Slope Operation




When double buffering is used, the buffer can be written at any time and the counter will still maintain correct operation. The period register is always updated on the update condition, as shown in Figure 36-13. This prevents wraparound and the generation of odd waveforms.

**Figure 36-13.** Changing the Period Using Buffering





© 2026 Microchip Technology Inc. and its subsidiaries

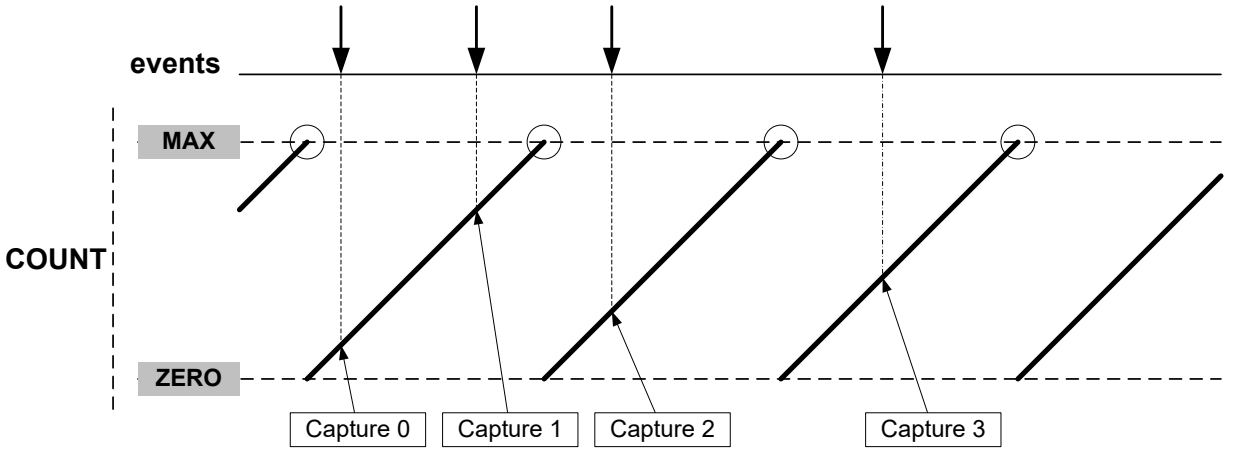
### 36.6.2.7. Capture Operations

To enable and use capture operations, the Match or Capture Channel x Event Input Enable bit in the Event Control register (EVCTRL.MCEIx) must be written to '1'. The capture channels to be used must also be enabled in the Capture Channel x Enable bit in the Control A register (CTRLA.CPTENx) before capturing can be performed.

**Event Capture Action**

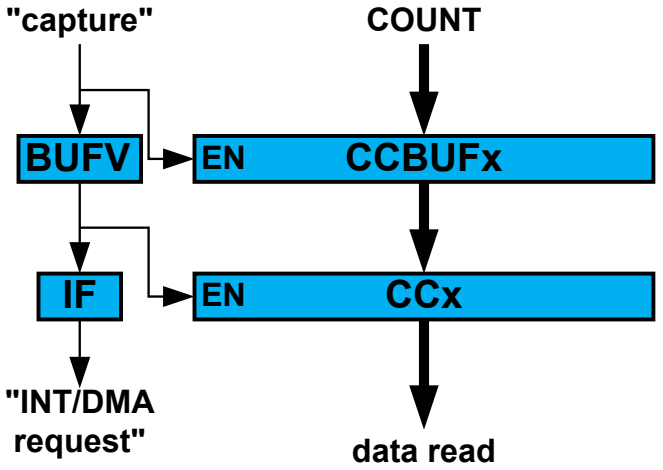
The compare/capture channels can be used as input capture channels to capture events from the Event System, and give them a timestamp. The following figure shows four capture events for one capture channel.

**Figure 36-14.** Input Capture Timing




For input capture, the buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The buffer valid flag is passed to set the CCx interrupt flag (IF) and generate the optional interrupt, event or DMA request. CCBUFx register value can't be read, all captured data must be read from CCx register.

**Figure 36-15.** Capture Double Buffering




The TCC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Buffer Valid flag (STATUS.CCBUFV) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

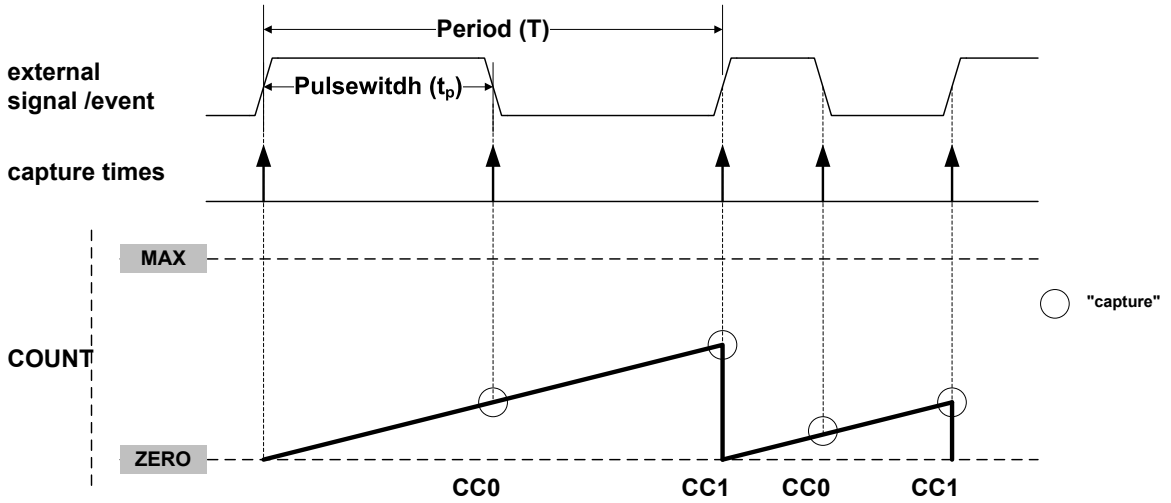
**Period and Pulse-Width (PPW) Capture Action**


© 2026 Microchip Technology Inc. and its subsidiaries

The TCC can perform two input captures and restart the counter on one of the edges. This enables the TCC to measure the pulse-width and period and to characterize the frequency *f* and *dutyCycle* of an input signal:

$$f = \frac{1}{T} \quad , \quad dutyCycle = \frac{t_p}{T}$$

**Figure 36-16.** PWP Capture




Selecting PWP or PPW in the Timer/Counter Event Input 1 Action bit group in the Event Control register (EVCTRL.EVACT1) enables the TCC to perform one capture action on the rising edge and the other one on the falling edge. When using PPW (period and pulse-width) event action, period *T* will be captured into CC0 and the pulse-width $t_p$ into CC1. The PWP (Pulse-width and Period) event action offers the same functionality, but *T* will be captured into CC1 and $t_p$ into CC0.

The Timer/Counter Event x Invert Enable bit in Event Control register (EVCTRL.TCEINVx) is used for event source x to select whether the wraparound should occur on the rising edge or the falling edge. If EVCTRL.TCEINVx=1, the wraparound will happen on the falling edge.

The corresponding capture is done only if the channel is enabled in capture mode (CTRLA.CPTENx=1). If not, the capture action will be ignored and the channel will be enabled in compare mode of operation. When only one of these channel is required, the other channel can be used for other purposes.

The TCC can detect capture overflow of the input capture channels: When a new capture event is detected while the INTFLAG.MCx is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

**Note:** When up-counting (CTRLBSET.DIR=0), counter values lower than 1 cannot be captured in Capture Minimum mode (FCTRLn.CAPTURE=CAPTMIN). To capture the full range including value 0, the TCC must be configured in down-counting mode (CTRLBSET.DIR=0).

**Note:** In dual-slope PWM operation, and when TOP is lower than MAX/2, the CCx MSB captures the CTRLB.DIR state to identify the ramp on which the capture has been done. For rising ramps CCx[MSB] is zero, for falling ramps CCx[MSB]=1.

## 36.6.3. Additional Features

### 36.6.3.1. One-Shot Operation

When one-shot is enabled, the counter automatically stops on the next counter overflow or underflow condition. When the counter is stopped, the Stop bit in the Status register (STATUS.STOP) is set and the waveform outputs are set to the value defined by DRVCTRL.NREx and DRVCTRL.NRVx.

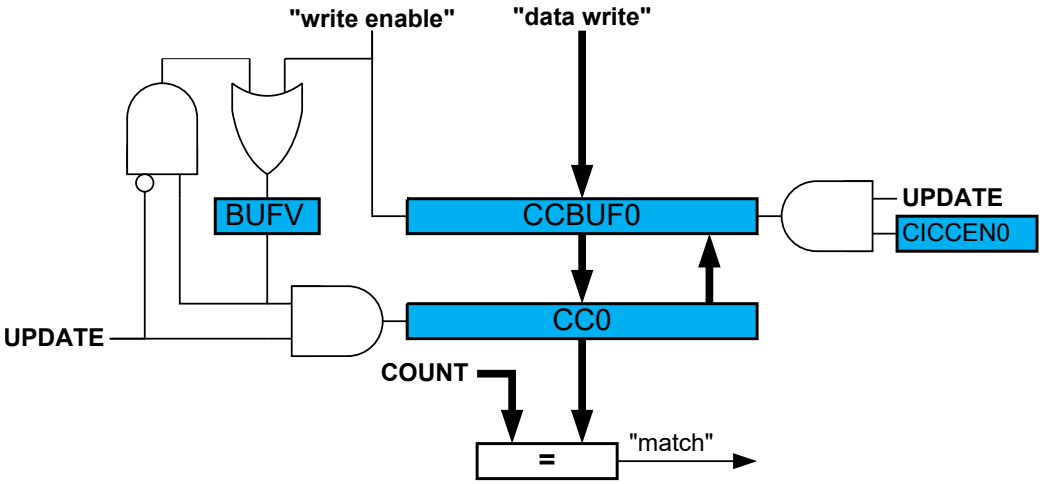

© 2026 Microchip Technology Inc. and its subsidiaries

One-shot operation can be enabled by writing a '1' to the One-Shot bit in the Control B Set register (CTRLBSET.ONESHOT) and disabled by writing a '1' to CTRLBCLR.ONESHOT. When enabled, the TCC will count until an overflow or underflow occurs and stop counting. The one-shot operation can be restarted by a re-trigger software command, a re-trigger event or a start event. When the counter restarts its operation, STATUS.STOP is automatically cleared.

### 36.6.3.2. Circular Buffer

The Period register (PER) and the Compare Channels register (CC0 to CC3) support circular buffer operation. When circular buffer operation is enabled, the PER or CCx values are copied into the corresponding buffer registers at each update condition. Circular buffering is dedicated to RAMP2, RAMP2A, and DSBOTH operations.

**Figure 36-17.** Circular Buffer on Channel 0




### 36.6.3.3. Dithering Operation

The TCC supports dithering on Pulse-width or Period on a 16, 32 or 64 PWM cycles frame.

Dithering consists in adding some extra clocks cycles in a frame of several PWM cycles, and can improve the accuracy of the *average* output pulse width and period. The extra clock cycles are added on some of the compare match signals, one at a time, through a "blue noise" process that minimizes the flickering on the resulting dither patterns.

Dithering is enabled by writing the corresponding configuration in the Enhanced Resolution bits in CTRLA register (CTRLA.RESOLUTION):

- DITH4 enable dithering every 16 PWM frames
- DITH5 enable dithering every 32 PWM frames
- DITH6 enable dithering every 64 PWM frames

The DITHERCY bits of COUNT, PER and CCx define the number of extra cycles to add into the frame (DITHERCY bits from the respective COUNT, PER or CCx registers). The remaining bits of COUNT, PER, CCx define the compare value itself.

The pseudo code, giving the extra cycles insertion regarding the cycle is:

```
int extra_cycle(resolution, dithercy, cycle){
  int MASK;
  int value
  switch (resolution){
    DITH4: MASK = 0x0f;
    DITH5: MASK = 0x1f;
    DITH6: MASK = 0x3f;
  }
  value = cycle * dithercy;
  if (((MASK & value) + dithercy) > MASK)
```


© 2026 Microchip Technology Inc. and its subsidiaries

```
    return 1;
 return 0;
}
```

#### Dithering on Period

Writing DITHERCY in PER will lead to an average PWM period configured by the following formulas.

DITH4 mode:

$$PwmPeriod = \left(\frac{\mathrm{DITHERCY}}{16} + \mathrm{PER}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

**Note:** If DITH4 mode is enabled, the last 4 significant bits from PER/CCx or COUNT register correspond to the DITHERCY value, rest of the bits corresponds to PER/CCx or COUNT value.

DITH5 mode:

$$PwmPeriod = \left(\frac{\mathrm{DITHERCY}}{32} + \mathrm{PER}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

DITH6 mode:

$$PwmPeriod = \left(\frac{\mathrm{DITHERCY}}{64} + \mathrm{PER}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

#### Dithering on Pulse Width

Writing DITHERCY in CCx will lead to an average PWM pulse width configured by the following formula.

DITH4 mode:

$$PwmPulseWidth = \left(\frac{\mathrm{DITHERCY}}{16} + \mathrm{CCx}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

DITH5 mode:

$$PwmPulseWidth = \left(\frac{\mathrm{DITHERCY}}{32} + \mathrm{CCx}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

DITH6 mode:

$$PwmPulseWidth = \left(\frac{\mathrm{DITHERCY}}{64} + \mathrm{CCx}\right)\left(\frac{1}{f_{\mathrm{GCLK_TCC}}}\right)$$

**Note:** The PWM period will remain static in this case.

### 36.6.3.4. Ramp Operations

Three ramp operation modes are supported. All of them require the timer/counter running in single-slope PWM generation. The ramp mode is selected by writing to the Ramp Mode bits in the Waveform Control register (WAVE.RAMP).

#### RAMP1 Operation

This is the default PWM operation, described in Single-Slope PWM Generation.

#### RAMP2 Operation

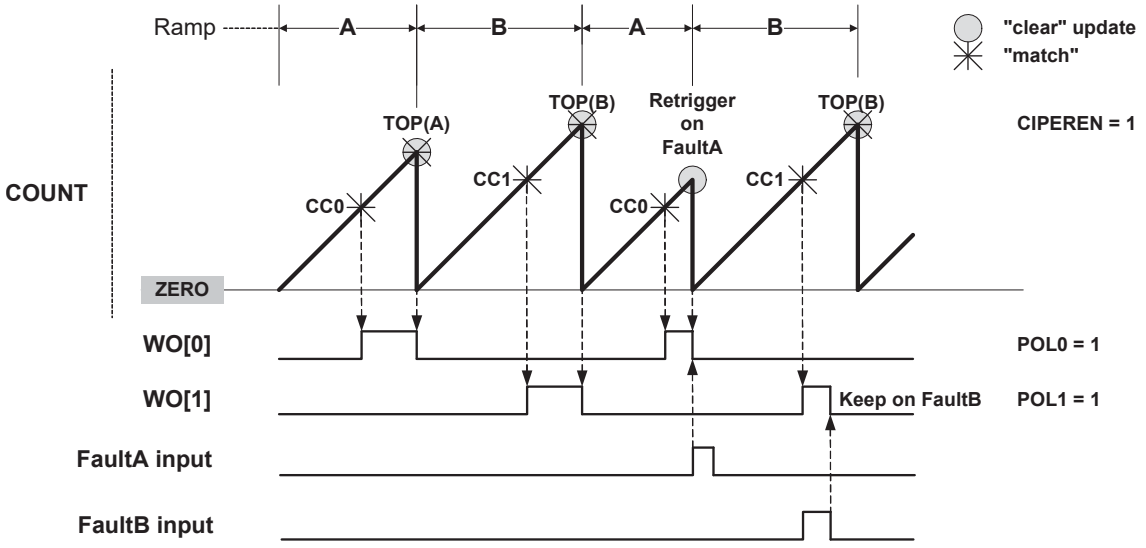
These operation modes are dedicated for power factor correction (PFC), Half-Bridge and Push-Pull SMPS topologies, where two consecutive timer/counter cycles are interleaved, see Figure 36-18. In cycle A, odd channel output is disabled, and in cycle B, even channel output is disabled. The ramp index changes after each update, but can be software modified using the Ramp index command bits in Control B Set register (CTRLBSET.IDXCMD).


© 2026 Microchip Technology Inc. and its subsidiaries

### Standard RAMP2 (RAMP2) Operation

Ramp A and B periods are controlled by the PER register value. The PER value can be different on each ramp by the Circular Period buffer option in the Wave register (WAVE.CIPEREN=1). This mode uses a two-channel TCC to generate two output signals, or one output signal with another CC channel enabled in capture mode.

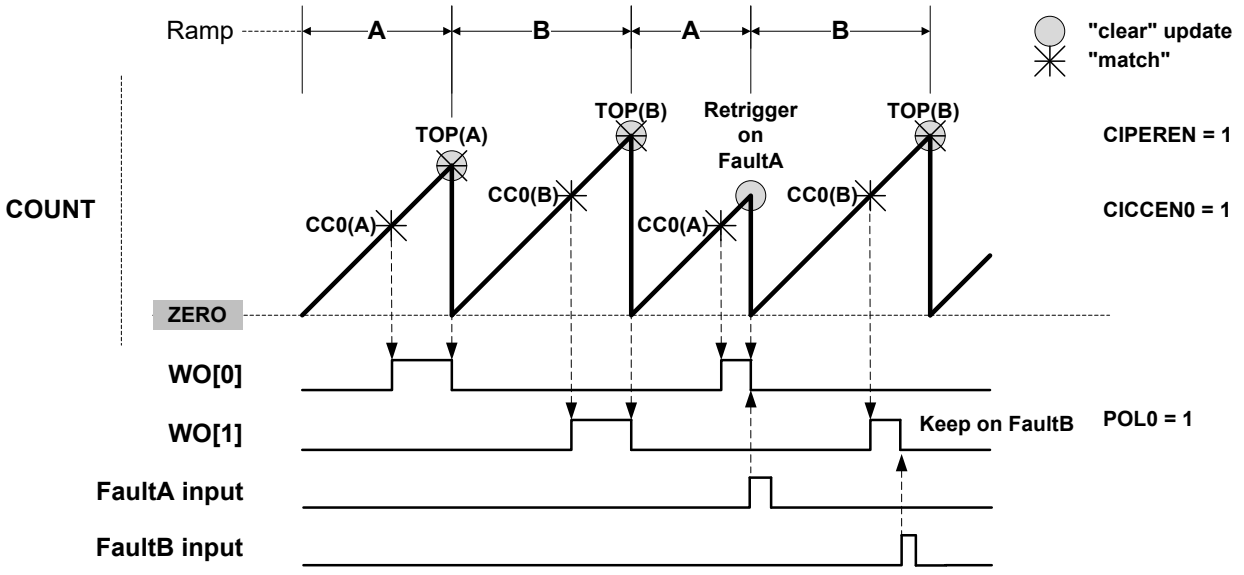
**Figure 36-18.** RAMP2 Standard Operation




### Alternate RAMP2 (RAMP2A) Operation

Alternate RAMP2 operation is similar to RAMP2, but CC0 controls both WO[0] and WO[1] waveforms when the corresponding circular buffer option is enabled (CIPEREN=1). The waveform polarity is the same on both outputs. Channel 1 can be used in capture mode.

**Figure 36-19.** RAMP2 Alternate Operation




### Critical RAMP2 (RAMP2C) Operation

Critical RAMP2 operation provides a way to cover RAMP2 operation requirements without the update constraint associated with the use of circular buffers. In this mode, CC0 is controlling the period of ramp A and PER is controlling the period of ramp B. When using more than two channels,


© 2026 Microchip Technology Inc. and its subsidiaries

WO[0] output is controlled by CC2 (HIGH) and CC0 (LOW). On TCC with 2 channels, a pulse on WO[0] will last the entire period of ramp A, if WAVE.POL0=0.

**Figure 36-20.** RAMP2 Critical Operation With More Than 2 Channels

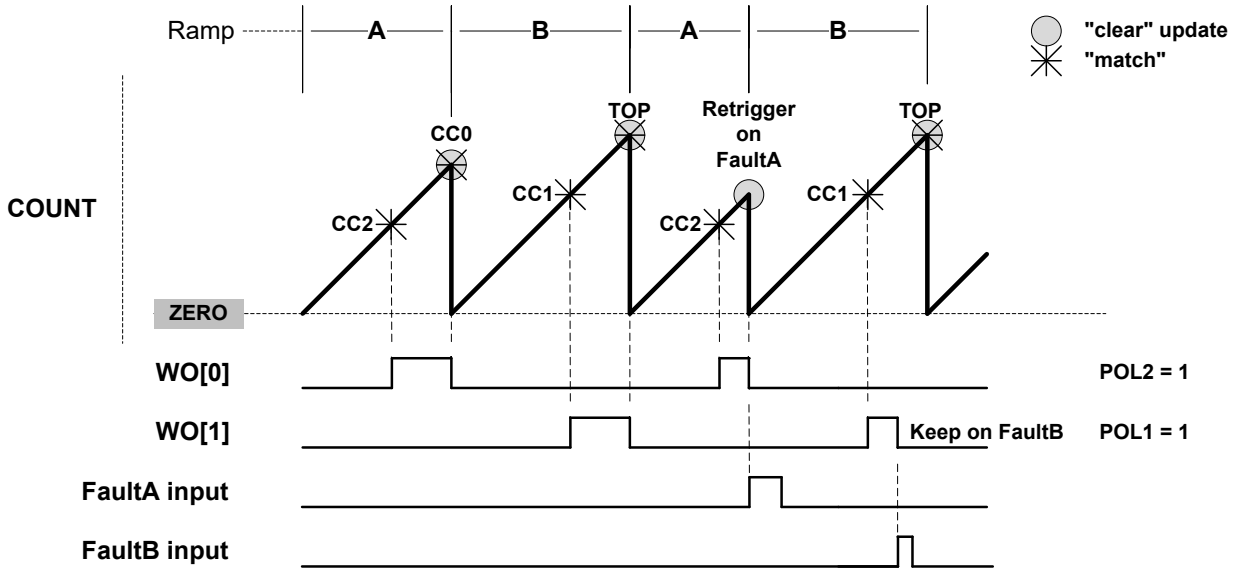
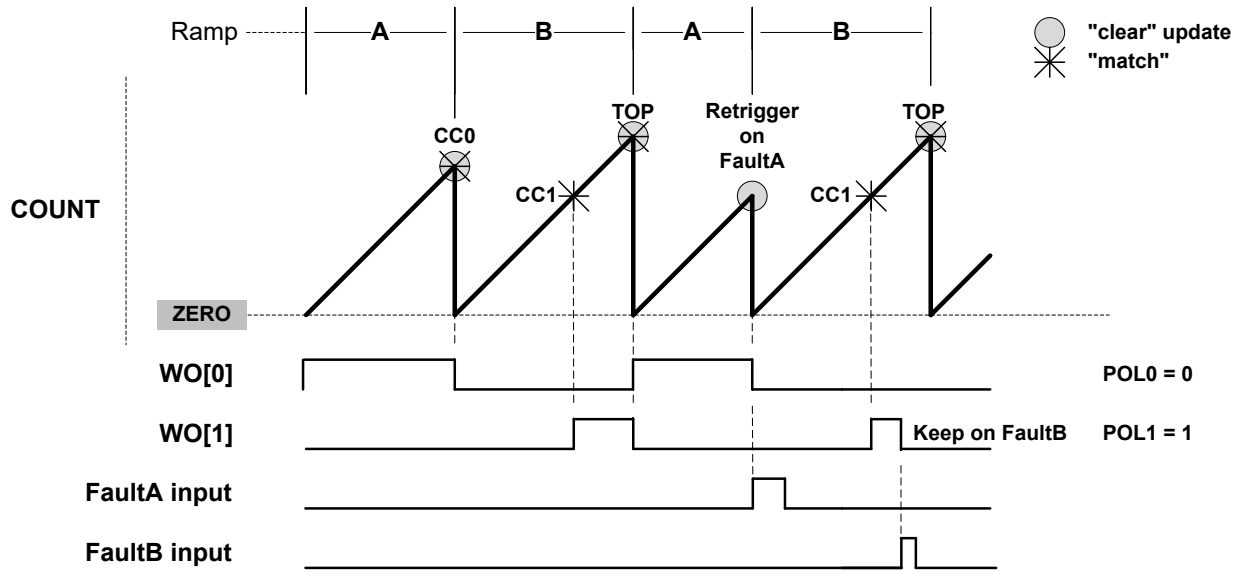



**Figure 36-21.** RAMP2 Critical Operation With 2 Channels




### 36.6.3.5. Recoverable Faults

Recoverable faults can restart or halt the timer/counter. Two faults, called Fault A and Fault B, can trigger recoverable fault actions on the compare channels CC0 and CC1 of the TCC. The compare channels' outputs can be clamped to inactive state either as long as the fault condition is present, or from the first valid fault condition detection on until the end of the timer/counter cycle.

#### Fault Inputs

The first two channel input events (TCCxMC0 and TCCxMC1) can be used as Fault A and Fault B inputs, respectively. Event system channels connected to these fault inputs must be configured as asynchronous. The TCC must work in a PWM mode.


© 2026 Microchip Technology Inc. and its subsidiaries

## Fault Filtering

There are three filters available for each input Fault A and Fault B. They are configured by the corresponding Recoverable Fault n Configuration registers (FCTRLA and FCTRLB). The three filters can either be used independently or in any combination.

**Input Filtering**

By default, the event detection is asynchronous. When the event occurs, the fault system will immediately and asynchronously perform the selected fault action on the compare channel output, also in device power modes where the clock is not available. To avoid false fault detection on external events (e.g. due to a glitch on an I/O port) a digital filter can be enabled and configured by the Fault B Filter Value bits in the Fault n Configuration registers (FCTRLn.FILTERVAL). If the event width is less than FILTERVAL (in clock cycles), the event will be discarded. A valid event will be delayed by FILTERVAL clock cycles.

**Fault Blanking**

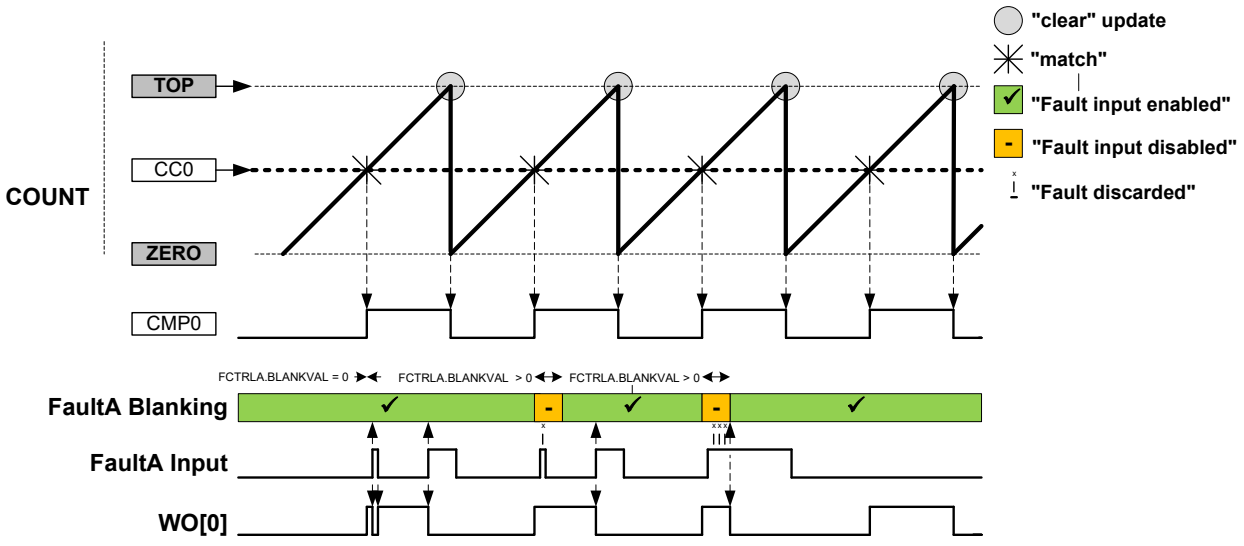
This ignores any fault input for a certain time just after a selected waveform output edge. This can be used to prevent false fault triggering due to signal bouncing, as shown in the figure below. Blanking can be enabled by writing an edge triggering configuration to the Fault n Blanking Mode bits in the Recoverable Fault n Configuration register (FCTRLn.BLANK). The desired duration of the blanking must be written to the Fault n Blanking Time bits (FCTRLn.BLANKVAL).
The blanking time $t_b$is calculated by

$$t_b = \frac{1 + \mathrm{BLANKVAL}}{f_{\mathrm{GCLK_TCCx_PRESC}}}$$

Here, $f_{\mathrm{GCLK_TCCx_PRESC}}$ is the frequency of the prescaled peripheral clock frequency $f_{\mathrm{GCLK_TCCx}}$.

The maximum blanking time (FCTRLn.BLANKVAL= 255) at $f_{\mathrm{GCLK_TCCx}}$=96MHz is 2.67µs (no prescaler) or 170µs (prescaling). For $f_{\mathrm{GCLK_TCCx}}$=1MHz, the maximum blanking time is either 170µs (no prescaling) or 10.9ms (prescaling enabled).

**Figure 36-22.** Fault Blanking in RAMP1 Operation with Inverted Polarity




**Fault Qualification**

This is enabled by writing a '1' to the Fault n Qualification bit in the Recoverable Fault n Configuration register (FCTRLn.QUAL). When the recoverable fault qualification is enabled (FCTRLn.QUAL=1), the fault input is disabled all the time the corresponding channel output has an inactive level, as shown in the figures below.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-23.** Fault Qualification in RAMP1 Operation

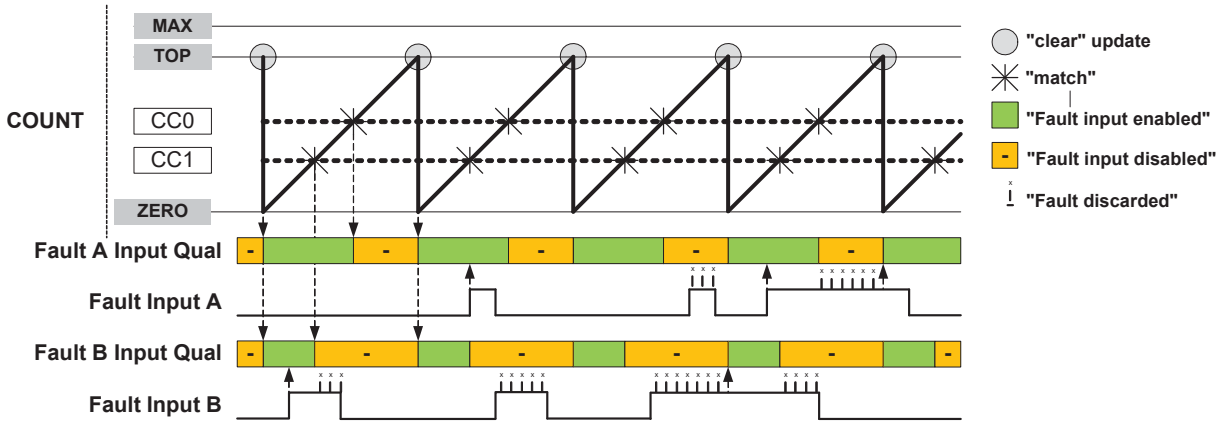
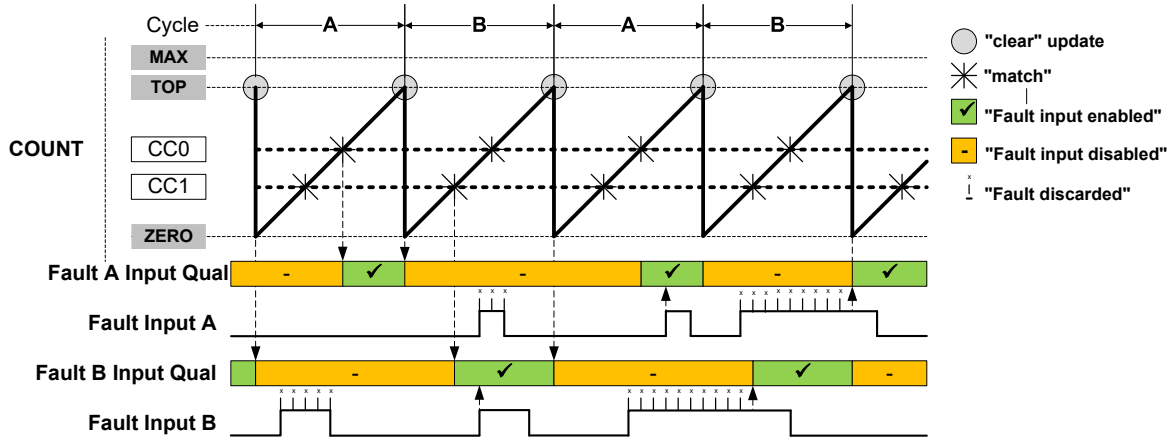



**Figure 36-24.** Fault Qualification in RAMP2 Operation with Inverted Polarity

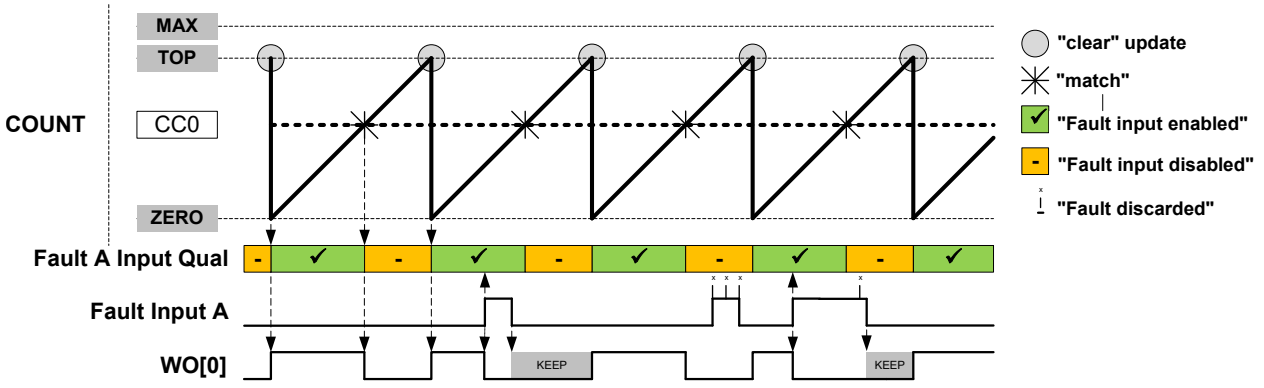



## Fault Actions

Different fault actions can be configured individually for Fault A and Fault B. Most fault actions are not mutually exclusive; hence two or more actions can be enabled at the same time to achieve a result that is a combination of fault actions.

**Keep Action** This is enabled by writing the Fault n Keeper bit in the Recoverable Fault n Configuration register (FCTRLn.KEEP) to '1'. When enabled, the corresponding channel output will be clamped to zero as long as the fault condition is present. The clamp will be released on the start of the first cycle after the fault condition is no longer present, see next Figure.

**Figure 36-25.** Waveform Generation with Fault Qualification and Keep Action





© 2026 Microchip Technology Inc. and its subsidiaries

**Restart Action**

This is enabled by writing the Fault n Restart bit in Recoverable Fault n Configuration register (FCTRLn.RESTART) to '1'. When enabled, the timer/counter will be restarted as soon as the corresponding fault condition is present. The ongoing cycle is stopped and the timer/counter starts a new cycle, see Figure 36-26. In Ramp 1 mode, when the new cycle starts, the compare outputs will be clamped to inactive level as long as the fault condition is present.

**Note:** For RAMP2 operation, when a new timer/counter cycle starts the cycle index will change automatically, see Figure 36-27. Fault A and Fault B are qualified only during the cycle A and cycle B respectively: Fault A is disabled during cycle B, and Fault B is disabled during cycle A.

**Figure 36-26.** Waveform Generation in RAMP1 mode with Restart Action

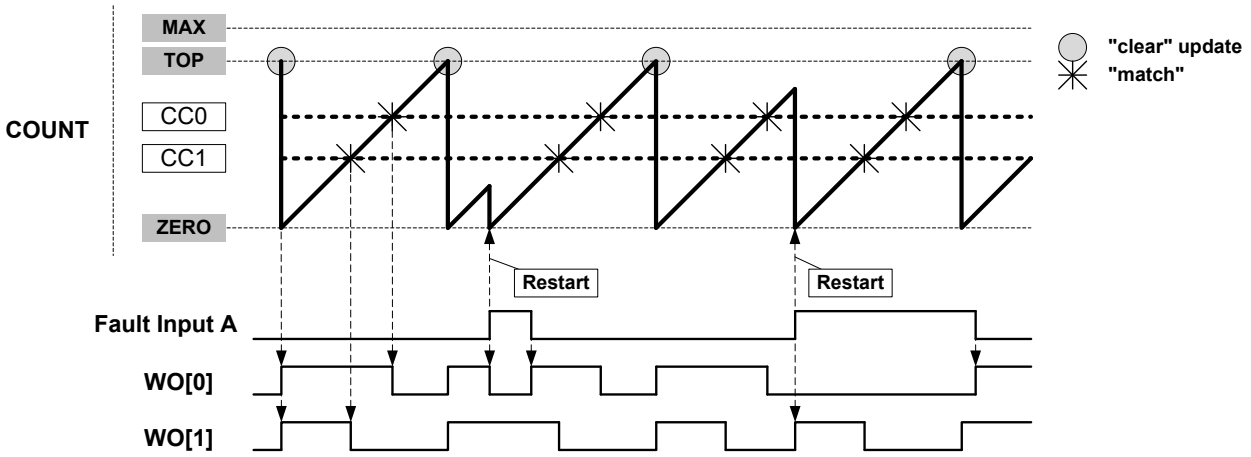
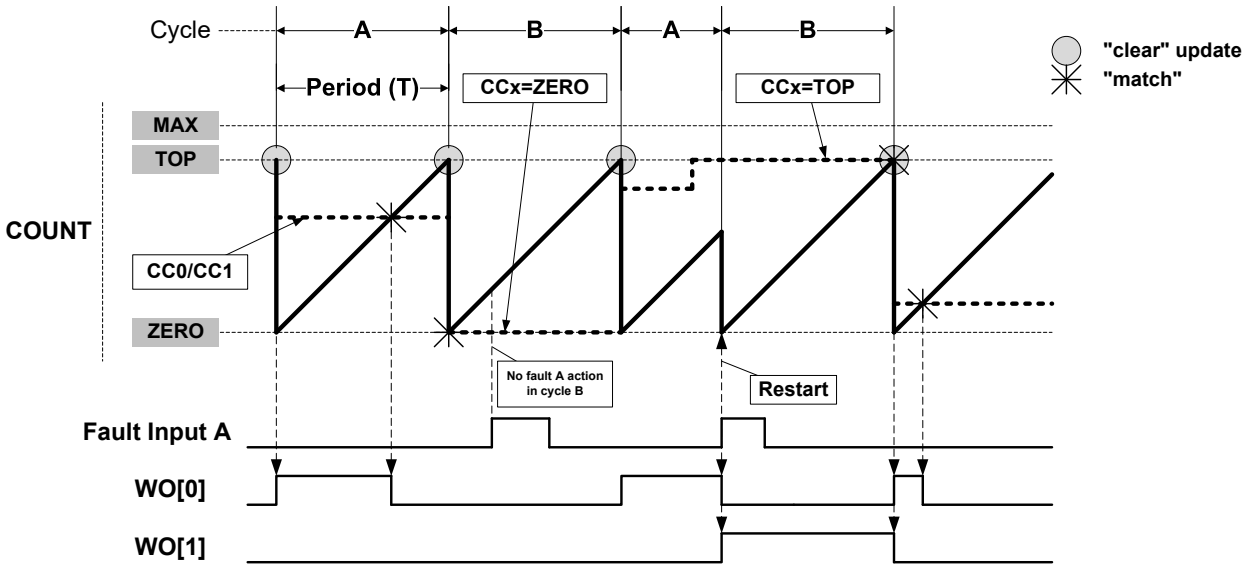



**Figure 36-27.** Waveform Generation in RAMP2 mode with Restart Action




**Capture Action**

Several capture actions can be selected by writing the Fault n Capture Action bits in the Fault n Control register (FCTRLn.CAPTURE). When one of the capture operations is selected, the counter value is captured when the fault occurs. These capture operations are available:

- CAPT - the equivalent to a standard capture operation, for further details refer to Capture Operations


© 2026 Microchip Technology Inc. and its subsidiaries

- CAPTMIN - gets the minimum time stamped value: on each new local minimum captured value, an event or interrupt is issued.
- CAPTMAX - gets the maximum time stamped value: on each new local maximum captured value, an event or interrupt (IT) is issued, see Figure 36-28.
- LOCMIN - notifies by event or interrupt when a local minimum captured value is detected.
- LOCMAX - notifies by event or interrupt when a local maximum captured value is detected.
- DERIV0 - notifies by event or interrupt when a local extreme captured value is detected, see Figure 36-29.

*CCx Content:*

In CAPTMIN and CAPTMAX operations, CCx keeps the respective extremum captured values, see Figure 36-28. In LOCMIN, LOCMAX or DERIV0 operation, CCx follows the counter value at fault time, see Figure 36-29.

Before enabling CAPTMIN or CAPTMAX mode of capture, the user must initialize the corresponding CCx register value to a value different from zero (for CAPTMIN) top (for CAPTMAX). If the CCx register initial value is zero (for CAPTMIN) top (for CAPTMAX), no captures will be performed using the corresponding channel.

*MCx Behaviour:*

In LOCMIN and LOCMAX operation, capture is performed on each capture event. The MCx interrupt flag is set only when the captured value is above or equal (for LOCMIN) or below or equal (for LOCMAX) to the previous captured value. So interrupt flag is set when a new relative local Minimum (for CAPTMIN) or Maximum (for CAPTMAX) value has been detected. DERIV0 is equivalent to an OR function of (LOCMIN, LOCMAX).

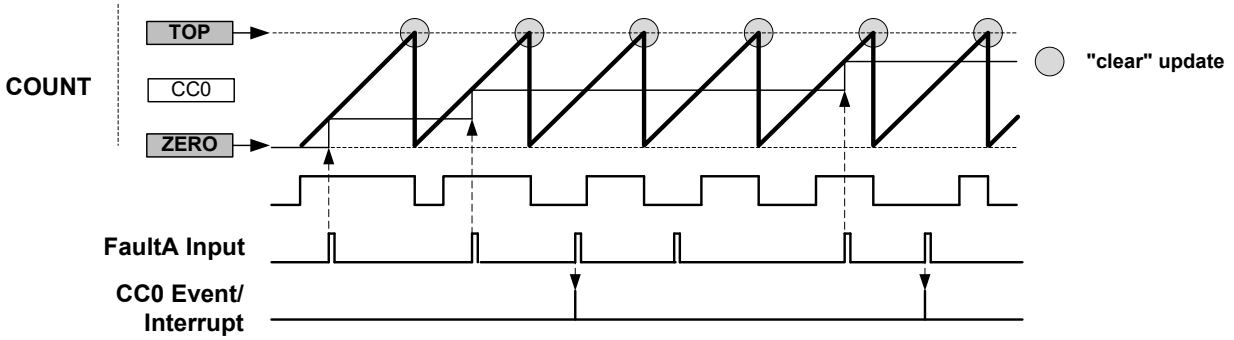
In CAPT operation, capture is performed on each capture event. The MCx interrupt flag is set on each new capture.

In CAPTMIN and CAPTMAX operation, capture is performed only when on capture event time, the counter value is lower (for CAPTMIN) or higher (for CAPMAX) than the last captured value. The MCx interrupt flag is set only when on capture event time, the counter value is higher or equal (for CAPTMIN) or lower or equal (for CAPTMAX) to the value captured on the previous event. So interrupt flag is set when a new absolute local Minimum (for CAPTMIN) or Maximum (for CAPTMAX) value has been detected.

*Interrupt Generation*

In CAPT mode, an interrupt is generated on each filtered Fault n and each dedicated CCx channel capture counter value. In other modes, an interrupt is only generated on an extreme captured value.

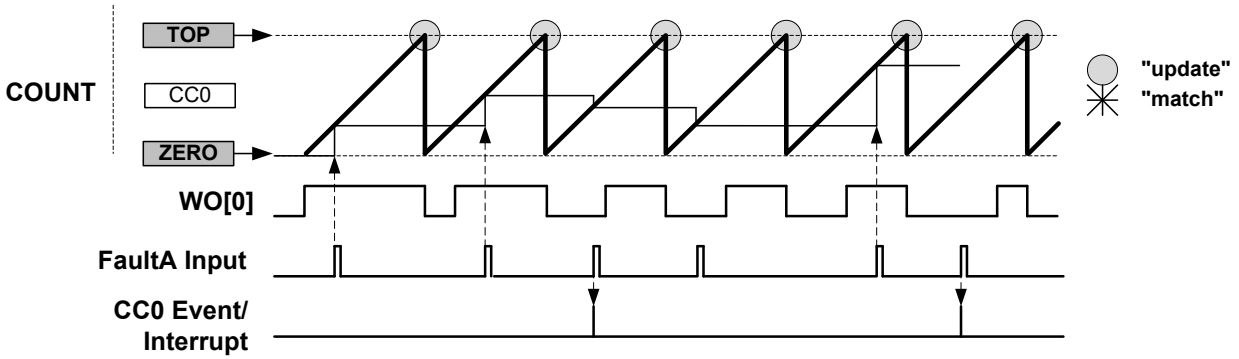
**Figure 36-28.** Capture Action “CAPTMAX”





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-29.** Capture Action “DERIV0”




**Hardware Halt Action**

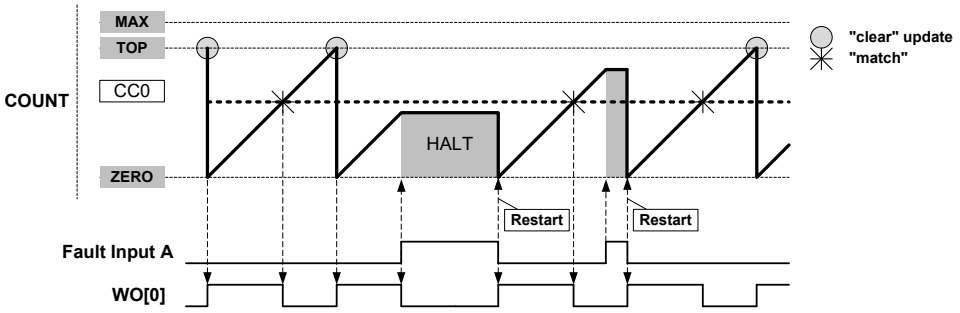
This is configured by writing 0x1 to the Fault n Halt mode bits in the Recoverable Fault n Configuration register (FCTRLn.HALT). When enabled, the timer/counter is halted and the cycle is extended as long as the corresponding fault is present.

The next figure ('Waveform Generation with Halt and Restart Actions') shows an example where both restart action and hardware halt action are enabled for Fault A. The compare channel 0 output is clamped to inactive level as long as the timer/counter is halted. The timer/counter resumes the counting operation as soon as the fault condition is no longer present. As the restart action is enabled in this example, the timer/counter is restarted after the fault condition is no longer present.

The figure after that ('Waveform Generation with Fault Qualification, Halt, and Restart Actions') shows a similar example, but with additionally enabled fault qualification. Here, counting is resumed after the fault condition is no longer present.

Note that in RAMP2 and RAMP2A operations, when a new timer/counter cycle starts, the cycle index will automatically change.

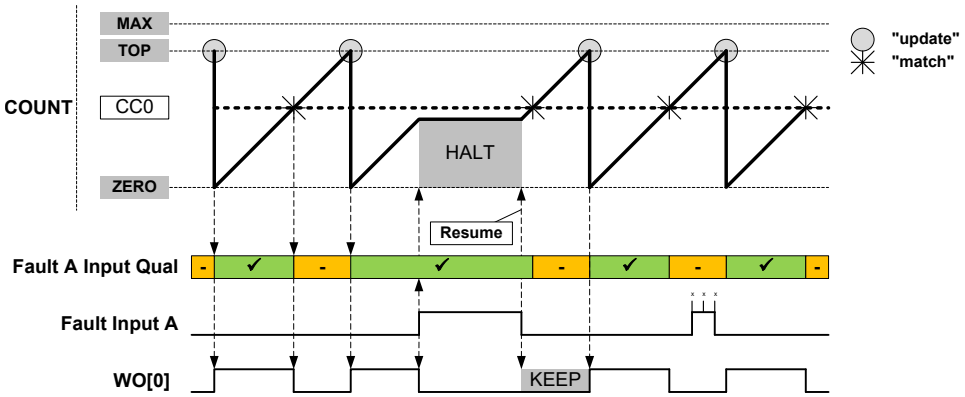
**Figure 36-30.** Waveform Generation with Halt and Restart Actions





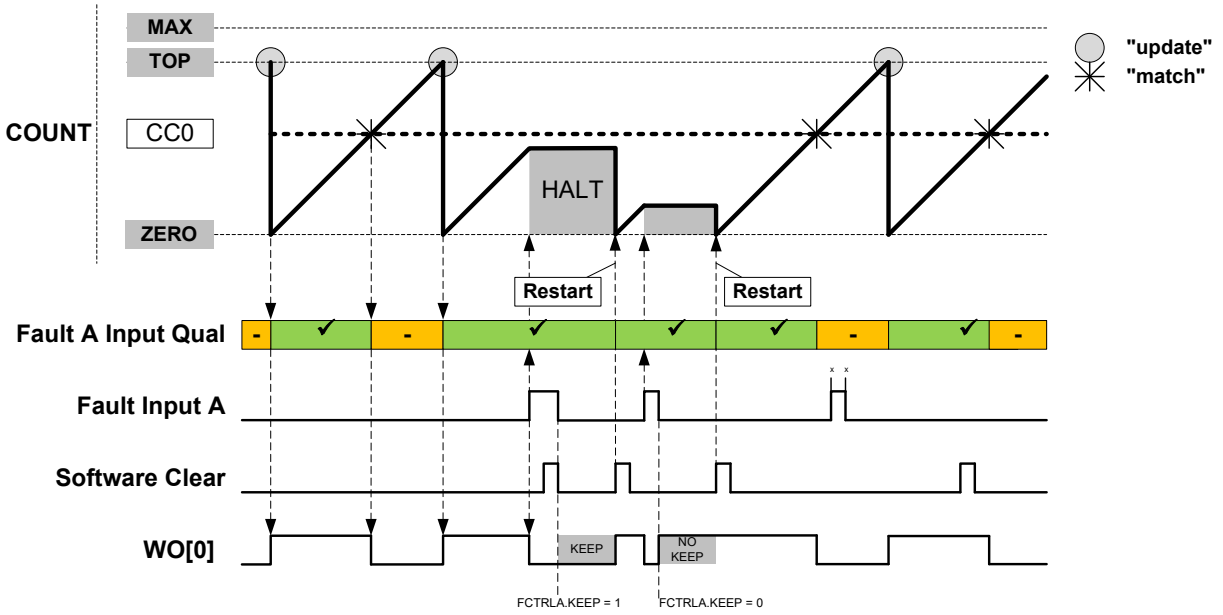
© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-31.** Waveform Generation with Fault Qualification, Halt, and Restart Actions




**Software Halt Action** This is configured by writing 0x2 to the Fault n Halt mode bits in the Recoverable Fault n configuration register (FCTRLn.HALT). Software halt action is similar to hardware halt action, but in order to restart the timer/counter, the corresponding fault condition must not be present anymore, and the corresponding FAULT n bit in the STATUS register must be cleared by software.

**Figure 36-32.** Waveform Generation with Software Halt, Fault Qualification, Keep and Restart Actions




### 36.6.3.6. Non-Recoverable Faults

The non-recoverable fault action will force all the compare outputs to a pre-defined level programmed into the Driver Control register (DRVCTRL.NRE and DRVCTRL.NRV). The non-recoverable fault input (EV0 and EV1) actions are enabled in Event Control register (EVCTRL.EVACT0 and EVCTRL.EVACT1).

To avoid false fault detection on external events (e.g. a glitch on an I/O port) a digital filter can be enabled using Non-Recoverable Fault Input x Filter Value bits in the Driver Control register (DRVCTRL.FILTERVALn). Therefore, the event detection is synchronous, and event action is delayed by the selected digital filter value clock cycles.


© 2026 Microchip Technology Inc. and its subsidiaries

When the Fault Detection on Debug Break Detection bit in Debug Control register (DGBCTRL.FDDBD) is written to '1', a non-recoverable Debug Faults State and an interrupt (DFS) is generated when the system goes in debug operation.

In RAMP2, RAMP2A, or DSBOTH operation, when the Lock Update bit in the Control B register is set by writing CTRLBSET.LUPD=1 and the ramp index or counter direction changes, a non-recoverable Update Fault State and the respective interrupt (UFS) are generated.

### 36.6.3.7. Waveform Extension

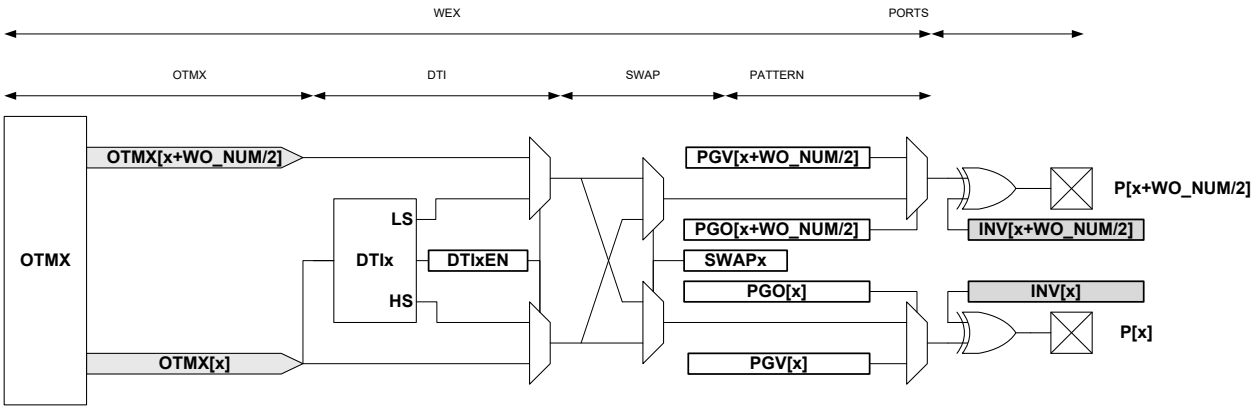
Figure 36-33 shows a schematic diagram of actions of the four optional units that follow the recoverable fault stage on a port pin pair: Output Matrix (OTMX), Dead-Time Insertion (DTI), SWAP and Pattern Generation. The DTI and SWAP units can be seen as a four port pair slices:

- Slice 0 DTI0 / SWAP0 acting on port pins (WO[0], WO[WO_NUM/2 +0])
- Slice 1 DTI1 / SWAP1 acting on port pins (WO[1], WO[WO_NUM/2 +1])

And more generally:

- Slice n DTIx / SWAPx acting on port pins (WO[x], WO[WO_NUM/2 +x])

**Figure 36-33.** Waveform Extension Stage Details




**The output matrix (OTMX)** unit distributes compare channels, according to the selectable configurations in Table 36-4.

**Table 36-4.** Output Matrix Channel Pin Routing Configuration

| Value | OTMX[x] | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0x0 | CC3 | CC2 | CC1 | CC0 | CC3 | CC2 | CC1 | CC0 |
| 0x1 | CC1 | CC0 | CC1 | CC0 | CC1 | CC0 | CC1 | CC0 |
| 0x2 | CC0 | CC0 | CC0 | CC0 | CC0 | CC0 | CC0 | CC0 |
| 0x3 | CC1 | CC1 | CC1 | CC1 | CC1 | CC1 | CC1 | CC0 |

Notes on Table 36-4:

- Configuration 0x0 is the default configuration. The channel location is the default one, and channels are distributed on outputs modulo the number of channels. Channel 0 is routed to the Output matrix output OTMX[0], and Channel 1 to OTMX[1]. If there are more outputs than channels, then channel 0 is duplicated to the Output matrix output OTMX[CC_NUM], channel 1 to OTMX[CC_NUM+1] and so on.
- Configuration 0x1 distributes the channels on output modulo half the number of channels. This assigns twice the number of output locations to the lower channels than the default configuration. This can be used, for example, to control the four transistors of a full bridge using only two compare channels.


© 2026 Microchip Technology Inc. and its subsidiaries

Using pattern generation, some of these four outputs can be overwritten by a constant level, enabling flexible drive of a full bridge in all quadrant configurations.

- Configuration 0x2 distributes compare channel 0 (CC0) to all port pins. With pattern generation, this configuration can control a stepper motor.
- Configuration 0x3 distributes the compare channel CC0 to the first output, and the channel CC1 to all other outputs. Together with pattern generation and the fault extension, this configuration can control up to seven LED strings, with a boost stage.

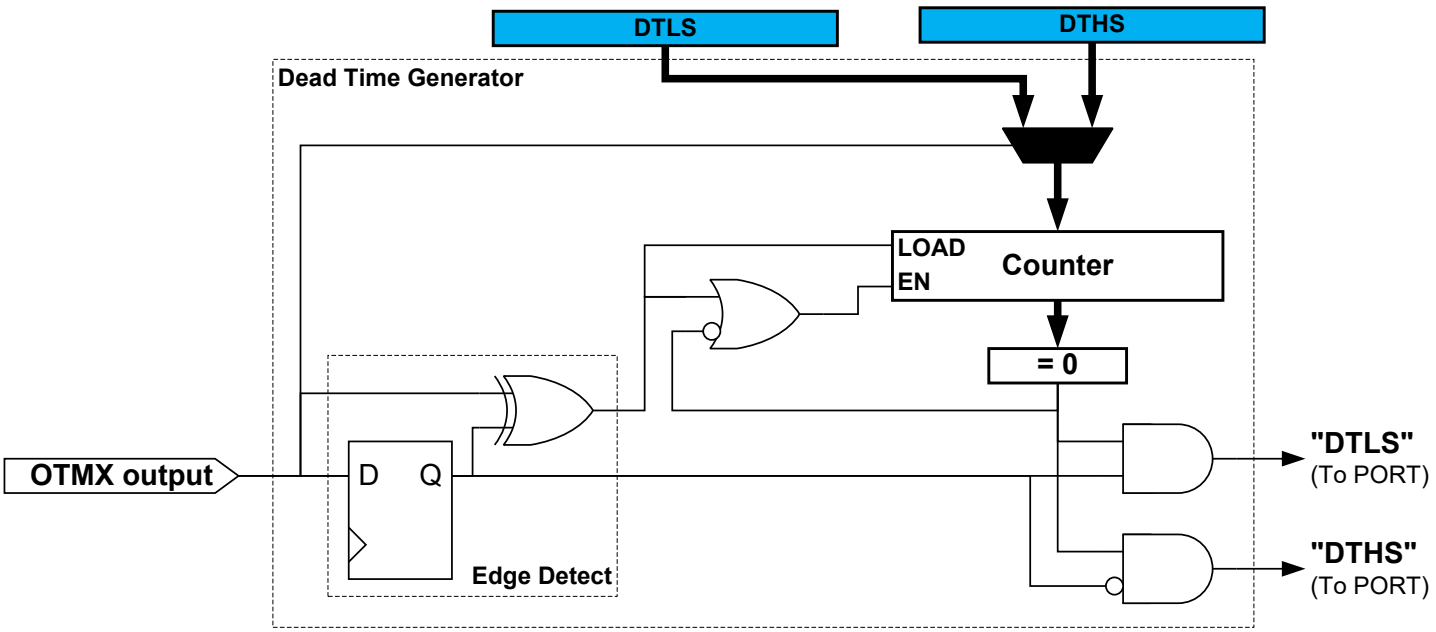
**Table 36-5.** Example: four compare channels on four outputs

| Value | OTMX[3] | OTMX[2] | OTMX[1] | OTMX[0] |
|---|---|---|---|---|
| 0x0 | CC3 | CC2 | CC1 | CC0 |
| 0x1 | CC1 | CC0 | CC1 | CC0 |
| 0x2 | CC0 | CC0 | CC0 | CC0 |
| 0x3 | CC1 | CC1 | CC1 | CC0 |

**The dead-time insertion (DTI)** unit generates OFF time with the non-inverted low side (LS) and inverted high side (HS) of the wave generator output forced at low level. This OFF time is called dead time. Dead-time insertion ensures that the LS and HS will never switch simultaneously.

The DTI stage consists of four equal dead-time insertion generators; one for each of the first four compare channels. Figure 36-34 shows the block diagram of one DTI generator. The four channels have a common register which controls the dead time, which is independent of high side and low side setting.

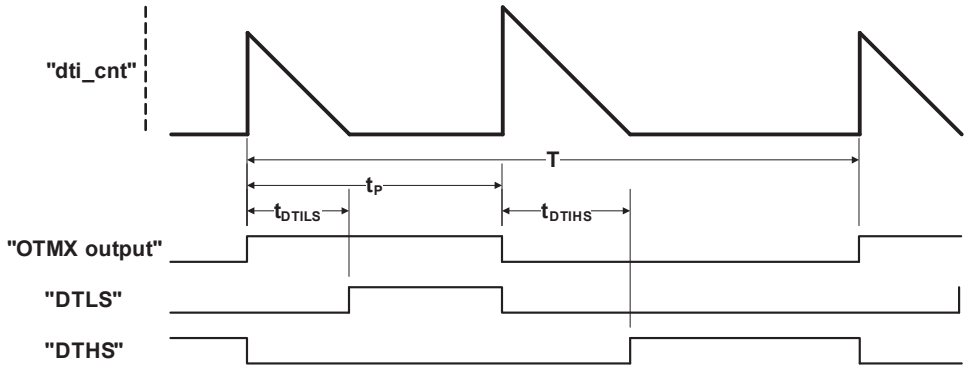
**Figure 36-34.** Dead-Time Generator Block Diagram




As shown in Figure 36-35, the 8-bit dead-time counter is decremented by one for each peripheral clock cycle until it reaches zero. A non-zero counter value will force both the low side and high side outputs into their OFF state. When the output matrix (OTMX) output changes, the dead-time counter is reloaded according to the edge of the input. When the output changes from low to high (positive edge) it initiates a counter reload of the DTLS register. When the output changes from high to low (negative edge) it reloads the DTHS register.

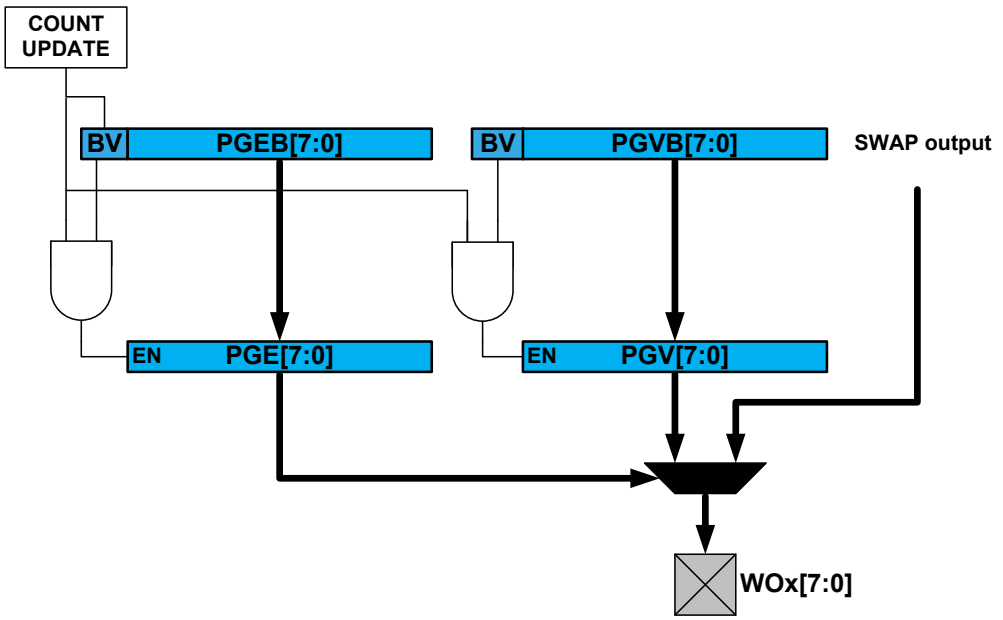

© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-35.** Dead-Time Generator Timing Diagram




**The pattern generator unit** produces a synchronized bit pattern across the port pins it is connected to. The pattern generation features are primarily intended for handling the commutation sequence in brushless DC motors (BLDC), stepper motors, and full bridge control. See also Figure 36-36.

**Figure 36-36.** Pattern Generator Block Diagram




As with other double-buffered timer/counter registers, the register update is synchronized to the UPDATE condition set by the timer/counter waveform generation operation. If synchronization is not required by the application, the software can simply access directly the PATT.PGE, PATT.PGV bits registers.

### 36.6.4. Host/Client Operation

Two TCC instances sharing the same GCLK_TCC clock, can be linked to provide more synchronized CC channels. The operation is enabled by setting the Host Synchronization bit in Control A register (CTRLA.MSYNC) in the Client instance. When the bit is set, the Client TCC instance will synchronize the CC channels to the Host counter.

**Related Links**

CTRLA

Control A


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.6.5. DMA, Interrupts, and Events

**Table 36-6.** Module Requests for TCC

| Condition | Interrupt request | Event output | Event input | DMA request | DMA request is cleared |
|---|---|---|---|---|---|
| Overflow / Underflow | Yes | Yes | | Yes[1] | On DMA acknowledge |
| Channel Compare Match or Capture | Yes | Yes | Yes[2] | Yes[3] | For circular buffering: on DMA acknowledge<br>For capture channel: when CCx register is read |
| Retrigger | Yes | Yes | | | |
| Count | Yes | Yes | | | |
| Capture Overflow Error | Yes | | | | |
| Debug Fault State | Yes | | | | |
| Recoverable Faults | Yes | | | | |
| Non-Recoverable Faults | Yes | | | | |
| TCCx Event 0 input | | | Yes[4] | | |
| TCCx Event 1 input | | | Yes[5] | | |

Notes:
1. DMA request set on overflow, underflow or re-trigger conditions.
2. Can perform capture or generate recoverable fault on an event input.
3. In capture or circular modes.
4. On event input, either action can be executed:
   - re-trigger counter
   - control counter direction
   - stop the counter
   - decrement the counter
   - perform period and pulse width capture
   - generate non-recoverable fault
5. On event input, either action can be executed:
   - re-trigger counter
   - increment or decrement counter depending on direction
   - start the counter
   - increment or decrement counter based on direction
   - increment counter regardless of direction
   - generate non-recoverable fault

#### 36.6.5.1. DMA Operation

The TCC can generate the following DMA requests:

**Counter overflow (OVF)**

If the Ones-shot Trigger mode in the control A register (CTRLA.DMAOS) is written to '0', the TCC generates a DMA request on each cycle when an update condition (overflow, underflow or re-trigger) is detected.
When an update condition (overflow, underflow or re-trigger) is detected while CTRLA.DMAOS=1, the TCC generates a DMA trigger on the cycle following the DMA One-Shot Command written to the Control B register (CTRLBSET.CMD=DMAOS).

In both cases, the request is cleared by hardware on DMA acknowledge.


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| **Channel Match (MCx)** | A DMA request is set only on a compare match if CTRLA.DMAOS=0. The request is cleared by hardware on DMA acknowledge.<br>When CTRLA.DMAOS=1, the DMA requests are not generated. |
| **Channel Capture (MCx)** | For a capture channel, the request is set when valid data is present in the CCx register, and cleared once the CCx register is read.<br>In this operation mode, the CTRLA.DMAOS bit value is ignored. |

### DMA Operation with Circular Buffer

When circular buffer operation is enabled, the buffer registers must be written in a correct order and synchronized to the update times of the timer. The DMA triggers of the TCC provide a way to ensure a safe and correct update of circular buffers.

**Note:** Circular buffer are intended to be used with RAMP2, RAMP2A and DSBOTH operation only.

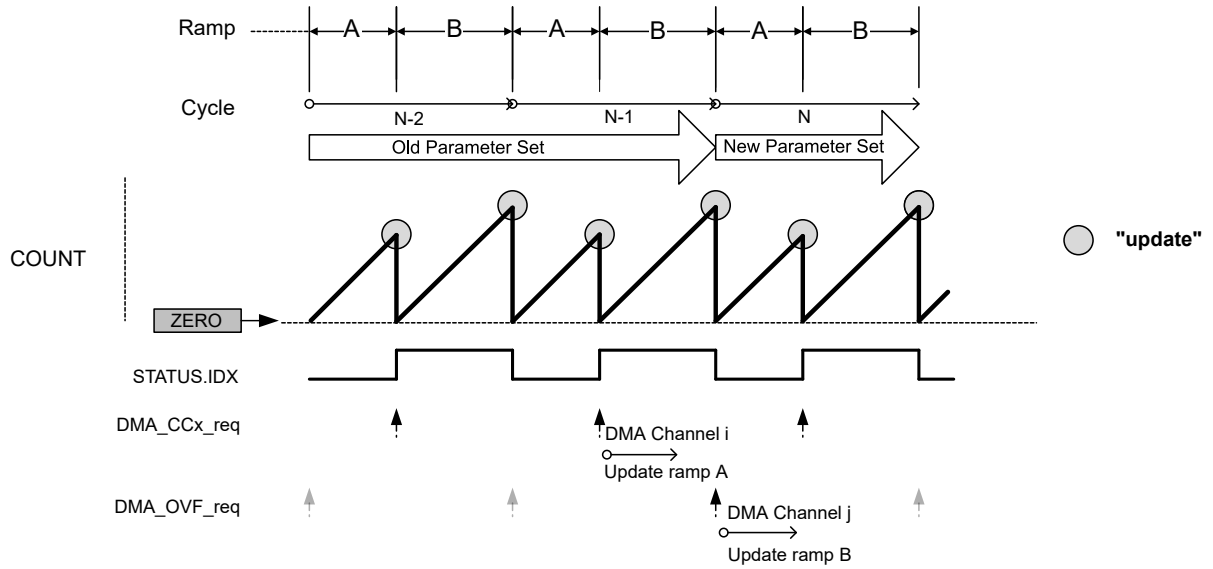
*DMA Operation with Circular Buffer in RAMP2 and RAMP2A Mode*

When a CCx channel is selected as a circular buffer, the related DMA request is not set on a compare match detection, but on start of ramp B.

If at least one circular buffer is enabled, the DMA overflow request is conditioned to the start of ramp A with an effective DMA transfer on previous ramp B (DMA acknowledge).

The update of all circular buffer values for ramp A can be done through a DMA channel triggered on a MC trigger. The update of all circular buffer values for ramp B, can be done through a second DMA channel triggered by the overflow DMA request.

**Figure 36-37.** DMA Triggers in RAMP and RAMP2 Operation Mode and Circular Buffer Enabled




DMA Operation with Circular Buffer in DSBOTH Mode

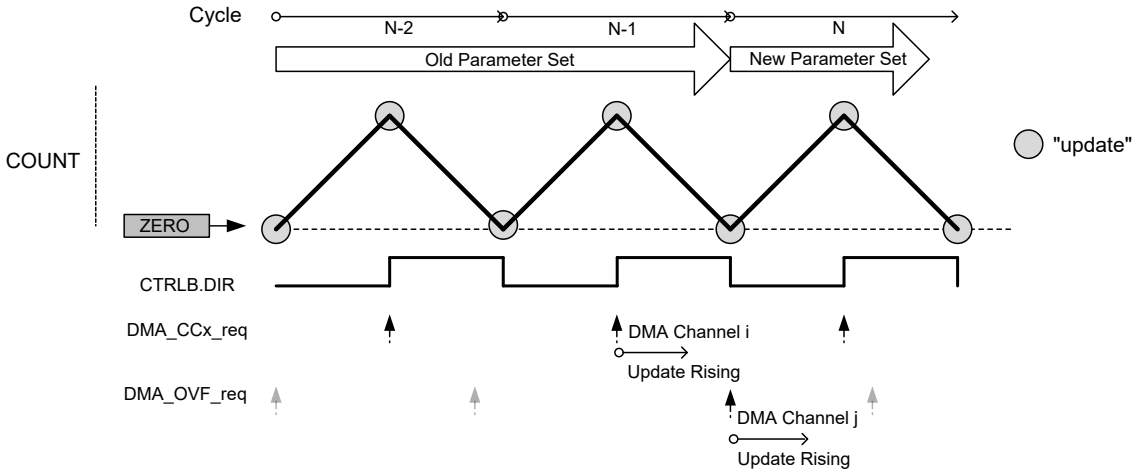
When a CC channel is selected as a circular buffer, the related DMA request is not set on a compare match detection, but on start of down-counting phase.

If at least one circular buffer is enabled, the DMA overflow request is conditioned to the start of up-counting phase with an effective DMA transfer on previous down-counting phase (DMA acknowledge).

When up-counting, all circular buffer values can be updated through a DMA channel triggered by MC trigger. When down-counting, all circular buffer values can be updated through a second DMA channel, triggered by the OVF DMA request.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 36-38.** DMA Triggers in DSBOTH Operation Mode and Circular Buffer Enabled




### 36.6.5.2. Interrupts

The TCC has the following interrupt sources:

- Overflow/Underflow (OVF)
- Retrigger (TRG)
- Count (CNT) - refer also to description of EVCTRL.CNTSEL.
- Capture Overflow Error (ERR)
- Non-Recoverable Update Fault (UFS)
- Debug Fault State (DFS)
- Recoverable Faults (FAULTn)
- Non-recoverable Faults (FAULTx)
- Compare Match or Capture Channels (MCx)

These interrupts are asynchronous wake-up sources. See Sleep Mode Entry and Exit Table in PM/ Sleep Mode Controller section for details.

Each interrupt source has an Interrupt flag associated with it. The Interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the Interrupt condition occurs. Each interrupt can be individually enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the Interrupt flag is cleared, the interrupt is disabled, or the TCC is reset. See INTFLAG for details on how to clear Interrupt flags. The TCC has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which Interrupt condition is present.

Note: Interrupts must be globally enabled for interrupt requests to be generated. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

Nested Vector Interrupt Controller
Sleep Mode Controller


© 2026 Microchip Technology Inc. and its subsidiaries

#### 36.6.5.3. Events

The TCC can generate the following output events:

- Overflow/Underflow (OVF)
- Trigger (TRG)
- Counter (CNT) For further details, refer to EVCTRL.CNTSEL description.
- Compare Match or Capture on compare/capture channels: MCx

Writing a '1' ('0') to an Event Output bit in the Event Control Register (EVCTRL.xxEO) enables (disables) the corresponding output event. Refer also to *EVSYS – Event System*.

The TCC can take the following actions on a channel input event (MCx):

- Capture event
- Generate a recoverable or non-recoverable fault

The TCC can take the following actions on counter Event 1 (TCCx EV1):

- Counter re-trigger
- Counter direction control
- Stop the counter
- Decrement the counter on event
- Period and pulse width capture
- Non-recoverable fault

The TCC can take the following actions on counter Event 0 (TCCx EV0):

- Counter re-trigger
- Count on event (increment or decrement, depending on counter direction)
- Counter start - start counting on the event rising edge. Further events will not restart the counter; the counter will keep on counting using prescaled GCLK_TCCx, until it reaches TOP or ZERO, depending on the direction.
- Counter increment on event. This will increment the counter, irrespective of the counter direction.
- Count during active state of an asynchronous event (increment or decrement, depending on counter direction). In this case, the counter will be incremented or decremented on each cycle of the prescaled clock, as long as the event is active.
- Non-recoverable fault

The counter Event Actions are available in the Event Control registers (EVCTRL.EVACT0 and EVCTRL.EVACT1). For further details, refer to EVCTRL.

Writing a '1' ('0') to an Event Input bit in the Event Control register (EVCTRL.MCEIx or EVCTRL.TCEIx) enables (disables) the corresponding action on input event.

**Note:** When several events are connected to the TCC, the enabled action will apply for each of the incoming events. Refer to *EVSYS – Event System* for details on how to configure the event system.

**Related Links**

Event System (EVSYS)

### 36.6.6. Sleep Mode Operation

The TCC can be configured to operate in any sleep mode. To be able to run in standby the RUNSTDBY bit in the Control A register (CTRLA.RUNSTDBY) must be '1'. The MODULE can in any sleep mode wake up the device using interrupts or perform actions through the Event System.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.6.7. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- The Software Reset and Enable bits in the Control A register (CTRLA.SWRST and CTRLA.ENABLE)

The following registers are synchronized when written:

- The Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- The Status register (STATUS)
- The Pattern and Pattern Buffer registers (PATT and PATTBUF)
- The Waveform register (WAVE)
- The Count Value register (COUNT)
- The Period Value and Period Buffer Value registers (PER and PERBUF)
- The Compare/Capture Channel x and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- The Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- The Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Required read-synchronization is denoted by the "Read-Synchronized" property in the register description.

**Related Links**

Register Synchronization
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 36.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | RESOLUTION[1:0] | | | | | ENABLE | SWRST |
| | | 15:8 | MSYNC | ALOCK | PRESCYNC[1:0] | | RUNSTDBY | PRESCALER[2:0] | | |
| | | 23:16 | DMAOS | | | | | | | |
| | | 31:24 | | | | | CPTEN3 | CPTEN2 | CPTEN1 | CPTEN0 |
| 0x04 | CTRLBCLR | 7:0 | CMD[2:0] | | | IDXCMD[1:0] | | ONESHOT | LUPD | DIR |
| 0x05 | CTRLBSET | 7:0 | CMD[2:0] | | | IDXCMD[1:0] | | ONESHOT | LUPD | DIR |
| 0x06 ... 0x07 | Reserved | | | | | | | | | |
| 0x08 | SYNCBUSY | 7:0 | PER | WAVE | PATT | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| | | 15:8 | | | | | CC3 | CC2 | CC1 | CC0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | FCTRLA | 7:0 | RESTART | BLANK[1:0] | | QUAL | KEEP | | SRC[1:0] | |
| | | 15:8 | | CAPTURE[2:0] | | | CHSEL[1:0] | | HALT[1:0] | |
| | | 23:16 | BLANKVAL[7:0] | | | | | | | |
| | | 31:24 | | | | | FILTERVAL[3:0] | | | |
| 0x10 | FCTRLB | 7:0 | RESTART | BLANK[1:0] | | QUAL | KEEP | | SRC[1:0] | |
| | | 15:8 | | CAPTURE[2:0] | | | CHSEL[1:0] | | HALT[1:0] | |
| | | 23:16 | BLANKVAL[7:0] | | | | | | | |
| | | 31:24 | | | | | FILTERVAL[3:0] | | | |
| 0x14 | WEXCTRL | 7:0 | | | | | | | OTMX[1:0] | |
| | | 15:8 | | | | | DTIEN3 | DTIEN2 | DTIEN1 | DTIEN0 |
| | | 23:16 | DTLS[7:0] | | | | | | | |
| | | 31:24 | DTHS[7:0] | | | | | | | |
| 0x18 | DRVCTRL | 7:0 | NRE7 | NRE6 | NRE5 | NRE4 | NRE3 | NRE2 | NRE1 | NRE0 |
| | | 15:8 | NRV7 | NRV6 | NRV5 | NRV4 | NRV3 | NRV2 | NRV1 | NRV0 |
| | | 23:16 | INVEN7 | INVEN6 | INVEN5 | INVEN4 | INVEN3 | INVEN2 | INVEN1 | INVEN0 |
| | | 31:24 | FILTERVAL1[3:0] | | | | FILTERVAL0[3:0] | | | |
| 0x1C ... 0x1D | Reserved | | | | | | | | | |
| 0x1E | DBGCTRL | 7:0 | | | | | | FDDBD | | DBGRUN |
| 0x1F | Reserved | | | | | | | | | |
| 0x20 | EVCTRL | 7:0 | CNTSEL[1:0] | | EVACT1[2:0] | | | EVACT0[2:0] | | |
| | | 15:8 | TCEI1 | TCEI0 | TCINV1 | TCINV0 | | CNTEO | TRGEO | OVFEO |
| | | 23:16 | | | | | MCEI3 | MCEI2 | MCEI1 | MCEI0 |
| | | 31:24 | | | | | MCEO3 | MCEO2 | MCEO1 | MCEO0 |
| 0x24 | INTENCLR | 7:0 | | | | | ERR | CNT | TRG | OVF |
| | | 15:8 | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| | | 23:16 | | | | | MC3 | MC2 | MC1 | MC0 |
| | | 31:24 | | | | | | | | |
| 0x28 | INTENSET | 7:0 | | | | | ERR | CNT | TRG | OVF |
| | | 15:8 | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| | | 23:16 | | | | | MC3 | MC2 | MC1 | MC0 |
| | | 31:24 | | | | | | | | |
| 0x2C | INTFLAG | 7:0 | | | | | ERR | CNT | TRG | OVF |
| | | 15:8 | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| | | 23:16 | | | | | MC3 | MC2 | MC1 | MC0 |
| | | 31:24 | | | | | | | | |
| 0x30 | STATUS | 7:0 | PERBUFV | WAVEBUFV | PATTBUFV | SLAVE | DFS | UFS | IDX | STOP |
| | | 15:8 | FAULT1 | FAULT0 | FAULTB | FAULTA | FAULT1IN | FAULT0IN | FAULTBIN | FAULTAIN |
| | | 23:16 | | | | | CCBUFV3 | CCBUFV2 | CCBUFV1 | CCBUFV0 |
| | | 31:24 | | | | | CMP3 | CMP2 | CMP1 | CMP0 |
| 0x34 | COUNT | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| | | 23:16 | COUNT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x38 | PATT | 7:0 | PGE7 | PGE6 | PGE5 | PGE4 | PGE3 | PGE2 | PGE1 | PGE0 |
| | | 15:8 | PGV7 | PGV6 | PGV5 | PGV4 | PGV3 | PGV2 | PGV1 | PGV0 |
| 0x3A ... 0x3B | Reserved | | | | | | | | | |
| 0x3C | WAVE | 7:0 | CIPEREN | | RAMP[1:0] | | | WAVEGEN[2:0] | | |
| | | 15:8 | | | | | CICCEN3 | CICCEN2 | CICCEN1 | CICCEN0 |
| | | 23:16 | | | | | POL3 | POL2 | POL1 | POL0 |
| | | 31:24 | | | | | SWAP3 | SWAP2 | SWAP1 | SWAP0 |
| 0x40 | PER | 7:0 | PER[1:0] | | DITHER[5:0] | | | | | |
| | | 15:8 | PER[9:2] | | | | | | | |
| | | 23:16 | PER[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x44 | CC0 | 7:0 | CC[1:0] | | DITHER[5:0] | | | | | |
| | | 15:8 | CC[9:2] | | | | | | | |
| | | 23:16 | CC[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x48 | CC1 | 7:0 | CC[1:0] | | DITHER[5:0] | | | | | |
| | | 15:8 | CC[9:2] | | | | | | | |
| | | 23:16 | CC[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x4C | CC2 | 7:0 | CC[1:0] | | DITHER[5:0] | | | | | |
| | | 15:8 | CC[9:2] | | | | | | | |
| | | 23:16 | CC[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x50 | CC3 | 7:0 | CC[1:0] | | DITHER[5:0] | | | | | |
| | | 15:8 | CC[9:2] | | | | | | | |
| | | 23:16 | CC[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x54 ... 0x63 | Reserved | | | | | | | | | |
| 0x64 | PATTBUF | 7:0 | PGEB7 | PGEB6 | PGEB5 | PGEB4 | PGEB3 | PGEB2 | PGEB1 | PGEB0 |
| | | 15:8 | PGVB7 | PGVB6 | PGVB5 | PGVB4 | PGVB3 | PGVB2 | PGVB1 | PGVB0 |
| 0x66 ... 0x67 | Reserved | | | | | | | | | |
| 0x68 | WAVEBUF | 7:0 | CIPERENB | | RAMPB[1:0] | | | WAVEGENB[2:0] | | |
| | | 15:8 | | | | | CICCENB3 | CICCENB2 | CICCENB1 | CICCENB0 |
| | | 23:16 | | | | | POLB3 | POLB2 | POLB1 | POLB0 |
| | | 31:24 | | | | | SWAPB 3 | SWAPB 2 | SWAPB 1 | SWAPB 0 |
| 0x6C | PERBUF | 7:0 | PERBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| | | 15:8 | PERBUF[9:2] | | | | | | | |
| | | 23:16 | PERBUF[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x70 | CCBUFx0 | 7:0 | CCBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| | | 15:8 | CCBUF[9:2] | | | | | | | |
| | | 23:16 | CCBUF[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x74 | CCBUFx1 | 7:0 | CCBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| | | 15:8 | CCBUF[9:2] | | | | | | | |
| | | 23:16 | CCBUF[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x78 | CCBUFx2 | 7:0 | CCBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| | | 15:8 | CCBUF[9:2] | | | | | | | |
| | | 23:16 | CCBUF[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x7C | CCBUFx3 | 7:0 | CCBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| | | 15:8 | CCBUF[9:2] | | | | | | | |
| | | 23:16 | CCBUF[17:10] | | | | | | | |
| | | 31:24 | | | | | | | | |

## 36.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected Bits, Write-Synchronized Bits

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CPTEN3 | CPTEN2 | CPTEN1 | CPTEN0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DMAOS | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | MSYNC | ALOCK | PRESCYNC[1:0] | | RUNSTDBY | PRESCALER[2:0] | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RESOLUTION[1:0] | | | | | ENABLE | SWRST |
| Access | | R/W | R/W | | | | R/W | R/W |
| Reset | | 0 | 0 | | | | 0 | 0 |

**Bits 24, 25, 26, 27 – CPTENx** Capture Channel x Enable
These bits are used to select the capture or compare operation on channel x.
Writing a '1' to CPTENx enables capture on channel x.
Writing a '0' to CPTENx disables capture on channel x.
**Note:** These bits are enable-protected. These bits are not synchronized.

**Bit 23 – DMAOS** DMA One-Shot Trigger Mode
This bit enables the DMA One-shot Trigger Mode.
Writing a '1' to this bit will generate a DMA trigger on TCC cycle following a TCC_CTRLBSET_CMD_DMAOS command.
Writing a '0' to this bit will generate DMA triggers on each TCC cycle.
**Note:** This bit is enable-protected. This bit is not synchronized.

**Bit 15 – MSYNC** Host Synchronization (only for TCC Client instance)
This bit must be set if the TCC counting operation must be synchronized on its Host TCC.
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TCC controls its own counter. |
| 1 | The counter is controlled by its Host TCC. |

**Bit 14 – ALOCK** Auto Lock
**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The Lock Update bit in the Control B register (CTRLB.LUPD) is not affected by overflow/underflow, and re-trigger events |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | CTRLB.LUPD is set to '1' on each overflow/underflow or re-trigger event. |

**Bits 13:12 – PRESCYNC[1:0]** Prescaler and Counter Synchronization

These bits select if on re-trigger event, the Counter is cleared or reloaded on either the next GCLK_TCCx clock, or on the next prescaled GCLK_TCCx clock. It is also possible to reset the prescaler on re-trigger event.

**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description | |
|---|---|---|---|
| | | Counter Reloaded | Prescaler |
| 0x0 | GCLK | Reload or reset Counter on next GCLK | - |
| 0x1 | PRESC | Reload or reset Counter on next prescaler clock | - |
| 0x2 | RESYNC | Reload or reset Counter on next GCLK | Reset prescaler counter |
| 0x3 | Reserved | | |

**Bit 11 – RUNSTDBY** Run in Standby

This bit is used to keep the TCC running in Standby mode.

**Note:** This bit is enable-protected. This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TCC is halted in standby. |
| 1 | The TCC continues to run in standby. |

**Bits 10:8 – PRESCALER[2:0]** Prescaler

These bits select the Counter prescaler factor.

**Note:** This bit field is enable-protected. This bit field is not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV1 | Prescaler: GCLK_TCCx |
| 0x1 | DIV2 | Prescaler: GCLK_TCCx/2 |
| 0x2 | DIV4 | Prescaler: GCLK_TCCx/4 |
| 0x3 | DIV8 | Prescaler: GCLK_TCCx/8 |
| 0x4 | DIV16 | Prescaler: GCLK_TCCx/16 |
| 0x5 | DIV64 | Prescaler: GCLK_TCCx/64 |
| 0x6 | DIV256 | Prescaler: GCLK_TCCx/256 |
| 0x7 | DIV1024 | Prescaler: GCLK_TCCx/1024 |

**Bits 6:5 – RESOLUTION[1:0]** Dithering Resolution

These bits increase the TCC resolution by enabling the dithering options.

**Note:** This bit field is enable-protected. This bit field is not synchronized.

**Table 36-7.** Dithering

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | The dithering is disabled. |
| 0x1 | DITH4 | Dithering is done every 16 PWM frames. PER[3:0] and CCx[3:0] contain dithering pattern selection. |
| 0x2 | DITH5 | Dithering is done every 32 PWM frames. PER[4:0] and CCx[4:0] contain dithering pattern selection. |
| 0x3 | DITH6 | Dithering is done every 64 PWM frames. PER[5:0] and CCx[5:0] contain dithering pattern selection. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – ENABLE** Enable

**Notes:**

1. This bit is not enable-protected.
2. This bit is write synchronized: SYNCBUSY.ENABLE must be checked to ensure the CTRLA.ENABLE synchronization is complete.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the TCC (except DBGCTRL) to their initial state, and the TCC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence; all other writes in the same write-operation will be discarded.

**Notes:**

1. This bit is not enable-protected.
2. This bit is write synchronized: SYNCBUSY.ENABLE must be checked to ensure the CTRLA.ENABLE synchronization is complete.

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing. |
| 1 | The Reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.2. Control B Clear

**Name:** CTRLBCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

This register allows the user to change this register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set (CTRLBSET) register.
**Note:** This register is write-synchronized: SYNCBUSY.CTRLB must be checked to ensure the CTRLBCLR register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | IDXCMD[1:0] | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** TCC Command
These bits can be used for software control of re-triggering and stop commands of the TCC. When a command has been executed, the CMD bit field will read back zero. The commands are executed on the next prescaled GCLK_TCC clock cycle.
Writing zero to this bit group has no effect.
Writing a '1' to any of these bits will clear the pending command.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RETRIGGER | Clear start, restart or retrigger |
| 0x2 | STOP | Force stop |
| 0x3 | UPDATE | Force update of double buffered registers |
| 0x4 | READSYNC | Force COUNT read synchronization |
| 0x5 | DMAOS | One-shot DMA trigger |

**Bits 4:3 – IDXCMD[1:0]** Ramp Index Command
These bits can be used to force cycle A and cycle B changes in RAMP2 and RAMP2A operation. On timer/counter update condition, the command is executed, the IDX flag in STATUS register is updated and the IDXCMD command is cleared.
Writing zero to these bits has no effect.
Writing a '1' to any of these bits will clear the pending command.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | DISABLE Command disabled: IDX toggles between cycles A and B |
| 0x1 | SET | Set IDX: cycle B will be forced in the next cycle |
| 0x2 | CLEAR | Clear IDX: cycle A will be forced in next cycle |
| 0x3 | HOLD | Hold IDX: the next cycle will be the same as the current cycle. |

**Bit 2 – ONESHOT** One-Shot
This bit controls one-shot operation of the TCC. When one-shot operation is enabled, the TCC will stop counting on the next overflow/underflow condition or on a stop command.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will disable the one-shot operation.

| Value | Description |
|---|---|
| 0 | The TCC will update the counter value on overflow/underflow condition and continue operation. |
| 1 | The TCC will stop counting on the next underflow/overflow condition. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – LUPD** Lock Update

This bit controls the update operation of the TCC buffered registers.
When CTRLB.LUPD is cleared, the hardware UPDATE registers with value from their buffered registers is enabled.
This bit has no effect when input capture operation is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will enable the registers updates on hardware UPDATE condition.

| Value | Description |
|---|---|
| 0 | The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values *are* copied into the corresponding CCx, PER, PGV, PGO and SWAPx registers on hardware update condition. |
| 1 | The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values are *not* copied into the corresponding CCx, PER, PGV, PGO and SWAPx registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction

This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.3. Control B Set

**Name:** CTRLBSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized, Read-Synchronized

This register allows the user to change this register without doing a read-modify-write operation. Changes in this register will also be reflected in the Control B Set (CTRLBCLR) register.
**Note:** This register is write-synchronized: SYNCBUSY.CTRLB must be checked to ensure the CTRLBSET register synchronization is complete.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CMD[2:0] | | | IDXCMD[1:0] | | ONESHOT | LUPD | DIR |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:5 – CMD[2:0]** TCC Command
These bits can be used for software control of re-triggering and stop commands of the TCC. When a command has been executed, the CMD bit field will be read back as zero. The commands are executed on the next prescaled GCLK_TCC clock cycle.
Writing zero to this bit group has no effect
Writing a valid value to this bit group will set the associated command.

| Value | Name | Description |
|---|---|---|
| 0x0 | NONE | No action |
| 0x1 | RETRIGGER | Force start, restart or retrigger |
| 0x2 | STOP | Force stop |
| 0x3 | UPDATE | Force update of double buffered registers |
| 0x4 | READSYNC | Force a read synchronization of COUNT |
| 0x5 | DMAOS | One-shot DMA trigger |

**Bits 4:3 – IDXCMD[1:0]** Ramp Index Command
These bits can be used to force cycle A and cycle B changes in the RAMP2 and RAMP2A operation. On timer/counter update condition, the command is executed, the IDX flag in the STATUS register is updated and the IDXCMD command is cleared.
Writing a zero to these bits has no effect.
Writing a valid value to these bits will set a command.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Command disabled: IDX toggles between cycles A and B |
| 0x1 | SET | Set IDX: cycle B will be forced in the next cycle |
| 0x2 | CLEAR | Clear IDX: cycle A will be forced in next cycle |
| 0x3 | HOLD | Hold IDX: the next cycle will be the same as the current cycle. |

**Bit 2 – ONESHOT** One-Shot
This bit controls one-shot operation of the TCC. When in one-shot operation, the TCC will stop counting on the next overflow/underflow condition or a stop command.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will enable the one-shot operation.

| Value | Description |
|---|---|
| 0 | The TCC will count continuously. |
| 1 | The TCC will stop counting on the next underflow/overflow condition. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – LUPD** Lock Update

This bit controls the update operation of the TCC buffered registers.
When CTRLB.LUPD is set, the hardware UPDATE registers with value from their buffered registers is disabled. Disabling the update ensures that all buffer registers are valid before an hardware update is performed. After all the buffer registers are loaded correctly, the buffered registers can be unlocked.
This bit has no effect when input capture operation is enabled.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will disable the registers updates on hardware UPDATE condition.

| Value | Description |
|---|---|
| 0 | The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values *are* copied into the corresponding CCx, PER, PGV, PGO and SWAPx registers on hardware update condition. |
| 1 | The CCBx, PERB, PGVB, PGOB, and SWAPBx buffer registers values are *not* copied into CCx, PER, PGV, PGO and SWAPx registers on hardware update condition. |

**Bit 0 – DIR** Counter Direction

This bit is used to change the direction of the counter.
Writing a '0' to this bit has no effect
Writing a '1' to this bit will clear the bit and make the counter count up.

| Value | Description |
|---|---|
| 0 | The timer/counter is counting up (incrementing). |
| 1 | The timer/counter is counting down (decrementing). |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.4. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CC3 | CC2 | CC1 | CC0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER | WAVE | PATT | COUNT | STATUS | CTRLB | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 8, 9, 10, 11 – CC** Compare/Capture Channel x Synchronization Busy
This bit is cleared when the synchronization of Compare/Capture Channel x register between the clock domains is complete.
This bit is set when the synchronization of Compare/Capture Channel x register between clock domains is started.
CCx bit is available only for existing Compare/Capture Channels. For details on CC channels number, refer to each TCC feature list.
This bit is set when the synchronization of CCx register between clock domains is started.

**Bit 7 – PER** PER Synchronization Busy
This bit is cleared when the synchronization of PER register between the clock domains is complete.
This bit is set when the synchronization of PER register between clock domains is started.

**Bit 6 – WAVE** WAVE Synchronization Busy
This bit is cleared when the synchronization of WAVE register between the clock domains is complete.
This bit is set when the synchronization of WAVE register between clock domains is started.

**Bit 5 – PATT** PATT Synchronization Busy
This bit is cleared when the synchronization of PATTERN register between the clock domains is complete.
This bit is set when the synchronization of PATTERN register between clock domains is started.

**Bit 4 – COUNT** COUNT Synchronization Busy
This bit is cleared when the synchronization of COUNT register between the clock domains is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

This bit is set when the synchronization of COUNT register between clock domains is started.

**Bit 3 – STATUS** STATUS Synchronization Busy
This bit is cleared when the synchronization of STATUS register between the clock domains is complete.
This bit is set when the synchronization of STATUS register between clock domains is started.

**Bit 2 – CTRLB** CTRLB Synchronization Busy
This bit is cleared when the synchronization of CTRLB register between the clock domains is complete.
This bit is set when the synchronization of CTRLB register between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE bit between the clock domains is complete.
This bit is set when the synchronization of ENABLE bit between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST bit between the clock domains is complete.
This bit is set when the synchronization of SWRST bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.5. Fault Control A

**Name:** FCTRLA
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | FILTERVAL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | BLANKVAL[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | CAPTURE[2:0] | | | CHSEL[1:0] | | HALT[1:0] | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RESTART | BLANK[1:0] | | QUAL | KEEP | | SRC[1:0] | |
| Access | R/W | R/W | R/W | R/W | R/W | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | | 0 | 0 |

**Bits 27:24 – FILTERVAL[3:0]** Recoverable Fault A Filter Value
These bits define the filter value applied on MCE0 event input line. The value must be set to zero when MCE0 event is used as synchronous event.

**Bits 23:16 – BLANKVAL[7:0]** Recoverable Fault A Blanking Value
These bits determine the duration of the blanking of the fault input source. Activation and edge selection of the blank filtering are done by the BLANK bits (FCTRLA.BLANK).
When enabled, the fault input source is internally disabled for BLANKVAL* prescaled GCLK_TCC periods after the detection of the waveform edge.

**Bits 14:12 – CAPTURE[2:0]** Recoverable Fault A Capture Action
These bits select the capture and Fault A interrupt/event conditions.

**Table 36-8.** Fault A Capture Action

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Capture on valid recoverable Fault A is disabled |
| 0x1 | CAPT | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTA flag rises on each new captured value. |
| 0x2 | CAPTMIN | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0], if COUNT value is lower than the last stored capture value (CC). INTFLAG.FAULTA flag rises on each local minimum detection. |
| 0x3 | CAPTMAX | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0], if COUNT value is higher than the last stored capture value (CC). INTFLAG.FAULTA flag rises on each local maximum detection. |
| 0x4 | LOCMIN | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTA flag rises on each local minimum value detection. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 36-8.** Fault A Capture Action (continued)

| Value | Name | Description |
|---|---|---|
| 0x5 | LOCMAX | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTA flag rises on each local maximum detection. |
| 0x6 | DERIV0 | On rising edge of a valid recoverable Fault A, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTA flag rises on each local maximum or minimum detection. |
| 0x7 | CAPTMARK | Capture with ramp index as MSB value. |

**Bits 11:10 – CHSEL[1:0]** Recoverable Fault A Capture Channel
These bits select the channel for capture operation triggered by recoverable Fault A.

| Value | Name | Description |
|---|---|---|
| 0x0 | CC0 | Capture value stored into CC0 |
| 0x1 | CC1 | Capture value stored into CC1 |
| 0x2 | CC2 | Capture value stored into CC2 |
| 0x3 | CC3 | Capture value stored into CC3 |

**Bits 9:8 – HALT[1:0]** Recoverable Fault A Halt Operation
These bits select the halt action for recoverable Fault A.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Halt action disabled |
| 0x1 | HW | Hardware halt action |
| 0x2 | SW | Software halt action |
| 0x3 | NR | Non-recoverable fault |

**Bit 7 – RESTART** Recoverable Fault A Restart
Setting this bit enables restart action for Fault A.

| Value | Description |
|---|---|
| 0 | Fault A restart action is disabled. |
| 1 | Fault A restart action is enabled. |

**Bits 6:5 – BLANK[1:0]** Recoverable Fault A Blanking Operation
These bits, select the blanking start point for recoverable Fault A.

| Value | Name | Description |
|---|---|---|
| 0x0 | START | Blanking applied from start of the Ramp period |
| 0x1 | RISE | Blanking applied from rising edge of the waveform output |
| 0x2 | FALL | Blanking applied from falling edge of the waveform output |
| 0x3 | BOTH | Blanking applied from each toggle of the waveform output |

**Bit 4 – QUAL** Recoverable Fault A Qualification
Setting this bit enables the recoverable Fault A input qualification.

| Value | Description |
|---|---|
| 0 | The recoverable Fault A input is not disabled on CMPx value condition. |
| 1 | The recoverable Fault A input is disabled when output signal is at inactive level (CMPx == 0). |

**Bit 3 – KEEP** Recoverable Fault A Keep
Setting this bit enables the Fault A keep action.

| Value | Description |
|---|---|
| 0 | The Fault A state is released as soon as the recoverable Fault A is released. |
| 1 | The Fault A state is released at the end of TCC cycle. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 1:0 – SRC[1:0]** Recoverable Fault A Source

These bits select the TCC event input for recoverable Fault A.
Event system channel connected to MCE0 event input, must be configured to route the event asynchronously, when used as a recoverable Fault A input.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Fault input disabled |
| 0x1 | ENABLE | MCE0 event input |
| 0x2 | INVERT | Inverted MCE0 event input |
| 0x3 | ALTFAULT | Alternate fault B state at the end of the previous period. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.6. Fault Control B

**Name:** FCTRLB
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | FILTERVAL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | BLANKVAL[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | CAPTURE[2:0] | | | CHSEL[1:0] | | HALT[1:0] | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RESTART | BLANK[1:0] | | QUAL | KEEP | | SRC[1:0] | |
| Access | R/W | R/W | R/W | R/W | R/W | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | | 0 | 0 |

**Bits 27:24 – FILTERVAL[3:0]** Recoverable Fault B Filter Value
These bits define the filter value applied on MCE1 event input line. The value must be set to zero when MCE1 event is used as synchronous event.

**Bits 23:16 – BLANKVAL[7:0]** Recoverable Fault B Blanking Value
These bits determine the duration of the blanking of the fault input source. Activation and edge selection of the blank filtering are done by the BLANK bits (FCTRLB.BLANK).
When enabled, the fault input source is internally disabled for BLANKVAL* prescaled GCLK_TCC periods after the detection of the waveform edge.

**Bits 14:12 – CAPTURE[2:0]** Recoverable Fault B Capture Action
These bits select the capture and Fault B interrupt/event conditions.

**Table 36-9.** Fault B Capture Action

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Capture on valid recoverable Fault B is disabled |
| 0x1 | CAPT | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTB flag rises on each new captured value. |
| 0x2 | CAPTMIN | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0], if COUNT value is lower than the last stored capture value (CC). INTFLAG.FAULTB flag rises on each local minimum detection. |
| 0x3 | CAPTMAX | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0], if COUNT value is higher than the last stored capture value (CC). INTFLAG.FAULTB flag rises on each local maximum detection. |
| 0x4 | LOCMIN | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTB flag rises on each local minimum value detection. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 36-9.** Fault B Capture Action (continued)

| Value | Name | Description |
|---|---|---|
| 0x5 | LOCMAX | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTB flag rises on each local maximum detection. |
| 0x6 | DERIV0 | On rising edge of a valid recoverable Fault B, capture counter value on channel selected by CHSEL[1:0]. INTFLAG.FAULTB flag rises on each local maximum or minimum detection. |
| 0x7 | CAPTMARK | Capture with ramp index as MSB value. |

**Bits 11:10 – CHSEL[1:0]** Recoverable Fault B Capture Channel
These bits select the channel for capture operation triggered by recoverable Fault B.

| Value | Name | Description |
|---|---|---|
| 0x0 | CC0 | Capture value stored into CC0 |
| 0x1 | CC1 | Capture value stored into CC1 |
| 0x2 | CC2 | Capture value stored into CC2 |
| 0x3 | CC3 | Capture value stored into CC3 |

**Bits 9:8 – HALT[1:0]** Recoverable Fault B Halt Operation
These bits select the halt action for recoverable Fault B.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Halt action disabled |
| 0x1 | HW | Hardware halt action |
| 0x2 | SW | Software halt action |
| 0x3 | NR | Non-recoverable fault |

**Bit 7 – RESTART** Recoverable Fault B Restart
Setting this bit enables restart action for Fault B.

| Value | Description |
|---|---|
| 0 | Fault B restart action is disabled. |
| 1 | Fault B restart action is enabled. |

**Bits 6:5 – BLANK[1:0]** Recoverable Fault B Blanking Operation
These bits, select the blanking start point for recoverable Fault B.

| Value | Name | Description |
|---|---|---|
| 0x0 | START | Blanking applied from start of the Ramp period |
| 0x1 | RISE | Blanking applied from rising edge of the waveform output |
| 0x2 | FALL | Blanking applied from falling edge of the waveform output |
| 0x3 | BOTH | Blanking applied from each toggle of the waveform output |

**Bit 4 – QUAL** Recoverable Fault B Qualification
Setting this bit enables the recoverable Fault B input qualification.

| Value | Description |
|---|---|
| 0 | The recoverable Fault B input is not disabled on CMPx value condition. |
| 1 | The recoverable Fault B input is disabled when output signal is at inactive level (CMPx == 0). |

**Bit 3 – KEEP** Recoverable Fault B Keep
Setting this bit enables the Fault B keep action.

| Value | Description |
|---|---|
| 0 | The Fault B state is released as soon as the recoverable Fault B is released. |
| 1 | The Fault B state is released at the end of TCC cycle. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 1:0 – SRC[1:0]** Recoverable Fault B Source

These bits select the TCC event input for recoverable Fault B.
Event system channel connected to MCEx event input, must be configured to route the event asynchronously, when used as a recoverable Fault B input.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Fault input disabled |
| 0x1 | ENABLE | MCE1 event input |
| 0x2 | INVERT | Inverted MCE1 event input |
| 0x3 | ALTFAULT | Alternate fault A state at the end of the previous period. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.7. Waveform Extension Control

**Name:** WEXCTRL
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DTHS[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DTLS[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DTIEN3 | DTIEN2 | DTIEN1 | DTIEN0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | OTMX[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 31:24 – DTHS[7:0]** Dead-Time High Side Outputs Value
This register holds the number of GCLK_TCC clock cycles for the dead-time high side.

**Bits 23:16 – DTLS[7:0]** Dead-time Low Side Outputs Value
This register holds the number of GCLK_TCC clock cycles for the dead-time low side.

**Bits 8, 9, 10, 11 – DTIEN** Dead-time Insertion Generator x Enable
Setting any of these bits enables the dead-time insertion generator for the corresponding output matrix. This will override the output matrix [x] and [x+WO_NUM/2], with the low side and high side waveform respectively.

| Value | Description |
|---|---|
| 0 | No dead-time insertion override. |
| 1 | Dead time insertion override on signal outputs[x] and [x+WO_NUM/2], from matrix outputs[x] signal. |

**Bits 1:0 – OTMX[1:0]** Output Matrix
These bits define the matrix routing of the TCC waveform generation outputs to the port pins, according to Table 36-4.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.8. Driver Control

**Name:** DRVCTRL
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | FILTERVAL1[3:0] | | | | FILTERVAL0[3:0] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | INVEN7 | INVEN6 | INVEN5 | INVEN4 | INVEN3 | INVEN2 | INVEN1 | INVEN0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NRV7 | NRV6 | NRV5 | NRV4 | NRV3 | NRV2 | NRV1 | NRV0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | NRE7 | NRE6 | NRE5 | NRE4 | NRE3 | NRE2 | NRE1 | NRE0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:28 – FILTERVAL1[3:0]** Non-Recoverable Fault Input 1 Filter Value
These bits define the filter value applied on TCE1 event input line. When the TCE1 event input line is configured as a synchronous event, this value must be 0x0.

**Bits 27:24 – FILTERVAL0[3:0]** Non-Recoverable Fault Input 0 Filter Value
These bits define the filter value applied on TCE0 event input line. When the TCE0 event input line is configured as a synchronous event, this value must be 0x0.

**Bits 16, 17, 18, 19, 20, 21, 22, 23 – INVEN** Waveform Output x Inversion
These bits are used to select inversion on the output of channel x.
Writing a '1' to INVENx inverts output from WO[x].
Writing a '0' to INVENx disables inversion of output from WO[x].

**Bits 8, 9, 10, 11, 12, 13, 14, 15 – NRV** NRVx Non-Recoverable State x Output Value
These bits define the value of the enabled override outputs, under non-recoverable fault condition.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – NRE** Non-Recoverable State x Output Enable
These bits enable the override of individual outputs by NRVx value, under non-recoverable fault condition.

| Value | Description |
|---|---|
| 0 | Non-recoverable fault tri-state the output. |
| 1 | Non-recoverable faults set the output to NRVx level. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.9. Debug control

**Name:** DBGCTRL
**Offset:** 0x1E
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | FDDBD | | DBGRUN |
| Access | | | | | | R/W | | R/W |
| Reset | | | | | | 0 | | 0 |

**Bit 2 – FDDBD** Fault Detection on Debug Break Detection
This bit is not affected by software reset and should not be changed by software while the TCC is enabled.
By default this bit is zero, and the on-chip debug (OCD) fault protection is disabled. When this bit is written to '1', OCD break request from the OCD system will trigger non-recoverable fault. When this bit is set, OCD fault protection is enabled and OCD break request from the OCD system will trigger a non-recoverable fault.

| Value | Description |
|---|---|
| 0 | No faults are generated when TCC is halted in debug mode. |
| 1 | A non recoverable fault is generated and FAULTD flag is set when TCC is halted in debug mode. |

**Bit 0 – DBGRUN** Debug Running State
This bit is not affected by software reset and should not be changed by software while the TCC is enabled.

| Value | Description |
|---|---|
| 0 | The TCC is halted when the device is halted in debug mode. |
| 1 | The TCC continues normal operation when the device is halted in debug mode. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.10. Event Control

**Name:** EVCTRL
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MCEO3 | MCEO2 | MCEO1 | MCEO0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MCEI3 | MCEI2 | MCEI1 | MCEI0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TCEI1 | TCEI0 | TCINV1 | TCINV0 | | CNTEO | TRGEO | OVFEO |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CNTSEL[1:0] | | EVACT1[2:0] | | | EVACT0[2:0] | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – MCEOx** Match or Capture Channel x Event Output Enable
These bits control if the match/capture event on channel x is enabled and will be generated for every match or capture.

| Value | Description |
|---|---|
| 0 | Match/capture x event is disabled and will not be generated. |
| 1 | Match/capture x event is enabled and will be generated for every compare/capture on channel x. |

**Bits 16, 17, 18, 19 – MCEIx** Match or Capture Channel x Event Input Enable
These bits indicate if the match/capture x incoming event is enabled
These bits are used to enable match or capture input events to the CCx channel of TCC.

| Value | Description |
|---|---|
| 0 | Incoming events are disabled. |
| 1 | Incoming events are enabled. |

**Bits 14, 15 – TCEIx** Timer/Counter Event Input x Enable
This bit is used to enable input event x to the TCC.

| Value | Description |
|---|---|
| 0 | Incoming event x is disabled. |
| 1 | Incoming event x is enabled. |

**Bits 12, 13 – TCINVx** Timer/Counter Event x Invert Enable
This bit inverts the event x input.

| Value | Description |
|---|---|
| 0 | Input event source x is not inverted. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Input event source x is inverted. |

**Bit 10 – CNTEO** Timer/Counter Event Output Enable

This bit is used to enable the counter cycle event. When enabled, an event will be generated on begin or end of counter cycle depending of CNTSEL[1:0] settings.

| Value | Description |
|---|---|
| 0 | Counter cycle output event is disabled and will not be generated. |
| 1 | Counter cycle output event is enabled and will be generated depend of CNTSEL[1:0] value. |

**Bit 9 – TRGEO** Retrigger Event Output Enable

This bit is used to enable the counter retrigger event. When enabled, an event will be generated when the counter retriggers operation.

| Value | Description |
|---|---|
| 0 | Counter retrigger event is disabled and will not be generated. |
| 1 | Counter retrigger event is enabled and will be generated for every counter retrigger. |

**Bit 8 – OVFEO** Overflow/Underflow Event Output Enable

This bit is used to enable the overflow/underflow event. When enabled an event will be generated when the counter reaches the TOP or the ZERO value.

| Value | Description |
|---|---|
| 0 | Overflow/underflow counter event is disabled and will not be generated. |
| 1 | Overflow/underflow counter event is enabled and will be generated for every counter overflow/underflow. |

**Bits 7:6 – CNTSEL[1:0]** Timer/Counter Interrupt and Event Output Selection

These bits define on which part of the counter cycle the counter event output is generated.

| Value | Name | Description |
|---|---|---|
| 0x0 | BEGIN | An interrupt/event is generated at begin of each counter cycle |
| 0x1 | END | An interrupt/event is generated at end of each counter cycle |
| 0x2 | BETWEEN | An interrupt/event is generated between each counter cycle. |
| 0x3 | BOUNDARY | An interrupt/event is generated at begin of first counter cycle, and end of last counter cycle. |

**Bits 5:3 – EVACT1[2:0]** Timer/Counter Event Input 1 Action

These bits define the action the TCC will perform on TCE1 event input.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Event action disabled. |
| 0x1 | RETRIGGER | Start, restart or re-trigger TC on event |
| 0x2 | DIR (asynch) | Direction control |
| 0x3 | STOP | Stop TC on event |
| 0x4 | DEC | Decrement TC on event |
| 0x5 | PPW | Period captured into CC0 Pulse Width on CC1 |
| 0x6 | PWP | Period captured into CC1 Pulse Width on CC0 |
| 0x7 | FAULT | Non-recoverable Fault |

**Bits 2:0 – EVACT0[2:0]** Timer/Counter Event Input 0 Action

These bits define the action the TCC will perform on TCE0 event input 0.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Event action disabled. |
| 0x1 | RETRIGGER | Start, restart or re-trigger TC on event |
| 0x2 | COUNTEV | Count on event. |
| 0x3 | START | Start TC on event |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x4 | INC | Increment TC on EVENT |
| 0x5 | COUNT (async) | Count on active state of asynchronous event |
| 0x6 | STAMP | Capture overflow times (Max value) |
| 0x7 | FAULT | Non-recoverable Fault |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.11. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MC3 | MC2 | MC1 | MC0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ERR | CNT | TRG | OVF |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – MC** Match or Capture Channel x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the corresponding Match or Capture Channel x Interrupt Disable/Enable bit, which disables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 15 – FAULT1** Non-Recoverable Fault x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault x Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault x interrupt.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Fault x interrupt is disabled. |
| 1 | The Non-Recoverable Fault x interrupt is enabled. |

**Bit 14 – FAULT0** Non-Recoverable Fault x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault x Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault x interrupt.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Fault x interrupt is disabled. |
| 1 | The Non-Recoverable Fault x interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 13 – FAULTB** Recoverable Fault B Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Recoverable Fault B Interrupt Disable/Enable bit, which disables the Recoverable Fault B interrupt.

| Value | Description |
|---|---|
| 0 | The Recoverable Fault B interrupt is disabled. |
| 1 | The Recoverable Fault B interrupt is enabled. |

**Bit 12 – FAULTA** Recoverable Fault A Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Recoverable Fault A Interrupt Disable/Enable bit, which disables the Recoverable Fault A interrupt.

| Value | Description |
|---|---|
| 0 | The Recoverable Fault A interrupt is disabled. |
| 1 | The Recoverable Fault A interrupt is enabled. |

**Bit 11 – DFS** Non-Recoverable Debug Fault Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Debug Fault State Interrupt Disable/Enable bit, which disables the Debug Fault State interrupt.

| Value | Description |
|---|---|
| 0 | The Debug Fault State interrupt is disabled. |
| 1 | The Debug Fault State interrupt is enabled. |

**Bit 10 – UFS** Non-Recoverable Update Fault Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Non-Recoverable Update Fault Interrupt Disable/Enable bit, which disables the Non-Recoverable Update Fault interrupt.
**Note:** This bit is only available on variant L devices. Refer to the *Configuration Summary* for more information.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Update Fault interrupt is disabled. |
| 1 | The Non-Recoverable Update Fault interrupt is enabled. |

**Bit 3 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Error Interrupt Disable/Enable bit, which disables the Compare interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 2 – CNT** Counter Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Counter Interrupt Disable/Enable bit, which disables the Counter interrupt.

| Value | Description |
|---|---|
| 0 | The Counter interrupt is disabled. |
| 1 | The Counter interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – TRG** Retrigger Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Retrigger Interrupt Disable/Enable bit, which disables the Retrigger interrupt.

| Value | Description |
|---|---|
| 0 | The Retrigger interrupt is disabled. |
| 1 | The Retrigger interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overflow Interrupt Disable/Enable bit, which disables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.12. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x28
**Reset:** 0x00000000
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MC3 | MC2 | MC1 | MC0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ERR | CNT | TRG | OVF |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – MC** Match or Capture Channel x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the corresponding Match or Capture Channel x Interrupt Disable/Enable bit, which enables the Match or Capture Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match or Capture Channel x interrupt is disabled. |
| 1 | The Match or Capture Channel x interrupt is enabled. |

**Bit 15 – FAULT1** Non-Recoverable Fault x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Non-Recoverable Fault x Interrupt Disable/Enable bit, which enables the Non-Recoverable Fault x interrupt.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Fault x interrupt is disabled. |
| 1 | The Non-Recoverable Fault x interrupt is enabled. |

**Bit 14 – FAULT0** Non-Recoverable Fault x Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Non-Recoverable Fault x Interrupt Disable/Enable bit, which disables the Non-Recoverable Fault x interrupt.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Fault x interrupt is disabled. |
| 1 | The Non-Recoverable Fault x interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 13 – FAULTB** Recoverable Fault B Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Recoverable Fault B Interrupt Disable/Enable bit, which enables the Recoverable Fault B interrupt.

| Value | Description |
|---|---|
| 0 | The Recoverable Fault B interrupt is disabled. |
| 1 | The Recoverable Fault B interrupt is enabled. |

**Bit 12 – FAULTA** Recoverable Fault A Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Recoverable Fault A Interrupt Disable/Enable bit, which enables the Recoverable Fault A interrupt.

| Value | Description |
|---|---|
| 0 | The Recoverable Fault A interrupt is disabled. |
| 1 | The Recoverable Fault A interrupt is enabled. |

**Bit 11 – DFS** Non-Recoverable Debug Fault Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Debug Fault State Interrupt Disable/Enable bit, which enables the Debug Fault State interrupt.

| Value | Description |
|---|---|
| 0 | The Debug Fault State interrupt is disabled. |
| 1 | The Debug Fault State interrupt is enabled. |

**Bit 10 – UFS** Non-Recoverable Update Fault Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Non-Recoverable Update Fault Interrupt Disable/Enable bit, which enables the Non-Recoverable Update Fault interrupt.
**Note:** This bit is only available on variant L devices. Refer to the *Configuration Summary* for more information.

| Value | Description |
|---|---|
| 0 | The Non-Recoverable Update Fault interrupt is disabled. |
| 1 | The Non-Recoverable Update Fault interrupt is enabled. |

**Bit 3 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Error Interrupt Disable/Enable bit, which enables the Compare interrupt.

| Value | Description |
|---|---|
| 0 | The Error interrupt is disabled. |
| 1 | The Error interrupt is enabled. |

**Bit 2 – CNT** Counter Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Retrigger Interrupt Disable/Enable bit, which enables the Counter interrupt.

| Value | Description |
|---|---|
| 0 | The Counter interrupt is disabled. |
| 1 | The Counter interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 1 – TRG** Retrigger Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Retrigger Interrupt Disable/Enable bit, which enables the Retrigger interrupt.

| Value | Description |
|---|---|
| 0 | The Retrigger interrupt is disabled. |
| 1 | The Retrigger interrupt is enabled. |

**Bit 0 – OVF** Overflow Interrupt Enable

Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overflow Interrupt Disable/Enable bit, which enables the Overflow interrupt request.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.13. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x2C
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MC3 | MC2 | MC1 | MC0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FAULT1 | FAULT0 | FAULTB | FAULTA | DFS | UFS | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ERR | CNT | TRG | OVF |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – MCx** Match or Capture Channel x Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a match with the compare condition or once CCx register contain a valid capture value.
Writing a '0' to one of these bits has no effect.
Writing a '1' to one of these bits will clear the corresponding Match or Capture Channel x interrupt flag
In Capture operation, this flag is automatically cleared when CCx register is read.

**Bit 15 – FAULT1** Non-Recoverable Fault 1 Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Non-Recoverable Fault x occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Non-Recoverable Fault 1 interrupt flag.

**Bit 14 – FAULT0** Non-Recoverable Fault 0 Interrupt Flag
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Non-Recoverable Fault 0 interrupt flag.

**Bit 13 – FAULTB** Recoverable Fault B Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Recoverable Fault B occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Recoverable Fault B interrupt flag.

**Bit 12 – FAULTA** Recoverable Fault A Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a Recoverable Fault B occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Recoverable Fault A interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 11 – DFS** Non-Recoverable Debug Fault State Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after an Debug Fault State occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Debug Fault State interrupt flag.

**Bit 10 – UFS** Non-Recoverable Update Fault
This flag is set when the RAMP index changes and the Lock Update bit is set (CTRLBSET.LUPD).
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Non-Recoverable Update Fault interrupt flag.
**Note:** This bit is only available on variant L devices. Refer to the *Configuration Summary* for more information.

**Bit 3 – ERR** Error Interrupt Flag
This flag is set if a new capture occurs on a channel when the corresponding Match or Capture Channel x interrupt flag is one. In which case there is nowhere to store the new capture.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the error interrupt flag.

**Bit 2 – CNT** Counter Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a counter event occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the CNT interrupt flag.

**Bit 1 – TRG** Retrigger Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after a counter retrigger occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the re-trigger interrupt flag.

**Bit 0 – OVF** Overflow Interrupt Flag
This flag is set on the next CLK_TCC_COUNT cycle after an overflow condition occurs.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overflow interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.14. Status

**Name:** STATUS
**Offset:** 0x30
**Reset:** 0x00000001
**Property:** Write-Synchronized, Read-Synchronized

**Note:** This register is read- and write-synchronized: SYNCBUSY.STATUS must be checked to ensure the STATUS register synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CMP3 | CMP2 | CMP1 | CMP0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CCBUFV3 | CCBUFV2 | CCBUFV1 | CCBUFV0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FAULT1 | FAULT0 | FAULTB | FAULTA | FAULT1IN | FAULT0IN | FAULTBIN | FAULTAIN |
| Access | R/W | R/W | R/W | R/W | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PERBUFV | WAVEBUFV | PATTBUFV | SLAVE | DFS | UFS | IDX | STOP |
| Access | R/W | R/W | R/W | R | R/W | R/W | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 24, 25, 26, 27 – CMP** Channel x Compare Value
This bit reflects the channel x output compare value.

| Value | Description |
|---|---|
| 0 | Channel compare output value is 0. |
| 1 | Channel compare output value is 1. |

**Bits 16, 17, 18, 19 – CCBUFV** Channel x Compare or Capture Buffer Valid
For a compare channel, this bit is set when a new value is written to the corresponding CCBUFx register. The bit is cleared either by writing a '1' to the corresponding location when CTRLB.LUPD is set, or automatically on an UPDATE condition.
For a capture channel, the bit is set when a valid capture value is stored in the CCBUFx register. The bit is automatically cleared when the CCx register is read.

**Bits 14, 15 – FAULT** Non-recoverable Fault x State
This bit is set by hardware as soon as non-recoverable Fault x condition occurs.
This bit is cleared by writing a one to this bit and when the corresponding FAULTxIN status bit is low. Once this bit is clear, the timer/counter will restart from the last COUNT value. To restart the timer/counter from BOTTOM, the timer/counter restart command must be executed before clearing the corresponding STATEx bit. For further details on timer/counter commands, refer to available commands description (CTRLBSET.CMD).

**Bit 13 – FAULTB** Recoverable Fault B State
This bit is set by hardware as soon as recoverable Fault B condition occurs.


© 2026 Microchip Technology Inc. and its subsidiaries

This bit can be clear by hardware when Fault B action is resumed, or by writing a '1' to this bit when the corresponding FAULTBIN bit is low. If software halt command is enabled (FAULTB.HALT=SW), clearing this bit will release the timer/counter.

**Bit 12 – FAULTA** Recoverable Fault A State
This bit is set by hardware as soon as recoverable Fault A condition occurs.
This bit can be clear by hardware when Fault A action is resumed, or by writing a '1' to this bit when the corresponding FAULTAIN bit is low. If software halt command is enabled (FAULTA.HALT=SW), clearing this bit will release the timer/counter.

**Bit 11 – FAULT1IN** Non-Recoverable Fault 1 Input
This bit is set while an active Non-Recoverable Fault 1 input is present.

**Bit 10 – FAULT0IN** Non-Recoverable Fault 0 Input
This bit is set while an active Non-Recoverable Fault 0 input is present.

**Bit 9 – FAULTBIN** Recoverable Fault B Input
This bit is set while an active Recoverable Fault B input is present.

**Bit 8 – FAULTAIN** Recoverable Fault A Input
This bit is set while an active Recoverable Fault A input is present.

**Bit 7 – PERBUFV** Period Buffer Valid
This bit is set when a new value is written to the PERBUF register. This bit is automatically cleared by hardware on UPDATE condition when CTRLB.LUPD is set, or by writing a '1' to this bit.

**Bit 6 – WAVEBUFV** Waveform Control Buffer Valid
This bit is set when a new value is written to the WAVEBUF register. This bit is automatically cleared by hardware on UPDATE condition when CTRLB.LUPD is set, or by writing a '1' to this bit.

**Bit 5 – PATTBUFV** Pattern Generator Value Buffer Valid
This bit is set when a new value is written to the PATTBUF register. This bit is automatically cleared by hardware on UPDATE condition when CTRLB.LUPD is set, or by writing a '1' to this bit.

**Bit 4 – SLAVE** Client
This bit is set when TCC is set in Client mode. This bit follows the CTRLA.MSYNC bit state.

**Bit 3 – DFS** Debug Fault State
This bit is set by hardware in Debug mode when DDBGCTRL.FDDBD bit is set. The bit is cleared by writing a '1' to this bit and when the TCC is not in Debug mode.
When the bit is set, the counter is halted and the Waveforms state depend on DRVCTRL.NRE and DRVCTRL.NRV registers.

**Bit 2 – UFS** Non-recoverable Update Fault State
This bit is set by hardware when the RAMP index changes and the Lock Update bit is set (CTRLBSET.LUPD). The bit is cleared by writing a one to this bit.
When the bit is set, the waveforms state depend on DRVCTRL.NRE and DRVCTRL.NRV registers.

**Bit 1 – IDX** Ramp Index
In RAMP2 and RAMP2A operation, the bit is cleared during the cycle A and set during the cycle B. In RAMP1 operation, the bit always reads zero. For details on ramp operations, refer to Ramp Operations.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 0 – STOP** Stop

This bit is set when the TCC is disabled either on a STOP command or on an UPDATE condition when One-Shot operation mode is enabled (CTRLBSET.ONESHOT=1).
This bit is clear on the next incoming counter increment or decrement.

| Value | Description |
|---|---|
| 0 | Counter is running. |
| 1 | Counter is stopped. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.15. Counter Value

**Name:** COUNT
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

**Note:** Prior to any read access, this register must be synchronized by the user by writing the according TCC Command value to the Control B Set register (CTRLBSET.CMD = READSYNC).

**Note:** This register is write-synchronized: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – COUNT[23:0]** Counter Value

These bits hold the value of the counter register.

**Note:** When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

**Note:** This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [23:m] |
|---|---|
| 0x0 - NONE | 23:0 (depicted) |
| 0x1 - DITH4 | 23:4 |
| 0x2 - DITH5 | 23:5 |
| 0x3 - DITH6 | 23:6 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.16. Pattern

**Name:** PATT
**Offset:** 0x38
**Reset:** 0x0000
**Property:** Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.PATT must be checked to ensure the PATT register synchronization is complete.

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PGV7 | PGV6 | PGV5 | PGV4 | PGV3 | PGV2 | PGV1 | PGV0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PGE7 | PGE6 | PGE5 | PGE4 | PGE3 | PGE2 | PGE1 | PGE0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 8, 9, 10, 11, 12, 13, 14, 15 – PGVx** Pattern Generation Output Value x [x = 7..0]
This register holds the values of pattern for each waveform output.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PGEx** Pattern Generation Output Enable x [x = 7..0]
This register holds the enable status of pattern generation for each waveform output. A bit written to '1' will override the corresponding SWAP output with the corresponding PGVx value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.17. Waveform

**Name:** WAVE
**Offset:** 0x3C
**Reset:** 0x00000000
**Property:** Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SWAP3 | SWAP2 | SWAP1 | SWAP0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | POL3 | POL2 | POL1 | POL0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CICCEN3 | CICCEN2 | CICCEN1 | CICCEN0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CIPEREN | | RAMP[1:0] | | | WAVEGEN[2:0] | | |
| Access | R/W | | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – SWAPx** Swap DTI Output Pair x [x = 3..0]
Setting these bits enables output swap of DTI outputs [x] and [x+WO_NUM/2]. Note the DTIxEN settings will not affect the swap operation.
**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.SWAPx synchronization is complete.

**Bits 16, 17, 18, 19 – POLx** Channel Polarity x [x = 3..0]
Setting these bits enables the output polarity in single-slope and dual-slope PWM operations.
**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.POLx synchronization is complete.

| Value | Name | Description |
|---|---|---|
| 0 | (single-slope PWM waveform generation) | Compare output is initialized to ~DIR and set to DIR when TCC counter matches CCx value |
| 1 | (single-slope PWM waveform generation) | Compare output is initialized to DIR and set to ~DIR when TCC counter matches CCx value. |
| 0 | (dual-slope PWM waveform generation) | Compare output is set to ~DIR when TCC counter matches CCx value |
| 1 | (dual-slope PWM waveform generation) | Compare output is set to DIR when TCC counter matches CCx value. |

**Bits 8, 9, 10, 11 – CICCENx** Circular CC Enable x [x = 3..0]
Setting this bits enables the compare circular buffer option on channel. When the bit is set, CCx register value is copied-back into the CCx register on UPDATE condition.
**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.CICCENx synchronization is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 7 – CIPEREN** Circular Period Enable

Setting this bits enable the period circular buffer option. When the bit is set, the PER register value is copied-back into the PERB register on UPDATE condition.

**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.CIPEREN synchronization is complete.

**Bits 5:4 – RAMP[1:0]** Ramp Operation

These bits select Ramp operation (RAMP).

**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.RAMP synchronization is complete.

| Value | Name | Description |
|---|---|---|
| 0x0 | RAMP1 | RAMP1 operation |
| 0x1 | RAMP2A | Alternative RAMP2 operation |
| 0x2 | RAMP2 | RAMP2 operation |
| 0x3 | RAMP2C. This bit is only available in variant L devices. Refer to *Configuration Summary* for more information. | Critical RAMP2 operation |

**Bits 2:0 – WAVEGEN[2:0]** Waveform Generation Operation

These bits select the waveform generation operation. The settings impact the top value and control if frequency or PWM waveform generation should be used.

**Note:** SYNCBUSY.WAVE must be checked to ensure that WAVE.WAVEGEN synchronization is complete.

| Value | Name | Description | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | Operation | Top | Update | Waveform Output On Match | Waveform Output On Update | OVFIF/Event Up | Down |
| 0x0 | NFRQ | Normal Frequency | PER | TOP/Zero | Toggle | Stable | TOP | Zero |
| 0x1 | MFRQ | Match Frequency | CC0 | TOP/Zero | Toggle | Stable | TOP | Zero |
| 0x2 | NPWM | Normal PWM | PER | TOP/Zero | Set | Clear | TOP | Zero |
| 0x3 | Reserved | - | - | - | - | -- | - | - |
| 0x4 | DSCRITICAL | Dual-slope PWM | PER | Zero | ~DIR | Stable | – | Zero |
| 0x5 | DSBOTTOM | Dual-slope PWM | PER | Zero | ~DIR | Stable | – | Zero |
| 0x6 | DSBOTH | Dual-slope PWM | PER | TOP & Zero | ~DIR | Stable | TOP | Zero |
| 0x7 | DSTOP | Dual-slope PWM | PER | Zero | ~DIR | Stable | TOP | – |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.18. Period Value

**Name:** PER
**Offset:** 0x40
**Reset:** 0xFFFFFFFF
**Property:** Write-Synchronized

**Note:** This register is write-synchronized: SYCBUSY.PER must be checked to ensure the PER register synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PER[17:10] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PER[9:2] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PER[1:0] | | DITHER[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 23:6 – PER[17:0]** Period Value
These bits hold the value of the period buffer register.
**Note:** When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

**Note:** This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [23:m] |
|---|---|
| 0x0 - NONE | 23:0 |
| 0x1 - DITH4 | 23:4 |
| 0x2 - DITH5 | 23:5 |
| 0x3 - DITH6 | 23:6 (depicted) |

**Bits 5:0 – DITHER[5:0]** Dithering Cycle Number
These bits hold the number of extra cycles that are added on the PWM pulse period every 64 PWM frames.
**Note:** This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [n:0] |
|---|---|
| 0x0 - NONE | - |
| 0x1 - DITH4 | 3:0 |
| 0x2 - DITH5 | 4:0 |
| 0x3 - DITH6 | 5:0 (depicted) |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.19. Compare/Capture Channel x

**Name:** CCx
**Offset:** 0x44 + x*0x04 [x=0..3]
**Reset:** 0x00000000
**Property:** Write-Synchronized

The full offset for this register is 0x44 + n*0x04 [n = 0..3 for TCC0; n = 0,1 for TCC1 and TCC2].

The CCx register represents the 16-bit, 24- bit value, CCx. The register has the following two functions, depending on the mode of operation:

For capture operation, this register represents the second buffer level and access point for the CPU and DMA.

For compare operation, this register is continuously compared to the counter value. Normally, the output form the comparator is then used for generating waveforms.

The CCx register is updated with the buffer value from their corresponding CCBUFx register when an UPDATE condition occurs.

In addition, in match frequency operation, the CC0 register controls the counter period.

**Note:** This bit is write-synchronized. SYNCBUSY.CCx must be checked to ensure that CCx synchronization is complete.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CC[17:10] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CC[9:2] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CC[1:0] | | DITHER[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:6 – CC[17:0]** Channel x Compare/Capture Value
These bits hold the value of the Channel x Compare/Capture register.
**Note:** When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

**Note:** This bit field occupies the m MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [23:m] |
|---|---|
| 0x0 - NONE | 23:0 |
| 0x1 - DITH4 | 23:4 |
| 0x2 - DITH5 | 23:5 |
| 0x3 - DITH6 | 23:6 (depicted) |


© 2026 Microchip Technology Inc. and its subsidiaries

**Bits 5:0 – DITHER[5:0]** Dithering Cycle Number

These bits hold the number of extra cycles that are added on the PWM pulse width every 64 PWM frames.

**Note:** This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [n:0] |
|---|---|
| 0x0 - NONE | - |
| 0x1 - DITH4 | 3:0 |
| 0x2 - DITH5 | 4:0 |
| 0x3 - DITH6 | 5:0 (depicted) |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.20. Pattern Buffer

**Name:** PATTBUF
**Offset:** 0x64
**Reset:** 0x0000
**Property:** Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.PATT must be checked to ensure the PATT register synchronization is complete. This register must be written with 16-bit accesses only (no 8-bit writes).

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PGVB7 | PGVB6 | PGVB5 | PGVB4 | PGVB3 | PGVB2 | PGVB1 | PGVB0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PGEB7 | PGEB6 | PGEB5 | PGEB4 | PGEB3 | PGEB2 | PGEB1 | PGEB0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 8, 9, 10, 11, 12, 13, 14, 15 – PGVBx** Pattern Generation Output Value Buffer x [x = 7..0]
This register is the buffer for the PGV register. If double buffering is used, valid content in this register is copied to the PGV register on an UPDATE condition or CTRLBSET.CMD = UPDATE command when CTRLBSET.LUPD = 1.

**Bits 0, 1, 2, 3, 4, 5, 6, 7 – PGEBx** Pattern Generation Output Enable Buffer x [x = 7..0]
This register is the buffer of the PGE register. If double buffering is used, valid content in this register is copied into the PGE register at an UPDATE condition or CTRLBSET.CMD = UPDATE command when CTRLBSET.LUPD = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.21. Waveform Buffer

**Name:** WAVEBUF
**Offset:** 0x68
**Reset:** 0x00000000
**Property:** Write-Synchronized

**Note:** This register is write-synchronized: SYNCBUSY.WAVE must be checked to ensure the WAVE register synchronization is complete. This register must be written with 32-bit accesses only (no 8-bit or 16-bit writes).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SWAPB 3 | SWAPB 2 | SWAPB 1 | SWAPB 0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | POLB3 | POLB2 | POLB1 | POLB0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CICCENB3 | CICCENB2 | CICCENB1 | CICCENB0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CIPERENB | | RAMPB[1:0] | | | WAVEGENB[2:0] | | |
| Access | R/W | | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – SWAPB** Swap DTI output pair x Buffer
These register bits are the buffer bits for the SWAP register bits. If double buffering is used, valid content in these bits is copied to the corresponding SWAPx bits on an UPDATE condition.

**Bits 16, 17, 18, 19 – POLB** Channel Polarity x Buffer
These register bits are the buffer bits for POLx register bits. If double buffering is used, valid content in these bits is copied to the corresponding POBx bits on an UPDATE condition.

**Bits 8, 9, 10, 11 – CICCENB** Circular CCx Buffer Enable
These register bits are the buffer bits for CICCENx register bits. If double buffering is used, valid content in these bits is copied to the corresponding CICCENx bits on a UPDATE condition.

**Bit 7 – CIPERENB** Circular Period Enable Buffer
This register bit is the buffer bit for CIPEREN register bit. If double buffering is used, valid content in this bit is copied to the corresponding CIPEREN bit on a UPDATE condition.

**Bits 5:4 – RAMPB[1:0]** Ramp Operation Buffer
These register bits are the buffer bits for RAMP register bits. If double buffering is used, valid content in these bits is copied to the corresponding RAMP bits on a UPDATE condition.

**Bits 2:0 – WAVEGENB[2:0]** Waveform Generation Operation Buffer
These register bits are the buffer bits for WAVEGEN register bits. If double buffering is used, valid content in these bits is copied to the corresponding WAVEGEN bits on a UPDATE condition.


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.22. Period Buffer Value

**Name:** PERBUF
**Offset:** 0x6C
**Reset:** 0xFFFFFFFF
**Property:** Write-Synchronized

**Note:** This bit is write-synchronized. SYNCBUSY.PER must be checked to ensure the PER register synchronization is complete. This register must be written with 32-bit accesses only (no 8-bit or 16-bit writes).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PERBUF[17:10] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | PERBUF[9:2] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PERBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Bits 23:6 – PERBUF[17:0]** Period Buffer Value
These bits hold the value of the period buffer register. The value is copied to PER register on UPDATE condition or CTRLBSET.CMD = UPDATE command when CTRLBSET.LUPD = 1.
**Note:** When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

**Note:** This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [23:m] |
|---|---|
| 0x0 - NONE | 23:0 |
| 0x1 - DITH4 | 23:4 |
| 0x2 - DITH5 | 23:5 |
| 0x3 - DITH6 | 23:6 (depicted) |

**Bits 5:0 – DITHERBUF[5:0]** Dithering Buffer Cycle Number
These bits represent the PER.DITHER bits buffer. When the double buffering is enabled, the value of this bit field is copied to the PER.DITHER bits on an UPDATE condition or CTRLBSET.CMD = UPDATE command when CTRLBSET.LUPD = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [n:0] |
|---|---|
| 0x0 - NONE | - |
| 0x1 - DITH4 | 3:0 |
| 0x2 - DITH5 | 4:0 |
| 0x3 - DITH6 | 5:0 (depicted) |


© 2026 Microchip Technology Inc. and its subsidiaries

### 36.8.23. Channel x Compare/Capture Buffer Value

**Name:** CCBUFx
**Offset:** 0x70 + x*0x04 [x=0..3]
**Reset:** 0x00000000
**Property:** Write-Synchronized

CCBUFx is copied into CCx at TCC update time.

**Note:** This register is write-synchronized: SYNCBUSY.CCx must be checked to ensure the CCx register synchronization is complete. This register must be written with 32-bit accesses only (no 8- or 16-bit writes).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[17:10] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[9:2] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CCBUF[1:0] | | DITHERBUF[5:0] | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:6 – CCBUF[17:0]** Channel x Compare/Capture Buffer Value
These bits hold the value of the Channel x Compare/Capture Buffer Value register. The register serves as the buffer for the associated compare or capture registers (CCx). Accessing this register using the CPU or DMA will affect the corresponding CCBUFVx status bit.
**Note:** When the TCC is configured as 16-bit timer/counter, the excess bits are read zero.

**Note:** This bit field occupies the MSB of the register, [23:m]. m is dependent on the Resolution bit in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [23:m] |
|---|---|
| 0x0 - NONE | 23:0 |
| 0x1 - DITH4 | 23:4 |
| 0x2 - DITH5 | 23:5 |
| 0x3 - DITH6 | 23:6 (depicted) |

**Bits 5:0 – DITHERBUF[5:0]** Dithering Buffer Cycle Number
These bits represent the CCx.DITHER bits buffer. When the double buffering is enable, DITHERBUF bits value is copied to the CCx.DITHER bits on an UPDATE condition or CTRLBSET.CMD = UPDATE command when CTRLBSET.LUPD = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:** This bit field consists of the n LSB of the register. n is dependent on the value of the Resolution bits in the Control A register (CTRLA.RESOLUTION):

| CTRLA.RESOLUTION | Bits [n:0] |
|---|---|
| 0x0 - NONE | - |
| 0x1 - DITH4 | 3:0 |
| 0x2 - DITH5 | 4:0 |
| 0x3 - DITH6 | 5:0 (depicted) |


© 2026 Microchip Technology Inc. and its subsidiaries

# 37. Configurable Custom Logic (CCL)

## 37.1. Overview

The Configurable Custom Logic (CCL) is a programmable logic peripheral which can be connected to the device pins, to events, or to other internal peripherals. This allows the user to eliminate logic gates for simple glue logic functions on the PCB.

Each LookUp Table (LUT) consists of three inputs, a truth table, an optional synchronizer/filter, and an optional edge detector. Each LUT can generate an output as a user programmable logic expression with three inputs. Inputs can be individually masked.

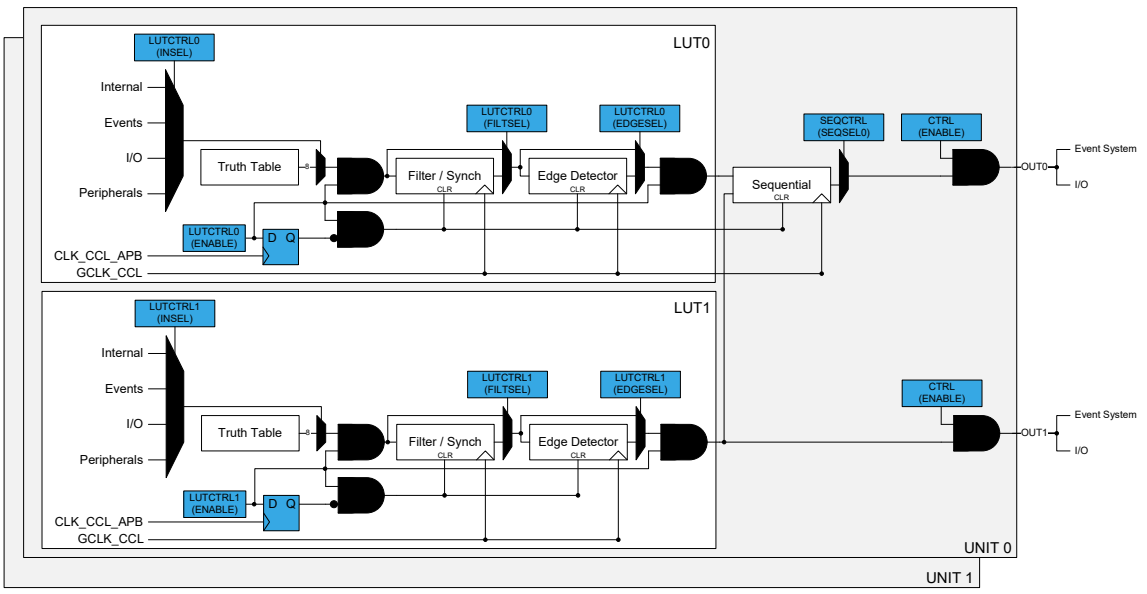
The output can be combinatorially generated from the inputs, and can be filtered to remove spikes. Optional sequential logic can be used. The inputs of the sequential module are individually controlled by two independent, adjacent LUT (LUT0/LUT1, LUT2/LUT3 etc.) outputs, enabling complex waveform generation.

## 37.2. Features

- Glue logic for general purpose PCB design
- 4 programmable LookUp Tables (LUTs)
- Combinatorial logic functions: AND, NAND, OR, NOR, XOR, XNOR, NOT
- Sequential logic functions: Gated D Flip-Flop, JK Flip-Flop, gated D Latch, RS Latch
- Flexible LUT inputs selection:
  - I/Os
  - Events
  - Internal peripherals
  - Subsequent LUT output
- Output can be connected to the I/O pins or the Event System
- Optional synchronizer, filter, or edge detector available on each LUT output

## 37.3. Block Diagram

**Figure 37-1.** Configurable Custom Logic





© 2026 Microchip Technology Inc. and its subsidiaries

## 37.4. Signal Description

| Pin Name | Type | Description |
|---|---|---|
| OUT[3:0] | Digital output | Output from lookup table |
| IN[11:0] | Digital input | Input to lookup table |

Refer to *"I/O Multiplexing and Considerations"* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**
I/O Multiplexing and Considerations

## 37.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 37.5.1. I/O Lines

The CCL can take inputs and generate output through I/O pins. For this to function properly, the I/O pins must be configured to be used by a Look Up Table (LUT).

**Related Links**
PORT - I/O Pin Controller

### 37.5.2. Power Management

This peripheral can continue to operate in any sleep mode where its source clock is running. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

**Related Links**
PM - Power Manager

### 37.5.3. Clocks

The CCL bus clock (CLK_CCL_APB) can be enabled and disabled in the Main Clock module, MCLK (see *MCLK - Main Clock*), and the default state of CLK_CCL_APB can be found in *Peripheral Clock Masking*.

A generic clock (GCLK_CCL) is optionally required to clock the CCL. This clock must be configured and enabled in the Generic Clock Controller (GCLK) before using input events, filter, edge detection or sequential logic. GCLK_CCL is required when input events, a filter, an edge detector, or a sequential sub-module is enabled. Refer to *GCLK - Generic Clock Controller* for details.

This generic clock is asynchronous to the user interface clock (CLK_CCL_APB).

**Related Links**
MCLK – Main Clock
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 37.5.4. DMA

Not applicable.

### 37.5.5. Interrupts

Not applicable.


© 2026 Microchip Technology Inc. and its subsidiaries

### 37.5.6. Events

The CCL can use events from other peripherals and generate events that can be used by other peripherals. For this feature to function, the events have to be configured properly. Refer to the Related Links below for more information about the event users and event generators.

**Related Links**
Event System (EVSYS)

### 37.5.7. Debug Operation

When the CPU is halted in Debug mode the CCL continues normal operation. However, the CCL cannot be halted when the CPU is halted in Debug mode. If the CCL is configured in a way that requires it to be periodically serviced by the CPU, improper operation or data loss may result during debugging.

### 37.5.8. Register Access Protection

All registers with write access can be write-protected optionally by the Peripheral Access Controller (PAC). Refer to *PAC - Peripheral Access Controller* for details.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 37.5.9. Analog Connections

Not applicable.

## 37.6. Functional Description

### 37.6.1. Principle of Operation

Configurable Custom Logic (CCL) is a programmable logic block that can use the device port pins, internal peripherals, and the internal Event System as both input and output channels. The CCL can serve as glue logic between the device and external devices. The CCL can eliminate the need for external logic component and can also help the designer overcome challenging real-time constrains by combining core independent peripherals in clever ways to handle the most time critical parts of the application independent of the CPU.

### 37.6.2. Operation

#### 37.6.2.1. Initialization

The following bits are enable-protected, meaning that they can only be written when the corresponding even LUT is disabled (LUTCTRLx.ENABLE=0):

- Sequential Selection bits in the Sequential Control x (SEQCTRLx.SEQSEL) register

The following registers are enable-protected, meaning that they can only be written when the corresponding LUT is disabled (LUTCTRLx.ENABLE=0):

- LUT Control x (LUTCTRLx) register, except the ENABLE bit

Enable-protected bits in the LUTCTRLx registers can be written at the same time as LUTCTRLx.ENABLE is written to '1', but not at the same time as LUTCTRLx.ENABLE is written to '0'.

Enable-protection is denoted by the Enable-Protected property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 37.6.2.2. Enabling, Disabling, and Resetting

The CCL is enabled by writing a '1' to the Enable bit in the Control register (CTRL.ENABLE). The CCL is disabled by writing a '0' to CTRL.ENABLE.

Each LUT is enabled by writing a '1' to the Enable bit in the LUT Control x register (LUTCTRLx.ENABLE). Each LUT is disabled by writing a '0' to LUTCTRLx.ENABLE.

The CCL is reset by writing a '1' to the Software Reset bit in the Control register (CTRL.SWRST). All registers in the CCL will be reset to their initial state, and the CCL will be disabled. Refer to CTRL for details.

### 37.6.2.3. Lookup Table Logic

The lookup table in each LUT unit can generate any logic expression OUT as a function of three inputs (IN[2:0]), as shown in Figure 37-2. One or more inputs can be masked. The truth table for the expression is defined by TRUTH bits in LUT Control x register (LUTCTRLx.TRUTH).

**Figure 37-2.** Truth Table Output Value Selection

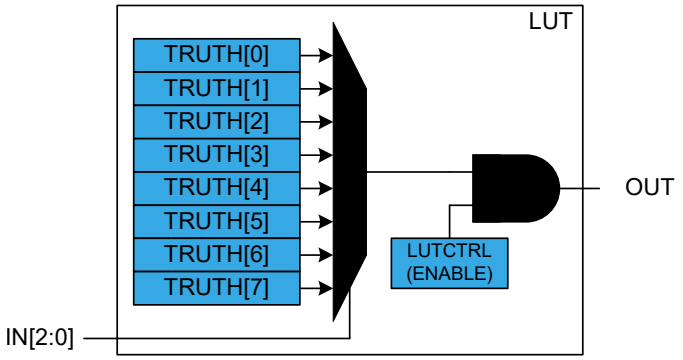



**Table 37-1.** Truth Table of LUT

| IN[2] | IN[1] | IN[0] | OUT |
|---|---|---|---|
| 0 | 0 | 0 | TRUTH[0] |
| 0 | 0 | 1 | TRUTH[1] |
| 0 | 1 | 0 | TRUTH[2] |
| 0 | 1 | 1 | TRUTH[3] |
| 1 | 0 | 0 | TRUTH[4] |
| 1 | 0 | 1 | TRUTH[5] |
| 1 | 1 | 0 | TRUTH[6] |
| 1 | 1 | 1 | TRUTH[7] |

### 37.6.2.4. Truth Table Inputs Selection

#### Input Overview

The inputs can be individually:

- Masked
- Driven by peripherals:
  - Analog comparator output (AC)
  - Timer/Counters waveform outputs (TC)
  - Serial Communication output transmit interface (SERCOM)
- Driven by internal events from Event System
- Driven by other CCL sub-modules

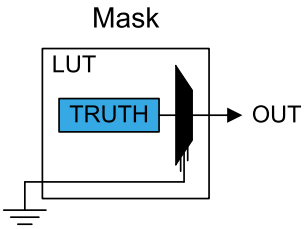

© 2026 Microchip Technology Inc. and its subsidiaries

The Input Selection for each input 'y' of LUT x is configured by writing the Input 'y' Source Selection bit in the LUT x Control register (LUTCTRLx.INSELy).

### Masked Inputs (MASK)

When a LUT input is masked (LUTCTRLx.INSELy = MASK), the corresponding TRUTH input (IN) is internally tied to zero, as shown in this figure:

**Figure 37-3.** Masked Input Selection




### Internal Feedback Inputs (FEEDBACK)

When selected (LUTCTRLx.INSELy = FEEDBACK), the Sequential (SEQ) output is used as input for the corresponding LUT.

The output from an internal sequential sub-module can be used as input source for the LUT, see figure below for an example for LUT0 and LUT1. The sequential selection for each LUT follows the formula:

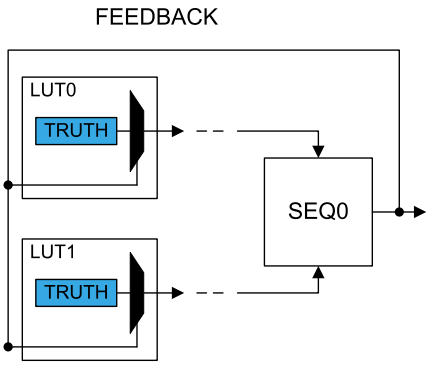
$IN[2N][i] = \mathrm{SEQ}[N]$

$IN[2N+1][i] = \mathrm{SEQ}[N]$

With *N* representing the sequencer number and *i*=0,1,2 representing the LUT input index.

For additional information, refer to Sequential Logic.

**Figure 37-4.** Feedback Input Selection

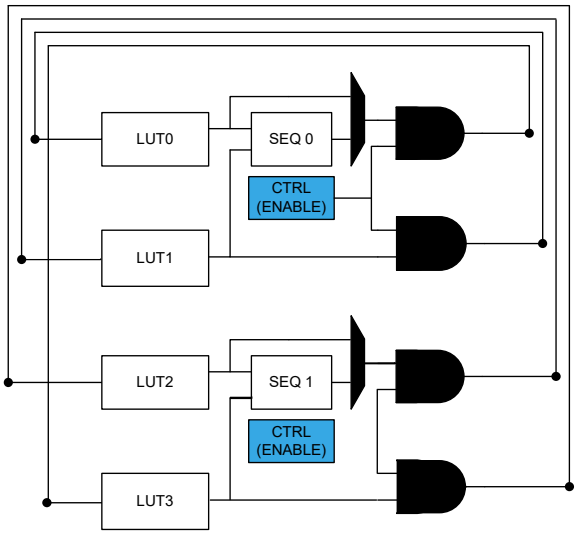



### Linked LUT (LINK)

When selected (LUTCTRLx.INSELy=LINK), the subsequent LUT output is used as the LUT input (for example, LUT2 is the input for LUT1), as shown in the figure below:


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 37-5.** Linked LUT Input Selection




### Internal Events Inputs Selection (EVENT)

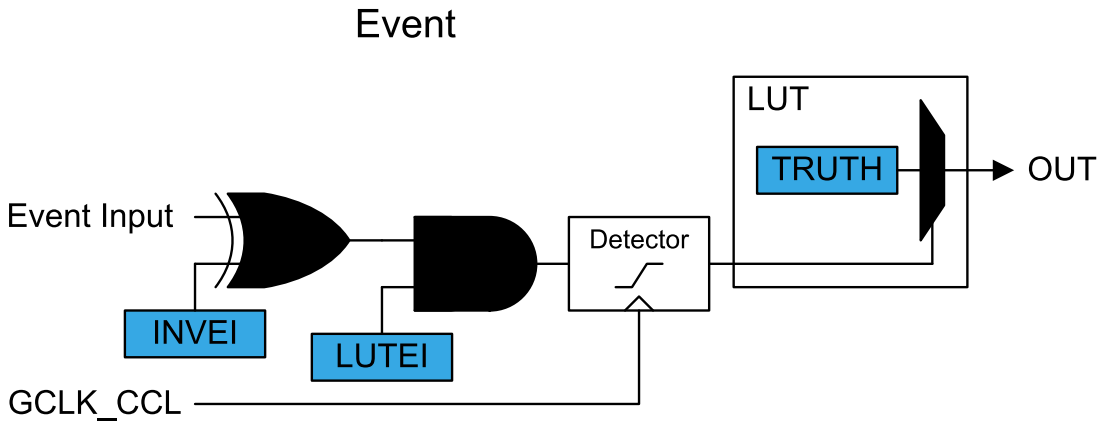
Asynchronous events from the Event System can be used as input selection, as shown in the following figure. For each LUT, one event input line is available and can be selected on each LUT input. Before enabling the event selection by writing LUTCTRLx.INSELy = EVENT, the Event System must be configured first.

By default, CCL includes an edge detector. When the event is received, an internal strobe is generated when a rising edge is detected. The pulse duration is one GCLK_CCL clock cycle. Writing the LUTCTRLx.INSELy = ASYNCEVENT will disable the edge detector (This feature is only available on SAM C20/C21 N variants). In this case, it is possible to combine an asynchronous event input with any other input source. This is typically useful with event levels inputs, for example external I/O pin events. The following steps ensure proper operation:

1. Enable the GCLK_CCL clock.
2. Configure the Event System to route the event asynchronously.
3. Select the event input type (LUTCTRLx.INSEL = ASYNCEVENT).
4. If a strobe must be generated on the event input falling edge, write a '1' to the Inverted Event Input Enable bit in the LUT Control register (LUTCTRLx.INVEI) .
5. Enable the event input by writing the Event Input Enable bit in the LUT Control register (LUTCTRLx.LUTEI) to '1'.


© 2026 Microchip Technology Inc. and its subsidiaries

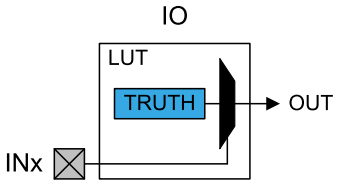
**Figure 37-6.** Event Input Selection




### I/O Pin Inputs (IO)

When the I/O pin is selected as LUT input (LUTCTRLx.INSELy = IO), the corresponding LUT input will be connected to the pin, as shown in the figure below.

**Figure 37-7.** I/O Pin Input Selection




### Analog Comparator Inputs (AC)

The AC outputs can be used as input source for the LUT (LUTCTRLx.INSELy=AC).

The analog comparator outputs are distributed following the formula:

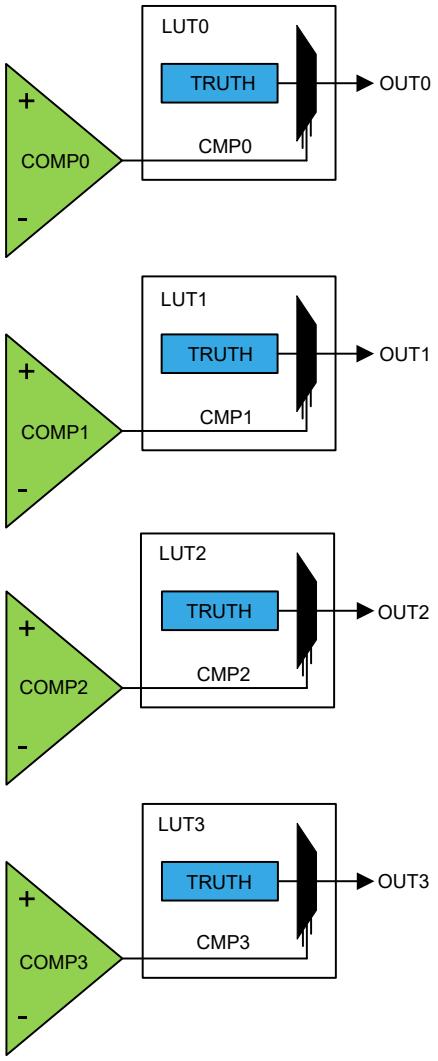
`IN[N][i]=AC[N % ComparatorOutput_Number]`

With *N* representing the LUT number and *i*=[0,1,2] representing the LUT input index.

Before selecting the comparator output, the AC must be configured first.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 37-8.** AC Input Selection




### Timer/Counter Inputs (TC)

The TC waveform output WO[0] can be used as input source for the LUT (LUTCTRLx.INSELy = TC). Only consecutive instances of the TC, that is, TCx and the subsequent TC(x+1), are available as default and alternative TC selections (for example, TC0 and TC1 are sources for LUT0, TC1 and TC2 are sources for LUT1). See the figure below for an example for LUT0. More general, the Timer/Counter selection for each LUT follows the formula:

$$\mathrm{IN}[N][i] = DefaultTC[N \,\%\, \mathrm{TC_Instance_Number}]$$

$$\mathrm{IN}[N][i] = AlternativeTC[(N+1) \,\%\, \mathrm{TC_Instance_Number}]$$

Where *N* represents the LUT number and *i* represents the LUT input index (i=0,1,2).

For SAM C20/C21 N variants only, it is also possible to enable a second alternative option (LUTCTRLx.INSEL= ALT2TC). This option is intended to relax the alternative pin function or PCB design constraints when the default or the alternative TC instances are used for other purposes. When enabled, the Timer/Counter selection for each LUT follows the formula:

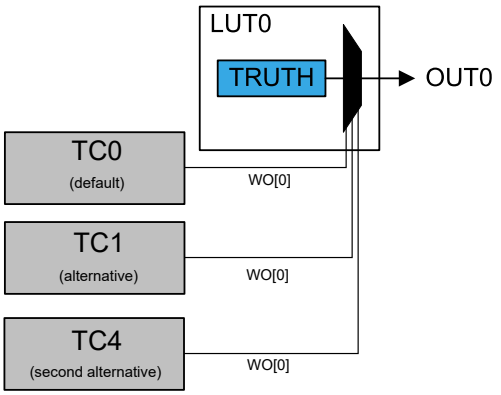
$$\mathrm{IN}[N][i] = SecondAlternativeTC[(N+4) \,\%\, \mathrm{TC_Instance_Number}]$$


© 2026 Microchip Technology Inc. and its subsidiaries

Note that for not implemented TC_Instance_Number, the corresponding input is tied to ground.

Before selecting the waveform outputs, the TC must be configured first.

**Figure 37-9.** TC Input Selection




### Timer/Counter for Control Application Inputs (TCC)

The TCC waveform outputs can be used as input source for the LUT. Only WO[2:0] outputs can be selected and routed to the respective LUT input (that is, IN0 is connected to WO0, IN1 to WO1, and IN2 to WO2), as shown in the figure below.

**Note:**
The TCC selection for each LUT follows the formula:

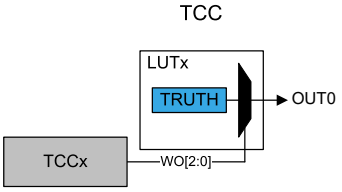
$\mathrm{IN}[N][i] = TCC[N\,\%\,\mathrm{TCC_Instance_Number}].\mathrm{WO}[\mathrm{i}]$

Where *N* represents the LUT number and i represents the LUT input index (i=0,1,2).

Before selecting the waveform outputs, the TCC must be configured first.

**Note:** TCC2 only outputs 2 WO signals, so TCC2.WO[0] is connected to both LUT2.IN[0] and LUT2.IN[2], and TCC2.WO[1] is connected to LUT2.IN[1].

**Figure 37-10.** TCC Input Selection




### Serial Communication Output Transmit Inputs (SERCOM)

The serial engine transmitter output from Serial Communication Interface (SERCOM TX, TXd for USART, MOSI for SPI) can be used as input source for the LUT. The figure below shows an example for LUT0 and LUT1. The SERCOM selection for each LUT follows the formula:

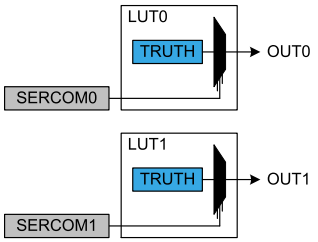
$\mathrm{IN}[N][i] = SERCOM[N\,\%\,\mathrm{SERCOM_Instance_Number}]$

With *N* representing the LUT number and *i*=0,1,2 representing the LUT input index.

Before selecting the SERCOM as input source, the SERCOM must be configured first: the SERCOM TX signal must be output on SERCOMn/pad[0], which serves as input pad to the CCL.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 37-11.** SERCOM Input Selection




**Related Links**

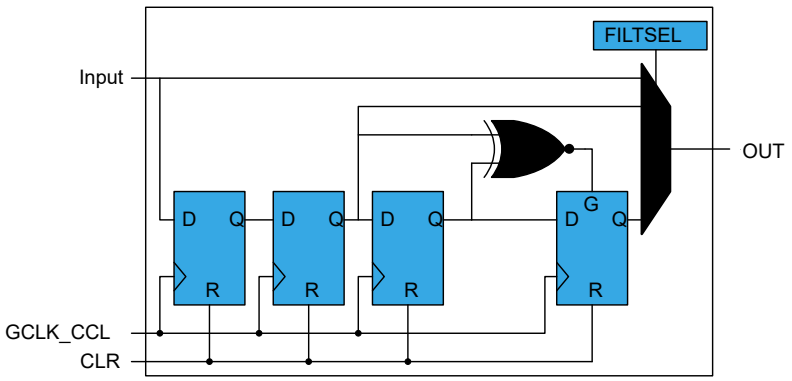
I/O Multiplexing and Considerations
PORT - I/O Pin Controller
GCLK - Generic Clock Controller
AC – Analog Comparators
Timer/Counter (TC)
Timer/Counter for Control Applications (TCC)
SERCOM – Serial Communication Interface
I/O Multiplexing and Considerations

### 37.6.2.5. Filter

By default, the LUT output is a combinatorial function of the LUT inputs. This may cause some short glitches when the inputs change value. These glitches can be removed by clocking through filters, if demanded by application needs.

The Filter Selection bits in LUT Control register (LUTCTRLx.FILTSEL) define the synchronizer or digital filter options. When a filter is enabled, the OUT output will be delayed by two to five GCLK cycles. One APB clock after the corresponding LUT is disabled, all internal filter logic is cleared.
**Note:** Events used as LUT input will also be filtered, if the filter is enabled.

**Figure 37-12.** Filter




### 37.6.2.6. Edge Detector

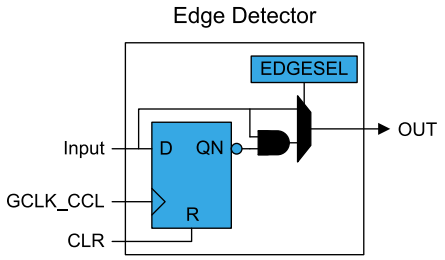
The edge detector can be used to generate a pulse when detecting a rising edge on its input. To detect a falling edge, the TRUTH table should be inverted.

The edge detector is enabled by writing '1' to the Edge Selection bit in LUT Control register (LUTCTRLx.EDGESEL). In order to avoid unpredictable behavior, either the filter or synchronizer must be enabled.

Edge detection is disabled by writing a '0' to LUTCTRLx.EDGESEL. After disabling a LUT, the corresponding internal Edge Detector logic is cleared one APB clock cycle later.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 37-13.** Edge Detector




### 37.6.2.7. Sequential Logic

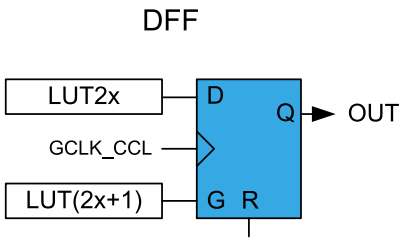
Each LUT pair can be connected to the internal sequential logic which can be configured to work as D flip flop, JK flip flop, gated D-latch or RS-latch by writing the Sequential Selection bits on the corresponding Sequential Control x register (SEQCTRLx.SEQSEL). Before using sequential logic, the GCLK_CCL clock and optionally each LUT filter or edge detector must be enabled.

**Note:** While configuring the sequential logic, the even LUT must be disabled. When configured the even LUT must be enabled.

#### Gated D Flip-Flop (DFF)

When the DFF is selected, the D-input is driven by the even LUT output (LUT0 and LUT2), and the G-input is driven by the odd LUT output (LUT1 and LUT3), as shown in Figure 37-14.

**Figure 37-14.** D Flip Flop




When the even LUT is disabled (LUTCTRL0.ENABLE=0 / LUTCTRL2.ENABLE=0), the flip-flop is asynchronously cleared. The reset command (R) is kept enabled for one APB clock cycle. In all other cases, the flip-flop output (OUT) is refreshed on rising edge of the GCLK_CCL, as shown in Table 37-2.

**Table 37-2.** DFF Characteristics

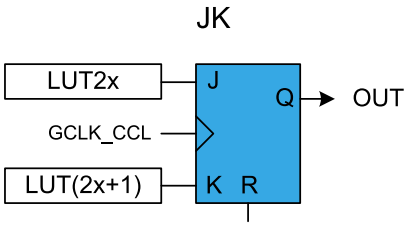
| R | G | D | OUT |
|---|---|---|---|
| 1 | X | X | Clear |
| 0 | 1 | 1 | Set |
| | | 0 | Clear |
| | 0 | X | Hold state (no change) |

#### JK Flip-Flop (JK)

When this configuration is selected, the J-input is driven by the even LUT output (LUT0 and LUT2), and the K-input is driven by the odd LUT output (LUT1 and LUT3), as shown in Figure 37-15.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 37-15.** JK Flip Flop




When the even LUT is disabled (LUTCTRL0.ENABLE=0 / LUTCTRL2.ENABLE=0), the flip-flop is asynchronously cleared. The reset command (R) is kept enabled for one APB clock cycle. In all other cases, the flip-flop output (OUT) is refreshed on rising edge of the GCLK_CCL, as shown in Table 37-3.

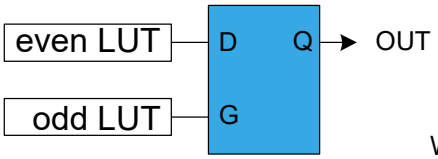
**Table 37-3.** JK Characteristics

| R | J | K | OUT |
|---|---|---|---|
| 1 | X | X | Clear |
| 0 | 0 | 0 | Hold state (no change) |
| 0 | 0 | 1 | Clear |
| 0 | 1 | 0 | Set |
| 0 | 1 | 1 | Toggle |

### Gated D-Latch (DLATCH)

When the DLATCH is selected, the D-input is driven by the even LUT output (LUT0 and LUT2), and the G-input is driven by the odd LUT output (LUT1 and LUT3), as shown in Figure 37-14.

**Figure 37-16.** D-Latch




When the even LUT is disabled (LUTCTRL0.ENABLE=0 / LUTCTRL2.ENABLE=0), the latch output will be cleared. The G-input is forced enabled for one more APB clock cycle, and the D-input to zero. In all other cases, the latch output (OUT) is refreshed as shown in Table 37-4.

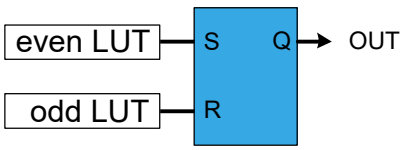
**Table 37-4.** D-Latch Characteristics

| G | D | OUT |
|---|---|---|
| 0 | X | Hold state (no change) |
| 1 | 0 | Clear |
| 1 | 1 | Set |

### RS Latch (RS)

When this configuration is selected, the S-input is driven by the even LUT output (LUT0 and LUT2), and the R-input is driven by the odd LUT output (LUT1 and LUT3), as shown in Figure 37-17.

**Figure 37-17.** RS-Latch





© 2026 Microchip Technology Inc. and its subsidiaries

When the even LUT is disabled LUTCTRL0.ENABLE=0 / LUTCTRL2.ENABLE=0), the latch output will be cleared. The R-input is forced enabled for one more APB clock cycle and S-input to zero. In all other cases, the latch output (OUT) is refreshed as shown in Table 37-5.

**Table 37-5.** RS-Latch Characteristics

| S | R | OUT |
|---|---|---|
| 0 | 0 | Hold state (no change) |
| 0 | 1 | Clear |
| 1 | 0 | Set |
| 1 | 1 | Forbidden state |

### 37.6.3. Events

The CCL can generate the following output events:

- LUTOUTn with n=0-3: Lookup Table Output Value

Writing a '1' to the LUT Control Event Output Enable bit (LUTCTRLx.LUTEO) enables the corresponding output event. Writing a '0' to this bit disables the corresponding output event.

The CCL can take the following actions on an input event:

- INSELx: The event is used as input for the TRUTH table. For additional information, refer to Events.

Writing a '1' to the LUT Control Event Input Enable bit (LUTCTRLx.LUTEI) enables the corresponding action on input event. Writing a '0' to this bit disables the corresponding action on input event.

**Related Links**

Event System (EVSYS)

### 37.6.4. Sleep Mode Operation

When using the GCLK_CCL internal clocking, writing the Run In Standby bit in the Control register (CTRL.RUNSTDBY) to '1' will allow GCLK_CCL to be enabled in Standby Sleep mode.

If CTRL.RUNSTDBY=0, the GCLK_CCL will be disabled in Standby Sleep mode. If the Filter, Edge Detector or Sequential logic are enabled, the LUT output will be forced to zero in STANDBY mode. In all other cases, the TRUTH table decoder will continue operation and the LUT output will be refreshed accordingly.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

## 37.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRL | 7:0 | | RUNSTDBY | | | | | ENABLE | SWRST |
| 0x01 ... 0x03 | Reserved | | | | | | | | | |
| 0x04 | SEQCTRL0 | 7:0 | | | | | SEQSEL[3:0] | | | |
| 0x05 | SEQCTRL1 | 7:0 | | | | | SEQSEL[3:0] | | | |
| 0x06 ... 0x07 | Reserved | | | | | | | | | |
| 0x08 | LUTCTRLn0 | 7:0 | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| | | 15:8 | INSEL1[3:0] | | | | INSEL0[3:0] | | | |
| | | 23:16 | | LUTEO | LUTEI | INVEI | INSEL2[3:0] | | | |
| | | 31:24 | TRUTH[7:0] | | | | | | | |
| 0x0C | LUTCTRLn1 | 7:0 | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| | | 15:8 | INSEL1[3:0] | | | | INSEL0[3:0] | | | |
| | | 23:16 | | LUTEO | LUTEI | INVEI | INSEL2[3:0] | | | |
| | | 31:24 | TRUTH[7:0] | | | | | | | |
| 0x10 | LUTCTRLn2 | 7:0 | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| | | 15:8 | INSEL1[3:0] | | | | INSEL0[3:0] | | | |
| | | 23:16 | | LUTEO | LUTEI | INVEI | INSEL2[3:0] | | | |
| | | 31:24 | TRUTH[7:0] | | | | | | | |
| 0x14 | LUTCTRLn3 | 7:0 | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| | | 15:8 | INSEL1[3:0] | | | | INSEL0[3:0] | | | |
| | | 23:16 | | LUTEO | LUTEI | INVEI | INSEL2[3:0] | | | |
| | | 31:24 | TRUTH[7:0] | | | | | | | |

## 37.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 37.8.1. Control

**Name:** CTRL
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection

**Note:** CTRL register (except the bits ENABLE & SWRST) is Enable Protected when CCL.CTRL.ENABLE = 1.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | | | | ENABLE | SWRST |
| Access | | R/W | | | | | R/W | W |
| Reset | | 0 | | | | | 0 | 0 |

**Bit 6 – RUNSTDBY** Run in Standby
This bit indicates if the GCLK_CCL clock must be kept running in standby mode. The setting is ignored for configurations where the generic clock is not required. For details refer to Sleep Mode Operation.

**Important:** This bit must be written before enabling the CCL.

| Value | Description |
|---|---|
| 0 | Generic clock is not required in standby sleep mode. |
| 1 | Generic clock is required in standby sleep mode. |

**Bit 1 – ENABLE** Enable

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the CCL to their initial state.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 37.8.2. Sequential Control x

**Name:** SEQCTRL
**Offset:** 0x04 + n*0x01 [n=0..1]
**Reset:** 0x00
**Property:** PAC Write-Protection

**Note:** SEQCTRL register is Enable Protected when CCL.CTRL.ENABLE = 1.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SEQSEL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – SEQSEL[3:0]** Sequential Selection
These bits select the sequential configuration:
Sequential Selection

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Sequential logic is disabled |
| 0x1 | DFF | D flip flop |
| 0x2 | JK | JK flip flop |
| 0x3 | LATCH | D latch |
| 0x4 | RS | RS latch |
| 0x5 - 0xF | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 37.8.3. LUT Control x

**Name:** LUTCTRLn
**Offset:** 0x08 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Enable-protected

**Note:** LUTCTRLn register is Enable Protected when CCL.LUTCTRLn.ENABLE = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TRUTH[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | LUTEO | LUTEI | INVEI | INSEL2[3:0] | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | INSEL1[3:0] | | | | INSEL0[3:0] | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| Access | R/W | | R/W | R/W | | | R/W | |
| Reset | 0 | | 0 | 0 | | | 0 | |

**Bits 31:24 – TRUTH[7:0]** Truth Table
These bits define the value of truth logic as a function of inputs IN[2:0].

**Bit 22 – LUTEO** LUT Event Output Enable

| Value | Description |
|---|---|
| 0 | LUT event output is disabled. |
| 1 | LUT event output is enabled. |

**Bit 21 – LUTEI** LUT Event Input Enable

| Value | Description |
|---|---|
| 0 | LUT incoming event is disabled. |
| 1 | LUT incoming event is enabled. |

**Bit 20 – INVEI** Inverted Event Input Enable

| Value | Description |
|---|---|
| 0 | Incoming event is not inverted. |
| 1 | Incoming event is inverted. |

**Bits 8:11, 12:15, 16:19 – INSELx** LUT Input x Source Selection
These bits select the LUT input x source:

| Value | Name | Description |
|---|---|---|
| 0x0 | MASK | Masked input |
| 0x1 | FEEDBACK | Feedback input source |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x2 | LINK | Linked LUT input source |
| 0x3 | EVENT | Event input source |
| 0x4 | IO | I/O pin input source |
| 0x5 | AC | AC input source: CMP[0] (LUT0) / CMP[1] (LUT1)/ CMP[2] (LUT2) / CMP[3] (LUT3) |
| 0x6 | TC | TC input source: TC0 (LUT0) / TC1 (LUT1)/ TC2 (LUT2) / TC3 (LUT3) |
| 0x7 | ALTTC | Alternative TC input source: TC1 (LUT0) / TC2 (LUT1) / TC3 (LUT2) / TC4 (LUT3) |
| 0x8 | TCC | TCC input source: TCC0 (LUT0) / TCC1 (LUT1) / TCC2 (LUT2) / TCC0 (LUT3) |
| 0x9 | SERCOM | SERCOM input source: SERCOM0 (LUT0) / SERCOM1 (LUT1)/ SERCOM2 (LUT2) / SERCOM3 (LUT3) |
| 0xA | ALT2TC | Second alternative TC input source: TC4 (LUT0) / TC5 (LUT1) / TC6 (LUT2) / TC7 (LUT3). Only available on SAM C20/C21 N variants. |
| 0xB | ASYNCEVENT | Asynchronous event input source. Only available on SAM C20/C21 N variants. |
| 0xC - 0xF | Reserved | Reserved |

**Bit 7 – EDGESEL** Edge Selection

| Value | Description |
|---|---|
| 0 | Edge detector is disabled. |
| 1 | Edge detector is enabled. |

**Bits 5:4 – FILTSEL[1:0]** Filter Selection

These bits select the LUT output filter options:
Filter Selection

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | Filter disabled |
| 0x1 | SYNCH | Synchronizer enabled |
| 0x2 | FILTER | Filter enabled |
| 0x3 | - | Reserved |

**Bit 1 – ENABLE** LUT Enable

| Value | Description |
|---|---|
| 0 | The LUT is disabled. |
| 1 | The LUT is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 38. ADC - Analog-to-Digital Converter

## 38.1. Overview

The Analog-to-Digital Converter (ADC) converts analog signals to digital values. The ADC has up to 12-bit resolution, and is capable of a sampling rate of up to 1 MSPS. The input selection is flexible, and both differential and single-ended measurements can be performed. In addition, several internal signal inputs are available. The ADC can provide both signed and unsigned results.

ADC measurements can be started by either application software or an incoming event from another peripheral in the device. ADC measurements can be started with predictable timing and without software intervention.

Both internal and external reference voltages can be used.

The INTREF voltage reference (supplied by the bandgap), as well as the scaled I/O and core voltages, can also be measured by the ADC.

The ADC has a compare function for accurate monitoring of user-defined thresholds, with minimum software intervention required.

The ADC can be configured for 8-bit, 10-bit or 12-bit results. ADC conversion results are provided left-adjusted or right-adjusted, which eases calculation when the result is represented as a signed value. It is possible to use DMA to move ADC results directly to memory or peripherals when conversions are done.

The SAM C20/C21 has two ADC instances, ADC0 and ADC1. The two inputs can be sampled simultaneously, as each ADC includes a dedicated sample and hold (S&H)circuit.

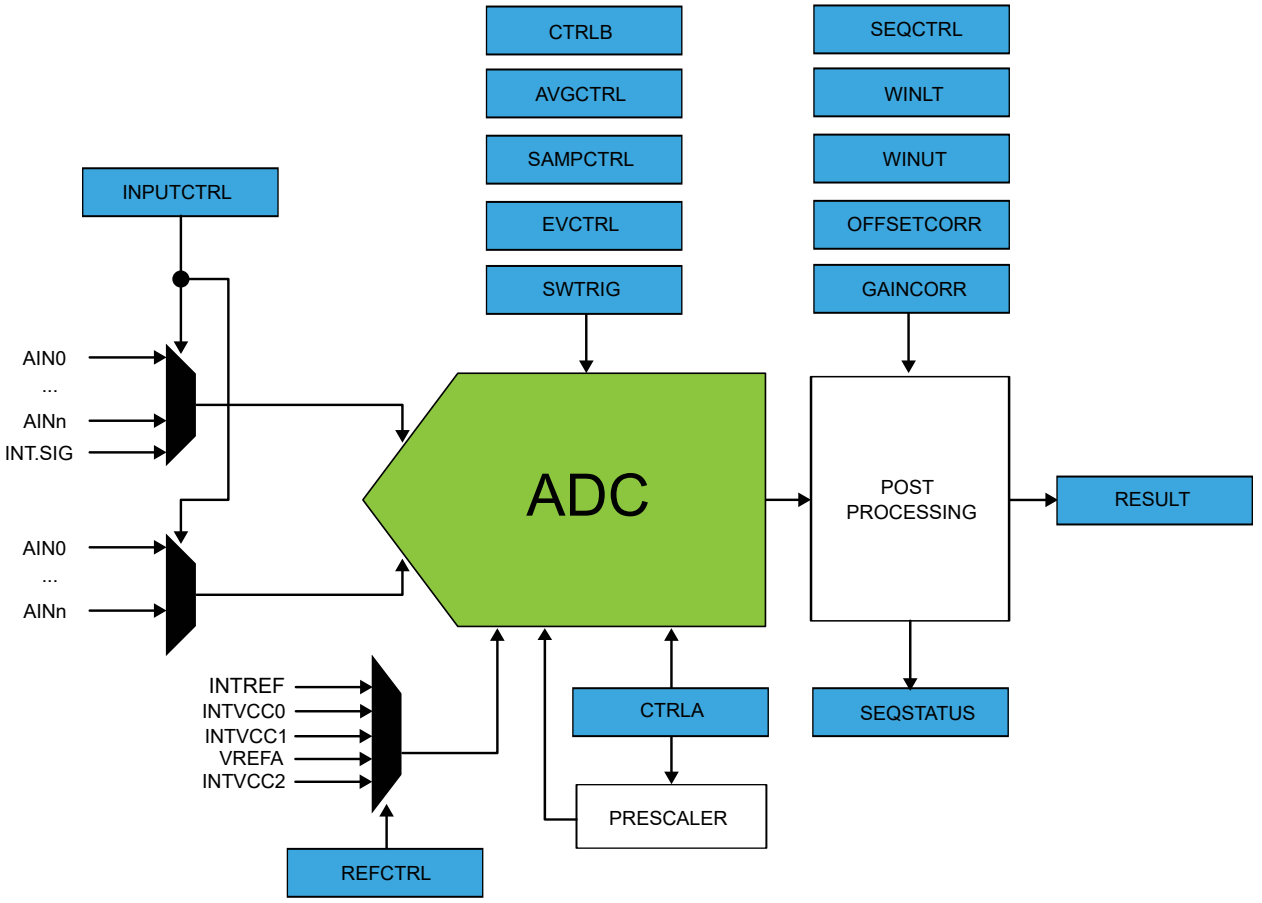
## 38.2. Features

- Two Analog-to-Digital Converters (ADC): ADC0 and ADC1
- 8-bit, 10-bit or 12-bit resolution
- Up to 1,000,000 samples per second (1 MSPS)
- Differential and single-ended inputs
  - Up to 12 analog inputs per ADC (20 unique channels total)
    8 positive and 7 negative (internal and external)
- Internal inputs:
  - INTREF voltage reference, supplied by the bandgap
  - Scaled core supply
  - Scaled I/O supply
- Single, continuous, and sequencing options
- Windowing monitor with selectable channel
- Conversion range: $V_{ref}$ = [2.0V to $VDD_{ANA}$ ]
- Built-in internal reference and external reference options
- Event-triggered conversion for accurate timing (one event input)
- Optional DMA transfer of conversion settings or result
- Hardware gain and offset compensation
- Averaging and oversampling with decimation to support up to 16-bit result
- Selectable sampling time
- Flexible Power or Throughput rate management


© 2026 Microchip Technology Inc. and its subsidiaries

## 38.3. Block Diagram

**Figure 38-1.** ADC Block Diagram




## 38.4. Signal Description

| Signal | Description | Type |
|---|---|---|
| VREFA | Analog input | External reference voltage |
| AIN[11..0] | Analog input | Analog input channels |

**Note:** One signal can be mapped on several pins.

**Related Links**
Configuration Summary
I/O Multiplexing and Considerations

## 38.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 38.5.1. I/O Lines

Using the ADC's I/O lines requires the I/O pins to be configured using the port configuration (PORT).

**Related Links**
PORT - I/O Pin Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.5.2. Power Management

The ADC will continue to operate in any sleep mode where the selected source clock is running. The ADC's interrupts except the OVERRUN interrupt, can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

**Related Links**
PM - Power Manager

### 38.5.3. Clocks

The ADC bus clocks (CLK_APB_ADCx) can be enabled in the Main Clock, which also defines the default state.

Each ADC requires a generic clock (GCLK_ADCx). This clock must be configured and enabled in the Generic Clock Controller (GCLK) before using the ADC.

A generic clock is asynchronous to the bus clock. Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to *Synchronization* for further details.

**Related Links**
Synchronization
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 38.5.4. DMA

The DMA request line is connected to the DMA Controller (DMAC). Using the ADC DMA requests requires the DMA Controller to be configured first.

**Related Links**
DMAC – Direct Memory Access Controller

### 38.5.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the ADC interrupt requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 38.5.6. Events

The events are connected to the Event System.

**Related Links**
Event System (EVSYS)

### 38.5.7. Debug Operation

When the CPU is halted in debug mode the ADC will halt normal operation. The ADC can be forced to continue operation during debugging. Refer to DBGCTRL register for details.

### 38.5.8. Register Access Protection

All registers with write-access are optionally write-protected by the Peripheral Access Controller (PAC), except the following register:

- Interrupt Flag Status and Clear (INTFLAG) register

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 38.5.9. Analog Connections

I/O-pins (AINx), as well as the VREFA reference voltage pin are analog inputs to the ADC. Any internal reference source, such as INTREF, supplied by the bandgap, or DAC must be configured and enabled prior to its use with the ADC.

### 38.5.10. Calibration

The BIAS and LINEARITY calibration values from the production test must be loaded from the NVM Software Calibration Area into the ADC Calibration register (CALIB) by software to achieve specified accuracy.

**Related Links**
NVM Software Calibration Area Mapping

## 38.6. Functional Description

### 38.6.1. Principle of Operation

By default, the ADC provides results with 12-bit resolution. 8-bit or 10-bit results can be selected in order to reduce the conversion time, see Conversion Timing and Sampling Rate.

The ADC has an oversampling with decimation option that can extend its resolution to 16 bits. The input values can be either internal or external (connected I/O pins). The user can also configure whether the conversion should be single-ended or differential.

### 38.6.2. Basic Operation

#### 38.6.2.1. Initialization

The following registers are enable-protected, meaning that they can only be written when the ADC is disabled (CTRLA.ENABLE=0):

- Control B (CTRLB) register
- Reference Control (REFCTRL) register
- Event Control (EVCTRL) register
- Calibration (CALIB) register

Enable-protection is denoted by the "Enable-Protected" property in the register description.

#### 38.6.2.2. Enabling, Disabling and Resetting

The ADC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The ADC is disabled by writing CTRLA.ENABLE=0.

The ADC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the ADC, except DBGCTRL, will be reset to their initial state, and the ADC will be disabled. Refer to CTRLA for details.

#### 38.6.2.3. Operation

In the most basic configuration, the ADC samples values from the configured internal or external sources (INPUTCTRL register). The rate of the conversion depends on the combination of the GCLK_ADCx frequency and the clock prescaler.

To convert analog values to digital values, the ADC needs to be initialized first, as described in the Initialization section. Data conversion can be started either manually by setting the Start bit in the Software Trigger register (SWTRIG.START=1), or automatically by configuring an automatic trigger


© 2026 Microchip Technology Inc. and its subsidiaries

to initiate the conversions. The ADC starts sampling the input only after the start of conversion is triggered. This means that even after the MUX selection is made, sample and hold (S&H) operation starts only on the conversion trigger. A free-running mode can be used (CTRLC.FREERUN = 1) to continuously convert an input channel. In that case, no trigger is required to start conversion, and subsequent conversions will automatically start when previous conversion ends.

The ADC starts sampling the input only after the start of a conversion is triggered. This means that even after the MUX selection is made, sample and hold operation starts only on the conversion trigger.

The result of the conversion is stored in the Result register (RESULT) overwriting the result from the previous conversion.

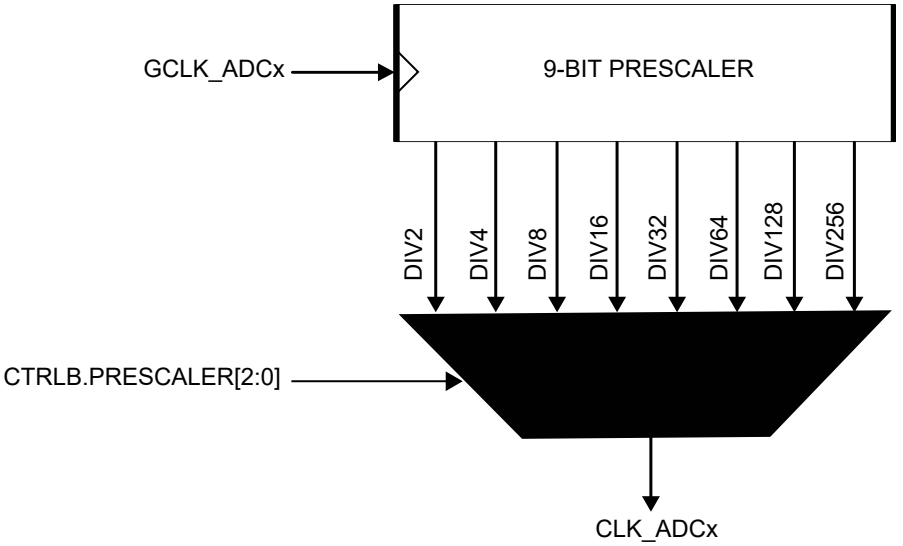
To avoid data loss, if more than one channel is enabled, the conversion result must be read as soon as it is available (INTFLAG.RESRDY). Failing to do so will result in an overrun error condition, indicated by the OVERRUN bit in the Interrupt Flag Status and Clear register (INTFLAG.OVERRUN).

To enable one of the available interrupts sources, the corresponding bit in the Interrupt Enable Set register (INTENSET) must be written to '1'.

### 38.6.2.4. Prescaler Selection

The ADC is clocked by GCLK_ADCx. There is also a prescaler in the ADC to enable conversion at lower clock rates. Refer to CTRLB for details on prescaler settings. Refer to Conversion Timing and Sampling Rate for details on timing and sampling rate.

**Figure 38-2.** ADC Prescaler




**Note:** The minimum prescaling factor is DIV2.

### 38.6.2.5. Reference Configuration

The ADC has various sources for its reference voltage $V_{REF}$. The Reference Voltage Selection bit field in the Reference Control register (REFCTRL.REFSEL) determines which reference is selected. By default, the internal voltage reference INTREF, supplied by the bandgap, is selected. Based on customer application requirements, the external or internal reference can be selected. Refer to REFCTRL.REFSEL for further details on available selections.

**Related Links**

REFCTRL

Reference Control

Analog-to-Digital Converter (ADC) Characteristics


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.6.2.6. ADC Resolution

The ADC supports 8-bit, 10-bit or 12-bit resolution. Resolution can be changed by writing the Resolution bit group in the Control C register (CTRLC.RESSEL). By default, the ADC resolution is set to 12 bits. The resolution affects the propagation delay, see also Conversion Timing and Sampling Rate.

### 38.6.2.7. Differential and Single-Ended Conversions

The ADC has two conversion options: differential and single-ended:

If the positive input is always positive, the single-ended conversion should be used in order to have full 12-bit resolution in the conversion.

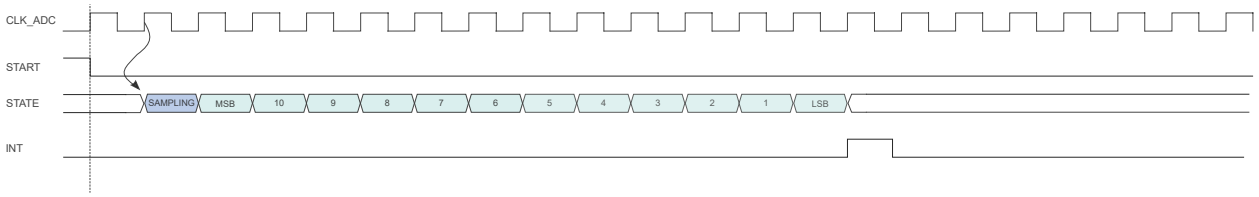
If the positive input may go below the negative input, the differential mode should be used in order to get correct results.

The differential mode is enabled by setting DIFFMODE bit in the Control C register (CTRLC.DIFFMODE). Both conversion types could be run in single mode or in free-running mode. When the free-running mode is selected, an ADC input will continuously sample the input and perform a new conversion. The INTFLAG.RESRDY bit will be set at the end of each conversion.

### 38.6.2.8. Conversion Timing and Sampling Rate

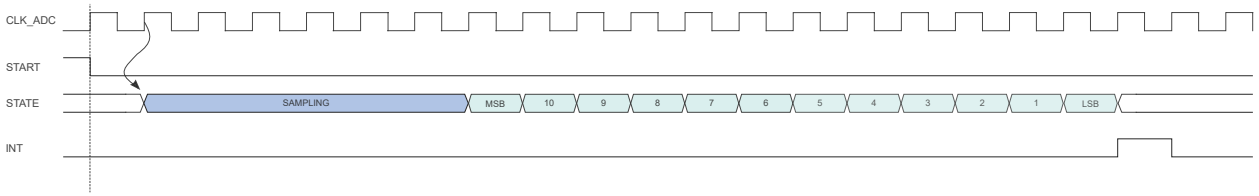
The following figure shows the ADC timing for a single conversion. A conversion starts after the software or event start are synchronized with the GCLK_ADCx clock. The input channel is sampled in the first half CLK_ADCx period.

**Figure 38-3.** ADC Timing for One Conversion in 12-bit Resolution




The sampling time can be increased by using the Sampling Time Length bit group in the Sampling Time Control register (SAMPCTRL.SAMPLEN). As example, the next figure is showing the timing conversion with sampling time increased to six CLK_ADC cycles.

**Figure 38-4.** ADC Timing for One Conversion with Increased Sampling Time, 12-bit




The ADC can also provide offset compensation, as shown in the following figure. The offset compensation is enabled by the Offset Compensation bit in the Sampling Control register (SAMPCTRL.OFFCOMP).

**Note:** ADC sampling time is fixed to 4 ADC Clock cycles when offset compensation (OFFCOMP =1) is used.

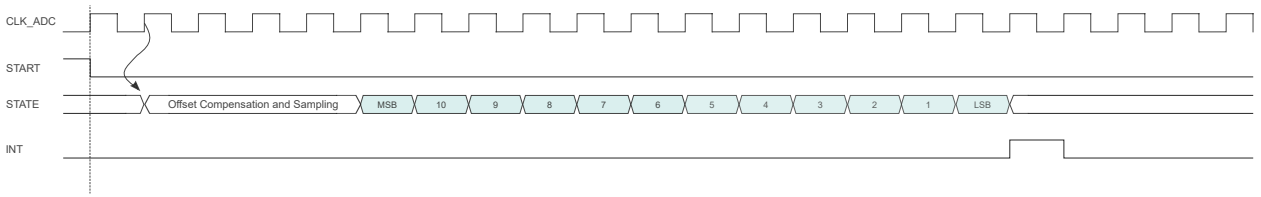
In free running mode, the sampling rate $R_S$ is calculated by

$R_S = f_{CLK_ADC} / ( n_{SAMPLING} + n_{OFFCOMP} + n_{DATA})$

Here, $n_{SAMPLING}$ is the sampling duration in CLK_ADC cycles, $n_{OFFCOMP}$ is the offset compensation duration in clock cycles, and $n_{DATA}$ is the bit resolution. $f_{CLK_ADC}$ is the ADC clock frequency from the internal prescaler: $f_{CLK_ADC} = f_{GCLK_ADC} / 2^{(1 + CTRLB.PRESCALER)}$

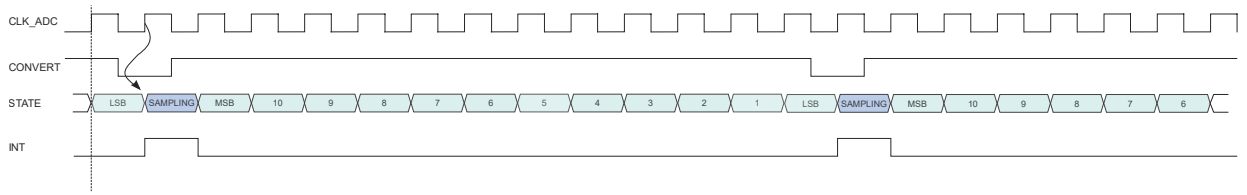

© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 38-5.** ADC Timing for One Conversion with Offset Compensation, 12-bit




The impact of resolution on the sampling rate is seen in the next two figures, where free-running sampling in 12-bit and 8-bit resolution are compared.

**Figure 38-6.** ADC Timing for Free Running in 12-bit Resolution




In free running mode, the sampling rate $R_S$ is calculated by:

$R_S = f_{CLK_ADC} / ( n_{SAMPLING} + n_{OFFCOMP} + n_{DATA})$

Where,

$n_{SAMPLING}$ is the sampling duration in CLK_ADC cycles, $n_{OFFCOMP}$ is the offset compensation duration in clock cycles, and $n_{DATA}$ is the bit resolution. $f_{CLK_ADC}$ is the ADC clock frequency from the internal prescaler: $f_{CLK_ADC} = f_{GCLK_ADC} / 2^{(1 + CTRLA.PRESCALER)}$.

The propagation delay of an ADC measurement is given by:

$$\text{PropagationDelay} = \frac{1 + \text{Resolution}}{f_{\text{ADC}}}$$

> **Example.** In order to obtain 1 MSPS in 12-bit resolution with a sampling time length of four CLK_ADC cycles, $f_{CLK_ADC}$ must be 1 Msps * (4 + 12) = 16 MHz. As the minimal division factor of the prescaler is 2, GCLK_ADC must be 32 MHz.

### 38.6.2.9. Accumulation

The result from multiple consecutive conversions can be accumulated. The number of samples to be accumulated is specified by the Sample Number field in the Average Control register (AVGCTRL.SAMPLENUM). When accumulating more than 16 samples, the result will be too large to match the 16-bit RESULT register size. To avoid overflow, the result is right shifted automatically to fit within the available register size. The number of automatic right shifts is specified in the table below.

**Note:** To perform the accumulation of two or more samples, the Conversion Result Resolution field in the Control C register (CTRLC.RESSEL) must be set.

**Table 38-1.** Accumulation

| Number of Accumulated Samples | AVGCTRL. SAMPLENUM | Number of Automatic Right Shifts | Final Result Precision | Automatic Division Factor |
|---|---|---|---|---|
| 1 | 0x0 | 0 | 12 bits | 0 |
| 2 | 0x1 | 0 | 13 bits | 0 |
| 4 | 0x2 | 0 | 14 bits | 0 |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 38-1.** Accumulation (continued)

| Number of Accumulated Samples | AVGCTRL. SAMPLENUM | Number of Automatic Right Shifts | Final Result Precision | Automatic Division Factor |
|---|---|---|---|---|
| 8 | 0x3 | 0 | 15 bits | 0 |
| 16 | 0x4 | 0 | 16 bits | 0 |
| 32 | 0x5 | 1 | 16 bits | 2 |
| 64 | 0x6 | 2 | 16 bits | 4 |
| 128 | 0x7 | 3 | 16 bits | 8 |
| 256 | 0x8 | 4 | 16 bits | 16 |
| 512 | 0x9 | 5 | 16 bits | 32 |
| 1024 | 0xA | 6 | 16 bits | 64 |
| Reserved | 0xB –0xF | | 12 bits | 0 |

### 38.6.2.10. Averaging

Averaging is a feature that increases the sample accuracy, at the cost of a reduced sampling rate. This feature is suitable when operating in noisy conditions.

Averaging is done by accumulating m samples, as described in Accumulation, then dividing the result by m. The averaged result is available in the RESULT register. The number of samples to be accumulated is specified by writing to AVGCTRL.SAMPLENUM as shown in Table 38-2.

The division is obtained by a combination of the automatic right shift described above, and an additional right shift that must be specified by writing to the Adjusting Result/Division Coefficient field in AVGCTRL (AVGCTRL.ADJRES), as described in Table 38-2.

**Note:** To perform the averaging of two or more samples, the Conversion Result Resolution field in the Control C register (CTRLC.RESSEL) must be set.

Averaging AVGCTRL.SAMPLENUM samples will reduce the un-averaged sampling rate by a factor $\frac{1}{\text{AVGCTRL.SAMPLENUM}}$.

When the averaged result is available, the INTFLAG.RESRDY bit will be set.

**Table 38-2.** Averaging

| Number of Accumulated Samples | AVGCTRL. SAMPLENUM | Intermediate Result Precision | Number of Automatic Right Shifts | Division Factor | AVGCTRL.ADJRES | Total Number of Right Shifts | Final Result Precision | Automatic Division Factor |
|---|---|---|---|---|---|---|---|---|
| 1 | 0x0 | 12 bits | 0 | 1 | 0x0 | | 12 bits | 0 |
| 2 | 0x1 | 13 | 0 | 2 | 0x1 | 1 | 12 bits | 0 |
| 4 | 0x2 | 14 | 0 | 4 | 0x2 | 2 | 12 bits | 0 |
| 8 | 0x3 | 15 | 0 | 8 | 0x3 | 3 | 12 bits | 0 |
| 16 | 0x4 | 16 | 0 | 16 | 0x4 | 4 | 12 bits | 0 |
| 32 | 0x5 | 17 | 1 | 16 | 0x4 | 5 | 12 bits | 2 |
| 64 | 0x6 | 18 | 2 | 16 | 0x4 | 6 | 12 bits | 4 |
| 128 | 0x7 | 19 | 3 | 16 | 0x4 | 7 | 12 bits | 8 |
| 256 | 0x8 | 20 | 4 | 16 | 0x4 | 8 | 12 bits | 16 |
| 512 | 0x9 | 21 | 5 | 16 | 0x4 | 9 | 12 bits | 32 |
| 1024 | 0xA | 22 | 6 | 16 | 0x4 | 10 | 12 bits | 64 |
| Reserved | 0xB –0xF | | | | 0x0 | | 12 bits | 0 |

### 38.6.2.11. Oversampling and Decimation

By using oversampling and decimation, the ADC resolution can be increased from 12 bits up to 16 bits, for the cost of reduced effective sampling rate.


© 2026 Microchip Technology Inc. and its subsidiaries

To increase the resolution by n bits, $4^n$ samples must be accumulated. The result must then be right-shifted by n bits. This right-shift is a combination of the automatic right-shift and the value written to AVGCTRL.ADJRES. To obtain the correct resolution, the ADJRES must be configured as described in the table below. This method will result in n bit extra LSB resolution.

**Table 38-3.** Configuration Required for Oversampling and Decimation

| Result Resolution | Number of Samples to Average | AVGCTRL.SAMPLENUM[3:0] | Number of Automatic Right Shifts | AVGCTRL.ADJRES[2:0] |
|---|---|---|---|---|
| 13 bits | $4^1 = 4$ | 0x2 | 0 | 0x1 |
| 14 bits | $4^2 = 16$ | 0x4 | 0 | 0x2 |
| 15 bits | $4^3 = 64$ | 0x6 | 2 | 0x1 |
| 16 bits | $4^4 = 256$ | 0x8 | 4 | 0x0 |

### 38.6.2.12. Automatic Sequences

The ADC has the ability to automatically sequence a series of conversions. This means that each time the ADC receives a start-of-conversion request, it can perform multiple conversions automatically. All of the positive inputs can be included in a sequence by writing to corresponding bits in the Sequence Control register (SEQCTRL). The order of the conversion in a sequence is the lower positive MUX selection to upper positive MUX (AIN0, AIN1, AIN2 ...). In differential mode, the negative inputs selected by MUXNEG field, will be used for the entire sequence.

When a sequence starts, the Sequence Busy status bit in Sequence Status register (SEQSTATUS.SEQBUSY) will be set. When the sequence is complete, the Sequence Busy status bit will be cleared.

Each time a conversion is completed, the Sequence State bit in Sequence Status register (SEQSTATUS.SEQSTATE) will store the input number from which the conversion is done. The result will be stored in the RESULT register, and the Result Ready Interrupt Flag (INTFLAG.RESRDY) is set.

If additional inputs must be scanned, the ADC will automatically start a new conversion on the next input present in the sequence list.

Note that if SEQCTRL register has no bits set to '1', the conversion is done with the selected MUXPOS input.

### 38.6.2.13. Window Monitor

The window monitor feature allows the conversion result in the RESULT register to be compared to predefined threshold values. The window mode is selected by setting the Window Monitor Mode bits in the Control C register (CTRLC.WINMODE). Threshold values must be written in the Window Monitor Lower Threshold register (WINLT) and Window Monitor Upper Threshold register (WINUT).

If differential input is selected, the WINLT and WINUT are evaluated as signed values. Otherwise they are evaluated as unsigned values. The significant WINLT and WINUT bits are given by the precision selected in the Conversion Result Resolution bit group in the Control C register (CTRLC.RESSEL). As an example in 8-bit mode, only the eight lower bits will be considered. In addition, in differential mode, the eighth bit will be considered as the sign bit, even if the ninth bit is zero.

The INTFLAG.WINMON interrupt flag will be set if the conversion result matches the window monitor condition.

### 38.6.2.14. Offset and Gain Correction

Inherent gain and offset errors affect the absolute accuracy of the ADC.

The offset error is defined as the deviation of the actual ADC transfer function from an ideal straight line at zero input voltage. The offset error cancellation is handled by the Offset Correction register (OFFSETCORR). The offset correction value is subtracted from the converted data before writing the Result register (RESULT).


© 2026 Microchip Technology Inc. and its subsidiaries

The gain error is defined as the deviation of the last output step's midpoint from the ideal straight line, after compensating for offset error. The gain error cancellation is handled by the Gain Correction register (GAINCORR).

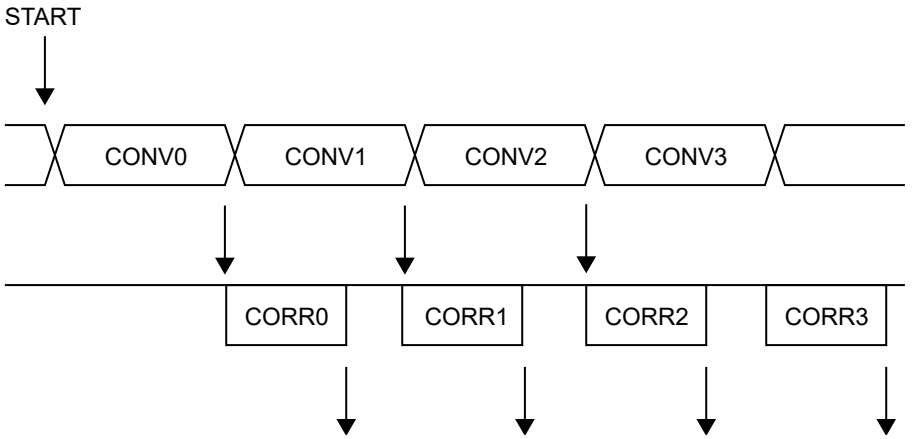
To correct these two errors, the Digital Correction Logic Enabled bit in the Control C register (CTRLC.CORREN) must be set.

Offset and gain error compensation results are both calculated according to:

$$\text{Result} = (\text{Conversion value} + - \text{OFFSETCORR}) \cdot \text{GAINCORR}$$

The correction will introduce a latency of 13 CLK_ADC clock cycles. In free running mode this latency is introduced on the first conversion only, since its duration is always less than the propagation delay. In single conversion mode this latency is introduced for each conversion.

**Figure 38-7.** ADC Timing Correction Enabled




#### 38.6.2.15. Reference Buffer Compensation Offset

A hardware compensation using a reference buffer can be used. When the REFCTRL.REFCOMP bit is set, the offset of the reference buffer is sensed during the ADC sampling phase. This offset will be then canceled during the conversion phase. This feature allows for the decrease of the overall gain error of the ADC.

There is a digital gain correction (refer to Offset and Gain Correction) but contrary to that digital gain correction, the hardware compensation will not introduce any latency.

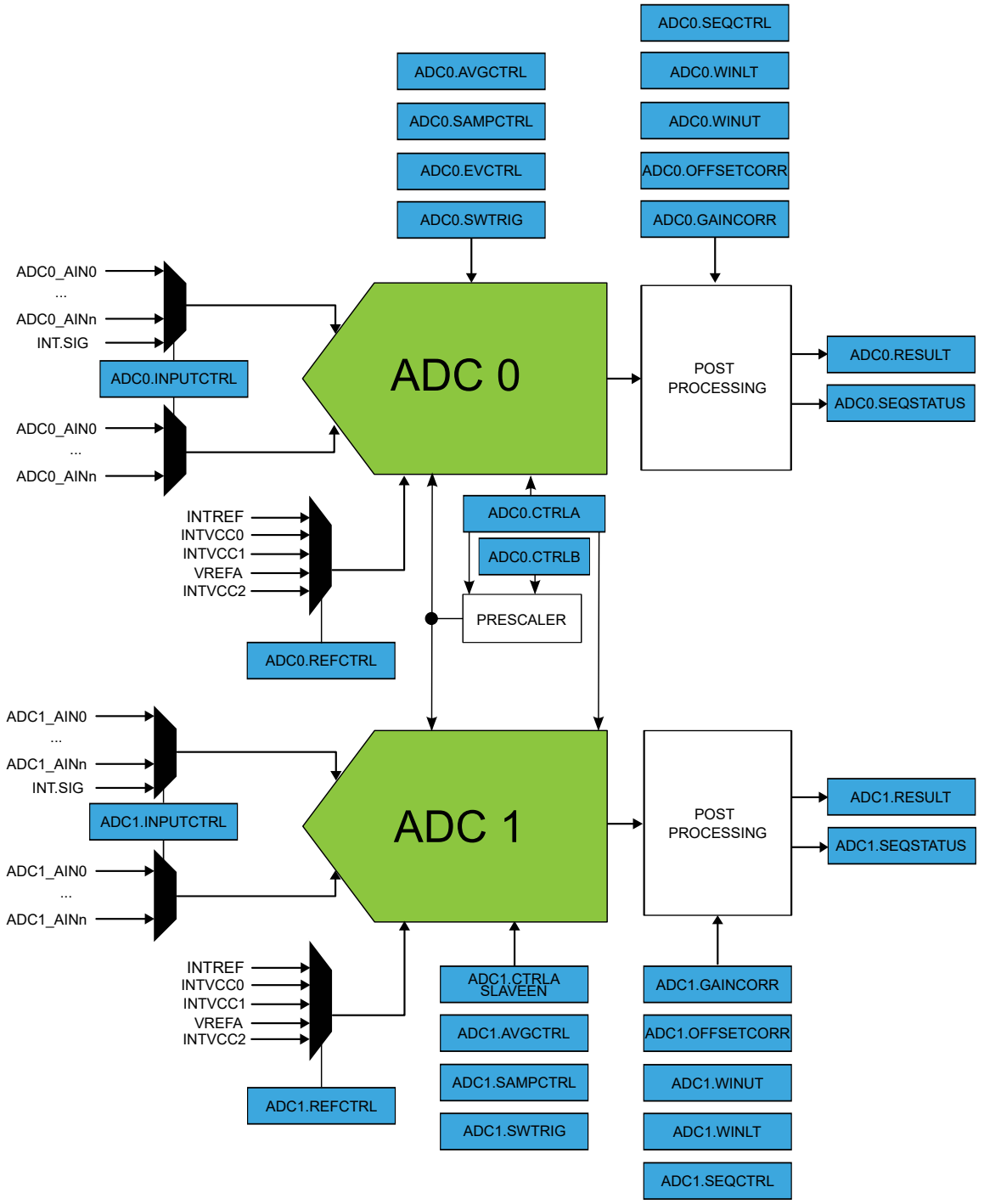
### 38.6.3. Additional Features

#### 38.6.3.1. Host - Client Operation

The Host - Client operation is available only on devices with two ADC instances. The ADC1 will be enabled as a Client of ADC0 instance when writing a one to the Client Enable bit in Control A register of the ADC1 instance (ADC1.CTRLA.ClientEN). When enabled, GCLK_ADC0 clock and ADC0 controls are internally routed to the ADC1 instance.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 38-8.** ADC Host - Client Block Diagram




In this mode of operation, the Client ADC is enabled by accessing the CTRLA register of Host ADC. In the same way, the Host ADC event inputs will be automatically routed to the Client ADC, meaning that the input events configuration must be done in the Host ADC (ADC0.EVCTRL).

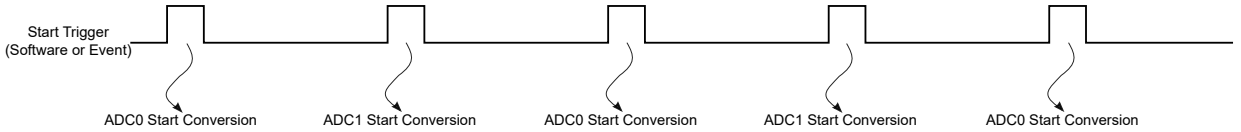
ADC measurements can be started simultaneously on both ADC's or interleaved. The trigger mode selection is available in the Host ADC Control C register (ADC0.CTRLC.DUALSEL).


© 2026 Microchip Technology Inc. and its subsidiaries

To restart an interleaved sequence, the user can apply different options:

- Flush the Host ADC (ADC0.SWTRIG.FLUSH = 1)
- Disable/re-enable the Host ADC (ADC0.CTRLA.ENABLE)
- Reset and reconfigure Host ADC (ADC0.CTRLA.SWRST = 1)

**Figure 38-9.** Interleaved Dual-Mode Trigger Selection




#### 38.6.3.2. Rail-to-Rail Operation

The accuracy of the ADC is highest when the input common mode voltage ($V_{CMIN}$) is close to $V_{REF}/2$. To enable a full range of common mode voltages (rail-to-rail operation), the Rail-to-Rail bit in the Control C register (CTRLC.R2R) should be written to one. Rail-to-rail operation requires a sampling period of four cycles. This is achieved by enabling offset compensation (SAMPCTRL.OFFCOMP = 1). Rail-to-rail operation should not be used when offset compensation is disabled.

#### 38.6.3.3. Double Buffering

The following registers are double buffered:

- Input Control (INPUTCTRL)
- Control C (CTRLC)
- Average Control (AVGCTRL)
- Sampling Time Control (SAMPCTRL)
- Window Monitor Lower Threshold (WINLT)
- Window Monitor Upper Threshold (WINUT)
- Gain Correction (GAINCORR)
- Offset Correction (OFFSETCORR)

When one of these registers is written, the data is stored in the corresponding buffer as long as the current conversion is not impacted, and the corresponding busy status will be set in the Synchronization Busy register (SYNCBUSY). When a new RESULT is available, data stored in the buffer registers will be transfered to the ADC and a new conversion can start.

### 38.6.4. DMA Operation

The ADC generates the following DMA request:

- Result Conversion Ready (RESRDY): the request is set when a conversion result is available and cleared when the RESULT register is read. When the averaging operation is enabled, the DMA request is set when the averaging is completed and result is available.

### 38.6.5. Interrupts

The ADC has the following interrupt sources:

- Result Conversion Ready: RESRDY
- Window Monitor: WINMON
- Overrun: OVERRUN

These interrupts, except the OVERRUN interrupt, are asynchronous wake-up sources. See *Sleep Mode Controller* for details.


© 2026 Microchip Technology Inc. and its subsidiaries

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the ADC is reset. See INTFLAG register for details on how to clear interrupt flags. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. Refer to *Nested Vector Interrupt Controller* for details. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

Nested Vector Interrupt Controller

Sleep Mode Controller

INTENCLR

Interrupt Enable Clear

INTENSET

Interrupt Enable Set

INTFLAG

Interrupt Flag Status and Clear

### 38.6.6. Events

The ADC can generate the following output events:

- Result Ready (RESRDY): Generated when the conversion is complete and the result is available. Refer to EVCTRL for details.
- Window Monitor (WINMON): Generated when the window monitor condition match. Refer to CTRLC for details.

Setting an Event Output bit in the Event Control Register (EVCTRL.xxEO=1) enables the corresponding output event. Clearing this bit disables the corresponding output event. Refer to the Event System chapter for details on configuring the event system.

The ADC can take the following actions on an input event:

- Start conversion (START): Start a conversion. Refer to SWTRIG for details.
- Conversion flush (FLUSH): Flush the conversion. Refer to SWTRIG for details.

Setting an Event Input bit in the Event Control register (EVCTRL.xxEI=1) enables the corresponding action on input event. Clearing this bit disables the corresponding action on input event.

The ADC uses only asynchronous events, so the asynchronous Event System channel path must be configured. By default, the ADC will detect a rising edge on the incoming event. If the ADC action must be performed on the falling edge of the incoming event, the event line must be inverted first. This is done by setting the corresponding Event Invert Enable bit in Event Control register (EVCTRL.xINV=1).

**Note:** If several events are connected to the ADC, the enabled action will be taken on any of the incoming events. If FLUSH and START events are available at the same time, the FLUSH event has priority.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

### 38.6.7. Sleep Mode Operation

The ONDEMAND and RUNSTDBY bits in the Control A register (CTRLA) control the behavior of the ADC during standby sleep mode, in cases where the ADC is enabled (CTRLA.ENABLE = 1). For further details on available options, refer to Table 38-4.

**Note:** When CTRLA.ONDEMAND=1, the analog block is powered-off when the conversion is complete. When a start request is detected, the system returns from sleep and starts a new conversion after the start-up time delay.

**Table 38-4.** ADC Sleep Behavior

| CTRLA.RUNSTDBY | CTRLA.ONDEMAND | CTRLA.ENABLE | Description |
|---|---|---|---|
| x | x | 0 | Disabled |
| 0 | 0 | 1 | Run in all sleep modes except STANDBY. |
| 0 | 1 | 1 | Run in all sleep modes on request, except STANDBY. |
| 1 | 0 | 1 | Run in all sleep modes. |
| 1 | 1 | 1 | Run in all sleep modes on request. |

### 38.6.8. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in Control A register (CTRLA.SWRST)
- Enable bit in Control A register (CTRLA.ENABLE)

The following registers are synchronized when written:

- Input Control register (INPUTCTRL)
- Control C register (CTRLC)
- Average control register (AVGCTRL)
- Sampling time control register (SAMPCTRL)
- Window Monitor Lower Threshold register (WINLT)
- Window Monitor Upper Threshold register (WINUT)
- Gain correction register (GAINCORR)
- Offset Correction register (OFFSETCORR)
- Software Trigger register (SWTRIG)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

**Related Links**
Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 38.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ONDEMAND | RUNSTDBY | SLAVEEN | | | | ENABLE | SWRST |
| 0x01 | CTRLB | 7:0 | | | | | | PRESCALER[2:0] | | |
| 0x02 | REFCTRL | 7:0 | REFCOMP | | | | REFSEL[3:0] | | | |
| 0x03 | EVCTRL | 7:0 | | | WINMONEO | RESRDYEO | STARTINV | FLUSHINV | STARTEI | FLUSHEI |
| 0x04 | INTENCLR | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x05 | INTENSET | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x06 | INTFLAG | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x07 | SEQSTATUS | 7:0 | SEQBUSY | | | SEQSTATE[4:0] | | | | |
| 0x08 | INPUTCTRL | 7:0 | | | | MUXPOS[4:0] | | | | |
| | | 15:8 | | | | MUXNEG[4:0] | | | | |
| 0x0A | CTRLC | 7:0 | R2R | | RESSEL[1:0] | | CORREN | FREERUN | LEFTADJ | DIFFMODE |
| | | 15:8 | | | DUALSEL[1:0] | | | WINMODE[2:0] | | |
| 0x0C | AVGCTRL | 7:0 | | | ADJRES[2:0] | | SAMPLENUM[3:0] | | | |
| 0x0D | SAMPCTRL | 7:0 | OFFCOMP | | SAMPLEN[5:0] | | | | | |
| 0x0E | WINLT | 7:0 | WINLT[7:0] | | | | | | | |
| | | 15:8 | WINLT[15:8] | | | | | | | |
| 0x10 | WINUT | 7:0 | WINUT[7:0] | | | | | | | |
| | | 15:8 | WINUT[15:8] | | | | | | | |
| 0x12 | GAINCORR | 7:0 | GAINCORR[7:0] | | | | | | | |
| | | 15:8 | | | | | GAINCORR[11:8] | | | |
| 0x14 | OFFSETCORR | 7:0 | OFFSETCORR[7:0] | | | | | | | |
| | | 15:8 | | | | | OFFSETCORR[11:8] | | | |
| 0x16 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | SWTRIG | 7:0 | | | | | | | START | FLUSH |
| 0x19 ... 0x1B | Reserved | | | | | | | | | |
| 0x1C | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x1D ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | SYNCBUSY | 7:0 | WINUT | WINLT | SAMPCTRL | AVGCTRL | CTRLC | INPUTCTRL | ENABLE | SWRST |
| | | 15:8 | | | | | | SWTRIG | OFFSETCORR | GAINCORR |
| 0x22 ... 0x23 | Reserved | | | | | | | | | |
| 0x24 | RESULT | 7:0 | RESULT[7:0] | | | | | | | |
| | | 15:8 | RESULT[15:8] | | | | | | | |
| 0x26 ... 0x27 | Reserved | | | | | | | | | |
| 0x28 | SEQCTRL | 7:0 | SEQENn[7:0] | | | | | | | |
| | | 15:8 | SEQENn[15:8] | | | | | | | |
| | | 23:16 | SEQENn[23:16] | | | | | | | |
| | | 31:24 | SEQENn[31:24] | | | | | | | |
| 0x2C | CALIB | 7:0 | | | | | | BIASCOMP[2:0] | | |
| | | 15:8 | | | | | | BIASREFBUF[2:0] | | |

## 38.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to the section on Synchronization.


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization section.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | SLAVEEN | | | | ENABLE | SWRST |
| Access | R/W | R/W | R/W | | | | R/W | R/W |
| Reset | 0 | 0 | 0 | | | | 0 | 0 |

**Bit 7 – ONDEMAND** On Demand Control

The On Demand operation mode allows the ADC to be enabled or disabled, depending on other peripheral requests.
In On Demand operation mode, i.e., if the ONDEMAND bit has been previously set, the ADC will only be running when requested by a peripheral. If there is no peripheral requesting the ADC will be in a disabled state.
If On Demand is disabled the ADC will always be running when enabled.
In standby sleep mode, the On Demand operation is still active if the CTRLA.RUNSTDBY bit is '1'. If CTRLA.RUNSTDBY is '0', the ADC is disabled.
This bit is not synchronized.
For the Client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1). ONDEMAND bit from Host ADC instance will control the On Demand operation mode.

| Value | Description |
|---|---|
| 0 | The ADC is always on , if enabled. |
| 1 | The ADC is enabled, when a peripheral is requesting the ADC conversion. The ADC is disabled if no peripheral is requesting it. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls how the ADC behaves during standby sleep mode.
This bit is not synchronized.
For the Client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1). RUNSTDBY bit from Host ADC instance will control the Client ADC operation in standby sleep mode.

| Value | Description |
|---|---|
| 0 | The ADC is halted during standby sleep mode. |
| 1 | The ADC is not stopped in standby sleep mode. If CTRLA.ONDEMAND=1, the ADC will be running when a peripheral is requesting it. If CTRLA.ONDEMAND=0, the ADC will always be running in standby sleep mode. |

**Bit 5 – SLAVEEN** Client Enable

This bit enables the Host/Client operation and it is available only in the Client ADC instance (ADC1).
This bit is not synchronized.
This bit can be set only for the Client ADC (ADC1). For the Host ADC (ADC0), this bit is always read zero.

| Value | Description |
|---|---|
| 0 | The Host-Client operation is disabled. |
| 1 | The ADC1 is enabled as a Client of ADC0 |

**Bit 1 – ENABLE** Enable

Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the ENABLE bit in the


© 2026 Microchip Technology Inc. and its subsidiaries

SYNCBUSY register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
For the Client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Description |
|---|---|
| 0 | The ADC is disabled. |
| 1 | The ADC is enabled. |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the ADC, except DBGCTRL, to their initial state, and the ADC will be disabled.
Writing a '1' to CTRL.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.2. Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | PRESCALER[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – PRESCALER[2:0]** Prescaler Configuration
This field defines the ADC clock relative to the peripheral clock.
This field is not synchronized. For the client ADC, these bits have no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | Peripheral clock divided by 2 |
| 0x1 | DIV4 | Peripheral clock divided by 4 |
| 0x2 | DIV8 | Peripheral clock divided by 8 |
| 0x3 | DIV16 | Peripheral clock divided by 16 |
| 0x4 | DIV32 | Peripheral clock divided by 32 |
| 0x5 | DIV64 | Peripheral clock divided by 64 |
| 0x6 | DIV128 | Peripheral clock divided by 128 |
| 0x7 | DIV256 | Peripheral clock divided by 256 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.3. Reference Control

**Name:** REFCTRL
**Offset:** 0x02
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | REFCOMP | | | | REFSEL[3:0] | | | |
| Access | R/W | | | | R/W | R/W | R/W | R/W |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 7 – REFCOMP** Reference Buffer Offset Compensation Enable
The gain error can be reduced by enabling the reference buffer offset compensation. This will increase the start-up time of the reference.

| Value | Description |
|---|---|
| 0 | Reference buffer offset compensation is disabled. |
| 1 | Reference buffer offset compensation is enabled. |

**Bits 3:0 – REFSEL[3:0]** Reference Selection
These bits select the reference for the ADC.

| Value | Name | Description |
|---|---|---|
| 0x0 | INTREF | internal reference voltage, supplied by the bandgap (refer to SUPC.VREF.SEL for voltage level information) |
| x01 | INTVCC0 | 1/1.6 VDDANA |
| 0x2 | INTVCC1 | 1/2 VDDANA (only for VDDANA > 4.0V) |
| 0x3 | VREFA | External reference |
| 0x4 | DAC | DAC internal output |
| 0x5 | INTVCC2 | VDDANA |
| 0x6 - 0xF | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.4. Event Control

**Name:** EVCTRL
**Offset:** 0x03
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WINMONEO | RESRDYEO | STARTINV | FLUSHINV | STARTEI | FLUSHEI |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 5 – WINMONEO** Window Monitor Event Out
This bit indicates whether the Window Monitor event output is enabled or not and an output event will be generated when the window monitor detects something.

| Value | Description |
|---|---|
| 0 | Window Monitor event output is disabled and an event will not be generated. |
| 1 | Window Monitor event output is enabled and an event will be generated. |

**Bit 4 – RESRDYEO** Result Ready Event Out
This bit indicates whether the Result Ready event output is enabled or not and an output event will be generated when the conversion result is available.

| Value | Description |
|---|---|
| 0 | Result Ready event output is disabled and an event will not be generated. |
| 1 | Result Ready event output is enabled and an event will be generated. |

**Bit 3 – STARTINV** Start Conversion Event Invert Enable
For the client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Description |
|---|---|
| 0 | Start event input source is not inverted. |
| 1 | Start event input source is inverted. |

**Bit 2 – FLUSHINV** Flush Event Invert Enable
For the client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Description |
|---|---|
| 0 | Flush event input source is not inverted. |
| 1 | Flush event input source is inverted. |

**Bit 1 – STARTEI** Start Conversion Event Input Enable
For the client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Description |
|---|---|
| 0 | A new conversion will not be triggered on any incoming event. |
| 1 | A new conversion will be triggered on any incoming event. |

**Bit 0 – FLUSHEI** Flush Event Input Enable
For the client ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

| Value | Description |
|---|---|
| 0 | A flush and new conversion will not be triggered on any incoming event. |
| 1 | A flush and new conversion will be triggered on any incoming event. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Window Monitor Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The window monitor interrupt is disabled. |
| 1 | The window monitor interrupt is enabled, and an interrupt request will be generated when the Window Monitor interrupt flag is set. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Overrun Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled, and an interrupt request will be generated when the Overrun interrupt flag is set. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Result Ready Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled, and an interrupt request will be generated when the Result Ready interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Window Monitor Interrupt bit, which enables the Window Monitor interrupt.

| Value | Description |
|---|---|
| 0 | The Window Monitor interrupt is disabled. |
| 1 | The Window Monitor interrupt is enabled. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Overrun Interrupt bit, which enables the Overrun interrupt.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Result Ready Interrupt bit, which enables the Result Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x06
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor
This flag is cleared by writing a '1' to the flag or by reading the RESULT register.
This flag is set on the next GCLK_ADC cycle after a match with the window monitor condition, and an interrupt request will be generated if INTENCLR/SET.WINMON is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Window Monitor interrupt flag.

**Bit 1 – OVERRUN** Overrun
This flag is cleared by writing a '1' to the flag.
This flag is set if RESULT is written before the previous value has been read by CPU, and an interrupt request will be generated if INTENCLR/SET.OVERRUN=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Overrun interrupt flag.

**Bit 0 – RESRDY** Result Ready
This flag is cleared by writing a '1' to the flag or by reading the RESULT register.
This flag is set when the conversion result is available, and an interrupt will be generated if INTENCLR/SET.RESRDY=1.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Result Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.8. Sequence Status

**Name:** SEQSTATUS
**Offset:** 0x07
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SEQBUSY | | | SEQSTATE[4:0] | | | | |
| Access | R | | | R | R | R | R | R |
| Reset | 0 | | | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – SEQBUSY** Sequence busy
This bit is set when the sequence start.
This bit is clear when the last conversion in a sequence is done.

**Bits 4:0 – SEQSTATE[4:0]** Sequence State
These bit fields are the pointer of sequence. This value identifies the last conversion done in the sequence.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.9. Input Control

**Name:** INPUTCTRL
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | MUXNEG[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | MUXPOS[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 12:8 – MUXNEG[4:0]** Negative MUX Input Selection
These bits define the MUX selection for the negative ADC input.

| Value | Name | Description |
|---|---|---|
| 0x00 | AIN0 | ADC AIN0 pin |
| 0x01 | AIN1 | ADC AIN1 pin |
| 0x02 | AIN2 | ADC AIN2 pin |
| 0x03 | AIN3 | ADC AIN3 pin |
| 0x04 | AIN4 | ADC AIN4 pin |
| 0x05 | AIN5 | ADC AIN5 pin |
| 0x06 – 0x17 | - | Reserved |
| 0x18 | GND | Internal ground |
| 0x19 – 0x1F | - | Reserved |

**Bits 4:0 – MUXPOS[4:0]** Positive MUX Input Selection
These bits define the MUX selection for the positive ADC input. If the internal INTREF voltage input channel is selected, then the Sampling Time Length bit group in the Sampling Control register must be written with a corresponding value.

| Value | Name | Description |
|---|---|---|
| 0x00 | AIN0 | ADC AIN0 pin |
| 0x01 | AIN1 | ADC AIN1 pin |
| 0x02 | AIN2 | ADC AIN2 pin |
| 0x03 | AIN3 | ADC AIN3 pin |
| 0x04 | AIN4 | ADC AIN4 pin |
| 0x05 | AIN5 | ADC AIN5 pin |
| 0x06 | AIN6 | ADC AIN6 pin |
| 0x07 | AIN7 | ADC AIN7 pin |
| 0x08 | AIN8 | ADC AIN8 pin |
| 0x09 | AIN9 | ADC AIN9 pin |
| 0x0A | AIN10 | ADC AIN10 pin |
| 0x0B | AIN11 | ADC AIN11 pin |
| 0xC – 0x18 | - | Reserved |
| 0x19 | INTREF | Internal voltage reference, supplied by the bandgap (refer to SUPC.VREF.SEL for voltage level information) |


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 0x1A | SCALEDVDDCORE | 1/4 Scaled VDDCORE Supply |
| 0x1B | SCALEDVDDANA | 1/4 Scaled VDDANA Supply |
| 0x1C - 0x1F | - | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.10. Control C

**Name:** CTRLC
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | DUALSEL[1:0] | | | WINMODE[2:0] | | |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 0 | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | R2R | | RESSEL[1:0] | | CORREN | FREERUN | LEFTADJ | DIFFMODE |
| Access | R/W | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 13:12 – DUALSEL[1:0]** Dual Mode Trigger Selection
These bits define the trigger mode. These bits are available in the Host ADC and have no effect if the Host-Client operation is disabled (ADC1.CTRLA.SLAVEEN=0).

| Value | Name | Description |
|---|---|---|
| 0x0 | BOTH | Start event or software trigger will start a conversion on both ADCs. |
| 0x1 | INTERLEAVE | Start event or software trigger will alternatingly start a conversion on ADC0 and ADC1. |
| 0x2 – 0x3 | - | Reserved |

**Bits 10:8 – WINMODE[2:0]** Window Monitor Mode
These bits enable and define the window monitor mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | No window mode (default) |
| 0x1 | MODE1 | RESULT > WINLT |
| 0x2 | MODE2 | RESULT < WINUT |
| 0x3 | MODE3 | WINLT < RESULT < WINUT |
| 0x4 | MODE4 | WINUT < RESULT < WINLT |
| 0x5 – 0x7 | | Reserved |

**Bit 7 – R2R** Rail-to-Rail Operation

| Value | Description |
|---|---|
| 0 | Disable rail-to-rail operation. |
| 1 | Enable rail-to-rail operation to increase the allowable range of the input common mode voltage ($V_{CMIN}$). When R2R is one, a sampling period of four cycles is required. Offset compensation (SAMPCTRL.OFFCOMP) must be written to one when using this period. |

**Bits 5:4 – RESSEL[1:0]** Conversion Result Resolution
These bits define whether the ADC completes the conversion 12-, 10- or 8-bit result resolution.

| Value | Name | Description |
|---|---|---|
| 0x0 | 12BIT | 12-bit result |
| 0x1 | 16BIT | Accumulation or Oversampling and Decimation modes |
| 0x2 | 10BIT | 10-bit result |
| 0x3 | 8BIT | 8-bit result |

**Bit 3 – CORREN** Digital Correction Logic Enabled


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Disable the digital result correction. |
| 1 | Enable the digital result correction. The ADC conversion result in the RESULT register is then corrected for gain and offset based on the values in the GAINCORR and OFFSETCORR registers. Conversion time will be increased by 13 cycles according to the value in the Offset Correction Value bit group in the Offset Correction register. |

**Bit 2 – FREERUN** Free Running Mode

| Value | Description |
|---|---|
| 0 | The ADC run in single conversion mode. |
| 1 | The ADC is in free running mode and a new conversion will be initiated when a previous conversion completes. |

**Bit 1 – LEFTADJ** Left-Adjusted Result

| Value | Description |
|---|---|
| 0 | The ADC conversion result is right-adjusted in the RESULT register. |
| 1 | The ADC conversion result is left-adjusted in the RESULT register. The high byte of the 12-bit result will be present in the upper part of the result register. Writing this bit to zero (default) will right-adjust the value in the RESULT register. |

**Bit 0 – DIFFMODE** Differential Mode

| Value | Description |
|---|---|
| 0 | The ADC is running in singled-ended mode. |
| 1 | The ADC is running in differential mode. In this mode, the voltage difference between the MUXPOS and MUXNEG inputs will be converted by the ADC. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.11. Average Control

**Name:** AVGCTRL
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | ADJRES[2:0] | | | SAMPLENUM[3:0] | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 6:4 – ADJRES[2:0]** Adjusting Result / Division Coefficient
These bits define the division coefficient in 2n steps.

**Bits 3:0 – SAMPLENUM[3:0]** Number of Samples to be Collected
These bits define how many samples are added together. The result will be available in the Result register (RESULT). Note: if the result width increases, CTRLC.RESSEL must be changed.

| Value | Description |
|---|---|
| 0x0 | 1 sample |
| 0x1 | 2 samples |
| 0x2 | 4 samples |
| 0x3 | 8 samples |
| 0x4 | 16 samples |
| 0x5 | 32 samples |
| 0x6 | 64 samples |
| 0x7 | 128 samples |
| 0x8 | 256 samples |
| 0x9 | 512 samples |
| 0xA | 1024 samples |
| 0xB - 0xF | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.12. Sampling Time Control

**Name:** SAMPCTRL
**Offset:** 0x0D
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OFFCOMP | | SAMPLEN[5:0] | | | | | |
| Access | R/W | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – OFFCOMP** Comparator Offset Compensation Enable
Setting this bit enables the offset compensation for each sampling period to ensure low offset and immunity to temperature or voltage drift. This compensation increases the sampling time by three clock cycles that results in a fixed sampling duration of 4 CLK_ADC cycles.
This bit must be set to zero to validate the SAMPLEN value. It's not possible to use OFFCOMP=1 and SAMPLEN>0.

**Bits 5:0 – SAMPLEN[5:0]** Sampling Time Length
These bits control the ADC sampling time in number of CLK_ADC cycles, depending of the prescaler value, thus controlling the ADC input impedance. Sampling time is set according to the equation given below:
$\text{Sampling time} = (\text{SAMPLEN}+1)/(f_{CLK_\text{ADC}})$


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.13. Window Monitor Lower Threshold

**Name:** WINLT
**Offset:** 0x0E
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – WINLT[15:0]** Window Lower Threshold
If the window monitor is enabled (CTRLC.WINMODE != 0), these bits define the lower threshold value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.14. Window Monitor Upper Threshold

**Name:** WINUT
**Offset:** 0x10
**Reset:** 0x0000
**Property:** PAV Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – WINUT[15:0]** Window Upper Threshold
If the window monitor is enabled (CTRLC.WINMODE != 0), these bits define the upper threshold value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.15. Gain Correction

**Name:** GAINCORR
**Offset:** 0x12
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | GAINCORR[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GAINCORR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 11:0 – GAINCORR[11:0]** Gain Correction Value
If CTRLC.CORREN=1, these bits define how the ADC conversion result is compensated for gain error before being written to the result register. The gain correction is a fractional value, a 1-bit integer plus an 11-bit fraction, and therefore ½ <= GAINCORR < 2. GAINCORR values range from 0.10000000000 to 1.11111111111.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.16. Offset Correction

**Name:** OFFSETCORR
**Offset:** 0x14
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OFFSETCORR[11:8] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETCORR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 11:0 – OFFSETCORR[11:0]** Offset Correction Value
If CTRLC.CORREN=1, these bits define how the ADC conversion result is compensated for offset error before being written to the Result register. This OFFSETCORR value is in two's complement format.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.17. Software Trigger

**Name:** SWTRIG
**Offset:** 0x18
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | START | FLUSH |
| Access | | | | | | | W | W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – START** ADC Start Conversion
Writing a '1' to this bit will start a conversion or sequence. The bit is cleared by hardware when the conversion has started. Writing a '1' to this bit when it is already set has no effect.
Writing a '0' to this bit will have no effect.

**Bit 0 – FLUSH** ADC Conversion Flush
Writing a '1' to this bit will flush the ADC pipeline. A flush will restart the ADC clock on the next peripheral clock edge, and all conversions in progress will be aborted and lost. This bit is cleared until the ADC has been flushed.
After the flush, the ADC will resume where it left off; i.e., if a conversion was pending, the ADC will start a new conversion.
Writing this bit to '0' will have no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.18. Debug Control

**Name:** DBGCTRL
**Offset:** 0x1C
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.
This bit should be written only while a conversion is not ongoing.

| Value | Description |
|---|---|
| 0 | The ADC is halted when the CPU is halted by an external debugger. |
| 1 | The ADC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.19. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x20
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SWTRIG | OFFSETCORR | GAINCORR |
| Access | | | | | | R | R | R |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINUT | WINLT | SAMPCTRL | AVGCTRL | CTRLC | INPUTCTRL | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 10 – SWTRIG** Software Trigger Synchronization Busy
This bit is cleared when the synchronization of SWTRIG register between the clock domains is complete.
This bit is set when the synchronization of SWTRIG register between clock domains is started.

**Bit 9 – OFFSETCORR** Offset Correction Synchronization Busy
This bit is cleared when the synchronization of OFFSETCORR register between the clock domains is complete.
This bit is set when the synchronization of OFFSETCORR register between clock domains is started.

**Bit 8 – GAINCORR** Gain Correction Synchronization Busy
This bit is cleared when the synchronization of GAINCORR register between the clock domains is complete.
This bit is set when the synchronization of GAINCORR register between clock domains is started.

**Bit 7 – WINUT** Window Monitor Lower Threshold Synchronization Busy
This bit is cleared when the synchronization of WINUT register between the clock domains is complete.
This bit is set when the synchronization of WINUT register between clock domains is started.

**Bit 6 – WINLT** Window Monitor Upper Threshold Synchronization Busy
This bit is cleared when the synchronization of WINLT register between the clock domains is complete.
This bit is set when the synchronization of WINLT register between clock domains is started.

**Bit 5 – SAMPCTRL** Sampling Time Control Synchronization Busy
This bit is cleared when the synchronization of SAMPCTRL register between the clock domains is complete.
This bit is set when the synchronization of SAMPCTRL register between clock domains is started.

**Bit 4 – AVGCTRL** Average Control Synchronization Busy
This bit is cleared when the synchronization of AVGCTRL register between the clock domains is complete.
This bit is set when the synchronization of AVGCTRL register between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 3 – CTRLC** Control C Synchronization Busy
This bit is cleared when the synchronization of CTRLC register between the clock domains is complete.
This bit is set when the synchronization of CTRLC register between clock domains is started.

**Bit 2 – INPUTCTRL** Input Control Synchronization Busy
This bit is cleared when the synchronization of INPUTCTRL register between the clock domains is complete.
This bit is set when the synchronization of INPUTCTRL register between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE register between the clock domains is complete.
This bit is set when the synchronization of ENABLE register between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST register between the clock domains is complete.
This bit is set when the synchronization of SWRST register between clock domains is started


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.20. Result

**Name:** RESULT
**Offset:** 0x24
**Reset:** 0x0000
**Property:** -

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – RESULT[15:0]** Result Conversion Value

These bits will hold up to a 16-bit ADC conversion result, depending on the configuration.
In single conversion mode without averaging, the ADC conversion will produce a 12-bit result, which can be left- or right-shifted, depending on the setting of CTRLC.LEFTADJ.
If the result is left-adjusted (CTRLC.LEFTADJ = 1), the high byte of the result will be in bit position [15:8], while the remaining 4 bits of the result will be placed in bit locations [7:4]. This can be used only if an 8-bit result is needed; i.e., one can read only the high byte of the entire 16-bit register.
If the result is not left-adjusted (CTRLC.LEFTADJ = 0) and no oversampling is used, the result will be available in bit locations [11:0], and the result is then 12 bits long. If oversampling is used, the result will be located in bit locations [15:0], depending on the settings of the Average Control register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.21. Sequence Control

**Name:** SEQCTRL
**Offset:** 0x28
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SEQENn[31:24] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SEQENn[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SEQENn[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SEQENn[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – SEQENn[31:0]** Enable Positive Input in the Sequence
For details on available positive mux selection, refer to INPUTCTRL.MUXPOS.
The sequence starts from the lowest input, and goes to the next enabled input automatically when the conversion is done. If no bits are set the sequence is disabled.

| Value | Description |
|---|---|
| 0 | Disable the positive input mux n selection from the sequence. |
| 1 | Enable the positive input mux n selection to the sequence. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 38.8.22. Calibration

**Name:** CALIB
**Offset:** 0x2C
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BIASREFBUF[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | BIASCOMP[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 10:8 – BIASREFBUF[2:0]** Bias Reference Buffer Scaling
This value from production test must be loaded from the NVM software calibration row into the CALIB register by software to achieve the specified accuracy.
The value must be copied only, and must not be changed.

**Bits 2:0 – BIASCOMP[2:0]** Bias Comparator Scaling
This value from production test must be loaded from the NVM software calibration row into the CALIB register by software to achieve the specified accuracy.
The value must be copied only, and must not be changed

**Related Links**
NVM Software Calibration Area Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

# 39. SDADC – Sigma-Delta Analog-to-Digital Converter (SAM C21 only)

## 39.1. Overview

The Sigma-Delta Analog-to-Digital Converter (SDADC) converts analog signals to digital values. The SDADC has 16-bit resolution, and is capable of converting up to 1.5 Msps divided by the data over sampling ratio (OSR). The input selection is up to three differential analog channels. The SDADC provides signed results.

ADC measurements can be started by either application software or an incoming event from another peripheral in the device. ADC measurements can be started with predictable timing, and without software intervention.

The SDADC also integrates a sleep mode and a conversion sequencer. These features reduce power consumption and processor intervention.

A set of reference voltages is generated internally.

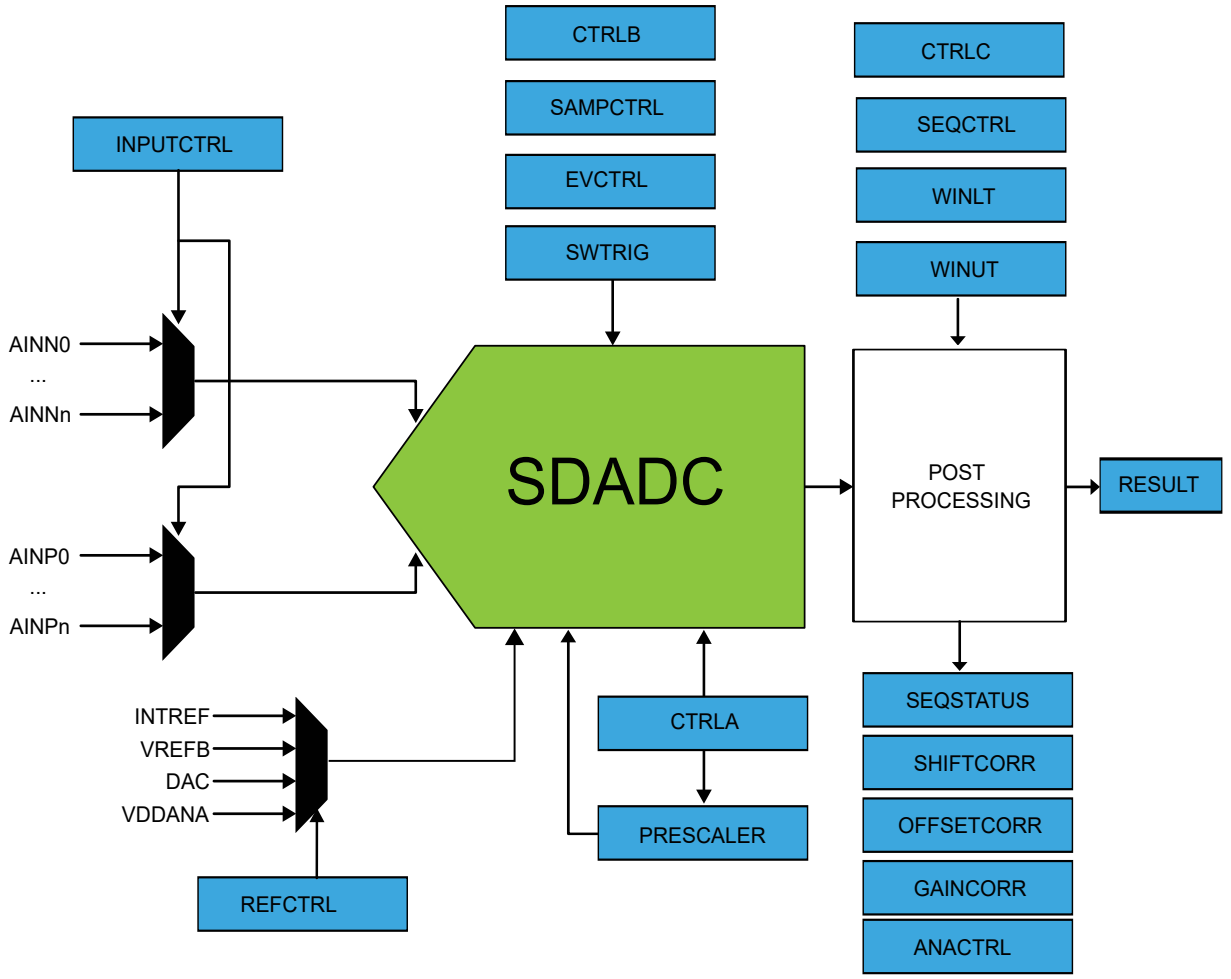
## 39.2. Features

- 16-bit resolution
- Up to 1,500,000 divided by the Over Sampling Ratio (OSR) samples per second
- Three analog differential inputs
  - Up to 3 external analog differential pairs
- Conversion Range:
  - Differential mode: $-V_{REF}$ to $+V_{REF}$
- Event-triggered conversion (one event input)
- Optional DMA transfer of conversion settings or result
- Single, continuous and sequencing options
- Hardware gain, offset and shift compensation
- Windowing monitor
- Chopper mode (offset reduction)


© 2026 Microchip Technology Inc. and its subsidiaries

## 39.3. Block Diagram

**Figure 39-1.** SDADC Block Diagram.




## 39.4. Signal Description

One signal can be mapped on several pins.

| Signal | Description | Type |
|---|---|---|
| VREF | Analog input | External reference voltage |
| AINN0 | Analog input | Analog input channel |
| AINP0 | Analog input | Analog input channel |
| AINN1 | Analog input | Analog input channel |
| AINP1 | Analog input | Analog input channel |
| AINN2 | Analog input | Analog input channel |
| AINP2 | Analog input | Analog input channel |

**Related Links**

I/O Multiplexing and Considerations


© 2026 Microchip Technology Inc. and its subsidiaries

## 39.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 39.5.1. I/O Lines

Using the SDADC's I/O lines requires the I/O pins to be configured using the port configuration (PORT).

**Related Links**

PORT - I/O Pin Controller

Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics

### 39.5.2. Power Management

The SDADC will continue to operate in any sleep mode where the selected source clock is running. The SDADC's interrupts can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes. Refer to the Power Manager chapter for details on the different sleep modes.

### 39.5.3. Clocks

The SDADC bus clock (CLK_SDADC_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_SDADC_APB can be found in the Peripheral Clock Masking section.

A generic clock (GCLK_SDADC) is required to generate the CLK_SDADC for the SDADC analog module. This clock must be configured and enabled in the Generic Clock Controller (GCLK) before using the SDADC. The CLK_SDADC is the SDADC clock connected to the SDADC analog module and its range is between GCLK_SDADC/2, (if PRESCALER is 0) and GCLK_SDADC/512 (if PRESCALER is 255). Please refer to the CTRLB register for more details.

The SDADC data sampling clock CLK_SDADC_FS in the SDADC analog module is the CLK_SDADC/4.

This GCLK_SDADC is asynchronous to the bus clock (CLK_SDADC_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains.

**Related Links**

Peripheral Clock Masking

GCLK - Generic Clock Controller

### 39.5.4. DMA

The DMA request line is connected to the DMA Controller (DMAC). Using the SDADC DMA requests requires the DMA

Controller to be configured first. .

**Related Links**

DMAC – Direct Memory Access Controller

### 39.5.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the SDADC interrupt requires the interrupt controller to be configured first.

**Related Links**

Nested Vector Interrupt Controller

### 39.5.6. Events

The events are connected to the Event System. Refer to the Event System section for details on how to configure the Event System.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

### 39.5.7. Debug Operation

When the CPU is halted in debug mode the SDADC will halt normal operation. The SDADC can be forced to continue operation

during debugging. Refer to DBGCTRL for details.

### 39.5.8. Register Access Protection

All registers with write-access are optionally write-protected by the Peripheral Access Controller (PAC), except the following register:

- Interrupt Flag Status and Clear (INTFLAG) register

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Write-protection does not apply for accesses through an external debugger.

### 39.5.9. Analog Connections

I/O-pins (AINx), as well as the REF reference voltage pins are analog inputs to the SDADC.

## 39.6. Functional Description

### 39.6.1. Principle of Operation

The Sigma-Delta Analog-to-Digital Converter (SDADC) converts analog signals to digital values. The SDADC has 16-bit resolution, and is capable of converting up to 1.5 Msps divided by the OSR data over sampling ratio. The input selection is up to three input analog channels. The SDADC provides unsigned results.

**Related Links**
CTRLB
Control B

### 39.6.2. Basic Operation

#### 39.6.2.1. Initialization

The following registers are enable-protected, meaning that they can only be written when the SDADC is disabled (CTRLA.ENABLE is zero):

- CTRLA ONEDEMAND and RUNSTDBY bits
- CTRLB
- CTRLC
- EVCTRL
- ANACTRL

Enable-protection is denoted by the Enable-Protected property in the register description.

#### 39.6.2.2. Enabling, Disabling and Resetting

The SDADC is enabled by writing a one to the Enable bit in the Control A register (CTRLA.ENABLE). The SDADC is disabled by writing a zero to CTRLA.ENABLE.

The SDADC is reset by writing a one to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the ADC will be reset to their initial state, and the SDADC will be disabled. Refer to CTRLA for details.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.6.2.3. Operation

In the most basic configuration, the SDADC samples values from the configured external sources (input ctrl register). The rate of the conversion depends on the combination of the GCLK_SDADC frequency, the clock prescaler from CTRLB.PRESCALER and the Over Sampling Ratio from CTRLB.OSR.

To convert analog values to digital values, the SDADC needs to be initialized first, as described in Initialization . Data conversion can be started either manually, by writing a one to the Start bit in the Software Trigger register (SWTRIG.START), or automatically, by configuring an automatic trigger to initiate the conversions. A free-running mode can be used to continuously convert an input channel. There is no need for a trigger to start the conversion. It will start automatically at the end of previous conversion.

The first valid sample starts from the third sample onward. It can skip the first few samples by programming the SKPCNT[3:0] in CTRLB register. The result of the conversion is stored in the Result register (RESULT) overwriting the result from the previous conversion.

To avoid data loss the conversion result must be read as soon as it is available (INTFLAG.RESRDY). Failing to do so will result in an overrun error condition, indicated by the OVERRUN bit in the Interrupt Flag Status and Clear register (INTFLAG.OVERRUN).

To use an interrupt handler, the corresponding bit in the Interrupt Enable Set register (INTENSET) must be written to one.

### 39.6.2.4. Conversion Reference

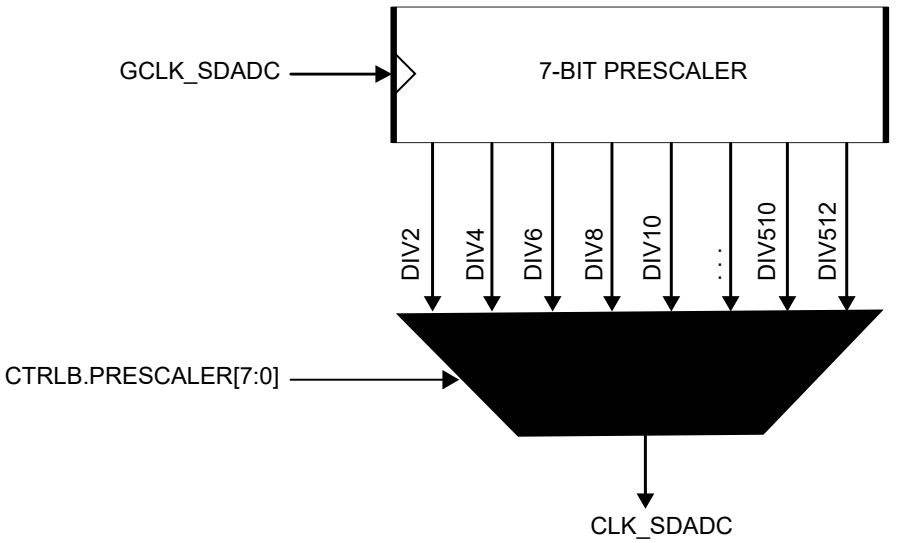
The conversion is performed on a full range between 0V and the reference voltage. Analog inputs between these voltages convert to values based on a linear conversion.

### 39.6.2.5. Prescaler Selection

The SDADC is clocked by GCLK_SDADC. There is also a prescaler in the SDADC to enable conversion at lower clock rates.

Refer to CTRLB for details on prescaler settings.

**Figure 39-2.** SDADC Prescaler Diagram.




### 39.6.2.6. SDADC Resolution

The SDADC provides 16-bit resolution.


© 2026 Microchip Technology Inc. and its subsidiaries

#### 39.6.2.7. Automatic Sequences

The SDADC has the ability to automatically sequence a series of conversion. This means that each time the SDADC receives a start-of-conversion request, it can perform multiple conversions automatically. All of the three inputs can be included in a sequence by writing to the Sequence Control register (SEQCTRL). The order of the conversion in a sequence is the lower positive input pair selection to upper positive input pair (AINN0, AINP0, AINN1, AINP1, AINN2, AINP2).

When a sequence starts, the Sequence Busy status bit in Sequence Status register (SEQSTATUS.SEQBUSY) will be set to one. When the sequence is complete, the Sequence Busy status bit will be cleared.

Each time a conversion is completed, the Sequence State status in Sequence Status register (SEQSTATUS.SEQSTATE) will store the input number from which the conversion is done. The result will be stored in RESULT register and the Result Ready Interrupt Flag (INTFLAG.RESRDY) is set.

If additional inputs must be scanned, the SDADC will automatically start a new conversion on the next input present in the sequence list.

Note that if SEQCTRL register has no bits set to one, the conversion is done with the selected INPUTCTRL input.

#### 39.6.2.8. Window Monitor

The window monitor feature allows the conversion result in the RESULT register to be compared to predefined threshold values. The window mode is selected by writing the Window Monitor Mode bits in the Window Monitor Control register (WINCTRL.WINMODE). Threshold values must be written in the Window Monitor Lower Threshold register (WINLT) and Window Monitor Upper Threshold register (WINUT).

The INTFLAG.WINMON interrupt flag will be set if the conversion result matches the window monitor condition.

### 39.6.3. CIC (Cascaded Integrator-Comb) Decimation Filter

#### 39.6.3.1. Description

The Analog-to-Digital Converter filters and decimates the sigma-delta ADC output bitstream. Its output is defined on 16bits unsigned format with the following programmable output rates: CLK_SDADC_FS/64, CLK_SDADC_FS/128, CLK_SDADC_FS/256, CLK_SDADC_FS/512 and CLK_SDADC_FS/1024, where CLK_SDADC_FS is the sigma-delta ADC's sampling frequency: CLK_SDADC_FS = CLK_SDADC_PRESCALER/4, the reduction comes from the phase generator between the prescaler and the SDADC.

The filtering and the decimation is performed by a SINC-based filter whose zeros are placed in order to minimize aliasing effects of the decimation.

#### 39.6.3.2. Decimation Filter

The sigma-delta architecture of the SDADC implies a filtering and a decimation of the bitstream at the output of the SDADC. The decimation filter decimates the bitstream by 64, 128 or 256, 512, 1024. To perform the decimation operation, a 3rd order SINC filter with programmable Over Sampling Ratio is implemented with the following transfer function:

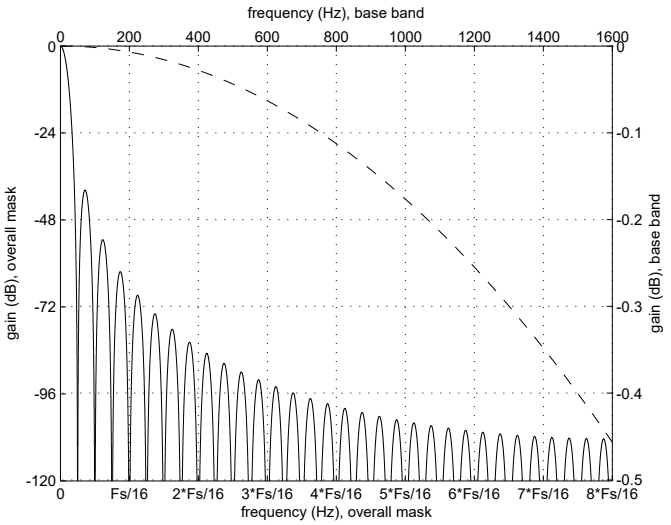
$$H(z) = \frac{1}{\mathrm{OSR}^3}\left(\sum_{i=0}^{\mathrm{OSR}+-1} z^{-i}\right)^3$$

OSR is the Over Sampling Ratio which can be modified to change the output data rate (See CTRLC for the setting of this parameter).

The DC gain of this filter is unity and does not depend on its OSR. However, as it generates a 3rd order zero at (CLK_SDADC_FS / OSR) frequency multiples, its frequency response depends on the OSR parameter. See next section for frequency plots.

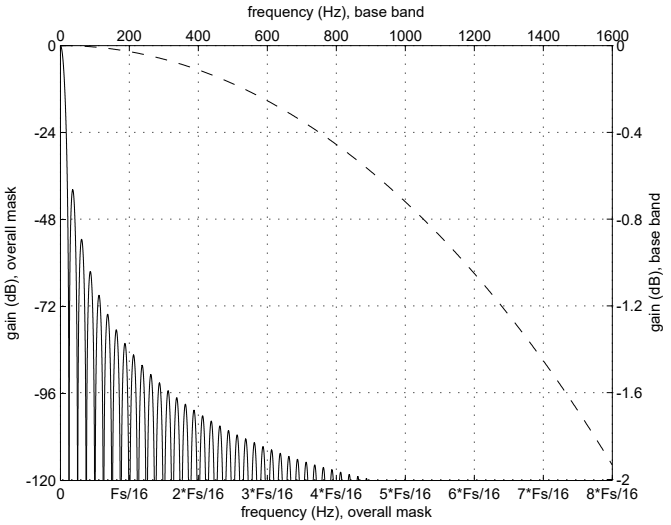

© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 39-3.** Spectral Mask of an OSR = 64, CLK_SDADC_FS = 1 MHz, 3rd Order Sinc Filter Overall Response (Continuous Line) and 0–1600Hz Bandwidth Response (Dashed Line)




The zeros of this filter are located at multiples of CLK_SDADC_FS/64.

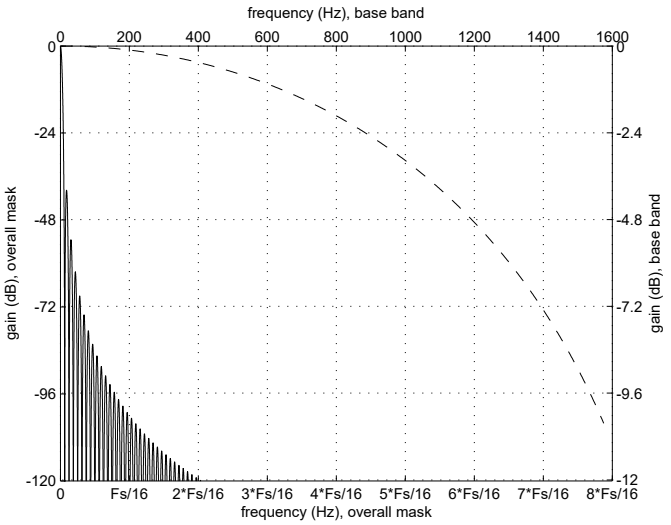
**Figure 39-4.** Spectral Mask of an OSR = 128, CLK_SDADC_FS = 1 MHz, 3rd Order Sinc Filter Overall Response (Continuous Line) and 0–1600 Hz Bandwidth Response (Dashed Line)




The zeros of this filter are located at multiples of CLK_SDADC_FS/128.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 39-5.** Spectral Mask of an OSR = 256, CLK_SDADC_FS = 1 MHz, 3rd Order Sinc Filter Overall Response (Continuous Line) and 0–1600 Hz Bandwidth Response (Dashed Line)




The zeros of this filter are located at multiples of CLK_SDADC_FS/256.

#### 39.6.3.3. Conversion Time

The time needed to convert a value depends on the selected OSR, PRESCALER and on the frequency of the SDADC.

For example, a sigma-delta converter running at CLK_GEN_SDADC = 1MHz with program the OSR of 64 and PRESCALER of 0. The output sampling rate equation is CLK_GEN_SDADC/(OSR * PRESCALER * 4) which means to converts data every (64*2*4)/1e6= 512µs. The output data rate is then 1.953ksps.

Note: The CLK_SDADC_PRESCAL clock range is CLK_GEN_SDADC/2, if PRESCAL is 0

The OSR and PRESCALER are described in CTRLB register.

#### 39.6.3.4. Gain and Offset Compensation

A specific offset, gain and shift can be applied to each source of the SDADC by performing the following operation:

$$\mathrm{Data} = (\mathrm{Data}_0 + \mathrm{OFFSET}) \times \frac{\mathrm{GAIN}}{2^{\mathrm{SHIFT}}}$$

Where:

Data0 is an unsigned integer defined on 16 bits. It is the output of the decimation filter.

OFFSET is a signed integer defined on 24 bits (OFFSETCORR register).

GAIN is an unsigned integer defined on 14 bits (GAINCORR register).

SHIFT is an unsigned integer defined on 4 bits (SHIFTCORR register).

The result of the operation is then saturated to be within [0:216-1] and the 16 LSBs of this saturation operation are sent to the controller as the result of the SDADC conversion.

Offset error can be compensated by setting the Chopper mode ON, refer to the ANACTRL.ONCHOP bit.

### 39.6.4. DMA Operation

The SDADC generates the following DMA request:

Result Conversion Ready (RESRDY): the request is set when a conversion result is available and cleared when


© 2026 Microchip Technology Inc. and its subsidiaries

the RESULT register is read.

### 39.6.5. Interrupts

The SDADC has the following interrupt sources:

- Result Conversion Ready: RESRDY
- Window Monitor: WINMON
- Overrun: OVERRUN

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the SDADC is reset. See INTFLAG for details on how to clear interrupt flags.
All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**
Nested Vector Interrupt Controller

### 39.6.6. Events

The SDADC can generate the following output events:

- Result Ready (RESRDY): Generated when the conversion is complete and the result is available. Refer to EVCTRL for details.
- Window Monitor (WINMON): Generated when the window monitor condition match. Refer to WINCTRL register for details.

Writing a one to an Event Output bit in the Event Control Register (EVCTRL.xxEO) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. Refer to the Event System chapter for details on configuring the event system.

The SDADC can take the following actions on an input event:

- Start conversion (START): Start a conversion. Refer to SWTRIG for details.
- Conversion flush (FLUSH): Flush the conversion. Refer to SWTRIG for details.

Writing a one to an Event Input bit into the Event Control register (EVCTRL.xxEI) enables the corresponding action on input event. Writing a zero to this bit disables the corresponding action on input event.

The SDADC uses only asynchronous events. Therefore, an asynchronous Event System channel path must be configured. By default, the SDADC will detect a rising edge on the incoming event. If the SDADC action must be performed on the falling edge of the incoming event, the event line must be inverted first, by writing to one the corresponding Event Invert Enable bit in Event Control register (EVCTRL.xINV).

Note that If FLUSH and START events are available at the same time, the FLUSH event has higher priority.


© 2026 Microchip Technology Inc. and its subsidiaries

**Related Links**
Event System (EVSYS)

### 39.6.7. Sleep Mode Operation

The ONDEMAND and RUNSTDBY bits in the Control A register (CTRLA) control the behavior of the SDADC during standby sleep mode, in cases where the SDADC is enabled (CTRLA.ENABLE = 1). Note that when CTRLA.ONDEMAND is one, the analog

block is powered-off when the conversion is complete. When a start request is detected, the system returns from sleep and starts a new conversion after the start-up time delay.

**Table 39-1.** SDADC Sleep Behavior

| CTRLA.RUNSTDBY | CTRLA.ONDEMAND | CTRLA.ENABLE | Description |
|---|---|---|---|
| x | x | 0 | Disabled |
| 0 | 0 | 1 | Run in all sleep modes except STANDBY. |
| 0 | 1 | 1 | Run in all sleep modes on request, except STANDBY. |
| 1 | 0 | 1 | Run in all sleep modes. |
| 1 | 1 | 1 | Run in all sleep modes on request. |

### 39.6.8. Synchronization

Due to the asynchronicity between CLK_SDADC_APB and CLK_GEN_SDADC some registers must be synchronized when accessed. A register can require:

- Synchronization when written
- Synchronization when read
- Synchronization when written and read
- No synchronization

When executing an operation that requires synchronization, the corresponding synchronization bit is set in Synchronization Busy register (SYNCBUSY) and cleared when synchronization is complete.

If an operation that require synchronization is executed while its busy bit is on, the operation is discarded and a bus error is generated.

The following bits need synchronization when written:

- Software Reset bit in Control A register (CTRLA.SWRST)
- Enable bit in Control A register (CTRLA.ENABLE)

Write-synchronization is denoted by the Write-Synchronized property in the register description.

The following registers need synchronization when written:

- Input Control register (INPUTCTRL)
- Reference Control register (REFCTRL)
- Control C register (CTRLC)
- Window Monitor Lower Threshold register (WINLT)
- Window Monitor Upper Threshold register (WINUT)
- Offset correction register (OFFSETCORR)
- Gain correction register (GAINCORR)
- Shift correction register (SHIFTCORR)
- Software Trigger register (SWTRIG)
- Analog Control Register (ANACTRL)


© 2026 Microchip Technology Inc. and its subsidiaries

Write-synchronization is denoted by the Write-Synchronized property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

## 39.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | ONDEMAND | RUNSTDBY | | | | | ENABLE | SWRST |
| 0x01 | REFCTRL | 7:0 | ONREFBUF | | | | | | REFSEL[1:0] | |
| 0x02 | CTRLB | 7:0 | PRESCALER[7:0] | | | | | | | |
| | | 15:8 | SKPCNT[3:0] | | | | | OSR[2:0] | | |
| 0x04 | EVCTRL | 7:0 | | | WINMONEO | RESRDYEO | STARTINV | FLUSHINV | STARTEI | FLUSHEI |
| 0x05 | INTENCLR | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x06 | INTENSET | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x07 | INTFLAG | 7:0 | | | | | | WINMON | OVERRUN | RESRDY |
| 0x08 | SEQSTATUS | 7:0 | SEQBUSY | | | | SEQSTATE[3:0] | | | |
| 0x09 | INPUTCTRL | 7:0 | | | | | MUXSEL[3:0] | | | |
| 0x0A | CTRLC | 7:0 | | | | | | | | FREERUN |
| 0x0B | WINCTRL | 7:0 | | | | | | WINMODE[2:0] | | |
| 0x0C | WINLT | 7:0 | WINLT[7:0] | | | | | | | |
| | | 15:8 | WINLT[15:8] | | | | | | | |
| | | 23:16 | WINLT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | WINUT | 7:0 | WINUT[7:0] | | | | | | | |
| | | 15:8 | WINUT[15:8] | | | | | | | |
| | | 23:16 | WINUT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | OFFSETCORR | 7:0 | OFFSETCORR[7:0] | | | | | | | |
| | | 15:8 | OFFSETCORR[15:8] | | | | | | | |
| | | 23:16 | OFFSETCORR[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | GAINCORR | 7:0 | GAINCORR[7:0] | | | | | | | |
| | | 15:8 | | | GAINCORR[13:8] | | | | | |
| 0x1A | SHIFTCORR | 7:0 | | | | | SHIFTCORR[3:0] | | | |
| 0x1B | Reserved | | | | | | | | | |
| 0x1C | SWTRIG | 7:0 | | | | | | | START | FLUSH |
| 0x1D ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | SYNCBUSY | 7:0 | OFFSETCORR | WINUT | WINLT | WINCTRL | MUXCTRL | CTRLC | ENABLE | SWRST |
| | | 15:8 | | | | | ANACTRL | SWTRIG | SHIFTCORR | GAINCORR |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 | RESULT | 7:0 | RESULT[7:0] | | | | | | | |
| | | 15:8 | RESULT[15:8] | | | | | | | |
| | | 23:16 | RESULT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x28 | SEQCTRL | 7:0 | | | | | | SEQENn[2:0] | | |
| 0x29 ... 0x2B | Reserved | | | | | | | | | |
| 0x2C | ANACTRL | 7:0 | BUFTEST | ONCHOP | | CTLSDADC[4:0] | | | | |
| 0x2D | Reserved | | | | | | | | | |
| 0x2E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |

## 39.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to


© 2026 Microchip Technology Inc. and its subsidiaries

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized (ENABLE, SWRST)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONDEMAND | RUNSTDBY | | | | | ENABLE | SWRST |
| Access | R/W | R/W | | | | | R/W | R/W |
| Reset | 0 | 0 | | | | | 0 | 0 |

**Bit 7 – ONDEMAND** On Demand Control

The On Demand operation modes allows the SDADC to be enabled or disabled, depending on other peripheral request.
In On Demand operation mode, i.e., if the ONDEMAND bit has been previously written to one, the SDADC will only be running when requested by a peripheral. If there is no peripheral requesting the SDADC will be in a disable state.
If On Demand is disable the SDADC will always be running when enabled.
In standby sleep mode, the On Demand operation is still active if the CTRLA.RUNSTDBY bit is one. If CTRLA.RUNSTDBY is zero, the SDADC is disabled.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The SDADC is always on , if enabled. |
| 1 | The SDADC is enabled, when a peripheral is requesting the SDADC conversion. The SDADC is disabled if no peripheral is requesting it. |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls how the SDADC behaves during standby sleep mode:
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The SDADC is halted during standby sleep mode. |
| 1 | The SDADC is not stopped in standby sleep mode. If CTRLA.ONDEMAND is one, the SDADC will be running when a peripheral is requesting it. If CTRLA.ONDEMAND is zero, the SDADC will always be running in standby sleep mode. |

**Bit 1 – ENABLE** Enable

Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the ENABLE bit in the SYNCBUSY register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

| Value | Description |
|---|---|
| 0 | The SDADC is disabled. |
| 1 | The SDADC is enabled. |

**Bit 0 – SWRST** Software Reset

Writing a zero to this bit has no effect.
Writing a one to this bit resets all registers in the SDADC, except SYNCBUSY , to their initial state, and the SDADC will be disabled.
Writing a one to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.


© 2026 Microchip Technology Inc. and its subsidiaries

Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.2. Reference Control

**Name:** REFCTRL
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ONREFBUF | | | | | | REFSEL[1:0] | |
| Access | R/W | | | | | | R/W | R/W |
| Reset | 0 | | | | | | 0 | 0 |

**Bit 7 – ONREFBUF** Reference Buffer On
Turning on the buffer increases the impedance seen on the external reference, so that the current load reduces from 5µA to 0.10µA. This needs to be matched with whatever type of reference circuit is used.

| Value | Description |
|---|---|
| 0 | Reference Buffer Off |
| 1 | Reference Buffer On |

**Bits 1:0 – REFSEL[1:0]** Reference Selection
These bits select the reference for the ADC.
**Note:** The reference buffer should be enabled (ONREFBUF=1) when using the internal INTREF or DAC output as reference.

| Value | Name | Description |
|---|---|---|
| 0x0 | INTREF | Internal voltage reference, supplied by the bandgap (refer to SUPC.VREF.SEL for voltage level information) |
| 0x1 | VREFB pin | External 1-5.5V |
| 0x2 | DAC output | Internal 1-5.5V |
| 0x3 | VDDANA | Supply 2.7-5.5V |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.3. Control B

**Name:** CTRLB
**Offset:** 0x02
**Reset:** 0x2000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SKPCNT[3:0] | | | | | OSR[2:0] | | |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 1 | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PRESCALER[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:12 – SKPCNT[3:0]** Skip Count
How many skip samples before retrieve the first valid sample.
The first valid sample starts from the third sample onward.

**Bits 10:8 – OSR[2:0]** Over Sampling Ratio
OSR is the Over Sampling Ratio which can be modified to change the output data rate.
The OSR must never be changed while the SDADC is running. One must first place the SDADC in reset state, modify the OSR and then run the SDADC again.
**Example:** The sampling rate of the SDADC is 1.5Msps/OSR. The maximum sampling rate is then 1.5MSPS/OSR64 ≅ 23.4ksps and the minimum sampling rate is 1.5Msps/OSR1024 ≅ 1.5ksps

| Value | Name | Description |
|---|---|---|
| 0x0 | OSR64 | Over Sampling Ratio is 64 |
| x01 | OSR128 | Over Sampling Ratio is 128 |
| 0x2 | OSR256 | Over Sampling Ratio is 256 |
| 0x3 | OSR512 | Over Sampling Ratio is 512 |
| 0x4 | OSR1024 | Over Sampling Ratio is 1024 |
| 0x4 – 0xF | - | Reserved |

**Bits 7:0 – PRESCALER[7:0]** Prescaler Configuration
The ADC uses the SDADC Clock to perform conversions.
The CLK_SDADC_PRESCAL clock range is between CLK_GEN_SDADC/2, if PRESCAL is 0, and CLK_GEN_SDADC/512, if PRESCAL is set to 255 (0xFF). PRESCAL must be programmed in order to provide an CLK_SDADC_PRESCAL clock frequency according to the parameters given in the product Electrical Characteristics section.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.4. Event Control

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WINMONEO | RESRDYEO | STARTINV | FLUSHINV | STARTEI | FLUSHEI |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 5 – WINMONEO** Window Monitor Event Out
This bit indicates whether the Window Monitor event output is enabled or not and an output event will be generated when the window monitor detects something.

| Value | Description |
|---|---|
| 0 | Window Monitor event output is disabled and an event will not be generated. |
| 1 | Window Monitor event output is enabled and an event will be generated. |

**Bit 4 – RESRDYEO** Result Ready Event Out
This bit indicates whether the Result Ready event output is enabled or not and an output event will be generated when the conversion result is available.

| Value | Description |
|---|---|
| 0 | Result Ready event output is disabled and an event will not be generated. |
| 1 | Result Ready event output is enabled and an event will be generated. |

**Bit 3 – STARTINV** Start Conversion Event Invert Enable

| Value | Description |
|---|---|
| 0 | start event input source is not inverted. |
| 1 | start event input source is inverted. |

**Bit 2 – FLUSHINV** Flush Event Invert Enable

| Value | Description |
|---|---|
| 0 | flush event input source is not inverted. |
| 1 | flush event input source is inverted. |

**Bit 1 – STARTEI** Start Conversion Event Input Enable

| Value | Description |
|---|---|
| 0 | A new conversion will not be triggered on any incoming event. |
| 1 | A new conversion will be triggered on any incoming event. |

**Bit 0 – FLUSHEI** Flush Event Input Enable

| Value | Description |
|---|---|
| 0 | A flush and new conversion will not be triggered on any incoming event. |
| 1 | A flush and new conversion will be triggered on any incoming event. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Window Monitor Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The window monitor interrupt is disabled. |
| 1 | The window monitor interrupt is enabled, and an interrupt request will be generated when the Window Monitor interrupt flag is set. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Overrun Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled, and an interrupt request will be generated when the Overrun interrupt flag is set. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Result Ready Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled, and an interrupt request will be generated when the Result Ready interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x06
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Window Monitor Interrupt bit, which enables the Window Monitor interrupt.

| Value | Description |
|---|---|
| 0 | The Window Monitor interrupt is disabled. |
| 1 | The Window Monitor interrupt is enabled. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Overrun Interrupt bit, which enables the Overrun interrupt.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Result Ready Interrupt bit, which enables the Result Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x07
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMON | OVERRUN | RESRDY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINMON** Window Monitor
This flag is cleared by writing a one to the flag or by reading the RESULT register.
This flag is set on the next CLK_GEN_SDADC cycle after a match with the window monitor condition, and an interrupt request will be generated if INTENCLR/SET.WINMON is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Window Monitor interrupt flag.

**Bit 1 – OVERRUN** Overrun
This flag is cleared by writing a one to the flag.
This flag is set if RESULT is written before the previous value has been read by CPU, and an interrupt request will be generated if INTENCLR/SET.OVERRUN is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Overrun interrupt flag.

**Bit 0 – RESRDY** Result Ready
This flag is cleared by writing a one to the flag or by reading the RESULT register.
This flag is set when the conversion result is available, and an interrupt will be generated if INTENCLR/SET.RESRDY is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Result Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.8. Sequence Status

**Name:** SEQSTATUS
**Offset:** 0x08
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SEQBUSY | | | | SEQSTATE[3:0] | | | |
| Access | R | | | | R | R | R | R |
| Reset | 0 | | | | 0 | 0 | 0 | 0 |

**Bit 7 – SEQBUSY** Sequence busy
This bit is set when the sequence start.
This bit is clear when the last conversion in a sequence is done.

**Bits 3:0 – SEQSTATE[3:0]** Sequence State
This bit field is the pointer of sequence. This value identifies the last conversion done in the sequence.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.9. Input Control

**Name:** INPUTCTRL
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | MUXSEL[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – MUXSEL[3:0]** ADC Analog Input Selection
These bits define the Mux selection for the SDADC input.

| Value | Name | Description |
|---|---|---|
| 0x00 | AIN0 | Select ADC AINN0 and AINP0 pins |
| 0x01 | AIN1 | Select ADC AINN1 and AINP1 pins |
| 0x02 | AIN2 | Select ADC AINN2 and AINP2 pins |
| 0x03 – 0x0F | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.10. Control C

**Name:** CTRLC
**Offset:** 0x0A
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | FREERUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – FREERUN** Free Running Mode

| Value | Description |
|---|---|
| 0 | The SDADC run in single conversion mode. |
| 1 | The SDADC is in free running mode and a new conversion will be initiated when a previous conversion completes. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.11. Window Monitor Control

**Name:** WINCTRL
**Offset:** 0x0B
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINMODE[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – WINMODE[2:0]** Window Monitor Mode
These bits enable and define the window monitor mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | No window mode (default) |
| 0x1 | ABOVE | RESULT > WINLT |
| 0x2 | BELOW | RESULT < WINUT |
| 0x3 | INSIDE | WINLT < RESULT < WINUT |
| 0x4 | OUTSIDE | WINUT < RESULT or RESULT < WINLT |
| 0x5 – 0x7 | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.12. Window Monitor Lower Threshold

**Name:** WINLT
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – WINLT[23:0]** Window Lower Threshold
If the window monitor is enabled, these bits define the lower threshold value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.13. Window Monitor Upper Threshold

**Name:** WINUT
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – WINUT[23:0]** Window Upper Threshold
If the window monitor is enabled, these bits define the upper threshold value.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.14. Offset Correction

**Name:** OFFSETCORR
**Offset:** 0x14
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETCORR[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETCORR[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETCORR[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – OFFSETCORR[23:0]** Offset Correction

The OFFSETCORR is a signed integer value.

A specific offset, gain and shift can be applied to SDADC by performing the following operation:

```
(RESULT + OFFSETCORR)*GAINCORR/2^SHIFTCORR
```


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.15. Gain Correction

**Name:** GAINCORR
**Offset:** 0x18
**Reset:** 0x0001
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | GAINCORR[13:8] | | | | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GAINCORR[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 13:0 – GAINCORR[13:0]** Gain Correction

A specific offset, gain and shift can be applied to SDADC by performing the following operation:

```
(RESULT + OFFSETCORR)*GAINCORR/2^SHIFTCORR
```


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.16. Shift Correction

**Name:** SHIFTCORR
**Offset:** 0x1A
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SHIFTCORR[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – SHIFTCORR[3:0]** Shift Correction
A specific offset, gain and shift can be applied to SDADC by performing the following operation:
`(RESULT + OFFSETCORR)*GAINCORR/2^SHIFTCORR`


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.17. Software Trigger

**Name:** SWTRIG
**Offset:** 0x1C
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | START | FLUSH |
| Access | | | | | | | W | W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – START** SDADC Start Conversion
Writing a one to this bit will start a conversion or sequence. The bit is cleared by hardware when the conversion has started. Setting this bit when it is already set has no effect.
Writing this bit to zero will have no effect.

**Bit 0 – FLUSH** SDADC Conversion Flush
Writing a one to this bit will be flush the SDADC pipeline. A flush will restart the SDADC conversion and all conversions in progress will be aborted and lost. This bit is cleared until the SDADC has been flushed.
After the flush, the ADC will resume where it left off; i.e., if a conversion was pending, the ADC will start a new conversion.
Writing this bit to zero will have no effect.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.18. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ANACTRL | SWTRIG | SHIFTCORR | GAINCORR |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETCORR | WINUT | WINLT | WINCTRL | MUXCTRL | CTRLC | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 11 – ANACTRL** Analog Control Synchronization Busy
This bit is cleared when the synchronization of ANACTRL register between the clock domains is complete.
This bit is set when the synchronization of ANACTRL register between clock domains is started.

**Bit 10 – SWTRIG** Software Trigger Synchronization Busy
This bit is cleared when the synchronization of SWTRIG register between the clock domains is complete.
This bit is set when the synchronization of SWTRIG register between clock domains is started.

**Bit 9 – SHIFTCORR** Shift Correction Synchronization Busy
This bit is cleared when the synchronization of SHIFTCORR register between the clock domains is complete.
This bit is set when the synchronization of SHIFTCORR register between clock domains is started.

**Bit 8 – GAINCORR** Gain Correction Synchronization Busy
This bit is cleared when the synchronization of GAINCORR register between the clock domains is complete.
This bit is set when the synchronization of GAINCORR register between clock domains is started.

**Bit 7 – OFFSETCORR** Offset Correction Synchronization Busy
This bit is cleared when the synchronization of OFFSETCORR register between the clock domains is complete.
This bit is set when the synchronization of OFFSETCORR register between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

**Bit 6 – WINUT** Window Monitor Lower Threshold Synchronization Busy
This bit is cleared when the synchronization of WINUT register between the clock domains is complete.
This bit is set when the synchronization of WINUT register between clock domains is started.

**Bit 5 – WINLT** Window Monitor Upper Threshold Synchronization Busy
This bit is cleared when the synchronization of WINLT register between the clock domains is complete.
This bit is set when the synchronization of WINLT register between clock domains is started.

**Bit 4 – WINCTRL** Window Monitor Control Synchronization Busy
This bit is cleared when the synchronization of WINCTRL register between the clock domains is complete.
This bit is set when the synchronization of WINCTRL register between clock domains is started.

**Bit 3 – MUXCTRL** Mux Control Synchronization Busy
This bit is cleared when the synchronization of MUXCTRL register between the clock domains is complete.
This bit is set when the synchronization of MUXCTRL register between clock domains is started.

**Bit 2 – CTRLC** Control C Synchronization Busy
This bit is cleared when the synchronization of CTRLC register between the clock domains is complete.
This bit is set when the synchronization of CTRLC register between clock domains is started.

**Bit 1 – ENABLE** ENABLE Synchronization Busy
This bit is cleared when the synchronization of ENABLE register between the clock domains is complete.
This bit is set when the synchronization of ENABLE register between clock domains is started.

**Bit 0 – SWRST** SWRST Synchronization Busy
This bit is cleared when the synchronization of SWRST register between the clock domains is complete.
This bit is set when the synchronization of SWRST register between clock domains is started


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.19. Result

**Name:** RESULT
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RESULT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – RESULT[23:0]** Result Conversion Value

The analog-to-digital conversion data is placed into this register at the end of a conversion and remains until a new conversion is completed.
The RESULT is a signed integer value with 24-bit size. The SDADC conversion result is left-adjusted in the RESULT register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.20. Sequence Control

**Name:** SEQCTRL
**Offset:** 0x28
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SEQENn[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bits 2:0 – SEQENn[2:0]** Enable Positive Input in the Sequence
For details on available mux selections, refer to INPUTCTRL.
The sequence start from the lowest input, and go to the next enabled input automatically when the conversion is done. If no bits are set the sequence is disabled.

| Value | Description |
|---|---|
| 0 | Disable the positive input mux n selection from the sequence. |
| 1 | Enable the positive input mux n selection to the sequence. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.21. Analog Control

**Name:** ANACTRL
**Offset:** 0x2C
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BUFTEST | ONCHOP | | CTLSDADC[4:0] | | | | |
| Access | R/W | R/W | | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | | 0 | 0 | 0 | 0 | 0 |

**Bit 7 – BUFTEST** Buffer Test

**Bit 6 – ONCHOP** ONCHOP

| Value | Description |
|---|---|
| 0 | No Chopper at SDADC input |
| 1 | Chopper at SDADC input |

**Bits 4:0 – CTLSDADC[4:0]** CTLSDADC
SDADC Bias Current Control and used for Debug/Characterization


© 2026 Microchip Technology Inc. and its subsidiaries

### 39.8.22. Debug Control

**Name:** DBGCTRL
**Offset:** 0x2E
**Reset:** 0x00
**Property:** PAC Write-Protectedion

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bit controls the functionality when the CPU is halted by an external debugger.
This bit should be written only while a conversion is not ongoing.

| Value | Description |
|---|---|
| 0 | The SDADC is halted when the CPU is halted by an external debugger. |
| 1 | The SDADC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 40. AC – Analog Comparators

## 40.1. Overview

The Analog Comparator (AC) supports multiple individual comparators. Each comparator (COMP) compares the voltage levels on two inputs, and provides a digital output based on this comparison. Each comparator may be configured to generate interrupt requests and/or peripheral events upon several different combinations of input change.

Hysteresis and propagation delay can be adjusted to achieve optimal operation for each application.

The input selection includes four shared analog port pins (only two, AIN[5:4], for CMP2 and CMP3 on E and G variants) and several internal signals. Each Comparator Output state can also be output on a pin for use by external devices.

The comparators are grouped in pairs on each port. The AC peripheral implements two pairs of comparators . These are called Comparator 0 (COMP0) and Comparator 1 (COMP1) for the first pair and Comparator 2 (COMP2) and Comparator 3 (COMP3) for the second pair. They have identical behaviors, but separate Control registers. Each pair can be set in Window mode to compare a signal to a voltage range instead of a single voltage level.

## 40.2. Features

- Four individual comparators
- Selectable propagation delay versus current consumption
- Hysteresis: On or Off
- Analog comparator outputs available on pins
  - Asynchronous or synchronous
- Flexible input selection:
  - Four pins selectable for positive or negative inputs
  - Ground (for zero crossing)
  - INTREF reference voltage, supplied by the bandgap
  - 64-level programmable VDD scaler per comparator
  - DAC (if available)
- Interrupt generation on:
  - Rising or falling edge
  - Toggle
  - End of comparison
- Window function interrupt generation on:
  - Signal above window
  - Signal inside window
  - Signal below window
  - Signal outside window
- Event generation on:
  - Comparator output
  - Window function inside/outside window
- Optional digital filter on comparator output
- Low-power option


© 2026 Microchip Technology Inc. and its subsidiaries

- Single-shot support

## 40.3. Block Diagram

**Figure 40-1.** Analog Comparator Block Diagram (First Pair)

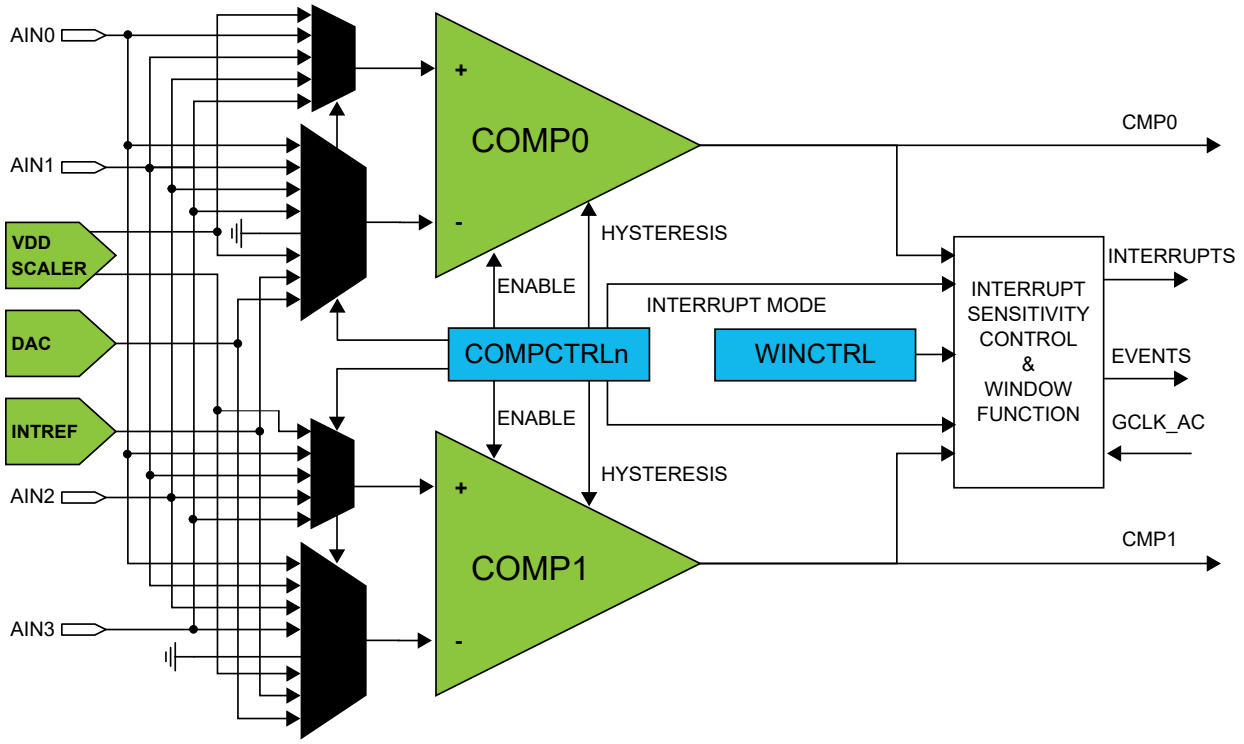
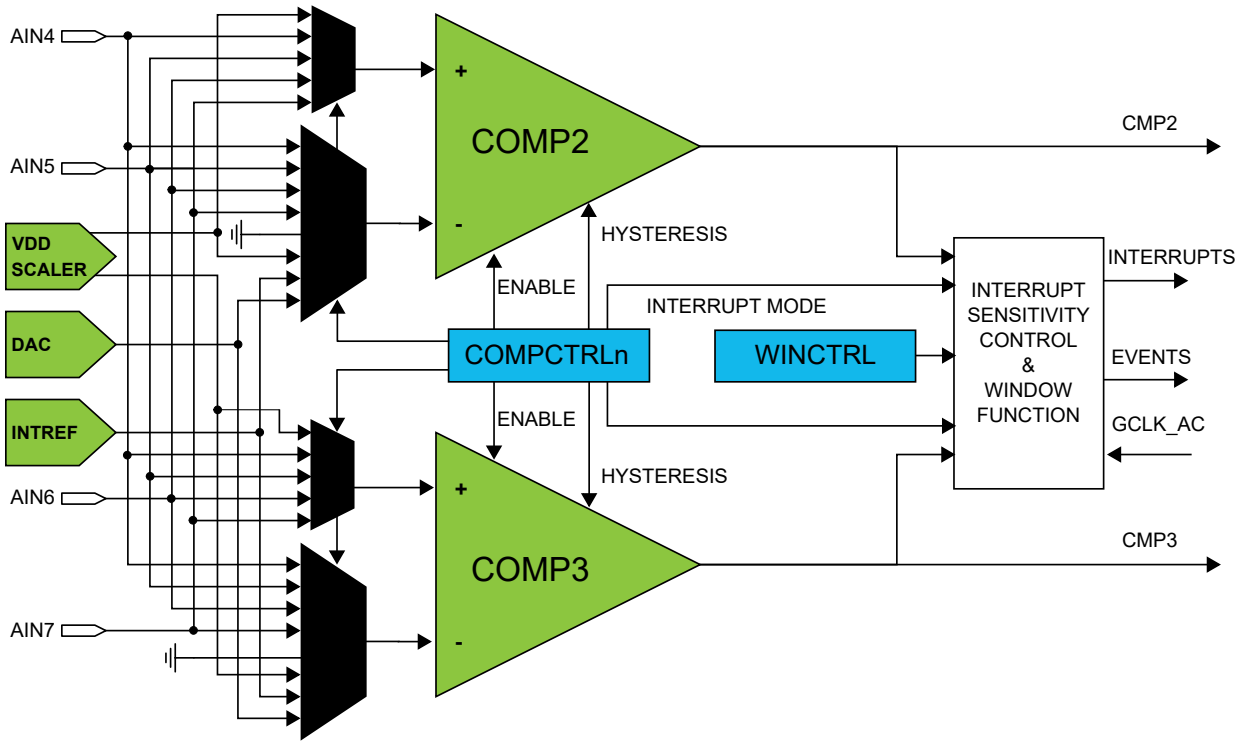



**Figure 40-2.** Analog Comparator Block Diagram (Second Pair)





© 2026 Microchip Technology Inc. and its subsidiaries

## 40.4. Signal Description

| Signal | Description | Type |
|---|---|---|
| AIN[7..0] | Analog input | Comparator inputs |
| CMP[3..0] | Digital output | Comparator outputs |

Refer to *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**
I/O Multiplexing and Considerations

## 40.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 40.5.1. I/O Lines

Using the AC's I/O lines requires the I/O pins to be configured. Refer to *PORT - I/O Pin Controller* for details.

**Related Links**
PORT - I/O Pin Controller

### 40.5.2. Power Management

The AC will continue to operate in any Sleep mode where the selected source clock is running. The AC's interrupts can be used to wake up the device from Sleep modes. Events connected to the Event System can trigger other operations in the system without exiting Sleep modes.

**Related Links**
PM - Power Manager

### 40.5.3. Clocks

The AC bus clock (CLK_AC_APB) can be enabled and disabled in the Main Clock module, MCLK (see *MCLK - Main Clock*), and the default state of CLK_AC_APB can be found in *Peripheral Clock Masking*.

A generic clock (GCLK_AC) is required to clock the AC. This clock must be configured and enabled in the generic clock controller before using the AC. Refer to the Generic Clock Controller chapter for details.

This generic clock is asynchronous to the bus clock (CLK_AC_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
PM - Power Manager

### 40.5.4. DMA

Not applicable.

### 40.5.5. Interrupts

The interrupt request lines are connected to the interrupt controller. Using the AC interrupts requires the interrupt controller to be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**
Nested Vector Interrupt Controller


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.5.6. Events

The events are connected to the Event System. Refer to *EVSYS – Event System* for details on how to configure the Event System.

**Related Links**
Event System (EVSYS)

### 40.5.7. Debug Operation

When the CPU is halted in debug mode, the AC will halt normal operation after any on-going comparison is completed. The AC can be forced to continue normal operation during debugging. Refer to DBGCTRL for details. If the AC is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 40.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except for the following registers:

- Control B register (CTRLB)
- Interrupt Flag register (INTFLAG)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

PAC write-protection does not apply to accesses through an external debugger.

**Related Links**
PAC - Peripheral Access Controller

### 40.5.9. Analog Connections

Each comparator has up to four I/O pins that can be used as analog inputs. Each pair of comparators shares the same four pins. These pins must be configured for analog operation before using them as comparator inputs.

Any internal reference source, such as a INTREF voltage reference, supplied by the bandgap, or DAC must be configured and enabled prior to its use as a comparator input.

## 40.6. Functional Description

### 40.6.1. Principle of Operation

Each comparator has one positive input and one negative input. Each positive input may be chosen from a selection of analog input pins. Each negative input may be chosen from a selection of both analog input pins and internal inputs, such as INTREF voltage reference.

The digital output from the comparator is '1' when the difference between the positive and the negative input voltage is positive, and '0' otherwise.

The individual comparators can be used independently (Normal mode) or paired to form a window comparison (Window mode).

### 40.6.2. Basic Operation

#### 40.6.2.1. Initialization

Some registers are enable-protected, meaning they can only be written when the module is disabled.

The following register is enable-protected:


© 2026 Microchip Technology Inc. and its subsidiaries

- Event Control register (EVCTRL)

Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

#### 40.6.2.2. Enabling, Disabling and Resetting

The AC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The AC is disabled writing a '0' to CTRLA.ENABLE.

The AC is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the AC will be reset to their initial state, and the AC will be disabled. Refer to *CTRLA* for details.

#### 40.6.2.3. Comparator Configuration

Each individual comparator must be configured by its respective Comparator Control register (COMPCTRLx) before that comparator is enabled. These settings cannot be changed while the comparator is enabled.

- Select the desired measurement mode with COMPCTRLx.SINGLE. See Starting a Comparison for more details.
- Select the desired hysteresis with COMPCTRLx.HYSTEN. See Input Hysteresis for more details.
- Select the comparator speed versus power with COMPCTRLx.SPEED. See Propagation Delay vs. Power Consumption for more details.
- Select the interrupt source with COMPCTRLx.INTSEL.
- Select the positive and negative input sources with the COMPCTRLx.MUXPOS and COMPCTRLx.MUXNEG bits. See Selecting Comparator Inputs for more details.
- Select the filtering option with COMPCTRLx.FLEN.
- Select standby operation with Run in Standby bit (COMPCTRLx.RUNSTDBY).

The individual comparators are enabled by writing a '1' to the Enable bit in the Comparator x Control registers (COMPCTRLx.ENABLE). The individual comparators are disabled by writing a '0' to COMPCTRLx.ENABLE. Writing a '0' to CTRLA.ENABLE will also disable all the comparators, but will not clear their COMPCTRLx.ENABLE bits.

#### 40.6.2.4. Starting a Comparison

Each comparator channel can be in one of two different measurement modes, determined by the COMPCTRLx.SINGLE bit:

- Continuous measurement
- Single-shot

After being enabled, a start-up delay is required before the result of the comparison is ready. This start-up time is measured automatically to account for environmental changes, such as temperature or voltage supply level, and is specified in the *Electrical Characteristics* chapters. During the start-up time, the COMP output is not available.

The comparator can be configured to generate interrupts when the output toggles, when the output changes from '0' to '1' (rising edge), when the output changes from '1' to '0' (falling edge) or at the end of the comparison. An end-of-comparison interrupt can be used with the Single-Shot mode to chain further events in the system, regardless of the state of the comparator outputs. The Interrupt mode is set by the Interrupt Selection bit group in the Comparator Control register (COMPCTRLx.INTSEL). Events are generated using the comparator output state, regardless of whether the interrupt is enabled or not.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)


© 2026 Microchip Technology Inc. and its subsidiaries

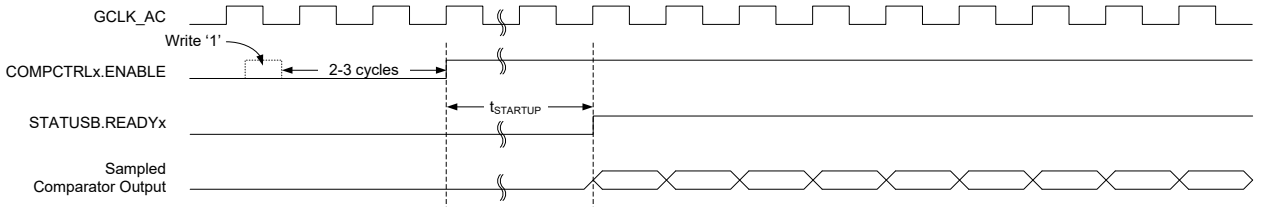
#### 40.6.2.4.1. Continuous Measurement

Continuous measurement is selected by writing COMPCTRLx.SINGLE to zero. In continuous mode, the comparator is continuously enabled and performing comparisons. This ensures that the result of the latest comparison is always available in the Current State bit in the Status A register (STATUSA.STATEx).

After the start-up time has passed, a comparison is done and STATUSA is updated. The Comparator x Ready bit in the Status B register (STATUSB.READYx) is set, and the appropriate peripheral events and interrupts are also generated. New comparisons are performed continuously until the COMPCTRLx.ENABLE bit is written to zero. The start-up time applies only to the first comparison.

In continuous operation, edge detection of the comparator output for interrupts is done by comparing the current and previous sample. The sampling rate is the GCLK_AC frequency. An example of continuous measurement is shown in the Figure 40-3.

**Figure 40-3.** Continuous Measurement Example




For low-power operation, comparisons can be performed during sleep modes without a clock. The comparator is enabled continuously, and changes of the comparator state are detected asynchronously. When a toggle occurs, the Power Manager will start GCLK_AC to register the appropriate peripheral events and interrupts. The GCLK_AC clock is then disabled again automatically, unless configured to wake up the system from sleep.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

#### 40.6.2.4.2. Single-Shot

Single-shot operation is selected by writing COMPCTRLx.SINGLE to '1'. During single-shot operation, the comparator is normally idle. The user starts a single comparison by writing '1' to the respective Start Comparison bit in the write-only Control B register (CTRLB.STARTx). The comparator is enabled, and after the start-up time has passed, a single comparison is done and STATUSA.STATEx is updated. Appropriate peripheral events and interrupts are also generated. No new comparisons will be performed.

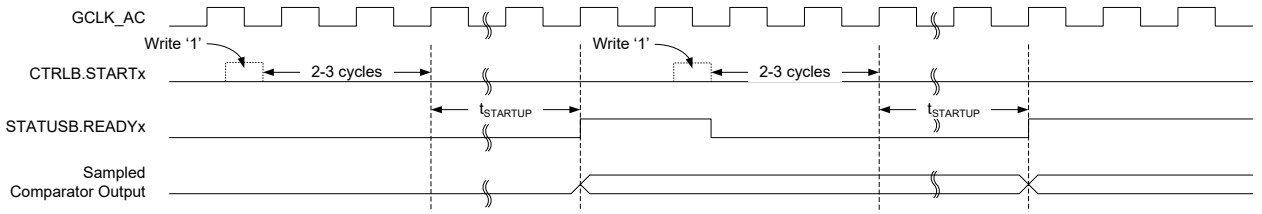
Writing '1' to CTRLB.STARTx also clears the Comparator x Ready bit in the Status B register (STATUSB.READYx). STATUSB.READYx is set automatically by hardware after the single comparison has completed.

A single-shot measurement can also be triggered by the Event System. Setting the Comparator x Event Input bit in the Event Control Register (EVCTRL.COMPEIx) enables triggering on incoming peripheral events. Each comparator can be triggered independently by separate events. Event-triggered operation is similar to user-triggered operation; the difference is that a peripheral event from another hardware module causes the hardware to automatically start the comparison and clear STATUSB.READYx.

To detect an edge of the comparator output in single-shot operation for the purpose of interrupts, the result of the current measurement is compared with the result of the previous measurement (one sampling period earlier). An example of single-shot operation is shown in Figure 40-4.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 40-4.** Single-Shot Example




For low-power operation, event-triggered measurements can be performed during sleep modes. When the event occurs, the Power Manager will start GCLK_AC. The comparator is enabled, and after the startup time has passed, a comparison is done and appropriate peripheral events and interrupts are also generated. The comparator and GCLK_AC are then disabled again automatically, unless configured to wake up the system from sleep.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

### 40.6.3. Selecting Comparator Inputs

Each comparator has one positive and one negative input. The positive input is one of the external input pins (AINx). The negative input can be fed either from an external input pin (AINx) or from one of the several internal reference voltage sources common to all comparators. The user selects the input source as follows:

- The positive input is selected by the Positive Input MUX Select bit group in the Comparator Control register (COMPCTRLx.MUXPOS)
- The negative input is selected by the Negative Input MUX Select bit group in the Comparator Control register (COMPCTRLx.MUXNEG)

In the case of using an external I/O pin, the selected pin must be configured for analog use in the PORT Controller by disabling the digital input and output. The switching of the analog input multiplexers is controlled to minimize crosstalk between the channels. The input selection must be changed only while the individual comparator is disabled.

**Note:** For internal use of the comparison results by the CCL, this bit must be 0x1 or 0x2.

### 40.6.4. Window Operation

Each comparator pair can be configured to work together in Window mode. In this mode, a voltage range is defined, and the comparators give information about whether an input signal is within this range or not. Window mode is enabled by the Window Enable x bit in the Window Control register (WINCTRL.WENx). Both comparators in a pair must have the same measurement mode setting in their respective Comparator Control Registers (COMPCTRLx.SINGLE).

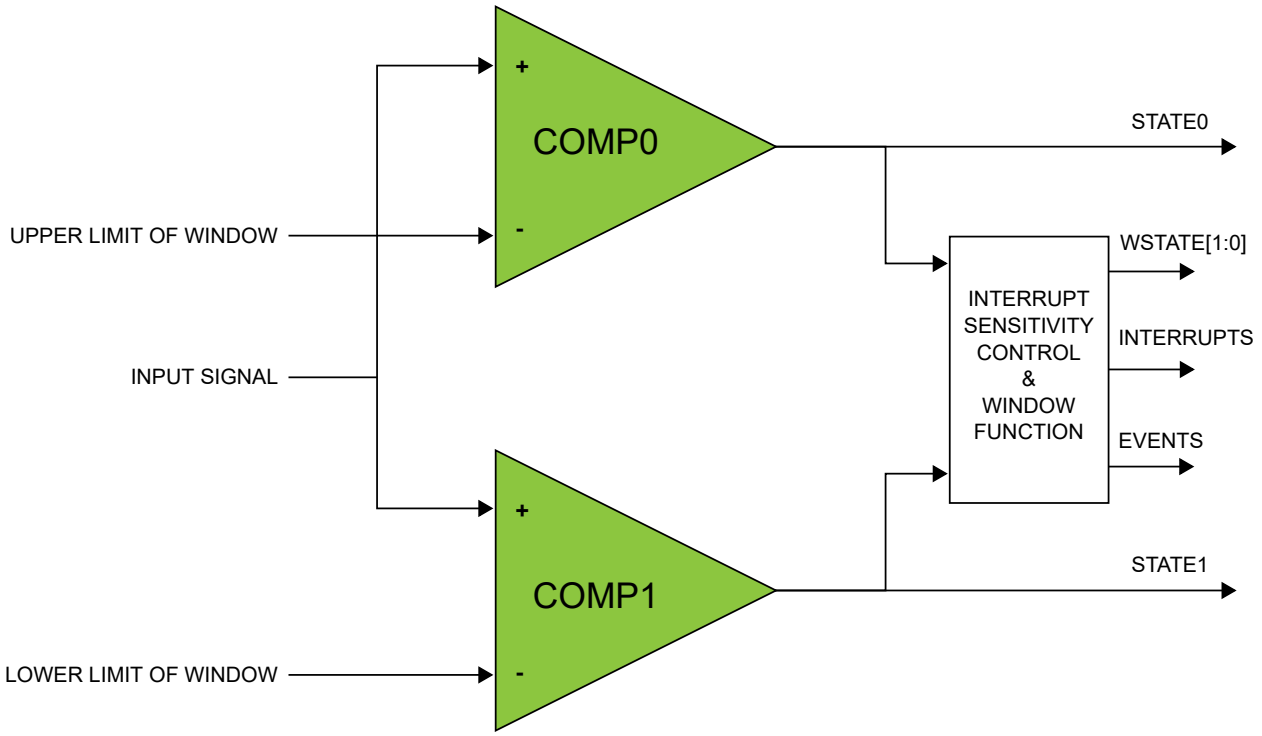
To physically configure the pair of comparators for Window mode, the same I/O pin must be chosen as positive input for each comparator, providing a shared input signal. The negative inputs define the range for the window. In Figure 40-5, COMP0 defines the upper limit and COMP1 defines the lower limit of the window, as shown but the window will also work in the opposite configuration with COMP0 lower and COMP1 higher. The current state of the window function is available in the Window x State bit group of the Status register (STATUSA.WSTATEx).

Window mode can be configured to generate interrupts when the input voltage reaches values below the window, above the window, inside the window or outside of the window. The interrupt selections are set by the Window Interrupt Selection bit field in the Window Control register (WINCTRL.WINTSEL). Events are generated using the inside/outside state of the window, regardless of whether the interrupt is enabled or not. Note that the individual comparator outputs, interrupts and events continue to function normally during Window mode.


© 2026 Microchip Technology Inc. and its subsidiaries

When the comparators are configured for Window mode and Single-shot mode, measurements are performed simultaneously on both comparators. Writing '1' to either Start Comparison bit in the Control B register (CTRLB.STARTx) will start a measurement. Likewise either peripheral event can start a measurement.

**Figure 40-5.** Comparators in Window Mode

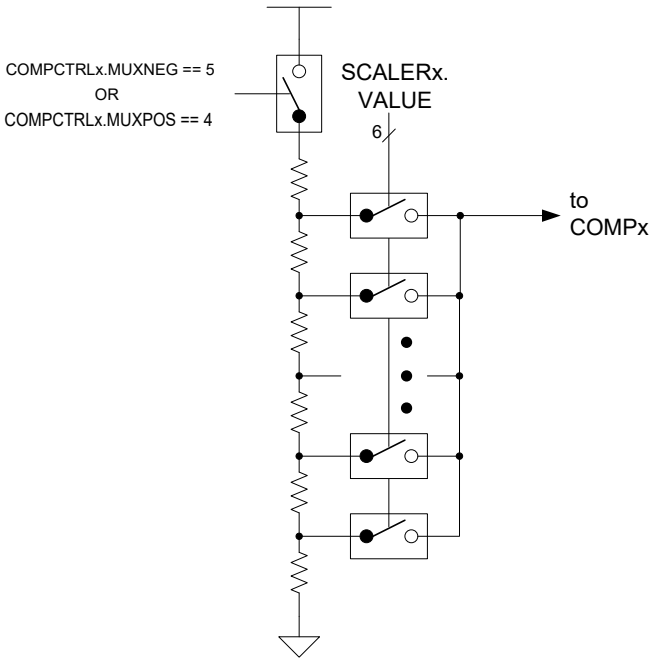



### 40.6.5. VDD Scaler

The VDD scaler generates a reference voltage that is a fraction of the device's supply voltage, with 64 levels. One independent voltage channel is dedicated for each comparator. The scaler of a comparator is enabled when the Negative Input Mux bit field or the Positive Input Mux in the respective Comparator Control register (COMPCTRLx.MUXNEG, or COMPCTRLx.MUXPOS) is set to 0x5 for Negative Input or 0x04 for Positive Input and the comparator is enabled. The voltage of each channel is selected by the Value bit field in the SCALERx registers (SCALERx.VALUE).


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 40-6.** VDD Scaler




### 40.6.6. Input Hysteresis

Application software can selectively enable/disable hysteresis for the comparison. Applying hysteresis will help prevent constant toggling of the output, which can be caused by noise when the input signals are close to each other.

Hysteresis is enabled for each comparator individually by the Hysteresis Enable bit in the Comparator x Control register (COMPCTRLx.HYSTEN). Hysteresis is available only in continuous mode (COMPCTRLx.SINGLE=0).

### 40.6.7. Propagation Delay vs. Power Consumption

It is possible to trade off comparison speed for power efficiency to get the shortest possible propagation delay or the lowest power consumption. The speed setting is configured for each comparator individually by the Speed bit group in the Comparator x Control register (COMPCTRLx.SPEED). The Speed bits select the amount of bias current provided to the comparator, and as such will also affect the start-up time.

### 40.6.8. Filtering

The output of the comparators can be filtered digitally to reduce noise. The filtering is determined by the Filter Length bits in the Comparator Control x register (COMPCTRLx.FLEN), and is independent for each comparator. Filtering is selectable from none, 3-bit majority (N=3) or 5-bit majority (N=5) functions. Any change in the comparator output is considered valid only if N/2+1 out of the last N samples agree. The filter sampling rate is the GCLK_AC frequency.

Note that filtering creates an additional delay of N-1 sampling cycles from when a comparison is started until the comparator output is validated. For continuous mode, the first valid output will occur when the required number of filter samples is taken. Subsequent outputs will be generated every cycle based on the current sample plus the previous N-1 samples, as shown in Figure 40-7. For single-shot mode, the comparison completes after the Nth filter sample, as shown in Figure 40-8.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 40-7.** Continuous Mode Filtering

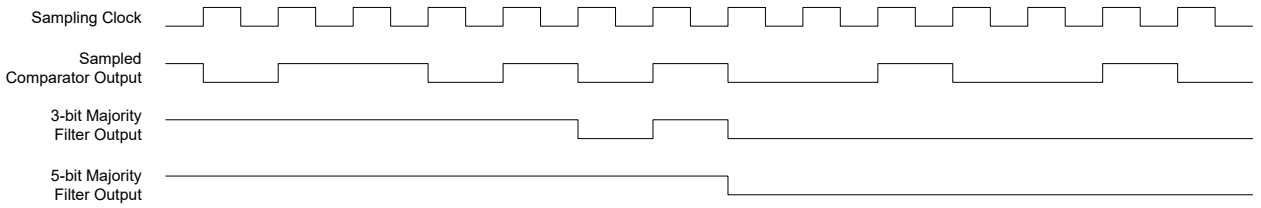
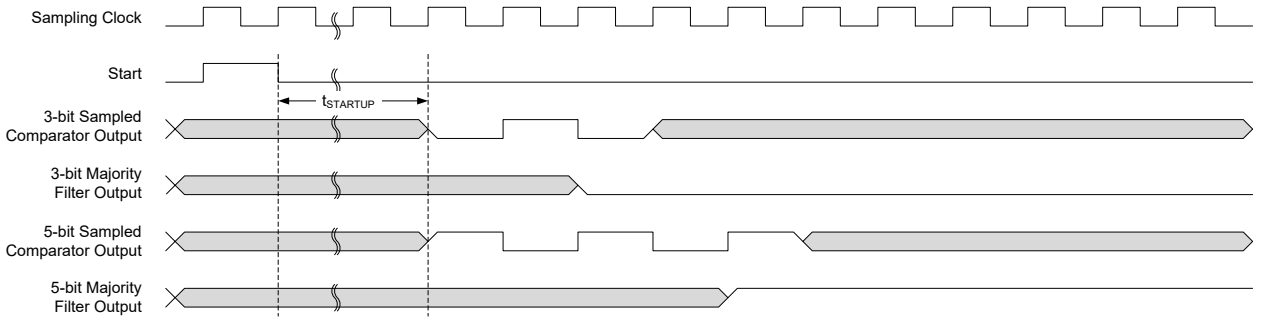



**Figure 40-8.** Single-Shot Filtering




During sleep modes, filtering is supported only for single-shot measurements. Filtering must be disabled if continuous measurements will be done during sleep modes, or the resulting interrupt/event may be generated incorrectly.

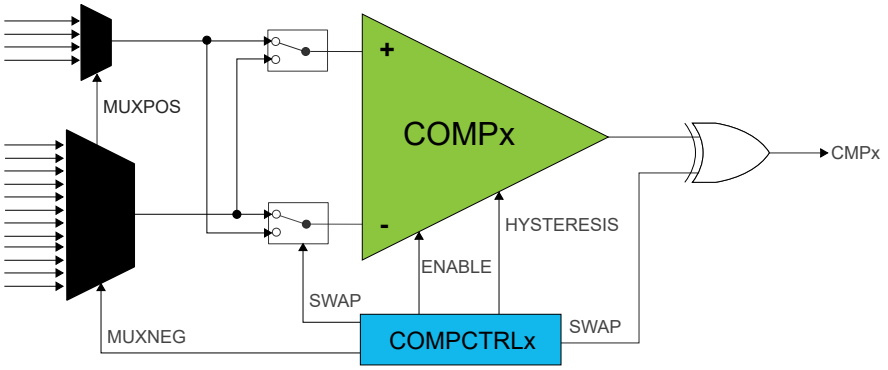
### 40.6.9. Comparator Output

The output of each comparator can be routed to an I/O pin by setting the Output bit group in the Comparator Control x register (COMPCTRLx.OUT). This allows the comparator to be used by external circuitry. Either the raw, non-synchronized output of the comparator or the CLK_AC-synchronized version, including filtering, can be used as the I/O signal source. The output appears on the corresponding CMP[x] pin.

### 40.6.10. Offset Compensation

The Swap bit in the Comparator Control registers (COMPCTRLx.SWAP) controls switching of the input signals to a comparator's positive and negative terminals. When the comparator terminals are swapped, the output signal from the comparator is also inverted, as shown in Figure 40-9. This allows the user to measure or compensate for the comparator input offset voltage. As part of the input selection, COMPCTRLx.SWAP can be changed only while the comparator is disabled.

**Figure 40-9.** Input Swapping for Offset Compensation




**Note:** The inputs are swapped only if the comparator inputs difference is within offset values.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.6.11. DMA Operation

Not applicable.

### 40.6.12. Interrupts

The AC has the following interrupt sources:

- Comparator (COMP0, COMP1, COMP2, COMP3): Indicates a change in comparator status.
- Window (WIN0, WIN1): Indicates a change in the window status.

Comparator interrupts are generated based on the conditions selected by the Interrupt Selection bit group in the Comparator Control registers (COMPCTRLx.INTSEL). Window interrupts are generated based on the conditions selected by the Window Interrupt Selection bit group in the Window Control register (WINCTRL.WINTSELx[1:0]).

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the AC is Reset. See INTFLAG register for details on how to clear interrupt flags. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**

Nested Vector Interrupt Controller

### 40.6.13. Events

The AC can generate the following output events:

- Comparator (COMP0, COMP1, COMP2, COMP3): Generated as a copy of the comparator status
- Window (WIN0, WIN1): Generated as a copy of the window inside/outside status

Writing a one to an Event Output bit in the Event Control Register (EVCTRL.xxEO) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. Refer to the Event System chapter for details on configuring the event system.

The AC can take the following action on an input event:

- Start comparison (START0, START1, START2, START3): Start a comparison.

Writing a one to an Event Input bit into the Event Control register (EVCTRL.COMPEIx) enables the corresponding action on input event. Writing a zero to this bit disables the corresponding action on input event. Note that if several events are connected to the AC, the enabled action will be taken on any of the incoming events. Refer to the Event System chapter for details on configuring the event system.

When EVCTRL.COMPEIx is one, the event will start a comparison on COMPx after the start-up time delay. In normal mode, each comparator responds to its corresponding input event independently. For a pair of comparators in window mode, either comparator event will trigger a comparison on both comparators simultaneously.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.6.14. Sleep Mode Operation

The Run in Standby bits in the Comparator x Control registers (COMPCTRLx.RUNSTDBY) control the behavior of the AC during standby sleep mode. Each RUNSTDBY bit controls one comparator. When the bit is zero, the comparator is disabled during sleep, but maintains its current configuration. When the bit is one, the comparator continues to operate during sleep. Note that when RUNSTDBY is zero, the analog blocks are powered off for the lowest power consumption. This necessitates a start-up time delay when the system returns from sleep.

For Window Mode operation, both comparators in a pair must have the same RUNSTDBY configuration.

When RUNSTDBY is one, any enabled AC interrupt source can wake up the CPU. The AC can also be used during sleep modes where the clock used by the AC is disabled, provided that the AC is still powered (not in shutdown). In this case, the behavior is slightly different and depends on the measurement mode, as listed in Table 40-1.

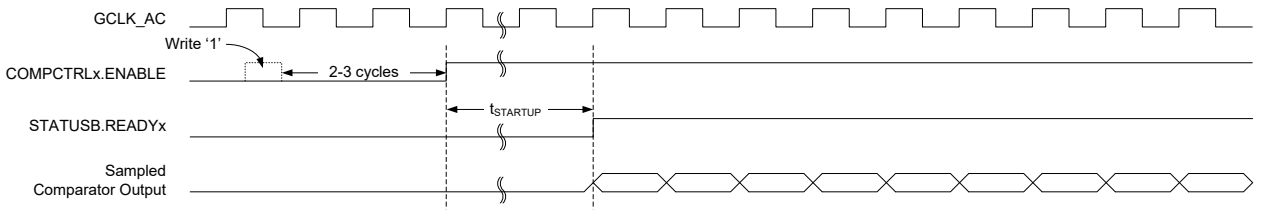
**Table 40-1.** Sleep Mode Operation

| COMPCTRLx.MODE | RUNSTDBY=0 | RUNSTDBY=1 |
|---|---|---|
| 0 (Continuous) | COMPx disabled | GCLK_AC stopped, COMPx enabled |
| 1 (Single-shot) | COMPx disabled | GCLK_AC stopped, COMPx enabled only when triggered by an input event |

#### 40.6.14.1. Continuous Measurement during Sleep

When a comparator is enabled in continuous measurement mode and GCLK_AC is disabled during sleep, the comparator will remain continuously enabled and will function asynchronously. The current state of the comparator is asynchronously monitored for changes. If an edge matching the interrupt condition is found, GCLK_AC is started to register the interrupt condition and generate events. If the interrupt is enabled in the Interrupt Enable registers (INTENCLR/SET), the AC can wake up the device; otherwise GCLK_AC is disabled until the next edge detection. Filtering is not possible with this configuration.

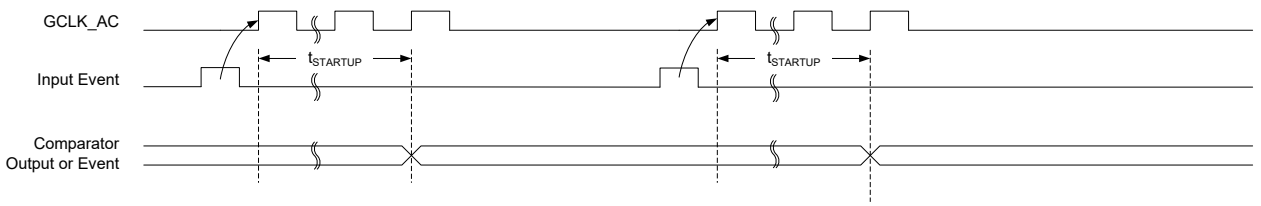
**Figure 40-10.** Continuous Mode SleepWalking




#### 40.6.14.2. Single-Shot Measurement during Sleep

For low-power operation, event-triggered measurements can be performed during sleep modes. When the event occurs, the Power Manager will start GCLK_AC. The comparator is enabled, and after the start-up time has passed, a comparison is done, with filtering if desired, and the appropriate peripheral events and interrupts are also generated, as shown in Figure 40-11. The comparator and GCLK_AC are then disabled again automatically, unless configured to wake the system from sleep. Filtering is allowed with this configuration.

**Figure 40-11.** Single-Shot SleepWalking





© 2026 Microchip Technology Inc. and its subsidiaries

### 40.6.15. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset bit in control register (CTRLA.SWRST)
- Enable bit in control register (CTRLA.ENABLE)
- Enable bit in Comparator Control register (COMPCTRLn.ENABLE)

The following registers are synchronized when written:

- Window Control register (WINCTRL)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

**Related Links**

Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 40.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | ENABLE | SWRST |
| 0x01 | CTRLB | 7:0 | | | | | START3 | START2 | START1 | START0 |
| 0x02 | EVCTRL | 7:0 | | | WINEO1 | WINEO0 | COMPEO3 | COMPEO2 | COMPEO1 | COMPEO0 |
| | | 15:8 | INVEI3 | INVEI2 | INVEI1 | INVEI0 | COMPEI3 | COMPEI2 | COMPEI1 | COMPEI0 |
| 0x04 | INTENCLR | 7:0 | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| 0x05 | INTENSET | 7:0 | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| 0x06 | INTFLAG | 7:0 | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| 0x07 | STATUSA | 7:0 | WSTATE1[1:0] | | WSTATE0[1:0] | | STATE3 | STATE2 | STATE1 | STATE0 |
| 0x08 | STATUSB | 7:0 | | | | | READY3 | READY2 | READY1 | READY0 |
| 0x09 | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0A | WINCTRL | 7:0 | | WINTSEL1[1:0] | | WEN1 | | WINTSEL0[1:0] | | WEN0 |
| 0x0B | Reserved | | | | | | | | | |
| 0x0C | SCALERn0 | 7:0 | | | VALUE[5:0] | | | | | |
| 0x0D | SCALERn1 | 7:0 | | | VALUE[5:0] | | | | | |
| 0x0E | SCALERn2 | 7:0 | | | VALUE[5:0] | | | | | |
| 0x0F | SCALERn3 | 7:0 | | | VALUE[5:0] | | | | | |
| 0x10 | COMPCTRL0 | 7:0 | | RUNSTDBY | | INTSEL[1:0] | | SINGLE | ENABLE | |
| | | 15:8 | SWAP | | MUXPOS[2:0] | | | | MUXNEG[2:0] | |
| | | 23:16 | | | | | HYSTEN | | SPEED[1:0] | |
| | | 31:24 | | | OUT[1:0] | | | | FLEN[2:0] | |
| 0x14 | COMPCTRL1 | 7:0 | | RUNSTDBY | | INTSEL[1:0] | | SINGLE | ENABLE | |
| | | 15:8 | SWAP | | MUXPOS[2:0] | | | | MUXNEG[2:0] | |
| | | 23:16 | | | | | HYSTEN | | SPEED[1:0] | |
| | | 31:24 | | | OUT[1:0] | | | | FLEN[2:0] | |
| 0x18 | COMPCTRL2 | 7:0 | | RUNSTDBY | | INTSEL[1:0] | | SINGLE | ENABLE | |
| | | 15:8 | SWAP | | MUXPOS[2:0] | | | | MUXNEG[2:0] | |
| | | 23:16 | | | | | HYSTEN | | SPEED[1:0] | |
| | | 31:24 | | | OUT[1:0] | | | | FLEN[2:0] | |
| 0x1C | COMPCTRL3 | 7:0 | | RUNSTDBY | | INTSEL[1:0] | | SINGLE | ENABLE | |
| | | 15:8 | SWAP | | MUXPOS[2:0] | | | | MUXNEG[2:0] | |
| | | 23:16 | | | | | HYSTEN | | SPEED[1:0] | |
| | | 31:24 | | | OUT[1:0] | | | | FLEN[2:0] | |
| 0x20 | SYNCBUSY | 7:0 | | COMPCTRL3 | COMPCTRL2 | COMPCTRL1 | COMPCTRL0 | WINCTRL | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 40.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R/W | W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** Enable

Due to synchronization, there is delay from updating the register until the peripheral is enabled/disabled. The value written to CTRL.ENABLE will read back immediately and the corresponding bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE is cleared when the peripheral is enabled/disabled.

| Value | Description |
|---|---|
| 0 | The AC is disabled. |
| 1 | The AC is enabled. Each comparator must also be enabled individually by the Enable bit in the Comparator Control register (COMPCTRLn.ENABLE). |

**Bit 0 – SWRST** Software Reset

Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the AC to their initial state, and the AC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.2. Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | START3 | START2 | START1 | START0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3 – STARTx** Comparator x Start Comparison
Writing a '0' to this field has no effect.
Writing a '1' to STARTx starts a single-shot comparison on COMPx if both the Single-Shot and Enable bits in the Comparator x Control Register are '1' (COMPCTRLx.SINGLE and COMPCTRLx.ENABLE). If comparator x is not enabled in single-shot mode, writing a '1' has no effect.
This bit always reads as zero.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.3. Event Control

**Name:** EVCTRL
**Offset:** 0x02
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | INVEI3 | INVEI2 | INVEI1 | INVEI0 | COMPEI3 | COMPEI2 | COMPEI1 | COMPEI0 |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WINEO1 | WINEO0 | COMPEO3 | COMPEO2 | COMPEO1 | COMPEO0 |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 12, 13, 14, 15 – INVEIx** Inverted Event Input Enable x

| Value | Description |
|---|---|
| 0 | Incoming event is not inverted for comparator x. |
| 1 | Incoming event is inverted for comparator x. |

**Bits 8, 9, 10, 11 – COMPEIx** Comparator x Event Input
Note that several actions can be enabled for incoming events. If several events are connected to the peripheral, the enabled action will be taken for any of the incoming events. There is no way to tell which of the incoming events caused the action.
These bits indicate whether a comparison will start or not on any incoming event.

| Value | Description |
|---|---|
| 0 | Comparison will not start on any incoming event. |
| 1 | Comparison will start on any incoming event. |

**Bits 4, 5 – WINEOx** Window x Event Output Enable
These bits indicate whether the window x function can generate a peripheral event or not.

| Value | Description |
|---|---|
| 0 | Window x Event is disabled. |
| 1 | Window x Event is enabled. |

**Bits 0, 1, 2, 3 – COMPEOx** Comparator x Event Output Enable
These bits indicate whether the comparator x output can generate a peripheral event or not.

| Value | Description |
|---|---|
| 0 | COMPx event generation is disabled. |
| 1 | COMPx event generation is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 4, 5 – WINx** Window x Interrupt Enable
Reading this bit returns the state of the Window x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Window x interrupt.

| Value | Description |
|---|---|
| 0 | The Window x interrupt is disabled. |
| 1 | The Window x interrupt is enabled. |

**Bits 0, 1, 2, 3 – COMPx** Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit disables the Comparator x interrupt.

| Value | Description |
|---|---|
| 0 | The Comparator x interrupt is disabled. |
| 1 | The Comparator x interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 4, 5 – WINx** Window x Interrupt Enable
Reading this bit returns the state of the Window x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit enables the Window x interrupt.

| Value | Description |
|---|---|
| 0 | The Window x interrupt is disabled. |
| 1 | The Window x interrupt is enabled. |

**Bits 0, 1, 2, 3 – COMPx** Comparator x Interrupt Enable
Reading this bit returns the state of the Comparator x interrupt enable.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Ready interrupt bit and enable the Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Comparator x interrupt is disabled. |
| 1 | The Comparator x interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x06
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | WIN1 | WIN0 | COMP3 | COMP2 | COMP1 | COMP0 |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 4, 5 – WINx** Window x
This flag is set according to the Window x Interrupt Selection bit group in the WINCTRL register (WINCTRL.WINTSELx) and will generate an interrupt if INTENCLR/SET.WINx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Window x interrupt flag.

**Bits 0, 1, 2, 3 – COMPx** Comparator x
Reading this bit returns the status of the Comparator x interrupt flag. If comparator x is not implemented, COMPx always reads as zero.
This flag is set according to the Interrupt Selection bit group in the Comparator x Control register (COMPCTRLx.INTSEL) and will generate an interrupt if INTENCLR/SET.COMPx is also one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Comparator x interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.7. Status A

**Name:** STATUSA
**Offset:** 0x07
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WSTATE1[1:0] | | WSTATE0[1:0] | | STATE3 | STATE2 | STATE1 | STATE0 |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:6 – WSTATE1[1:0]** Window 1 Current State
These bits show the current state of the signal if the window 1 mode is enabled.

| Value | Name | Description |
|---|---|---|
| 0x0 | ABOVE | Signal is above window |
| 0x1 | INSIDE | Signal is inside window |
| 0x2 | BELOW | Signal is below window |
| 0x3 | | Reserved |

**Bits 5:4 – WSTATE0[1:0]** Window 0 Current State
These bits show the current state of the signal if the window 0 mode is enabled.

| Value | Name | Description |
|---|---|---|
| 0x0 | ABOVE | Signal is above window |
| 0x1 | INSIDE | Signal is inside window |
| 0x2 | BELOW | Signal is below window |
| 0x3 | | Reserved |

**Bits 0, 1, 2, 3 – STATEx** Comparator x Current State
This bit shows the current state of the output signal from COMPx. STATEx is valid only when STATUSB.READYx is one.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.8. Status B

**Name:** STATUSB
**Offset:** 0x08
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | READY3 | READY2 | READY1 | READY0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 0, 1, 2, 3 – READYx** Comparator x Ready
This bit is cleared when the comparator x output is not ready.
This bit is set when the comparator x output is ready.
If comparator x is not implemented, READYx always reads as zero.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.9. Debug Control

**Name:** DBGCTRL
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bits controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The AC is halted when the CPU is halted by an external debugger. Any on-going comparison will complete. |
| 1 | The AC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.10. Window Control

**Name:** WINCTRL
**Offset:** 0x0A
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | WINTSEL1[1:0] | | WEN1 | | WINTSEL0[1:0] | | WEN0 |
| Access | | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bits 6:5 – WINTSEL1[1:0]** Window 1 Interrupt Selection
These bits configure the interrupt mode for the comparator window 1 mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | ABOVE | Interrupt on signal above window |
| 0x1 | INSIDE | Interrupt on signal inside window |
| 0x2 | BELOW | Interrupt on signal below window |
| 0x3 | OUTSIDE | Interrupt on signal outside window |

**Bit 4 – WEN1** Window 1 Mode Enable

| Value | Description |
|---|---|
| 0 | Window mode is disabled for comparators 2 and 3. |
| 1 | Window mode is enabled for comparators 2 and 3. |

**Bits 2:1 – WINTSEL0[1:0]** Window 0 Interrupt Selection
These bits configure the interrupt mode for the comparator window 0 mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | ABOVE | Interrupt on signal above window |
| 0x1 | INSIDE | Interrupt on signal inside window |
| 0x2 | BELOW | Interrupt on signal below window |
| 0x3 | OUTSIDE | Interrupt on signal outside window |

**Bit 0 – WEN0** Window 0 Mode Enable

| Value | Description |
|---|---|
| 0 | Window mode is disabled for comparators 0 and 1. |
| 1 | Window mode is enabled for comparators 0 and 1. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.11. Scaler n

**Name:** SCALERn
**Offset:** 0x0C + n*0x01 [n=0..3]
**Reset:** 0x00
**Property:** Write-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | VALUE[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 5:0 – VALUE[5:0]** Scaler Value
These bits define the scaling factor for channel n of the $V_{DD}$ voltage scaler. The output voltage, $V_{SCALE}$, is:

$$V_{SCALE} = \frac{V_{DD} \cdot (VALUE+1)}{64}$$


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.12. Comparator Control n

**Name:** COMPCTRL
**Offset:** 0x10 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | OUT[1:0] | | | FLEN[2:0] | | |
| Access | | | R/W | R/W | | R/W | R/W | R/W |
| Reset | | | 0 | 0 | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | HYSTEN | | SPEED[1:0] | |
| Access | | | | | R/W | | R/W | R/W |
| Reset | | | | | 0 | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SWAP | MUXPOS[2:0] | | | | MUXNEG[2:0] | | |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | INTSEL[1:0] | | SINGLE | ENABLE | |
| Access | | R/W | | R/W | R/W | R/W | R/W | |
| Reset | | 0 | | 0 | 0 | 0 | 0 | |

**Bits 29:28 – OUT[1:0]** Output
These bits configure the output selection for comparator n. COMPCTRLn.OUT can be written only while COMPCTRLn.ENABLE is zero.
**Note:** For internal use of the comparison results by the CCL, this bit must be 0x1 or 0x2.

These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | The output of COMPn is not routed to the COMPn I/O port |
| 0x1 | ASYNC | The asynchronous output of COMPn is routed to the COMPn I/O port |
| 0x2 | SYNC | The synchronous output (including filtering) of COMPn is routed to the COMPn I/O port |
| 0x3 | N/A | Reserved |

**Bits 26:24 – FLEN[2:0]** Filter Length
These bits configure the filtering for comparator n. COMPCTRLn.FLEN can only be written while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | No filtering |
| 0x1 | MAJ3 | 3-bit majority function (2 of 3) |
| 0x2 | MAJ5 | 5-bit majority function (3 of 5) |
| 0x3-0x7 | N/A | Reserved |

**Bit 19 – HYSTEN** Hysteresis Enable
This bit indicates the hysteresis mode of comparator n. Hysteresis is available only for continuous mode (COMPCTRLn.SINGLE=0).
This bit is not synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Hysteresis is disabled. |
| 1 | Hysteresis is enabled. |

**Bits 17:16 – SPEED[1:0]** Speed Selection

This bit indicates the speed/propagation delay mode of comparator n. COMPCTRLn.SPEED can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | LOW | Low speed |
| 0x3 | HIGH | High speed |

**Bit 15 – SWAP** Swap Inputs and Invert

This bit swaps the positive and negative inputs to COMPn and inverts the output. This function can be used for offset cancellation. COMPCTRLn.SWAP can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | The output of MUXPOS connects to the positive input, and the output of MUXNEG connects to the negative input. |
| 1 | The output of MUXNEG connects to the positive input, and the output of MUXPOS connects to the negative input. |

**Bits 14:12 – MUXPOS[2:0]** Positive Input Mux Selection

These bits select which input will be connected to the positive input of comparator n. COMPCTRLn.MUXPOS can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | PIN0 | AIN[0] for COMP0/1, AIN[4] for COMP2/3 |
| 0x1 | PIN1 | AIN[1] for COMP0/1, AIN[5] for COMP2/3 |
| 0x2 | PIN2 | AIN[2] for COMP0/1, AIN[6] for COMP2/3 |
| 0x3 | PIN3 | AIN[3] for COMP0/1, AIN[7] for COMP2/3 |
| 0x4 | VSCALE | VDD scaler |
| 0x5-0x7 | | Reserved |

**Bits 10:8 – MUXNEG[2:0]** Negative Input Mux Selection

These bits select which input will be connected to the negative input of comparator n. COMPCTRLn.MUXNEG can only be written while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | PIN0 | AIN[0] for COMP0/1, AIN[4] for COMP2/3 |
| 0x1 | PIN1 | AIN[1] for COMP0/1, AIN[5] for COMP2/3 |
| 0x2 | PIN2 | AIN[2] for COMP0/1, AIN[6] for COMP2/3 |
| 0x3 | PIN3 | AIN[3] for COMP0/1, AIN[7] for COMP2/3 |
| 0x4 | GND | Ground |
| 0x5 | VSCALE | VDD scaler |
| 0x6 | INTREF | Internal voltage reference, supplied by the bandgap (refer to SUPC.VREF.SEL for voltage level information) |
| 0x7 | DAC | DAC output |

**Bit 6 – RUNSTDBY** Run in Standby

This bit controls the behavior of the comparator during standby sleep mode.
This bit is not synchronized


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | The comparator is disabled during sleep. |
| 1 | The comparator continues to operate during sleep. |

**Bits 4:3 – INTSEL[1:0]** Interrupt Selection

These bits select the condition for comparator n to generate an interrupt or event. COMPCTRLn.INTSEL can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Name | Description |
|---|---|---|
| 0x0 | TOGGLE | Interrupt on comparator output toggle |
| 0x1 | RISING | Interrupt on comparator output rising |
| 0x2 | FALLING | Interrupt on comparator output falling |
| 0x3 | EOC | Interrupt on end of comparison (single-shot mode only) |

**Bit 2 – SINGLE** Single-Shot Mode

This bit determines the operation of comparator n. COMPCTRLn.SINGLE can be written only while COMPCTRLn.ENABLE is zero.
These bits are not synchronized.

| Value | Description |
|---|---|
| 0 | Comparator n operates in continuous measurement mode. |
| 1 | Comparator n operates in single-shot mode. |

**Bit 1 – ENABLE** Enable

Writing a zero to this bit disables comparator n.
Writing a one to this bit enables comparator n.
Due to synchronization, there is delay from updating the register until the comparator is enabled/disabled. The value written to COMPCTRLn.ENABLE will read back immediately after being written. SYNCBUSY.COMPCTRLn is set. SYNCBUSY.COMPCTRLn is cleared when the peripheral is enabled/disabled.
Writing a one to COMPCTRLn.ENABLE will prevent further changes to the other bits in COMPCTRLn. These bits remain protected until COMPCTRLn.ENABLE is written to zero and the write is synchronized.


© 2026 Microchip Technology Inc. and its subsidiaries

### 40.8.13. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | COMPCTRL3 | COMPCTRL2 | COMPCTRL1 | COMPCTRL0 | WINCTRL | ENABLE | SWRST |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 3, 4, 5, 6 – COMPCTRLx** COMPCTRLx Synchronization Busy
This bit is cleared when the synchronization of the COMPCTRLx register between the clock domains is complete.
This bit is set when the synchronization of the COMPCTRLx register between clock domains is started.

**Bit 2 – WINCTRL** WINCTRL Synchronization Busy
This bit is cleared when the synchronization of the WINCTRL register between the clock domains is complete.
This bit is set when the synchronization of the WINCTRL register between clock domains is started.

**Bit 1 – ENABLE** Enable Synchronization Busy
This bit is cleared when the synchronization of the CTRLA.ENABLE bit between the clock domains is complete.
This bit is set when the synchronization of the CTRLA.ENABLE bit between clock domains is started.

**Bit 0 – SWRST** Software Reset Synchronization Busy
This bit is cleared when the synchronization of the CTRLA.SWRST bit between the clock domains is complete.
This bit is set when the synchronization of the CTRLA.SWRST bit between clock domains is started.


© 2026 Microchip Technology Inc. and its subsidiaries

# 41. DAC – Digital-to-Analog Converter (SAM C21 only)

## 41.1. Overview

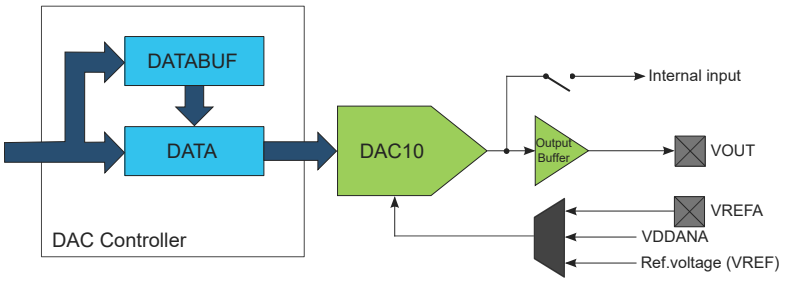
The Digital-to-Analog Converter (DAC) converts a digital value to a voltage. The DAC has one channel with 10-bit resolution, and it is capable of converting up to 350,000 samples per second (350ksps).

## 41.2. Features

- DAC with 10-bit resolution
- Up to 350ksps conversion rate
- Hardware support for 14-bit using dithering
- Multiple trigger sources
- High-drive capabilities
- Output can be used as input to the Analog Comparator (AC), ADC or SDADC
- DMA support

## 41.3. Block Diagram

**Figure 41-1.** DAC Block Diagram




## 41.4. Signal Description

| Signal Name | Type | Description |
|---|---|---|
| VOUT | Analog output | DAC output |
| VREFA | Analog input | External reference |

**Related Links**

I/O Multiplexing and Considerations

## 41.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 41.5.1. I/O Lines

Using the DAC Controller's I/O lines requires the I/O pins to be configured using the port configuration (PORT).

**Related Links**

PORT - I/O Pin Controller

### 41.5.2. Power Management

The DAC will continue to operate in any Sleep mode where the selected source clock is running.


© 2026 Microchip Technology Inc. and its subsidiaries

The DAC interrupts can be used to wake up the device from sleep modes.

Events connected to the event system can trigger other operations in the system without exiting sleep modes.

**Related Links**
PM - Power Manager

### 41.5.3. Clocks

The DAC bus clock (CLK_DAC_APB) can be enabled and disabled by the Main Clock module, and the default state of CLK_DAC_APB can be found in the *Peripheral Clock Masking* section.

A generic clock (GCLK_DAC) is required to clock the DAC Controller. This clock must be configured and enabled in the Generic Clock Controller before using the DAC Controller. Refer to *GCLK – Generic Clock Controller* for details.

This generic clock is asynchronous to the bus clock (CLK_DAC_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for further details.

**Related Links**
GCLK - Generic Clock Controller

### 41.5.4. DMA

The DMA request line is connected to the DMA Controller (DMAC). Using the DAC Controller DMA requests requires to configure the DMAC first.

**Related Links**
DMAC – Direct Memory Access Controller

### 41.5.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the DAC Controller interrupt(s) requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 41.5.6. Events

The events are connected to the Event System.

**Related Links**
Event System (EVSYS)

### 41.5.7. Debug Operation

When the CPU is halted in debug mode the DAC will halt normal operation. Any on-going conversions will be completed. The DAC can be forced to continue normal operation during debugging. If the DAC is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

### 41.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except the following registers:

- Interrupt Flag Status and Clear (INTFLAG) register
- Data Buffer (DATABUF) register

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

PAC write-protection does not apply to accesses through an external debugger

**Related Links**
PAC - Peripheral Access Controller

### 41.5.9. Analog Connections

The DAC has one output pin (VOUT) and one analog input voltage reference pin (VREFA) that must be configured first.

When internal input is used, it must be enabled before DAC controller is enabled.

## 41.6. Functional Description

### 41.6.1. Principle of Operation

The DAC converts the digital value located in the Data register (DATA) into an analog voltage on the DAC output (VOUT).

A conversion can be started two different ways:

- When a new data is written in the DATA register.
- By a START input event from the Event System. The data previously written in DATABUF is then copied in DATA and the conversion started.

The resulting voltage is available on the DAC output after the conversion time.

### 41.6.2. Basic Operation

#### 41.6.2.1. Initialization

The following registers are enable-protected, meaning they can only be written when the DAC is disabled (CTRLA.ENABLE is zero):

- Control B register (CTRLB)
- Event Control register (EVCTRL)

Enable-protection is denoted by the Enable-Protected property in the register description.

Before enabling the DAC, it must be configured by selecting the voltage reference using the Reference Selection bits in the Control B register (CTRLB.REFSEL).

#### 41.6.2.2. Enabling, Disabling and Resetting

The DAC Controller is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The DAC Controller is disabled by writing a '0' to CTRLA.ENABLE.

The DAC Controller is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the DAC will be reset to their initial state, and the DAC Controller will be disabled. Refer to the CTRLA register for details.

#### 41.6.2.3. Enabling the Output Buffer

To enable the DAC output on the $V_{OUT}$ pin, the output driver must be enabled by writing a one to the External Output Enable bit in the Control B register (CTRLB.EOEN).

The DAC output buffer provides a high-drive-strength output, and is capable of driving both resistive and capacitive loads. To minimize power consumption, the output buffer should be enabled only when external output is needed.

#### 41.6.2.4. Digital to Analog Conversion

The DAC converts a digital value (stored in the DATA register) into an analog voltage. The conversion range is between GND and the selected DAC voltage reference. The default voltage reference is the internal reference voltage. Other voltage reference options are the analog supply voltage (VDDANA) and the external voltage reference (VREFA). The voltage reference is selected by writing to the Reference Selection bits in the Control B register (CTRLB.REFSEL).


© 2026 Microchip Technology Inc. and its subsidiaries

The output voltage from the DAC can be calculated using the following formula:

$$V_{OUT} = \frac{DATA}{0x3FF} \cdot VREF$$

A new conversion starts as soon as a new value is loaded into DATA. DATA can either be loaded via the APB bus during a CPU write operation, using DMA, or from the DATABUF register when a START event occurs. Refer to Events for details. As there is no automatic indication that a conversion is done, the sampling period must be greater than or equal to the specified conversion time.

### 41.6.3. DMA Operation

The DAC generates the following DMA request:

- Data Buffer Empty (EMPTY): The request is set when data is transferred from DATABUF to the internal data buffer of DAC. The request is cleared when DATABUF register is written, or by writing a one to the EMPTY bit in the Interrupt Flag register (INTFLAG.EMPTY).

For each Start Conversion event, DATABUF is transferred into DATA and the conversion starts. When DATABUF is empty, the DAC generates the DMA request for new data. As DATABUF is initially empty, a DMA request is generated whenever the DAC is enabled.

If the CPU accesses the registers that are the source of a DMA request set/clear condition, the DMA request can be lost or the DMA transfer can be corrupted, if enabled.

### 41.6.4. Interrupts

The DAC Controller has the following interrupt sources:

- Data Buffer Empty (EMPTY): Indicates that the internal data buffer of the DAC is empty.
- Underrun (UNDERRUN): Indicates that the internal data buffer of the DAC is empty and a DAC start of conversion event occurred. This interrupt can only occur when events are used to start a conversion. Refer to Events for details.

Each interrupt source has an Interrupt flag associated with it. The Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the Interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the Interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the Interrupt flag is cleared, the interrupt is disabled or the DAC is reset. See INTFLAG register for details on how to clear Interrupt flags.

All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which Interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated..

**Related Links**

Nested Vector Interrupt Controller

### 41.6.5. Events

The DAC Controller can generate the following output events:

- Data Buffer Empty (EMPTY): Generated when the internal data buffer of the DAC is empty. Refer to DMA Operation for details.

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.EMPTYEO) enables the corresponding output event. Writing a '0' to this bit disables the corresponding output event.


© 2026 Microchip Technology Inc. and its subsidiaries

The DAC can take the following action on an input event:

- Start Conversion (START): DATABUF value is transferred into DATA as soon as the DAC is ready for the next conversion, and then conversion is started. START is considered as asynchronous to GCLK_DAC thus it is resynchronized in DAC Controller. Refer to Digital to Analog Conversion for details.

Writing a '1' to an Event Input bit in the Event Control register (EVCTRL.STARTEI) enables the corresponding action on an input event. Writing a '0' to this bit disables the corresponding action on input event.

**Note:** When several events are connected to the DAC Controller, the enabled action will be taken on any of the incoming events.

By default, DAC Controller detects rising edge events. Falling edge detection can be enabled by writing a '1' to EVCTRL.INVEIx.

**Related Links**
Event System (EVSYS)

### 41.6.6. Sleep Mode Operation

The generic clock for the DAC is running in idle sleep mode. If the Run In Standby bit in the Control A register (CTRLA.RUNSTDBY) is one, the DAC output buffer will keep its value in standby sleep mode. If CTRLA.RUNSTDBY is zero, the DAC output buffer will be disabled in standby sleep mode.

### 41.6.7. Synchronization

Due to the asynchronicity between main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read. A register can require:

- Synchronization when written
- Synchronization when read
- Synchronization when written and read
- No synchronization

When executing an operation that requires synchronization, the corresponding status bit in the Synchronization Busy register (SYNCBUSY) will be set immediately, and cleared when synchronization is complete.

If an operation that requires synchronization is executed while its busy bit is one, the operation is discarded and an error is generated.

The following bits need synchronization when written:

- Software Reset bit in the Control A register (CTRLA.SWRST)
- Enable bit in the Control A register (CTRLA.ENABLE)
- All bits in the Data register (DATA)
- All bits in the Data Buffer register (DATABUF)

Write-synchronization is denoted by the Write-Synchronized property in the register description.

No bits need synchronization when read.

### 41.6.8. Additional Features

#### 41.6.8.1. DAC as an Internal Reference

The DAC output can be internally enabled as input to the analog comparator. This is enabled by writing a one to the Internal Output Enable bit in the Control B register (CTRLB.IOEN). It is possible to have the internal and external output enabled simultaneously.


© 2026 Microchip Technology Inc. and its subsidiaries

The DAC output can also be enabled as input to the Analog-to-Digital Converter. In this case, the output buffer must be enabled.

#### 41.6.8.2. Data Buffer

The Data Buffer register (DATABUF) and the Data register (DATA) are linked together to form a two-stage FIFO. The DAC uses the Start Conversion event to load data from DATABUF into DATA and start a new conversion. The Start Conversion event is enabled by writing a one to the Start Event Input bit in the Event Control register (EVCTRL.STARTEI). If a Start Conversion event occurs when DATABUF is empty, an Underrun interrupt request is generated if the Underrun interrupt is enabled.

The DAC can generate a Data Buffer Empty event when DATABUF becomes empty and new data can be loaded to the buffer. The Data Buffer Empty event is enabled by writing a one to the Empty Event Output bit in the Event Control register (EVCTRL.EMPTYEO). A Data Buffer Empty interrupt request is generated if the Data Buffer Empty interrupt is enabled.

#### 41.6.8.3. Voltage Pump

When the DAC is used at operating voltages lower than 2.5V, the voltage pump must be enabled. This enabling is done automatically, depending on operating voltage.

The voltage pump can be disabled by writing a one to the Voltage Pump Disable bit in the Control B register (CTRLB.VPD). This can be used to reduce power consumption when the operating voltage is above 2.5V.

The voltage pump uses the asynchronous GCLK_DAC clock, and requires that the clock frequency be at least four times higher than the sampling period.

#### 41.6.8.4. Dithering mode

Dithering is enabled by setting CTRLB.DITHER to 1. In dithering mode, DATA is a 14-bit unsigned value where DATA[13:4] is the 10-bit data converted by DAC and DATA[3:0] represents the dither bits, used for minimizing the quantization error. The principle is to make 16 sub-conversions of the DATA[13:4] value or the (DATA[13:4] + 1) value, so that by averaging those values, the conversion result of the 14-bit value (DATA[13:0]) has improved accuracy due to minimized quantization error.

To use the dithering feature, EVSYS is used for generating a periodic STARTEI. And the STARTEI event must be configured (EVCTRL.STARTEI = 1) to generate 16 events for each DATA[13:0] conversion, and DATABUFx must be loaded every 16 DAC conversions. EMPTYx event and DMA request are therefore generated every 16 DATABUF to DATA transfers. Using the DMA with dithering is optional. If the DMA is not used, it is required to poll the INTFLAG.EMTPY flag, or use an interrupt on EMPTY to add a new value in DATABUF.

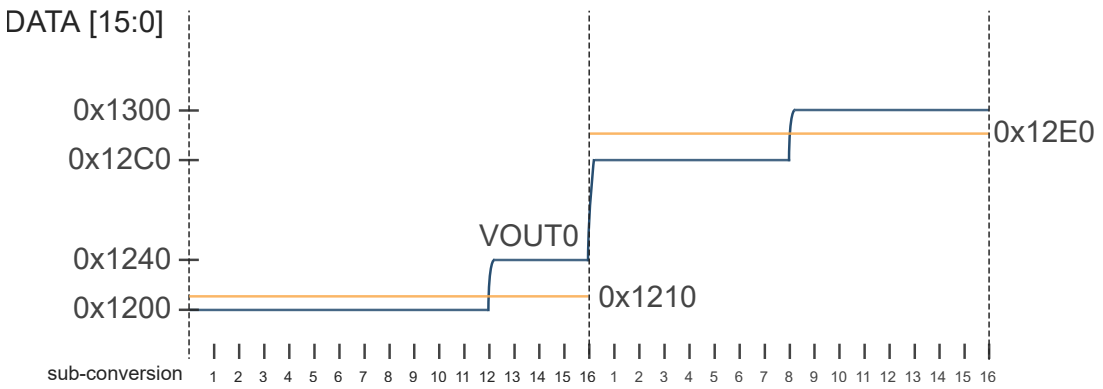
The input value for DAC is positioned in the DATA register, based on CTRLB.LEFTADJ as shown in the following figure. For additional information, refer to SEQSTATUS register. If LEFTADJ = 0: the user writes DATA[13:4], and the dithering function will take care of the DATA[3:0] bit during the 16 sub-conversions.

If LEFTADJ = 1: the user writes DATA[15:6], and the dithering function will take care of the DATA[5:2] bit during the 16 sub-conversions.

Following timing diagram shows a typical example with DATA[15:0] = 0x1210, DATA[15:0] = 0x12E0, and CTRLB.LEFTADJ = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 41-2.** DAC Conversions in Dithering Mode (CTRLB.LEFTADJ=1)





© 2026 Microchip Technology Inc. and its subsidiaries

## 41.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | RUNSTDBY | | | | | ENABLE | SWRST |
| 0x01 | CTRLB | 7:0 | REFSEL[1:0] | | DITHER | | VPD | LEFTADJ | IOEN | EOEN |
| 0x02 | EVCTRL | 7:0 | | | | | | INVEI | EMPTYEO | STARTEI |
| 0x03 | Reserved | | | | | | | | | |
| 0x04 | INTENCLR | 7:0 | | | | | | | EMPTY | UNDERRUN |
| 0x05 | INTENSET | 7:0 | | | | | | | EMPTY | UNDERRUN |
| 0x06 | INTFLAG | 7:0 | | | | | | | EMPTY | UNDERRUN |
| 0x07 | STATUS | 7:0 | | | | | | | | READY |
| 0x08 | DATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | DATA[15:8] | | | | | | | |
| 0x0A ... 0x0B | Reserved | | | | | | | | | |
| 0x0C | DATABUF | 7:0 | DATABUF[7:0] | | | | | | | |
| | | 15:8 | DATABUF[15:8] | | | | | | | |
| 0x0E ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | | | | DATABUF | DATA | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 ... 0x17 | Reserved | | | | | | | | | |
| 0x18 | DBGCTRL | 7:0 | | | | | | | | DBGRUN |

## 41.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16- and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description. For details, refer to Register Access Protection.

Some registers are synchronized when read and/or written. Synchronization is denoted by the "Write-Synchronized" or the "Read-Synchronized" property in each individual register description. For details, refer to Synchronization.

Some registers are enable-protected, meaning they can only be written when the peripheral is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | | | | ENABLE | SWRST |
| Access | | R/W | | | | | R/W | R/W |
| Reset | | 0 | | | | | 0 | 0 |

**Bit 6 – RUNSTDBY** Run in Standby
This bit is not synchronized

| Value | Description |
|---|---|
| 0 | The DAC output buffer is disabled in standby sleep mode. |
| 1 | The DAC output buffer can be enabled in standby sleep mode. |

**Bit 1 – ENABLE** Enable DAC Controller
Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the corresponding bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled or being disabled. |
| 1 | The peripheral is enabled or being enabled. |

**Bit 0 – SWRST** Software Reset
Writing '0' to this bit has no effect.
Writing '1' to this bit resets all registers in the DAC to their initial state, and the DAC will be disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.2. Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | REFSEL[1:0] | | DITHER | | VPD | LEFTADJ | IOEN | EOEN |
| Access | R/W | R/W | R/W | | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | | 0 | 0 | 0 | 0 |

**Bits 7:6 – REFSEL[1:0]** Reference Selection
This bit field selects the Reference Voltage for the DAC.

| Value | Name | Description |
|---|---|---|
| 0x0 | INTREF | Internal voltage reference, supplied by the bandgap (refer to SUPC.VREF.SEL for voltage level information) |
| 0x1 | VDDANA | Analog voltage supply |
| 0x2 | VREFA | External reference |
| 0x3 | | Reserved |

**Bit 5 – DITHER** Dithering Mode
This bit controls dithering operation according to Dithering mode.

| Value | Description |
|---|---|
| 0 | Dithering mode is disabled. |
| 1 | Dithering mode is enabled. |

**Bit 3 – VPD** Voltage Pump Disabled
This bit controls the behavior of the voltage pump.

| Value | Description |
|---|---|
| 0 | Voltage pump is turned on/off automatically |
| 1 | Voltage pump is disabled. |

**Bit 2 – LEFTADJ** Left-Adjusted Data
This bit controls how the 10-bit conversion data is adjusted in the Data and Data Buffer registers.

| Value | Description |
|---|---|
| 0 | DATA and DATABUF registers are right-adjusted. |
| 1 | DATA and DATABUF registers are left-adjusted. |

**Bit 1 – IOEN** Internal Output Enable

| Value | Description |
|---|---|
| 0 | Internal DAC output not enabled. |
| 1 | Internal DAC output enabled to be used by the AC or ADC. |

**Bit 0 – EOEN** External Output Enable

| Value | Description |
|---|---|
| 0 | The DAC output is turned off. |
| 1 | The high-drive output buffer drives the DAC output to the $V_{OUT}$ pin. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.3. Event Control

**Name:** EVCTRL
**Offset:** 0x02
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | INVEI | EMPTYEO | STARTEI |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – INVEI** Enable Inversion of Start Event Input
This bit defines the edge detection of the input event for STARTEI.

| Value | Description |
|---|---|
| 0 | Rising edge. |
| 1 | Falling edge. |

**Bit 1 – EMPTYEO** Data Buffer Empty Event Output
This bit indicates whether or not the Data Buffer Empty event is enabled and will be generated when the Data Buffer register is empty.

| Value | Description |
|---|---|
| 0 | Data Buffer Empty event is disabled and will not be generated. |
| 1 | Data Buffer Empty event is enabled and will be generated. |

**Bit 0 – STARTEI** Start Conversion Event Input
This bit indicates whether or not the Start Conversion event is enabled and data are loaded from the Data Buffer register to the Data register upon event reception.

| Value | Description |
|---|---|
| 0 | A new conversion will not be triggered on any incoming event. |
| 1 | A new conversion will be triggered on any incoming event. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set register (INTENSET).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | EMPTY | UNDERRUN |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – EMPTY** Data Buffer Empty Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Data Buffer Empty Interrupt Enable bit, which disables the Data Buffer Empty interrupt.

| Value | Description |
|---|---|
| 0 | The Data Buffer Empty interrupt is disabled. |
| 1 | The Data Buffer Empty interrupt is enabled. |

**Bit 0 – UNDERRUN** Underrun Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Data Buffer Underrun Interrupt Enable bit, which disables the Data Buffer Underrun interrupt.

| Value | Description |
|---|---|
| 0 | The Data Buffer Underrun interrupt is disabled. |
| 1 | The Data Buffer Underrun interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear register (INTENCLR).

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | EMPTY | UNDERRUN |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – EMPTY** Data Buffer Empty Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Data Buffer Empty Interrupt Enable bit, which enables the Data Buffer Empty interrupt.

| Value | Description |
|---|---|
| 0 | The Data Buffer Empty interrupt is disabled. |
| 1 | The Data Buffer Empty interrupt is enabled. |

**Bit 0 – UNDERRUN** Underrun Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Data Buffer Underrun Interrupt Enable bit, which enables the Data Buffer Underrun interrupt.

| Value | Description |
|---|---|
| 0 | The Data Buffer Underrun interrupt is disabled. |
| 1 | The Data Buffer Underrun interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x06
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | EMPTY | UNDERRUN |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – EMPTY** Data Buffer Empty
This flag is cleared by writing a '1' to it or by writing new data to DATABUF.
This flag is set when data is transferred from DATABUF to DATA, and the DAC is ready to receive new data in DATABUF, and will generate an interrupt request if INTENCLR/SET.EMPTY is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Data Buffer Empty interrupt flag.

**Bit 0 – UNDERRUN** Underrun
This flag is cleared by writing a '1' to it.
This flag is set when a start conversion event occurs when DATABUF is empty, and will generate an interrupt request if INTENCLR/SET.UNDERRUN is one.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Underrun interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.7. Status

**Name:** STATUS
**Offset:** 0x07
**Reset:** 0x00
**Property:** -

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | READY |
| Access | | | | | | | | R |
| Reset | | | | | | | | 0 |

**Bit 0 – READY** DAC Ready

| Value | Description |
|---|---|
| 0 | DAC is not ready for conversion. |
| 1 | Startup time has elapsed, DAC is ready for conversion. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.8. Data DAC

**Name:** DATA
**Offset:** 0x08
**Reset:** 0x0000
**Property:** PAC Write-Protection, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DATA[15:8] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATA[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – DATA[15:0]** Data value to be converted

DATA register contains the 10-bit value that is converted to a voltage by the DAC. The adjustment of these 10 bits within the 16-bit register is controlled by CTRLB.LEFTADJ.
Four additional bits are also used for the dithering feature according to Dithering mode.

**Table 41-1.** Valid Data Bits

| CTRLB.DITHER | CTRLB.LEFTADJ | DATA | Description |
|---|---|---|---|
| 0 | 0 | DATA[9:0] | Right adjusted, 10-bits |
| 0 | 1 | DATA[15:6] | Left adjusted, 10-bits |
| 1 | 0 | DATA[13:4], DATA[3:0] | Right adjusted, 14-bits |
| 1 | 1 | DATA[15:6], DATA[5:2] | Left adjusted, 14-bits |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.9. Data Buffer

**Name:** DATABUF
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DATABUF[15:8] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DATABUF[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – DATABUF[15:0]** Data Buffer
DATABUF contains the value to be transferred into DATA register.


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.10. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DATABUF | DATA | ENABLE | SWRST |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – DATABUF** Data Buffer
This bit is set when DATABUF register is written.
This bit is cleared when DATABUF synchronization is completed.

| Value | Description |
|---|---|
| 0 | No ongoing synchronized access. |
| 1 | Synchronized access is ongoing. |

**Bit 2 – DATA** Data
This bit is set when DATA register is written.
This bit is cleared when DATA synchronization is completed.

| Value | Description |
|---|---|
| 0 | No ongoing synchronized access. |
| 1 | Synchronized access is ongoing. |

**Bit 1 – ENABLE** DAC Enable Status
This bit is set when CTRLA.ENABLE bit is written.
This bit is cleared when CTRLA.ENABLE synchronization is completed.

| Value | Description |
|---|---|
| 0 | No ongoing synchronization. |
| 1 | Synchronization is ongoing. |

**Bit 0 – SWRST** Software Reset
This bit is set when CTRLA.SWRST bit is written.
This bit is cleared when CTRLA.SWRST synchronization is completed.


© 2026 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | No ongoing synchronization. |
| 1 | Synchronization is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 41.8.11. Debug Control

**Name:** DBGCTRL
**Offset:** 0x18
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bits controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The DAC is halted when the CPU is halted by an external debugger. Any ongoing conversion will complete. |
| 1 | The DAC continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 42. Peripheral Touch Controller (PTC)

## 42.1. Overview

The Peripheral Touch Controller (PTC) acquires signals in order to detect a touch on the capacitive sensors. The external capacitive touch sensor is typically formed on a PCB, and the sensor electrodes are connected to the analog front end of the PTC through the I/O pins in the device. The PTC supports both self and mutual capacitance sensors.

In the Mutual Capacitance mode, sensing is done using capacitive touch matrices in various X-Y configurations, including indium tin oxide (ITO) sensor grids. The PTC requires one pin per X-line and one pin per Y-line.

In the Self Capacitance mode, the PTC requires only one pin (Y-line) for each touch sensor.

The number of available pins and the assignment of X- and Y-lines is depending on both package type and device configuration. Refer to the Configuration Summary and I/O Multiplexing table for details.

**Related Links**

Configuration Summary
I/O Multiplexing and Considerations

## 42.2. Features

- Low-Power, High-Sensitivity, Environmentally Robust Capacitive Touch Buttons, Sliders, and Wheels
- Supports Wake-up on Touch from standby Sleep mode
- Supports Mutual Capacitance and Self Capacitance Sensing
  - Mix-and-Match Mutual and Self Capacitance Sensors
- One Pin per Electrode – No External Components
- Load Compensating Charge Sensing
  - Parasitic capacitance compensation and adjustable gain for superior sensitivity
- Zero Drift Over the Temperature and $V_{DD}$ Range
  - Auto calibration and recalibration of sensors
- Single-shot Charge Measurement
- Hardware Noise Filtering and Noise Signal Desynchronization for High Conducted Immunity
- Selectable channel change delay allows choosing the settling time on a new channel, as required
- Acquisition-start triggered by command or through auto-triggering feature
- Low CPU utilization through interrupt on acquisition-complete

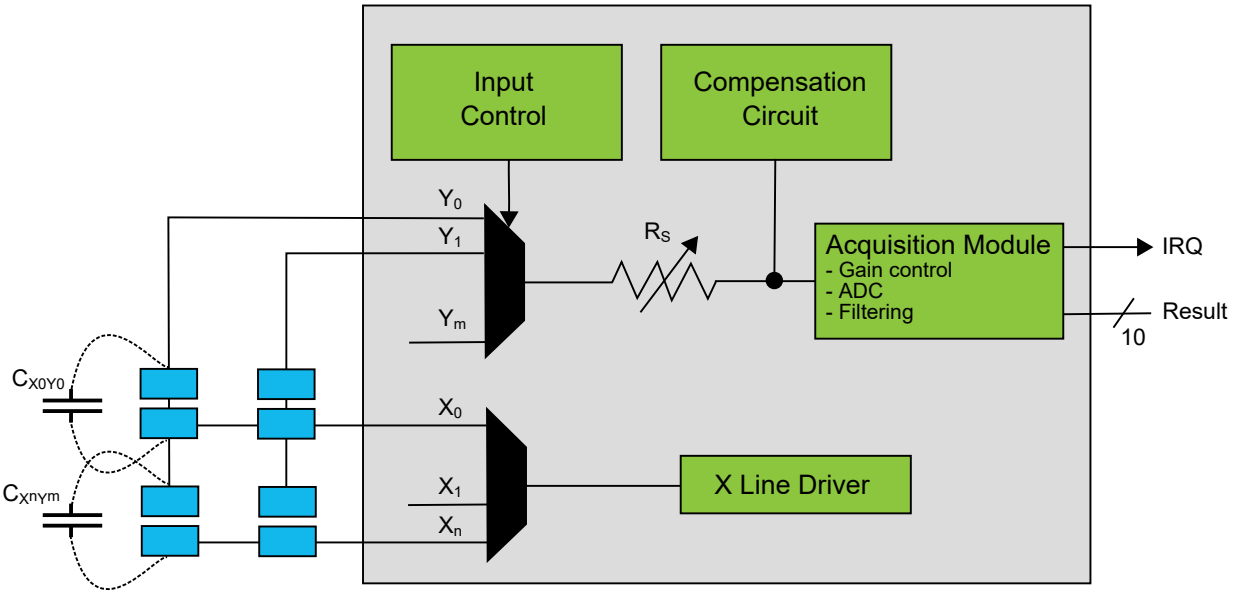
**Related Links**

Configuration Summary
I/O Multiplexing and Considerations


© 2026 Microchip Technology Inc. and its subsidiaries

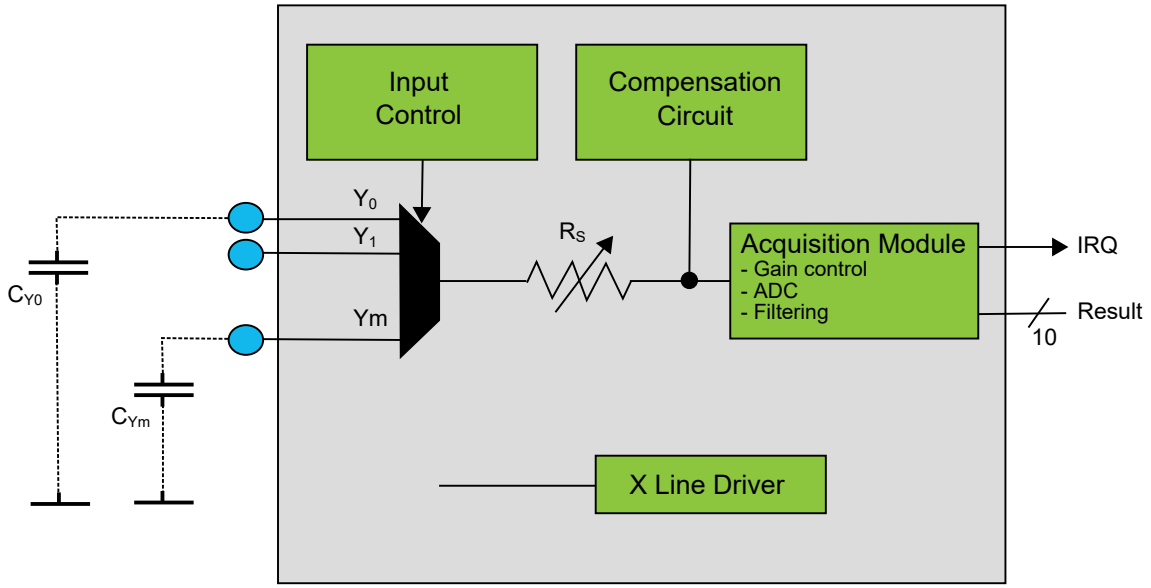
## 42.3. Block Diagram

**Figure 42-1.** PTC Block Diagram Mutual Capacitance




**Note:** For SAM C20/C21 the $R_S$ = 0, 20, 50, 100 KΩ.

**Figure 42-2.** PTC Block Diagram Self Capacitance




**Note:** For SAM C20/C21 the RS = 0, 20, 50, 100 KΩ.


© 2026 Microchip Technology Inc. and its subsidiaries

## 42.4. Signal Description

**Table 42-1.** Signal Description for PTC

| Name | Type | Description |
|---|---|---|
| Y[m:0] | Analog | Y-line (Input/Output) |
| X[n:0] | Digital | X-line (Output) |

**Note:** The number of X- and Y-lines are device dependent. Refer to *Configuration Summary* for details.

Refer to *I/O Multiplexing and Considerations* for details on the pin mapping for this peripheral. One signal can be mapped on several pins.

**Related Links**

Configuration Summary
I/O Multiplexing and Considerations

## 42.5. System Dependencies

In order to use this peripheral, configure the other components of the system as described in the following sections.

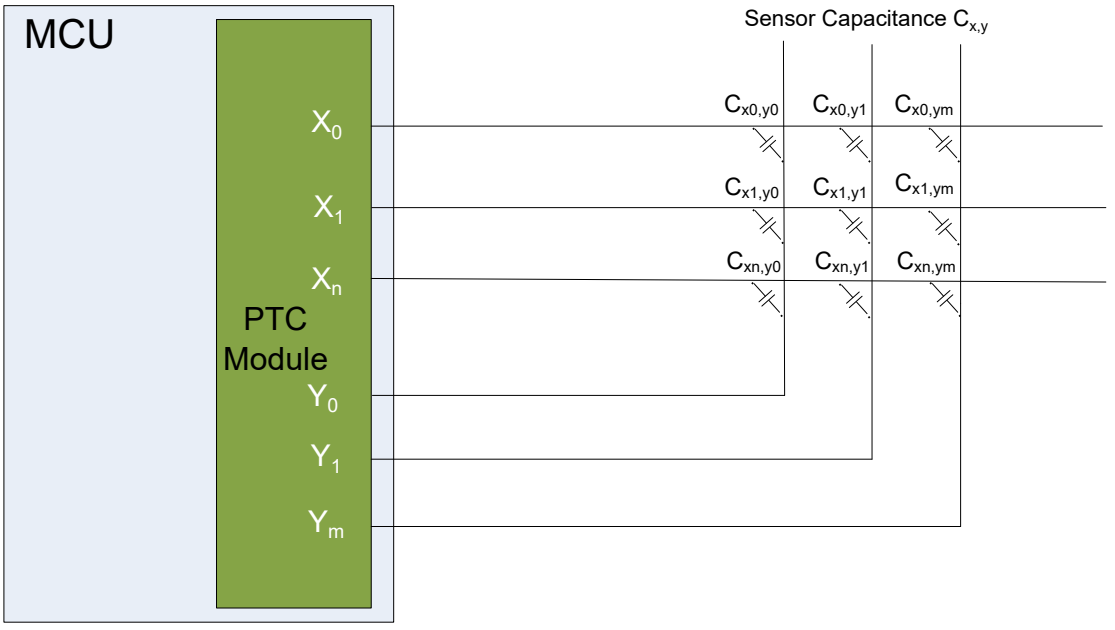
### 42.5.1. I/O Lines

The I/O lines used for analog X-lines and Y-lines must be connected to external capacitive touch sensor electrodes. External components are not required for normal operation. However, to improve the EMC performance, a series resistor of 1 kΩ or more can be used on X-lines and Y-lines.

#### 42.5.1.1. Mutual Capacitance Sensor Arrangement

A mutual capacitance sensor is formed between two I/O lines - an X electrode for transmitting and Y electrode for sensing. The mutual capacitance between the X and Y electrode is measured by the peripheral touch controller.

**Figure 42-3.** Mutual Capacitance Sensor Arrangement



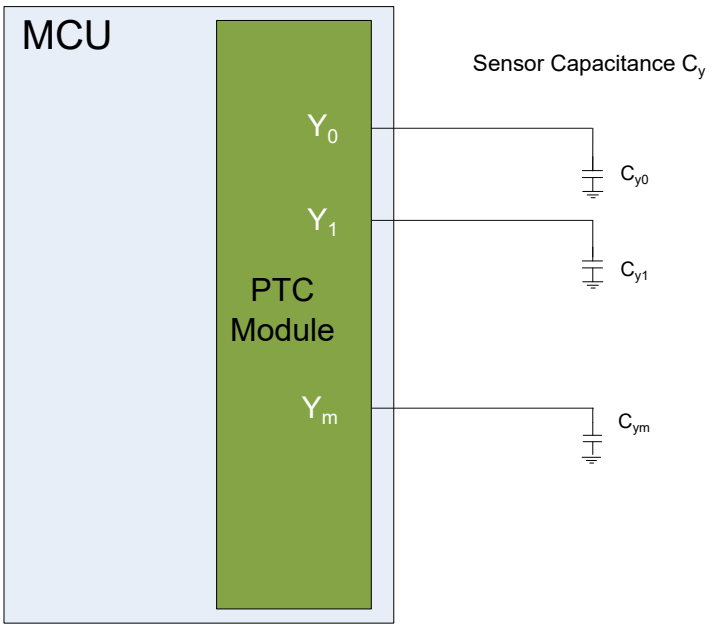


© 2026 Microchip Technology Inc. and its subsidiaries

#### 42.5.1.2. Self Capacitance Sensor Arrangement

A self capacitance sensor is connected to a single pin on the peripheral touch controller through the Y electrode for sensing the signal. The sense electrode capacitance is measured by the peripheral touch controller.

**Figure 42-4.** Self-Capacitance Sensor Arrangement

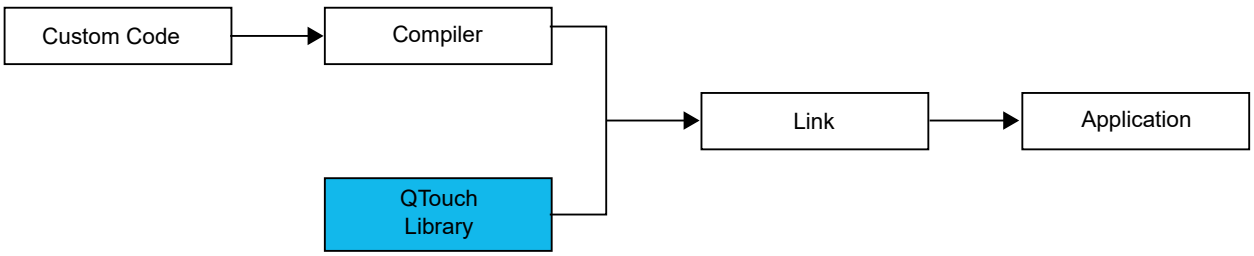



For more information about designing the touch sensor, refer to Buttons, Sliders and Wheels Touch Sensor Design Guide.

## 42.6. Functional Description

In order to access the PTC, the user must use the Atmel Start QTouch® Configurator to configure and link the QTouch Library firmware with the application software. QTouch Library can be used to implement buttons, sliders, and wheels in a variety of combinations on a single interface.

**Figure 42-5.** QTouch Library Usage




For more information about QTouch Library, refer to the QTouch Library Peripheral Touch Controller User Guide.


© 2026 Microchip Technology Inc. and its subsidiaries

# 43. TSENS – Temperature Sensor

## 43.1. Overview

The Temperature Sensor (TSENS) can be used to accurately measure the operating temperature of the device.

**Related Links**

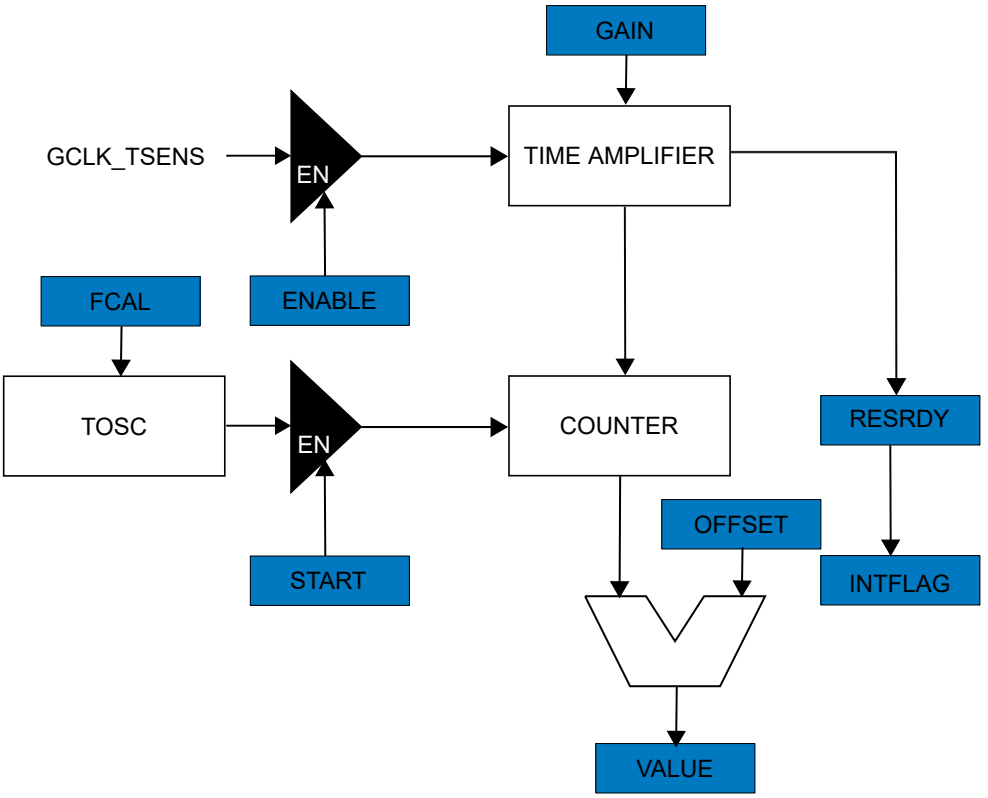
Temperature Sensor Characteristics

## 43.2. Features

- Accurately measures a temperature
- A selectable reference clock source

## 43.3. Block Diagram

**Figure 43-1.** Temperature Sensor Block Diagram.




## 43.4. Signal Description

Not applicable.

## 43.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 43.5.1. I/O Lines

Not applicable.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.5.2. Power Management

The TSENS will continue to operate in any sleep mode where the selected source clock is running. The TSENS's interrupts can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes. Refer to the Power Manager chapter for details on the different sleep modes.

### 43.5.3. Clocks

The TSENS bus clock (CLK_TSENS_APB) can be enabled and disabled in the Main Clock module, and the default state of CLK_TSENS_APB can be found in the Peripheral Clock Masking section.

A generic clock (GCLK_TSENS) is required to clock the TSENS. This clock must be configured and enabled in the generic clock controller before using the TSENS.

This generic clock is asynchronous to the bus clock (CLK_TSENS_APB). Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to Synchronization for details.

**Related Links**
Peripheral Clock Masking
GCLK - Generic Clock Controller

### 43.5.4. DMA

The DMA request line is connected to the DMA Controller (DMAC). Using the TSENS Controller DMA request requires the DMA Controller to be configured first.

**Related Links**
DMAC – Direct Memory Access Controller

### 43.5.5. Interrupts

The interrupt request lines are connected to the interrupt controller. Using the TSENS interrupts requires the interrupt controller to be configured first.

**Related Links**
Nested Vector Interrupt Controller

### 43.5.6. Events

The events are connected to the Event System. Refer to the Event System section for details on how to configure the Event System.

**Related Links**
Event System (EVSYS)

### 43.5.7. Debug Operation

When the CPU is halted in debug mode the TSENS will halt normal operation. Any on-going measurements will be completed. The TSENS can be forced to continue operation during debugging. Refer to DBGCTRL for details.

### 43.5.8. Register Access Protection

All registers with write-access are optionally write-protected by the Peripheral Access Controller (PAC), except the following registers:

- Control B (CTRLB) register
- Interrupt Flag Status and Clear (INTFLAG) register

Write-protection is denoted by the PAC Write-Protection property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

Write-protection does not apply to accesses through an external debugger. Refer to the Peripheral Access Controller chapter for details.

### 43.5.9. Calibration

The GAIN, OFFSET, FCAL, and TCAL calibration values from the production test must be loaded from the NVM Temperature Calibration Area into the TSENS Gain register (GAIN), Offset register (OFFSET) and Calibration register (CAL) by software to achieve specified accuracy.

**Related Links**

NVM Software Calibration Area Mapping

Temperature Sensor Characteristics

## 43.6. Functional Description

### 43.6.1. Principle of Operation

The TSENS accurately measures the operating temperature of the device by comparing the difference in two temperature dependent frequencies to a known frequency. The frequency of the temperature dependent oscillator (TOSC) is measured twice: first with the min configuration and next with the max configuration. The number of periods of GCLK_TSENS used for the measurement is defined by the GAIN register. The width of the resulting pulse is measured using a counter clocked by GCLK_TSENS in the up direction for the 1st phase and in the down 2nd phase.

The resulting signed value is proportional to the temperature and is corrected for offset by the contents of the OFFSET register.

$$\mathrm{VALUE} = \mathrm{OFFSET} + \mathrm{GAIN} \times \left( \frac{f_{\mathrm{TOSCMIN}}}{f_{\mathrm{GCLK}}} + - \frac{f_{\mathrm{TOSCMAX}}}{f_{\mathrm{GCLK}}} \right)$$

**Notes:**

- The values of GAIN and OFFSET are factory programmed to give a specific temperature slope when using the undivided internal 48MHz oscillator (OSC48M) as the GCLK_TSENS source. Other frequencies/sources may be used, but the GAIN setting and/or expected slope will need to be scaled accordingly.
- The calibration value should be copied and written into the GAIN and OFFSET registers to get the specified accuracy.

**Related Links**

VALUE

Value

### 43.6.2. Basic Operation

#### 43.6.2.1. Initialization

The generic clocks (GCLK_TSENS) should be configured and enabled. Refer to the Generic Clock Controller chapter for details.

The following bits are enable-protected, meaning that they can only be written when the TSENS is disabled (CTRLA.ENABLE is zero):

- Run in Standby bit in Control A register (CTRLA.RUNSTDBY)

The following registers are enable-protected:

- Control C (CTRLC)
- Event Control (EVCTRL)
- Window Monitor Lower Threhold (WINLT)


© 2026 Microchip Technology Inc. and its subsidiaries

- Window Monitor Upper Threshold (WINUT)
- Gain Correction (GAIN)
- Offset Correction (OFFSET)
- Calibration (CAL)

Enable-protection is denoted by the Enable-Protected property in the register description.

#### 43.6.2.2. Enabling, Disabling and Resetting

The TSENS is enabled by writing a one to the Enable bit in the Control A register (CTRLA.ENABLE). The TSENS is disabled by writing a zero to CTRLA.ENABLE.

The TSENS is reset by writing a one to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TSENS will be reset to their initial state, and the TSENS will be disabled. Refer to CTRLA for details.

#### 43.6.2.3. Measurement

After the TSENS is enabled, a measurement can be started either manually, by writing a one to the START bit in Control B register (CTRLB.START), or automatically by configuring an event input. A free-running mode can be used to continuously measure the temperature. When the Free running bit in the Control C register (CTRLC.FREERUN) is written to one, there is no need for a trigger to start the measurement. It will start automatically at the end of previous measurement.

The result of the measurement is stored in the Value register (VALUE), overwriting the result from the previous measurement and setting the Result Ready flag in the Interrupt Flag Status and Clear register (INTFLAG.RESRDY). To avoid data loss, the conversion result must be read as soon as it is available. Failing to do so will result in an overrun error condition, indicated by the OVERRUN bit in the Interrupt Flag Status and Clear register (INTFLAG.OVERRUN).

To use an interrupt handler, the corresponding bit in the Interrupt Enable Set register (INTENSET) must be written to one.

To prevent any discrepancies in the temperature measurement, an average on 10 measurements is recommended.

**Related Links**

VALUE

Value

#### 43.6.2.4. Window Monitor

The window monitor feature allows the measurement result in the VALUE register to be compared to predefined threshold values. The window mode is selected by writing the Window Monitor Mode bits in the Control C register (CTRLC.WINMODE). Threshold values must be written in the Window Monitor Lower Threshold register (WINLT) and Window Monitor Upper Threshold register (WINUT).

### 43.6.3. DMA Operation

The TSENS generates the following DMA request:

- Result Ready (RESRDY): the request is set when a measurement result is available, and cleared when the VALUE register is read. The request is generated independent of any Window Monitor condition.

**Related Links**

DMAC – Direct Memory Access Controller

### 43.6.4. Interrupts

The TSENS has the following interrupt sources:

- Result Ready (RESRDY): Indicates when a measurement result is available.


© 2026 Microchip Technology Inc. and its subsidiaries

- Window Monitor (WINMON): Generated when the measurement result matches the window monitor condition. Refer to CTRLC for details.
- Overrun (OVERRUN): Indicates that a new result is ready before the previous result has been read.
- Overflow (OVF): Indicates that the result is invalid because the result required more than 16 bits and overflowed the VALUE register.

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the TSENS is reset. See INTFLAG for details on how to clear interrupt flags.
All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

**Related Links**
Nested Vector Interrupt Controller

### 43.6.5. Events

The TSENS can generate the following output event:

- Window Monitor (WINMON): Generated when the measurement results matches the window monitor condition. Refer to CTRLC for details.

Writing a one to an Event Output bit in the Event Control Register (EVCTRL.WINEO) enables the corresponding output event. Writing a zero to this bit disables the corresponding output event. Refer to the Event System chapter for details on configuring the event system.

The TSENS can take the following action on an input event:

- Start measurement (START): Start a measurement. Refer to CTRLB for details.

Writing a one to an Event Input bit into the Event Control register (EVCTRL.STARTEI) enables the corresponding action on input event. Writing a zero to this bit disables the corresponding action on input event. Refer to the Event System chapter for details. By default, the TSENS will detect a rising edge on the incoming event. If the TSENS action must be performed on the falling edge of the incoming event, the event line must be inverted first, by writing to one the corresponding Event Invert Enable bit in Event Control register (EVCTRL.STARTINV).

**Related Links**
Event System (EVSYS)

### 43.6.6. Sleep Mode Operation

The Run in Standby bit in the Control A register (CTRLA.RUNSTDBY) controls the behavior of the TSENS during standby sleep mode, in cases where the TSENS is enabled (CTRLA.ENABLE = 1).

**Table 43-1.** TSENS Sleep Behavior

| CTRLA.RUNSTDBY | CTRLC.FREERUN | CTRLA.ENABLE | Description |
|---|---|---|---|
| x | x | 0 | Disabled |
| 0 | 0 | 1 | Run in all sleep modes on request, except STANDBY. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 43-1.** TSENS Sleep Behavior (continued)

| CTRLA.RUNSTDBY | CTRLC.FREERUN | CTRLA.ENABLE | Description |
|---|---|---|---|
| 0 | 1 | 1 | Run in all sleep modes, except STANDBY. |
| 1 | 0 | 1 | Run in all sleep modes on request. |
| 1 | 1 | 1 | Run in all sleep modes. |

### 43.6.7. Synchronization

Due to the asynchronicity between the main clock domain (CLK_TSENS_APB) and the peripheral clock domain (GCLK_TSENS) some registers are synchronized when written. When a write-synchronized register is written, the corresponding bit in the Synchronization Busy register (SYNCBUSY) is set immediately. When the write-synchronization is complete, this bit is cleared. Reading a write-synchronized register while the synchronization is ongoing will return the value written, and not the current value in the peripheral clock domain. To read the current value in the peripheral clock domain after writing a register, the user must wait for the corresponding SYNCBUSY bit to be cleared before reading the value.

If an operation that require synchronization is executed while its busy bit is on, the operation is discarded and a bus error is generated.

The following bits need synchronization when written:

- Software Reset bit in Control A register (CTRLA.SWRST)
- Enable bit in Control A register (CTRLA.ENABLE)

Write-synchronization is denoted by the Write-Synchronized property in the register description.


© 2026 Microchip Technology Inc. and its subsidiaries

## 43.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | RUNSTDBY | | | | | ENABLE | SWRST |
| 0x01 | CTRLB | 7:0 | | | | | | | | START |
| 0x02 | CTRLC | 7:0 | | | | FREERUN | | WINMODE[2:0] | | |
| 0x03 | EVCTRL | 7:0 | | | | | | WINEO | STARTINV | STARTEI |
| 0x04 | INTENCLR | 7:0 | | | | | OVF | WINMON | OVERRUN | RESRDY |
| 0x05 | INTENSET | 7:0 | | | | | OVF | WINMON | OVERRUN | RESRDY |
| 0x06 | INTFLAG | 7:0 | | | | | OVF | WINMON | OVERRUN | RESRDY |
| 0x07 | STATUS | 7:0 | | | | | | | | OVF |
| 0x08 | SYNCBUSY | 7:0 | | | | | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | VALUE | 7:0 | VALUE[7:0] | | | | | | | |
| | | 15:8 | VALUE[15:8] | | | | | | | |
| | | 23:16 | VALUE[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | WINLT | 7:0 | WINLT[7:0] | | | | | | | |
| | | 15:8 | WINLT[15:8] | | | | | | | |
| | | 23:16 | WINLT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | WINUT | 7:0 | WINUT[7:0] | | | | | | | |
| | | 15:8 | WINUT[15:8] | | | | | | | |
| | | 23:16 | WINUT[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | GAIN | 7:0 | GAIN[7:0] | | | | | | | |
| | | 15:8 | GAIN[15:8] | | | | | | | |
| | | 23:16 | GAIN[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | OFFSET | 7:0 | OFFSETC[7:0] | | | | | | | |
| | | 15:8 | OFFSETC[15:8] | | | | | | | |
| | | 23:16 | OFFSETC[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 | CAL | 7:0 | | | FCAL[5:0] | | | | | |
| | | 15:8 | | | TCAL[5:0] | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 | DBGCTRL | 7:0 | | | | | | | | DBGRUN |

## 43.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection, Write-Synchronized (ENABLE, SWRST)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | RUNSTDBY | | | | | ENABLE | SWRST |
| Access | | R/W | | | | | R/W | R/W |
| Reset | | 0 | | | | | 0 | 0 |

**Bit 6 – RUNSTDBY** Run in Standby
This bit controls how the TSENS behaves during standby sleep mode:
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | The TSENS is halted during standby sleep mode. |
| 1 | The TSENS is not stopped in standby sleep mode. If CTRLC.FREERUN is zero, the TSENS will be running when a peripheral is requesting it. If CTRLC.FREERUN is one, the TSENS will always be running in standby sleep mode. |

**Bit 1 – ENABLE** Enable
Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the ENABLE bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a zero to this bit has no effect.
Writing a one to this bit resets all registers in the TSENS, except GAIN, OFFSET, CAL and DBGCTRL, to their initial state, and the TSENS will be disabled.
Writing a one to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no reset operation ongoing. |
| 1 | The reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.2. Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | START |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – START** Start Measurement

| Value | Description |
|---|---|
| 0 | Writing a zero to this bit has no effect. |
| 1 | Writing a one to this bit starts a measurement |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.3. Control C

**Name:** CTRLC
**Offset:** 0x02
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | FREERUN | | WINMODE[2:0] | | |
| Access | | | | R/W | | R/W | R/W | R/W |
| Reset | | | | 0 | | 0 | 0 | 0 |

**Bit 4 – FREERUN** Free Running Measurement

| Value | Description |
|---|---|
| 0 | TSENS operates in single measurement mode. |
| 1 | TSENS is in free running mode and a new measurement will be initiated when the previous measurement completes. |

**Bits 2:0 – WINMODE[2:0]** Window Monitor Mode
These bits enable and define the window monitor mode.

| Value | Name | Description |
|---|---|---|
| 0x0 | DISABLE | No window mode (default) |
| 0x1 | ABOVE | VALUE > WINLT |
| 0x2 | BELOW | VALUE < WINUT |
| 0x3 | INSIDE | WINLT < VALUE < WINUT |
| 0x4 | OUTSIDE | WINUT < VALUE < WINLT |
| 0x5 | HYST_ABOVE | VALUE > WINUT with hysteresis to WINLT |
| 0x6 | HYST_BELOW | VALUE < WINLT with hysteresis to WINUT |
| 0x07 | | Reserved |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.4. Event Control

**Name:** EVCTRL
**Offset:** 0x03
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | WINEO | STARTINV | STARTEI |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 0 | 0 |

**Bit 2 – WINEO** Window Monitor Event Out
This bit indicates whether the Window Monitor event output is enabled or not and an output event will be generated when the window monitor detects something.

| Value | Description |
|---|---|
| 0 | Window Monitor event output is disabled and an event will not be generated. |
| 1 | Window Monitor event output is enabled and an event will be generated. |

**Bit 1 – STARTINV** Start Conversion Event Invert Enable

| Value | Description |
|---|---|
| 0 | start event input source is not inverted. |
| 1 | start event input source is inverted. |

**Bit 0 – STARTEI** Start Conversion Event Input Enable

| Value | Description |
|---|---|
| 0 | A new conversion will not be triggered on any incoming event. |
| 1 | A new conversion will be triggered on any incoming event. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.5. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x04
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVF | WINMON | OVERRUN | RESRDY |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – OVF** Overflow Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Overflow Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The overflow interrupt is disabled. |
| 1 | The overflow interrupt is enabled, and an interrupt request will be generated when the Overflow interrupt flag is set. |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Window Monitor Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The window monitor interrupt is disabled. |
| 1 | The window monitor interrupt is enabled, and an interrupt request will be generated when the Window Monitor interrupt flag is set. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Overrun Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled, and an interrupt request will be generated when the Overrun interrupt flag is set. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will clear the Result Ready Interrupt Enable bit, which disables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled, and an interrupt request will be generated when the Result Ready interrupt flag is set. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.6. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x05
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to enable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVF | WINMON | OVERRUN | RESRDY |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – OVF** Overflow Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Overflow Interrupt bit, which enables the Overflow interrupt.

| Value | Description |
|---|---|
| 0 | The Overflow interrupt is disabled. |
| 1 | The Overflow interrupt is enabled. |

**Bit 2 – WINMON** Window Monitor Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Window Monitor Interrupt bit, which enables the Window Monitor interrupt.

| Value | Description |
|---|---|
| 0 | The Window Monitor interrupt is disabled. |
| 1 | The Window Monitor interrupt is enabled. |

**Bit 1 – OVERRUN** Overrun Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Overrun Interrupt Enable bit, which enables the corresponding interrupt request.

| Value | Description |
|---|---|
| 0 | The Overrun interrupt is disabled. |
| 1 | The Overrun interrupt is enabled. |

**Bit 0 – RESRDY** Result Ready Interrupt Enable
Writing a zero to this bit has no effect.
Writing a one to this bit will set the Result Ready Interrupt bit, which enables the Result Ready interrupt.

| Value | Description |
|---|---|
| 0 | The Result Ready interrupt is disabled. |
| 1 | The Result Ready interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.7. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x06
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OVF | WINMON | OVERRUN | RESRDY |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – OVF** Overflow
This flag is cleared by writing a one to the flag.
This flag is set when the conversion result requires more than 24 bits and overflows the VALUE register, and an interrupt request will be generated if INTENCLR/SET.OVF is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Overflow interrupt flag.

**Bit 2 – WINMON** Window Monitor
This flag is cleared by writing a one to the flag or by reading the VALUE register.
This flag is set on the next cycle after a match with the window monitor condition, and an interrupt request will be generated if INTENCLR/SET.WINMON is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Window Monitor interrupt flag.

**Bit 1 – OVERRUN** Overrun
This flag is cleared by writing a one to the flag.
This flag is set if a valid VALUE is updated before the previous valid value has been read by the CPU, and an interrupt will be generated if INTENCLR/SET.OVERRUN is one.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Overrun interrupt flag.

**Bit 0 – RESRDY** Result Ready
This flag is cleared by writing a one to the flag or by reading the VALUE register.
This flag is set when the conversion result is available, and an interrupt will be generated if INTENCLR/SET.RESRDY is one.
This flag will not set if an overflow occurs during the conversion.
Writing a zero to this bit has no effect.
Writing a one to this bit clears the Result Ready interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.8. Status

**Name:** STATUS
**Offset:** 0x07
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | OVF |
| Access | | | | | | | | R |
| Reset | | | | | | | | 0 |

**Bit 0 – OVF** Result Overflow
Writing a zero to this bit has no effect.
Writing a one to this bit has no effect.

| Value | Description |
|---|---|
| 0 | No overflow in the VALUE register has occurred. The result is valid. |
| 1 | An overflow occurred in the VALUE register. The result is not valid. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.9. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** Enable Busy
This bit is cleared when the synchronization of CTRLA.ENABLE is complete.
This bit is set when the synchronization of CTRLA.ENABLE is started.

**Bit 0 – SWRST** Software Reset Busy
This bit is cleared when the synchronization of CTRLA.SWRST is complete.
This bit is set when the synchronization of CTRLA.SWRST is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.10. Value

**Name:** VALUE
**Offset:** 0x0C
**Reset:** 0x0000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – VALUE[23:0]** Measurement Value

Result from measurement. This VALUE is in two's complement format.

**Example:** If the TSENS GAIN and OFFSET registers are setup with values stored in the NVM Temperature Calibration Area (Refer to Table 9-6), the TSENS resolution is set at 100 which will result in the following values

| Temperature | VALUE |
|---|---|
| T = 25°C | 2500 = 0x09C4 |
| T = -25°C | -2500 = 0xFFF63C |


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.11. Window Monitor Lower Threshold

**Name:** WINLT
**Offset:** 0x10
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINLT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – WINLT[23:0]** Window Lower Threshold
If the window monitor is enabled, these bits define the lower threshold value. This WINLT value is in two's complement format.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.12. Window Monitor Upper Threshold

**Name:** WINUT
**Offset:** 0x14
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WINUT[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – WINUT[23:0]** Window Upper Threshold
If the window monitor is enabled, these bits define the upper threshold value. This WINUT value is in two's complement format.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.13. Gain

**Name:** GAIN
**Offset:** 0x18
**Reset:** 0x0000
**Property:** Enable-Protected, PAC Write-Protection, not reset by a software reset

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | GAIN[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | GAIN[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | GAIN[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – GAIN[23:0]** Time Amplifier Gain

This value from production test must be loaded from the NVM temperature calibration row into the register by software to achieve the specified accuracy.
The bitfield can also be written by CPU.
The GAIN value defines the number of GCLK_TSENS periods that will be used for a measurement cycle.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.14. Offset

**Name:** OFFSET
**Offset:** 0x1C
**Reset:** 0x0000
**Property:** Enable-Protected, PAC Write-Protection, not reset by a software reset

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETC[23:16] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETC[15:8] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OFFSETC[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – OFFSETC[23:0]** Offset Correction

This value from production test must be loaded from the NVM temperature calibration row into the register by software to achieve the specified accuracy.
The bitfield can also be written by CPU.
These bits define how the TSENS measurement result is compensated for offset error before being written to the VALUE register. This OFFSET value is in two's complement format.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.15. Calibration

**Name:** CAL
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** Enable-Protected, PAC Write-Protection, not reset by a software reset

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | TCAL[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | FCAL[5:0] | | | | | |
| Access | | | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 13:8 – TCAL[5:0]** Temperature Calibration
This value from production test must be loaded from the NVM software calibration row into the CAL register by software to achieve the specified accuracy. The value must be copied only, and must not be changed.

**Bits 5:0 – FCAL[5:0]** Frequency Calibration
This value from production test must be loaded from the NVM software calibration row into the CAL register by software to achieve the specified accuracy. The value must be copied only, and must not be changed.


© 2026 Microchip Technology Inc. and its subsidiaries

### 43.8.16. Debug Control

**Name:** DBGCTRL
**Offset:** 0x24
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DBGRUN |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DBGRUN** Debug Run
This bit is not reset by a software reset.
This bits controls the functionality when the CPU is halted by an external debugger.

| Value | Description |
|---|---|
| 0 | The TSENS is halted when the CPU is halted by an external debugger. Any on-going measurement will complete. |
| 1 | The TSENS continues normal operation when the CPU is halted by an external debugger. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 44. FREQM – Frequency Meter

## 44.1. Overview

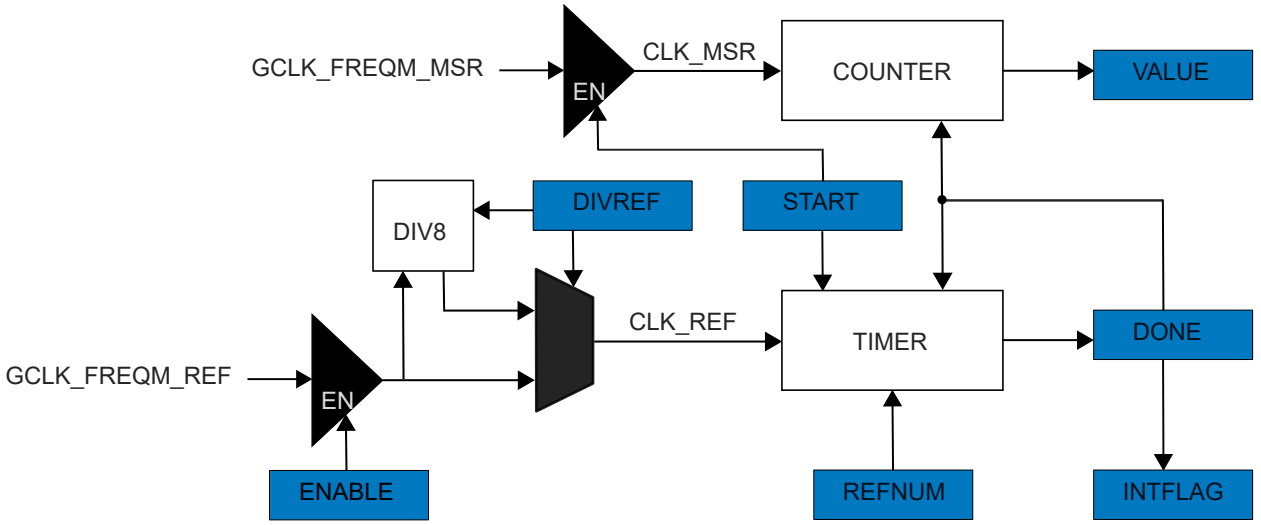
The Frequency Meter (FREQM) can be used to accurately measure the frequency of a clock by comparing it to a known reference clock.

## 44.2. Features

- Ratio can be measured with 24-bit accuracy
- Accurately measures the frequency of an input clock with respect to a reference clock
- Reference clock can be selected from the available GCLK_FREQM_REF sources
- Measured clock can be selected from the available GCLK_FREQM_MSR sources

## 44.3. Block Diagram

**Figure 44-1.** FREQM Block Diagram




**Note:** DIVREF is only available on SAM C20/C21 N variants.

## 44.4. Signal Description

Not applicable.

## 44.5. Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

### 44.5.1. I/O Lines

The GCLK I/O lines (GCLK_IO[7:0]) can be used as measurement or reference clock sources. This requires the I/O pins to be configured.

### 44.5.2. Power Management

The FREQM will continue to operate in idle sleep mode where the selected source clock is running. The FREQM's interrupts can be used to wake up the device from idle sleep mode. Refer to the Power Manager chapter for details on the different sleep modes.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.5.3. Clocks

The clock for the FREQM bus interface (CLK_FREQM_APB) is enabled and disabled by the Main Clock Controller, the default state of CLK_APB_FREQM can be found in Peripheral Clock Masking.

Two generic clocks are used by the FREQM: Reference Clock (GCLK_FREQM_REF) and Measurement Clock (GCLK_FREQM_MSR).

GCLK_FREQM_REF is required to clock the internal reference timer, which acts as the frequency reference.

GCLK_FREQM_MSR is required to clock a ripple counter for frequency measurement. These clocks must be configured and enabled in the generic clock controller before using the FREQM.

**Related Links**

MCLK – Main Clock

Peripheral Clock Masking

GCLK - Generic Clock Controller

### 44.5.4. DMA

Not applicable.

### 44.5.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using FREQM interrupt requires the interrupt controller to be configured first.

**Related Links**

Nested Vector Interrupt Controller

### 44.5.6. Events

Not applicable

### 44.5.7. Debug Operation

When the CPU is halted in debug mode the FREQM continues its normal operation. The FREQM cannot be halted when the CPU is halted in debug mode. If the FREQM is configured in a way that requires it to be periodically serviced by the CPU, improper operation or data loss may result during debugging.

### 44.5.8. Register Access Protection

All registers with write-access can be write-protected optionally by the Peripheral Access Controller (PAC), except the following registers:

- Control B register (CTRLB)
- Interrupt Flag Status and Clear register (INTFLAG)
- Status register (STATUS)

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Write-protection does not apply to accesses through an external debugger.

**Related Links**

PAC - Peripheral Access Controller


© 2026 Microchip Technology Inc. and its subsidiaries

## 44.6. Functional Description

### 44.6.1. Principle of Operation

FREQM counts the number of periods of the measured clock (GCLK_FREQM_MSR) with respect to the reference clock (GCLK_FREQM_REF). The measurement is done for a period of REFNUM/$f_{CLK_REF}$ and stored in the Value register (VALUE.VALUE). REFNUM is the number of Reference clock cycles selected in the Configuration A register (CFGA.REFNUM).

The frequency of the measured clock, $f_{CLK_MSR}$, is calculated by

$$f_{CLK_MSR} = \left(\frac{VALUE}{REFNUM}\right) f_{CLK_REF}$$

### 44.6.2. Basic Operation

#### 44.6.2.1. Initialization

Before enabling FREQM, the device and peripheral must be configured:

- Each of the generic clocks (GCLK_FREQM_REF and GCLK_FREQM_MSR) must be configured and enabled.
- 

> **Important:** The reference clock must be slower than the measurement clock.

- Write the number of Reference clock cycles for which the measurement is to be done in the Configuration A register (CFGA.REFNUM). This must be a non-zero number.

The following register is enable-protected, meaning that it can only be written when the FREQM is disabled (CTRLA.ENABLE=0):

- Configuration A register (CFGA)

Enable-protection is denoted by the "Enable-Protected" property in the register description.

**Related Links**
GCLK - Generic Clock Controller

#### 44.6.2.2. Enabling, Disabling and Resetting

The FREQM is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The peripheral is disabled by writing CTRLA.ENABLE=0.

The FREQM is reset by writing a '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). On software reset, all registers in the FREQM will be reset to their initial state, and the FREQM will be disabled.

Then ENABLE and SWRST bits are write-synchronized.

**Related Links**
Synchronization

#### 44.6.2.3. Measurement

In the Configuration A register, the Number of Reference Clock Cycles field (CFGA.REFNUM) selects the duration of the measurement. The measurement is given in number of GCLK_FREQM_REF periods.
**Note:** The REFNUM field must be written before the FREQM is enabled.

After the FREQM is enabled, writing a '1' to the START bit in the Control B register (CTRLB.START) starts the measurement. The BUSY bit in Status register (STATUS.BUSY) is set when the measurement starts, and cleared when the measurement is complete.


© 2026 Microchip Technology Inc. and its subsidiaries

There is also an interrupt request for Measurement Done: When the Measurement Done bit in Interrupt Enable Set register (INTENSET.DONE) is '1' and a measurement is finished, the Measurement Done bit in the Interrupt Flag Status and Clear register (INTFLAG.DONE) will be set and an interrupt request is generated.

The result of the measurement can be read from the Value register (VALUE.VALUE). The frequency of the measured clock GCLK_FREQM_MSR is then:

$$f_{CLK_MSR} = \left(\frac{VALUE}{REFNUM}\right) f_{CLK_REF}$$

**Note:** In order to make sure the measurement result (VALUE.VALUE[23:0]) is valid, the overflow status (STATUS.OVF) should be checked.

In case an overflow condition occurred, indicated by the Overflow bit in the STATUS register (STATUS.OVF), either the number of reference clock cycles must be reduced (CFGA.REFNUM), or a faster reference clock must be configured. Once the configuration is adjusted, clear the overflow status by writing a '1' to STATUS.OVF. Then another measurement can be started by writing a '1' to CTRLB.START.

### 44.6.3. DMA Operation

Not applicable.

### 44.6.4. Interrupts

The FREQM has one interrupt source:

- DONE: A frequency measurement is done.

The interrupt flag in the Interrupt Flag Status and Clear (INTFLAG) register is set when the interrupt condition occurs. The interrupt can be enabled by writing a '1' to the corresponding bit in the Interrupt Enable Set (INTENSET) register, and disabled by writing a '1' to the corresponding bit in the Interrupt Enable Clear (INTENCLR) register. The status of enabled interrupts can be read from either INTENSET or INTENCLR.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled, or the FREQM is reset. See INTFLAG for details on how to clear interrupt flags. All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC. The user must read the INTFLAG register to determine which interrupt condition is present.

This interrupt is a synchronous wake-up source.

Note that interrupts must be globally enabled for interrupt requests to be generated.

### 44.6.5. Events

Not applicable.

### 44.6.6. Sleep Mode Operation

The FREQM will continue to operate in Idle Sleep mode where the selected source clock is running. The FREQM's interrupts can be used to wake up the device from Idle Sleep mode.

For lowest chip power consumption in sleep modes, FREQM should be disabled before entering a Sleep mode.

**Related Links**

PM - Power Manager


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.6.7. Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits and registers are write-synchronized:

- Software Reset bit in Control A register (CTRLA.SWRST)
- Enable bit in Control A register (CTRLA.ENABLE)

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

**Related Links**

Register Synchronization


© 2026 Microchip Technology Inc. and its subsidiaries

## 44.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | | | ENABLE | SWRST |
| 0x01 | CTRLB | 7:0 | | | | | | | | START |
| 0x02 | CFGA | 7:0 | REFNUM[7:0] | | | | | | | |
| | | 15:8 | DIVREF | | | | | | | |
| 0x04 ... 0x07 | Reserved | | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | | | | | | | | DONE |
| 0x09 | INTENSET | 7:0 | | | | | | | | DONE |
| 0x0A | INTFLAG | 7:0 | | | | | | | | DONE |
| 0x0B | STATUS | 7:0 | | | | | | | OVF | BUSY |
| 0x0C | SYNCBUSY | 7:0 | | | | | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | VALUE | 7:0 | VALUE[7:0] | | | | | | | |
| | | 15:8 | VALUE[15:8] | | | | | | | |
| | | 23:16 | VALUE[23:16] | | | | | | | |
| | | 31:24 | | | | | | | | |

## 44.8. Register Description

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

Some registers are optionally write-protected by the Peripheral Access Controller (PAC). Optional PAC write-protection is denoted by the "PAC Write-Protection" property in each individual register description.


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.1. Control A

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** Enable
Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE will read back immediately and the ENABLE bit in the Synchronization Busy register (SYNCBUSY.ENABLE) will be set. SYNCBUSY.ENABLE will be cleared when the operation is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | The peripheral is disabled. |
| 1 | The peripheral is enabled. |

**Bit 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets all registers in the FREQM to their initial state, and the FREQM will be disabled. Writing a '1' to this bit will always take precedence, meaning that all other writes in the same write-operation will be discarded.
Due to synchronization there is a delay from writing CTRLA.SWRST until the Reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the Reset is complete.
This bit is not enable-protected.

| Value | Description |
|---|---|
| 0 | There is no ongoing Reset operation. |
| 1 | The Reset operation is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.2. Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | START |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – START** Start Measurement

| Value | Description |
|---|---|
| 0 | Writing a '0' has no effect. |
| 1 | Writing a '1' starts a measurement. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.3. Configuration A

**Name:** CFGA
**Offset:** 0x02
**Reset:** 0x0000
**Property:** PAC Write-Protection, Enable-protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | DIVREF | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | REFNUM[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – DIVREF** Divide Reference Clock. Only available on SAM C20/C21 N variants.
Divides the reference clock by 8

| Value | Description |
|---|---|
| 0 | The reference clock is divided by 1. |
| 1 | The reference clock is divided by 8. |

**Bits 7:0 – REFNUM[7:0]** Number of Reference Clock Cycles
Selects the duration of a measurement in number of CLK_FREQM_REF cycles. This must be a non-zero value, i.e. 0x01 (one cycle) to 0xFF (255 cycles).


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.4. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x08
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DONE |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DONE** Measurement Done Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Measurement Done Interrupt Enable bit, which disables the Measurement Done interrupt.

| Value | Description |
|---|---|
| 0 | The Measurement Done interrupt is disabled. |
| 1 | The Measurement Done interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.5. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DONE |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DONE** Measurement Done Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Measurement Done Interrupt Enable bit, which enables the Measurement Done interrupt.

| Value | Description |
|---|---|
| 0 | The Measurement Done interrupt is disabled. |
| 1 | The Measurement Done interrupt is enabled. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.6. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x0A
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | DONE |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – DONE** Mesurement Done
This flag is cleared by writing a '1' to it.
This flag is set when the STATUS.BUSY bit has a one-to-zero transition.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the DONE interrupt flag.


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.7. Status

**Name:** STATUS
**Offset:** 0x0B
**Reset:** 0x00
**Property:** –

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | OVF | BUSY |
| Access | | | | | | | R/W | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – OVF** Sticky Count Value Overflow
This bit is cleared by writing a '1' to it.
This bit is set when an overflow condition occurs to the value counter.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the OVF status.

**Bit 0 – BUSY** FREQM Status

| Value | Description |
|---|---|
| 0 | No ongoing frequency measurement. |
| 1 | Frequency measurement is ongoing. |


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.8. Synchronization Busy

**Name:** SYNCBUSY
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ENABLE | SWRST |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – ENABLE** Enable
This bit is cleared when the synchronization of CTRLA.ENABLE is complete.
This bit is set when the synchronization of CTRLA.ENABLE is started.

**Bit 0 – SWRST** Synchronization Busy
This bit is cleared when the synchronization of CTRLA.SWRST is complete.
This bit is set when the synchronization of CTRLA.SWRST is started.


© 2026 Microchip Technology Inc. and its subsidiaries

### 44.8.9. Value

**Name:** VALUE
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** –

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[23:16] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[15:8] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | VALUE[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 23:0 – VALUE[23:0]** Measurement Value
Result from measurement.


© 2026 Microchip Technology Inc. and its subsidiaries

# 45. Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

**Related Links**

Electrical Characteristics 105°C (SAM C20/C21 E/G/J)

## 45.1. Disclaimer

All typical values are measured at Ta = 25°C unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

This chapter only contains characteristics specific for SAM C20/C21 E/G/J.

## 45.2. Absolute Maximum Ratings

Stresses beyond those listed in the below table may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**Table 45-1.** Absolute maximum ratings

| Symbol | Parameter | Min. | Max. | Units |
|---|---|---|---|---|
| $V_{DD}$ | Power supply voltage | 0 | 6.1 | V |
| $I_{VDD}$ | Current into a $V_{DD}$ pin | - | 92 | mA |
| $I_{GND}$ | Current out of a GND pin | - | 130 | mA |
| $V_{PIN}$ | Pin voltage with respect to GND and $V_{DD}$ | GND - 0.3V | $V_{DD}$ + 0.3V | V |
| $T_{STORAGE}$ | Storage temperature | -60 | 150 | °C |

**CAUTION** This device is sensitive to electrostatic discharges (ESD). Improper handling may lead to permanent performance degradation or malfunctioning.
Handle the device following best practice ESD protection rules: Be aware that the human body can accumulate charges large enough to impair functionality or destroy the device.

**CAUTION** In debugger cold-plugging mode, NVM erase operations are not protected by the BODVDD and BODCORE. NVM erase operation at supply voltages below specified minimum can cause corruption of NVM areas that are mandatory for correct device behavior.

**Related Links**

GPIO Clusters

## 45.3. General Operating Ratings

The device must operate within the ratings listed in the table below in order for all other electrical characteristics and typical characteristics of the device to be valid.

NVM erase operations are not protected by the BODVDD and BODCORE in debugger cold-plugging mode. NVM erase operation at supply voltages below product specification minimum can cause corruption of the calibration and other areas mandatory for a correct product behavior.

**Table 45-2.** General operating conditions

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $V_{DDIN}$ | Power supply voltage | 2.7(1) | 5.0 | 5.5 | V |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-2.** General operating conditions (continued)

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $V_{DDANA}$ | Analog supply voltage | 2.7[1] | 5.0 | 5.5 | V |
| $V_{DDIO}$ | IO supply voltage | 2.7[1] | 5.0 | 5.5 | V |
| $T_A$ | Temperature range | -40 | 25 | 85 | °C |
| $T_J$ | Junction temperature | - | - | 100 | °C |

**Note:**

1. With BODVDD disabled. If the BODVDD is enabled, refer to Table 45-18.

**Note:** The same voltage must be applied to VDDIN and VDDANA. VDDIO should be lower or equal to VDDIN/ VDDANA. The common voltage is referred to as VDD in the data sheet. Some I/O are in the VDDIO cluster, but can be multiplexed as analog functions (inputs or outputs). In such a case, VDDANA is used to power the I/O. Using this configuration may result in an electrical conflict if the VDDIO voltage is lower than the VDDIN/VDDANA.

**Related Links**

BODVDD - Brown Out Detector Characteristics

## 45.4. Injection Current

Stresses beyond those listed in the table below may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**Table 45-3.** Injection Current[1]

| Symbol | Description | min | max | Unit |
|---|---|---|---|---|
| $I_{INJ1}$[2] | IO pin injection current | -1 | +1 | mA |
| $I_{INJ2}$[3] | IO pin injection current | -15 | +15 | mA |
| $I_{INJtotal}$ | Sum of IO pins injection current | -45 | +45 | mA |

**Note:**

1. Injecting current may have an effect on the accuracy of the analog blocks.
2. Conditions for $V_{PIN}$: $V_{PIN}$<GND-0.6V or 5.5V<$V_{PIN}$<=6.1V.
   Conditions for VDD 4.9V<VDD<=5.5V.
   If Vpin is lower than GND-0.6V, then a current limiting resistor is required. The negative DC injection current limiting resistor is calculated as R = $|(GND-0.6V - V_{PIN})/I_{INJ1}|$. If Vpin is greater than VDD+0.6V, then a current limiting resistor is required. The positive DC injection current limiting resistor is calculated as R = $(V_{PIN}-(VDD+0.6))/I_{INJ1}$.
3. Conditions for $V_{PIN}$: $V_{PIN}$<GND-0.6V or $V_{PIN}$<=5.5V.
   Conditions for VDD: VDD<=4.9V.
   If Vpin is lower than GND-0.6V, then a current limiting resistor is required. The negative DC injection current limiting resistor is calculated as R = $|(GND-0.6V - V_{PIN})/I_{INJ2}|$. If Vpin is greater than VDD+0.6V, then a current limiting resistor is required. The positive DC injection current limiting resistor is calculated as R = $(V_{PIN}-(VDD+0.6))/I_{INJ2}$.

## 45.5. Supply Characteristics


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-4.** Supply Characteristics

| Symbol | Conditions | Voltage | | |
|---|---|---|---|---|
| | | Min. | Max. | Units |
| $V_{DDIO}$<br>$V_{DDIN}$<br>$V_{DDANA}$ | Full Voltage Range | 2.7 | 5.5 | V |

**Table 45-5.** Supply Rise Rates

| Symbol | Parameter | Fall Rate | Rise Rate | Units |
|---|---|---|---|---|
| | | Max | Max. | |
| $V_{DDIO}$ | DC supply peripheral I/Os, internal regulator and analog supply | 0.05 | 0.1 | V/µs |
| $V_{DDIN}$ | | 0.05 | 0.1 | |
| $V_{DDANA}$ | | 0.05 | 0.1 | |

**Table 45-6.** Power Supply Current Requirement

| Symbol | Conditions | Current | Units |
|---|---|---|---|
| | | Max | |
| $I_{INPUT}$(1) | Power up Maximum current | 1.9 | mA |

**Note:**

1. $I_{INPUT}$ is the minimum requirement for the power supply connected to the device.

**Related Links**

Power Supply and Start-Up Considerations

## 45.6. Maximum Clock Frequencies

**Table 45-7.** Maximum GCLK Generator Output Frequencies

| Symbol | Max. | Units |
|---|---|---|
| $f_{GCLKGEN0}$ / $f_{GCLK_MAIN}$ | 96 | MHz |
| $f_{GCLKGEN1}$ | | |
| $f_{GCLKGEN2}$ | | |
| $f_{GCLKGEN3}$ | 66 | MHz |
| $f_{GCLKGEN4}$ | | |
| $f_{GCLKGEN5}$ | | |
| $f_{GCLKGEN6}$ | | |
| $f_{GCLKGEN7}$ | | |
| $f_{GCLKGEN8}$ | | |

**Table 45-8.** Maximum CPU and Bus Clock Frequencies for Parts Running at 64 MHz (1)

| Symbol | Description | Max. | Units |
|---|---|---|---|
| $f_{CPU}$ | CPU clock frequency | 64 | MHz |
| $f_{AHB}$ | AHB clock frequency | 64 | MHz |
| $f_{APBA}$ | APBA clock frequency | 64 | MHz |
| $f_{APBB}$ | APBB clock frequency | 64 | MHz |
| $f_{APBC}$ | APBC clock frequency | 64 | MHz |
| $f_{APBD}$ | APBD clock frequency | 64 | MHz |

**Note:**

1. See the Ordering Information section to determine the parts running at 64 MHz.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-9.** Maximum CPU and Bus Clock Frequencies for Parts Limited to 48 MHz (1)

| Symbol | Description | Max. | Units |
|---|---|---|---|
| $f_{CPU}$ | CPU clock frequency | 48 | MHz |
| $f_{AHB}$ | AHB clock frequency | 48 | MHz |
| $f_{APBA}$ | APBA clock frequency | 48 | MHz |
| $f_{APBB}$ | APBB clock frequency | 48 | MHz |
| $f_{APBC}$ | APBC clock frequency | 48 | MHz |
| $f_{APBD}$ | APBD clock frequency | 48 | MHz |

**Note:**

1. See the Ordering Information section to determine the parts limited to 48 MHz.

**Table 45-10.** Maximum Peripheral Clock Frequencies

| Symbol | Description | Max. | Units |
|---|---|---|---|
| $f_{GCLK_DPLL}$ | FDPLL96M Reference clock frequency | 2 | MHz |
| $f_{GCLK_DPLL_32K}$ | FDPLL96M 32k Reference clock frequency | 33 | KHz |
| $f_{GCLK_EIC}$ | EIC input clock frequency | 48 | MHz |
| $f_{GCLK_FREQM_MSR}$ | FREQM Measure | 96 | MHz |
| $f_{GCLK_FREQM_REF}$ | FREQM Reference | 48 | MHz |
| $f_{GCLK_TSENS}$ | TSENS input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_0}$ | EVSYS channel 0 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_1}$ | EVSYS channel 1 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_2}$ | EVSYS channel 2 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_3}$ | EVSYS channel 3 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_4}$ | EVSYS channel 4 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_5}$ | EVSYS channel 5 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_6}$ | EVSYS channel 6 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_7}$ | EVSYS channel 7 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_8}$ | EVSYS channel 8 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_9}$ | EVSYS channel 9 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_10}$ | EVSYS channel 10 input clock frequency | 48 | MHz |
| $f_{GCLK_EVSYS_CHANNEL_11}$ | EVSYS channel 11 input clock frequency | 48 | MHz |
| $f_{GCLK_SERCOMx_SLOW}$ | Common SERCOM slow input clock frequency | 5 | MHz |
| $f_{GCLK_SERCOMn_CORE}$ | SERCOMn input clock frequency | 48 | MHz |
| $f_{GCLK_CANn}$ | CAN input clock frequency | 48 | MHz |
| $f_{GCLK_TCC0,\ 1}$ | TCCn input clock frequency | 92 | MHz |
| $f_{GCLK_TCC2}$ | TCC2 input clock frequency | 48 | MHz |
| $f_{GCLK_TCn}$ | TCn input clock frequency | 48 | MHz |
| $f_{GCLK_ADCn}$ | ADCn input clock frequency | 48 | MHz |
| $f_{GCLK_SDADC}$ | SDADC input clock frequency | 48 | MHz |
| $f_{GCLK_DAC}$ | DAC input clock frequency | 48 | MHz |
| $f_{GCLK_PTC}$ | PTC input clock frequency | 48 | MHz |
| $f_{GCLK_CCL}$ | CCL input clock frequency | 48 | MHz |
| $f_{GCLK_AC}$ | AC digital input clock frequency | 48 | MHz |

## 45.7. Power Consumption

The values in the Power Consumption table below are measured values of power consumption under the following conditions, except where noted:


© 2026 Microchip Technology Inc. and its subsidiaries

- Operating conditions
  - VDDIN = 3.0 V, 5.0V
- Oscillators
  - XOSC (Crystal oscillator) stopped
  - XOSC32K (32.768 kHz Crystal oscillator) running with external 32.768 kHz crystal
  - FDPLL using XOSC32K as reference and running at 48 MHz or 64 MHz
- Clocks
  - FDPLL used as main clock source, except otherwise specified
  - CPU, AHB clocks undivided
  - All peripheral clocks stopped
- I/Os are inactive with input trigger disable
- CPU is running on Flash with Wait states specified in NVM Max Speed Characteristics
- NVMCTRL cache enabled
- BODVDD disabled

**Table 45-11.** Current Consumption with CPU Running at 48 MHz(1)

| Mode | conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 3.8 | 4.2 | mA |
| | | 85°C | 5.0V | 3.9 | 4.3 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 3.7 | 4.1 | mA |
| | | 85°C | 3.0V | 3.9 | 4.3 | |
| | CPU running a While 1 algorithm, with GCLKIN as reference | 25°C | 5.0V | 71*Freq+160 | 78*Freq+162 | µA (with freq in MHz) |
| | | 85°C | 5.0V | 71*Freq+253 | 74*Freq+447 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 4.7 | 5.2 | mA |
| | | 85°C | 5.0V | 4.8 | 5.3 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 4.7 | 5.1 | mA |
| | | 85°C | 3.0V | 4.8 | 5.3 | |
| | CPU running a Fibonacci algorithm, with GCLKIN as reference | 25°C | 5.0V | 90*Freq+163 | 99*Freq+168 | µA (with freq in MHz) |
| | | 85°C | 5.0V | 90*Freq+258 | 95*Freq+450 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 5.9 | 6.4 | mA |
| | | 85°C | 5.0V | 6.1 | 6.6 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 5.2 | 5.7 | mA |
| | | 85°C | 3.0V | 5.4 | 5.8 | |
| | CPU running a CoreMark algorithm, with GCLKIN as reference | 25°C | 5.0V | 115*Freq+167 | 126*Freq+167 | µA (with freq in MHz) |
| | | 85°C | 5.0V | 117*Freq+261 | 122*Freq+454 | |
| IDLE0 | | 25°C | 5.0V | 1.8 | 1.9 | mA |
| | | 85°C | 5.0V | 1.9 | 2.2 | |
| IDLE2 | | 25°C | 5.0V | 1.2 | 1.277 | mA |
| | | 85°C | 5.0V | 1.3 | 1.583 | |
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37.0 | µA |
| | | 85°C | 5.0V | 89.8 | 302.0 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35.0 | |
| | | 85°C | 5.0V | 87.8 | 300.0 | |

**Note:**

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-12.** Current Consumption with CPU Running at 64 MHz [1,2]

| Mode | Conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 4.5 | 4.8 | mA |
| | | 85°C | 5.0V | 4.7 | 5.1 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 4.5 | 4.8 | mA |
| | | 85°C | 3.0V | 4.7 | 5.1 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 6.2 | 6.6 | mA |
| | | 85°C | 5.0V | 6.4 | 6.9 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 6.2 | 6.6 | mA |
| | | 85°C | 3.0V | 6.3 | 6.8 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 6.9 | 7.3 | mA |
| | | 85°C | 5.0V | 7.1 | 7.6 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 6.8 | 7.2 | mA |
| | | 85°C | 3.0V | 7.0 | 7.5 | |
| IDLE0 | | 25°C | 5,0V | 2.2 | 2.4 | mA |
| | | 85°C | 5,0V | 2.4 | 2.8 | |
| IDLE2 | | 25°C | 5.0V | 1.5 | 1.6 | |
| | | 85°C | 5.0V | 1.7 | 2.0 | |
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37.0 | µA |
| | | 85°C | 5.0V | 89.8 | 302.0 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35.0 | µA |
| | | 85°C | 5.0V | 87.8 | 300.0 | |

**Notes:**

1. These are based on characterization.
2. See the Ordering Information section to determine the parts running at 64 MHz.

## 45.8. Wake-Up Time

Conditions:

- $V_{DD}$ = 5.0V
- CPU clock = OSC48M @8Mhz
- 0 Wait-state
- Cache enabled
- Flash in WAKEUPINSTANT mode (NVMCTRL.CTRLB.SLEEPPRM=1)

CPU sets an IO by writing PORT->IOBUS without jumping in an interrupt handler (Cortex M0+ register PRIMASK=1). The wakeup time is measured between the edge of the wakeup input signal and the edge of the GPIO pin.

**Table 45-13.** Wake-up Timings

| Sleep Mode | Typ. | Unit |
|---|---|---|
| Idle (idle0 or idle2) | 15.2 | µs |
| Standby | 48 | µs |

## 45.9. I/O Pin Characteristics

There are two different pin types with two different speeds: Normal and High Sink(2). The Drive Strengh bit is located in the Pin Configuration register PORT (PORT.PINCFG.DRVSTR).


© 2026 Microchip Technology Inc. and its subsidiaries

The pins with $I^2C$ alternative mode available are compliant with $I^2C$ specifications. All $I^2C$ pins support Standard (Sm), Fast (Fm), Fast plus (Fm+) and High speed (Hs) modes. The available $I^2C$ pins are listed in the I/O Multiplexing section.

The RESET_N pin has the same properties as standard GPIOs.

**Table 45-14.** I/O Pins Common Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input low-level voltage | VDD=2.7-4.5V | - | - | 0.3*VDD | V |
| | | VDD=4.5-5.5V | - | - | 0.3*VDD | |
| $V_{IH}$ | Input high-level voltage | VDD=2.7-4.5V | 0.7*VDD | - | - | |
| | | VDD=4.5-5.5V | 0.7*VDD | - | - | |
| $V_{OL}$ | Output low-level voltage | VDD>2.7V, $I_{OL}$ max | - | 0.1*VDD | 0.2*VDD | |
| $V_{OH}$ | Output high-level voltage | VDD>2.7V, $I_{OH}$ max | 0.8*VDD | 0.9*VDD | - | |
| $R_{PULL}$ | Pull-up - Pull-down resistance | All pins | 20 | 40 | 60 | kΩ |
| $I_{LEAK}$ | Input leakage current | Pull-up resistors disabled | -1 | - | 1 | µA |

**Table 45-15.** I/O Pins Maximum Output Current

| Symbol | Parameter | Conditions | Normal pins | High Sink pins | Normal pins | High Sink pins | Units |
|---|---|---|---|---|---|---|---|
| | | | DRVSTR=0 | | DRVSTR=1 | | |
| $I_{OL}$ | Maximum output low-level current | VDD=2.7V-4.5V | 2.5 | 5 | 5 | 10 | mA |
| | | VDD=4.5V-5.5V | 5 | 10 | 10 | 20 | |
| $I_{OH}$ | Maximum output high-level current | VDD=2.7V-4.5V | 1.5 | 3 | 3 | 6 | |
| | | VDD=4.5V-5.5V | 3 | 6 | 6 | 12 | |

**Table 45-16.** I/O Pins Dynamic Characteristics [(1)]

| Symbol | Parameter | Conditions | Normal pins | High Sink pins | Normal pins | High Sink pins | Units |
|---|---|---|---|---|---|---|---|
| | | | DRVSTR=0 | | DRVSTR=1 | | |
| $t_{RISE}$ | Maximum rise time | VDD = 5.0V, load = 20pF | 15 | 12 | 8 | 7 | ns |
| $t_{FALL}$ | Maximum fall time | VDD = 5.0V, load = 20pF | 14 | 11 | 7 | 7 | |

**Note:**

1. These values are based on simulation and are not covered by test limits in production or characterization.
2. The following pins are High-Sink pins and these pins have different properties than normal pins: PA10, PA11, PB10, PB11.

**Related Links**

I/O Multiplexing and Considerations

PINCFG

Pin Configuration

## 45.10. Analog Characteristics

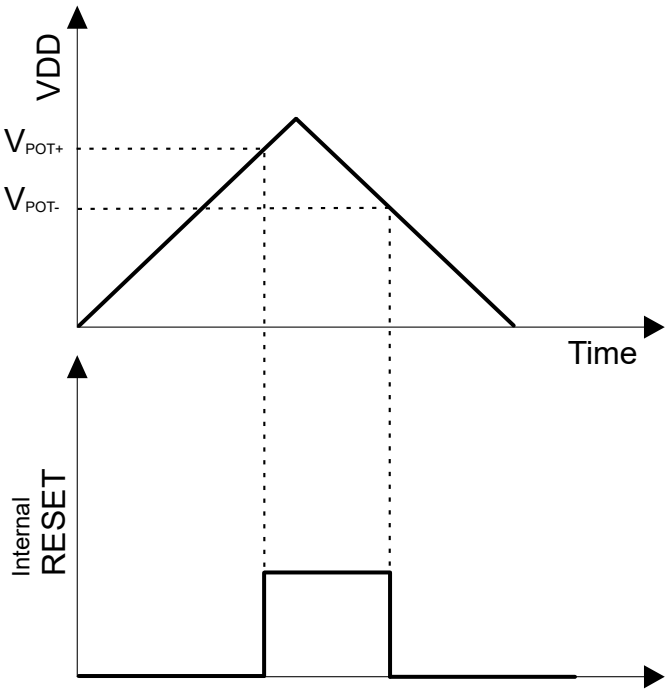
### 45.10.1. POR - Power On Reset Characteristics

**Table 45-17.** POR Characteristics

| Symbol | Parameters | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $V_{POT+}$ | Voltage threshold Level on Vddin rising | - | 2.55 | - | V |
| $V_{POT-}$ | Voltage threshold Level on Vddin falling | 1.53 | 1.75 | 1.97 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 45-1.** POR Operating Principle




## 45.10.2. BODVDD - Brown Out Detector Characteristics

Refer to NVM User Row Mapping for the BODVDD default value settings. These values are based on simulation and are not covered by test limits in production or characterization.

**Figure 45-2.** BODVDD Hysteresis OFF

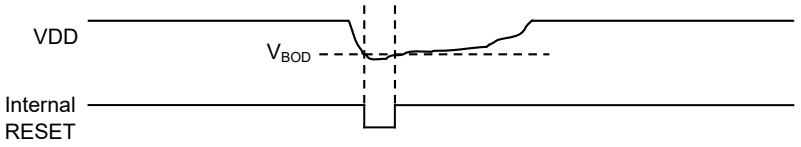



**Figure 45-3.** BODVDD Hysteresis ON

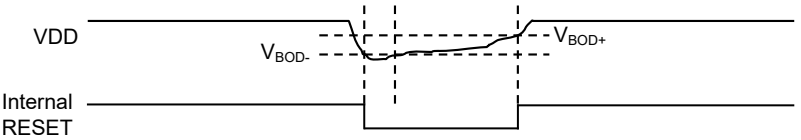



**Table 45-18.** BODVDD Characteristics[2]

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| VBOD+[1] | BODVDD high threshold Level | VDD level, BOD setting = 8 (default) | - | 2.86 | 2.97 | V |
| | | VDD level, BOD setting = 9 | - | 2.92 | 3.0 | |
| | | VDD level, Bod setting = 44 | - | 4.57 | 4.81 | |
| VBOD- / VBOD[1] | BODVDD low threshold Level | VDD level, BOD setting = 8 (default) | 2.71 | 2.8 | 2.89 | |
| | | VDD level, BOD setting = 9 | 2.75 | 2.85 | 2.95 | |
| | | VDD level, BOD setting = 44 | 4.37 | 4.51 | 4.66 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-18.** BODVDD Characteristics[2] (continued)

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| | Step size | | - | 60 | - | mV |
| VHys[1] | Hysteresis (VBOD+ - VBOD-) BODVDD.LEVEL = 8 to 48 | VDD | 40 | - | 75 | mV |
| $T_{START}$[3] | Startup time | Time from enable to RDY | - | 3.1 | - | µs |

**Note:**

1. These values are based on characterization.
2. BODVDD in continuous mode.
3. This value is based on simulation and not covered by test or characterization.

**Table 45-19.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | IDLE, Mode CONT | VDD = 2.7V | Max 85°C Typ 25°C | 22.5 | 26.3 | µA |
| | | VDD = 5.0V | | 41.0 | 47.1 | |
| | IDLE, Mode SAMPL | VDD = 2.7V | | 0.1 | 1.2 | |
| | | VDD = 5.0V | | 0.1 | 1.2 | |
| | STANDBY, Mode SAMPL | VDD = 2.7V | | 0.8 | 1.6 | |
| | | VDD = 5.0V | | 3.5 | 4.6 | |

**Note:**

1. These values are based on characterization.

**Related Links**

NVM User Row Mapping

### 45.10.3. Voltage Regulator Characteristics

**Table 45-20.** Voltage Regulator Characteristics

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| VDDIN | Input voltage range | 2.7 | - | 5.5 | V |
| VDDCORE | DC calibrated output voltage | - | 1.23 | - | V |

**Table 45-21.** Decoupling Requirements

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Cin[1] | Input regulator capacitor | Tantalum or electrolitic dielectric | - | 10 | - | uF |
| | | ceramic dielectric X7R | - | 100 | - | nF |
| Cout | Output regulator capacitor | Tantalum or electrolitic dielectric | 0.8 | 1 | 1.2 | uF |
| | | ceramic dielectric X7R | - | 100 | - | nF |
| ESR Cout | External Series Resistance of Cout | | - | - | 0.5 | Ω |

**Note:**

1. It is recommended to use ceramic X7R, tantalum or electrolitic capacitors with low-series resistance. Refer to Power Supply Connections for a typical circuit connections.


© 2026 Microchip Technology Inc. and its subsidiaries

## 45.10.4. Analog-to-Digital Converter (ADC) Characteristics

**Table 45-22.** Operating Conditions[1]

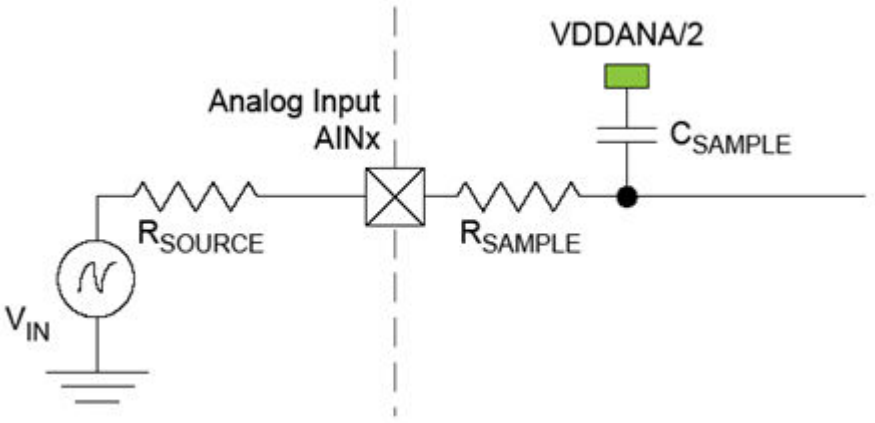
| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| Res | Resolution | | - | - | 12 | bits |
| Rs | Sampling rate[2] | SAMPLEN = 3 resolution 12 bit (CTRLC.RESSEL= 0) | 10 | - | 1000 | ksps |
| Nb_cycles | Differential mode Number of ADC clock cycles SAMPCTRL.OFFCOMP=1 | resolution 12 bit (CTRLC.RESSEL= 0) | - | 16 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | 14 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | 12 | | |
| | Differential mode Number of ADC clock cycles SAMPCTRL.OFFCOMP=0 SAMPLEN corresponds to the decimal value of SAMPCTRL.SAMPLEN[5:0] register | resolution 12 bit (CTRLC.RESSEL= 0) | - | SAMPLEN+13 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | SAMPLEN+11 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | SAMPLEN+9 | | |
| | Single-ended mode Number of ADC clock cycles SAMPCTRL.OFFCOMP=1 | resolution 12 bit (CTRLC.RESSEL= 0) | - | 16 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | 15 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | 13 | | |
| | Single-ended mode Number of ADC clock cycles SAMPCTRL.OFFCOMP=0 SAMPLEN corresponds to the decimal value of SAMPCTRL.SAMPLEN[5:0] register | resolution 12 bit (CTRLC.RESSEL= 0) | - | SAMPLEN+13 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | SAMPLEN+12 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | SAMPLEN+10 | | |
| $f_{adc}$ | ADC Clock frequency | - | 160 | - | 16000 | kHz |
| Ts | Sampling time | SAMPCTRL.OFFCOMP = 1 | 250 | - | 25000 | ns |
| | | SAMPCTRL.OFFCOMP = 0 | (SAMPLEN+1)/ fadc | - | - | s |
| | Sampling time with INTREF as input | | 10 | - | - | us |
| $V_{CNV}$ | Conversion range | Differential mode | -VREF | - | +VREF | V |
| | Conversion range | Single-ended mode | 0 | - | VREF | |
| Vref | Reference input | - | 2 | - | VDDANA-0.6 | V |
| Vin | Input channel range | - | 0 | - | VDDANA | V |
| Vcmin | Input common mode voltage | CTRLC.R2R = 1 | 0.2 | - | VREF-0.2 | V |
| | | CTRLC.R2R = 0 | VREF/2-0.2 | - | VREF/2+0.2 | V |
| CSAMPLE | Input sampling capacitance | | - | - | 3.2 | pF |
| RSAMPLE | Input sampling on-resistance | For a sampling rate at 1 Msps | - | 1000 | 1715 | Ω |
| Rref | Reference input source resistance | | 0 | - | 1000 | kΩ |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. These values are based on simulation and not covered by test limits in production or characterization.
2. Sampling rate (in samples per second) is equal to (fadc/Nb_cycles).

**Figure 45-4.** ADC Analog Input AINx




The minimum sampling time $t_{samplehold}$ for a given $R_{source}$ can be calculated using this formula:

$$t_{samplehold} \geq \left(R_{sample} + R_{source}\right) \times C_{sample} \times \left(n + 2\right) \times \ln\left(2\right)$$

For 12-bit accuracy:

$$t_{samplehold} \geq \left(R_{sample} + R_{source}\right) \times C_{sample} \times 9.7$$

**Table 45-23.** Differential Mode (1)

| Symbol | Parameter | Conditions | Measurement | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min. | Typ | Max. | |
| ENOB | Effective Number of bits | Vddana = 5.0V Vref = Vddana | 10.5 | 10.8 | 11.3 | bits |
| | | Vddana = 2.7V Vref = 2.0V | 9.9 | 10.0 | 11.2 | |
| TUE | Total Unadjusted Error Condition: R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | 4.2 | 6.7 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | 4.8 | 7.9 | |
| INL | Integral Non Linearity | Vddana = 5.0V Vref = Vddana | - | +/-1.5 | +/-3 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | +/-3.2 | +/-3.9 | |
| DNL | Differential Non Linearity | Vddana = 5.0V Vref = Vddana | - | -0.8/+1.1 | -1/+1.9 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | -0.9/+1.3 | -1/+2.1 | |
| GE | Gain Error | Vddana = 2.7V Vref = 2.0V | - | +/-18 | +/-57 | mV |
| | | Vddana = 5.0V Vref = 4.096V | - | +/-41 | +/-100 | |
| | | Vddana = 3.0V Vref = Vddana | - | +/-17 | +/-66 | |
| | | Vddana = 5.0V Vref = Vddana | - | +/-39 | +/-81 | |
| TCg | Gain Drift | Vddana = 5.0V Vref = Vddana | -250 | -210 | -170 | uV/°C |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-23.** Differential Mode [1] (continued)

| Symbol | Parameter | Conditions | Measurement | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min. | Typ | Max. | |
| OE | Offset Error | Vddana = 2.7V Vref = 2.0V | - | +/-1.4 | +/-11 | mV |
| | | Vddana = 5.0V Vref = 4.096V | - | +/-6 | +/-18 | |
| | | Vddana = 3.0V Vref=Vddana | - | +/-2 | +/-9 | |
| | | Vddana = 5.0V Vre f= Vddana | - | +/-0.2 | +/-23 | |
| Tco | Offset Drift | Vddana = 5.0V Vref = Vddana | 20 | 80 | 120 | uV/°C |
| SFDR | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana = 5.0V Vref = Vddana | 71 | 75 | 81 | dB |
| SINAD | Signal to Noise and Distortion ratio | | 65 | 67 | 68 | |
| SNR | Signal to Noise ratio | | 67 | 68 | 69 | |
| THD | | | -77 | -74 | -70 | |
| | Noise RMS | External Reference voltage | - | 0.5 | 2.0 | mV |

**Note:**

1. These values are based on characterization and not covered by test limits in production.

**Table 45-24.** Single-Ended Mode [1]

| Symbol | Parameter | Conditions | Measurement | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min. | Typ | Max. | |
| ENOB | Effective Number of bits | Vddana = 5.0V Vref=Vddana | 9.1 | 9.7 | 10 | bits |
| | | Vddana = 2.7V Vref=2.0V | 9.0 | 9.2 | 10 | |
| TUE | Total Unadjusted Error Condition: R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | 18.4 | 26.5 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | 30.4 | 53.8 | |
| INL | Integral Non Linearity | Vddana = 5.0V Vref = Vddana | - | +/-2.2 | +/-4 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | +/-4.1 | +/-6 | |
| DNL | Differential Non Linearity | Vddana = 5.0V Vref=Vddana | - | -0.8/+1 | -1/+1.9 | LSB |
| | | Vddana = 2.7V Vref = 2.0V | - | -1/+1.1 | -1/+2.4 | |
| GE | Gain Error | Vddana = 2.7V Vref = 2.0V | - | +/-13 | +/-28 | mV |
| | | Vddana = 5.0V Vref = 4.096V | - | +/-26 | +/-52 | |
| | | Vddana = 3.0V Vref = Vddana | - | +/-14 | +/-24 | |
| | | Vddana = 5.0V Vref = Vddana | - | +/-22 | +/-42 | |
| TCg | Gain Drift | Vddana = 5.0V Vref = Vddana | -170 | -140 | -80 | uV/°C |
| OE | Offset Error | Vddana = 2.7V Vref = 2.0V | - | +/-2.2 | +/-21 | mV |
| | | Vddana = 5.0V Vref = 4.096V | - | +/-2.3 | +/-61 | |
| | | Vddana = 3.0V Vref=Vddana | - | +/-15 | +/-42 | |
| | | Vddana = 5.0V Vref = Vddana | - | +/-31 | +/-80 | |
| Tco | Offset Drift | Vddana = 5.0V Vref = Vddana | 160 | 180 | 210 | uV/°C |
| SFDR | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana = 5.0V Vref = Vddana | 69 | 71 | 73 | dB |
| SINAD | Signal to Noise and Distortion ratio | | 57 | 60 | 61 | |
| SNR | Signal to Noise ratio | | 57 | 61 | 61 | |
| THD | | | -72 | -70 | -66 | |
| | Noise RMS | External Reference voltage | - | 0.7 | 2.0 | mV |


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:**

1. These values are based on characterization and not covered by test limits in production.

**Table 45-25.** Power Consumption (1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Differential mode | fs = 1Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | Max 85°C Typ 25°C | 905 | 1021 | uA |
| | | fs = 1Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref = 5.5V | | 1144 | 1403 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 381 | 460 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref = 5.5V | | 609 | 857 | |
| | Single Ended mode | fs = 1Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref = 5.5V | Max 85°C Typ 25°C | 984 | 1077 | uA |
| | | fs = 1Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref=5.5V | | 1178 | 1444 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref = 5.5V | | 437 | 528 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = Vref = 5.5V | | 635 | 888 | |

**Note:**

1. These values are based on characterization.

## 45.10.5. Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics

**Table 45-26.** Operating Conditions(1)

| Symbol | Parameters | Conditions | | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|---|
| Res | Resolution | Differential mode | | - | 16 | - | bits |
| | | Single-Ended mode (3) | | - | 15 | - | |
| CLK_SDADC | Sampling Clock Speed | - | | 1 | - | 6 | MHz |
| CLK_SDADC_FS | Conversion rate | - | | CLK_SDADC/4 | | | |
| fs | Output Data Rate | Free Running mode | | CLK_SDADC_FS / OSR | | | |
| | | Single Conversion mode SKPCNT = N | | (CLK_SDADC_FS / OSR) x (N+1) | | | |
| OSR | Oversampling ratio | Differential mode | | 64 | 256 | 1024 | Cycles |
| Vin | Input Conversion range | VREF<VDDANA-0.3V | Differential mode Gaincorr = 0x1 | - VREF | - | VREF | V |
| | | | Single-Ended mode Gaincorr = 0x1(3) | 0 | - | VREF | |
| | | VREF> = VDDANA-0.3V | Differential mode Gaincorr = 0x1 | -0.7xVREF | - | 0.7xVREF | V |
| | | | Single-Ended mode Gaincorr = 0x1(3) | 0 | - | 0.7xVREF | |
| Vref | Reference Voltage range | | | 1 | - | VDDANA | V |
| Vcom | Common mode voltage | Differential mode | | 0 | - | VDDANA | V |
| Cin | Input capacitance | | | 0.425 | 0.5 | 0.575 | pF |
| Zin | Input impedance | Differential mode | | 1/(Cin x CLK_SDADC_FS) | | | kΩ |
| | | Single-Ended mode (3) | | 1/(Cin x CLK_SDADC_FS x 2) | | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-26.** Operating Conditions[1] (continued)

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| | Input anti-alias filter recommendation [2] | Rext | - | 1.0 | - | kΩ |
| | | Cext | 3.3 | - | 10 | nF |

**Note:**

1. These values are based on simulation and not covered by test or characterization.
2. External anti-alias filter must be placed in front of each SDADC input to ensure high-frequency signals to not alias into measurement bandwidth. Use capacitors of X5R type for DC measurement, or capacitors of COG or NPO type for AC measurement.
3. This mode corresponds to a differential mode where the selected AINNx pin is externally grounded.

**Table 45-27.** SDADC DC Performance: Differential Input Mode [1][2]

| Symbol | Parameters | Conditions (2) | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| INL | Integral Non Linearity | CLK_SDADC = 6MHz; VREF = 1.2V | - | +/-1.3 | +/-2 | LSB |
| | | CLK_SDADC = 6MHz; INT VREF = 5.5V | - | +/-5.3 | +/-11 | |
| DNL | Differential Non Linearity | CLK_SDADC = 6MHz; VREF = 1.2V | - | +1.4/-1 | +1.3/-1 | LSB |
| | | CLK_SDADC = 6MHz; INT VREF = 5.5V | - | +2.1/-1 | +1.7/-1 | |
| Off | Offset Error | CLK_SDADC = 6MHz; VREF = 1.2V | - | +/-0.6 | +/-3 | mV |
| | | CLK_SDADC = 6MHz; INT VREF = 5.5V | - | +/-3.9 | +/-6 | |
| Tco | Offset Error Drift | CLK_SDADC = 6MHz; VREF = 1.2V | 2.3 | 3.6 | 5.0 | uV/°C |
| Eg | Gain Errors | CLK_SDADC = 6MHz; VREF = 1.2V | - | +/-1.1 | +/-3.7 | % |
| | | CLK_SDADC = 6MHz; INT VREF = 5.5V | - | +/-1.1 | +/-3.4 | |
| TCg | Gain Drift | CLK_SDADC = 6MHz; VREF = 1.2V | -10.9 | 1.2 | 7.6 | ppm/°C |
| Input noise rms | AC Input noise rms | OSR = 256 | - | 0.08 | 0.12 | mVrms |

**Note:**

1. These values are based on characterization.
2. OSR = 256, Chopper ON.

**Table 45-28.** SDADC AC Performance: : Differential Input Mode[1]

| Symbol | Parameters | Conditions (2) | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| ENOB | Effective Number Of Bits | Ext ref = 1.2V | 13.5 | 14.2 | 14.4 | bits |
| | | Int Ref = 5.5V | 11 | 11.2 | 11.4 | |
| DR | Dynamic Range | Ext ref = 1.2V | 89 | 91 | 92 | dB |
| | | Int Ref = 5.5V | 83 | 92 | 96 | |
| SNR | Signal to Noise Ratio | Ext ref = 1.2V | 84 | 88 | 89 | dB |
| | | Int Ref = 5.5V | 77 | 79 | 80 | |
| SINAD | Signal to Noise + Distortion Ratio | Ext ref = 1.2V | 83 | 87 | 89 | dB |
| | | Int Ref = 5.5V | 68 | 69 | 71 | |
| THD | Total Harmonic Distortion | Ext ref = 1.2V | -105 | -100 | -92 | dB |
| | | Int Ref = 5.5V | -70 | -69 | -69 | |

**Note:**

1. These values are based on characterization.
2. OSR = 256, Fs = 6 MHz, Fin = 13 kHz.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-29.** Power consumption (1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Power consumption | CTLSDADC = 0x0 External Ref - VDDANA = 5.5V Vref = 2V Ref buf on SCLK_SDADC = 6 MHz | Max 85°C Typ 25°C | 644 | 695 | µA |
| | | CTLSDADC = 0x0 Internal Ref - VDDANA = Vref = 5.5V Ref buf off SCLK_SDADC = 6 MHz | | 605 | 636 | |

**Note:**

1. These are based on characterization.

## 45.10.6. Digital-to-Analog Converter (DAC) Characteristics

**Table 45-30.** Operating Conditions(1)

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| RES | Input resolution | - | - | - | 10 | Bits |
| VDDANA | Analog supply voltage | - | 2.7 | - | 5.5 | V |
| AVREF | External reference voltage | - | 1 | - | VDDANA - 0.6 | V |
| | INTREF | VREF.SEL = 0x0 | - | 1.024 | - | V |
| | | VREF.SEL = 0x2 | - | 2.048 | - | |
| | | VREF.SEL = 0x3 | - | 4.096 (2) | - | |
| | VDDANA | - | - | VDDANA | - | V |
| | Linear output voltage range | - | 0.05 | - | VDDANA - 0.05 | V |
| | Minimum resistive load | - | 5 | - | - | kΩ |
| | Maximum capacitance load | - | - | - | 100 | pF |

**Note:**

1. These values are based on simulation and not covered by test or characterization.
2. For VDDANA > 4.5V.

**Table 45-31.** Clock and Timing(1)

| Symbol | Parameter | Conditions | | Max. | Units |
|---|---|---|---|---|---|
| | Conversion rate | Cload = 100 pF Rload > 5kΩ | Normal mode | 350 | ksps |
| | | | For DDATA = ±1 | 1000 | |
| | Startup time | | | 3 | µs |

**Note:**

1. These values are based on simulation and not covered by test limits in production or characterization.

**Table 45-32.** Accuracy Characteristics(1)

| Symbol | Parameter | Conditions | | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INL | Integral non-linearity | VREF = Ext 2.0V | VDD = 2.7V | +/-0.7 | +/-1.3 | LSB |
| | | | VDD = 5.5V | +/-0.5 | +/-0.6 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.6 | +/-0.9 | |
| | | | VDD = 5.5V | +/-0.4 | +/-0.6 | |
| | | VREF = 1.024V INTREF | VDD = 2.7V | +/-1 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.5 | +/-3.5 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-32.** Accuracy Characteristics[1] (continued)

| Symbol | Parameter | Conditions | | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| DNL | Differential non-linearity | VREF = Ext 2.0V | VDD = 2.7V | +/-0.3 | +/-0.4 | LSB |
| | | | VDD = 5.5V | +/-0.4 | +/-0.5 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.2 | +/-0.3 | |
| | | | VDD = 5.5V | +/-0.2 | +/-0.3 | |
| | | VREF = 1.024V INTREF | VDD = 2.7V | +/-1.0 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.4 | +/-3.5 | |
| GE | Gain error | Ext. VREF | | +/-8 | +/-20 | mV |
| OE | Offset error | Ext. VREF | | +/-4 | +/-20 | mV |

**Note:**

1. These values are based on characterization and not covered by test limits in production.

**Table 45-33.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | DC supply current | Output buffer On VREF = VDDANA = 5.0V | Max 85°C Typ 25°C | 318 | 404 | µA |
| | | Output buffer Off VREF = VDDANA=5.0V | | 74 | 81 | |

**Note:**

1. These values are based on characterization.

## 45.10.7. Analog Comparator Characteristics

**Table 45-34.** Analog Comparator Characteristics

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| PNIVR | Positive and Negative input range voltage | | 0 | - | VDDANA | V |
| ICMR | Input common mode range | | 0 | - | VDDANA | V |
| Off [1][2] | Offset | Low power COMPCTRLn.SPEED = 0x0 | -36 | +0.4/+2 | +48 | mV |
| | | High speed COMPCTRLn.SPEED = 0x3 | -12 | -0.1/+1 | +20 | |
| $V_{HYS}$ [1][3] | Hysteresis | Low Power COMPCTRLn.SPEED = 0x0 | 25 | 100 | 248 | mV |
| | | High speed COMPCTRLn.SPEED = 0x3 | 29 | 100 | 190 | |
| $T_{PD}$ [1, 4] | Propagation Delay Vcm=Vddana/2 Vin = ±100mV overdrive from Vcm | Low power COMPCTRLn.SPEED = 0x0 | - | 133 | 237 | ns |
| | | High speed COMPCTRLn.SPEED = 0x3 | - | 38 | 73 | |
| $T_{START}$ [1] | Startup time | Low power COMPCTRLn.SPEED = 0x0 | - | 6.5 | 8.5 | µs |
| | | High speed COMPCTRLn.SPEED = 0x3 | - | 2 | 3 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-34.** Analog Comparator Characteristics (continued)

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $V_{SCALE}$ (1) | INL | | - | 0.2 | - | LSB |
| | DNL | | - | 0.023 | - | |
| | Offset Error | | - | 0.049 | - | |
| | Gain Error | | - | 0.064 | - | |

**Notes:**

1. These are based on characterization.
2. Hysteresis disabled.
3. Hysteresis enabled.
4. Tpd is measured from Vin transition to ACOUT (AC direct output) toggle. It takes into account only analog propagation delay.

**Table 45-35.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDDANA | Current consumption - $V_{CM}$= VDDANA/2<br>±100 mV overdrive from Vcm<br>Voltage scaler disabled | COMPCTRLn.SPEED = 0x0<br>VDDANA = 5.0V | Max 85°C<br>Typ 25°C | 10 | 16 | µA |
| | | COMPCTRLn.SPEED = 0x3<br>VDDANA = 5.0V | | 39 | 58 | |
| | Current consumption Voltage scaler only | VDDANA = 5.0V | | 43 | 60 | |

**Note:**

1. These values are based on characterization.

## 45.10.8. Voltage Reference Characteristics

**Table 45-36.** Voltage Reference Characteristics(1)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INTREF | ADC, SDADC, DAC, AC Internal reference | nom. 1.024V<br>VDDANA = 5.0V<br>Ta = 25°C | 1.003 | 1.024 | 1.045 | V |
| | | nom. 2.048V<br>VDDANA = 5.0V<br>Ta = 25°C | 2.007 | 2.048 | 2.089 | |
| | | nom. 4.096V<br>VDDANA = 5.0V<br>Ta = 25°C | 4.014 | 4.096 | 4.178 | |
| | Reference temperature coefficient | Drift over [-40, +25]°C | - | -0.025, 0.04 | - | %/°C |
| | | Drift over [+25, +85]°C | - | -0.015, 0.03 | - | |
| | | Drift over [+25, +105]°C | - | -0.015, 0.03 | - | |
| | Reference supply coefficient | Drift over [2.7, 5.5]V | - | -0.2, 0.3 | - | %/V |

**Note:**

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

## 45.10.9. Temperature Sensor Characteristics

**Table 45-37.** Temperature Sensor Characteristics[1]

| Parameter | Condition | Min. | Max. | Unit |
|---|---|---|---|---|
| Accuracy | [0,60]°C | -11.3 | 6.2 | °C |
| | [-40,85]°C | -14.6 | 10.5 | |

**Note:**

1. These values are based on characterization. Data has been obtained by averaging 10 TSENS acquisitions per measurement.

## 45.10.10. PTC Characteristics

**Table 45-38.** Sensor Load Capacitance [1]

| Symbol | Mode | PTC channel | Max. | Units |
|---|---|---|---|---|
| Cload | Self-capacitance | Y0 | 18 | pF |
| Cload | Self-capacitance | Y1 | 25 | pF |
| | | Y2 | | |
| | | Y3 | | |
| | | Y4 | | |
| | | Y5 | | |
| | | Y6 | | |
| | | Y7 | | |
| | | Y8 | | |
| | | Y9 | | |
| | | Y10 | | |
| | | Y11 | | |
| | | Y12 | | |
| | | Y13 | | |
| | | Y14 | | |
| | | Y15 | | |
| | | Y16 | | |
| | | Y17 | | |
| | | Y18 | | |
| | | Y19 | | |
| | | Y20 | | |
| | | Y21 | | |
| | | Y22 | | |
| | | Y23 | | |
| | | Y24 | | |
| | | Y25 | | |
| Cload | Self-capacitance | Y26 | 22 | pF |
| | | Y27 | | |
| Cload | Self-capacitance | Y28 | 25 | pF |
| | | Y29 | | |
| | | Y30 | | |
| | | Y31 | | |
| Cload | Mutual-capacitance | All | 31 | pF |


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:**

1. Capacitance load that the PTC circuitry can compensate for each channel.

**Table 45-39.** Analog Gain Settings (1) (2)

| Symbol | Setting | Average |
|---|---|---|
| Gain | GAIN_1 | 1 |
| | GAIN_2 | 2.1 |
| | GAIN_4 | 4.3 |
| | GAIN_8 | 9.9 |
| | GAIN_16 | - |
| | GAIN_32 | - |

**Notes:**

1. Analog Gain is a parameter of the QTouch Library. Refer to the QTouch Library Peripheral Touch Controller User Guide for more information.
2. GAIN_16 and GAIN_32 settings are not recommended; otherwise, the PTC measurements might become unstable.

#### 45.10.10.1. PTC Power Consumption

The values given in the table below are measured values of power consumption under the following conditions:

- **Operating conditions**
  - VDD = 5.0V
- **Clocks**
  - OSC48M used as main clock source, running undivided at 48 MHz
  - CPU is running on flash with 2 wait states, at 48 MHz
  - PTC running at 4 MHz
- **PTC configuration**
  - Mutual-capacitance mode
  - One-touch channel
- **System configuration**
  - Standby Sleep mode enabled
  - RTC running on ULP32K: used to define the PTC scan rate, through the event system
  - Drift Calibration disabled: no interrupts, PTC scans are performed in standby mode
  - Drift Calibration enabled: RTC interrupts (wakeup) the CPU to perform PTC scans. PTC drift calibration is performed every 1.5 sec.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-40.** Power Consumption [1]

| Symbol | Parameters | Drift Calibration | PTC scan rate | Oversamples | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|---|
| IDD | Current Consumption | Disabled | 10 | 4 | Max 85°C Typ 25°C | 24 | 308 | µA |
| | | | | 16 | | 43 | 330 | |
| | | | 50 | 4 | | 17 | 301 | |
| | | | | 16 | | 22 | 307 | |
| | | | 100 | 4 | | 16 | 301 | |
| | | | | 16 | | 19 | 304 | |
| | | | 200 | 4 | | 16 | 301 | |
| | | | | 16 | | 17 | 303 | |
| | | Enabled | 10 | 4 | | 30 | 319 | |
| | | | | 16 | | 50 | 342 | |
| | | | 50 | 4 | | 20 | 306 | |
| | | | | 16 | | 24 | 311 | |
| | | | 100 | 4 | | 18 | 305 | |
| | | | | 16 | | 21 | 308 | |
| | | | 200 | 4 | | 17 | 304 | |
| | | | | 16 | | 19 | 305 | |

**Note:**

1. These values are based on characterization.

## 45.11. NVM Characteristics

**Table 45-41.** NVM Max Speed Characteristics

| CPU $F_{MAX}$ (MHz) | 0WS | 1WS | 2WS |
|---|---|---|---|
| VDD>2.7V | 19 | 38 | 64 |
| VDD>4.5V | 20 | 38 | 64 |

**Table 45-42.** NVM Timing Characteristics

| Symbol | Parameter | Max. | Units |
|---|---|---|---|
| $t_{FPP}$ | Page Write [1] | 2.5 | ms |
| $t_{FRE}$ | Row erase [1] | 6 | ms |
| $t_{FSE}$ | Chip erase [1] | 260 | ms |

**Note:**

1. These values are based on simulation, and are not covered by test or characterization. For this Flash technology, a maximum number of 8 consecutive writes is allowed per row. Once this number is reached, a row erase is mandatory.

**Table 45-43.** Flash Endurance and Data Retention

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Ret_{NVM25k}$ | Retention after up to 25k | Average ambient 55°C | 10 | 50 | Years |
| $Ret_{NVM2.5k}$ | Retention after up to 2.5k | Average ambient 55°C | 20 | 100 | Years |
| $Ret_{NVM100}$ | Retention after up to 100 | Average ambient 55°C | 25 | >100 | Years |
| $Cyc_{NVM}$ | Cycling Endurance[1] | -40°C < $T_A$ < 85°C | 25k | - | Cycles |

**Note:**

1. An endurance cycle is a write and an erase operation.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-44.** EEPROM Emulation(1) Endurance and Data Retention

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Ret_{EEPROM100k}$ | Retention after up to 100k | Average ambient 55°C | 10 | 50 | Years |
| $Ret_{EEPROM10k}$ | Retention after up to 10k | Average ambient 55°C | 20 | 100 | Years |
| $Cyc_{EEPROM}$ | Cycling Endurance(2) | -40°C < Ta < 85°C | 100k | - | Cycles |

**Note:**

1. The EEPROM emulation is a software emulation described in the Application Note “AT03265”.
2. An endurance cycle is a write and an erase operation.

**Table 45-45.** Flash erase and programming current

| Symbol | Parameter | Typ. | Units |
|---|---|---|---|
| $I_{DDIN}$ | Maximum Current (peak) during whole programming or erase operation | 10 | mA |

## 45.12. Oscillator Characteristics

### 45.12.1. Crystal Oscillator (XOSC) Characteristics

#### 45.12.1.1. Digital Clock Characteristics

The following table describes the characteristics for the oscillator when a digital clock is applied on XIN.

**Table 45-46.** Digital Clock Characteristics

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{CPXIN}$ | XIN clock frequency | Digital mode | - | - | 48 | MHz |
| $DC_{XIN}$(1) | XIN clock duty cycle | Digital mode | 40 | 50 | 60 | % |

1. These are based on simulation. These values are not covered by test or characterization

#### 45.12.1.2. Crystal Oscillator Characteristics

The following table describes the characteristics for the oscillator when a crystal is connected between XIN and XOUT as shown in Figure 45-5. The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of $C_L$ can be found in the crystal data sheet. The capacitance of the external capacitors ($C_{LEXT}$) can then be computed as follows:

$C_{LOAD} = \{ ( [C_{XIN} + C_{LEXT}] * [C_{XOUT} + C_{LEXT}] ) / [C_{XIN} + C_{LEXT} + C_{LEXT} + C_{XOUT}] \} + C_{STRAY}$.

Where CSTRAY is the capacitance of the PCB.

For $C_{XIN} \neq C_{XOUT}$, but with a difference smaller than 4pF, we can consider $C_{XIN} \approx C_{XOUT} \approx C_{Xeff} = ( C_{XIN} + C_{XOUT}) / 2$.

This results in a simplified formula for computing $C_{LEXT}$, as follows:

$C_{LEXT} = 2*CL - 2*C_{STRAY} - C_{Xeff}$


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 45-5.** Oscillator Connection

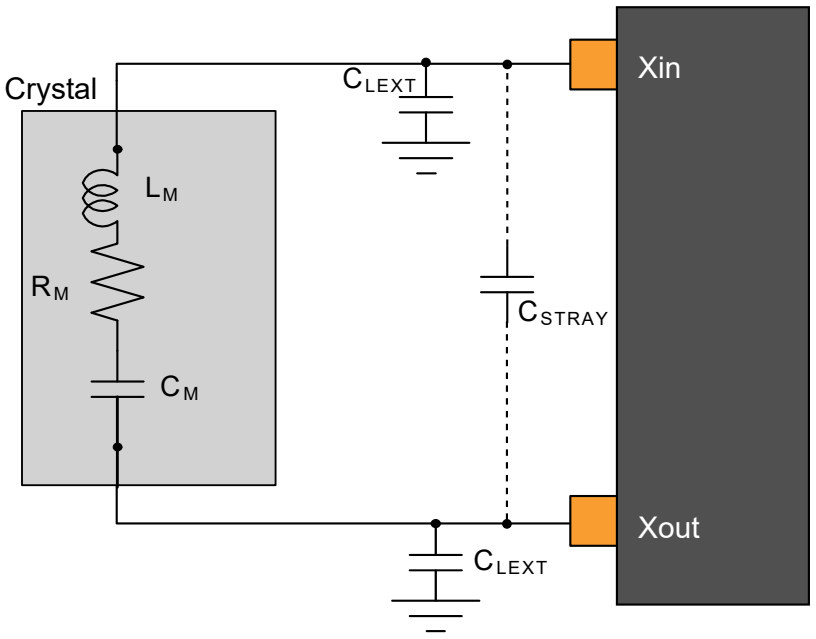



**Table 45-47.** Multi Crystal Oscillator Electrical Characteristics (1)

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| Fout | Crystal oscillator frequency | | 0.4 | - | 32 | MHz |
| CL | Crystal Load | F = 0.455 MHz | - | - | 100 | pF |
| | | F = 2 MHz | - | - | 20 | |
| | | F = 4 MHz | - | - | 20 | |
| | | F = 8 MHz | - | - | 20 | |
| | | F = 16 MHz | - | - | 20 | |
| | | F = 32 MHz | - | - | 18 | |
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 0.455 MHz<br>CL = 100pF<br>XOSC.GAIN = 0 | - | - | 443 | Ω |
| | | F = 2MHz<br>CL=20pF<br>XOSC.GAIN=0 | - | - | 383 | |
| | | F = 4MHz<br>CL=20pF<br>XOSC.GAIN=1 | - | - | 218 | |
| | | F = 8MHz<br>CL=20pF<br>XOSC.GAIN=2 | - | - | 114 | |
| | | F = 16MHz<br>CL=20pF<br>XOSC.GAIN=3 | - | - | 61 | |
| | | F = 32MHz<br>CL=18pF<br>XOSC.GAIN=4 | - | - | 41 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-47.** Multi Crystal Oscillator Electrical Characteristics [1] (continued)

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| Cxin | Parasitic load capacitor | | - | 5.9 | - | pF |
| Cxout | | | - | 3.1 | - | |
| Tstart | Startup time | F = 2MHz<br>CL=20pF<br>XOSC.GAIN=0 | - | 12.3 | 35.3 | KCycles |
| | | F = 4MHz<br>CL=20pF<br>XOSC.GAIN=1 | - | 8.2 | 21.4 | |
| | | F = 8MHz<br>CL=20pF<br>XOSC.GAIN=2 | - | 6.2 | 14.3 | |
| | | F = 16MHz<br>CL=20pF<br>XOSC.GAIN=3 | - | 10.8 | 18.1 | |
| | | F = 32MHz<br>CL=18pF<br>XOSC.GAIN=4 | - | 8.7 | 15.4 | |

1. These are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-48.** Power Consumption (1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD | Current consumption | F = 2MHz<br>CL=20pF<br>XOSC.GAIN=0<br>VDD = 5.0V<br>AGC=OFF | Max 85°C<br>Typ 25°C | 150 | 202 | µA |
| | | AGC=ON | | 138 | 192 | |
| | | F = 4MHz<br>CL=20pF<br>XOSC.GAIN=1<br>VDD = 5.0V<br>AGC=OFF | | 220 | 288 | |
| | | AGC=ON | | 175 | 260 | |
| | | F = 8MHz<br>CL=20pF<br>XOSC.GAIN=2<br>VDD = 5.0V<br>AGC=OFF | | 350 | 416 | |
| | | AGC=ON | | 247 | 321 | |
| | | F = 16MHz<br>CL=20pF<br>XOSC.GAIN=3<br>VDD = 5.0V<br>AGC=OFF | | 663 | 843 | |
| | | AGC=ON | | 429 | 699 | |
| | | F = 32MHz<br>CL=18pF<br>XOSC.GAIN=4<br>VDD = 5.0V<br>AGC=OFF | | 1975 | 2329 | |
| | | AGC=ON | | 874 | 1181 | |

## 45.12.2. External 32.768 kHz Crystal Oscillator (XOSC32K) Characteristics

### 45.12.2.1. Digital Clock Characteristics

The following table describes the characteristics for the oscillator when a digital clock is applied on XIN32 pin.

**Table 45-49.** Digital Clock Characteristics

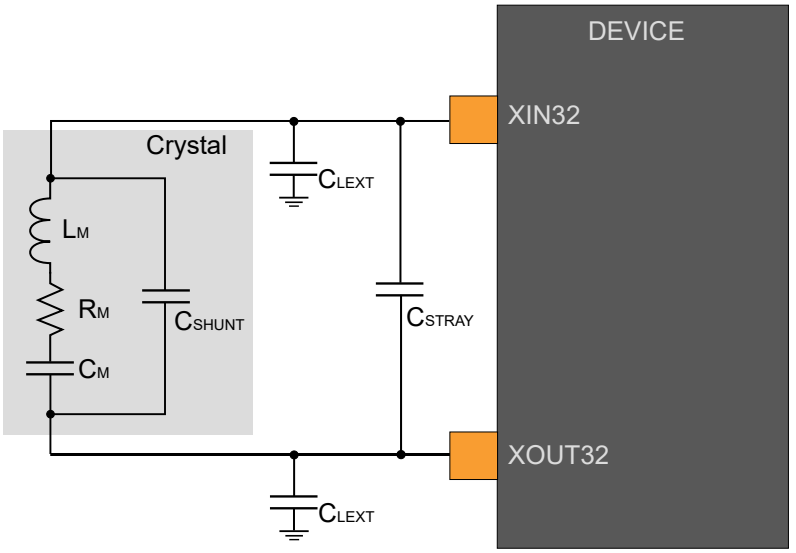
| Symbol | Parameter | Condition | Typ | Units |
|---|---|---|---|---|
| $f_{CPXIN32}$ | XIN32 clock frequency | Digital mode | 32.768 | kHz |
| $DC_{XIN32}$ | XIN32 clock duty cycle | Digital mode | 50 | % |

### 45.12.2.2. Crystal Oscillator Characteristics

The following table describes the characteristics for the oscillator when a crystal is connected between XIN32 and XOUT32.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 45-6.** Oscillator Connection




The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of CL can be found in the crystal data sheet. The capacitance of the external capacitors (CLEXT) can then be computed as follows:

$C_{LEXT}=2\ (C_L-C_{STRAY}-C_{SHUNT})$

where $C_{STRAY}$ is the capacitance of the pins and PCB and $C_{SHUNT}$ is the shunt capacitance of the crystal.

**Table 45-50.** 32.768 kHz Crystal Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$ (1) | Crystal oscillator frequency | | - | 32768 | - | Hz |
| $C_L$ (1) | Crystal load capacitance | | - | - | 12.5 | pF |
| $C_{SHUNT}$ (1) | Crystal shunt capacitance | | - | - | 1.75 | pF |
| Cm (1) | Motional capacitance | | - | 1.25 | - | fF |
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 32.768 kHz, $C_L$=12.5 pF | - | - | 79 | kΩ |
| Cxin32k | Parasitic capacitor load | | - | 2.9 | - | pF |
| Cxout32k | | | - | 3.2 | - | pF |
| Tstart | Startup time | F = 32.768 kHz, $C_L$=12.5 pF | - | 16 | 24 | Kcycles |

1. These are based on simulation. These values are not covered by test or characterization

**Table 45-51.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | VDD = 5.0V | Max 85°C<br>Typ 25°C | 1528 | 1720 | nA |

1. These are based on characterization.

## 45.12.3. Digital Phase Locked Loop (DPLL) Characteristics

**Table 45-52.** Fractional Digital Phase Locked Loop Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $f_{IN}$(1) | Input frequency | | 32 | | 2000 | KHz |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 45-52.** Fractional Digital Phase Locked Loop Characteristics (continued)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$(1) | DCO Output frequency | | 48 | | 96 | MHz |
| Jp(2) | Period jitter (Peak-Peak value) | $f_{IN}$ = 32 kHz, $f_{OUT}$ = 48 MHz | - | 1.5 | 3.0 | % |
| | | $f_{IN}$ = 32 kHz, $f_{OUT}$ = 64 MHz | - | 1.7 | 4.0 | |
| | | $f_{IN}$ = 32 kHz, $f_{OUT}$ = 96 MHz | - | 2.7 | 8.0 | |
| | | $f_{IN}$ = 2 MHz, $f_{OUT}$ = 48 MHz | - | 1.8 | 4.0 | |
| | | $f_{IN}$ = 2 MHz, $f_{OUT}$ = 64 MHz | - | 1.9 | 4.0 | |
| | | $f_{IN}$ = 2 MHz, $f_{OUT}$ = 96 MHz | - | 2.5 | 6.0 | |
| $t_{LOCK}$(2) | Lock Time | After startup, time to get lock signal. $f_{IN}$ = 32 kHz, $f_{OUT}$ = 96 MHz | - | 1.1 | 1.5 | ms |
| | | After startup, time to get lock signal. $f_{IN}$ = 2 MHz, $f_{OUT}$ = 96 MHz | - | 25 | 35 | µs |
| Duty(1) | Duty cycle | - | - | 50 | - | % |

1. These values are based on simulation, and are not covered by test limits in production or characterization.
2. These values are based on characterization.
3. DPLL jitter is sensitive to digital on-chip activity, which is application dependent.

**Table 45-53.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD | Current Consumption | Ck = 48 MHz, VDD = 5.0V | Max 85°C Typ 25°C | 536 | 612 | µA |
| | | Ck = 64 MHz, VDD = 5.0V | | 640 | 721 | |
| | | Ck = 96 MHz, VDD = 5.0V | | 865 | 970 | |

1. These values are based on characterization.

### 45.12.4. 32.768 kHz Internal oscillator (OSC32K) Characteristics

**Table 45-54.** 32.768 kHz RC Oscillator Characteristics

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$ | Output frequency | T =25°C, VDDANA = 5.0V | 32.112 | 32.768 | 33.423 | kHz |
| | | T =25°C, over [2.7, 5.5]V | 29.491 | 32.768 | 36.044 | |
| | | over [-40, 85]°C, over [2.7, 5.5]V | 25.559 | 32.768 | 37.355 | |
| $t_{STARTUP}$ | Startup time | | | 1 | 2 | cycle |
| Duty(1) | Duty Cycle | | | 50 | | % |

1. These are based on simulation. These values are not covered by test or characterization.

**Table 45-55.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD | Current consumption | VDD = 5.0V | Max 85°C Typ 25°C | 0.864 | 1.080 | µA |

1. These are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 45.12.5. Ultra Low Power Internal 32.768 kHz RC Oscillator (OSCULP32K) Characteristics

**Table 45-56.** Ultra Low Power Internal 32.768 kHz RC Oscillator Electrical Characteristics

| Symbol | Parameter | Condition | Min | Typ | Max | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$ | Output frequency | T =25°C, VDDANA = 5.0V | 30.965 | 32.768 | 34.57 | kHz |
| | | T =25°C, over [2.7, 5.5]V | 30.801 | 32.768 | 34.73 | |
| | | Over [-40, 85]°C, over [2.7, 5.5]V | 22.937 | 32.768 | 38.99 | |
| Duty | Duty Cycle | | | 50 | | % |

### 45.12.6. 48 MHz RC Oscillator (OSC48M) Characteristics

**Table 45-57.** RC 48 MHz Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output Frequency (1) | -40 to 85°c | 45.6 | 48 | 50.4 | MHz |
| | | | -5.0 | - | +5.0 | % Error |
| | Output Frequency (2) | 0 to 40°c | 47.52 | 48 | 48.48 | MHz |
| | | | -1.0 | - | +1.0 | % Error |
| | | -20 to 85°c | 47.28 | 48 | 48.72 | MHz |
| | | | -1.5 | - | +1.5 | % Error |
| | | -40 to 85°c | 47.04 | 48 | 48.96 | MHz |
| | | | -2.0 | - | +2.0 | % Error |
| | Output Frequency(5) | 0 to 40°c | 47.04 | 48 | 48.96 | MHz |
| | | | -2.0 | - | +2.0 | % Error |
| | | -20 to 85°c | 46.80 | 48 | 49.20 | MHz |
| | | | -2.5 | - | +2.5 | % Error |
| | | -40 to 85°c | 46.56 | 48 | 49.44 | MHz |
| | | | -3.0 | | +3.0 | % Error |
| $T_{START}$ (3) | Startup time | - | - | 3.9 | 15 | µs |
| Duty (4) | Duty Cycle | - | - | 50 | - | % |

**Notes:**

1. Standard Accuracy factory calibration.
2. Enhanced Accuracy factory calibration.
3. The OSC48MSTUP.STARTUP field must be set accordingly.
4. These values are based on simulation, and are not covered by test or characterization.
5. WLCSP package only.

**Table 45-58.** Power Consumption

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | Fout = 48 MHz, VDD = 5.0V | Max. 85°C<br>Typ. 25°C | 452 | 479 | µA |


© 2026 Microchip Technology Inc. and its subsidiaries

# 45.13. Timing Characteristics

## 45.13.1. SERCOM in SPI Mode Timing

**Table 45-59.** SPI Timing Characteristics and Requirements (1)

| Symbol | Parameter | Conditions | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| $t_{SCK}$(10) | SCK period | Host | Reception | $2*(t_{MIS}+t_{CLIENT_OUT})$ (3) | - | - | ns |
| | | Host | Transmission | $2*(t_{MOV}+t_{CLIENT_IN})$ (4) | - | - | |
| $t_{SCKW}$ | SCK high/low width | Host | | - | $0.5*t_{SCK}$ | - | ns |
| $t_{SCKR}$ | SCK rise time (2) | Host | | - | $0.25*t_{SCK}$ | - | ns |
| $t_{SCKF}$ | SCK fall time (2) | Host | | - | $0.25*t_{SCK}$ | - | ns |
| $t_{MIS}$ | MISO setup to SCK | Host, VDD>4.5V | | 50.7 | - | - | ns |
| | | Host, VDD>2.7V | | 60.6 | - | - | |
| $t_{MIH}$ | MISO hold after SCK | Host, VDD>4.5V | | 0 | - | - | ns |
| | | Host, VDD>2.7V | | 0 | - | - | |
| $t_{MOV}$ | MOSI output valid SCK | Host, VDD>4.5V | | - | - | 17.1 | ns |
| | | Host, VDD>2.7V | | - | - | 23.6 | |
| $t_{MOH}$ | MOSI hold after SCK | Host, VDD>4.5V | | 2.5 | - | - | ns |
| | | Host, VDD>2.7V | | 2.5 | - | - | |
| $t_{SSCK}$ | Client SCK Period | Client | Reception | $2*(t_{SIS}+t_{HOST_OUT})$ (5) | - | - | ns |
| | | Client | Transmission | $2*(t_{SOV}+t_{HOST_IN})$ (6) | - | - | |
| $t_{SSCKW}$ | SCK high/low width | Client | | - | $0.5*t_{SSCK}$ | - | ns |
| $t_{SSCKR}$ | SCK rise time (2) | Client | | - | $0.25*t_{SSCK}$ | - | ns |
| $t_{SSCKF}$ | SCK fall time (2) | Client | | - | $0.25*t_{SSCK}$ | - | ns |
| $t_{SIS}$ | MOSI setup to SCK | Client, VDD>4.5V | | 13.6 | - | - | ns |
| | | Client, VDD>2.7V | | 14.1 | - | - | |
| $t_{SIH}$ | MOSI hold after SCK | Client, VDD>4.5V | | 0 | - | - | ns |
| | | Client, VDD>2.7V | | 0 | - | - | |
| $t_{SSS}$ | $\overline{SS}$ setup to SCK | Client | PRELOADEN=1 | $t_{SOSS}+t_{EXT_MIS}+2*t_{APBC}$ (8) (9) | - | - | ns |
| | | | PRELOADEN=0 | $t_{SOSS}+t_{EXT_MIS}$ (8) | - | - | |
| $t_{SSH}$ | $\overline{SS}$ hold after SCK | Client | | $0.5*t_{SSCK}$ | - | - | ns |
| $t_{SOV}$ | MISO output valid SCK | Client, VDD>4.5V | | - | - | 45 | ns |
| | | Client, VDD>2.7V | | - | - | 55.1 | |
| $t_{SOH}$ | MISO hold after SCK | Client, VDD>4.5V | | 11.9 | - | - | ns |
| | | Client, VDD>2.7V | | 11.9 | - | - | |
| $t_{SOSS}$ | MISO setup after $\overline{SS}$ low | Client, VDD>4.5V | | - | - | 41 | ns |
| | | Client, VDD>2.7V | | - | - | 50.7 | |
| $t_{SOSH}$ | MISO hold after $\overline{SS}$ high | Client, VDD>4.5V | | 11.1 | - | - | ns |
| | | Client, VDD>2.7V | | 11.1 | - | - | |

1. These values are based on simulation. These values are not covered by test limits in production.
2. See I/O pin characteristics.
3. Where $t_{CLIENT_OUT}$ is the client external device output response time, generally $t_{EXT_SOV}+t_{LINE_DELAY}$ (7).
4. Where $t_{CLIENT_IN}$ is the client external device input constraint, generally $t_{EXT_SIS}+t_{LINE_DELAY}$ (7).
5. Where $t_{HOST_OUT}$ is the host external device output response time, generally $t_{EXT_MOV}+t_{LINE_DELAY}$ (7).


© 2026 Microchip Technology Inc. and its subsidiaries

6. Where $t_{HOST_IN}$ is the host external device input constraint, generally $t_{EXT_MIS}+t_{LINE_DELAY}$ (7).
7. $t_{LINE_DELAY}$ is the transmission line time delay.
8. $t_{EXT_MIS}$ is the input constraint for the host external device.
9. $t_{APBC}$ is the APB period for SERCOM.
10. When the integrity of communication is required to maintain both transmission and reception, the maximum SPI clock frequency should be the lowest value of the reception or transmission mode maximum frequency as shown in the following equations.
    - Reception: $t_{SCK} = 2*(t_{MIS}+t_{CLIENT_OUT})$
    - Transmission: $t_{SCK} = 2*(t_{MOV}+t_{CLIENT_IN})$

**Figure 45-7.** SPI Timing Requirements in Host Mode

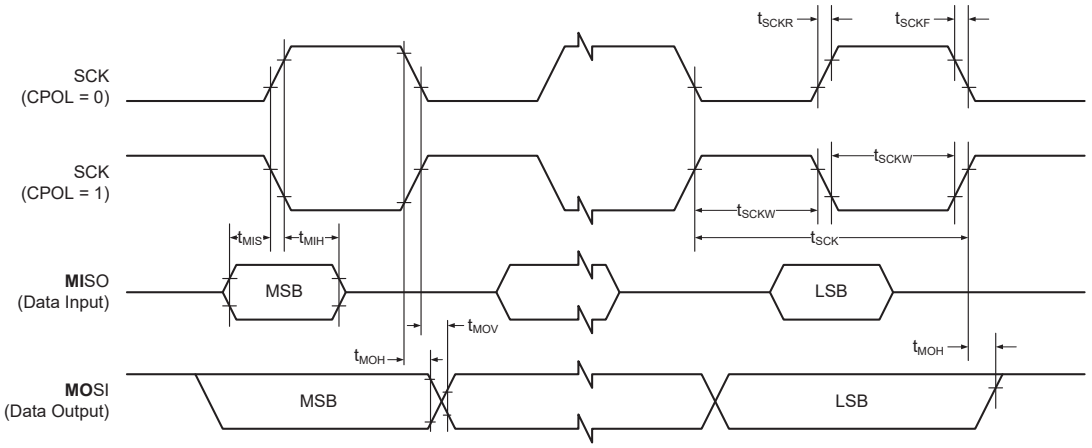
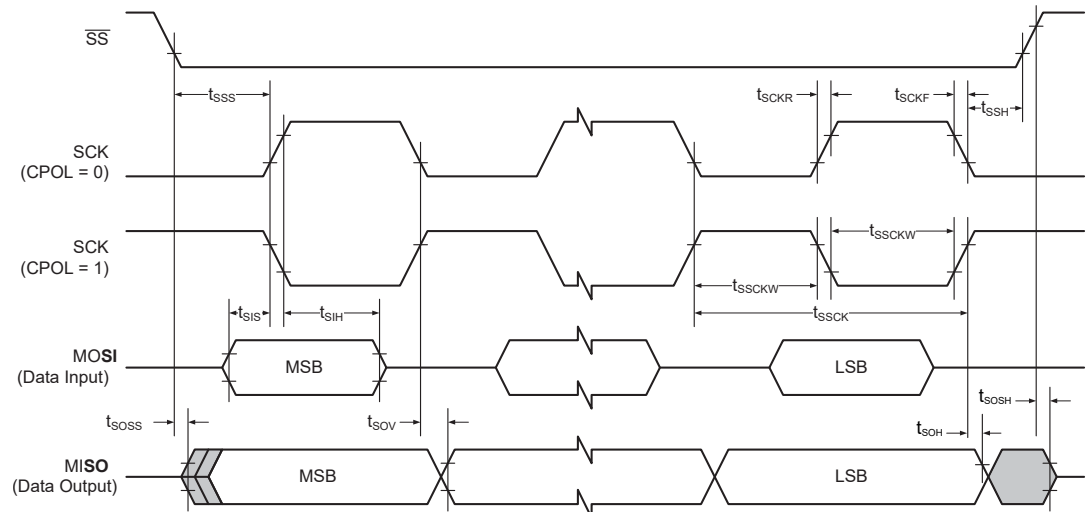



**Figure 45-8.** SPI Timing Requirements in Client Mode




## 45.13.2. External Reset

**Table 45-60.** External Reset Characteristics(1)

| Symbol | Parameter | Min. | Units |
|---|---|---|---|
| $t_{EXT}$ | Minimum reset pulse width | 1 | µs |


© 2026 Microchip Technology Inc. and its subsidiaries

1. These are based on simulation. These values are not covered by test or characterization

### 45.13.3. CAN Timing

**Table 45-61.** CAN Physical Layer Timing(1)

| Parameter | Conditions | Max. | Units |
|---|---|---|---|
| $TX_{CAN}$ output delay | VDD = 2.7V<br>Load = 20pF<br>VOL/VOH = VDD/2 | 13.9 | ns |
| | VDD = 4.5V<br>Load = 20pF<br>VOL/VOH = VDD/2 | 12.55 | |
| $RX_{CAN}$ input delay | VDD = 2.7V<br>VOL/VOH = VDD/2 | 27.4 | |
| | VDD = 4.5V<br>VOL/VOH = VDD/2 | 18.9 | |

1. These values are based on simulation. These values are not covered by test limits in production.


© 2026 Microchip Technology Inc. and its subsidiaries

# 46. Electrical Characteristics 105°C (SAM C20/C21 E/G/J)

## 46.1. Disclaimer

All typical values are measured at Ta = 25°C unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

This chapter contains only characteristics specific for the SAM C20/C21 E/G/J (Ta = 105°C). For all other values or missing characteristics, refer to the SAM C20/C21 E/G/J 85°C chapter.

## 46.2. General Operating Ratings

The device must operate within the ratings listed in the table below in order for all other electrical characteristics and typical characteristics of the device to be valid.

**Table 46-1.** General operating conditions

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $T_A$ | Temperature range | -40 | 25 | 105 | °C |
| $T_J$ | Junction temperature | - | - | 125 | °C |

## 46.3. Power Consumption

The values in the Power Consumption table below are measured values of power consumption under the following conditions, except where noted:

- Operating conditions
  - VDDIN = 3.0 V, 5.0V
- Oscillators
  - XOSC (crystal oscillator) stopped
  - XOSC32K (32.768 kHz crystal oscillator) running with external 32.768 kHz crystal
  - FDPLL using XOSC32K as reference and running at 48 MHz
- Clocks
  - FDPLL used as main clock source, except otherwise specified
  - CPU, AHB clocks undivided
  - All peripheral clocks stopped
- I/Os are inactive with input trigger disable
- CPU is running on Flash with Wait states specified in NVM Max Speed Characteristics
- NVMCTRL cache enabled
- BODVDD disabled


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 46-2.** Current Consumption[1]

| Mode | Conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 3.8 | 4.2 | mA |
| | | 105°C | 5.0V | 4.0 | 4.5 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 3.7 | 4.1 | mA |
| | | 105°C | 3.0V | 4.0 | 4.5 | |
| | CPU running a While 1 algorithm. with GCLKIN as reference | 25°C | 5.0V | 71*Freq+160 | 78*Freq+162 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 71*Freq+374 | 72*Freq+819 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 4.7 | 5.2 | mA |
| | | 105°C | 5.0V | 5.0 | 5.5 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 4.7 | 5.1 | mA |
| | | 105°C | 3.0V | 5.0 | 5.5 | |
| | CPU running a Fibonacci algorithm. with GCLKIN as reference | 25°C | 5.0V | 90*Freq+163 | 99*Freq+168 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 90*Freq+379 | 92*Freq+820 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 5.9 | 6.4 | mA |
| | | 105°C | 5.0V | 6.3 | 6.9 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 5.2 | 5.7 | mA |
| | | 105°C | 3.0V | 5.5 | 6.1 | |
| | CPU running a CoreMark algorithm. with GCLKIN as reference | 25°C | 5.0V | 115*Freq+167 | 126*Freq+167 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 118*Freq+383 | 121*Freq+823 | |
| IDLE0 | | 25°C | 5.0V | 1.8 | 1.9 | mA |
| | | 105°C | 5.0V | 2.0 | 2.5 | |
| IDLE2 | | 25°C | 5.0V | 1.2 | 1.277 | mA |
| | | 105°C | 5.0V | 1.5 | 1.956 | |
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37.0 | µA |
| | | 105°C | 5.0V | 187.0 | 512.0 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35.0 | |
| | | 105°C | 5.0V | 185.0 | 510.0 | |

**Note:**

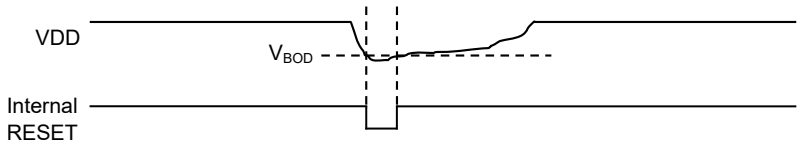
1. These are based on characterization.

## 46.4. Analog Characteristics

### 46.4.1. BODVDD - Brown Out Detector Characteristics

See NVM User Row Mapping for the BODVDD default value settings. These values are based on simulation and are not covered by test limits in production or characterization.

**Figure 46-1.** BODVDD Hysteresis OFF





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 46-2.** BODVDD Hysteresis ON

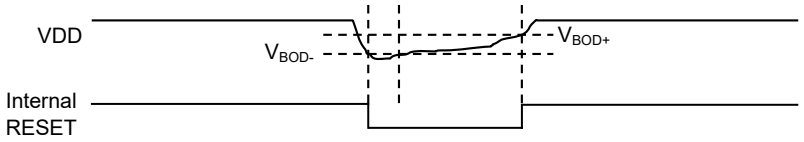



**Table 46-3.** Power Consumption (see Note 1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | IDLE, Mode CONT | VDD = 2.7V | Max 105°C | 22.5 | 26.7 | µA |
| | | VDD = 5.0V | Typ 25°C | 41.0 | 47.9 | |
| | IDLE, Mode SAMPL | VDD = 2.7V | | 0.1 | 1.5 | |
| | | VDD = 5.0V | | 0.1 | 1.9 | |
| | STANDBY, Mode SAMPL | VDD = 2.7V | | 0.8 | 2.1 | |
| | | VDD = 5.0V | | 3.5 | 4.9 | |

**Note:**

1. These values are based on characterization.

**Table 46-4.** BODVDD Characteristics (see Note 2)

| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| VBOD+ (1) | BODVDD high threshold Level | VDD level, BOD setting = 8 (default) | - | 2.86 | 2.98 | V |
| | | VDD level, BOD setting = 9 | - | 2.92 | 3.01 | |
| | | VDD level, BOD setting = 44 | - | 4.57 | 4.82 | |
| VBOD- / VBOD (1) | BODVDD low threshold Level | VDD level, BOD setting = 8 (default) | 2.71 | 2.8 | 2.90 | |
| | | VDD level, BOD setting = 9 | 2.75 | 2.85 | 2.96 | |
| | | VDD level, Bod setting = 44 | 4.37 | 4.51 | 4.66 | |
| | Step size | | - | 60 | - | mV |
| VHys (1) | Hysteresis (VBOD+ - VBOD-) BODVDD.LEVEL = 8 to 48 | VDD | 40 | - | 75 | mV |
| Tstart (3) | Startup time | Time from enable to RDY | - | 3.1 | - | µs |

**Notes:**

1. These values are based on characterization.
2. BODVDD in Continuous mode.
3. This value is based on simulation and not covered by test or characterization.

**Related Links**

NVM User Row Mapping
NVM User Row Mapping


© 2026 Microchip Technology Inc. and its subsidiaries

### 46.4.2. Analog-to-Digital Converter (ADC) Characteristics

**Table 46-5.** Power Consumption [1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Differential mode | fs = 1 Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | Max 105°C Typ 25°C | 905 | 1084 | uA |
| | | fs = 1 Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 1144 | 1425 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 381 | 518 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 609 | 877 | |
| | Single Ended mode | fs = 1 Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref=5.5V | Max 105°C Typ 25°C | 984 | 1154 | uA |
| | | fs = 1 Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref=5.5V | | 1178 | 1467 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 437 | 588 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 635 | 908 | |

1. These are based on characterization.

### 46.4.3. Sigma-Delta-Analog-to-Digital Converter (SDADC) Characteristics

**Table 46-6.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Power consumption | CTLSDADC=0x0 External Ref - VDDANA = 5.5V Vref = 2V Ref buf on SCLK_SDADC = 6 MHz | Max 105°C Typ 25°C | 644 | 710 | uA |
| | | CTLSDADC=0x0 Internal Ref - VDDANA=Vref= 5.5V Ref buf off SCLK_SDADC = 6 MHz | | 605 | 649 | |

1. These are based on characterization.

### 46.4.4. Digital-to-Analog Converter (DAC) Characteristics

**Table 46-7.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | DC supply current | Output buffer On, VREF = VDDANA=5.0V | Max 105°C Typ 25°C | 318 | 414 | µA |
| | | Output buffer Off, VREF = VDDANA=5.0V | | 74 | 82 | |

1. These values are based on characterization.

### 46.4.5. Analog Comparator (AC) Characteristics

**Table 46-8.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDDANA | Current consumption - Vcm=Vddana/2, +-100 mV overdrive from Vcm, Voltage scaler disabled | COMPCTRLn.SPEED = 0x0, VDDANA =5.0V | Max 105°C Typ 25°C | 10 | 18 | µA |
| | | COMPCTRLn.SPEED = 0x3, VDDANA =5.0V | | 39 | 60 | |
| | Current consumption Voltage scaler only | VDDANA =5.0V | | 43 | 63 | |

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 46.4.6. Temperature Sensor Characteristics

**Table 46-9.** Temperature Sensor Characteristics(1)

| Parameter | Condition | Min. | Max. | Unit |
|---|---|---|---|---|
| Accuracy | [-40,105]°C | -14.6 | 10.5 | °C |

1. These are based on characterization. Data has been obtained by averaging 10 TSENS acquisitions per measurement.

### 46.4.7. PTC Characteristics

#### 46.4.7.1. PTC Power Consumption

The values given in the table below are measured values of power consumption under the following conditions:

- **Operating conditions**
  - VDD = 5.0V
- **Clocks**
  - OSC48M used as main clock source, running undivided at 48 MHz
  - CPU is running on Flash with 2 wait states at 48 MHz
  - PTC running at 4 MHz
- **PTC configuration**
  - Mutual-Capacitance mode
  - One-touch channel
- **System configuration**
  - Standby Sleep mode enabled
  - RTC running on ULP32K: used to define the PTC scan rate through the event system
  - Drift Calibration disabled: no interrupts, PTC scans are performed in Standby mode
  - Drift Calibration enabled: RTC interrupts (wake up) the CPU to perform PTC scans. PTC drift calibration is performed every 1.5 second.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 46-10.** Power Consumption [1]

| Symbol | Parameters | Drift Calibration | PTC scan rate | Oversamples | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|---|
| IDD | Current Consumption | Disabled | 10 | 4 | Max. 105°C Typ. 25°C | 24 | 519 | µA |
| | | | | 16 | | 43 | 543 | |
| | | | 50 | 4 | | 17 | 512 | |
| | | | | 16 | | 22 | 518 | |
| | | | 100 | 4 | | 16 | 511 | |
| | | | | 16 | | 19 | 515 | |
| | | | 200 | 4 | | 16 | 511 | |
| | | | | 16 | | 17 | 513 | |
| | | Enabled | 10 | 4 | | 30 | 530 | |
| | | | | 16 | | 50 | 554 | |
| | | | 50 | 4 | | 20 | 517 | |
| | | | | 16 | | 24 | 522 | |
| | | | 100 | 4 | | 18 | 516 | |
| | | | | 16 | | 21 | 518 | |
| | | | 200 | 4 | | 17 | 515 | |
| | | | | 16 | | 19 | 517 | |

**Note:**

1. These values are based on characterization.

## 46.5. NVM Characteristics

**Table 46-11.** Flash Endurance

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Cyc_{NVM}$ | Cycling Endurance[1] | -40°C < $T_A$ < 105°C | 5k | - | Cycles |

1. An endurance cycle is a write and an erase operation.

**Table 46-12.** EEPROM Emulation[1] Endurance

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Cyc_{EEPROM}$ | Cycling Endurance[2] | -40°C < $T_A$ < 105°C | 20k | - | Cycles |

1. The EEPROM emulation is a software emulation as described in the *"AT03265 Application Note"*.
2. An endurance cycle is a write and an erase operation.


© 2026 Microchip Technology Inc. and its subsidiaries

# 46.6. Oscillator Characteristics

## 46.6.1. Crystal Oscillator (XOSC) Characteristics

**Table 46-13.** Power Consumption(1)

| Symbol | Parameters | Conditions | | $T_A$ | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| IDD | Current consumption | F = 2 MHz<br>CL = 20 pF<br>XOSC.GAIN = 0<br>VDD = 5.0V | AGC = OFF | Max. 105°C<br>Typ. 25°C | 150 | 206 | µA |
| | | | AGC = ON | | 138 | 198 | |
| | | F = 4 MHz<br>CL= 20 pF<br>XOSC.GAIN = 1<br>VDD = 5.0V | AGC = OFF | | 220 | 293 | |
| | | | AGC = ON | | 175 | 267 | |
| | | F = 8 MHz<br>CL = 20 pF<br>XOSC.GAIN = 2<br>VDD = 5.0V | AGC = OFF | | 350 | 425 | |
| | | | AGC = ON | | 247 | 331 | |
| | | F = 16 MHz<br>CL = 20 pF<br>XOSC.GAIN = 3<br>VDD = 5.0V | AGC = OFF | | 663 | 861 | |
| | | | AGC = ON | | 429 | 725 | |
| | | F = 32 MHz<br>CL = 18 pF<br>XOSC.GAIN = 4<br>VDD = 5.0V | AGC = OFF | | 1975 | 2397 | |
| | | | AGC = ON | | 874 | 1252 | |

1. These values are based on characterization.

## 46.6.2. External 32.768 kHz Crystal Oscillator (XOSC32K) Characteristics

**Table 46-14.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD | Current consumption | $V_{DD}$ = 5.0V | Max. 105°C<br>Typ. 25°C | 1528 | 1740 | nA |

1. These values are based on characterization.

## 46.6.3. Digital Phase Locked Loop (DPLL) Characteristics

**Table 46-15.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current Consumption | Ck = 48 MHz<br>$V_{DD}$ = 5.0V | Max. 105°C<br>Typ. 25°C | 536 | 629 | µA |
| | | Ck = 96 MHz<br>$V_{DD}$ = 5.0V | | 865 | 986 | |

1. These are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 46.6.4. 32.768 kHz Internal Oscillator (OSC32K) Characteristics

**Table 46-16.** 32.768 kHz RC Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $F_{OUT}$ | Output frequency | T = 25°C<br>VDDANA = 5.0V | 32.112 | 32.768 | 33.423 | kHz |
| | | T = 25°C<br>Over [2.7, 5.5]V | 29.491 | 32.768 | 36.044 | kHz |
| | | Over [-40,105]°C<br>Over [2.7, 5.5]V | 25.559 | 32.768 | 37.683 | kHz |
| Tstartup | Startup time | | - | 1 | 2 | cycles |
| Duty (1) | Duty cycle | | - | 50 | - | % |

1. These values are based on simulation and are not covered by test or characterization.

**Table 46-17.** Power Consumption

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | VDD = 5.0V | Max. 105°C<br>Typ. 25°C | 0.864 | 1.116 | µA |

1. These values are based on characterization.

### 46.6.5. Ultra-Low Power Internal 32.768 kHz RC Oscillator (OSCULP32K) Characteristics

**Table 46-18.** Ultra-Low Power Internal 32.768 kHz RC Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output frequency | T = 25°C<br>VDDANA = 5.0V | 30.965 | 32.768 | 34.57 | kHz |
| | | T = 25°C<br>Over [2.7, 5.5]V | 30.801 | 32.768 | 34.734 | kHz |
| | | Over [-40, 105]°C<br>Over [2.7, 5.5]V | 22.937 | 32.768 | 40.632 | kHz |
| Duty | Duty Cycle | | - | 50 | - | % |

### 46.6.6. 48 MHz RC Oscillator (OSC48M) Characteristics

**Table 46-19.** RC 48 MHz Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output Frequency (1) | -40 to 105°c | 45.6 | 48 | 50.4 | MHz |
| | | | -5.0 | - | +5.0 | % Error |
| | Output Frequency (2) | 0 to 40°c | 47.52 | 48 | 48.48 | MHz |
| | | | -1.0 | - | +1.0 | % Error |
| | | -20 to 85°c | 47.28 | 48 | 48.72 | MHz |
| | | | -1.5 | - | +1.5 | % Error |
| | | -40 to 85°c | 47.04 | 48 | 48.96 | MHz |
| | | | -2.0 | - | +2.0 | % Error |
| | | -40 to 105°c | 46.8 | 48 | 49.2 | MHz |
| | | | -2.5 | - | +2.5 | % Error |
| $T_{START}$ (3) | Startup time | | - | 3.9 | 15 | µs |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 46-19.** RC 48 MHz Oscillator Electrical Characteristics (continued)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Duty (4) | Duty Cycle | | - | 50 | - | % |

**Notes:**

1. Standard Accuracy factory calibration.
2. Enhanced Accuracy factory calibration.
3. The OSC48MSTUP.STARTUP field must be set accordingly.
4. These values are based on simulation, and are not covered by test or characterization.

**Table 46-20.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | $F_{OUT}$ = 48 MHz<br>$V_{DD}$ = 5.0V | Max. 105°C<br>Typ. 25°C | 452 | 485 | µA |

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

# 47. Electrical Characteristics 105°C (SAM C20/C21 N)

## 47.1. Disclaimer

All typical values are measured at Ta = 25°C unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

This chapter contains only characteristics specific for the SAM C20/C21 N devices (Ta = 105°C). For all other values or missing characteristics, refer to the SAM C20/C21 E/G/J 85°C and 105°C chapters.

**Related Links**

Electrical Characteristics 85°C (SAM C20/C21 E/G/J)

## 47.2. General Operating Ratings

The device must operate within the ratings listed in the table below in order for all other electrical characteristics and typical characteristics of the device to be valid.

**Table 47-1.** General operating conditions

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $T_A$ | Temperature range | -40 | 25 | 105 | °C |
| $T_J$ | Junction temperature | - | - | 125 | °C |

## 47.3. Power Consumption

**Table 47-2.** Current Consumption(1)

| Mode | Conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 3.8 | 4.2 | mA |
| | | 105°C | 5.0V | 4.0 | 5.0 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 3.7 | 4.1 | mA |
| | | 105°C | 3.0V | 4.0 | 5.0 | |
| | CPU running a While 1 algorithm. with GCLKIN as reference | 25°C | 5.0V | 71*Freq+160 | 78*Freq+162 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 71*Freq+374 | 68*Freq+1564 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 4.7 | 5.2 | mA |
| | | 105°C | 5.0V | 5.0 | 6.1 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 4.7 | 5.1 | mA |
| | | 105°C | 3.0V | 5.0 | 6.0 | |
| | CPU running a Fibonacci algorithm. with GCLKIN as reference | 25°C | 5.0V | 90*Freq+163 | 99*Freq+168 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 90*Freq+379 | 90*Freq+1568 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 5.9 | 6.4 | mA |
| | | 105°C | 5.0V | 6.3 | 7.1 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 5.2 | 5.7 | mA |
| | | 105°C | 3.0V | 5.5 | 6.6 | |
| | CPU running a CoreMark algorithm. with GCLKIN as reference | 25°C | 5.0V | 115*Freq+167 | 126*Freq+167 | µA (with freq in MHz) |
| | | 105°C | 5.0V | 118*Freq+383 | 110*Freq+1583 | |
| IDLE0 | | 25°C | 5.0V | 1.8 | 2.4 | mA |
| | | 105°C | 5.0V | 2.0 | 3.2 | |
| IDLE2 | | 25°C | 5.0V | 1.2 | 1.7 | mA |
| | | 105°C | 5.0V | 1.5 | 2.6 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-2.** Current Consumption[1] (continued)

| Mode | Conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37.0 | µA |
| | | 105°C | 5.0V | 187.0 | 602.0 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35.0 | |
| | | 105°C | 5.0V | 185.0 | 600.0 | |

1. These are based on characterization.

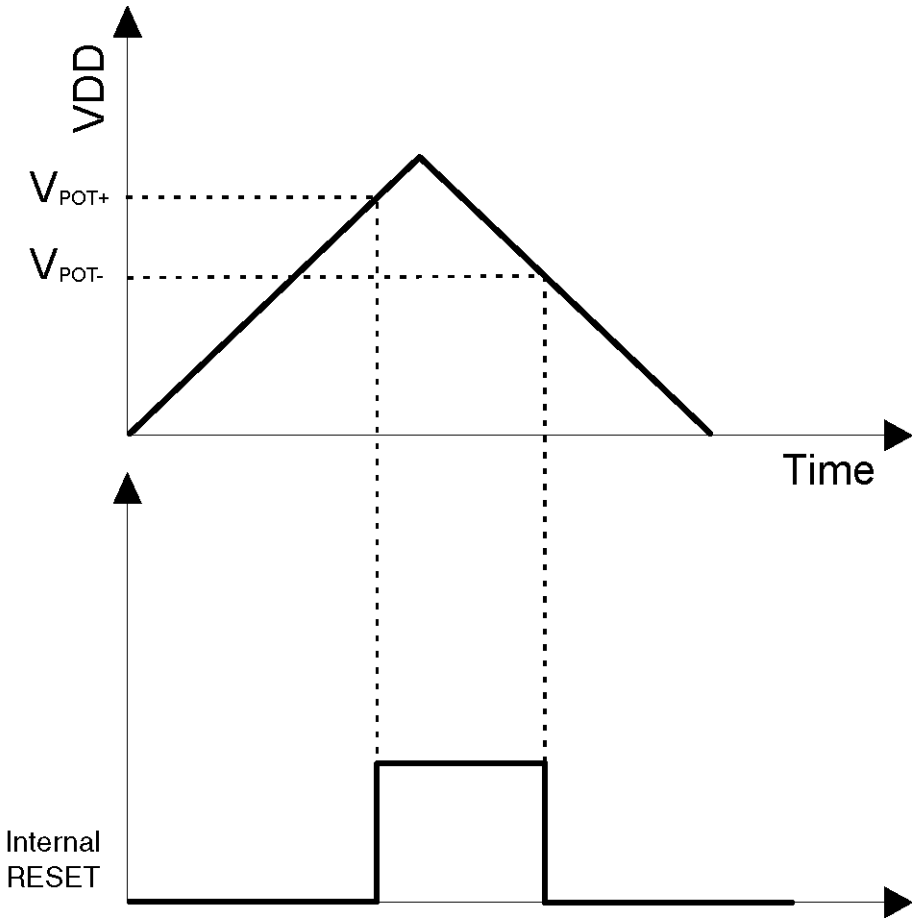
## 47.4. Analog Characteristics

### 47.4.1. Power-on Reset (POR) Characteristics

**Table 47-3.** POR Characteristics

| Symbol | Parameters | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| $V_{POT+}$ | Voltage threshold Level on VDDIN rising | - | 2.55 | - | V |
| $V_{POT-}$ | Voltage threshold Level on VDDIN falling | 1.77 | 1.92 | 2.04 | |

**Figure 47-1.** POR Operating Principle




### 47.4.2. BODVDD - Brown-out Detector Characteristics

See NVM User Row Mapping for the BODVDD default value settings. These values are based on simulation and are not covered by test limits in production or characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 47-2.** BODVDD Hysteresis OFF

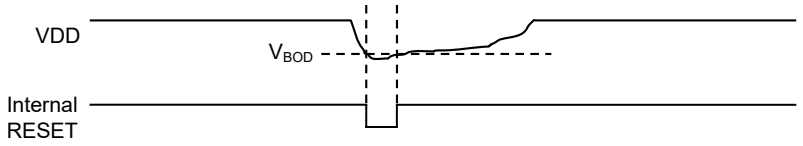



**Figure 47-3.** BODVDD Hysteresis ON

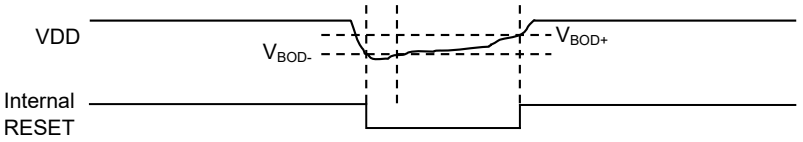



**Table 47-4.** BODVDD Characteristics[2]

| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| VBOD+[1] | BODVDD high threshold Level | VDD level, Bod setting = 8 (default) | - | 2.86 | 2.98 | V |
| | | VDD level, Bod setting = 9 | - | 2.92 | 3.01 | |
| | | VDD level, Bod setting = 44 | - | 4.57 | 4.82 | |
| VBOD- / VBOD[1] | BODVDD low threshold Level | VDD level, Bod setting = 8 (default) | 2.71 | 2.80 | 2.90 | |
| | | VDD level, Bod setting = 9 | 2.75 | 2.85 | 2.96 | |
| | | VDD level, Bod setting = 44 | 4.37 | 4.51 | 4.66 | |
| | Step size | | - | 60 | - | mV |
| VHys[1] | Hysteresis (VBOD+ - VBOD-) BODVDD.LEVEL = 8 to 48 | VDD | 40 | - | 75 | mV |
| $T_{START}$[3] | Startup time | Time from enable to RDY | - | 3.1 | - | µs |

1. These values are based on characterization.
2. BODVDD in continuous mode.
3. This value is based on simulation and not covered by test or characterization.

### 47.4.3. Analog-to-Digital Converter (ADC) Characteristics

**Table 47-5.** Operating Conditions[1]

| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Res | Resolution | | - | - | 12 | bits |
| Rs | Sampling rate[2] | SAMPLEN = 3 resolution 12 bit (CTRLC.RESSEL = 0) | 10 | - | 1000 | ksps |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-5.** Operating Conditions[1] (continued)

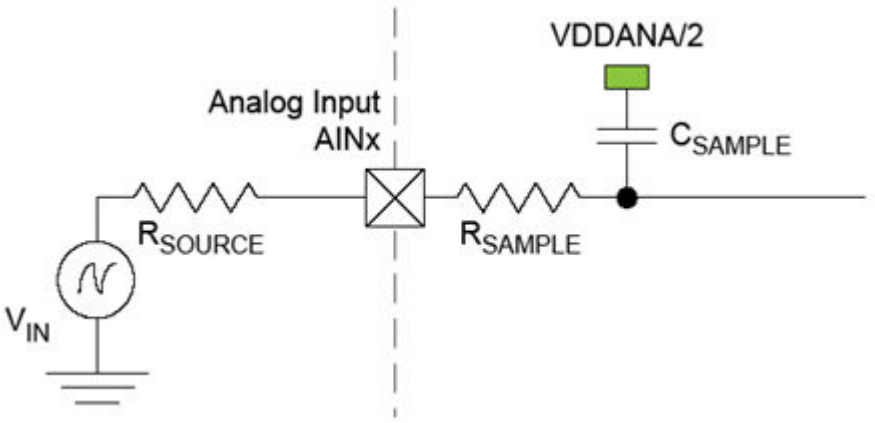
| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| Nb_cycles | Differential mode Number of ADC clock cycles SAMPCTRL.OFFCOMP = 1 | resolution 12 bit (CTRLC.RESSEL = 0) | - | 16 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL = 2) | | 14 | | |
| | | resolution 8 bit (CTRLC.RESSEL = 3) | | 12 | | |
| | Differential mode Number of ADC clock cycles SAMPCTRL.OFFCOMP = 0 SAMPLEN corresponds to the decimal value of SAMPCTRL.SAMPLEN[5:0] register | resolution 12 bit (CTRLC.RESSEL = 0) | - | SAMPLEN+13 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL = 2) | | SAMPLEN+11 | | |
| | | resolution 8 bit (CTRLC.RESSEL = 3) | | SAMPLEN+9 | | |
| | Single-ended mode Number of ADC clock cycles SAMPCTRL.OFFCOMP = 1 | resolution 12 bit (CTRLC.RESSEL= 0) | - | 16 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | 15 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | 13 | | |
| | Single-ended mode Number of ADC clock cycles SAMPCTRL.OFFCOMP = 0 SAMPLEN corresponds to the decimal value of SAMPCTRL.SAMPLEN[5:0] register | resolution 12 bit (CTRLC.RESSEL= 0) | - | SAMPLEN+13 | - | cycles |
| | | resolution 10 bit (CTRLC.RESSEL= 2) | | SAMPLEN+12 | | |
| | | resolution 8 bit (CTRLC.RESSEL= 3) | | SAMPLEN+10 | | |
| fadc | ADC Clock frequency | | 160 | - | 16000 | kHz |
| Ts | Sampling time | SAMPCTRL.OFFCOMP = 1 | 250 | - | 25000 | ns |
| | | SAMPCTRL.OFFCOMP = 0 | (SAMPLEN+1)/ fadc | - | - | s |
| | Sampling time with DAC as input | - | 3 | - | - | µs |
| | Sampling time with Bandgap as input | - | 10 | - | - | µs |
| Vcnv | Conversion range | Differential mode | -VREF | - | +VREF | V |
| | | Single-ended mode | 0 | - | VREF | |
| Vref | Reference input | - | 2 | - | VDDANA-0.6 | V |
| Vin | Input channel range | - | 0 | - | VDDANA | V |
| Vcmin | Input common mode voltage | CTRLC.R2R = 1 | 0.2 | - | VREF-0.2 | V |
| | | CTRLC.R2R = 0 | VREF/2-0.2 | - | VREF/2+0.2 | V |
| CSAMPLE | Input sampling capacitance | | - | - | 3.2 | pF |
| RSAMPLE | Input sampling on-resistance | For a sampling rate at 1 Msps | - | 1000 | 1715 | Ω |
| Rref | Reference input source resistance | | 0 | - | 1000 | kΩ |

**Notes:**

1. These values are based on simulation. These values are not covered by test limits in production or characterization.
2. Sampling rate (in samples per second) is equal to Nb_cycles/fadc.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 47-4.** ADC Analog Input AINx




The minimum sampling time $t_{samplehold}$ for a given $R_{source}$ can be found using this formula:

$$t_{samplehold} \geq \left(R_{sample} + R_{source}\right) \times C_{sample} \times \left(n + 2\right) \times \ln\left(2\right)$$

For 12-bit accuracy:

$$t_{samplehold} \geq \left(R_{sample} + R_{source}\right) \times C_{sample} \times 9.7$$

**Table 47-6.** Differential Mode

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| ENOB[1] | Effective Number of bits | Fadc = 500 ksps - R2R disabled | Vddana = 5.0V Vref = Vddana | 9.9 | 10.7 | 11.4 | bits |
| | | | Vddana = 2.7V Vref = 2.0V | 10.0 | 10.8 | 11.3 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | 9.7 | 10.6 | 11.3 | |
| | | | Vddana = 2.7V Vref = 2.0V | 9.8 | 10.6 | 11.2 | |
| | | Fadc = 500 ksps - R2R Enabled[2] | Vddana = 5.0V Vref = Vddana | 9.8 | 11.3 | 11.9 | |
| | | Fadc = 1 Msps - R2R Enabled[2] | Vddana = 5.0V Vref = Vddana | 9.7 | 11.1 | 11.8 | |
| TUE | Total Unadjusted Error | Fadc = 500 ksps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | +/-3.4 | +/-5 | LSB |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-3 | +/-5.6 | |
| | | Fadc = 1 Msps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | +/-4.2 | +/-6.3 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-3.6 | +/-7.7 | |
| INL | Integral Non Linearity | Fadc = 500 ksps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | +/-1.9 | +/-3.5 | LSB |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-1.6 | +/-3.5 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | +/-2 | +/-3.3 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-1.9 | +/-3.6 | |
| DNL | Differential Non Linearity | Fadc = 500 ksps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | -0.9/+1 | -1/+1.2 | LSB |
| | | | Vddana = 2.7V Vref = 2.0V | - | -0.9/+1.1 | -1/+2.1 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | -0.9/+1 | -1/+1 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | -1/+1.6 | -1/+3.6 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-6.** Differential Mode (continued)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| GE | Gain Error | Fadc = 1 Msps - R2R disabled w/o gain compensation | Vddana = 5.0V Vref = Vddana | - | +/-0.06 | +/-0.3 | % |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-0.06 | +/-1.2 | |
| | | | Vddana = 5.0V 1V internal Ref | - | +/-1.9 | +/-6.5 | |
| | | | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.11 | +/-0.82 | |
| | | Fadc = 1 Msps - R2R disabled with gain compensation | Vddana = 2.7V Vref = 2.0V | - | +/-0.03 | +/-0.46 | |
| | | | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.13 | +/-0.58 | |
| OE | Offset Error | Fadc = 1 Msps - R2R disabled without offset compensation | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.8 | +/-13 | mV |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-0.7 | +/-9.7 | |
| | | Fadc = 1 Msps - R2R disabled with offset compensation | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.01 | +/-5.6 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-0.4 | +/-4.2 | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana = 5.0V Vref = Vddana | 63 | 71 | 81 | dB |
| SINAD(1) | | Signal to Noise and Distortion ratio | | 60 | 65 | 70 | |
| SNR at -3 db FS | | Signal to Noise ratio | | 64 | 67 | 70 | |
| THD | | Total Harmonic Distortion | | 63 | -70 | 81 | |
| | | Noise RMS | External Reference voltage | - | 0.4 | 3.2 | mV |

1. Referred to Full Scale.
2. Dynamical input range is +/-6% of Full scale.

**Table 47-7.** Single-Ended Mode

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| ENOB(1) | Effective Number of bits | Fadc = 500 ksps - R2R disabled | Vddana = 3.0V Vref = Vddana | 9.0 | 9.7 | 10.2 | bits |
| | | | Vddana = 3.0V Vref = 2.0V | 9.0 | 9.6 | 10.1 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 3.0V Vref = Vddana | 8.9 | 9.6 | 10.0 | |
| | | | Vddana = 3.0V Vref = 2.0V | 8.9 | 9.4 | 9.7 | |
| TUE | Total Unadjusted Error | Fadc = 500 ksps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | +/-12.9 | +/-25.2 | LSB |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-25 | +/-49.6 | |
| | | Fadc = 1 Msps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana = 5.0V Vref = Vddana | - | +/-13.5 | +/-26.4 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-27 | +/-52 | |
| INL | Integral Non Linearity | Fadc = 500 ksps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | +/-3.7 | +/-6.5 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-3.4 | +/-5.9 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | +/-4.2 | +/-7.4 | LSB |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-3.5 | +/-6.2 | |
| DNL | Differential Non Linearity | Fadc = 500 ksps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | -0.9/+1.2 | -1/+1.6 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | -0.9/+1.3 | -1/+2.3 | |
| | | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | -1/+1.1 | -1/+1.3 | |
| | | | Vddana = 2.7V Vref = 2.0V | - | -1/+1.4 | -1/+3.1 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-7.** Single-Ended Mode (continued)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| GE | Gain Error | Fadc = 1 Msps - R2R disabled w/o gain compensation | Vddana = 5.0V Vref = Vddana | - | +/-0.2 | +/-0.7 | % |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-0.3 | +/-1.4 | |
| | | | Vddana = 5.0V 1V internal Ref | - | +/-1.6 | +/-6.6 | |
| | | | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.2 | +/-1.1 | |
| | | Fadc = 1 Msps - R2R disabled with gain compensation | Vddana = 2.7V Vref = 2.0V | - | +/-0.3 | +/-0.8 | |
| | | | Vddana = 5.0V Vref = Vddana/2 | - | +/-0.1 | +/-0.5 | |
| OE | Offset Error | Fadc = 1 Msps - R2R disabled | Vddana = 5.0V Vref = Vddana | - | +/-7 | +/-63 | mV |
| | | | Vddana = 2.7V Vref = 2.0V | - | +/-7 | +/-64 | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana = 5.0V Vref = Vddana | 57 | 66 | 73 | dB |
| SINAD(1) | | Signal to Noise and Distortion ratio | | 54 | 59 | 62 | |
| SNR at -3 db FS | | Signal to Noise ratio | | 57 | 60 | 62 | |
| THD | | Total Harmonic Distortion | | -71 | -64 | -56 | |
| | | Noise RMS | External Reference voltage | - | 0.6 | 1.9 | mV |

1. Referred to Full Scale.

### 47.4.4. Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics

**Table 47-8.** Operating Conditions[1]

| Symbol | Parameters | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| Res | Resolution | Differential mode | | - | 16 | - | bits |
| | | Single-Ended mode | | - | 15 | - | |
| CLK_SDADC | Sampling Clock Speed | | | 1 | - | 6 | MHz |
| CLK_SDADC_FS | Conversion rate | | | CLK_SDADC/4 | | | |
| fs | Output Data Rate | Free running mode | | CLK_SDADC_FS / OSR | | | |
| | | Single conversion mode SKPCNT = N | | (CLK_SDADC_FS / OSR) x (N+1) | | | |
| OSR | Oversampling ratio | Differential mode | | 64 | 256 | 1024 | Cycles |
| Vin | Input Conversion range | VREF<VDDANA-0.3V | Differential mode Gaincorr = 0x1 | - VREF | - | VREF | V |
| | | | Single-Ended mode[3] Gaincorr = 0x1 | 0 | - | VREF | |
| | | VREF>=VDDANA-0.3V | Differential mode Gaincorr = 0x1 | -0.7xVREF | - | 0.7xVREF | V |
| | | | Single-Ended mode[3] Gaincorr = 0x1 | 0 | - | 0.7xVREF | |
| Vref | Reference Voltage range | | | 1 | - | VDDANA | V |
| Vcom | Common mode voltage | Differential mode | | 0 | - | VDDANA | V |
| Cin | Input capacitance | | | 0.425 | 0.5 | 0.575 | pF |
| Zin | Input impedance | Differential mode | | 1/(Cin x CLK_SDADC_FS) | | | kΩ |
| | | Single-Ended mode[3] | | 1/(Cin x CLK_SDADC_FS x 2) | | | |
| | Input anti-alias filter recommendation[2] | Rext | | - | 1.0 | - | kΩ |
| | | Cext | | 3.3 | - | 10 | nF |

1. These are based on simulation. These values are not covered by test or characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

2. External Anti-alias filter must be placed in front of each SDADC input to ensure high-frequency signals to not alias into measurement bandwidth. Use capacitors of X5R type for DC measurement. or capacitors of COG or NPO type for AC measurement.
3. This mode corresponds to a differential mode where the selected AINNx pin is externally grounded.

**Table 47-9.** SDADC DC Performance: Differential Input Mode. Chopper ON(1)

| Symbol | Parameters | Conditions (2) | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| INL | Integral Non Linearity | CLK_SDADC = 3MHz VREF = 1.2V | - | +/-2.9 | +/-3.9 | LSB |
| | | CLK_SDADC = 3MHz INT VREF = 5.5V | - | +/-8.4 | +/-9.3 | |
| DNL | Differential Non Linearity | CLK_SDADC = 3MHz VREF = 1.2V | - | +/-1.5 | +/-2.1 | LSB |
| | | CLK_SDADC = 3MHz INT VREF = 5.5V | - | +/-1.7 | +/-2.3 | |
| Eg | Gain Errors | CLK_SDADC = 3MHz VREF = 1.2V | - | +/-0.3 | +/-1.9 | % |
| | | CLK_SDADC = 3MHz INT VREF = 5.5V | - | +/-0.3 | +/-1.7 | |
| TCg | Gain Drift | CLK_SDADC = 3MHz VREF = 1.2V | -0.9 | 3.9 | 17.5 | ppm/°C |
| Off | Offset Error | CLK_SDADC = 3MHz VREF = 1.2V | - | +/-2.3 | +/-3.7 | mV |
| | | CLK_SDADC = 3MHz INT VREF = 5.5V | - | +/-0.3 | +/-2.4 | |
| Tco | Offset Error Drift | CLK_SDADC = 3MHz VREF = 1.2V | -1.4 | 0.01 | 0.6 | uV/°C |

1. OSR=256

**Table 47-10.** SDADC DC Performance: Differential Input Mode. Chopper OFF(1)

| Symbol | Parameters | Conditions (2) | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| INL | Integral Non Linearity | CLK_SDADC = 6MHz VREF = 1.2V | - | +/-5.5 | +/-9.3 | LSB |
| | | CLK_SDADC = 6MHz INT VREF = 5.5V | - | +/-8.9 | +/-10.1 | |
| DNL | Differential Non Linearity | CLK_SDADC = 6MHz VREF = 1.2V | - | +/-2.8 | +/-4.1 | LSB |
| | | CLK_SDADC = 6MHz INT VREF = 5.5V | - | +/-1.8 | +/-3 | |
| Eg | Gain Errors | CLK_SDADC = 6MHz VREF = 1.2V | - | +/-0.6 | +/-2.1 | % |
| | | CLK_SDADC = 6MHz INT VREF = 5.5V | - | +/-0.3 | +/-1.7 | |
| TCg | Gain Drift | CLK_SDADC = 6MHz VREF = 1.2V | -19.7 | 2.2 | 20.9 | ppm/°C |
| Off | Offset Error | CLK_SDADC = 6MHz VREF = 1.2V | - | +/-1.7 | +/-14.3 | mV |
| | | CLK_SDADC = 6MHz INT VREF = 5.5V | - | +/-4.9 | +/-13.2 | |
| Tco | Offset Error Drift | CLK_SDADC = 6MHz VREF = 1.2V | -14 | 12.4 | 60 | µV/°C |
| Input noise rms | AC Input noise rms | OSR = 256 VREF = 1.2V | - | 19 | 20 | |
| | | OSR = 256 VREF = 5.5V | - | 59 | 76 | mVrms |

1. OSR=256

**Table 47-11.** SDADC AC Performance: : Differential Input Mode(1)

| Symbol | Parameters | Conditions (2) | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| ENOB | Effective Number Of Bits | Ext ref = 1.2V | 12 | 15.3 | 15.4 | bits |
| | | Int Ref = 5.5V | 12.9 | 13.1 | 14 | |
| DR | Dynamic Range | Ext ref = 1.2V | 90.5 | 92.4 | 93.2 | dB |
| | | Int Ref = 5.5V | 83.0 | 95.6 | 97.0 | |
| SNR | Signal to Noise Ratio | Ext ref = 1.2V | 68.7 | 88.7 | 89 | dB |
| | | Int Ref = 5.5V | 83 | 95.6 | 97 | |
| SINAD | Signal to Noise + Distortion Ratio | Ext ref = 1.2V | 71.1 | 90.7 | 91.7 | dB |
| | | Int Ref = 5.5V | 77.1 | 78.6 | 83.2 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-11.** SDADC AC Performance: : Differential Input Mode[1] (continued)

| Symbol | Parameters | Conditions (2) | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| THD | Total Harmonic Distortion | Ext ref = 1.2V | -102.3 | -94.6 | -75.3 | dB |
| | | Int Ref = 5.5V | -99.9 | -94.7 | -85.4 | |

1. Values based on characterization.
2. OSR=256, Chopper OFF, Sampling Clock Speed at 6MHz.

### 47.4.5. Digital to Analog Converter (DAC) Characteristics

**Table 47-12.** Operating Conditions[1]

| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| RES | Input resolution | | - | - | 10 | Bits |
| VDDANA | Analog supply voltage | | 2.7 | - | 5.5 | V |
| AVREF | External reference voltage | | 1 | - | VDDANA - 0.6 | V |
| | INTREF | VREF.SEL = 0x0 | - | 1.024 | - | V |
| | | VREF.SEL = 0x2 | - | 2.048 | - | |
| | | VREF.SEL = 0x3 | - | 4.096 [2] | - | |
| | VDDANA | | - | VDDANA | - | V |
| | Linear output voltage range | | 0.05 | - | VDDANA - 0.05 | V |
| | Minimum resistive load | | 5 | - | - | kΩ |
| | Maximum capacitance load | | - | - | 100 | pF |

1. These are based on simulation. These values are not covered by test or characterization.
2. For VDDANA > 4.5V.

**Table 47-13.** Clock and Timing[1]

| Symbol | Parameter | Conditions | | Max. | Units |
|---|---|---|---|---|---|
| | Conversion rate | Cload=100pF<br>Rload > 5kΩ | Normal mode | 350 | ksps |
| | | | For DDATA=±1 | 1000 | |
| | Startup time | | | 3 | µs |

1. These values are based on simulation. These values are not covered by test limits in production or characterization.

**Table 47-14.** Accuracy Characteristics[1]

| Symbol | Parameter | Conditions | | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INL | Integral non-linearity | VREF= Ext 2.0V | VDD = 2.7V | +/-0.7 | +/-2.4 | LSB |
| | | | VDD = 5.5V | +/-0.5 | +/-1.6 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.6 | +/-2.0 | |
| | | | VDD = 5.5V | +/-0.4 | +/-1.6 | |
| | | VREF= 1.024V INTREF | VDD = 2.7V | +/-1.0 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.5 | +/-3.5 | |
| DNL | Differential non-linearity | VREF= Ext 2.0V | VDD = 2.7V | +/-0.3 | +/-2.3 | LSB |
| | | | VDD = 5.5V | +/-0.4 | +/-2.2 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.2 | +/-2.1 | |
| | | | VDD = 5.5V | +/-0.2 | +/-2.1 | |
| | | VREF= 1.024V INTREF | VDD = 2.7V | +/-1.0 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.4 | +/-3.5 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-14.** Accuracy Characteristics[1] (continued)

| Symbol | Parameter | Conditions | | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| GE | Gain error | Ext. VREF | | +/-8 | +/-28 | mV |
| OE | Offset error | Ext. VREF | | +/-4 | +/-26 | mV |

1. These values are based on characterization. These values are not covered by test limits in production.

## 47.4.6. Analog Comparator Characteristics

**Table 47-15.** Analog Comparator Characteristics

| Symbol | Parameters | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| PNIVR | Positive and Negative input range voltage | | 0 | - | VDDANA | V |
| ICMR | Input common mode range | | 0 | - | VDDANA | V |
| Off [1][2] | Offset | Low power<br>COMPCTRLn.SPEED = 0x0 | -55 | -4/+2 | 51 | mV |
| | | High speed<br>COMPCTRLn.SPEED = 0x3 | -22 | -2/+1 | 20 | |
| $V_{HYS}$ [1][3] | Hysteresis | High speed<br>COMPCTRLn.SPEED = 0x3 | 39 | 106 | 156 | mV |
| $T_{PD}$ [1, 4] | Propagation Delay Vcm=Vddana/2<br>Vin = ±100mV overdrive from Vcm | Low power<br>COMPCTRLn.SPEED = 0x0 | - | 149 | 268 | ns |
| | | High speed<br>COMPCTRLn.SPEED =0x3 | - | 41 | 73 | |
| $T_{START}$ [1] | Startup time | Low power<br>COMPCTRLn.SPEED = 0x0 | - | 6.8 | 10.4 | µs |
| | | High speed<br>COMPCTRLn.SPEED = 0x3 | - | 2.2 | 3.7 | |
| $V_{SCALE}$ [1] | INL | | - | 0.569 | - | LSB |
| | DNL | | - | 0.053 | - | |
| | Offset Error | | - | 0.042 | - | |
| | Gain Error | | - | 0.041 | - | |

1. These are based on characterization.
2. Hysteresis disabled.
3. Hysteresis enabled.
4. Tpd is measured from Vin transition to ACOUT (AC direct output) toggle. It takes into account only analog propagation delay.


© 2026 Microchip Technology Inc. and its subsidiaries

### 47.4.7. Voltage Reference Characteristics

**Table 47-16.** Voltage Reference Characteristics(1)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INTREF | ADC, SDADC, DAC, AC Internal reference | nom. 1.024V VDDANA=5.0V Ta= 25°C | 1.003 | 1.024 | 1.045 | V |
| | | nom. 2.048V VDDANA=5.0V Ta= 25°C | 2.007 | 2.048 | 2.089 | |
| | | nom. 4.096V VDDANA=5.0V Ta= 25°C | 4.014 | 4.096 | 4.178 | |
| | Reference temperature coefficient | Drift over [-40, +25]°C | - | -0.016, 0.028 | - | %/°C |
| | | Drift over [+25, +85]°C | - | -0.022, 0.029 | - | |
| | | Drift over [+25, +105]°C | - | -0.031, 0.03 | - | |
| | Reference supply coefficient | Drift over [2.7, 5.5]V | - | -0.2, 0.3 | - | %/V |

1. These are based on characterization.

### 47.4.8. PTC Characteristics

**Table 47-17.** Sensor Load Capacitance (1)

| Symbol | Mode | PTC channel | Max. | Units |
|---|---|---|---|---|
| Cload | Self-capacitance | Y0 | 16 | pF |
| Cload | Self-capacitance | Y1 | 24 | pF |
| | | Y2 | | |
| | | Y3 | | |
| | | Y4 | | |
| | | Y5 | | |
| | | Y6 | | |
| | | Y7 | | |
| | | Y8 | | |
| | | Y9 | | |
| | | Y10 | | |
| | | Y11 | | |
| | | Y12 | | |
| | | Y13 | | |
| | | Y14 | | |
| | | Y15 | | |
| | | Y16 | | |
| | | Y17 | | |
| | | Y18 | | |
| | | Y19 | | |
| | | Y20 | | |
| | | Y21 | | |
| | | Y22 | | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-17.** Sensor Load Capacitance [1] (continued)

| Symbol | Mode | PTC channel | Max. | Units |
|---|---|---|---|---|
| Cload | Self-capacitance | Y23 | 22 | pF |
| | | Y24 | | |
| Cload | Self-capacitance | Y25 | 24 | pF |
| | | Y26 | | |
| | | Y27 | | |
| | | Y28 | | |
| | | Y29 | | |
| | | Y30 | | |
| | | Y31 | | |
| Cload | Mutual-capacitance | All | 31 | pF |

**Note:**

1. Capacitance load that the PTC circuitry can compensate for each channel.

#### 47.4.8.1. PTC Power Consumption

The values given in the table below are measured values of power consumption under the following conditions:

- **Operating conditions**
  - VDD = 5.0V
- **Clocks**
  - OSC48M used as main clock source, running undivided at 48 MHz
  - CPU is running on Flash with 2 wait states at 48 MHz
  - PTC running at 4 MHz
- **PTC configuration**
  - Mutual-Capacitance mode
  - One-touch channel
- **System configuration**
  - Standby Sleep mode enabled
  - RTC running on ULP32K: used to define the PTC scan rate through the event system
  - Drift Calibration disabled: no interrupts, PTC scans are performed in Standby mode
  - Drift Calibration enabled: RTC interrupts (wake up) the CPU to perform PTC scans. PTC drift calibration is performed every 1.5 sec.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-18.** Power Consumption [1]

| Symbol | Parameters | Drift Calibration | PTC scan rate | Oversamples | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|---|
| IDD | Current Consumption | Disabled | 10 | 4 | Max. 105°C Typ. 25°C | 24 | 609 | µA |
| | | | | 16 | | 43 | 633 | |
| | | | 50 | 4 | | 17 | 602 | |
| | | | | 16 | | 22 | 608 | |
| | | | 100 | 4 | | 16 | 601 | |
| | | | | 16 | | 19 | 605 | |
| | | | 200 | 4 | | 16 | 601 | |
| | | | | 16 | | 17 | 603 | |
| | | Enabled | 10 | 4 | | 30 | 620 | |
| | | | | 16 | | 50 | 644 | |
| | | | 50 | 4 | | 20 | 607 | |
| | | | | 16 | | 24 | 612 | |
| | | | 100 | 4 | | 18 | 606 | |
| | | | | 16 | | 21 | 608 | |
| | | | 200 | 4 | | 17 | 605 | |
| | | | | 16 | | 19 | 607 | |

**Note:**

1. These values are based on characterization.

## 47.5. NVM Characteristics

**Table 47-19.** NVM Max Speed caracteristics

| CPU $F_{MAX}$ (MHz) | 0WS | 1WS | 2WS |
|---|---|---|---|
| VDD>2.7V | 19 | 38 | 48 |
| VDD>4.5V | 19 | 38 | 48 |

## 47.6. Oscillator Characteristics

### 47.6.1. Crystal Oscillator (XOSC) Characteristics

The following table describes the characteristics for the oscillator when a digital clock is applied on XIN.

**Table 47-20.** Digital Clock Characteristics

| Symbol | Parameter | Condition | Min | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $f_{CPXIN}$ | XIN clock frequency | Digital mode | - | - | 48 | MHz |
| $DC_{XIN}$ [1] | XIN clock duty cycle | Digital mode | 40 | 50 | 60 | % |

1. These values are based on simulation and are not covered by test or characterization

The following table describes the characteristics for the oscillator when a crystal is connected between XIN and XOUT as shown in the figure below. The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of $C_L$ can be found in the crystal data sheet. The capacitance of the external capacitors ($C_{LEXT}$) can then be computed as follows:

$$C_{LEXT} = 2(C_L + - C_{STRAY} - C_{SHUNT})$$

where,

$C_{STRAY}$ is the capacitance of the pins and PCB, $C_{SHUNT}$ is the shunt capacitance of the crystal.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 47-5.** Oscillator Connection

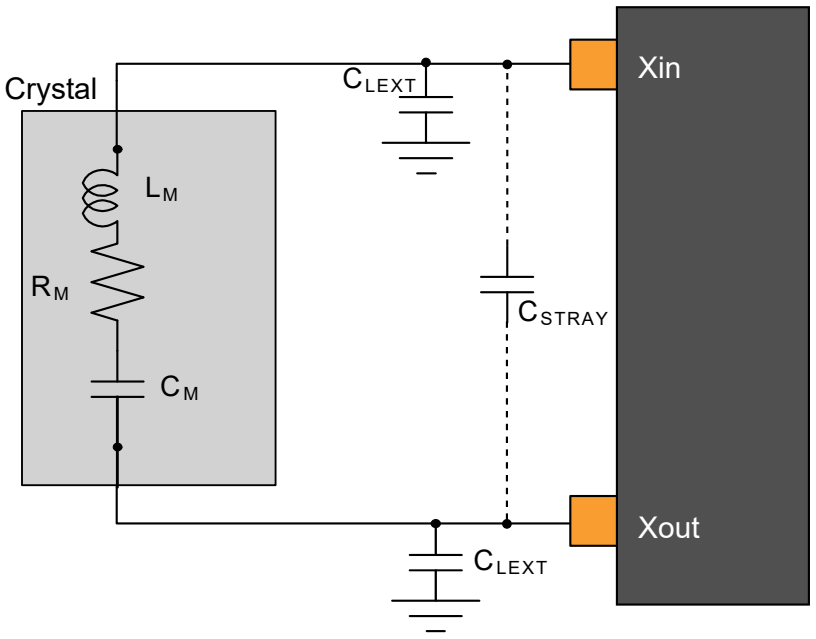



**Table 47-21.** Multi Crystal Oscillator Electrical Characteristics [1]

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Crystal oscillator frequency | | 0.4 | - | 32 | MHz |
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 0.455 MHz<br>CL = 100 pF<br>XOSC.GAIN = 0 | - | - | 443 | Ω |
| | | F = 2 MHz<br>CL = 20 pF<br>XOSC.GAIN=0 | - | - | 383 | |
| | | F = 4 MHz<br>CL = 20 pF<br>XOSC.GAIN = 1 | - | - | 218 | |
| | | F = 8 MHz<br>CL= 20 pF<br>XOSC.GAIN = 2 | - | - | 114 | |
| | | F = 16 MHz<br>CL=20 pF<br>XOSC.GAIN = 3 | - | - | 58 | |
| | | F = 32 MHz<br>CL= 12 pF<br>XOSC.GAIN = 4 | - | - | 62 | |
| Cxin | Parasitic load capacitor | | - | 6.7 | - | pF |
| Cxout | | | - | 4.1 | - | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 47-21.** Multi Crystal Oscillator Electrical Characteristics [1] (continued)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Tstart | Startup time | F = 2 MHz<br>CL=20 pF<br>XOSC.GAIN = 0 | - | 12.3 | 48.7 | KCycles |
| | | F = 4 MHz<br>CL= 20 pF<br>XOSC.GAIN = 1 | - | 8.2 | 30.1 | |
| | | F = 8 MHz<br>CL=20 pF<br>XOSC.GAIN = 2 | - | 6.2 | 19.9 | |
| | | F = 16 MHz<br>CL= 20 pF<br>XOSC.GAIN = 3 | - | 10.8 | 30.1 | |
| | | F = 32 MHz<br>CL= 12 pF<br>XOSC.GAIN = 4 | - | 8.7 | 23.6 | |

1. These values are based on characterization.

### 47.6.2. External 32.768 kHz Crystal Oscillator (XOSC32K) Characteristics

The following table describes the characteristics for the oscillator when a digital clock is applied on XIN32 pin.

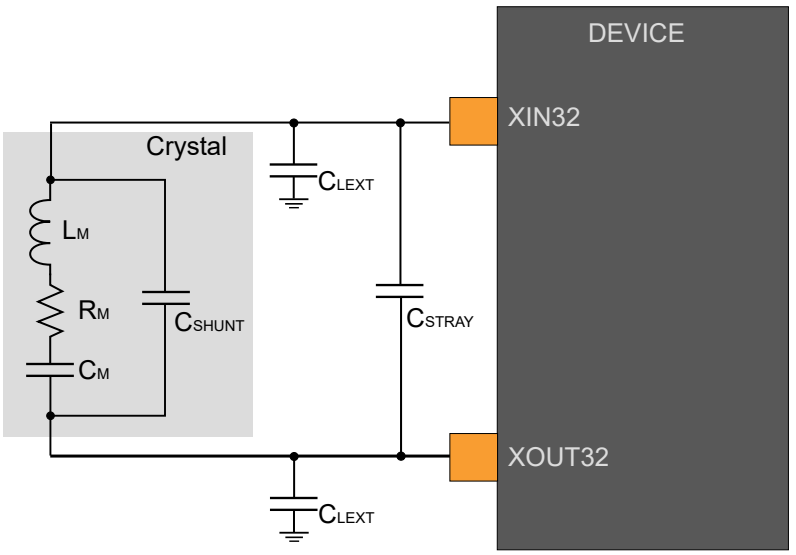
**Table 47-22.** Digital Clock Characteristics[1]

| Symbol | Parameter | Condition | Typ | Units |
|---|---|---|---|---|
| $f_{CPXIN32}$ | XIN32 clock frequency | Digital mode | 32.768 | kHz |
| $DC_{XIN32}$ | XIN32 clock duty cycle | Digital mode | 50 | % |

1. These are based on simulation. These values are not covered by test or characterization

The following table describes the characteristics for the oscillator when a crystal is connected between XIN32 and XOUT32.

**Figure 47-6.** Oscillator Connection





© 2026 Microchip Technology Inc. and its subsidiaries

The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of CL can be found in the crystal data sheet. The capacitance of the external capacitors (CLEXT) can then be computed as follows:

$C_{LEXT}=2\ (C_L-C_{STRAY}-C_{SHUNT})$

where $C_{STRAY}$ is the capacitance of the pins and PCB and $C_{SHUNT}$ is the shunt capacitance of the crystal.

**Table 47-23.** 32.768 kHz Crystal Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$ (1) | Crystal oscillator frequency | | - | 32768 | - | Hz |
| $C_L$ (1) | Crystal load capacitance | | - | - | 12.5 | pF |
| $C_{SHUNT}$ (1) | Crystal shunt capacitance | | - | - | 1.75 | pF |
| $C_M$ (1) | Motional capacitance | | - | 1.25 | - | fF |
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 32.768 kHz, $C_L$=12.5 pF | - | - | 70 | kΩ |
| $C_{XIN32K}$ | Parasitic capacitor load | | - | 3.8 | - | pF |
| $C_{XOUT32K}$ | | | - | 4.1 | - | pF |
| $T_{START}$ | Startup time | F = 32.768 kHz, $C_L$=12.5 pF | - | 16 | 24 | Kcycles |

1. These are based on simulation. These values are not covered by test or characterization

### 47.6.3. 32.768 kHz Internal Oscillator (OSC32K) Characteristics

**Table 47-24.** 32.768 kHz RC Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $F_{OUT}$ | Output frequency | Ta=25°C<br>VDDANA = 5.0V | 30.965 | 32.768 | 34.570 | kHz |
| | | Ta=25°C<br>Over [2.7, 5.5]V | 29.164 | 32.768 | 36.044 | kHz |
| | | Over [-40,105]°C<br>Over [2.7, 5.5]V | 25.559 | 32.768 | 37.683 | kHz |
| $T_{STARTUP}$ | Startup time | | - | 1 | 2 | cycles |
| Duty (1) | Duty cycle | | - | 50 | - | % |

1. These are based on simulation. These values are not covered by test or characterization.

### 47.6.4. 48MHz RC Oscillator (OSC48M) Characteristics

**Table 47-25.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | $F_{OUT}$ = 48 MHz<br>VDD =5.0V | Max 105°C<br>Typ 25°C | 452 | 485 | µA |

1. These are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

# 48. Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 E/G/J)

**Note:** TSENS is not available in AEC - Q100 qualified device part numbers.

## 48.1. Disclaimer

All typical values are measured at Ta = 25°C unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

This chapter contains only characteristics specific for the SAM C20/C21 E/G/J AEC - Q100 Grade 1. For all other values or missing characteristics, refer to the SAM C20/C21 E/G/J 85°C chapter.

## 48.2. General Operating Ratings

The device must operate within the ratings listed in the table below in order for all other electrical characteristics and typical characteristics of the device to be valid.

NVM erase operations are not protected by the BODVDD and BODCORE in debugger cold-plugging mode. NVM erase operation at supply voltages below product specification minimum can cause corruption of the calibration and other areas mandatory for a correct product behavior.

**Table 48-1.** General operating conditions

| Symbol | Parameter | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|
| $T_A$ | Temperature range | -40 | 25 | 125 | °C |
| $T_J$ | Junction temperature | - | - | 145 | °C |

## 48.3. Supply Characteristics

**Table 48-2.** Power Supply Current Requirement

| Symbol | Conditions | Current | Units |
|---|---|---|---|
| | | Max | |
| $I_{INPUT}$(1) | Power up Maximum current | 2.5 | mA |

1. $I_{INPUT}$ is the minimum requirement for the power supply connected to the device, until the device comes out of POR.

## 48.4. Power Consumption

The values in the Power Consumption table below are measured values of power consumption under the following conditions, except where noted:

- Operating conditions
  - VDDIN = 3.0 V, 5.0V
- Oscillators
  - XOSC (crystal oscillator) stopped
  - XOSC32K (32.768 kHz crystal oscillator) running with external 32.768 kHz crystal
  - FDPLL using XOSC32K as reference and running at 48 MHz
- Clocks
  - FDPLL used as main clock source, except otherwise specified
  - CPU, AHB clocks undivided
  - All peripheral clocks stopped
- I/Os are inactive with input trigger disable
- CPU is running on Flash with Wait states specified in NVM Max Speed Characteristics


© 2026 Microchip Technology Inc. and its subsidiaries

- NVMCTRL cache enabled
- BODVDD disabled

**Table 48-3.** Current Consumption (1)

| Mode | conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 3.8 | 4.2 | mA |
| | | 125°C | 5.0V | 4.4 | 5.4 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 3.7 | 4.2 | mA |
| | | 125°C | 3.0V | 4.4 | 5.4 | |
| | CPU running a While 1 algorithm, with GCLKIN as reference | 25°C | 5.0V | 71*Freq+160 | 79*Freq+166 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 72*Freq+659 | 68*Freq+1866 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 4.7 | 5.2 | mA |
| | | 125°C | 5.0V | 5.3 | 6.4 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 4.7 | 5.2 | mA |
| | | 125°C | 3.0V | 5.3 | 6.3 | |
| | CPU running a Fibonacci algorithm, with GCLKIN as reference | 25°C | 5.0V | 90*Freq+163 | 100*Freq+173 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 91*Freq+664 | 88*Freq+1863 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 5.9 | 6.5 | mA |
| | | 125°C | 5.0V | 6.7 | 8.0 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 5.2 | 5.7 | mA |
| | | 125°C | 3.0V | 5.9 | 6.9 | |
| | CPU running a CoreMark algorithm, with GCLKIN as reference | 25°C | 5.0V | 115*Freq+167 | 127*Freq+169 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 119*Freq+668 | 122*Freq+1849 | |
| IDLE0 | | 25°C | 5.0V | 1.8 | 1.9 | mA |
| | | 125°C | 5.0V | 2.3 | 3.4 | |
| IDLE2 | | 25°C | 5.0V | 1.2 | 1.3 | mA |
| | | 125°C | 5.0V | 1.8 | 3.1 | |
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37 | µA |
| | | 125°C | 5.0V | 363.4 | 950 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35 | |
| | | 125°C | 5.0V | 361.4 | 953 | |

1. The values are based on characterization.

## 48.5. I/O Pin Characteristics

Two different pin types with two different speeds are Normal and High-Sink pins(2). The Drive Strength bit is located in the Pin Configuration register PORT (PORT.PINCFG.DRVSTR).

The pins with $I^2C$ alternative mode available are compliant with $I^2C$ specifications. All $I^2C$ pins support Standard (Sm), Fast (Fm), Fast plus (Fm+) and High-Speed (HS) modes. The available $I^2C$ pins are listed in the I/O Multiplexing section.

The RESET_N pin has the same properties as standard GPIOs.

**Table 48-4.** I/O Pins Dynamic Characteristics (1)

| Symbol | Parameter | Conditions | Normal pins | High-Sink pins | Normal pins | High-Sink pins | Units |
|---|---|---|---|---|---|---|---|
| | | | DRVSTR = 0 | | DRVSTR = 1 | | |
| $t_{RISE}$ | Maximum rise time | VDD = 5.0V, load = 20 pF | 15 | 12 | 8 | 11 | ns |
| $t_{FALL}$ | Maximum fall time | VDD = 5.0V, load = 20 pF | 15 | 11 | 7 | 10 | |


© 2026 Microchip Technology Inc. and its subsidiaries

1. These values are based on simulation and not covered by test limits in production or characterization.
2. The following pins are High-Sink pins and have different properties than normal pins: PA10, PA11, PB10, PB11.

## 48.6. Analog Characteristics

### 48.6.1. BODVDD - Brown-out Detector Characteristics

Refer to the NVM User Row Mapping for the BODVDD default value settings. These values are based on simulation and are not covered by test limits in production or characterization.

**Table 48-5.** BODVDD Characteristics(1) (2)

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| VBOD+ | BODVDD high threshold Level | VDD level, Bod setting = 8 (default) | - | 2.86 | 3.00 | V |
| | | VDD level, Bod setting = 44 | - | 4.57 | 4.85 | |
| VBOD- / VBOD | BODVDD low threshold Level | VDD level, Bod setting = 8 (default) | 2.60 | 2.8 | 2.95 | |
| | | VDD level, Bod setting = 44 | 4.10 | 4.51 | 4.80 | |
| | Step size | | - | 60 | - | mV |
| VHys | Hysteresis (VBOD+ - VBOD-) BODVDD.LEVEL = 8 to 48 | VDD | 40 | - | 75 | mV |
| Tstart | Startup time | time from enable to RDY | - | 8 | - | us |

**Notes:**

1. These values are based on characterization.
2. BODVDD in continuous mode.

**Table 48-6.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | IDLE, Mode CONT | VDD = 2.7V | Max. 125°C | 22.5 | 32.1 | µA |
| | | VDD = 5.0V | Typ. 25°C | 41.0 | 51.4 | |
| | IDLE, Mode SAMPL | VDD = 2.7V | | 0.1 | 4.4 | |
| | | VDD = 5.0V | | 0.1 | 3.3 | |
| | STANDBY, Mode SAMPL | VDD = 2.7V | | 0.8 | 4.0 | |
| | | VDD = 5.0V | | 3.5 | 6.3 | |

1. These values are based on characterization.

### 48.6.2. Analog-to-Digital (ADC) Characteristics

**Table 48-7.** Differential Mode (1)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min. | Typ. | Max. | |
| ENOB | Effective Number of bits | R2R Disabled | VDDANA = 5.0V, VREF = VDDANA | 9.7 | 10.6 | 11.3 | bits |
| | | | VDDANA = 2.7V, VREF = 2.0V | 9.8 | 10.6 | 11.2 | |
| | | R2R Enabled | VDDANA = 5.0V, VREF = VDDANA | 9.7 | 11.1 | 11.8 | |
| TUE | Total Unadjusted Error | R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | VDDANA = 5.0V, VREF = VDDANA | - | +/-4.2 | +/-7.1 | LSB |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-4.8 | +/-8.3 | |
| INL | Integral Non Linearity | R2R disabled | VDDANA = 5.0V, VREF = VDDANA | - | +/-1.5 | +/-4 | LSB |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-3.2 | +/-4.1 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 48-7.** Differential Mode [1] (continued)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min. | Typ. | Max. | |
| DNL | Differential Non Linearity | R2R disabled | VDDANA = 5.0V, VREF = VDDANA | - | -0.8/+1.1 | -1/+2.0 | LSB |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | -0.9/+1.3 | -1/+2.2 | |
| GE | Gain Error | R2R disabled w/o gain compensation | VDDANA = 5.0V, VREF = VDDANA | - | +/-0.8 | +/-1.8 | % |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-0.9 | +/-3.2 | |
| | | | VDDANA = 5.0V, 1V internal Ref | - | +/-1.9 | +/-7.3 | |
| | | | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-0.1 | +/-1.3 | |
| | | R2R disabled with gain compensation | VDDANA = 2.7V, VREF = 2.0V | - | +/-0.1 | +/-0.5 | |
| | | | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-0.2 | +/-0.7 | |
| OE | Offset Error | R2R disabled without offset compensation | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-0.2 | +/-25 | mV |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-1.4 | +/-17 | |
| | | R2R disabled with offset compensation | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-2 | +/-10 | |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-0.2 | +/-11 | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps/Fin = 14 kHz/Full range Input signal VDDANA = 5.0V, VREF = VDDANA | 63 | 75 | 81 | dB |
| SINAD | | Signal to Noise and Distortion ratio | | 60 | 67 | 70 | |
| SNR | | Signal to Noise ratio | | 64 | 68 | 70 | |
| THD | | Total Harmonic Distortion | | -81 | -74 | -63 | |
| | | Noise RMS | External Reference voltage | - | 0.5 | 3.3 | mV |

**Note:**

1. These values are based on characterization. These values are not covered by test limits in production.

**Table 48-8.** Single Ended Mode [1]

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| ENOB | Effective Number of bits | R2R disabled | VDDANA = 3.0V, VREF = VDDANA | 8.9 | 9.7 | 10.0 | bits |
| | | | VDDANA = 3.0V, VREF = 2.0V | 8.9 | 9.2 | 9.7 | |
| TUE | Total Unadjusted Error | R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | VDDANA = 5.0V, VREF = VDDANA | - | 18 | +/-28 | LSB |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | 30 | +/-57 | |
| INL | Integral Non Linearity | R2R disabled | VDDANA = 5.0V, VREF = VDDANA | - | +/-2.2 | +/-7.5 | LSB |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-4.1 | +/-7.1 | |
| DNL | Differential Non Linearity | R2R disabled | VDDANA = 5.0V, VREF = VDDANA | - | -0.8/+1 | -1/+1.9 | |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | -1/+1.1 | -1/+2.5 | |
| GE | Gain Error | R2R disabled w/o gain compensation | VDDANA = 5.0V, VREF = VDDANA | - | +/-0.4 | +/-1.1 | % |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-0.7 | +/-1.9 | |
| | | | VDDANA = 5.0V, 1V internal Ref | - | +/-1.6 | +/-7.4 | |
| | | | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-0.2 | +/-1.4 | |
| | | R2R disabled with gain compensation | VDDANA = 2.7V, VREF = 2.0V | - | +/-0.3 | +/-0.9 | |
| | | | VDDANA = 5.0V, VREF = VDDANA/2 | - | +/-0.1 | +/-1.4 | |
| OE | Offset Error | R2R disabled | VDDANA = 5.0V, VREF = VDDANA | - | +/-31 | +/-76 | mV |
| | | | VDDANA = 2.7V, VREF = 2.0V | - | +/-2.2 | +/-64 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 48-8.** Single Ended Mode [1] (continued)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps/Fin = 14 kHz/Full range Input signal VDDANA = 5.0V, VREF = VDDANA | 57 | 71 | 73 | dB |
| SINAD | | Signal to Noise and Distortion ratio | | 54 | 60 | 62 | |
| SNR | | Signal to Noise ratio | | 57 | 61 | 62 | |
| THD | | Total Harmonic Distortion | | -71 | -70 | -56 | |
| | | Noise RMS | External Reference voltage | - | 0.7 | 2 | mV |

**Note:**

1. These values are based on characterization. These values are not covered by test limits in production.

**Table 48-9.** Power Consumption [1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Differential mode | fs = 1Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = VREF = 5.5V | Max 125°C Typ 25°C | 905 | 1162 | uA |
| | | fs = 1Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111' VDDANA = VREF = 5.5V | | 1144 | 1507 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | | 381 | 555 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | | 609 | 952 | |
| | Single Ended mode | fs = 1Msps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | Max 125°C Typ 25°C | 984 | 1183 | uA |
| | | fs = 1Msps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | | 1178 | 1548 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | | 437 | 629 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '110', BIASREFCOMP = '111', VDDANA = VREF = 5.5V | | 635 | 983 | |

**Note:**

1. These values are based on characterization.

### 48.6.3. Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics

**Table 48-10.** SDADC DC Performance: Differential Input Mode, Chopper ON [1]

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| INL | Integral Non Linearity | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-2.9 | +/-4.1 | LSB |
| | | CLK_SDADC = 3 MHz, INT VREF = 5.5V | - | +/-8.4 | +/-9.3 | |
| DNL | Differential Non Linearity | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +1.5/-1 | +2.2/-1 | LSB |
| | | CLK_SDADC = 3 MHz, INT VREF = 5.5V | - | +2.1/-1 | +2.9/-1 | |
| Eg | Gain Errors | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-0.7 | +/-2.4 | % |
| | | CLK_SDADC = 3 MHz, INT VREF = 5.5V | - | +/-0.9 | +/-2.2 | |
| TCg | Gain Drift | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | -6.9 | 4.4 | 17.5 | ppm/°C |
| Off | Offset Error | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-3.1 | +/-10.7 | mV |
| | | CLK_SDADC = 3 MHz, INT VREF = 5.5V | - | +/-0.5 | +/-3.3 | |
| Tco | Offset Error Drift | CLK_SDADC = 3 MHz, VREF = 1.2V, VDDANA = 2.7V | -1.5 | -0.1 | 1.2 | µV/°C |


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:**

1. OSR = 256

**Table 48-11.** SDADC DC Performance: Differential Input Mode, Chopper OFF [1]

| Symbol | Parameters | Conditions | Min. | Typ | Max. | Unit |
|---|---|---|---|---|---|---|
| INL | Integral Non Linearity | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-5.5 | +/-10.2 | LSB |
| | | CLK_SDADC = 6 MHz, INT VREF = 5.5V | - | +/-8.9 | +/-10.8 | |
| DNL | Differential Non Linearity | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +2.8/-1 | +4.1/-1 | LSB |
| | | CLK_SDADC = 6 MHz, INT VREF = 5.5V | - | +2.5/-1 | +4.8/-1 | |
| Eg | Gain Errors | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-0.7 | +/-3.1 | % |
| | | CLK_SDADC = 6 MHz, INT VREF = 5.5V | - | +/-0.9 | +/-2.2 | |
| TCg | Gain Drift | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | -19.7 | 5.2 | 20.9 | ppm/°C |
| Off | Offset Error | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | - | +/-2.2 | +/-21.2 | mV |
| | | CLK_SDADC = 6 MHz, INT VREF = 5.5V | - | +/-4.9 | +/-25.7 | |
| Tco | Offset Error Drift | CLK_SDADC = 6 MHz, VREF = 1.2V, VDDANA = 2.7V | -14.2 | 12.4 | 60 | µV/°C |
| Input noise rms | AC Input noise rms | OSR = 256, VREF = 1.2V, VDDANA = 2.7V | - | 19 | 20 | µVrms |
| | | OSR = 256, VREF = 5.5V | - | 59 | 76 | |

**Note:**

1. OSR = 256

**Table 48-12.** SDADC AC Performance: Differential Input Mode [1]

| Symbol | Parameters | Conditions [2] | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| ENOB | Effective Number Of Bits | Ext ref = 1.2V, VDDANA = 2.7V | 12 | 14.2 | - | bit |
| | | Int Ref = 5.5V | 11 | 11.2 | - | |
| DR | Dynamic Range | Ext ref = 1.2V, VDDANA = 2.7V | 89.0 | 91.0 | - | dB |
| | | Int Ref = 5.5V | 83.0 | 92.0 | - | |
| SNR | Signal to Noise Ratio | Ext ref = 1.2V, VDDANA = 2.7V | 68.7 | 88 | - | dB |
| | | Int Ref = 5.5V | 77 | 79 | - | |
| SINAD | Signal to Noise + Distortion Ratio | Ext ref = 1.2V, VDDANA = 2.7V | 73.9 | 87 | - | dB |
| | | Int Ref = 5.5V | 68 | 69 | - | |
| THD | Total Harmonic Distortion | Ext ref = 1.2V, VDDANA = 2.7V | - | -94.6 | -74.4 | dB |
| | | Int Ref = 5.5V | - | -69 | -68 | |

**Notes:**

1. These values are based on characterization.
2. OSR = 256, Chopper OFF, Sampling Clock Speed at 6Mhz, Fin = 13 kHz.

**Table 48-13.** Power Consumption [1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Power consumption | CTLSDADC = 0x0 External Ref - VDDANA = 5.5V, VREF = 2V Ref buf on SCLK_SDADC = 6 MHz | Max 125°C Typ 25°C | 644 | 764 | uA |
| | | CTLSDADC=0x0 Internal Ref - VDDANA = VREF = 5.5V Ref buf off SCLK_SDADC = 6 MHz | | 605 | 696 | uA |

**Note:**

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 48.6.4. Digital-to-Analog Converter (DAC) Characteristics

**Table 48-14.** Accuracy Characteristics [1]

| Symbol | Parameter | Conditions | | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INL | Integral non-linearity | VREF = Ext 2.0V | VDD = 2.7V | +/-0.7 | +/-2.4 | LSB |
| | | | VDD = 5.5V | +/-0.6 | +/-1.7 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.6 | +/-2 | |
| | | | VDD = 5.5V | +/-0.5 | +/-1.7 | |
| | | VREF = 1.024V INTREF | VDD = 2.7V | +/-1 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.5 | +/-3.5 | |
| DNL | Differential non-linearity | VREF = Ext 2.0V | VDD = 2.7V | +/-0.5 | +/-2.3 | LSB |
| | | | VDD = 5.5V | +/-0.5 | +/-2.2 | |
| | | VREF = VDDANA | VDD = 2.7V | +/-0.4 | +/-2.1 | |
| | | | VDD = 5.5V | +/-0.4 | +/-2.1 | |
| | | VREF= 1.024V INTREF | VDD = 2.7V | +/-1 | +/-2.5 | |
| | | | VDD = 5.5V | +/-1.4 | +/-3.5 | |
| GE | Gain error | Ext. VREF | | +/-10 | +/-28 | mV |
| OE | Offset error | Ext. VREF | | +/-4 | +/-26 | mV |

**Note:**

1. These values are based on characterization and are not covered by test limits in production.

**Table 48-15.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | DC supply current | Output buffer On VREF = VDDANA = 5.0V | Max. 125°C Typ. 25°C | 318 | 481 | µA |
| | | Output buffer Off VREF = VDDANA = 5.0V | | 74 | 100 | |

1. These values are based on characterization.

### 48.6.5. Analog Comparator (AC) Characteristics

**Table 48-16.** Analog Comparator Characteristics

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| Off [1][2] | Offset | Low-power COMPCTRLn.SPEED = 0x0 | -62 | +/-3 | 62 | mV |
| | | High-speed COMPCTRLn.SPEED = 0x3 | -25 | +/-2 | 25 | |
| Vhys [1][3] | Hysteresis | Low-power COMPCTRLn.SPEED = 0x0 | 25 | 100 | 276 | mV |
| | | High-speed COMPCTRLn.SPEED = 0x3 | 29 | 100 | 211 | |
| Tstart [1] | Startup time | Low-power COMPCTRLn.SPEED = 0x0 | - | 7.4 | 13.6 | us |
| | | High-speed COMPCTRLn.SPEED = 0x3 | - | 2.1 | 4 | |
| Vscale [1] | INL | - | - | 0.51 | - | LSB |
| | DNL | - | - | 0.04 | - | LSB |
| | Offset Error | - | - | 0.05 | - | LSB |
| | Gain Error | - | - | 0.03 | - | LSB |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. These values are based on characterization.
2. Hysteresis disabled.
3. Hysteresis enabled.

**Table 48-17.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDDANA | Current consumption - $V_{CM}$= VDDANA/2 ± 100 mV overdrive from Vcm Voltage scaler disabled | COMPCTRLn.SPEED = 0x0 VDDANA = 5.0V | Max 125°C Typ 25°C | 10 | 20 | µA |
| | | COMPCTRLn.SPEED = 0x3 VDDANA = 5.0V | | 39 | 66 | |
| | Current consumption Voltage scaler only | VDDANA = 5.0V | | 43 | 68 | |

1. These values are based on characterization.

### 48.6.6. Voltage Reference Characteristics

**Table 48-18.** Voltage Reference Characteristics(1)

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| INTREF | Reference temperature coefficient | Drift over [+25, +125]°C | - | -0.015, 0.03 | - | %/°C |

1. These values are based on characterization.

### 48.6.7. PTC Characteristics

#### 48.6.7.1. PTC Power Consumption

The values given in the table below are measured values of power consumption under the following conditions:

- **Operating conditions**
  - VDD = 5.0V
- **Clocks**
  - OSC48M used as main clock source, running undivided at 48 MHz
  - CPU is running on Flash with 2 wait states, at 48 MHz
  - PTC running at 4 MHz
- **PTC configuration**
  - Mutual-Capacitance mode
  - One-touch channel
- **System configuration**
  - Standby Sleep mode enabled.
  - RTC running on ULP32K: Used to define the PTC scan rate, through the event system.
  - Drift Calibration disabled: No interrupts, PTC scans are performed in Standby mode
  - Drift Calibration enabled: RTC interrupts (wakeup) the CPU to perform PTC scans. PTC drift calibration is performed every 1.5 sec.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 48-19.** Power Consumption [1]

| Symbol | Parameters | Drift Calibration | PTC scan rate | Oversamples | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|---|---|
| IDD | Current Consumption | Disabled | 10 | 4 | Max 125°C Typ 25°C | 24 | 909 | µA |
| | | | | 16 | | 43 | 933 | |
| | | | 50 | 4 | | 17 | 902 | |
| | | | | 16 | | 22 | 908 | |
| | | | 100 | 4 | | 16 | 901 | |
| | | | | 16 | | 19 | 905 | |
| | | | 200 | 4 | | 16 | 901 | |
| | | | | 16 | | 17 | 903 | |
| | | Enabled | 10 | 4 | | 30 | 920 | |
| | | | | 16 | | 50 | 944 | |
| | | | 50 | 4 | | 20 | 907 | |
| | | | | 16 | | 24 | 912 | |
| | | | 100 | 4 | | 18 | 906 | |
| | | | | 16 | | 21 | 908 | |
| | | | 200 | 4 | | 17 | 905 | |
| | | | | 16 | | 19 | 907 | |

**Note:**

1. These values are based on characterization.

## 48.7. NVM Characteristics

**Table 48-20.** NVM Max Speed Characteristics

| CPU $F_{MAX}$ (MHz) | 0WS | 1WS | 2WS | 3WS |
|---|---|---|---|---|
| VDD>2.7V | 14 | 35 | 47 | 48 |
| VDD>4.5V | 17 | 35 | 48 | 48 |

**Table 48-21.** Flash Endurance and Data Retention

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Cyc_{NVM}$ | Cycling Endurance[1] | -40°C < $T_A$ < 125°C | 5k | - | Cycles |

1. An endurance cycle is a write and an erase operation.

**Table 48-22.** EEPROM Emulation[1] Endurance and Data Retention

| Symbol | Parameter | Conditions | Min. | Typ. | Units |
|---|---|---|---|---|---|
| $Cyc_{EEPROM}$ | Cycling Endurance[2] | -40°C < Ta < 125°C | 20k | - | Cycles |

1. The EEPROM emulation is a software emulation as described in the *"AT03265.Application Note"*.
2. An endurance cycle is a write and an erase operation.

## 48.8. Oscillator Characteristics

### 48.8.1. Crystal Oscillator (XOSC) Characteristics

#### 48.8.1.1. Crystal Oscillator Characteristics

**Table 48-23.** Multi Crystal Oscillator Electrical Characteristics [1]

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| CL | Crystal Load | F = 32 MHz | - | - | 12 | pF |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 48-23.** Multi Crystal Oscillator Electrical Characteristics [1] (continued)

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 16 MHz, CL= 20 pF, XOSC, GAIN = 3 | - | - | 58 | Ω |
| | | F = 32 MHz, CL = 12 pF, XOSC, GAIN = 4 | - | - | 62 | |
| Tstart | Startup time | F = 2 MHz, CL = 20 pF, XOSC,GAIN = 0 | - | 12.3 | 54.7 | K cycles |
| | | F = 4 MHz, CL = 20 pF, XOSC ,GAIN = 1 | - | 8.2 | 33.9 | |
| | | F = 8 MHz, CL = 20 pF, XOSC,GAIN = 2 | - | 6.2 | 22.2 | |
| | | F = 16 MHz, CL= 20 pF, XOSC,GAIN = 3 | - | 10.8 | 33.8 | |
| | | F = 32 MHz, CL= 12 pF, XOSC,GAIN = 4 | - | 8.7 | 25.6 | |

**Note:**

1. These values are based on characterization.

**Table 48-24.** Power Consumption [1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| IDD | Current consumption | F = 2 MH, CL= 20 pF, XOSC,GAIN = 0,VDD = 5.0V, AGC = OFF | Max. 125°C Typ 25°C | 150 | 224 | µA |
| | | AGC = ON | | 138 | 218 | |
| | | F = 4 MHz, CL = 20 pF, XOSC,GAIN = 1,VDD = 5.0V, AGC = OFF | | 220 | 315 | |
| | | AGC = ON | | 175 | 296 | |
| | | F = 8 MHz, CL = 20 pF, XOSC,GAIN = 2,VDD = 5.0V, AGC = OFF | | 350 | 463 | |
| | | AGC = ON | | 247 | 361 | |
| | | F = 16 MHz, CL=20 pF XOSC,GAIN = 3,VDD = 5.0V, AGC = OFF | | 663 | 924 | |
| | | AGC = ON | | 429 | 840 | |
| | | F = 32 MHz - CL= 12 pF, XOSC,GAIN = 4,VDD = 5.0V, AGC = OFF | | 1975 | 2806 | |
| | | AGC = ON | | 874 | 1436 | |

**Note:**

1. These values are based on characterization.

## 48.8.2. External 32.768 kHz Crystal Oscillator (XOSC32K) Characteristics

### 48.8.2.1. Crystal Oscillator Characteristics

The following table describes the characteristics for the oscillator when a crystal is connected between XIN32 and XOUT32.

**Table 48-25.** 32.768 kHz Crystal Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| ESR | Crystal Equivalent Series Resistance - SF = 3 | F = 32.768 kHz, $C_L$=12.5 pF | - | - | 70 | kΩ |
| Tstart | Startup time | F = 32.768 kHz, $C_L$=12.5 pF | - | 16 | 26 | Kcycles |

1. These are based on simulation. These values are not covered by test or characterization

**Table 48-26.** Power Consumption[1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | VDD = 5.0V | Max 125°C<br>Typ 25°C | 1528 | 1900 | nA |


© 2026 Microchip Technology Inc. and its subsidiaries

1. These are based on characterization.

### 48.8.3. Digital Phase Locked Loop (DPLL) Characteristics

**Table 48-27.** Digital Phase Locked Loop Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| fIN (1) | Input frequency | | 32 | - | 2000 | KHz |
| fOUT (1) | Output frequency | | 48 | - | 96 | MHz |
| Jp (2) | Period jitter (Peak-Peak value) | fIN= 32 kHz, fOUT= 48 MHz | - | 1.5 | 4.0 | % |
| | | fIN= 32 kHz, fOUT= 96 MHz | - | 2.7 | 10.0 | |
| | | fIN= 2 MHz, fOUT= 48 MHz | - | 1.8 | 5.0 | |
| | | fIN= 2 MHz, fOUT= 96 MHz | - | 2.5 | 8.0 | |
| tLOCK (2) | Lock Time | After startup, time to get lock signal. fIN= 32 kHz, fOUT= 96 MHz | - | 1.1 | 1.6 | ms |
| | | After startup, time to get lock signal. fIN= 2 MHz, fOUT= 96 MHz | - | 25 | 40 | µs |
| Duty (1) | Duty cycle | | - | 50 | - | % |

1. These values are based on simulation. These values are not covered by test limits in production or characterization.
2. These values are based on characterization.

**Table 48-28.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD | Current Consumption | Ck=48MHz, VDD=5.0V | Max 125°C | 536 | 693 | µA |
| | | Ck=96MHz, VDD=5.0V | Typ 25°C | 865 | 1048 | |

1. These values are based on characterization.

### 48.8.4. 32.768 kHz Internal Oscillator (OSC32K) Characteristics

**Table 48-29.** 32.768 kHz RC Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output frequency | at 25 degC , at Vddana = 5.0V | 31.948 | 32.768 | 33.587 | kHz |
| | | at 25 degC , over [2.7, 5.5]V | 28.835 | 32.768 | 36.700 | kHz |
| | | over[-40,125]C, over [2.7, 5.5]V | 24.282 | 32.768 | 38.010 | kHz |
| Tstartup | Startup time | | - | 1 | 2 | cycles |
| Duty (1) | Duty Cycle | | - | 50 | - | % |

1. These values are based on simulation, and are not covered by test or characterization.

**Table 48-30.** Power Consumption(1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD | Current consumption | VDD = 5.0V | Max 125°C<br>Typ 25°C | 0.864 | 1.4 | µA |

1. These values are based on characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 48.8.5. Ultra Low-Power Internal 32.768 kHz RC Oscillator (OSCULP32K) Characteristics

**Table 48-31.** Ultra Low-Power Internal 32.768 kHz RC Oscillator Electrical Characteristics

| Symbol | Parameter | Condition | Min. | Typ | Max. | Units |
|---|---|---|---|---|---|---|
| $f_{OUT}$ | Output frequency | Over [-40, 125]°C, over [2.7, 5.5]V | 22.937 | 32.768 | 40.960 | kHz |
| Duty | Duty Cycle | | | 50 | | % |

### 48.8.6. 48 MHz RC Oscillator (OSC48M) Characteristics

**Table 48-32.** RC 48 MHz Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $F_{OUT}$ | Output frequency (1) | -40 to 125 °C | 45.24 | 48 | 50.76 | MHz |
| | | | -5.75 | - | +5.75 | % Error |
| $T_{START}$ (2) | Startup time | | - | 3.9 | 15 | µs |
| Duty (3) | Duty Cycle | | - | 50 | - | % |

1. Standard Accuracy factory calibration.
2. OSC48MSTUP.STARTUP field must be set accordingly.
3. These values are based on simulation, and are not covered by test or characterization.

**Table 48-33.** Power Consumption

| Symbol | Parameters | Conditions | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| $I_{DD}$ | Current consumption | Fout = 48 MHz, VDD =5.0V | Max 125°C<br>Typ 25°C | 452 | 504 | µA |

## 48.9. Timing Characteristics

### 48.9.1. SERCOM in SPI Mode Timing

**Table 48-34.** SPI Timing Characteristics and Requirements (1)

| Symbol | Parameter | Conditions | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| $t_{SCK}$ (10) | SCK period | Host | Reception | $2*(t_{MIS}+t_{CLIENT_OUT})$ (3) | - | - | ns |
| | | Host | Transmission | $2*(t_{MOV}+t_{CLIENT_IN})$ (4) | - | - | |
| $t_{SCKW}$ | SCK high/low width | Host | | - | $0.5*t_{SCK}$ | - | ns |
| $t_{SCKR}$ | SCK rise time (2) | Host | | - | $0.25*t_{SCK}$ | - | ns |
| $t_{SCKF}$ | SCK fall time (2) | Host | | - | $0.25*t_{SCK}$ | - | ns |
| $t_{MIS}$ | MISO setup to SCK | Host, VDD>4.5V | | 51.7 | - | - | ns |
| | | Host, VDD>2.7V | | 61.6 | - | - | |
| $t_{MIH}$ | MISO hold after SCK | Host, VDD>4.5V | | 0 | - | - | ns |
| | | Host, VDD>2.7V | | 0 | - | - | |
| $t_{MOV}$ | MOSI output valid SCK | Host, VDD>4.5V | | - | - | 18.1 | ns |
| | | Host, VDD>2.7V | | - | - | 24.6 | |
| $t_{MOH}$ | MOSI hold after SCK | Host, VDD>4.5V | | 2.5 | - | - | ns |
| | | Host, VDD>2.7V | | 2.5 | - | - | |
| $t_{SSCK}$ | Client SCK Period | Client | Reception | $2*(t_{SIS}+t_{HOST_OUT})$ (5) | - | - | ns |
| | | Client | Transmission | $2*(t_{SOV}+t_{HOST_IN})$ (6) | - | - | |
| $t_{SSCKW}$ | SCK high/low width | Client | | - | $0.5*t_{SSCK}$ | - | ns |
| $t_{SSCKR}$ | SCK rise time (2) | Client | | - | $0.25*t_{SSCK}$ | - | ns |
| $t_{SSCKF}$ | SCK fall time (2) | Client | | - | $0.25*t_{SSCK}$ | - | ns |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 48-34.** SPI Timing Characteristics and Requirements [1] (continued)

| Symbol | Parameter | Conditions | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| $t_{SIS}$ | MOSI setup to SCK | Client, VDD>4.5V | | 14.6 | - | - | ns |
| | | Client, VDD>2.7V | | 15.1 | - | - | |
| $t_{SIH}$ | MOSI hold after SCK | Client, VDD>4.5V | | 0 | - | - | ns |
| | | Client, VDD>2.7V | | 0 | - | - | |
| $t_{SSS}$ | $\overline{SS}$ setup to SCK | Client | PRELOADEN=1 | $t_{SOSS}+t_{EXT_MIS}+2*t_{APBC}$ [8] [9] | - | - | ns |
| | | | PRELOADEN=0 | $t_{SOSS}+t_{EXT_MIS}$ [8] | - | - | |
| $t_{SSH}$ | $\overline{SS}$ hold after SCK | Client | | $0.5*t_{SSCK}$ | - | - | ns |
| $t_{SOV}$ | MISO output valid SCK | Client, VDD>4.5V | | - | - | 46 | ns |
| | | Client, VDD>2.7V | | - | - | 56.1 | |
| $t_{SOH}$ | MISO hold after SCK | Client, VDD>4.5V | | 11.9 | - | - | ns |
| | | Client, VDD>2.7V | | 11.9 | - | - | |
| $t_{SOSS}$ | MISO setup after $\overline{SS}$ low | Client, VDD>4.5V | | - | - | 42 | ns |
| | | Client, VDD>2.7V | | - | - | 51.7 | |
| $t_{SOSH}$ | MISO hold after $\overline{SS}$ high | Client, VDD>4.5V | | 11.1 | - | - | ns |
| | | Client, VDD>2.7V | | 11.1 | - | - | |

1. These values are based on simulation. These values are not covered by test limits in production.
2. See I/O pin characteristics.
3. Where $t_{CLIENT_OUT}$ is the Client external device output response time, generally $t_{EXT_SOV}+t_{LINE_DELAY}$ [7].
4. Where $t_{CLIENT_IN}$ is the Client external device input constraint, generally $t_{EXT_SIS}+t_{LINE_DELAY}$ [7].
5. Where $t_{HOST_OUT}$ is the Host external device output response time, generally $t_{EXT_MOV}+t_{LINE_DELAY}$ [7].
6. Where $t_{HOST_IN}$ is the Host external device input constraint, generally $t_{EXT_MIS}+t_{LINE_DELAY}$ [7].
7. $t_{LINE_DELAY}$ is the transmission line time delay.
8. $t_{EXT_MIS}$ is the input constraint for the Host external device.
9. $t_{APBC}$ is the APB period for SERCOM.
10. When the integrity of communication is required to maintain both transmission and reception, the maximum SPI clock frequency should be the lower value of the reception or transmission mode maximum frequency as shown in the following equations.
    - Reception: $t_{SCK} = 2*(t_{MIS}+t_{CLIENT_OUT})$
    - Transmission: $t_{SCK} = 2*(t_{MOV}+t_{CLIENT_IN})$

### 48.9.2. External Reset

**Table 48-35.** External Reset Characteristics[1]

| Symbol | Parameter | Min. | Units |
|---|---|---|---|
| $t_{EXT}$ | Minimum reset pulse width | 1.1 | µs |

1. These are based on simulation. These values are not covered by test or characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 48.9.3. CAN Timing

**Table 48-36.** CAN Physical Layer Timing[1][2]

| Parameter | Conditions | Max. | Units |
|---|---|---|---|
| TxCAN output delay | VDD = 2.7V<br>Load = 20pF<br>VOL/VOH = VDD/2 | 29.4 | ns |
| | VDD = 4.5V<br>Load = 20pF<br>VOL/VOH = VDD/2 | 20.6 | |

1. These values are based on simulation. These values are not covered by test limits in production.
2. These values are obtained with Output Driver Strength Selection as DRVSTR = 1.


© 2026 Microchip Technology Inc. and its subsidiaries

# 49. Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 N)

**Note:** TSENS is not available in AEC - Q100 qualified device part numbers.

## 49.1. Disclaimer

All typical values are measured at Ta = 25°C unless otherwise specified. All minimum and maximum values are valid across operating temperature and voltage unless otherwise specified.

This chapter only contains characteristics specific for SAM C20/C21 N AEC - Q100 (Ta = 125°C). For all other values or missing characteristics, refer first to the SAM C20/C21 E/G/J AEC - Q100 chapter and then to the SAM C20/C21 E/G/J 85°C chapter.

## 49.2. Power Consumption

The values in the Power Consumption table below are measured values of power consumption under the following conditions, except where noted:

- **Operating Conditions:**
  - VDDIN = 3.0 V, 5.0V
- **Oscillators:**
  - XOSC (crystal oscillator) stopped
  - XOSC32K (32.768 kHz crystal oscillator) running with external 32.768 kHz crystal
  - FDPLL using XOSC32K as reference and running at 48MHz
- **Clocks:**
  - FDPLL used as main clock source, except otherwise specified.
  - CPU, AHB clocks undivided
  - All peripheral clocks stopped
- I/Os are inactive with input trigger disable
- CPU is running on Flash with required Wait states as recommended in the NVM Max Speed Characteristics section
- NVMCTRL cache enabled
- BODVDD disabled


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 49-1.** Current Consumption [1]

| Mode | Conditions | Ta | Vcc | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| ACTIVE | CPU running a While 1 algorithm | 25°C | 5.0V | 3.8 | 4.2 | mA |
| | | 125°C | 5.0V | 4.4 | 6.9 | |
| | CPU running a While 1 algorithm | 25°C | 3.0V | 3.7 | 4.2 | mA |
| | | 125°C | 3.0V | 4.4 | 6.9 | |
| | CPU running a While 1 algorithm, with GCLKIN as reference | 25°C | 5.0V | 71*Freq+160 | 76*Freq+180 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 72*Freq+659 | 67*Freq+3474 | |
| | CPU running a Fibonacci algorithm | 25°C | 5.0V | 4.7 | 5.2 | mA |
| | | 125°C | 5.0V | 5.3 | 8.0 | |
| | CPU running a Fibonacci algorithm | 25°C | 3.0V | 4.7 | 5.2 | mA |
| | | 125°C | 3.0V | 5.3 | 7.9 | |
| | CPU running a Fibonacci algorithm, with GCLKIN as reference | 25°C | 5.0V | 90*Freq+163 | 94*Freq+208 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 91*Freq+664 | 89*Freq+3477 | |
| | CPU running a CoreMark algorithm | 25°C | 5.0V | 5.9 | 6.5 | mA |
| | | 125°C | 5.0V | 6.7 | 9.0 | |
| | CPU running a CoreMark algorithm | 25°C | 3.0V | 5.2 | 5.7 | mA |
| | | 125°C | 3.0V | 5.9 | 8.5 | |
| | CPU running a CoreMark algorithm, with GCLKIN as reference | 25°C | 5.0V | 115*Freq+167 | 114*Freq+206 | µA (with freq in MHz) |
| | | 125°C | 5.0V | 119*Freq+668 | 110*Freq+3484 | |
| IDLE0 | | 25°C | 5.0V | 1.8 | 2.6 | mA |
| | | 125°C | 5.0V | 2.3 | 5.1 | |
| IDLE2 | | 25°C | 5.0V | 1.2 | 1.8 | mA |
| | | 125°C | 5.0V | 1.8 | 4.6 | |
| STANDBY | XOSC32K running RTC running at 1.024 kHz | 25°C | 5.0V | 15.9 | 37 | µA |
| | | 125°C | 5.0V | 363.4 | 1103 | |
| | XOSC32K and RTC stopped | 25°C | 5.0V | 14.6 | 35 | |
| | | 125°C | 5.0V | 361.4 | 1100 | |

**Note:**

1. These are based on characterization.

## 49.3. Analog Characteristics

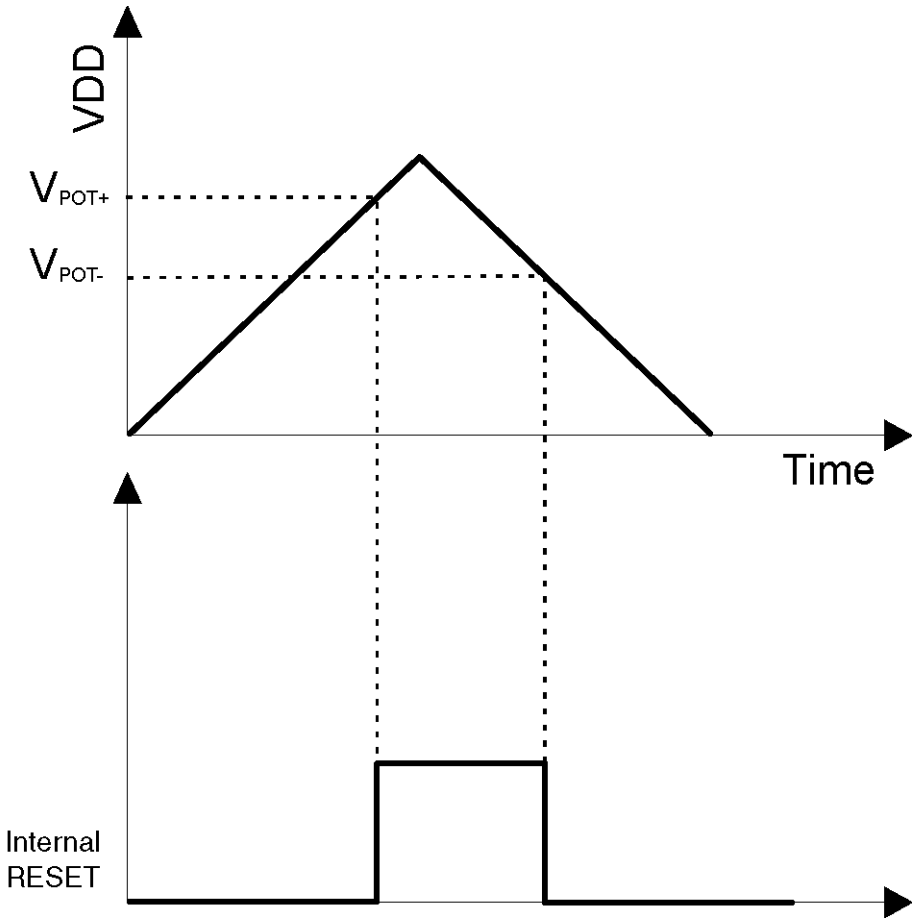
### 49.3.1. Power-on Reset (POR) Characteristics

**Table 49-2.** POR Characteristics

| Symbol | Parameters | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Vpot+ | Voltage threshold Level on Vddin rising | - | 2.55 | - | V |
| Vpot- | Voltage threshold Level on Vddin falling | 1.75 | 1.92 | 2.07 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 49-1.** POR Operating Principle




### 49.3.2. BODVDD - Brown Out Detector Characteristics

See NVM User Row Mapping for the BODVDD default value settings. These values are based on simulation and are not covered by test limits in production or characterization.

**Figure 49-2.** BODVDD Hysteresis Off

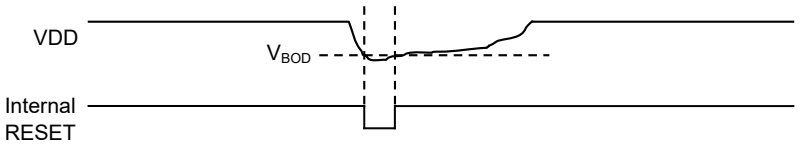
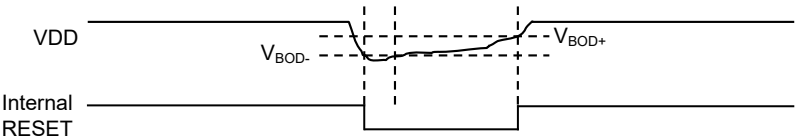



**Figure 49-3.** BODVDD Hysteresis On





© 2026 Microchip Technology Inc. and its subsidiaries

**Table 49-3.** BODVDD Characteristics[2]

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| VBOD+[1] | BODVDD high threshold Level | VDD level, BOD setting = 8 (default) | - | 2.86 | 3.00 | V |
| | | VDD level, Bod setting = 44 | - | 4.57 | 4.86 | |
| VBOD- / VBOD[1] | BODVDD low threshold Level | VDD level, BOD setting = 8 (default) | 2.69 | 2.8 | 2.92 | |
| | | VDD level, BOD setting = 44 | 4.35 | 4.51 | 4.70 | |
| | Step size | | - | 60 | - | mV |
| VHys[1] | Hysteresis (VBOD+ - VBOD-) BODVDD.LEVEL = 8 to 48 | VDD | 40 | - | 75 | mV |
| $T_{START}$ | Startup time | Time from enable to RDY | - | 8 | - | µs |

**Notes:**

1. These are based on characterization.
2. BODVDD in continuous mode.

### 49.3.3. Analog-to-Digital Converter (ADC) Characteristics

**Table 49-4.** Differential Mode [1]

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| ENOB | Effective Number of bits | Fadc = 500ksps - R2R disabled | Vddana=5.0V Vref=Vddana | 9.9 | 10.7 | 11.4 | bits |
| | | | Vddana=2.7V Vref=2.0V | 10.0 | 10.8 | 11.3 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | 9.7 | 10.6 | 11.3 | |
| | | | Vddana=2.7V Vref=2.0V | 9.8 | 10.6 | 11.2 | |
| | | Fadc = 500ksps - R2R Enabled | Vddana=5.0V Vref=Vddana | 9.8 | 11.3 | 11.9 | |
| | | Fadc = 1Msps - R2R Enabled | Vddana=5.0V Vref=Vddana | 9.7 | 11.1 | 11.8 | |
| TUE | Total Unadjusted Error | Fadc = 500ksps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana=5.0V Vref=Vddana | - | +/-4.6 | +/-8.3 | LSB |
| | | | Vddana=2.7V Vref=2.0V | - | +/-4.3 | +/-8.3 | |
| | | Fadc = 1Msps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana=5.0V Vref=Vddana | - | +/-4.2 | +/-7.1 | |
| | | | Vddana=2.7V Vref=2.0V | - | +/-4.8 | +/-8.3 | |
| INL | Integral Non Linearity | Fadc = 500ksps - R2R disabled | Vddana=5.0V Vref=Vddana | - | +/-1.6 | +/-3.5 | LSB |
| | | | Vddana=2.7V Vref=2.0V | - | +/-1.9 | +/-3.9 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | - | +/-1.5 | +/-4 | |
| | | | Vddana=2.7V Vref=2.0V | - | +/-3.2 | +/-4.1 | |
| DNL | Differential Non Linearity | Fadc = 500ksps - R2R disabled | Vddana=5.0V Vref=Vddana | - | -0.8/+1 | -1/+2.0 | LSB |
| | | | Vddana=2.7V Vref=2.0V | - | -0.7/+1.3 | -1/+2.2 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | - | -0.8/+1.1 | -1/+2.0 | |
| | | | Vddana=2.7V Vref=2.0V | - | -0.9/+1.6 | -1/+3.7 | |
| GE | Gain Error | Fadc = 1Msps - R2R disabled w/o gain compensation | Vddana=5.0V Vref=Vddana | - | +/-0.8 | +/-1.8 | % |
| | | | Vddana=2.7V Vref=2.0V | - | +/-0.9 | +/-3.2 | |
| | | | Vddana=5.0V 1V internal Ref | - | +/-1.9 | +/-7.3 | |
| | | | Vddana=5.0V Vref=Vddana/2 | - | +/-0.1 | +/-1.3 | |
| | | Fadc = 1Msps - R2R disabled with gain compensation | Vddana=2.7V Vref=2.0V | - | +/-0.1 | +/-0.5 | |
| | | | Vddana=5.0V Vref=Vddana/2 | - | +/-0.2 | +/-0.7 | |
| OE | Offset Error | Fadc = 1Msps - R2R disabled without offset compensation | Vddana=5.0V Vref=Vddana/2 | - | +/-0.2 | +/-25 | mV |
| | | | Vddana=2.7V Vref=2.0V | - | +/-1.4 | +/-17 | |
| | | Fadc = 1Msps - R2R disabled with offset compensation | Vddana=5.0V Vref=Vddana/2 | - | +/-2 | +/-10 | |
| | | | Vddana=2.7V Vref=2.0V | - | +/-0.2 | +/-11 | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 49-4.** Differential Mode [1] (continued)

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana=5.0V Vref=Vddana | 63 | 75 | 81 | dB |
| SINAD | | Signal to Noise and Distortion ratio | | 60 | 67 | 70 | |
| SNR | | Signal to Noise ratio | | 64 | 68 | 70 | |
| THD | | Total Harmonic Distortion | | -81 | -74 | -63 | |
| | | Noise RMS | External Reference voltage | - | 0.5 | 3.3 | mV |

**Note:**

1. These values are based on characterization. These values are not covered by test limits in production.

**Table 49-5.** Single Ended Mode [1]

| Symbol | Parameter | Conditions | | Measurement | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| ENOB | Effective Number of bits | Fadc = 500ksps - R2R disabled | Vddana=3.0V Vref=Vddana | 9.0 | 9.7 | 10.2 | bits |
| | | | Vddana=3.0V Vref=2.0V | 9.0 | 9.4 | 10.1 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=3.0V Vref=Vddana | 8.9 | 9.7 | 10.0 | |
| | | | Vddana=3.0V Vref=2.0V | 8.9 | 9.2 | 9.7 | |
| TUE | Total Unadjusted Error | Fadc = 500ksps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana=5.0V Vref=Vddana | - | 18 | +/-28 | LSB |
| | | | Vddana=2.7V Vref=2.0V | - | 30 | +/-55 | |
| | | Fadc = 1Msps - R2R disabled with offset and gain compensation (REFCOMP = 1 and OFFCOMP = 1) | Vddana=5.0V Vref=Vddana | - | 18 | +/-28 | |
| | | | Vddana=2.7V Vref=2.0V | - | 30 | +/-57 | |
| INL | Integral Non Linearity | Fadc = 500ksps - R2R disabled | Vddana=5.0V Vref=Vddana | - | +/-2.4 | +/-6.5 | |
| | | | Vddana=2.7V Vref=2.0V | - | +/-3.7 | +/-7.1 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | - | +/-2.2 | +/-7.5 | |
| | | | Vddana=2.7V Vref=2.0V | - | +/-4.1 | +/-7.1 | |
| DNL | Differential Non Linearity | Fadc = 500ksps - R2R disabled | Vddana=5.0V Vref=Vddana | - | -0.8/+1.1 | -1/+1.9 | LSB |
| | | | Vddana=2.7V Vref=2.0V | - | -0.8/+1.1 | -1/+2.5 | |
| | | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | - | -0.8/+1 | -1/+1.9 | |
| | | | Vddana=2.7V Vref=2.0V | - | -1/+1.4 | -1/+3.1 | |
| GE | Gain Error | Fadc = 1Msps - R2R disabled w/o gain compensation | Vddana=5.0V Vref=Vddana | - | +/-0.4 | +/-1.1 | % |
| | | | Vddana=2.7V Vref=2.0V | - | +/-0.7 | +/-1.9 | |
| | | | Vddana=5.0V 1V internal Ref | - | +/-1.6 | +/-7.4 | |
| | | | Vddana=5.0V Vref=Vddana/2 | - | +/-0.2 | +/-1.4 | |
| | | Fadc = 1Msps - R2R disabled with gain compensation | Vddana=2.7V Vref=2.0V | - | +/-0.3 | +/-0.9 | |
| | | | Vddana=5.0V Vref=Vddana/2 | - | +/-0.1 | +/-1.4 | |
| OE | Offset Error | Fadc = 1Msps - R2R disabled | Vddana=5.0V Vref=Vddana | - | +/-31 | +/-76 | mV |
| | | | Vddana=2.7V Vref=2.0V | - | +/-2.2 | +/-64 | |
| SFDR | | Spurious Free Dynamic Range | Fs = 1Msps / Fin = 14 kHz / Full range Input signal Vddana=5.0V Vref=Vddana | 57 | 71 | 73 | dB |
| SINAD | | Signal to Noise and Distortion ratio | | 54 | 60 | 62 | |
| SNR | | Signal to Noise ratio | | 57 | 61 | 62 | |
| THD | | Total Harmonic Distortion | | -71 | -70 | -56 | |
| | | Noise RMS | External Reference voltage | - | 0.7 | 2 | mV |

**Note:**

1. These values are based on characterization. These values are not covered by test limits in production.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 49-6.** Power Consumption (1)

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD VDDANA | Differential mode | fs = 1 Msps / Reference buffer disabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | Max 125°C Typ 25°C | 905 | 1091 | uA |
| | | fs = 1 Msps / Reference buffer enabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 1062 | 1310 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 381 | 510 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 525 | 751 | |
| | Single Ended mode | fs = 1 Msps / Reference buffer disabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref=5.5V | Max 125°C Typ 25°C | 984 | 1143 | uA |
| | | fs = 1 Msps / Reference buffer enabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref=5.5V | | 1103 | 1357 | |
| | | fs = 10 ksps / Reference buffer disabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 437 | 585 | |
| | | fs = 10 ksps / Reference buffer enabled / BIASREFBUF = '111', BIASREFCOMP = '111' VDDANA=Vref= 5.5V | | 553 | 786 | |

**Note:**

1. These are based on characterization.

## 49.3.4. Analog Comparator Characteristics

**Table 49-7.** Analog Comparator Characteristics

| Symbol | Parameters | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| PNIVR | Positive and Negative input range voltage | | 0 | - | VDDANA | V |
| ICMR | Input common mode range | Hysteresis Disable COMPCTRLn.HYSTEN = 0x0 | 0 | - | VDDANA | V |
| Off (1)(2) | Offset | Low power COMPCTRLn.SPEED = 0x0 | -66 | +/-4 | 66 | mV |
| | | High speed COMPCTRLn.SPEED = 0x3 | -27 | +/-2 | 27 | |
| $T_{PD}$ (1, 3) | Propagation Delay Vcm=Vddana/2Vin = ±100mV overdrive from Vcm | Low power COMPCTRLn.SPEED = 0x0 | - | 149 | 290 | ns |
| | | High speed COMPCTRLn.SPEED = 0x3 | - | 41 | 78 | |
| $T_{START}$ (1) | Startup time | Low power COMPCTRLn.SPEED = 0x0 | - | 6.8 | 11.2 | µs |
| | | High speed COMPCTRLn.SPEED = 0x3 | - | 2.2 | 4 | |
| $V_{SCALE}$ (1) | INL | | - | 0.2 | - | LSB |
| | DNL | | - | 0.03 | - | |
| | Offset Error | | - | 0.050 | - | |
| | Gain Error | | - | 0.07 | - | |

**Notes:**

1. These are based on characterization.
2. Hysteresis disabled.
3. Tpd is measured from Vin transition to ACOUT (AC direct output) toggle. It takes into account only analog propagation delay.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 49-8.** Power Consumption [1]

| Symbol | Parameters | Conditions | Ta | Typ. | Max | Units |
|---|---|---|---|---|---|---|
| IDD ANA | Voltage scaler disabled | COMPCTRLn.SPEED = 0x0, VDDANA =5.0V | Max 125°C Typ 25°C | 10 | 20 | µA |
| | | COMPCTRLn.SPEED = 0x3, VDDANA =5.0V | | 39 | 66 | |
| | Voltage scaler only | VDDANA =5.0V | | 43 | 68 | |

**Note:**

1. These are based on characterization.

## 49.3.5. PTC Characteristics

**Table 49-9.** Sensor Load Capacitance [1]

| Symbol | Mode | PTC channel | Max. | Units |
|---|---|---|---|---|
| Cload | Self-capacitance | Y0 | 16 | pF |
| Cload | Self-capacitance | Y1 | 24 | pF |
| | | Y2 | | |
| | | Y3 | | |
| | | Y4 | | |
| | | Y5 | | |
| | | Y6 | | |
| | | Y7 | | |
| | | Y8 | | |
| | | Y9 | | |
| | | Y10 | | |
| | | Y11 | | |
| | | Y12 | | |
| | | Y13 | | |
| | | Y14 | | |
| | | Y15 | | |
| | | Y16 | | |
| | | Y17 | | |
| | | Y18 | | |
| | | Y19 | | |
| | | Y20 | | |
| | | Y21 | | |
| | | Y22 | | |
| Cload | Self-capacitance | Y23 | 22 | pF |
| | | Y24 | | |
| Cload | Self-capacitance | Y25 | 24 | pF |
| | | Y26 | | |
| | | Y27 | | |
| | | Y28 | | |
| | | Y29 | | |
| | | Y30 | | |
| | | Y31 | | |
| Cload | Mutual-capacitance | All | 31 | pF |


© 2026 Microchip Technology Inc. and its subsidiaries

**Note:**

1. Capacitance load that the PTC circuitry can compensate for each channel.

#### 49.3.5.1. PTC Power Consumption

The values in the Power Consumption table below are measured values of power consumption under the following conditions:

- **Operating Conditions:**
  - VDD = 5.0V
- **Clocks:**
  - OSC48M used as main clock source, running undivided at 48 MHz
  - CPU is running on Flash with 2 wait states at 48 MHz
  - PTC running at 4 MHz
- **PTC Configuration:**
  - Mutual-Capacitance mode
  - One-touch channel
- **System Configuration:**
  - Standby Sleep mode enabled
  - RTC running on ULP32K: used to define the PTC scan rate through the event system
  - Drift Calibration disabled: no interrupts, PTC scans are performed in Standby mode
  - Drift Calibration enabled: RTC interrupts (wake up) the CPU to perform PTC scans. The PTC drift calibration is performed every 1.5 sec.

**Table 49-10.** Power Consumption (1)

| Symbol | Parameters | Drift Calibration | PTC Scan Rate | Oversamples | Ta | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|---|
| IDD | Current Consumption | Disabled | 10 | 4 | Max. 125°C Typ 25°C | 24 | 1112 | µA |
| | | | | 16 | | 43 | 1136 | |
| | | | 50 | 4 | | 17 | 1105 | |
| | | | | 16 | | 22 | 1111 | |
| | | | 100 | 4 | | 16 | 1104 | |
| | | | | 16 | | 19 | 1108 | |
| | | | 200 | 4 | | 16 | 1104 | |
| | | | | 16 | | 17 | 1106 | |
| | | Enabled | 10 | 4 | | 30 | 1123 | |
| | | | | 16 | | 50 | 1147 | |
| | | | 50 | 4 | | 20 | 1110 | |
| | | | | 16 | | 24 | 1115 | |
| | | | 100 | 4 | | 18 | 1109 | |
| | | | | 16 | | 21 | 1111 | |
| | | | 200 | 4 | | 17 | 1108 | |
| | | | | 16 | | 19 | 1110 | |

**Note:**

1. These values are based on characterization.

## 49.4. Oscillator Characteristics

### 49.4.1. Crystal Oscillator Characteristics

The following table describes the characteristics for the oscillator when a crystal is connected between XIN and XOUT as shown in Figure 45-5. The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of $C_L$ can be


© 2026 Microchip Technology Inc. and its subsidiaries

found in the crystal data sheet. The capacitance of the external capacitors ($C_{LEXT}$) can be calculated as follows:

$C_{LOAD} = \{ ( [C_{XIN} + C_{LEXT}] * [C_{XOUT} + C_{LEXT}] ) / [C_{XIN} + C_{LEXT} + C_{LEXT} + C_{XOUT}] \} + C_{STRAY}$.

Where,

CSTRAY is the capacitance of the PCB.

For $C_{XIN} \neq C_{XOUT}$, but with a difference smaller than 4 pF, we can consider $C_{XIN} \approx C_{XOUT} \approx C_{Xeff} = ( C_{XIN} + C_{XOUT}) / 2$.

This results in a simplified formula for computing $C_{LEXT}$, as follows:

$C_{LEXT} = 2*CL - 2*C_{STRAY} - C_{Xeff}$

**Figure 49-4.** Oscillator Connection

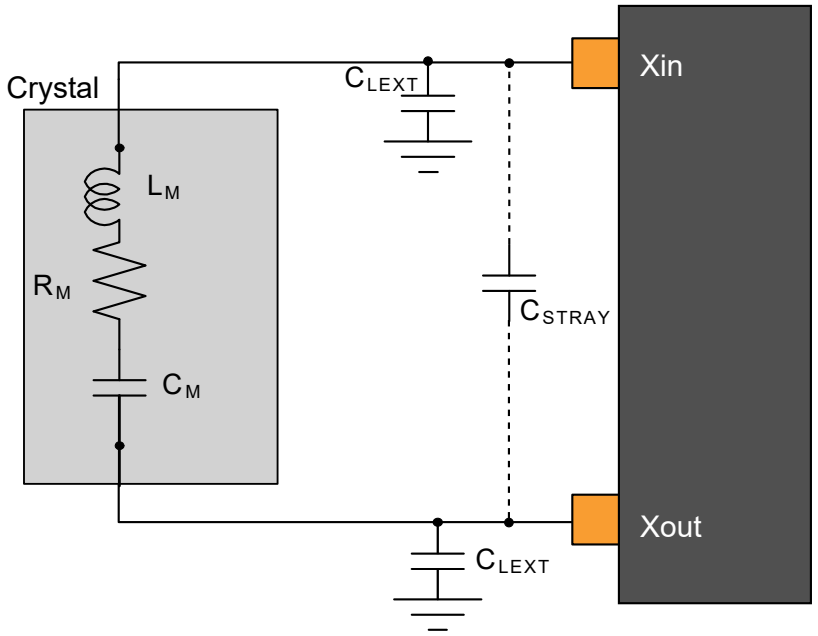



**Table 49-11.** Multi-Crystal Oscillator Electrical Characteristics [1]

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Cxin | Parasitic load capacitor | | - | 6.7 | - | pF |
| Cxout | | | - | 4.1 | - | |

**Note:**

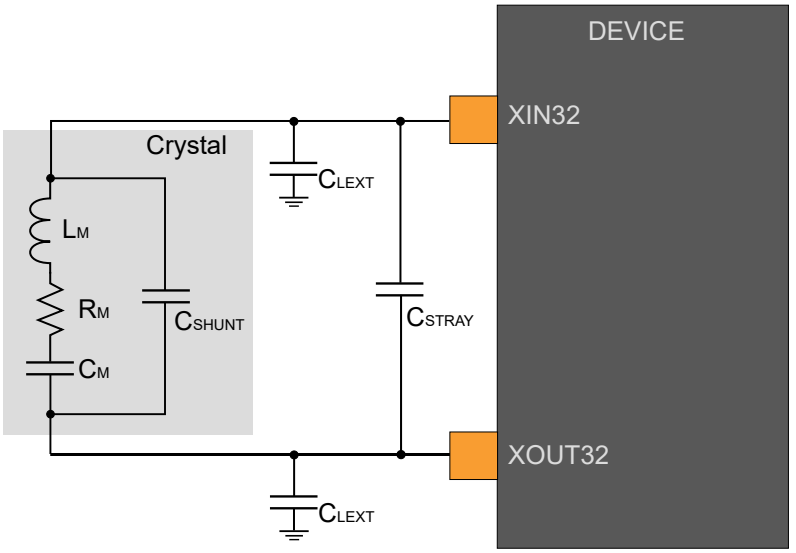
1. These values are based on characterization.

### 49.4.2. Crystal Oscillator Characteristics (XOSC32K)

The following table describes the characteristics for the oscillator when a crystal is connected between XIN32 and XOUT32.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 49-5.** Oscillator Connection




The user must choose a crystal oscillator where the crystal load capacitance $C_L$ is within the range given in the table. The exact value of CL can be found in the “Crystal Data Sheet”. The capacitance of the external capacitors (CLEXT) can be calculated as follows:

$C_{LEXT}=2\ (C_L-C_{STRAY}-C_{SHUNT})$

where,

$C_{STRAY}$ is the capacitance of the pins, and PCB and $C_{SHUNT}$ is the shunt capacitance of the crystal.

**Table 49-12.** 32.768 kHz Crystal Oscillator Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Cxin32k | Parasitic capacitor load | - | - | 3.8 | - | pF |
| Cxout32k | | | - | 4.1 | - | pF |

### 49.4.3. 32.768 kHz Internal Oscillator (OSC32K) Characteristics

**Table 49-13.** 32.768 kHz RC Oscillator Electrical Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output frequency | at 25°C , at VDDANA = 5.0V | 30.965 | 32.768 | 34.57 | kHz |
| | | at 25°C , over [2.7, 5.5]V | 28.835 | 32.768 | 36.7 | kHz |
| | | over[-40,125]°C, over [2.7, 5.5]V | 24.282 | 32.768 | 38.01 | kHz |
| Tstartup | Startup time | - | - | 1 | 2 | cycles |
| Duty (1) | Duty Cycle | - | - | 50 | - | % |

**Note:**

1. These values are based on simulation, and are not covered by test or characterization.


© 2026 Microchip Technology Inc. and its subsidiaries

### 49.4.4. Ultra Low-Power Internal 32.768 kHz RC Oscillator (OSCULP32K) Characteristics

**Table 49-14.** Ultra Low-Power Internal 32.768 kHz RC Oscillator (OSCULP32K) Characteristics

| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|
| Fout | Output frequency | at 25°C , at VDDANA = 5.0V | 30.474 | 32.768 | 34.57 | kHz |
| | | at 25°C , over [2.7, 5.5]V | 30.474 | 32.768 | 35.062 | kHz |
| | | over[-40,125]°C, over [2.7, 5.5]V | 22.937 | 32.768 | 40.960 | kHz |
| Duty | Duty Cycle | | - | 50 | - | % |

## 49.5. Timing Characteristics

### 49.5.1. SERCOM in SPI Mode Timing Characteristics

**Table 49-15.** SPI Timing Characteristics and Requirements (1)

| Symbol | Parameter | Conditions | | Min. | Typ. | Max. | Units |
|---|---|---|---|---|---|---|---|
| tSCK | SCK period | Host | Reception | 2*(tMIS+tCLIENT_OUT) (3) | - | - | ns |
| | | Host | Transmission | 2*(tMOV+tCLIENT_IN) (4) | - | - | |
| tSCKW | SCK high/low width | Host | | - | 0.5*tSCK | - | |
| tSCKR | SCK rise time (2) | Host | | - | 0.25*tSCK | - | |
| tSCKF | SCK fall time (2) | Host | | - | 0.25*tSCK | - | |
| tMIS | MISO setup to SCK | Host. VDD>4.5V | | 62.19 | - | - | |
| | | Host. VDD>2.7V | | 72.09 | - | - | |
| tMIH | MISO hold after SCK | Host. VDD>4.5V | | 0 | - | - | |
| | | Host. VDD>2.7V | | 0 | - | - | |
| tMOV | MOSI output valid SCK | Host. VDD>4.5V | | - | - | 18.1 | |
| | | Host. VDD>2.7V | | - | - | 24.6 | |
| tMOH | MOSI hold after SCK | Host. VDD>4.5V | | 5.78 | - | - | |
| | | Host. VDD>2.7V | | 5.78 | - | - | |
| tSSCK | Client SCK Period | Client | Reception | 2*(tSIS+tHOST_OUT) (5) | - | - | |
| | | Client | Transmission | 2*(tSOV+tHOST_IN) (6) | - | - | |
| tSSCKW | SCK high/low width | Client | | - | 0.5*tSSCK | - | |
| tSSCKR | SCK rise time (2) | Client | | - | 0.25*tSSCK | - | |
| tSSCKF | SCK fall time (2) | Client | | - | 0.25*tSSCK | - | ns |
| tSIS | MOSI setup to SCK | Client. VDD>4.5V | | 14.77 | - | - | |
| | | Client. VDD>2.7V | | 15.27 | - | - | |
| tSIH | MOSI hold after SCK | Client. VDD>4.5V | | 0 | - | - | |
| | | Client. VDD>2.7V | | 0 | - | - | |
| tSSS | $\overline{SS}$ setup to SCK | Client | PRELOADEN=1 | tSOSS+tEXT_MIS+2*tAPBC (8) (9) | - | - | |
| | | | PRELOADEN=0 | tSOSS+tEXT_MIS (8) | - | - | |
| tSSH | $\overline{SS}$ hold after SCK | Client | | 0.5*tSSCK | - | - | |
| tSOV | MISO output valid SCK | Client. VDD>4.5V | | - | - | 51.56 | |
| | | Client. VDD>2.7V | | - | - | 61.66 | |
| tSOH | MISO hold after SCK | Client. VDD>4.5V | | 14.66 | - | - | |
| | | Client. VDD>2.7V | | 14.66 | - | - | |
| tSOSS | MISO setup after $\overline{SS}$ low | Client. VDD>4.5V | | - | - | 47.25 | |
| | | Client. VDD>2.7V | | - | - | 56.95 | |
| tSOSH | MISO hold after $\overline{SS}$ high | Client. VDD>4.5V | | 8.57 | - | - | |
| | | Client. VDD>2.7V | | 8.57 | - | - | |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. These values are based on simulation and are not covered by test limits in production.
2. Refer to the section “I/O pin characteristics”.
3. Where tCLIENT_OUT is the Client external device output response time, generally tEXT_SOV+tLINE_DELAY (7).
4. Where tCLIENT_IN is the Client external device input constraint, generally tEXT_SIS+tLINE_DELAY (7).
5. Where tHOST_OUT is the Host external device output response time, generally tEXT_MOV+tLINE_DELAY (7).
6. Where tHOST_IN is the Host external device input constraint, generally tEXT_MIS+tLINE_DELAY (7).
7. tLINE_DELAY is the transmission line time delay.
8. tEXT_MIS is the input constraint for the Host external device.
9. tAPBC is the APB period for SERCOM.
10. When the integrity of communication is required to maintain both transmission and reception, the maximum SPI clock frequency must be the lower value of the reception or transmission mode maximum frequency as shown in the following equations:
    - Reception: tSCK = 2*(tMIS+tCLIENT_OUT)
    - Transmission: tSCK = 2*(tMOV+tCLIENT_IN)

### 49.5.2. External Reset

**Table 49-16.** External Reset Characteristics (1)

| Symbol | Parameter | Min. | Units |
|---|---|---|---|
| tEXT | Minimum reset pulse width | 1 | µs |

**Note:**

1. These values are based on simulation, and are not covered by test or characterization.

### 49.5.3. CAN Timing

**Table 49-17.** CAN Physical Layer Timing (1,2)

| Parameter | Conditions | Max. | Units |
|---|---|---|---|
| TxCAN output delay | VDD = 2.7 V Load = 20 pF VOL/VOH = VDD/2 | 34.3 | ns |
| | VDD = 4.5 V Load = 20 pF VOL/VOH = VDD/2 | 24.6 | |

**Notes:**

1. These values are based on simulation. These values are not covered by test limits in production.
2. These values are obtained with Output Driver Strength Selection at 1 (DRVSTR).


© 2026 Microchip Technology Inc. and its subsidiaries

# 50. Appendix A

## 50.1. ISELED FULL License Enabled Functional Devices

Microchip offers ISELED full license enabled devices, which can utilize the ISELED software stack and library available for download from the Microchip Website. Included in the download will be the ISELED stack source code file, header, the library binary, and the library user's manual. Refer to the Inova data sheet INLC10AQ (ISELED control commands; document AN-INLC_04). For more information on Microchip's ISELED solutions, please visit the Microchip Website (www.microchip.com/iseled).

Contact your Microchip Sales Office for more details on the ISELED full license enabled devices, or to request a part number which is not available in the following Ordering Information.

## 50.2. Ordering Information

The following table lists the ISELED full license enabled devices which can utilize the ISELED full license software stack and library.

**Table 50-1.** C21 ISELED Licensed Devices

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| **C21N:** | | | | | |
| ATSAMC21N18A-AZT510 | 256K | 32K | -40°C to 125°C | TQFP100 | Tape & Reel |
| ATSAMC21N18A-AZ510 | 256K | 32K | -40°C to 125°C | TQFP100 | Tray |
| **C21J:** | | | | | |
| ATSAMC21J18A-AZT510 | 256K | 32K | -40°C to 125°C | TQFP64 | Tape & Reel |
| ATSAMC21J18A-AZ510 | 256K | 32K | -40°C to 125°C | TQFP64 | Tray |
| ATSAMC21J17A-AZT510 | 128K | 16K | -40°C to 125°C | TQFP64 | Tape & Reel |
| ATSAMC21J17A-AZ510 | 128K | 16K | -40°C to 125°C | TQFP64 | Tray |
| **C21G:** | | | | | |
| ATSAMC21G17A-MZ510 | 128K | 16K | -40°C to 125°C | VQFN48 | Tray |
| ATSAMC21G18A-AZT510 | 256K | 32K | -40°C to 125°C | TQFP48 | Tape & Reel |
| ATSAMC21G18A-AZ510 | 256K | 32K | -40°C to 125°C | TQFP48 | Tray |
| **C21E** | | | | | |
| ATSAMC21E18A-AZT510 | 256K | 32K | -40°C to 125°C | TQFP32 | Tape & Reel |
| ATSAMC21E18A-AZ510 | 256K | 32K | -40°C to 125°C | TQFP32 | Tray |
| ATSAMC21E15A-MZT510 | 32K | 4K | -40°C to 125°C | VQFN32 | Tape & Reel |
| ATSAMC21E15A-MZ510 | 32K | 4K | -40°C to 125°C | VQFN32 | Tray |


© 2026 Microchip Technology Inc. and its subsidiaries

# 51. Appendix B

## 51.1. SIL 2-Enabled Functional Safety Devices

Microchip offers IEC 61508 SIL 2-enabled devices which can utilize the self-test library available on request from the Microchip sales office. The download includes the library binary, library user's manual, and user's checklist for integration of the library. Refer to the "Embex SIL 2 Library User's Manual" for additional information on using the IEC 61508 SIL 2-enabled Microchip devices.

Contact the Microchip sales office for additional information on the IEC 61508 SIL 2-enabled devices, or to request a part number which is not shown in the following Ordering Information.

## 51.2. Ordering Information

The following list displays the SIL 2-enabled devices, which can utilize the SIL 2 qualified self-test library (STL).

**Table 51-1.** SAM C21N

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| ATSAMC21N18A-ANT-SLL | 256K | 32K | -40°C to 105°C | TQFP100 | Tape & Reel |

**Table 51-2.** SAM C20N

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| ATSAMC20N18A-ANT-SLL | 256K | 32K | -40°C to 105°C | TQFP100 | Tape & Reel |

**Table 51-3.** SAM C21J

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| ATSAMC21J18A-ANT-SLL | 256K | 32K | -40°C to 105°C | TQFP64 | Tape & Reel |
| ATSAMC21J18A-MNT-SLL | 256K | 32K | -40°C to 105°C | VQFN64 | Tape & Reel |

**Table 51-4.** SAM C21G

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| ATSAMC21G18A-ANT-SLL | 256K | 32K | -40°C to 105°C | TQFP48 | Tape & Reel |
| ATSAMC21G18A-MNT-SLL | 256K | 32K | -40°C to 105°C | VQFN48 | Tape & Reel |

**Table 51-5.** SAM C21E

| Ordering Code | Flash (bytes) | SRAM (bytes) | Temperature Grade | Package | Carrier Type |
|---|---|---|---|---|---|
| ATSAMC21E18A-ANT-SLL | 256K | 32K | -40°C to 105°C | TQFP32 | Tape & Reel |
| ATSAMC21E18A-MNT-SLL | 256K | 32K | -40°C to 105°C | VQFN32 | Tape & Reel |


© 2026 Microchip Technology Inc. and its subsidiaries

# 52. Packaging Information

## 52.1. Package Marking Information

All devices are marked with the Atmel logo and the ordering code.

Where:

- "YY": Manufacturing year
- "WW": Manufacturing week
- "R": Internal Code
- "XXXXXXX": Lot number

**Figure 52-1.** C21 32-Pin TQFP




**Figure 52-2.** C20 32-pin TQFP





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 52-3.** C21 48-Pin TQFP




**Figure 52-4.** C20 48-Pin TQFP




**Figure 52-5.** C21 64-Pin TQFP





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 52-6.** C20 64-Pin TQFP




**Figure 52-7.** C21 100-Pin TQFP




**Figure 52-8.** C20 100-Pin TQFP





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 52-9.** C21 64-Pin VQFN

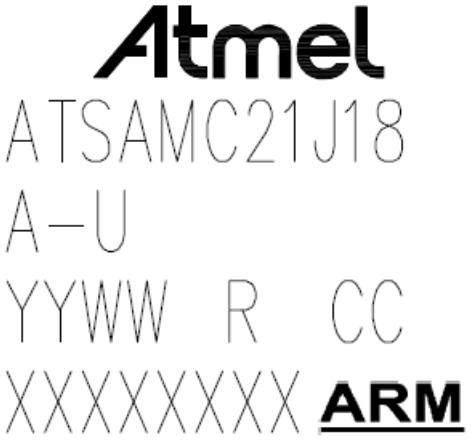



**Figure 52-10.** C20 64-Pin VQFN

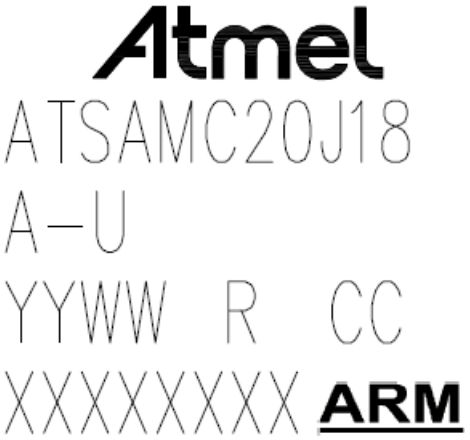



**Figure 52-11.** C21 48-Pin VQFN





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 52-12.** C20 48-Pin VQFN




**Figure 52-13.** C21 32-Pin VQFN




**Figure 52-14.** C20 32-Pin VQFN




## 52.2. Thermal Considerations

### 52.2.1. Thermal Resistance Data

The following table summarizes the thermal resistance data depending on the package.

**Table 52-1.** Thermal Resistance Data

| Package Type | $\theta_{JA}$ | $\theta_{JC}$ |
|---|---|---|
| 32-pin TQFP | 63.1°C/W | 14.3°C/W |
| 48-pin TQFP | 62.7°C/W | 11.6°C/W |
| 64-pin TQFP | 56.3°C/W | 11.1°C/W |
| 100-pin TQFP | 55.0°C/W | 11.1°C/W |
| 32-pin VQFN | 40.5°C/W | 16.0°C/W |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-1.** Thermal Resistance Data (continued)

| Package Type | $\theta_{JA}$ | $\theta_{JC}$ |
|---|---|---|
| 48-pin VQFN | 30.9°C/W | 10.4°C/W |
| 64-pin VQFN (TMB) | 31.4°C/W | 10.2°C/W |
| 64-Lead VQFN (5LX) | 23.9 °C/W | 8.9 °C/W |
| 56-ball WLCSP | 37.5°C/W | 5.48°C/W |

### 52.2.2. Junction Temperature

The average chip-junction temperature, $T_J$, in °C can be obtained from the following:

1. $T_J = T_A + (P_D \times \theta_{JA})$
2. $T_J = T_A + (P_D \times (\theta_{HEATSINK} + \theta_{JC}))$

where:

- $\theta_{JA}$ = Package thermal resistance, Junction-to-ambient (°C/W), see Thermal Resistance Data
- $\theta_{JC}$ = Package thermal resistance, Junction-to-case thermal resistance (°C/W), see Thermal Resistance Data
- $\theta_{HEATSINK}$ = Thermal resistance (°C/W) specification of the external cooling device
- $P_D$ = Device power consumption (W)
- $T_A$ = Ambient temperature (°C)

From the first equation, the user can derive the estimated lifetime of the chip and decide if a cooling device is necessary or not. If a cooling device has to be fitted on the chip, the second equation should be used to compute the resulting average chip-junction temperature $T_J$ in °C.

## 52.3. Package Drawings

**Note:** For current package drawings, refer to the Microchip Packaging Specification, which is available at http://www.microchip.com/packaging.

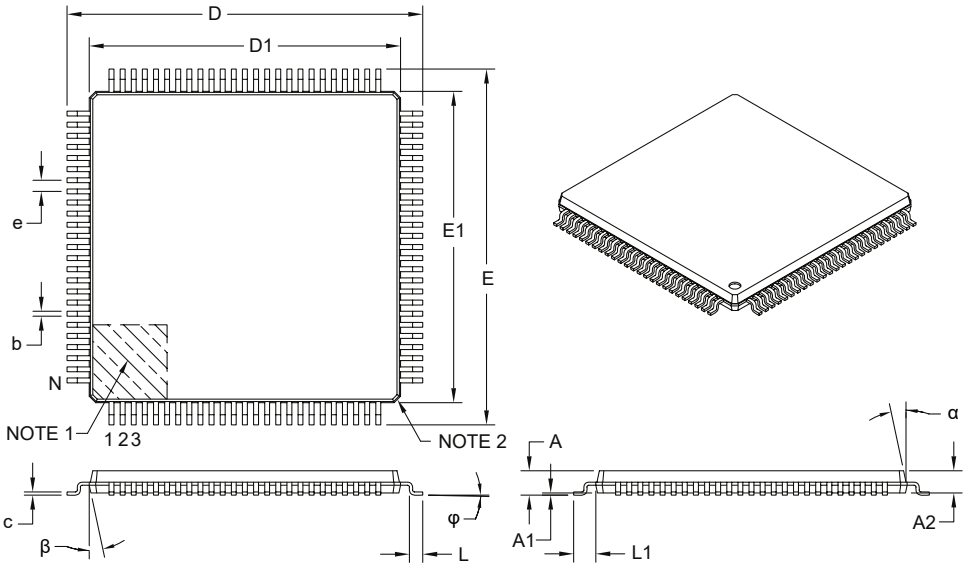
**Note:** For QFN packages with an exposed die attach pad: The exposed die attach pad is not connected electrically inside the device. It is recommenced to attach (solder) the exposed pad to a matching perimeter landing beneath the package punctuated with vias to the ground layer.


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.1. 100-Pin TQFP

**100-Lead Plastic Thin Quad Flatpack (PF) – 14x14x1 mm Body, 2.00 mm [TQFP]**

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 100 | |
| Lead Pitch | e | | 0.50 BSC | |
| Overall Height | A | – | – | 1.20 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | – | 0.15 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | φ | 0° | 3.5° | 7° |
| Overall Width | E | | 16.00 BSC | |
| Overall Length | D | | 16.00 BSC | |
| Molded Package Width | E1 | | 14.00 BSC | |
| Molded Package Length | D1 | | 14.00 BSC | |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.17 | 0.22 | 0.27 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

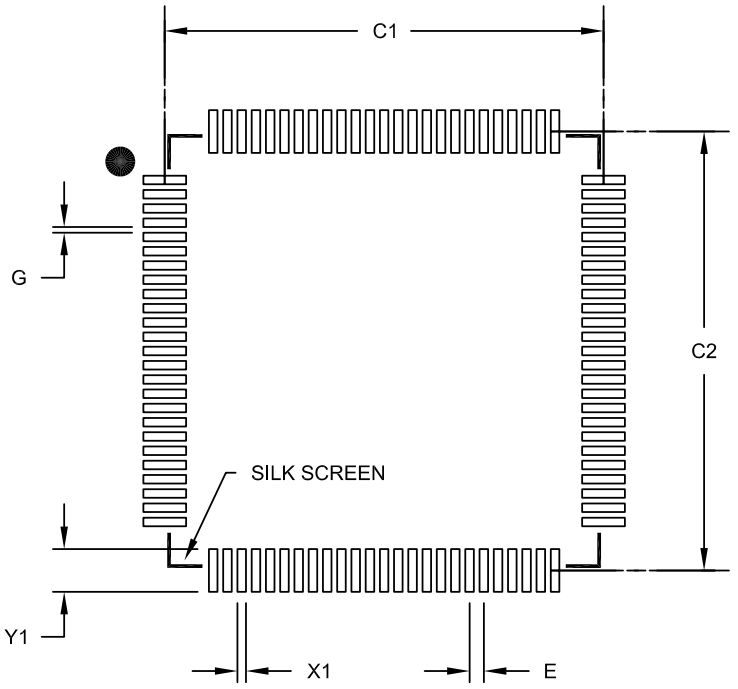
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-110B


© 2026 Microchip Technology Inc. and its subsidiaries

100-Lead Plastic Thin Quad Flatpack (PF) - 14x14x1 mm Body 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Contact Pad Spacing | C1 | | 15.40 | |
| Contact Pad Spacing | C2 | | 15.40 | |
| Contact Pad Width (X100) | X1 | | | 0.30 |
| Contact Pad Length (X100) | Y1 | | | 1.50 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2110B

**Table 52-2.** Device and Package Maximum Weight

| | |
|---|---|
| 500 | mg |

**Table 52-3.** Package Reference

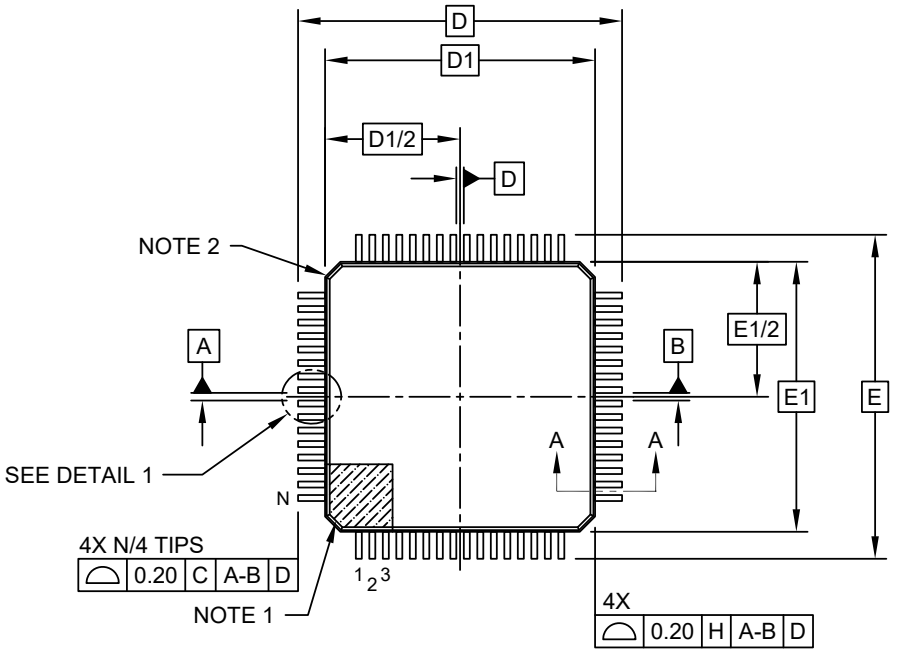
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-110 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

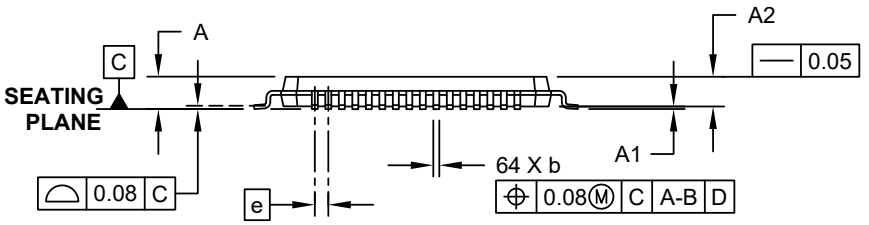
### 52.3.2. 64-Pin TQFP

**64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|




TOP VIEW



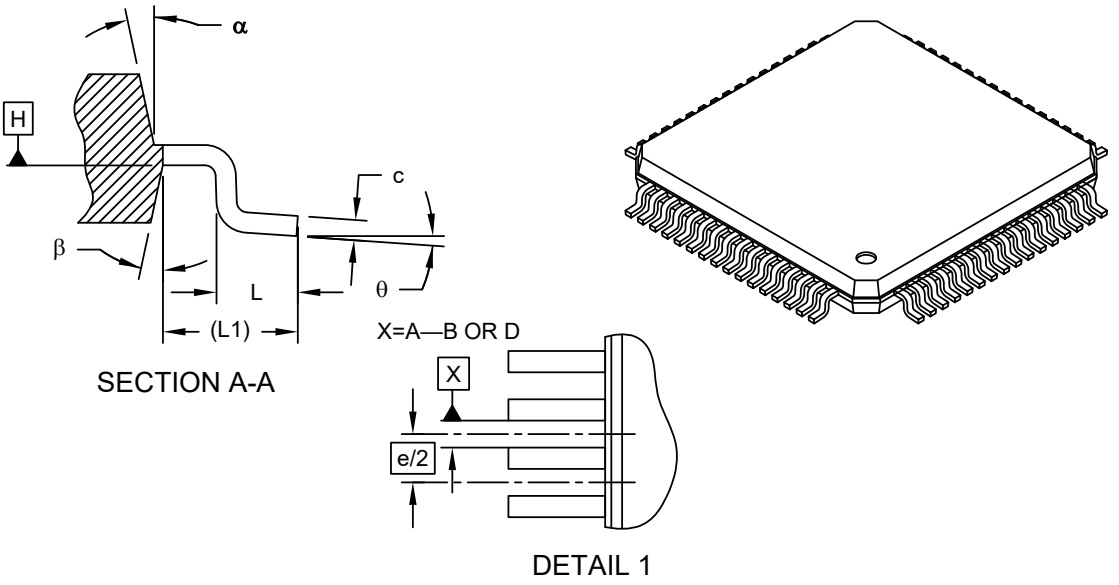

SIDE VIEW

Microchip Technology Drawing C04-085C Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 64 | |
| Lead Pitch | e | | 0.50 BSC | |
| Overall Height | A | - | - | 1.20 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | - | 0.15 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | ϕ | 0° | 3.5° | 7° |
| Overall Width | E | | 12.00 BSC | |
| Overall Length | D | | 12.00 BSC | |
| Molded Package Width | E1 | | 10.00 BSC | |
| Molded Package Length | D1 | | 10.00 BSC | |
| Lead Thickness | c | 0.09 | - | 0.20 |
| Lead Width | b | 0.17 | 0.22 | 0.27 |
| Mold Draft Angle Top | α | 11° | 12° | 13° |
| Mold Draft Angle Bottom | β | 11° | 12° | 13° |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

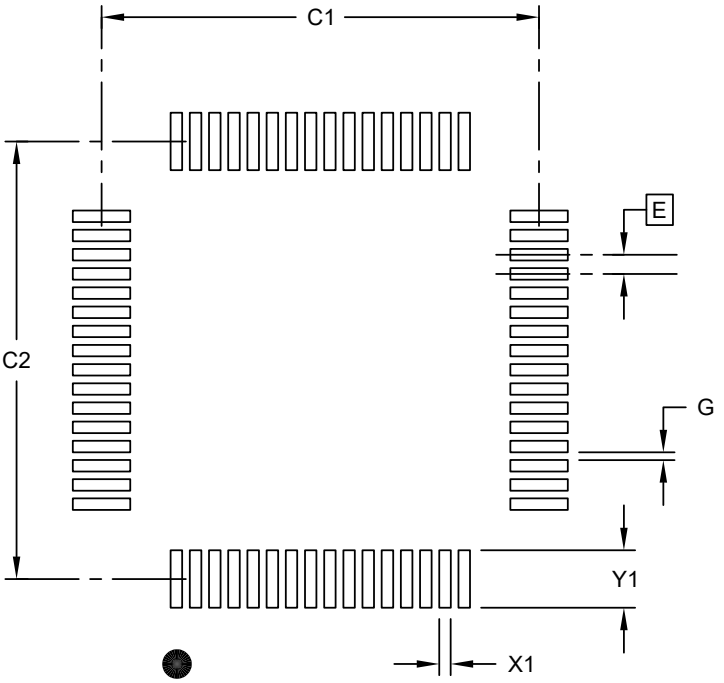
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085C Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

### 64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Contact Pad Spacing | C1 | | 11.40 | |
| Contact Pad Spacing | C2 | | 11.40 | |
| Contact Pad Width (X28) | X1 | | | 0.30 |
| Contact Pad Length (X28) | Y1 | | | 1.50 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2085B Sheet 1 of 1

**Table 52-4.** Device and Package Maximum Weight

| | |
|---|---|
| 300 | mg |

**Table 52-5.** Package Reference

| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-00085 |

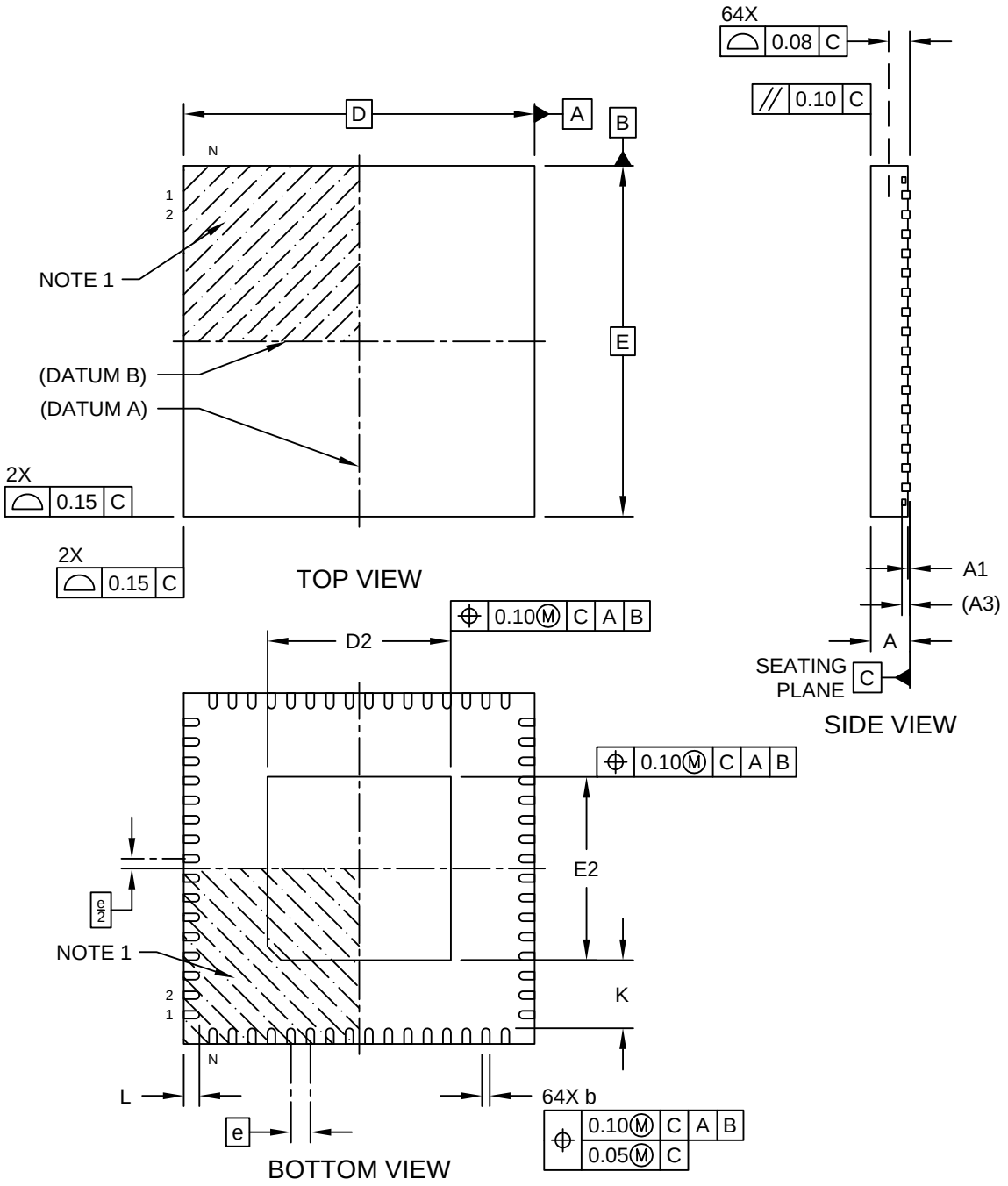

© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| JESD97 Classification | E3 |

### 52.3.3. 64-Pin VQFN

**64-Lead Very Thin Plastic Quad Flat, No Lead Package (TMB) - 9x9 mm Body [VQFN]**
**Atmel Legacy Global Package Code ZST**

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging

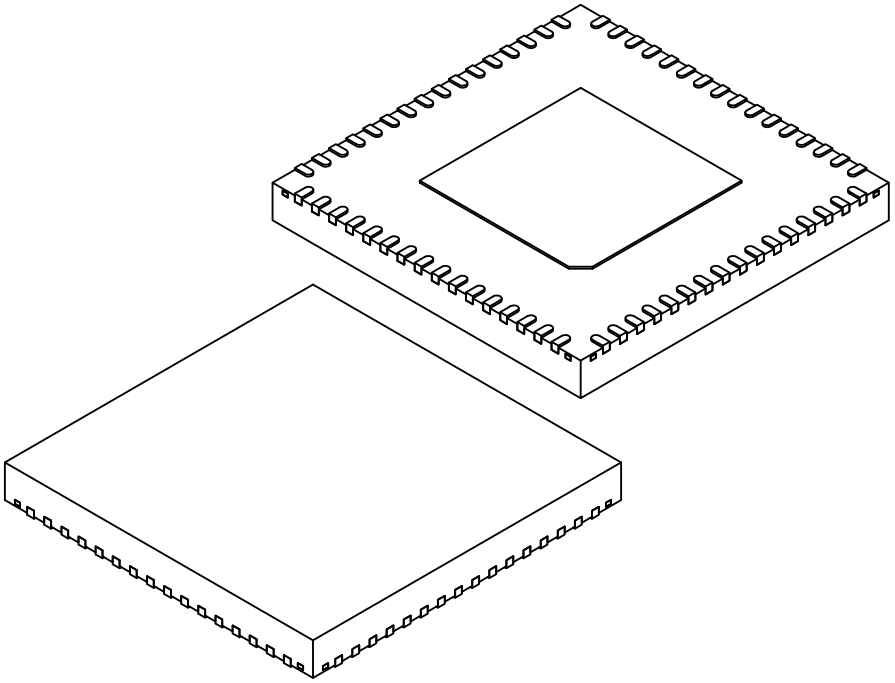



Microchip Technology Drawing C04-21441-TMB Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Very Thin Plastic Quad Flat, No Lead Package (TMB) - 9x9 mm Body [VQFN]
## Atmel Legacy Global Package Code ZST

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 64 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.203 REF | |
| Overall Length | D | | 9.00 BSC | |
| Exposed Pad Length | D2 | 4.60 | 4.70 | 4.80 |
| Overall Width | E | | 9.00 BSC | |
| Exposed Pad Width | E2 | 4.60 | 4.70 | 4.80 |
| Terminal Width | b | 0.15 | 0.20 | 0.25 |
| Terminal Length | L | 0.30 | 0.40 | 0.55 |
| Terminal-to-Exposed-Pad | K | 0.20 | - | - |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

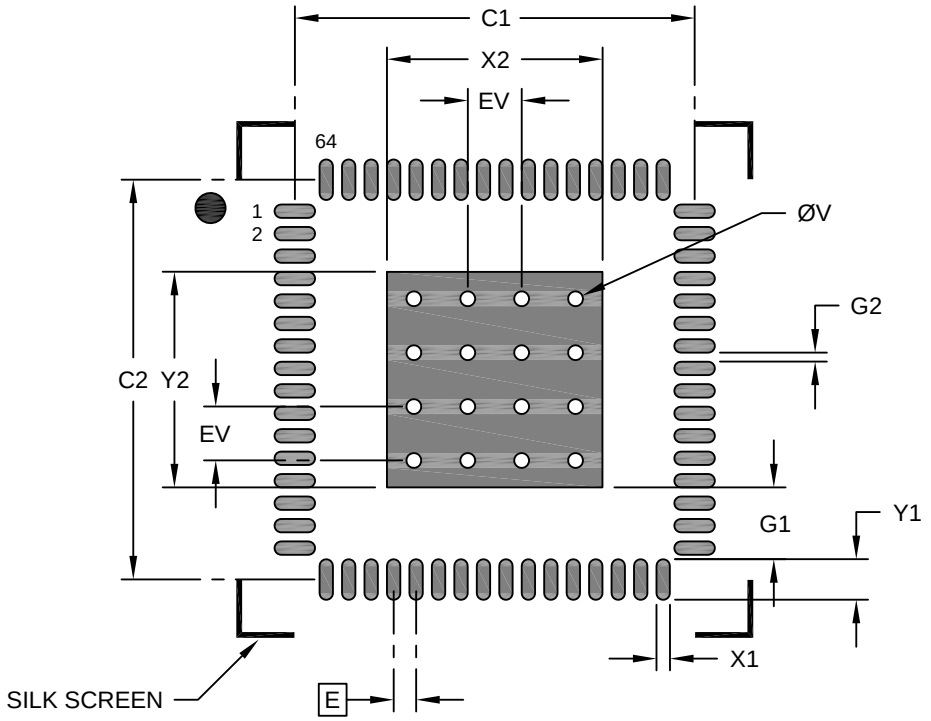
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21441-TMB Rev A Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Very Thin Plastic Quad Flat, No Lead Package (TMB) - 9x9 mm Body [VQFN]
## Atmel Legacy Global Package Code ZST

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 4.80 |
| Optional Center Pad Length | Y2 | | | 4.80 |
| Contact Pad Spacing | C1 | | 8.90 | |
| Contact Pad Spacing | C2 | | 8.90 | |
| Contact Pad Width (X64) | X1 | | | 0.30 |
| Contact Pad Length (X64) | Y1 | | | 0.90 |
| Contact Pad to Center Pad (X64) | G1 | 1.60 | | |
| Contact Pad to Contact Pad (X60) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.30 | |
| Thermal Via Pitch | EV | | 1.00 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-21441-TMB Rev A Sheet 1 of 2

**Note:** The exposed die attach pad is not connected electrically inside the device. It is recommenced to attach (solder) the exposed pad to a matching perimeter landing beneath the package punctuated with vias to the ground layer.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-6.** Device and Package Maximum Weight

| | |
|---|---|
| 200 | mg |

**Table 52-7.** Package Reference

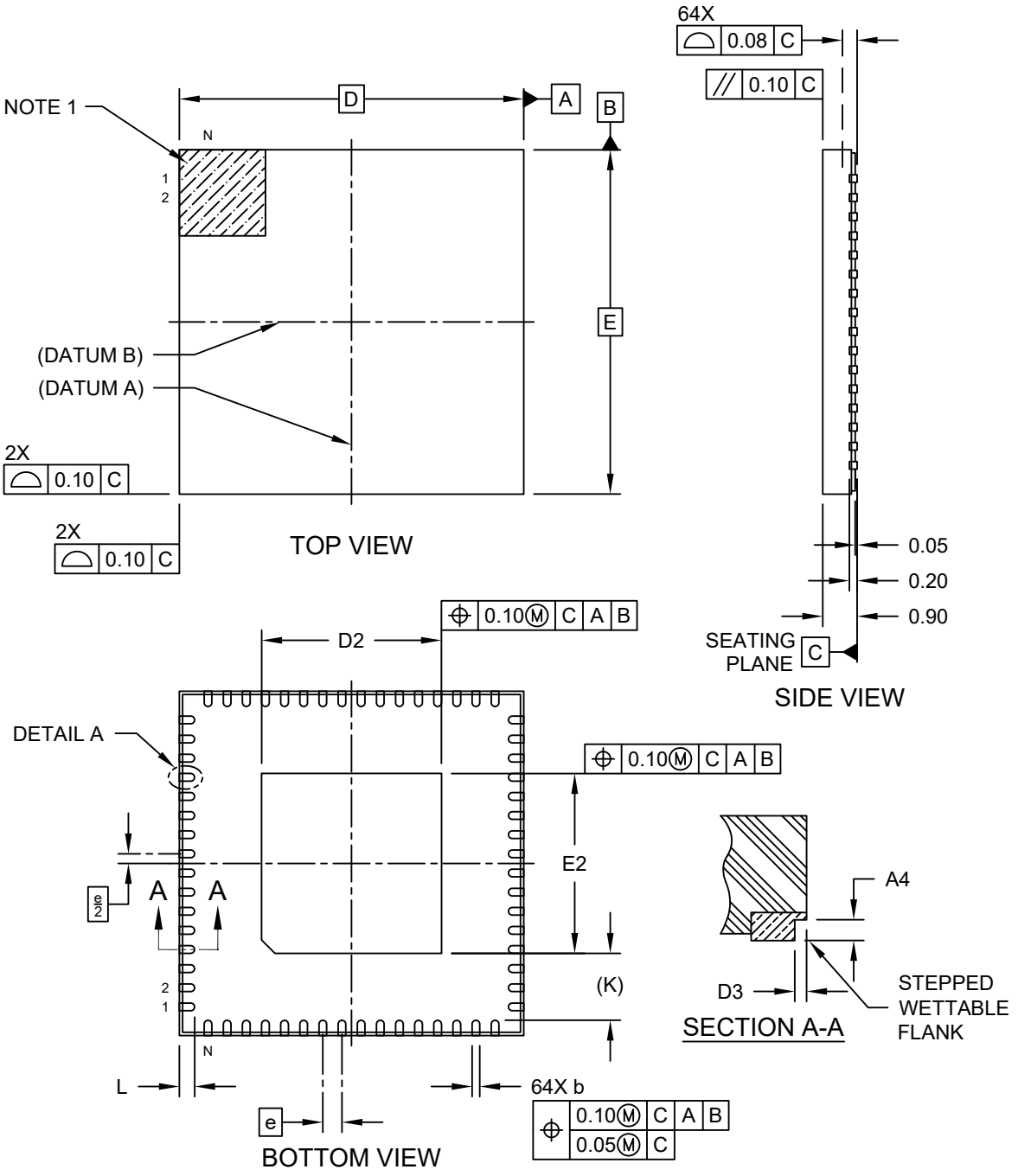
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21441 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.4. 64-Pin VQFN AEC - Q100

**64-Lead Very Thin Plastic Quad Flat, No Lead Package (U6B) - 9x9 mm Body [VQFN] With 4.7 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZRB**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

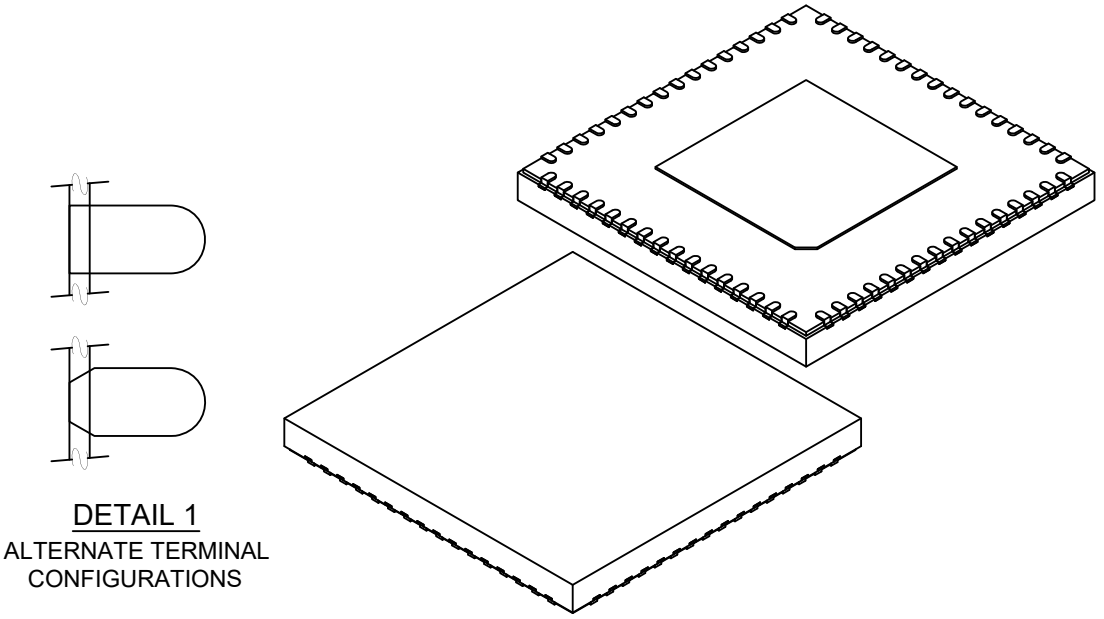



Microchip Technology Drawing C04-21497 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Very Thin Plastic Quad Flat, No Lead Package (U6B) - 9x9 mm Body [VQFN] With 4.7 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZRB

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



DETAIL 1
ALTERNATE TERMINAL CONFIGURATIONS

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 64 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.85 | 0.90 |
| Standoff | A1 | 0.00 | 0.035 | 0.05 |
| Terminal Thickness | A3 | | 0.203 REF | |
| Overall Length | D | | 9.00 BSC | |
| Exposed Pad Length | D2 | 4.60 | 4.70 | 4.80 |
| Overall Width | E | | 9.00 BSC | |
| Exposed Pad Width | E2 | 4.60 | 4.70 | 4.80 |
| Terminal Width | b | 0.15 | 0.20 | 0.25 |
| Terminal Length | L | 0.35 | 0.40 | 0.45 |
| Terminal-to-Exposed-Pad | K | | 1.75 REF | |
| Wettable Flank Step Length | D3 | - | - | 0.085 |
| Wettable Flank Step Height | A4 | 0.10 | - | 0.19 |

Notes:

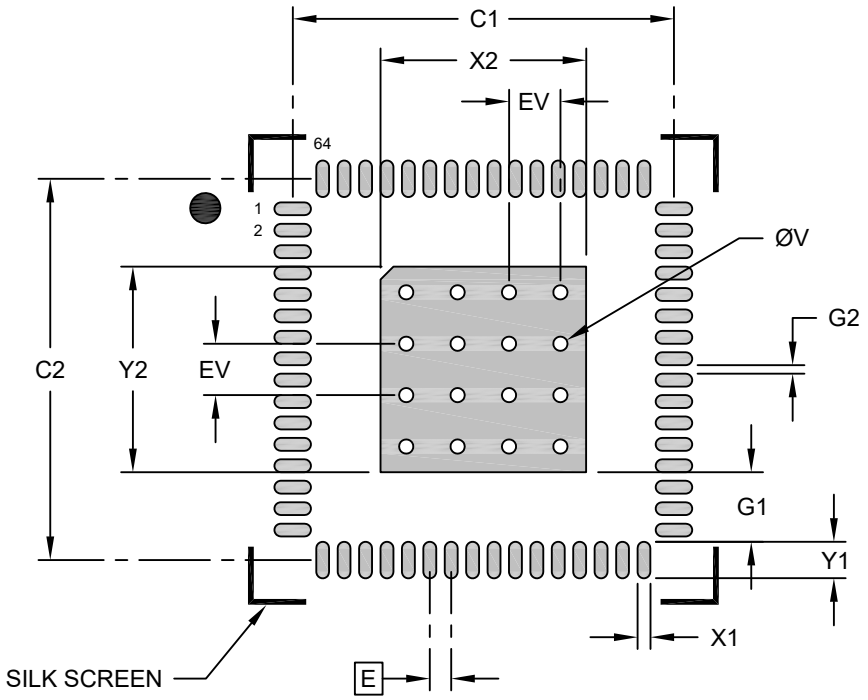
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21497 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Very Thin Plastic Quad Flat, No Lead Package (U6B) - 9x9 mm Body [VQFN] With 4.7 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZRB

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 4.80 |
| Optional Center Pad Length | Y2 | | | 4.80 |
| Contact Pad Spacing | C1 | | 8.90 | |
| Contact Pad Spacing | C2 | | 8.90 | |
| Contact Pad Width (X64) | X1 | | | 0.30 |
| Contact Pad Length (X64) | Y1 | | | 0.85 |
| Contact Pad to Center Pad (X64) | G1 | 1.63 | | |
| Contact Pad to Contact Pad (X60) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.33 | |
| Thermal Via Pitch | EV | | 1.20 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23497 Rev A

**Table 52-8.** Device and Package Maximum Weight

| | |
|---|---|
| 232.4 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-9.** Package Reference

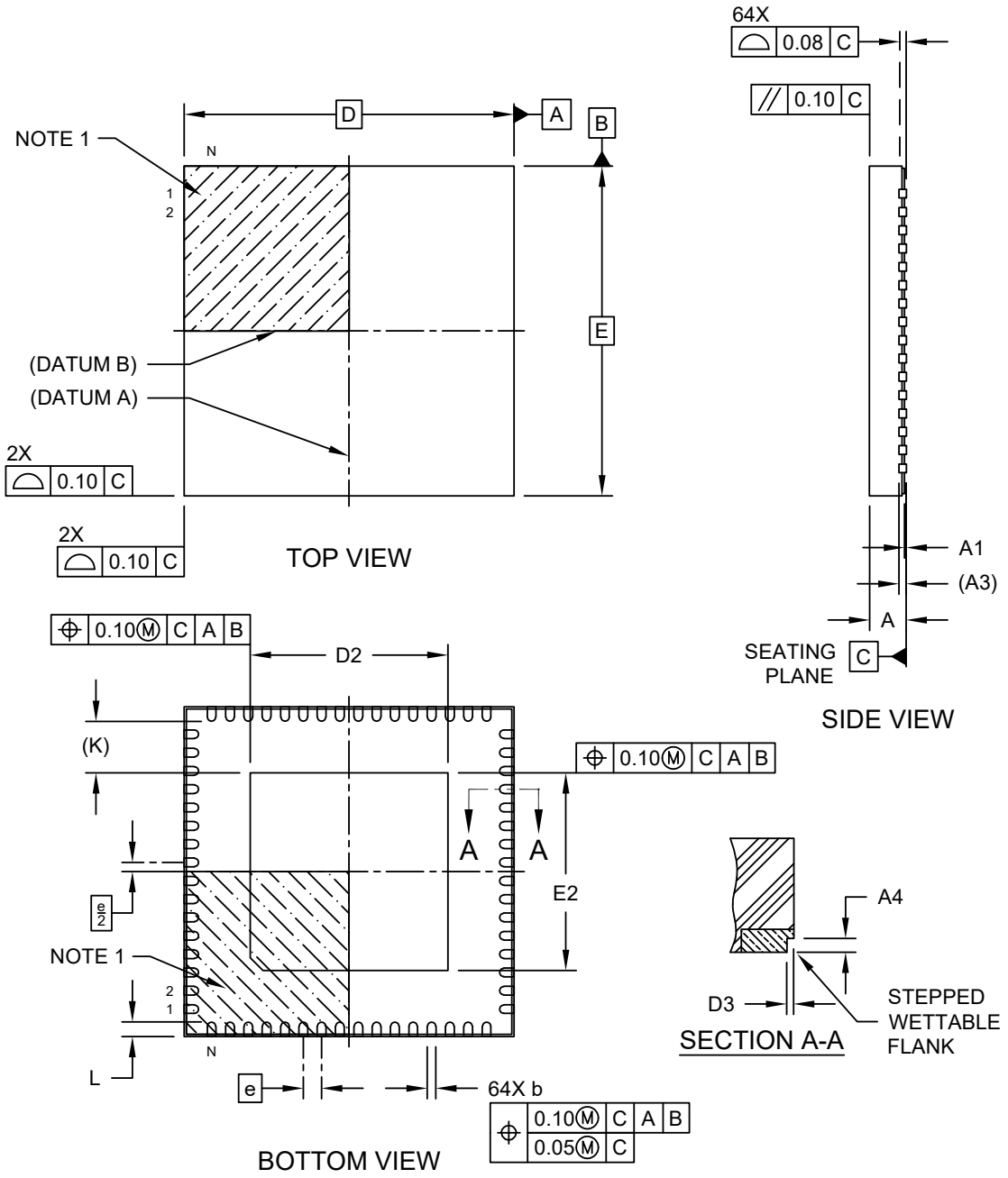
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21497 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.5. 64-Pin VQFN

**64-Lead Very Thin Plastic Quad Flat, No Lead Package (5LX) - 9x9x1.0 mm Body [VQFN] With 5.4 mm Exposed Pad and Stepped Wettable Flanks**

| | |
|---|---|
| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |

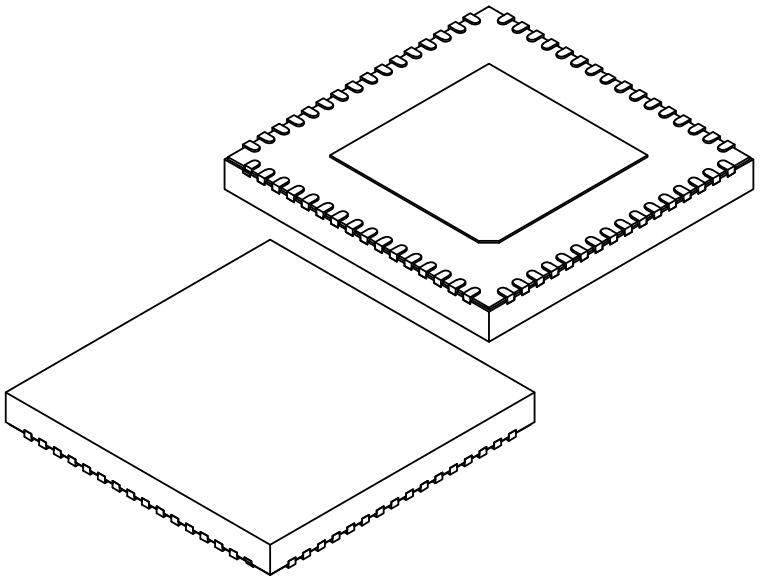



Microchip Technology Drawing C04-483 Rev E Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 64-Lead Very Thin Plastic Quad Flat, No Lead Package (5LX) - 9x9x1.0 mm Body [VQFN] With 5.4 mm Exposed Pad and Stepped Wettable Flanks

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 64 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.203 REF | |
| Overall Length | D | | 9.00 BSC | |
| Exposed Pad Length | D2 | 5.30 | 5.40 | 5.50 |
| Overall Width | E | | 9.00 BSC | |
| Exposed Pad Width | E2 | 5.30 | 5.40 | 5.50 |
| Terminal Width | b | 0.20 | 0.25 | 0.30 |
| Terminal Length | L | 0.30 | 0.40 | 0.50 |
| Terminal-to-Exposed-Pad | K | | 1.40 REF | |
| Wettable Flank Step Length | D3 | 0.035 | 0.060 | 0.085 |
| Wettable Flank Step Height | A4 | 0.10 | - | 0.19 |

Notes:

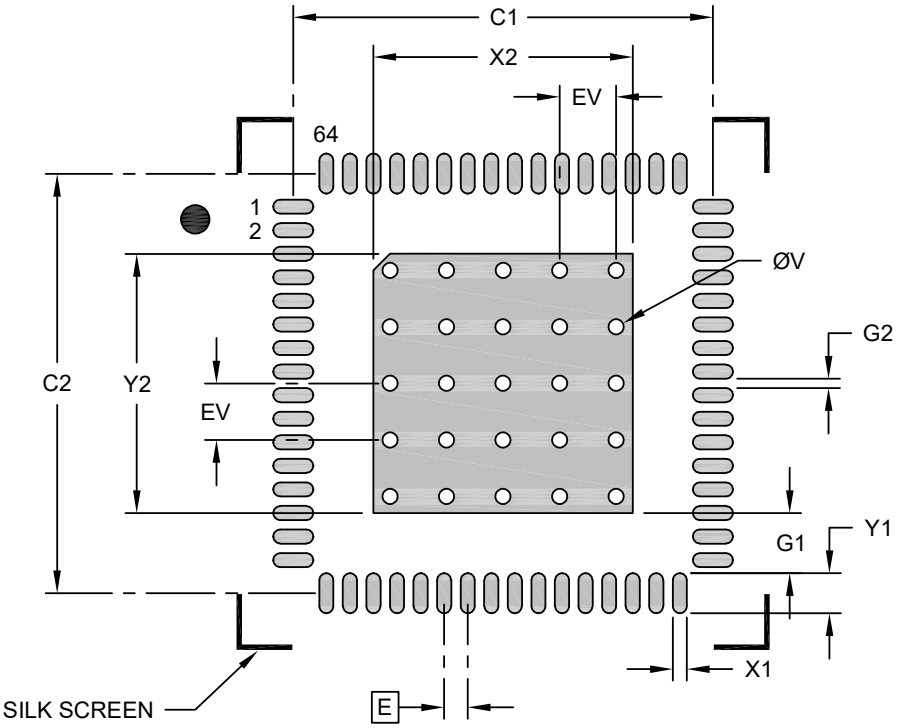
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-483 Rev E Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

### 64-Lead Very Thin Plastic Quad Flat, No Lead Package (5LX) - 9x9x1.0 mm Body [VQFN] With 5.4 mm Exposed Pad and Stepped Wettable Flanks

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 5.50 |
| Optional Center Pad Length | Y2 | | | 5.50 |
| Contact Pad Spacing | C1 | | 8.90 | |
| Contact Pad Spacing | C2 | | 8.90 | |
| Contact Pad Width (X64) | X1 | | | 0.30 |
| Contact Pad Length (X64) | Y1 | | | 0.85 |
| Contact Pad to Center Pad (X64) | G1 | 1.28 | | |
| Contact Pad to Contact Pad (X60) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.33 | |
| Thermal Via Pitch | EV | | 1.20 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2483 Rev E

**Table 52-10.** Device and Package Maximum Weight

| | |
|---|---|
| 232.4 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-11.** Package Reference

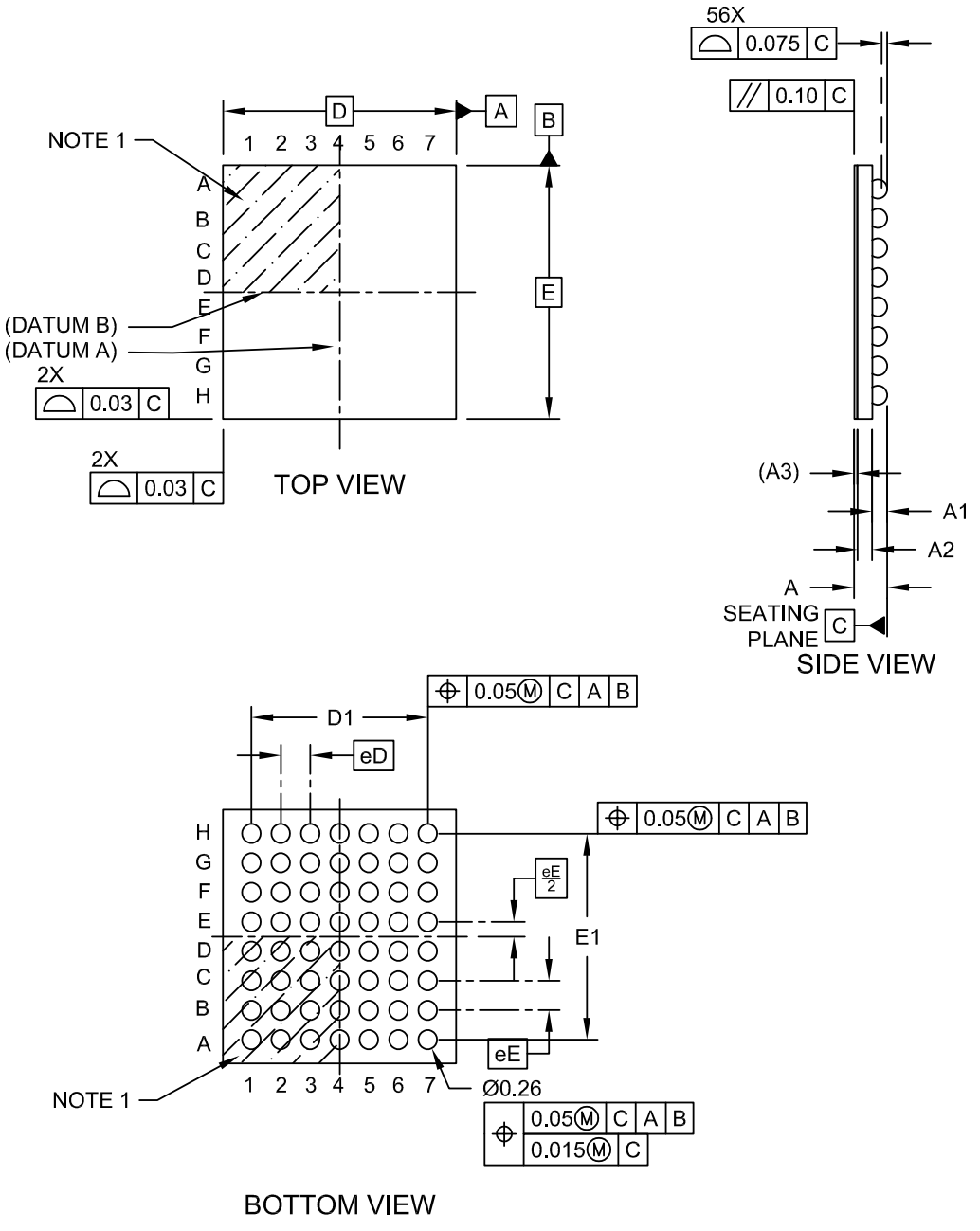
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-00483 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.6. 56-Ball WLCSP

**56-ball Wafer Level Chip Scale Package (FZB) - 3.170x3.444x0.483 mm Body [WLCSP]**
**Atmel Legacy GPC GBL**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

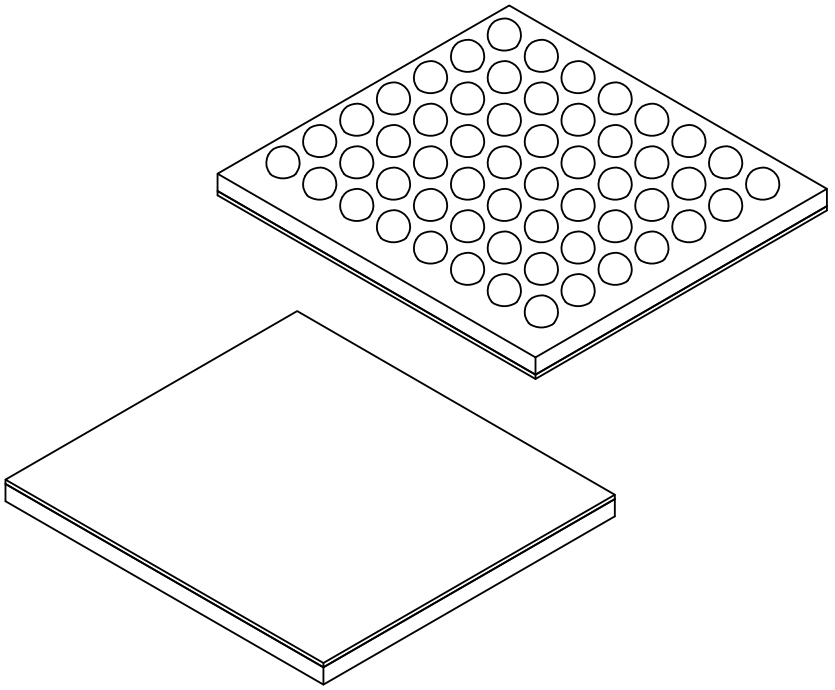



Microchip Technology Drawing C04-21254 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 56-ball Wafer Level Chip Scale Package (FZB) - 3.170x3.444x0.483 mm Body [WLCSP] Atmel Legacy GPC GBL

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 56 | |
| Pitch | eD | | 0.40 BSC | |
| Pitch | eE | | 0.40 BSC | |
| Overall Height | A | 0.403 | 0.443 | 0.483 |
| Ball Height | A1 | 0.17 | - | 0.23 |
| Wafer Thickness | A2 | 0.178 | 0.203 | 0.228 |
| Backside Coat Thickness | A3 | | 0.040 REF | |
| Overall Length | D | | 3.170 BSC | |
| Ball Array Length | D1 | | 2.400 BSC | |
| Overall Width | E | | 3.444 BSC | |
| Ball Array Width | E1 | | 2.800 BSC | |
| Ball Width | b | 0.23 | 0.26 | 0.29 |
| Ball Diameter | | | 0.25 REF | |

Notes:

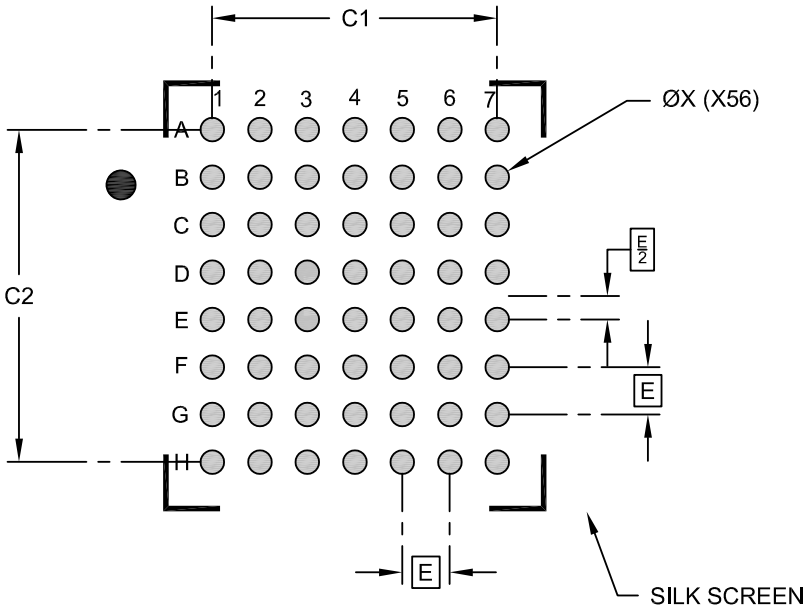
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21254 Rev A Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 56-ball Wafer Level Chip Scale Package (FZB) - 3.170x3.444x0.483 mm Body [WLCSP] Atmel Legacy GPC GBL

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.40 BSC | |
| Center Ball Diameter (X56) | X | | | 0.200 |
| Contact Pad Spacing | C1 | | 2.40 | |
| Contact Pad Spacing | C2 | | 2.80 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23254 Rev A

**Table 52-12.** Device and Package Maximum Weight

| | |
|---|---|
| 9.63 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-13.** Package Reference

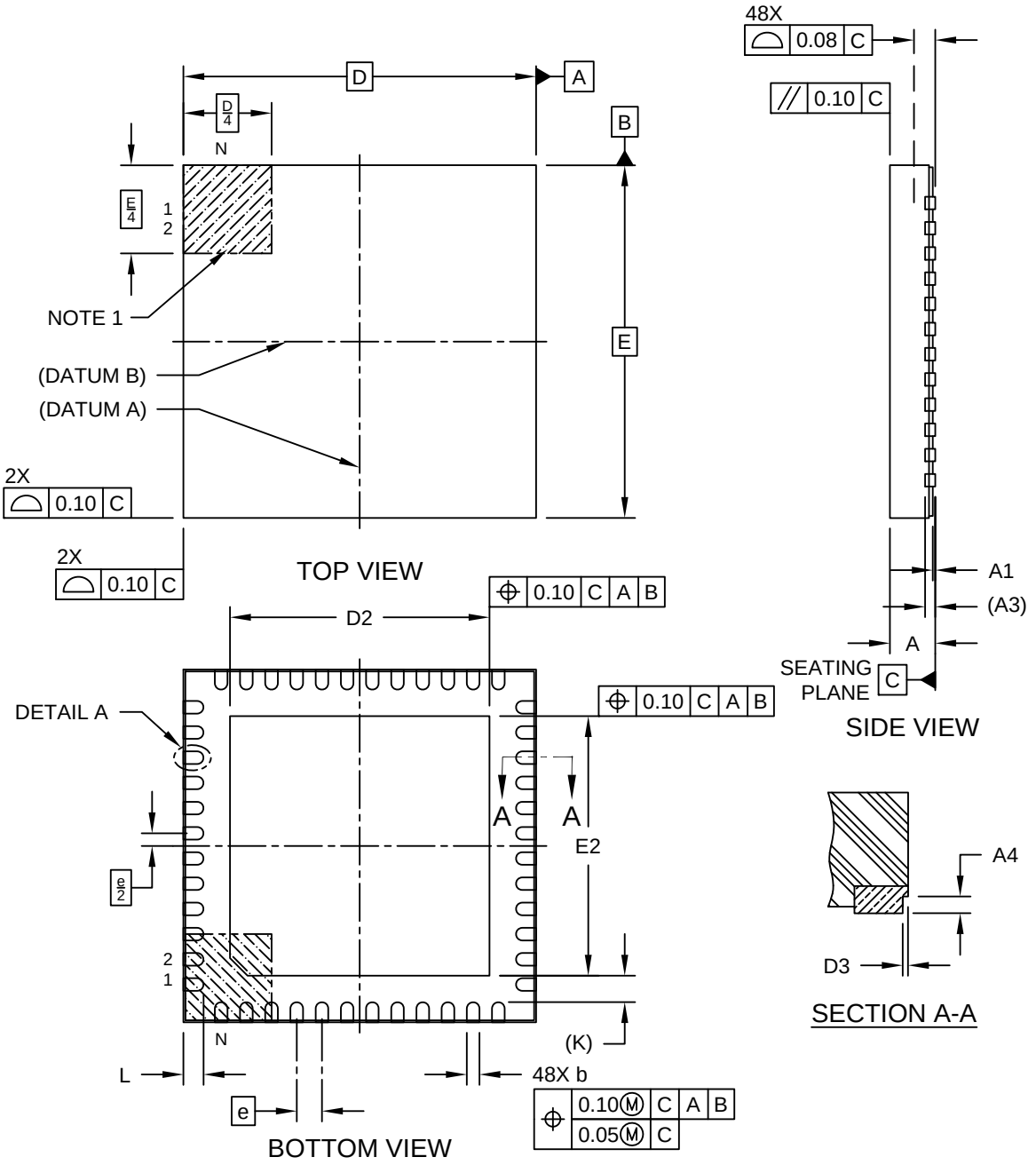
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-2125 |
| JESD97 Classification | E1 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.7. 48-Pin VQFN AEC - Q100

**48-Lead Very Thin Plastic Quad Flat, No Lead Package (U5B) - 7x7 mm Body [VQFN] With 5.15 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZLH**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

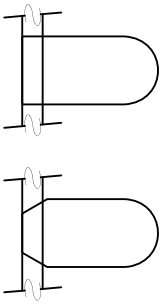



Microchip Technology Drawing C04-21493 Rev A Sheet 1 of 2

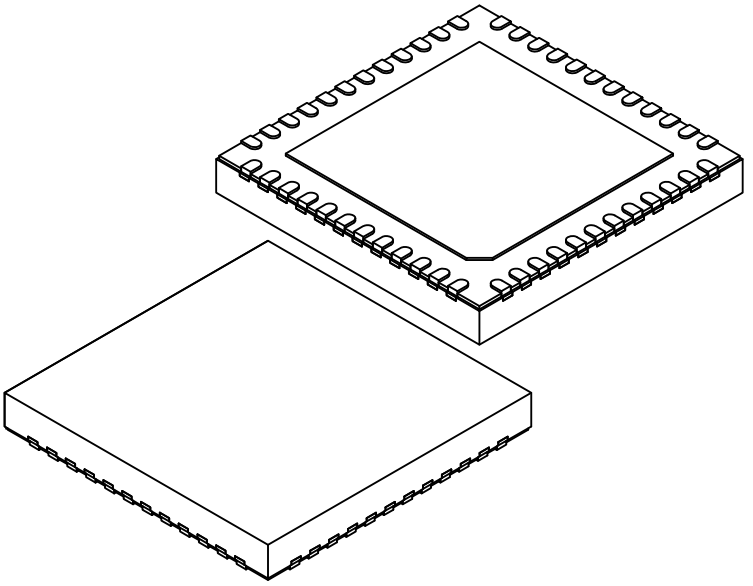

© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Very Thin Plastic Quad Flat, No Lead Package (U5B) - 7x7 mm Body [VQFN] With 5.15 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZLH

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



DETAIL 1
ALTERNATE TERMINAL CONFIGURATIONS



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 48 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0,80 | 0.85 | 0.90 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.203 REF | |
| Overall Length | D | | 7.00 BSC | |
| Exposed Pad Length | D2 | 5.05 | 5.15 | 5.25 |
| Overall Width | E | | 7.00 BSC | |
| Exposed Pad Width | E2 | 5.05 | 5.15 | 5.25 |
| Terminal Width | b | 0.20 | 0.25 | 0.30 |
| Terminal Length | L | 0.35 | 0.40 | 0.45 |
| Terminal-to-Exposed-Pad | K | | 0.53 REF | |
| Wettable Flank Step Length | D3 | - | - | 0.085 |
| Wettable Flank Step Height | A4 | 0.10 | - | 0.19 |

Notes:

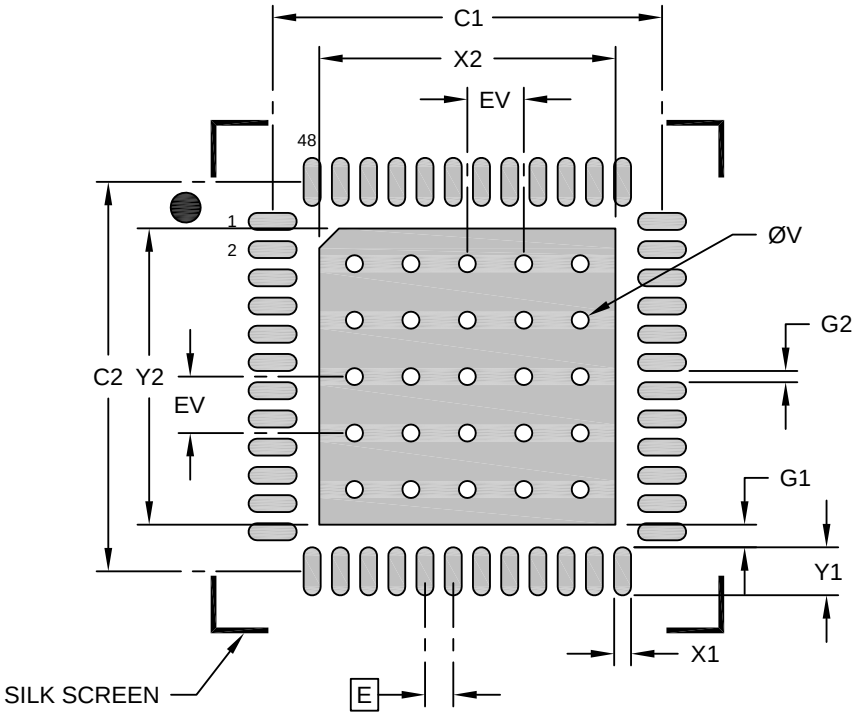
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21493 Rev A Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Very Thin Plastic Quad Flat, No Lead Package (U5B) - 7x7 mm Body [VQFN] With 5.15 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZLH

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 5.25 |
| Optional Center Pad Length | Y2 | | | 5.25 |
| Contact Pad Spacing | C1 | | 6.90 | |
| Contact Pad Spacing | C2 | | 6.90 | |
| Contact Pad Width (X48) | X1 | | | 0.30 |
| Contact Pad Length (X48) | Y1 | | | 0.85 |
| Contact Pad to Center Pad (X48) | G1 | 0.20 | | |
| Contact Pad to Center Pad (X44) | G2 | 0.40 | | |
| Thermal Via Diameter | V | | 0.30 | |
| Thermal Via Pitch | EV | | 1.00 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23493 Rev A

**Table 52-14.** Device and Package Maximum Weight

| | |
|---|---|
| 137.2 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-15.** Package Reference

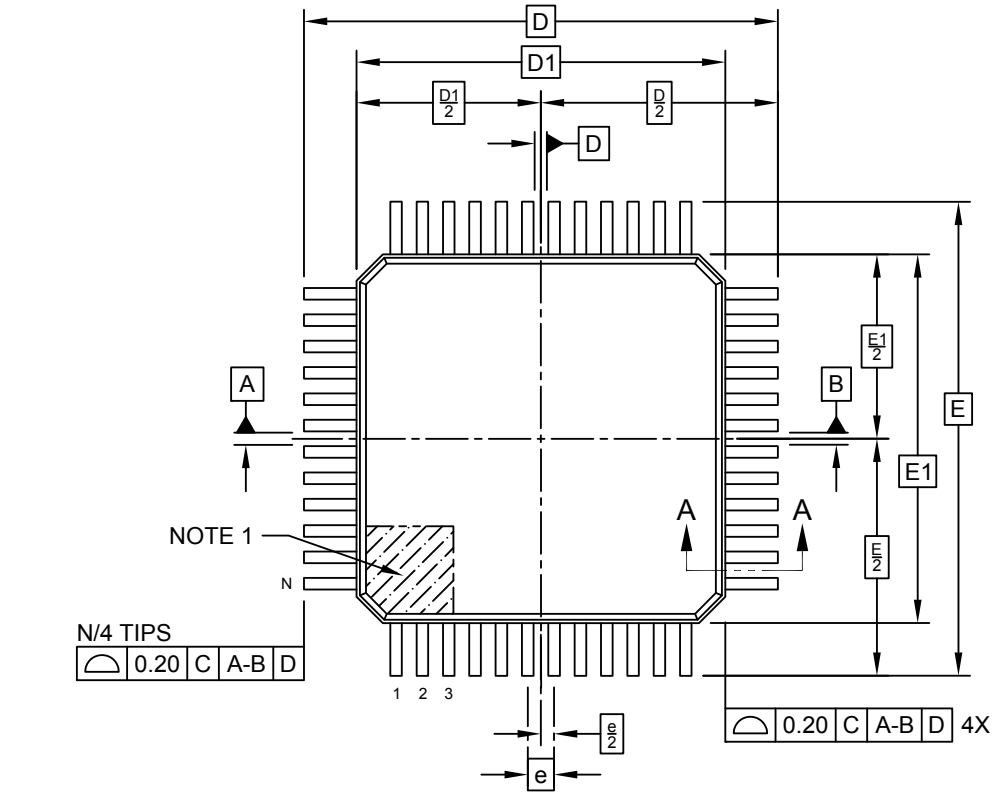
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21493 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

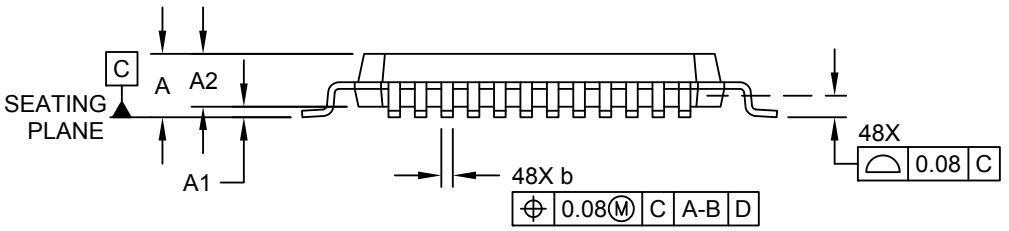
### 52.3.8. 48-Pin TQFP

**48-Lead Plastic Thin Quad Flatpack (Y8) - 7x7x1.0 mm Body [TQFP]**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|




TOP VIEW



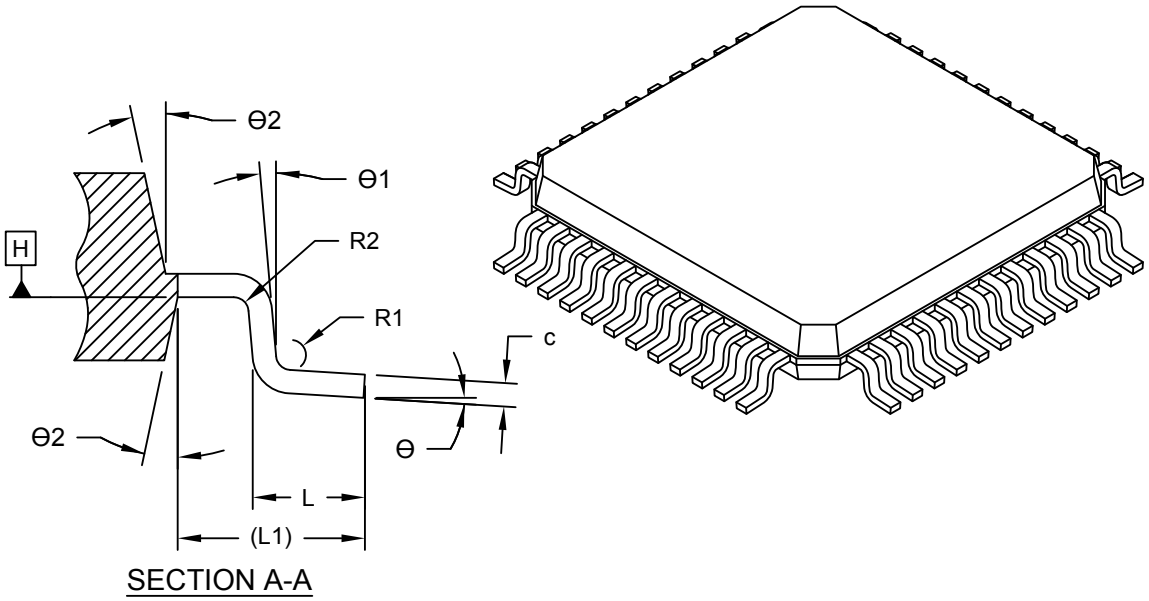

SIDE VIEW

Microchip Technology Drawing C04-300-Y8 Rev D Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Plastic Thin Quad Flatpack (Y8) - 7x7x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




SECTION A-A

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 48 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | - | - | 1.20 |
| Standoff | A1 | 0.05 | - | 0.15 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Overall Length | D | | 9.00 BSC | |
| Molded Package Length | D1 | | 7.00 BSC | |
| Overall Width | E | | 9.00 BSC | |
| Molded Package Width | E1 | | 7.00 BSC | |
| Terminal Width | b | 0.17 | 0.22 | 0.27 |
| Terminal Thickness | c | 0.09 | - | 0.16 |
| Terminal Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Lead Bend Radius | R1 | 0.08 | - | - |
| Lead Bend Radius | R2 | 0.08 | - | 0.20 |
| Foot Angle | Θ | 0° | 3.5° | 7° |
| Lead Angle | Θ1 | 0° | - | - |
| Mold Draft Angle | Θ2 | 11° | 12° | 13° |

Notes:

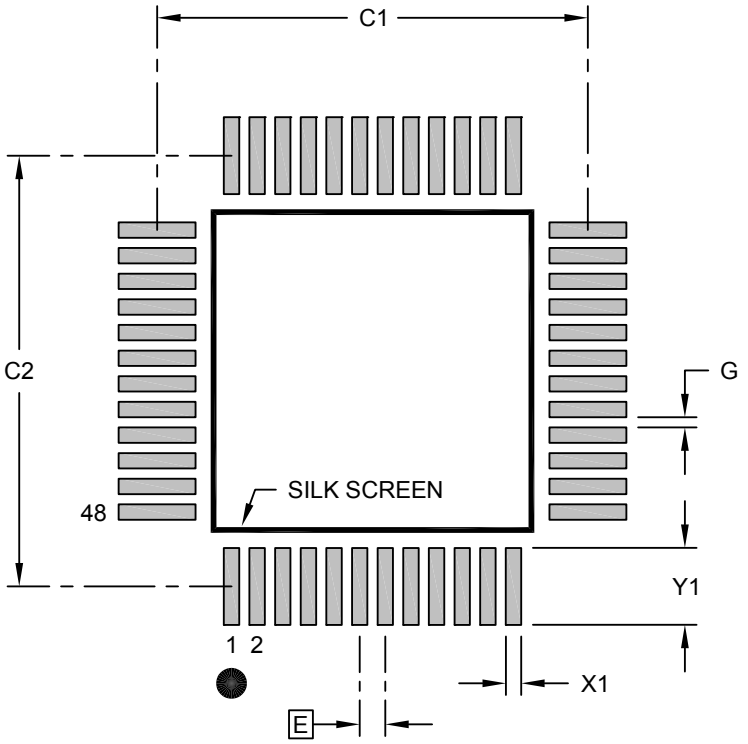
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-300-Y8 Rev D Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Plastic Thin Quad Flatpack (Y8) - 7x7x1.0 mm Body [TQFP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Contact Pad Spacing | C1 | | 8.40 | |
| Contact Pad Spacing | C2 | | 8.40 | |
| Contact Pad Width (X48) | X1 | | | 0.30 |
| Contact Pad Length (X48) | Y1 | | | 1.50 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-2300-Y8 Rev D

**Table 52-16.** Device and Package Maximum Weight

| | |
|---|---|
| 140 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-17.** Package Reference

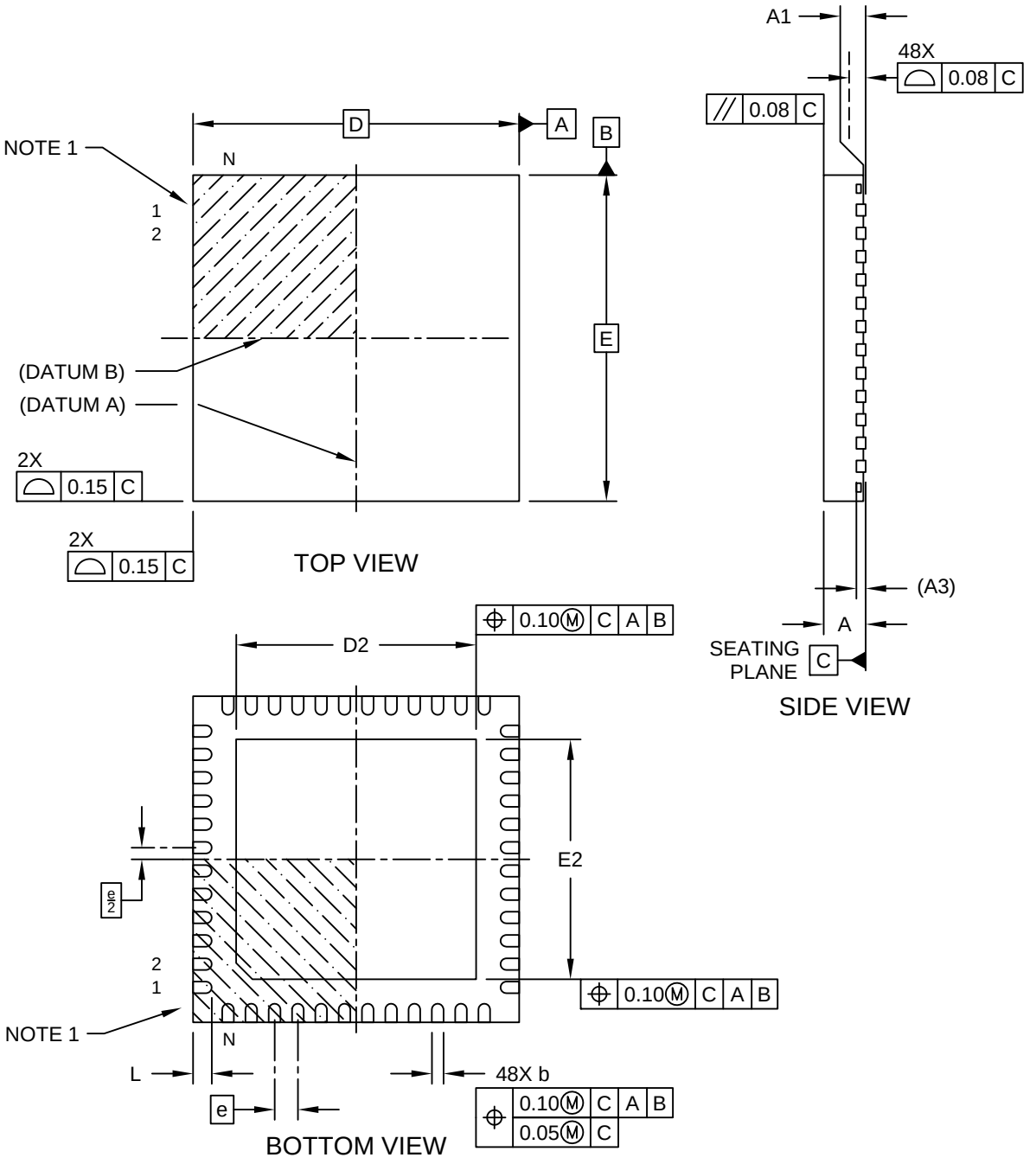
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-00300 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.9. 48-Pin VQFN

**48-Lead Very Thin Quad Flat Pack No-Leads (T3B) 7x7x0.9 mm [VQFN]**
**With 5.15x5.15 mm Exposed Pad; Atmel Legacy Global Package Code ZLG**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

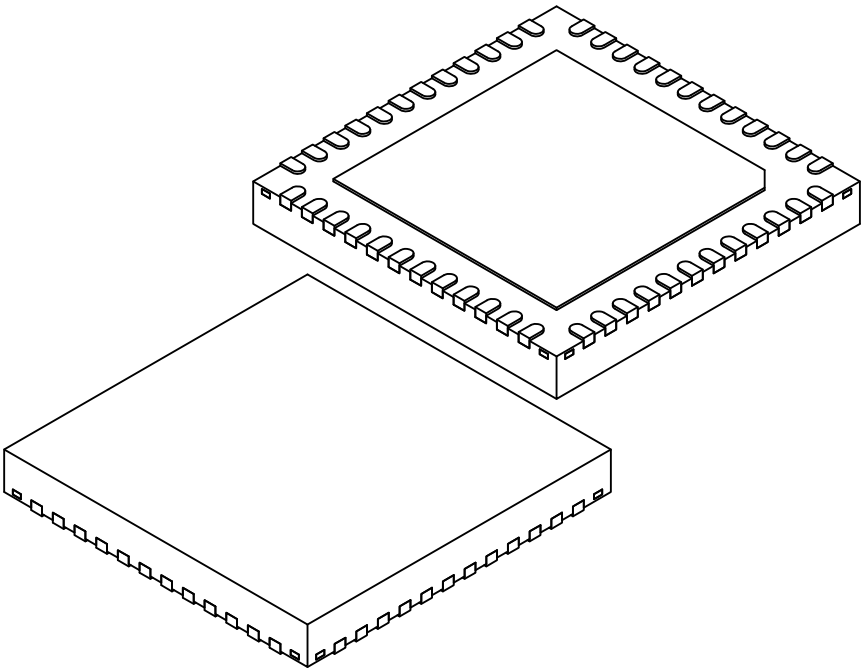



Microchip Technology Drawing C04-21425 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Very Thin Quad Flat Pack No-Leads (T3B) 7x7x0.9 mm [VQFN] With 5.15x5.15 mm Exposed Pad; Atmel Legacy Global Package Code ZLG

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 48 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.85 | 0.90 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.20 REF | |
| Overall Length | D | | 7.00 BSC | |
| Exposed Pad Length | D2 | 5.05 | 5.15 | 5.25 |
| Overall Width | E | | 7.00 BSC | |
| Exposed Pad Width | E2 | 5.05 | 5.15 | 5.25 |
| Terminal Width | b | 0.18 | 0.25 | 0.30 |
| Terminal Length | L | 0.30 | 0.40 | 0.50 |
| Terminal-to-Exposed-Pad | K | 0.20 | - | - |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

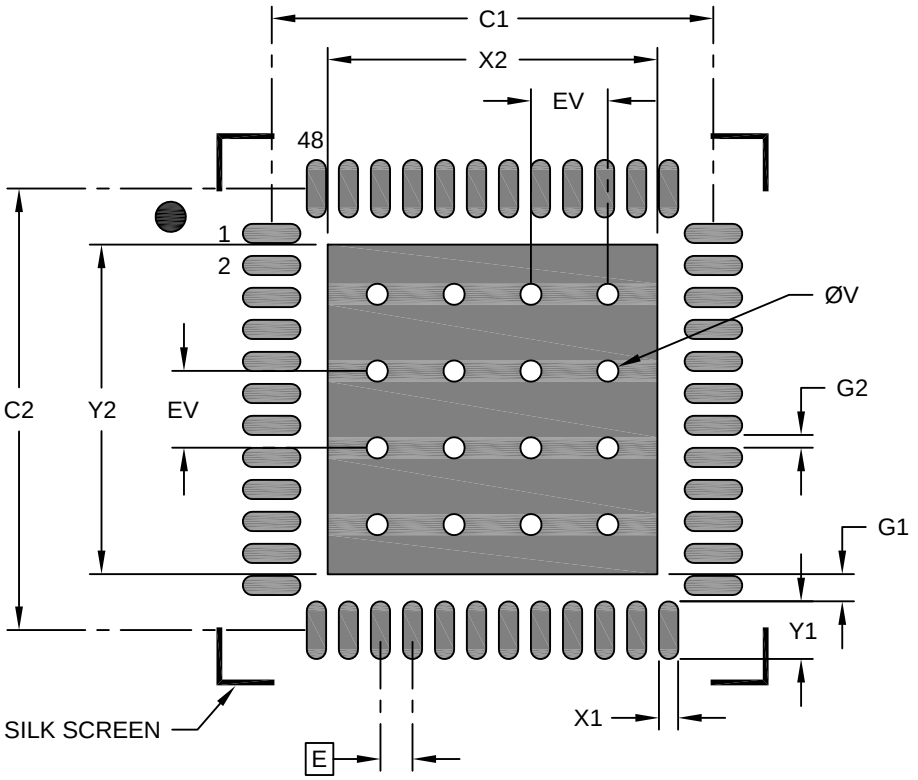
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21425 Rev A Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 48-Lead Very Thin Quad Flat Pack No-Leads (T3B) 7x7x0.9 mm [VQFN] With 5.15x5.15 mm Exposed Pad; Atmel Legacy Global Package Code ZLG

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 5.15 |
| Optional Center Pad Length | Y2 | | | 5.15 |
| Contact Pad Spacing | C1 | | 6.90 | |
| Contact Pad Spacing | C2 | | 6.90 | |
| Contact Pad Width (X20) | X1 | | | 0.30 |
| Contact Pad Length (X20) | Y1 | | | 0.90 |
| Contact Pad to Center Pad (X48) | G1 | 0.20 | | |
| Contact Pad to Contact Pad (X44) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.33 | |
| Thermal Via Pitch | EV | | 1.20 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23425 Rev A

**Note:** The exposed die attach pad is not connected electrically inside the device. It is recommenced to attach (solder) the exposed pad to a matching perimeter landing beneath the package punctuated with vias to the ground layer.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-18.** Device and Package Maximum Weight

| | |
|---|---|
| 140 | mg |

**Table 52-19.** Package Reference

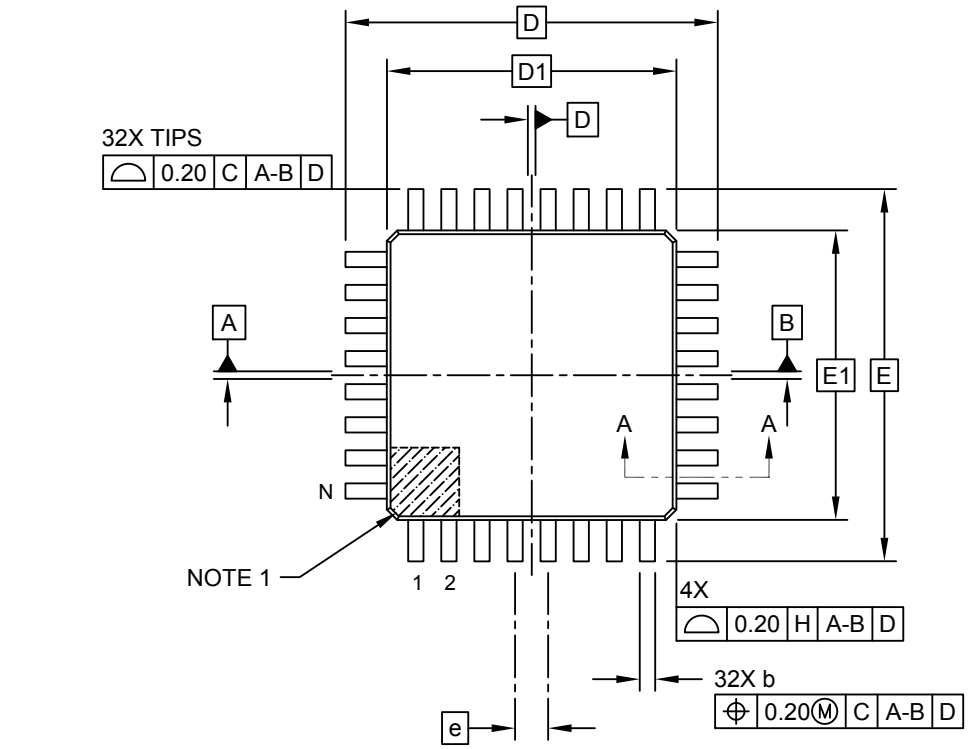
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21425 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

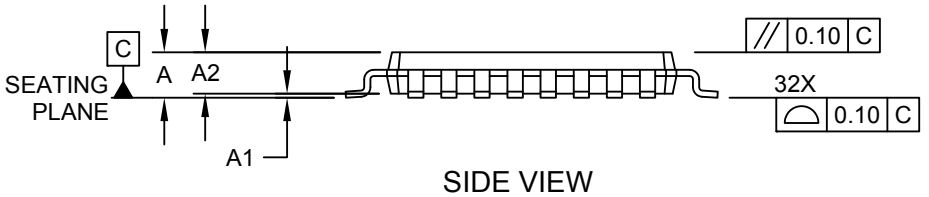
### 52.3.10. 32-Pin TQFP

**32-Lead Plastic Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]**
**2.00 mm Footprint; Also Atmel Legacy Global Package Code AUT**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|




TOP VIEW



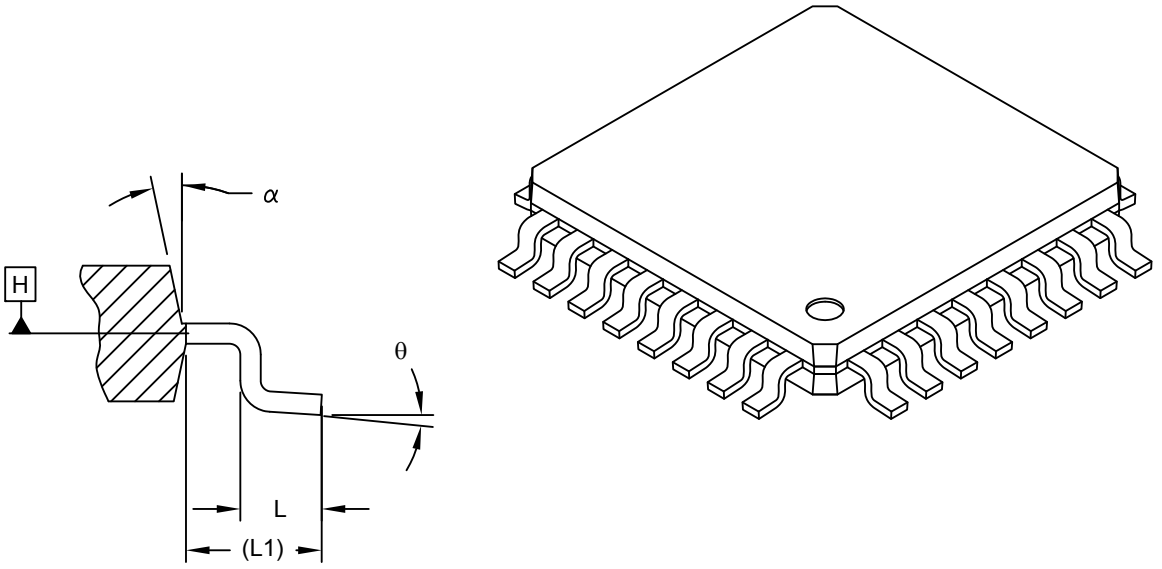

SIDE VIEW

Microchip Technology Drawing C04-074 Rev C Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Plastic Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]
## 2.00 mm Footprint; Also Atmel Legacy Global Package Code AUT

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




SECTION A-A

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Leads | N | | 32 | |
| Lead Pitch | e | | 0.80 BSC | |
| Overall Height | A | - | - | 1.20 |
| Standoff | A1 | 0.05 | - | 0.15 |
| Molded Package Thickness | A2 | 0.95 | 1.00 | 1.05 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | θ | 0° | - | 7° |
| Overall Width | E | | 9.00 BSC | |
| Overall Length | D | | 9.00 BSC | |
| Molded Package Width | E1 | | 7.00 BSC | |
| Molded Package Length | D1 | | 7.00 BSC | |
| Lead Width | b | 0.30 | 0.37 | 0.45 |
| Mold Draft Angle Top | α | 11° | - | 13° |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

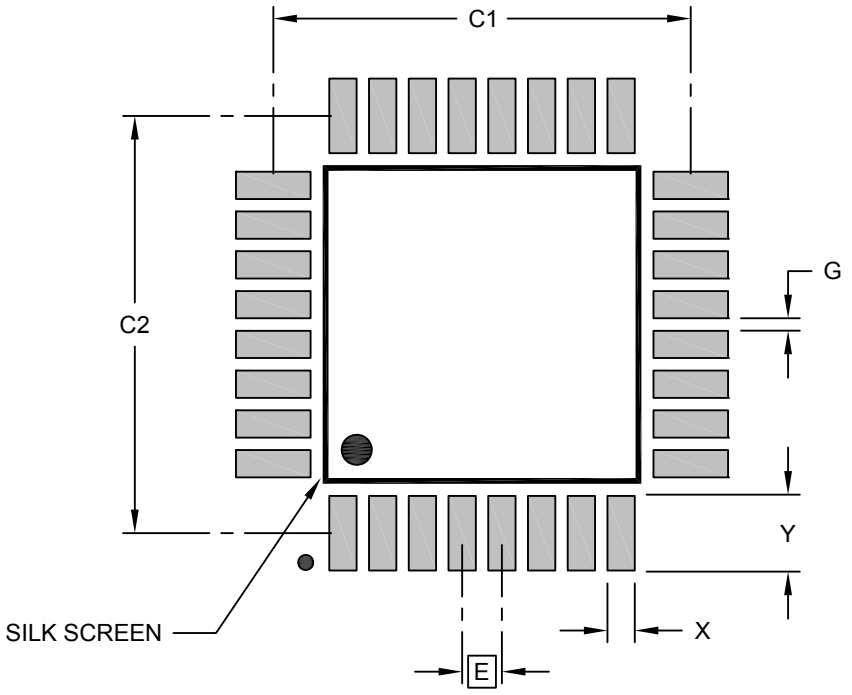
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-074 Rev C Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Thin Plastic Quad Flatpack (PT) - 7x7 mm Body [TQFP]
## 2.00 mm Footprint; Also Atmel Legacy Global Package Code AUT

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.80 BSC | |
| Contact Pad Spacing | C1 | | 8.40 | |
| Contact Pad Spacing | C2 | | 8.40 | |
| Contact Pad Width (Xnn) | X | | | 0.55 |
| Contact Pad Length (Xnn) | Y | | | 1.55 |
| Contact Pad to Contact Pad (Xnn) | G | 0.25 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2074 Rev C

**Table 52-20.** Device and Package Maximum Weight

| | |
|---|---|
| 100 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-21.** Package Reference

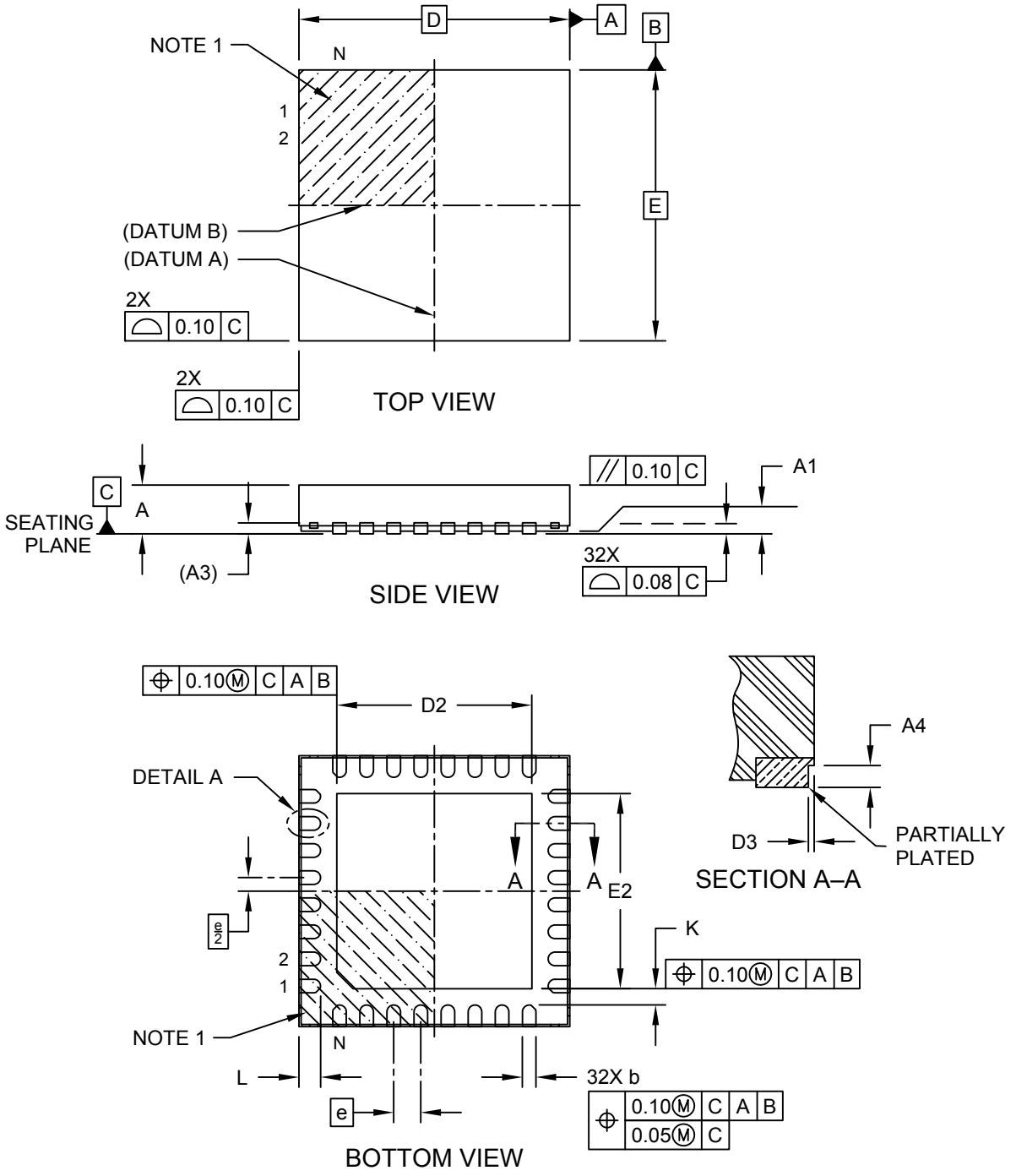
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-00074 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.11. 32-Pin VQFN AEC - Q100

**32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

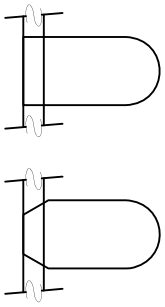



Microchip Technology Drawing C04-21391 Rev E Sheet 1 of 2

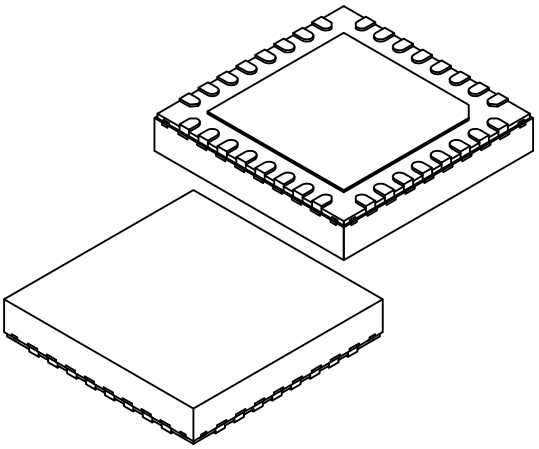

© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



DETAIL 1
ALTERNATE TERMINAL CONFIGURATIONS



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 32 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.035 | 0.05 |
| Terminal Thickness | A3 | | 0.203 REF | |
| Overall Length | D | | 5.00 BSC | |
| Exposed Pad Length | D2 | 3.50 | 3.60 | 3.70 |
| Overall Width | E | | 5.00 BSC | |
| Exposed Pad Width | E2 | 3.50 | 3.60 | 3.70 |
| Terminal Width | b | 0.20 | 0.25 | 0.30 |
| Terminal Length | L | 0.35 | 0.40 | 0.45 |
| Terminal-to-Exposed-Pad | K | 0.20 | - | - |
| Wettable Flank Step Cut Width | D3 | - | - | 0.085 |
| Wettable Flank Step Cut Depth | A4 | 0.10 | - | 0.19 |

Dimensions D3 and A4 above apply to all new products released after November 1, and all products shipped after January 1, 2019, and supersede dimensions D3 and A4 below.
No physical changes are being made to any package; this update is to align cosmetic and tolerance variations from existing suppliers.

| | | | | |
|---|---|---|---|---|
| Wettable Flank Step Length | D3 | 0.035 | 0.06 | 0.085 |
| Wettable Flank Step Height | A4 | 0.10 | - | 0.19 |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

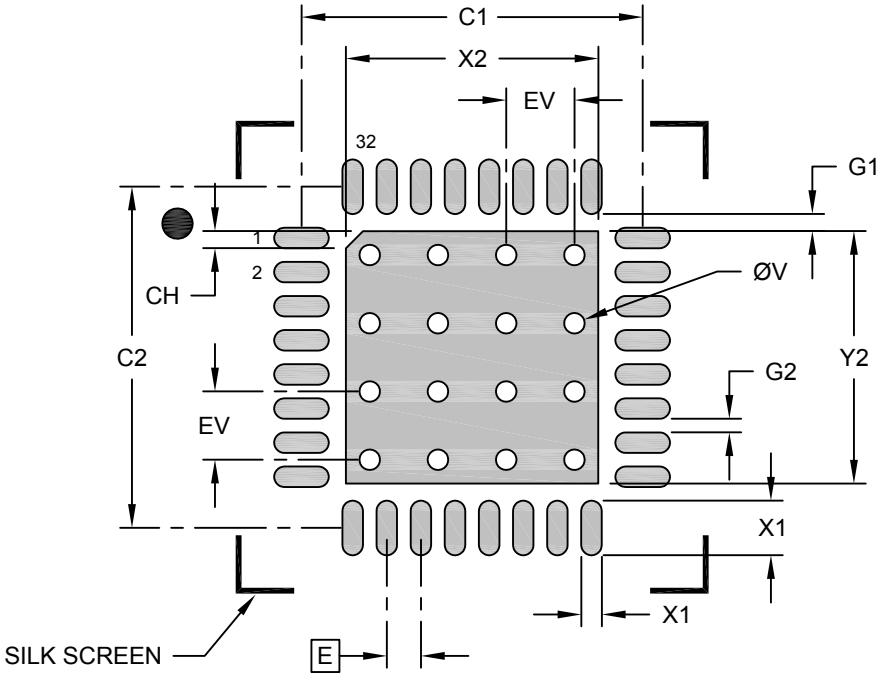
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21391 Rev E Sheet 2 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Very Thin Plastic Quad Flat, No Lead Package (RTB) - 5x5 mm Body [VQFN] With 3.6x3.6 mm Exposed Pad and Stepped Wettable Flanks; Atmel Legacy ZBS

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 3.70 |
| Optional Center Pad Length | Y2 | | | 3.70 |
| Exposed Pad 45° Corner Chamfer | CH | | 0.25 | |
| Contact Pad Spacing | C1 | | 5.00 | |
| Contact Pad Spacing | C2 | | 5.00 | |
| Contact Pad Width (X32) | X1 | | | 0.30 |
| Contact Pad Length (X32) | Y1 | | | 0.80 |
| Contact Pad to Center Pad (X32) | G1 | 0.25 | | |
| Contact Pad to Contact Pad (X28) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.30 | |
| Thermal Via Pitch | EV | | 1.00 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23391 Rev. E

**Table 52-22.** Device and Package Maximum Weight

| | |
|---|---|
| 67.7 | mg |


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-23.** Package Reference

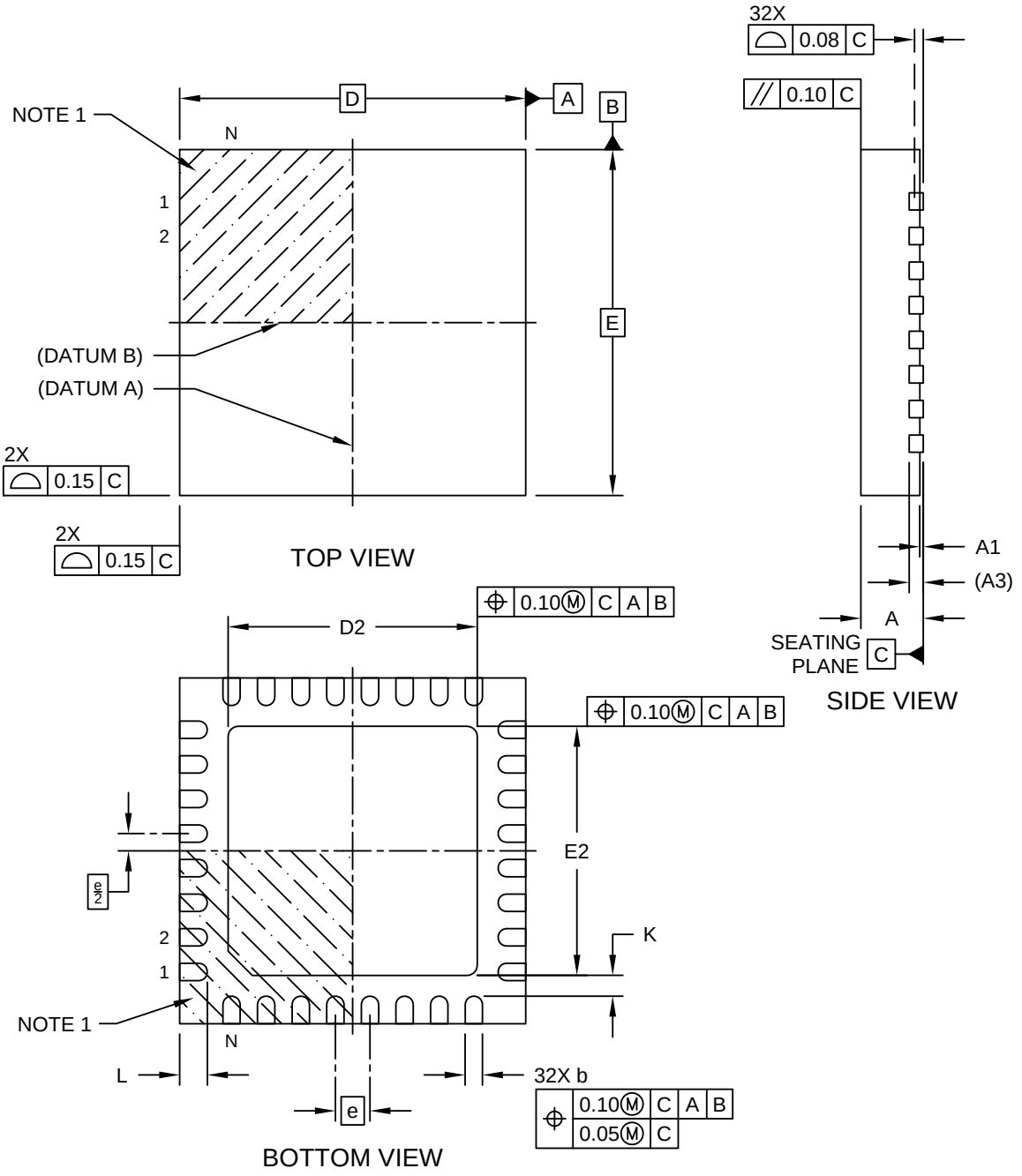
| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21391 |
| JESD97 Classification | E3 |


© 2026 Microchip Technology Inc. and its subsidiaries

### 52.3.12. 32-Pin VQFN

**32-Lead Very Thin Plastic Quad Flat, No Lead Package (S8B) - 5x5 mm Body [VQFN] With 3.60 mm Exposed Pad; Atmel Legacy Global Package Code ZKV**

| **Note:** | For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging |
|---|---|

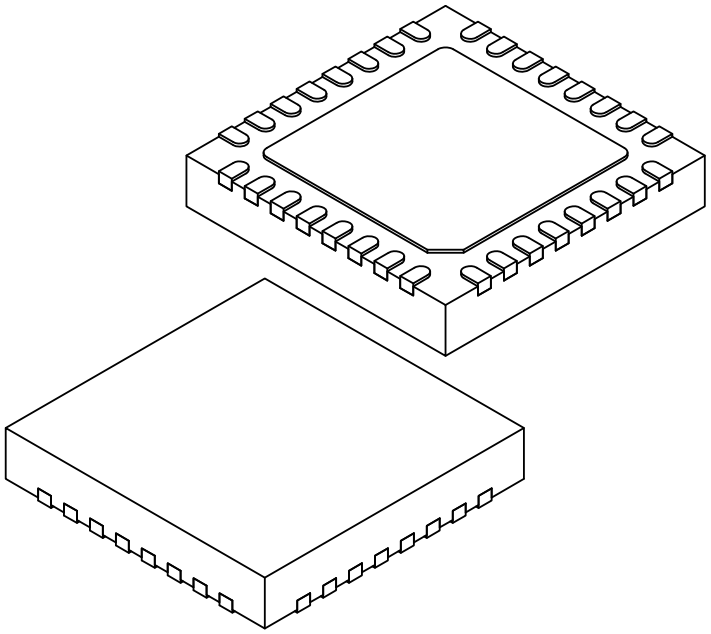



Microchip Technology Drawing C04-21402 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Very Thin Plastic Quad Flat, No Lead Package (S8B) - 5x5 mm Body [VQFN] With 3.60 mm Exposed Pad; Atmel Legacy Global Package Code ZKV

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Terminals | N | | 32 | |
| Pitch | e | | 0.50 BSC | |
| Overall Height | A | 0.80 | 0.90 | 1.00 |
| Standoff | A1 | 0.00 | 0.02 | 0.05 |
| Terminal Thickness | A3 | | 0.20 REF | |
| Overall Length | D | | 5.00 BSC | |
| Exposed Pad Length | D2 | 3.50 | 3.60 | 3.70 |
| Overall Width | E | | 5.00 BSC | |
| Exposed Pad Width | E2 | 3.50 | 3.60 | 3.70 |
| Terminal Width | b | 0.18 | 0.25 | 0.30 |
| Terminal Length | L | 0.30 | 0.40 | 0.50 |
| Terminal-to-Exposed-Pad | K | 0.20 | - | - |

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

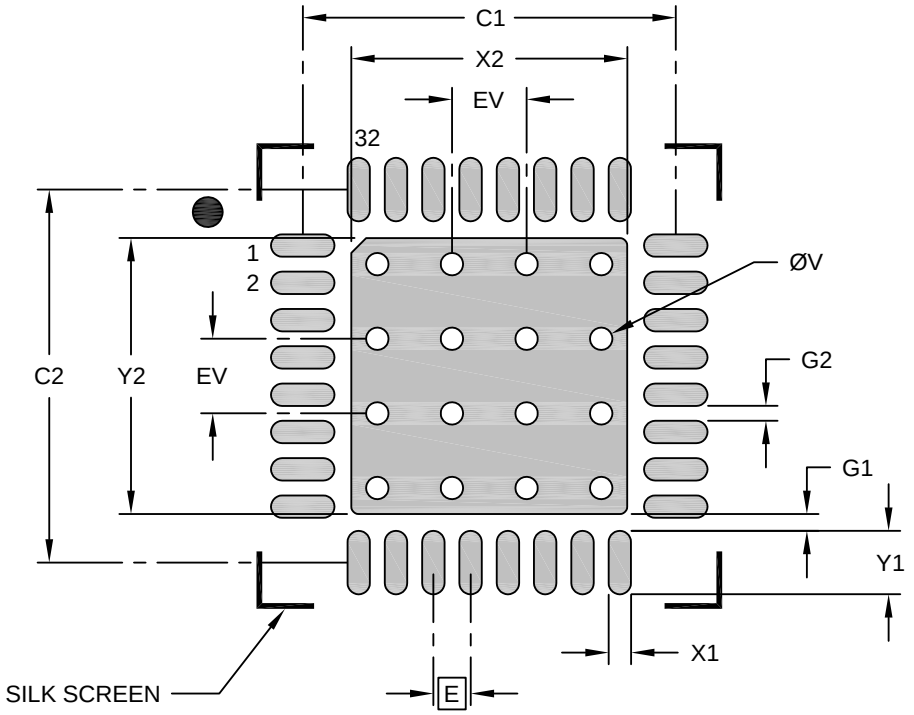
   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-21402 Rev A Sheet 1 of 2


© 2026 Microchip Technology Inc. and its subsidiaries

## 32-Lead Very Thin Plastic Quad Flat, No Lead Package (S8B) - 5x5 mm Body [VQFN] With 3.60 mm Exposed Pad; Atmel Legacy Global Package Code ZKV

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | | 0.50 BSC | |
| Optional Center Pad Width | X2 | | | 3.70 |
| Optional Center Pad Length | Y2 | | | 3.70 |
| Contact Pad Spacing | C1 | | 5.00 | |
| Contact Pad Spacing | C2 | | 5.00 | |
| Contact Pad Width (X32) | X1 | | | 0.30 |
| Contact Pad Length (X32) | Y1 | | | 0.85 |
| Contact Pad to Center Pad (X32) | G1 | 0.23 | | |
| Contact Pad to Contact Pad (X28) | G2 | 0.20 | | |
| Thermal Via Diameter | V | | 0.30 | |
| Thermal Via Pitch | EV | | 1.00 | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.
2. For best soldering results, thermal vias, if used, should be filled or tented to avoid solder loss during reflow process

Microchip Technology Drawing C04-23402 Rev A

**Note:** The exposed die attach pad is connected inside the device to GND and GNDANA.


© 2026 Microchip Technology Inc. and its subsidiaries

**Table 52-24.** Device and Package Maximum Weight

| | |
|---|---|
| 90 | mg |

**Table 52-25.** Package Reference

| | |
|---|---|
| Package Outline Drawing MCHP reference | C04-21402 |
| JESD97 Classification | E3 |

## 52.4. Soldering Profile

The following table gives the recommended soldering profile from J-STD-20.

**Table 52-26.** Recommended Soldering Profile

| Profile Feature | Green Package |
|---|---|
| Average Ramp-up Rate (217°C to peak) | 3°C/s max. |
| Preheat Temperature 175°C ±25°C | 150-200°C |
| Time Maintained Above 217°C | 60-150s |
| Time within 5°C of Actual Peak Temperature | 30s |
| Peak Temperature Range | 260°C |
| Ramp-down Rate | 6°C/s max. |
| Time 25°C to Peak Temperature | 8 minutes max. |

A maximum of three reflow passes is allowed per component.


© 2026 Microchip Technology Inc. and its subsidiaries

# 53. Schematic Checklist

## 53.1. Introduction

This chapter describes a common checklist which should be used when starting and reviewing the schematics for a SAM C20/C21 design. This chapter illustrates the recommended power supply connections, how to connect external analog references, programmer, debugger, oscillator and crystal.

**Note:** Refer to the "Basic 32-Bit MCU Design and Troubleshooting Checklist" guide on the Microchip web site, www.microchip.com.

## 53.2. Operation in Noisy Environment

If the device is operating in an environment with much electromagnetic noise it must be protected from this noise to ensure reliable operation. In addition to following best practice EMC design guidelines, the recommendations listed in the schematic checklist sections must be followed. In particular placing decoupling capacitors very close to the power pins, a RC-filter on the $\overline{\text{RESET}}$ pin, and a pull-up resistor on the SWCLK pin is critical for reliable operations. It is also relevant to eliminate or attenuate noise in order to avoid that it reaches supply pins, I/O pins and crystals.

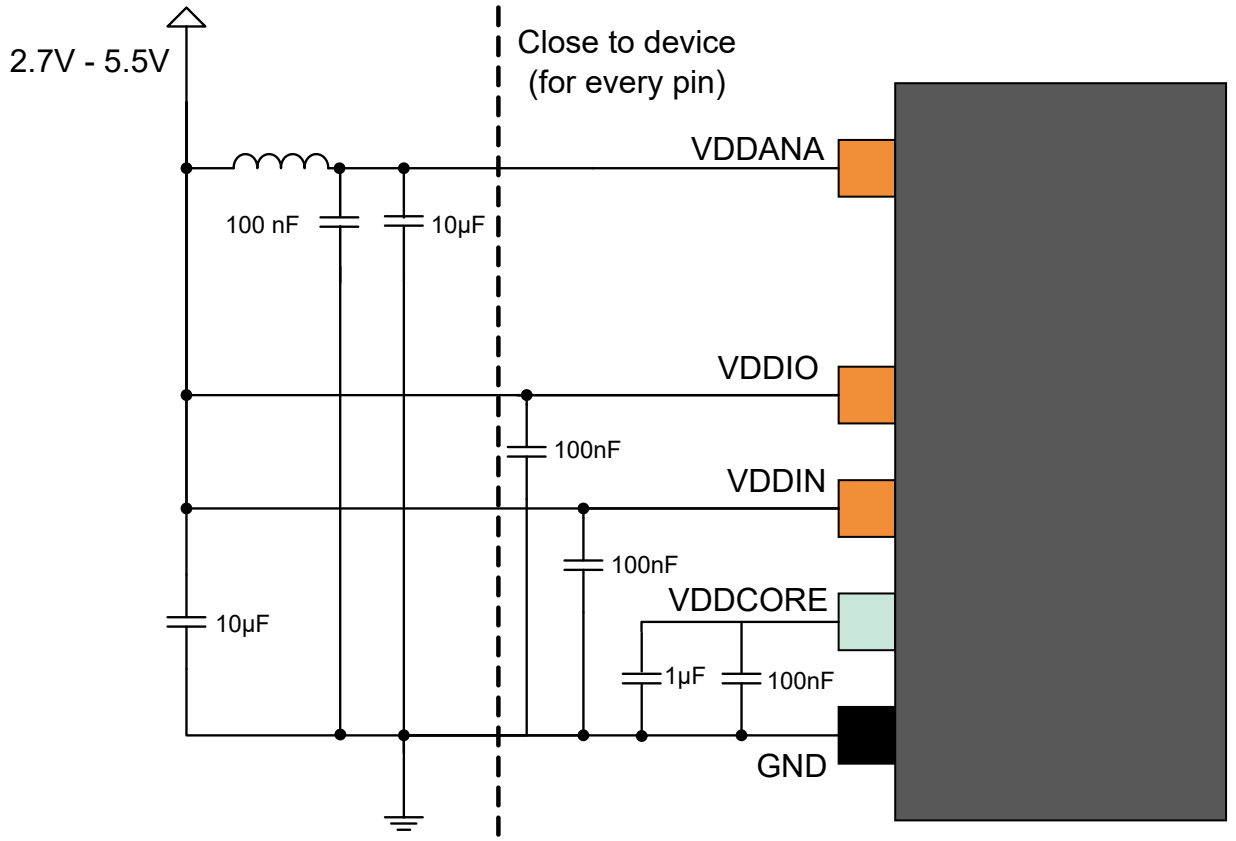
Refer to the "EMI, EMC, EFT and ESD Circuit Design Consideration for 32-bit Microcontrollers" guide on the Microchip web site, www.microchip.com.

## 53.3. Power Supply

The SAM C20/C21 supports a single power supply or dual power supplies from 2.7 to 5.5V.

### 53.3.1. Power Supply Connections

**Figure 53-1.** Single Power Supply Schematic





© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-2.** Dual Power Supply Schematic

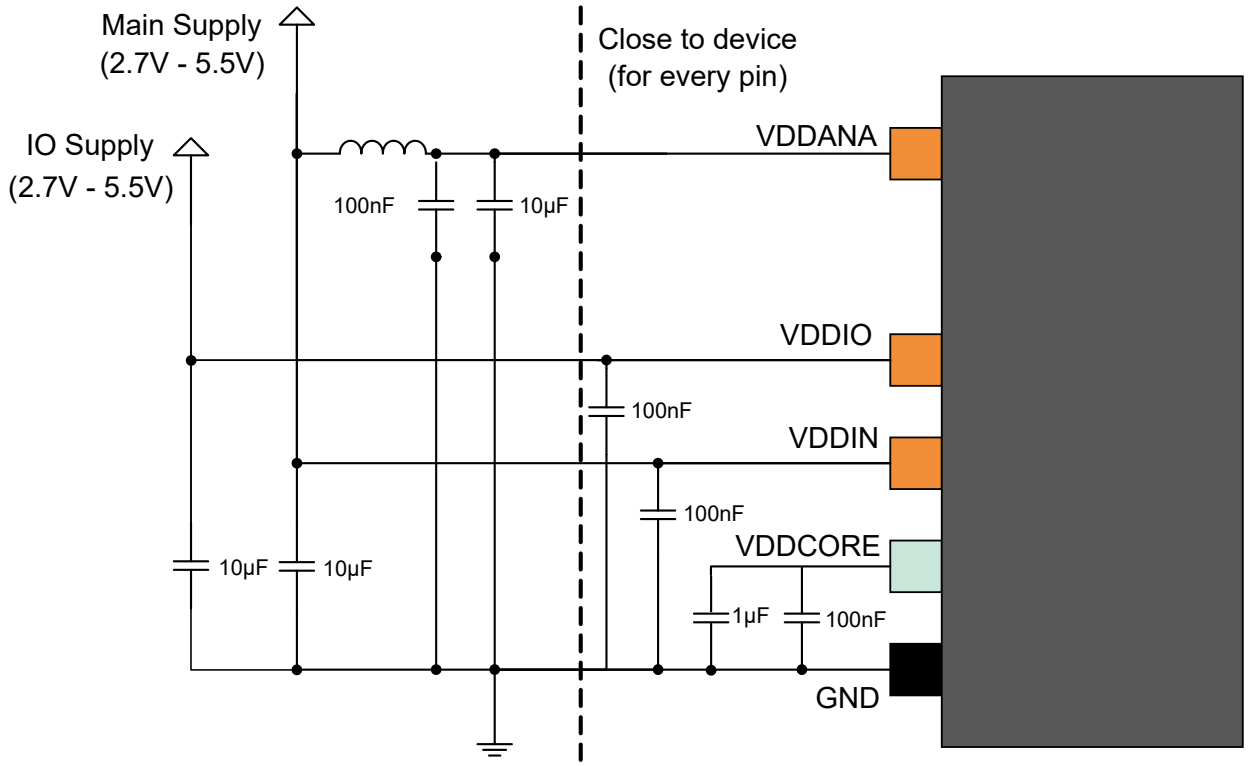



**Table 53-1.** Power Supply Connections, $V_{DDCORE}$ From Internal Regulator

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| $V_{DDIO}$ | 2.7V to 5.5V<br>Decoupling/filtering capacitors 100nF(1)(2) and 10µF(1)<br>Decoupling/filtering inductor 10µH(1)(3) | I/O supply voltage |
| $V_{DDANA}$ | 2.7V to 5.5V<br>Decoupling/filtering capacitors 100nF(1)(2) and 10µF(1)<br>Ferrite bead(4) prevents the $V_{DD}$ noise interfering with $V_{DDANA}$ | Analog supply voltage |
| $V_{DDIN}$ | 2.7V to 5.5V<br>Decoupling/filtering capacitors 100nF(1)(2) and 10µF(1)<br>Decoupling/filtering inductor 10µH(1)(3) | Digital supply voltage |
| $V_{DDCORE}$ | 1.1V to 1.3V typical<br>Decoupling/filtering capacitors 100nF(1)(2)and 1µF(1) | Core supply voltage / external decoupling pin |
| GND | | Ground |
| $GND_{ANA}$ | | Ground for the analog power domain |

1. These values are only given as a typical example.

2. Decoupling capacitors should be placed close to the device for each supply pin pair in the signal group, low ESR capacitors should be used for better decoupling.

3. An inductor should be added between the external power and the $V_{DD}$ for power filtering.

4. A ferrite bead has better filtering performance compared to standard inductor at high frequencies. A ferrite bead can be added between the main power supply ($V_{DD}$) and $V_{DDANA}$ to prevent digital noise from entering the analog power domain. The bead should provide enough impedance (e.g. 50Ω at 20MHz and 220Ω at 100MHz) to separate the digital and analog power


© 2026 Microchip Technology Inc. and its subsidiaries

domains. Make sure to select a ferrite bead designed for filtering applications with a low DC resistance to avoid a large voltage drop across the ferrite bead.

## 53.4. External Analog Reference Connections

The following schematic checklist is only necessary if the application is using one or more of the external analog references. If the internal references are used instead, the following circuits are not necessary.

**Figure 53-3.** External Analog Reference Schematic With Two References

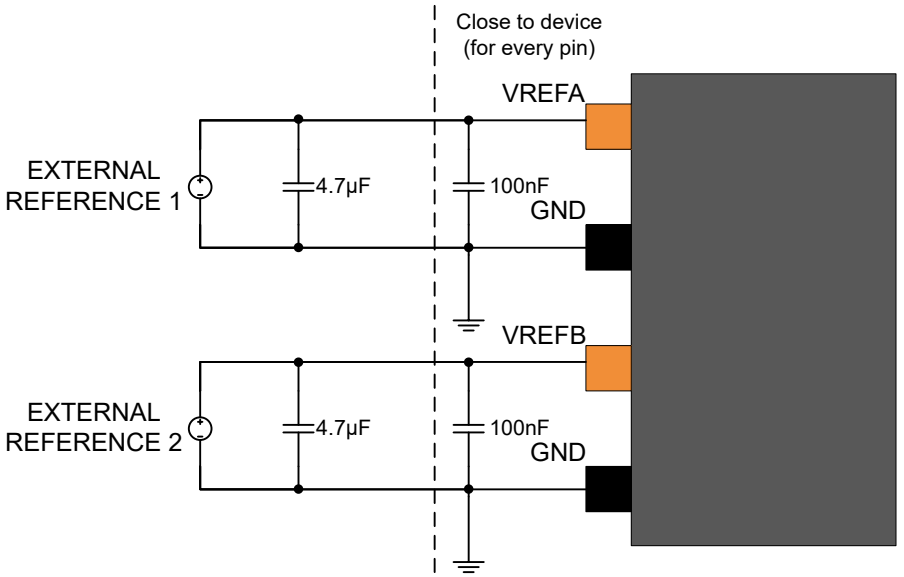
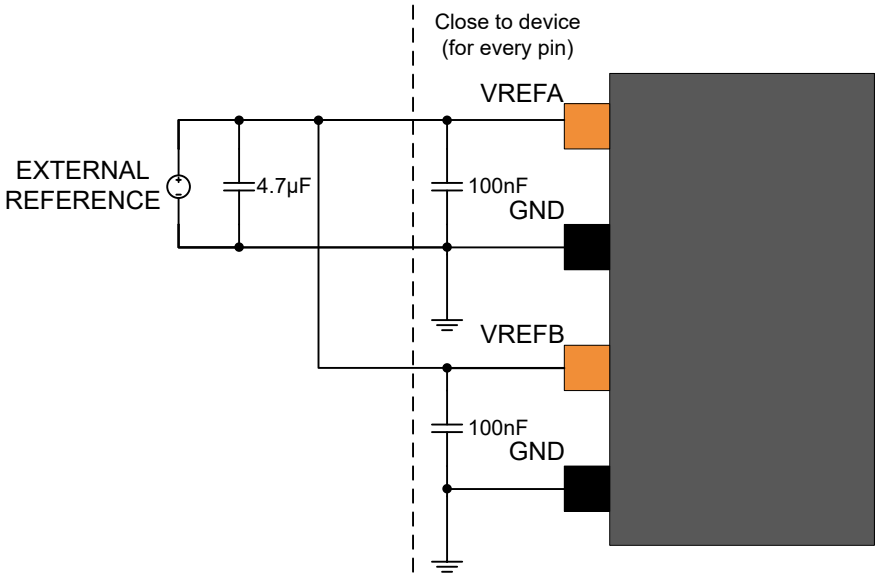



**Figure 53-4.** External Analog Reference Schematic With One Reference





© 2026 Microchip Technology Inc. and its subsidiaries

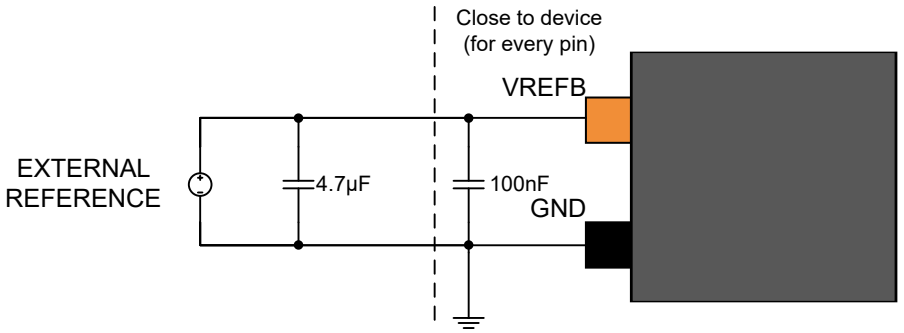



**Table 53-2.** External Analog Reference Connections

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| VREFA | 2.0V to $V_{DDANA}$ - 0.6V for ADC<br>1.0V to $V_{DDANA}$ - 0.6V for DAC<br>Decoupling/filtering capacitors: 100nF(1)(2) and 4.7µF(1) | External reference from VREFA pin on the analog port. |
| VREFB | 1.0V to 5.5V for SDADC<br>Decoupling/filtering capacitors: 100nF(1)(2) and 4.7µF(1) | External reference from VREFB pin on the analog port. |
| GND | | Ground |

**Notes:**

1. These values are given as a typical example.
2. Decoupling capacitor should be placed close to the device for each supply pin pair in the signal group.

## 53.5. External Reset Circuit

The external Reset circuit is connected to the $\overline{\text{RESET}}$ pin when the external Reset function is used. The circuit is not necessary when the $\overline{\text{RESET}}$ pin is not driven low externally by the application circuitry.

The reset switch can also be removed, if a manual reset is not desired. The $\overline{\text{RESET}}$ pin itself has an internal pull-up resistor, therefore it is optional to add any external pull-up resistor.

A pull-up resistor makes sure that the reset does not go low and unintentionally causing a device reset. An additional resistor has been added in series with the switch to safely discharge the filtering capacitor, that is, preventing a current surge when shorting the filtering capacitor which again can cause a noise spike that can have a negative effect on the system.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-5.** External Reset Circuit Schematic

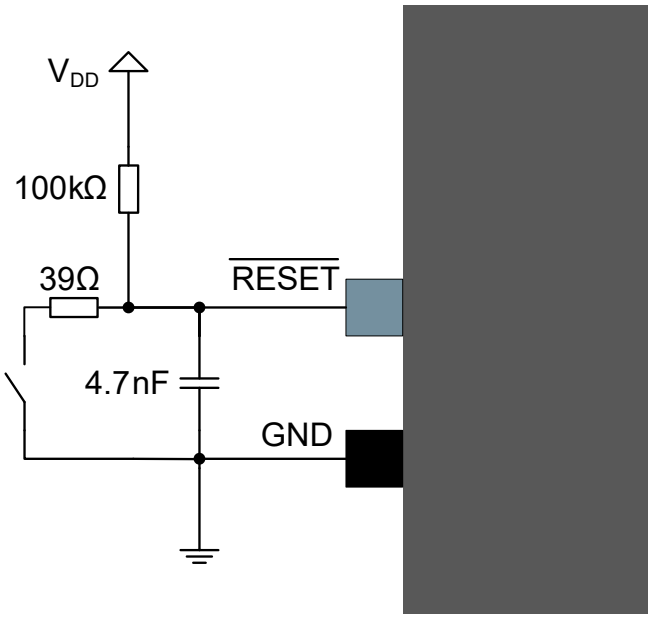
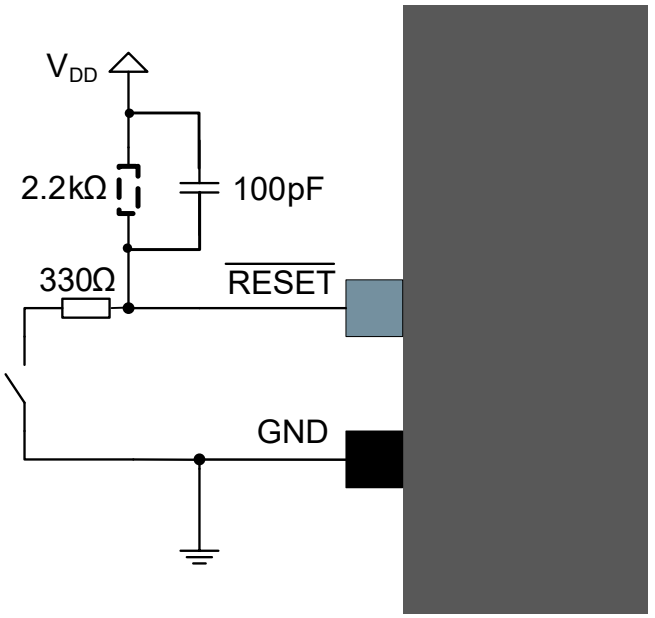



**Figure 53-6.** External Reset Circuit Schematic (EFT Immunity Enhancement)




**Note:** This reset circuit is intended to improve EFT immunity, but does not filter low-frequency glitches, which makes it not suitable as an example for applications requiring debouncing on a reset button.

**Table 53-3.** Reset Circuit Connections

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| $\overline{\text{RESET}}$ | Reset low level threshold voltage<br>$V_{DDIO}$ = 2.7V - 5.5V: Below 0.3 * $V_{DDIO}$<br>Decoupling/filter capacitor 100 pF(1)<br>Pull-up resistor 2.2 kΩ(1)(2)<br>Resistor in series with the switch 330Ω(1) | Reset pin |


© 2026 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. These values are only given as a typical example.
2. The SAM C20/C21 features an internal pull-up resistor on the $\overline{\text{RESET}}$ pin; therefore, an external pull-up is optional.

**Note:** Reset circuit recommendations to improve EFT/ESD/EMI immunity can be found in the "Basic 32-Bit MCU Design and Troubleshooting Checklist" guide on the Microchip web site, www.microchip.com.

## 53.6. Unused or Unconnected Pins

Unused or unconnected pins (unless marked as NC where applicable) should not be left unconnected and floating. Floating pins will add to the overall power consumption of the device. To prevent this one should always draw the pin voltage towards a given level, either VDD or GND, through a pull up/down resistor. External or internal pull up/down resistors can be used, e.g. the pins can be configured in pull-up or pull-down mode eliminating the need for external components. There are no obvious benefit in choosing external vs. internal pull resistors.

**Related Links**

PORT - I/O Pin Controller

## 53.7. Clocks and Crystal Oscillators

The SAM C20/C21 can be run from internal or external clock sources, or a mix of internal and external sources. An example of usage will be to use the internal 48MHz oscillator as source for the system clock, and an external 32.768 kHz watch crystal as clock source for the Real-Time counter (RTC).

### 53.7.1. External Clock Source

**Figure 53-7.** External Clock Source Schematic

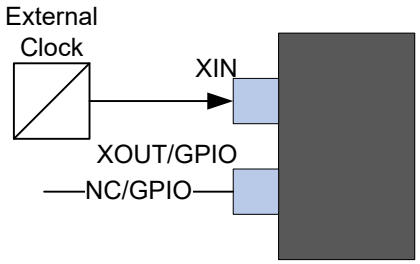



**Table 53-4.** External Clock Source Connections

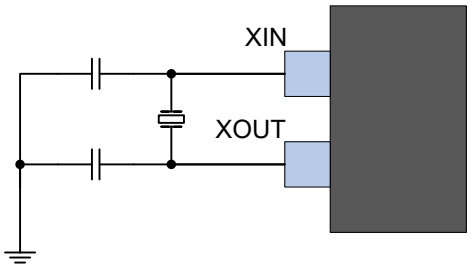
| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| XIN | XIN is used as input for an external clock signal | Input for inverting oscillator pin |
| XOUT/GPIO | Can be left unconnected or used as normal GPIO | NC/GPIO |

### 53.7.2. Crystal Oscillator

The crystal oscillator schematic is shown below:


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-8.** Crystal Oscillator Schematic




The crystal must be located as close to the device as possible. Long-signal lines may cause too high of a load to operate the crystal, and cause crosstalk to other parts of the system.

**Table 53-5.** Crystal Oscillator Checklist

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| XIN | Load capacitor. Value depends on Crystal manufacturers recommendation.(1) | External crystal between 0.4 to 32 MHz |
| XOUT | Load capacitor. Value depends on Crystal manufacturers recommendation.(1) | |

**Note:**

1. The capacitors must be placed close to the device for each supply pin pair in the signal group.

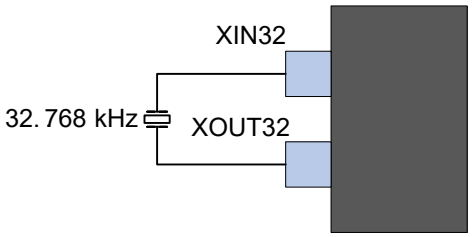
### 53.7.3. External Real Time Oscillator

The low-frequency crystal oscillator is optimized for use with a 32.768 kHz watch crystal. When selecting crystals, load capacitance and the crystal's Equivalent Series Resistance (ESR) must be considered. Both values are specified by the crystal vendor.

The SAM C20/C21 oscillator is optimized for very low-power consumption, hence close attention must be paid when selecting crystals.

The typical parasitic load capacitance values are available in the Electrical Characteristics section. This capacitance and PCB capacitance can allow using a crystal inferior to 12.5 pF load capacitance without external capacitors as shown in Figure 53-9.

**Figure 53-9.** External Real Time Oscillator without Load Capacitor




To improve accuracy and safety factor, the crystal data sheet recommend adding external capacitors as shown in Figure 53-10.

To find suitable load capacitance for a 32.768 kHz crystal, consult the crystal data sheet.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-10.** External Real Time Oscillator with Load Capacitor

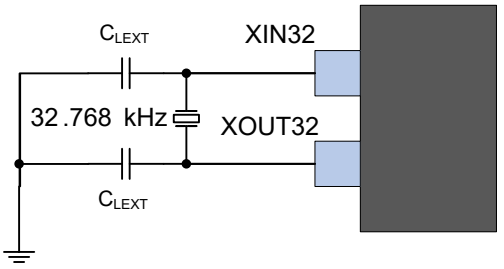



**Table 53-6.** External Real Time Oscillator Checklist

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| XIN32 | Load capacitor 22pF(1)(2) | Timer oscillator input |
| XOUT32 | Load capacitor 22pF(1)(2) | Timer oscillator output |

1. These values are only given as typical examples.

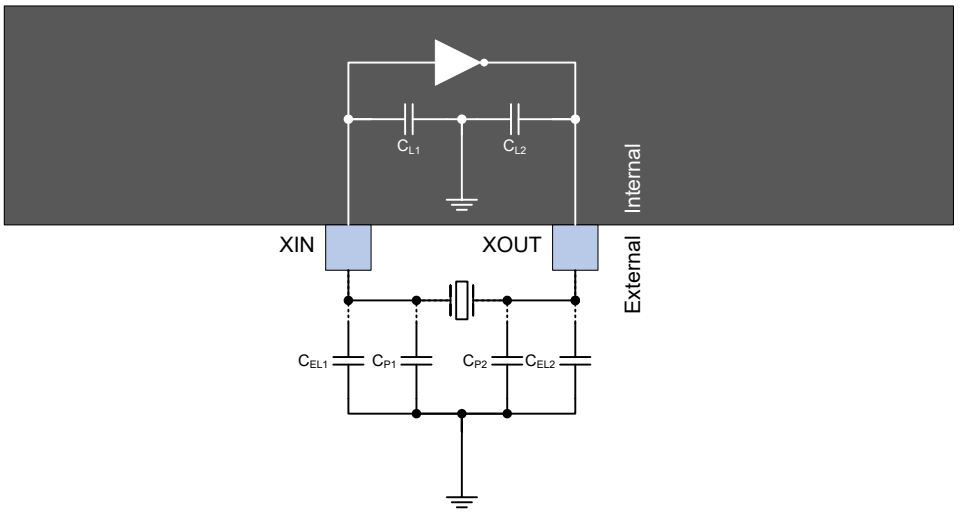
2. The capacitors must be placed close to the device for each supply pin pair in the signal group.

**Note:** In order to minimize the cycle-to-cycle jitter of the external oscillator, keep the neighboring pins as steady as possible. For neighboring pin details, refer to the section “Oscillator Pinout”.

### 53.7.4. Calculating the Correct Crystal Decoupling Capacitor

The model shown in Figure 53-11 can be used to calculate correct load capacitor for a given crystal. This model includes internal capacitors $C_{Ln}$, external parasitic capacitance $C_{ELn}$ and external load capacitance $C_{Pn}$.

**Figure 53-11.** Crystal Circuit With Internal, External and Parasitic Capacitance




Using this model, the total capacitive load for the crystal can be calculated as shown in the equation below:

$$\sum C_{\text{tot}} = \frac{(C_{L1} + C_{P1} + C_{\text{EL1}})(C_{L2} + C_{P2} + C_{\text{EL2}})}{C_{L1} + C_{P1} + C_{\text{EL1}} + C_{L2} + C_{P2} + C_{\text{EL2}}}$$

where:

$C_{tot}$ is the total load capacitance seen by the crystal. This value must be equal to the load capacitance value found in the crystal data sheet.


© 2026 Microchip Technology Inc. and its subsidiaries

The parasitic capacitance $C_{ELn}$ can be in most applications be disregarded as these are usually very small. If accounted for, these values are dependent on the PCB material and PCB layout.

For some crystal, the internal capacitive load provided by the device itself can be enough. To calculate the total load capacitance in this case. $C_{ELn}$ and $C_{Pn}$ are both zero, $C_{L1} = C_{L2} = C_L$, and the equation reduces to the following:

$$\sum C_{\text{tot}} = \frac{C_L}{2}$$

For equivalent internal pin capacitance values, refer to the section Crystal Oscillator Characteristics.

**Related Links**
Crystal Oscillator Characteristics

## 53.8. Programming and Debug Ports

For programming and debugging the SAM C20/C21 device, it must be connected using the Serial Wire Debug (SWD) interface. Currently the SWD interface is supported by several Microchip and third party programmers and debuggers, such as SAM-ICE and JTAGICE3.

Refer to the SAM-ICE and JTAGICE3 user's guides for details on debugging and programming connections and options. For connecting to any other programming or debugging tool, refer to the specific programmer or debugger user's guide.

A pull-up resistor on the SWCLK pin is critical for reliable operations. Refer to related link for additional information.

**Figure 53-12.** SWCLK Circuit Connections

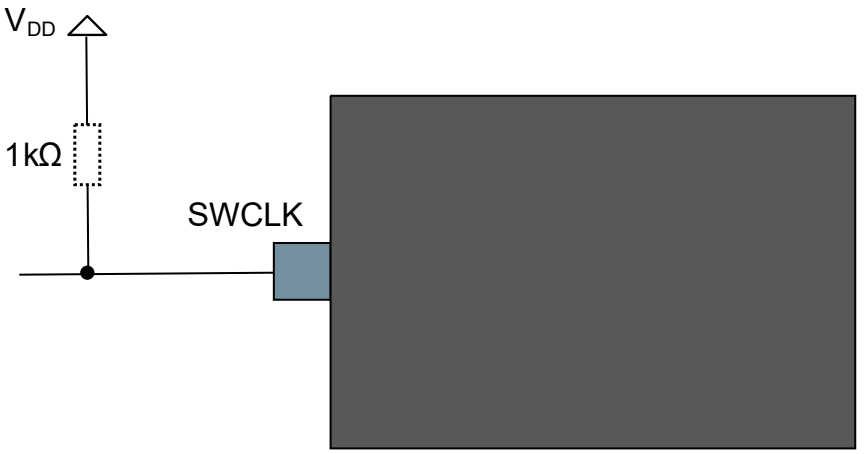



**Table 53-7.** SWCLK Circuit Connections

| Pin Name | Description | Recommended Pin Connection |
|---|---|---|
| SWCLK | Serial wire clock pin | Pull-up resistor 1kΩ recommended |

**Related Links**
Operation in Noisy Environment

### 53.8.1. Cortex Debug Connector (10-pin)

For debuggers and programmers that support the Cortex Debug Connector (10-pin) interface the signals should be connected as shown in the following figure with details described in the corresponding table.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-13.** Cortex Debug Connector (10-pin)

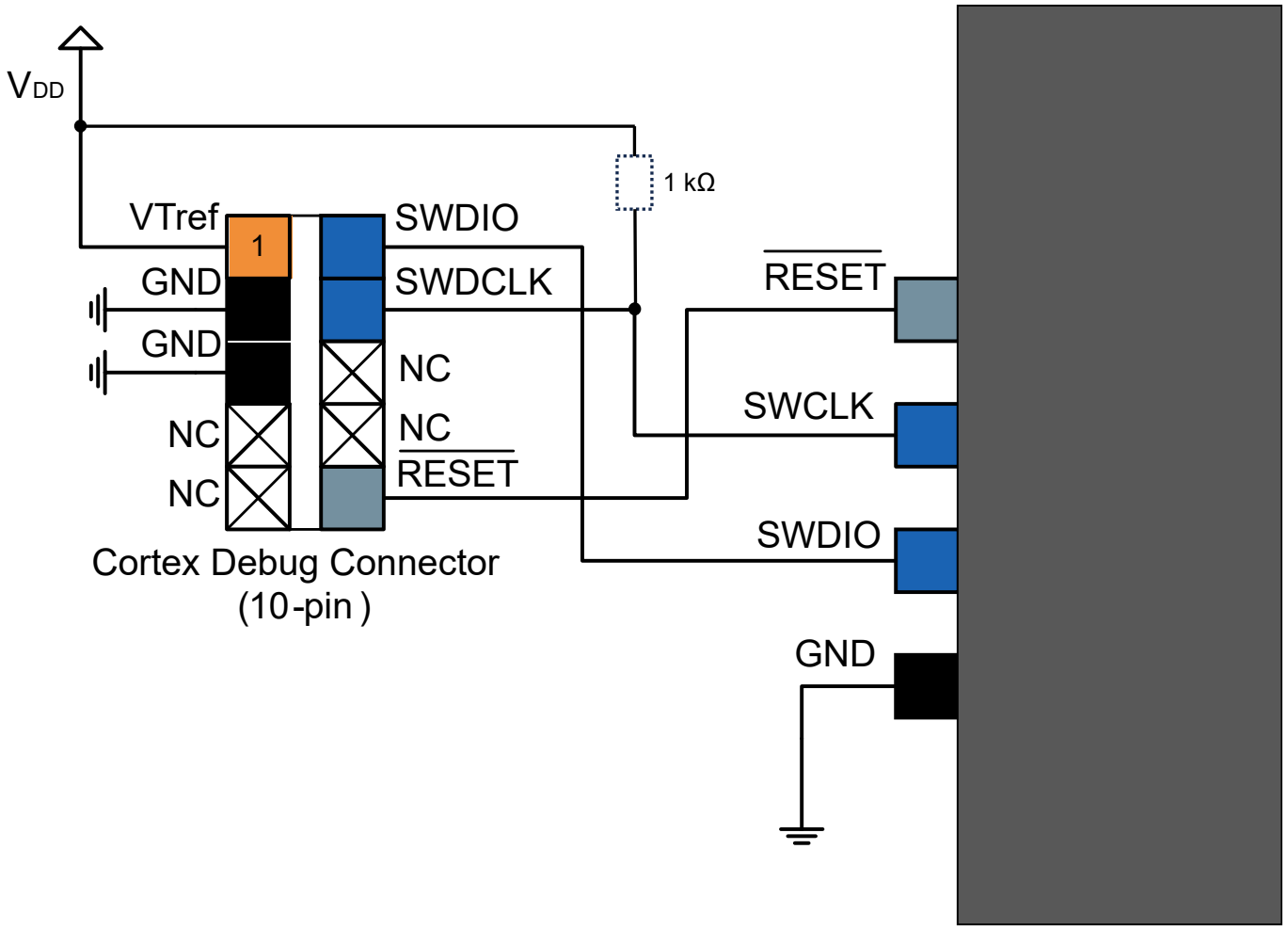



**Table 53-8.** Cortex Debug Connector (10-pin)

| Header Signal Name | Description |
|---|---|
| SWDCLK | Serial wire clock pin |
| SWDIO | Serial wire bidirectional data pin |
| $\overline{\text{RESET}}$ | Target device reset pin, active low |
| VTref | Target voltage sense, must be connected to the device $V_{DD}$ |
| GND | Ground |

**Note:** JTAG and Serial Wire Interface recommendations can be found in the "Basic 32-Bit MCU Design and Troubleshooting Checklist" guide on the Microchip web site, www.microchip.com.

### 53.8.2. 10-pin JTAGICE3 Compatible Serial Wire Debug Interface

The JTAGICE3 debugger and programmer does not support the Cortex Debug Connector (10-pin) directly, hence a special pinout is needed to directly connect the SAM C20/C21 to the JTAGICE3, alternatively one can use the JTAGICE3 squid cable and manually match the signals between the JTAGICE3 and SAM C20/C21. The following figure describes how to connect a 10-pin header that support connecting the JTAGICE3 directly to the SAM C20/C21 without the need for a squid cable. This can also be used for the Atmel-ICE AVR connector port.


© 2026 Microchip Technology Inc. and its subsidiaries

The JTAGICE3 squid cable or the JTACICE3 50 mil cable can be used to connect the JTAGICE3 programmer and debugger to the SAM C20/C21. The following figure illustrates the correct pinout for the JTAGICE3 50 mil, and details are given in the table.

**Figure 53-14.** 10-pin JTAGICE3 Compatible Serial Wire Debug Interface

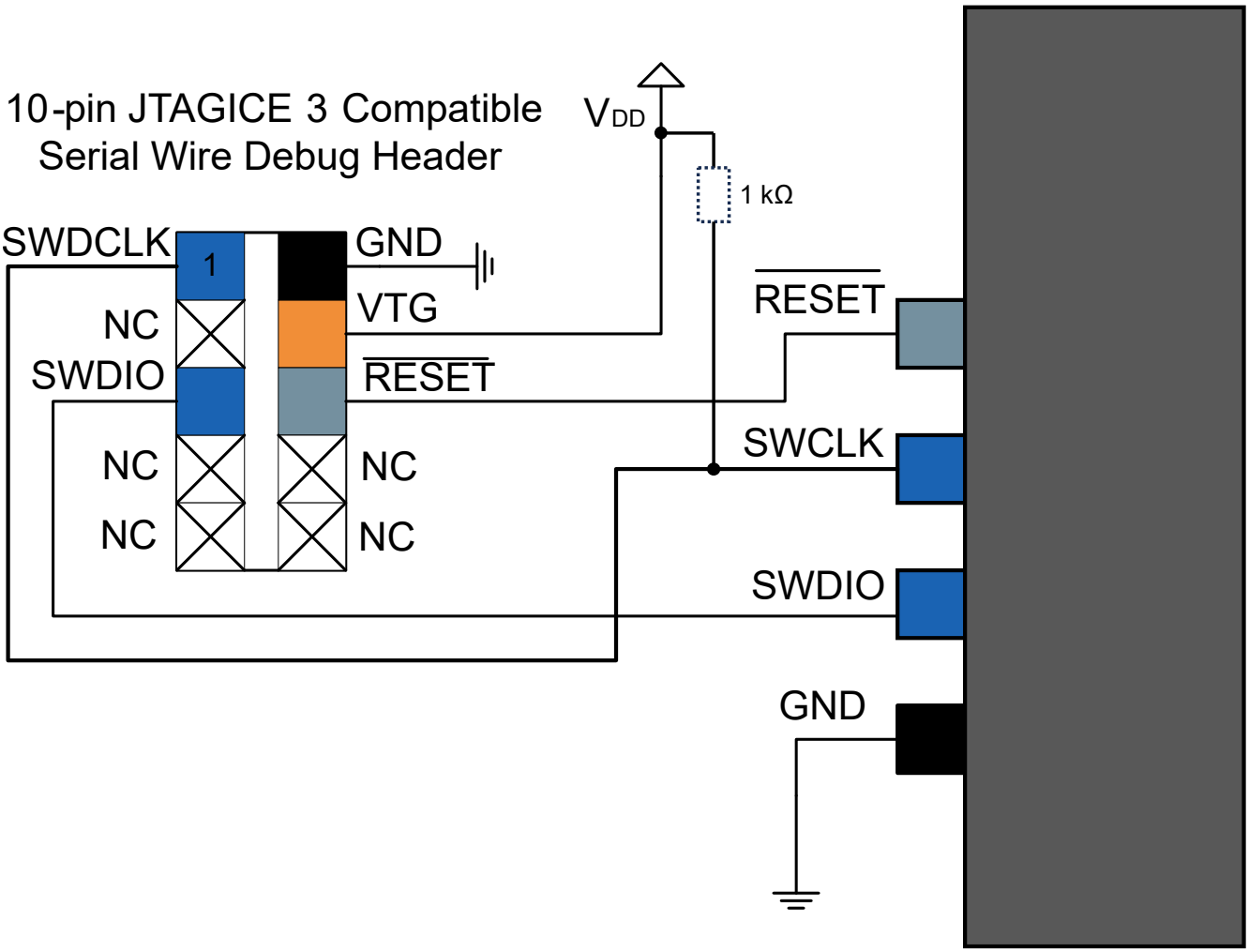



**Table 53-9.** 10-pin JTAGICE3 Compatible Serial Wire Debug Interface

| Header Signal Name | Description |
|---|---|
| SWDCLK | Serial wire clock pin |
| SWDIO | Serial wire bidirectional data pin |
| $\overline{\text{RESET}}$ | Target device reset pin, active low |
| VTG | Target voltage sense, must be connected to the device $V_{DD}$ |
| GND | Ground |

**Note:** JTAG and Serial Wire Interface recommendations can be found in the "Basic 32-Bit MCU Design and Troubleshooting Checklist" guide on the Microchip web site, www.microchip.com.

### 53.8.3. 20-pin IDC JTAG Connector

For debuggers and programmers that support the 20-pin IDC JTAG Connector, that is, the SAM-ICE, the signals must be connected as shown in the following figure with details described in the table.


© 2026 Microchip Technology Inc. and its subsidiaries

**Figure 53-15.** 20-pin IDC JTAG Connector

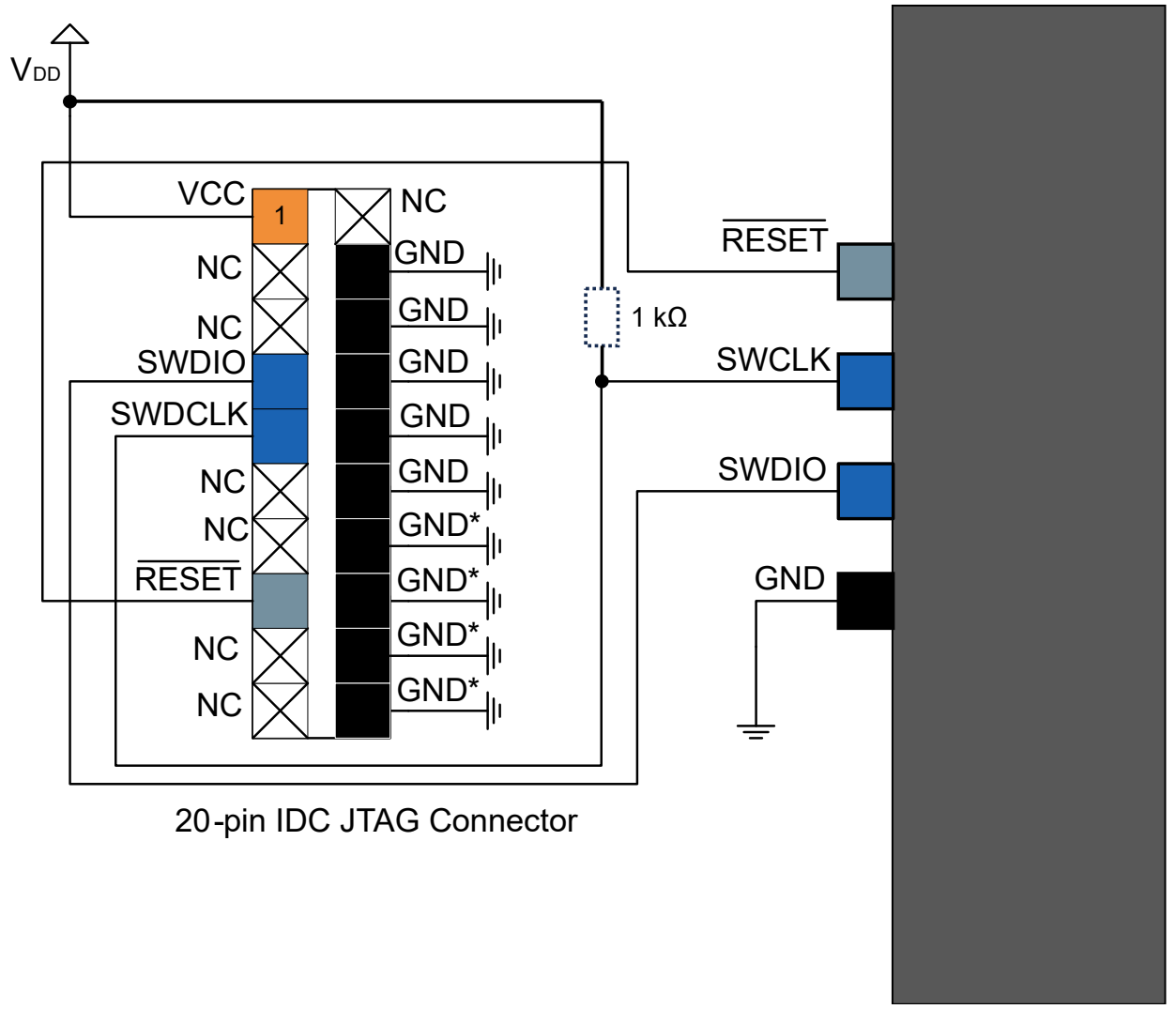



**Table 53-10.** 20-pin IDC JTAG Connector

| Header Signal Name | Description |
|---|---|
| SWDCLK | Serial wire clock pin |
| SWDIO | Serial wire bidirectional data pin |
| $\overline{\text{RESET}}$ | Target device reset pin, active low |
| VCC | Target voltage sense, should be connected to the device $V_{DD}$ |
| GND | Ground |
| GND* | These pins are reserved for firmware extension purposes. They can be left unconnected or connected to GND in normal debug environment. They are not essential for SWD in general. |

**Note:** JTAG and Serial Wire Interface recommendations can be found in the “Basic 32-Bit MCU Design and Troubleshooting Checklist” guide on the Microchip web site, www.microchip.com.


© 2026 Microchip Technology Inc. and its subsidiaries

# 54. Revision History

## 54.1. Revision History

### Revision N - 10/2026

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Ordering Information | • Added information for the VAO Automotive Qualification |

### Revision M - 04/2025

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Ordering Information | • Updated the OTP Lock information in the Diagram to *Program and Debug Interface Disable* |
| Device Service Unit (DSU) | • Added a new section, *Program and Debug Interface Disable (PDID)* |

### Revision L - 01/2025

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Ordering Information | Updated the figure to include the reference for OTP and added a note regarding where to find OTP information. |

### Revision K - 08/2023

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Peripheral Access Controller | Updated reset value of DSU Bit in STATUSB Register.<br>Updated reset value in INTFLAGAHB, INTFLAGA, INTFLAGB, INTFLAGC, INTFLAGD, STATUSA, STATUSB, STATUSC and STATUSD Registers. |
| Clock System | Updated Figure 15-1. |
| PM - Power Manager | Updated the table of VREGSMOD Bits in STDBYCFG Register. |
| OSCCTRL - Oscillators Controller | Updated Figure 20-2.<br>Added note in CLKFAIL Bit in STATUS Register. |
| OSC32KCTRL - 32.768 kHz Oscillators Controller | Added the fifth paragraph in 32.768 kHz Internal Oscillator (OSC32K) Operation. |
| RTC - Real-Time Counter | Updated Property field in COUNT Register. |
| DMAC - Direct Memory Access Controller | Changed *CLK_DMAC_APB* by *CLK_DMAC_AHB* in Clocks.<br>Changed (BEATSIZE + 1) by (BEATSIZE) in Addressing. |
| Nonvolatile Memory Controller (NVMCTRL) | Updated second paragraph in Memory Organization.<br>Updated the table of PSZ Bits in PARAM Register. |
| Event System (EVSYS) | Updated *User Multiplexer Number* Table with info of Users m=45 and m=46. Added note 1. |
| SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter | Updated second paragraph in Data Transmission.<br>Added note in LIN Host.<br>Added note in Sample Adjustment.<br>Added note in HDRDLY and BRKLEN Bits in CTRLC Register. |
| SERCOM SPI – SERCOM Serial Peripheral Interface | Updated first paragraph in Hardware Controlled SS.<br>Added note in MSSEN Bit in CTRLB Register. |


© 2026 Microchip Technology Inc. and its subsidiaries

**Revision History** (continued)

| Section | Updates |
|---|---|
| CAN - Control Area Network (SAM C21 Only) | Changed *CLK_CANx_APB* by *CLK_CANx_AHB* in Sleep Mode Operation and Synchronization.<br>Changed *CLK_CAN_APB* by *CLK_CAN_AHB* in CCCR Register, Standard Message ID Filter Element and Extended Message ID Filter Element.<br>Updated description in NBTP Register. |
| Timer/Counter (TC) | Added the fourth note in Capture Operations.<br>Updated Property field and added note in CAPTMODE1, CAPTMODE0, COPENx, CAPTENx, ALOCK, PRESCALER, ONDEMAND, RUNSTDBY, PRESCSYNC, MODE, ENABLE and SWRST Bits in CTRLA Register.<br>Added info of PER Bit in SYNCBUSY Register. |
| Timer/Counter for Control Applications (TCC) | Updated Property field and added note in CPTENx, DMAOS, MSYNC, ALOCK, PRESCYNC, RUNSTDBY, PRESCALER, RESOLUTION, ENABLE and SWRST Bits in CTRLA Register.<br>Updated Property field and added note in CTRLBCLR, STATUS, COUNT, CCx, and WAVEBUF Registers.<br>Added note in CTRLBSET and PER Registers.<br>Updated info of PGVx and PGEx Bits and added note in PATT Register.<br>Updated info of SWAPx, POLx and CICCENx Bits and added note in SWAPx, POLx, CICCENx, CIPEREN, RAMP and WAVEGEN Bits in WAVE Register.<br>Updated Property field, added note and updated info of PGVBx and PGEBx Bits in PATTBUF Register.<br>Updated Property field, added note and updated info of PERBUF and DITHERBUF Bits in PERBUF Register.<br>Updated Name, Offset and Property fields, added note and updated info of DITHERBUF Bits in CCBUFx Register. |
| ADC - Analog-to-Digital Converter | Removed an erroneous reference to DAC in Features, Figure 38-1 and Figure 38-8.<br>Updated second paragraph in Operation.<br>Updated Conversion Timing and Sampling Rate with new information after Figure 38-6.<br>Updated table of MUXPOS Bits in INPUTCTRL Register. |
| AC - Analog Comparators | Added note in Offset Compensation.<br>Updated table of MUXPOS and MUXNEG Bits in COMPCTRL Register. |
| DAC – Digital-to-Analog Converter (SAM C21 only) | Updated second paragraph in Principle of Operation.<br>Updated second bullet in Interrupts. |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | Updated Min. and Max. Values of $V_{PIN}$ in Table 45-1.<br>Updated Max. Values of IDLE2 in Table 45-11. Deleted note 2.<br>Updated $T_S$ in Table 45-22.<br>Updated $f_{OUT}$ in Table 45-52 and added note 3.<br>Updated Table 45-57. Added note 5.<br>Updated Typ. and Max. Values of $I_{DD}$ in Table 45-58. |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | Updated Max. Values of IDLE2 in Table 46-2. Deleted note 2.<br>Updated Min. Value of $Cyc_{EEPROM}$ in Table 46-12.<br>Updated Typ. and Max. Values of $I_{DD}$ in Table 46-20. |
| Electrical Characteristics 105°C (SAM C20/C21 N) | Updated Typ. and Max. Values of $I_{DD}$ in Table 47-25. |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 E/G/J) | Updated Typ. and Max. Values of $I_{DD}$ in Table 48-33. |


© 2026 Microchip Technology Inc. and its subsidiaries

| Revision History (continued) | |
|---|---|
| **Section** | **Updates** |
| Schematic Checklist | Added second paragraph in Introduction.<br>Added second paragraph in Operation in Noisy Environment.<br>Updated Figure 53-5 and added note in External Reset Circuit.<br>Updated Figure 53-12 and Table 53-7.<br>Updated Figure 53-13, deleted note and added new note after Table 53-8.<br>Updated Figure 53-14, deleted note and added new note after Table 53-9.<br>Updated Figure 53-15 and added new note after Table 53-10. |

## Revision J - 12/2021

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Features | Added new 64 MHz information |
| Configuration Summary | Added new 64 MHz information and new notes to the tables. |
| Ordering Information | Added information related to the new 64 MHz packages. |
| Block Diagram | Updated the Processor section of the Diagram to denote both 48 MHz and 64 MHz parts. |
| PM - Power Manager | Updated the table in Regulator Automatic Low Power Mode with a new note. |
| SUPC - Supply Controller | • Updated Enabling, Disabling, and Resetting with new text<br>• Updated the VREG Register with new information for the ENABLE bit |
| DMAC - Direct Memory Access Controller | Updated Addressing with a new Register name. |
| PORT - I/O Pin Controller | • Updated the Overview of the PORT Block Diagram in the Functional Description<br>• Added a new note to Events |
| EVSYS - Event System | Added the CHANNELn.PATH column to the table in Sleep Mode Operation. |
| CAN - Controller Area Network | Updated the TSCV Register to properly display the TSC bitfield. |
| ADC - Analog-to-Digital Controller | Removed an erroneous reference to OPAMP from Features. |
| Electrical Characteristics at 85°C (SAM C20/C21 E/G/J) | • Updated the tables in Maximum Clock Frequencies to account for new information on 64 MHz devices<br>• Updated Power Consumption with new 64 MHz information and a new table<br>• Updated NVM Characteristics with 64 MHz information for table 45-38<br>• Updated Digital Phase Locked Loop (DPLL) Characteristics with new 64 MHz information<br>• Updated the table and notes for table 45-54 in 48 MHz RC Oscillator (OSC48M) Characteristics |
| Electrical Characteristics at 105°C (SAM C20/C21 E/G/J) | • Updated table 46-2 with new data for IDLE2 in Power Consumption<br>• Updated table 46-19 with new information and new notes in 48 MHz RC Oscillator (OSC48M) Characteristics |
| Electrical Characteristics AEC-Q100 Grade 1, 125°C (SAM C20/C21 E/G/J) | Updated table 48-32 with new information and notes in 48 MHz RC Oscillator (OSC48M) Characteristics. |
| Packaging Information | Added a new note applicable to the QFN packages with an exposed die to Package Drawings. |


© 2026 Microchip Technology Inc. and its subsidiaries

| **Revision History** (continued) | |
|---|---|
| **Section** | **Updates** |
| Schematic Checklist | • Updated the figures in:<br>– Power Supply Connections<br>– External Reset Circuit<br>– Programming and Debug Ports<br>– Cortex Debug Connector (10-pin)<br>– 10-pin JTAGICE3 Compatible Serial Wire Debug Interface<br>– 20-pin IDC JTAG Connector |

## Revision H - 09/2021

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| General | The SPI and I²C standards use the terminology "Master" and "Slave". The equivalent Microchip terminology, "Host" and "Client," is used in this document. This terminology has been updated throughout this document for this revision. |
| Features | Added new bulleted items for the PWM under Peripherals. |
| Configuration Summary | Updated the TC and Waveform PWM entries in the following tables:<br>• C21 Family Features<br>• C20 Family Features |
| Signal Descriptions List | Updated the TC and TCC entries with new verbiage for waveform/PWM outputs |
| Power Supply and Start-Up Considerations | Updated the title of Power-Up Sequence to Power-Up Sequence - Maximum Rise Rate, and consolidated the text. |
| Processor and Architecture | Removed Erroneous PM-Power Manager mentions in the Interrupt Line Mapping tables. |
| MCLK | • Removed incorrect CLKCFG text from Clock Ready Flag<br>• Updated the INTFLAG Register changing CLKCFG to CPUDIV |
| SUPC | Added new text to the bit description for the BODVDDRDY bitfield in the STATUS register. |
| EVSYS | • Updated the Event Generators table in the CHANNELn Register<br>• Updated the User Multiplexer Number table in the USERm Register |
| SERCOM - USART | Removed erroneous text from I/O Lines. |
| SERCOM - SPI | • Removed erroneous text from I/O Lines<br>• Updated the text for the DOPO bitfield in the CTRLA Register |
| SERCOM I²C | Updated the verbiage in the DRDY bitfield for the INTFLAG Register. |
| TCC | Updated the Register Offset equation for the CCx Register. |
| CCL | Updated the Linked Lut Input Selection figure in Truth Table Inputs Selection with a new image. |
| ADC | Updated the equation for the SAMPLEN bitfield in the SAMPCTRL Register. |
| Electrical Characteristics at 85°C | • Updated the POR Operating Principle image in POR - Power ON Reset Characteristics<br>• Updated the images BODVDD Hysteresis OFF and BODVDD Hysteresis ON in BODVDD - Brown Out Detector Characteristics<br>• Updated the Differential Mode, and Single-Ended Mode tables with new Conditions for the TUE Parameter in Analog-to-Digital Converter (ADC) Characteristics<br>• Removed erroneous $\overline{SS}$ line from the SPI Timing Requirements in Host Mode figure in SERCOM in SPI Mode Timing |


© 2026 Microchip Technology Inc. and its subsidiaries

| Revision History (continued) | |
|---|---|
| **Section** | **Updates** |
| Electrical Characteristics at 105°C (SAM C20/C21 N) | • Updated the Differential Mode, and Single-Ended Mode tables with new Conditions for the TUE Parameter in Analog-to-Digital Converter (ADC) Characteristics<br>• Updated the images BODVDD Hysteresis OFF and BODVDD Hysteresis ON in BODVDD - Brown Out Detector Characteristics |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 E/G/J) | Updated the Differential Mode, and Single-Ended Mode tables with new Conditions for the TUE Parameter in Analog-to-Digital Converter (ADC) Characteristics. |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 N) | • Updated the Differential Mode, and Single-Ended Mode tables with new Conditions for the TUE Parameter in Analog-to-Digital Converter (ADC) Characteristics<br>• Updated the images BODVDD Hysteresis OFF and BODVDD Hysteresis ON in BODVDD - Brown Out Detector Characteristics |
| Appendix A | Updated the Ordering Information with all new tables. |
| Packaging Information | Updated the note in the following packages:<br>• 64-Pin VQFN<br>• 48-Pin VQFN |
| Schematic Checklist | Updated External Reset Circuit with new images and a new note. |

## Revision G - 11/2020

The following updates were added in this revision of the document.

| Section | Updates |
|---|---|
| Configuration Summary | Updated the number of ADC channels for the SAM C21 in the SAM C21 Family Features table. |
| Processor and Architecture | The bit address and reset value in SRAM Quality of Service. |
| DSU | Updated Testing of On-Board Memories MBIST with new test actions in the MBIST Operation Phases table. |
| Clock System | Updated the Clock Request Routing image in On-demand, Clock Requests. |
| GCLK | Updated the CHEN bit of the PCHCTRLm Register with a new verbiage. |
| MCLK | • Updated Peripheral Clock Masking topic with the removal of the table<br>• Updated the APBCMASK Register with a new R/W property for the DAC bit |
| OSC32KCTRL | Updated the XOSC32K register with a new verbiage for the STARTUP bit |
| RTC | Updated the following registers with new notes:<br>• FREQCORR<br>• COUNT (COUNT32 Mode)<br>• COMP (COUNT32 Mode)<br>• COUNT (COUNT16 Mode)<br>• PER (COUNT16 Mode)<br>• COMPn (COUNT16 Mode)<br>• CLOCK (Clock/Calendar Mode)<br>• ALARM (Clock/Calendar Mode)<br>• MASK (Clock/Calendar Mode) |
| DMAC | Updated the DESCADDR Register with 64 as the new number of bits for alignment in the DECADDR bit. |
| NVMCTRL | • Updated the ADDR Register with a new verbiage and added an example<br>• Updated the LOCK Register with a new Register reset value, and a new verbiage for the LOCK bit |


© 2026 Microchip Technology Inc. and its subsidiaries

**Revision History** (continued)

| Section | Updates |
|---|---|
| EVSYS | • Updated Features with a new text for the last bulleted item<br>• Updated the table in the CHANNEL bit of the USERm register |
| SERCOM $I^2C$ | Updated the Signal Description with a new cross reference. |
| CAN | Updated the RF1L bit verbiage in the RXF1S Register. |
| TC | Updated the following topics with additions to the titles, a new note, and image updates:<br>• Capture Operations<br>• Event and I/O pin Capture Action<br>• Period and Pulse-Width (PPW) Capture Action on Events<br>• Pulse-Width Capture Action on Events<br>• Time-Stamp Capture on Event or pin<br>Updated the following registers with a new verbiage:<br>• CTRLBCLR with updates to the LUPD bit<br>• CTRLBSET with updated to the LUPD bit |
| TCC | Updated the CTRLA Register with the addition of the DMAOS bit. |
| CCL | Updated this chapter with the following items:<br>• Clarified the 4 programmable Lookup Tables in Features<br>• Added a new Block Diagram<br>• Updated the Signal Description with new Pin names in the table |
| AC | Updated the COMPCTRL Register with new Descriptions in the tables for the MUXPOS and MUXNEG bits |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | Updated the following Topics:<br>• I/O Pin Characteristics with the addition of verbiage for the RESET_N pin<br>• Updated the Diagrams in BODVDD - Brown Out Detector Characteristics<br>• Updated Analog-to-Digital Converter (ADC) Characteristics by removing an erroneous equation segment. Updated table formatting to better display the tables within this chapter.<br>• Updated the table formatting in Digital to Analog Converter (DAC) Characteristics to better display the information<br>• Updated the table formatting and verbiage in Voltage Reference Characteristics |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | Updated the following Topics:<br>• Updated the Diagrams in BODVDD - Brown Out Detector Characteristics |
| Electrical Characteristics 105°C (SAM C20/C21 N) | Updated the following Topics:<br>• Updated the Diagrams in BODVDD - Brown Out Detector Characteristics<br>• Updated Analog-to-Digital Converter (ADC) Characteristics by removing an erroneous equation segment. Updated table formatting to better display the tables within this chapter.<br>• Updated the table formatting in Digital to Analog Converter (DAC) Characteristics to better display the information<br>• Updated the table formatting and verbiage in Voltage Reference Characteristics |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21E/G/J) | Updated the following Topics:<br>• I/O Pin Characteristics with the addition of verbiage for the RESET_N pin<br>• Updated the table formatting in Digital to Analog Converter (DAC) Characteristics to better display the information<br>• Updated the table formatting and verbiage in Voltage Reference Characteristics |


© 2026 Microchip Technology Inc. and its subsidiaries

| **Revision History** (continued) | |
|---|---|
| **Section** | **Updates** |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21N) | Updated the following Topics:<br>• Updated the Diagrams in BODVDD - Brown Out Detector Characteristics<br>• Updated Analog-to-Digital Converter (ADC) Characteristics with new table formatting to better display the tables within this chapter |
| Appendix A | Added a new appendix for ISELED parts specification and ordering information. |
| Appendix B | Added a new Appendix for SIL 2 Enabled Functional Safety Devices. |

## Revision F - 09/2020

This revision included numerous typographical updates throughout the document.

| Section | Updates |
|---|---|
| General | • Typical Powering Schematics removed from Power Supply and Startup Considerations chapter. It now refers to the Schematic Checklist chapter.<br>• OSCCTRL XOSCCTRL.GAIN value 0x4 updated to 'max 32MHz recommended frequency'<br>• SERCOM baud rate equations in table 30-2: added asynchronous fractional mode<br>• SERCOM SPI CTRLA.DOPO description updated<br>• ADC bit field names in NVM Software Calibration Area Mapping table 9-5<br>• Addition of IDLE0 and IDLE2 modes in PM SLEEPCFG register<br>• SDADC REFCTRL.REFRANGE bit field removed<br>• DMAC SRCADDR and DSTADDR description update<br>• NVMCTRL LOCK register default state after erase updated to 0xFFFF<br>• CAN Sleep Mode Operation updated with regards to IDLE0 and IDLE2 modes<br>• CAN CCCR.NISO bit added<br>• CAN IR.TC bit field description updated<br>• TC PER register added for 8-bit mode<br>• TCC EVCRTL.EVACT value 0x6 added as STAMP<br>• CCL: added a note about TCC2 outputs<br>• CCL LUTCTRLn.INSELx value 0x8 added for TCC |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | • AC Characteristics table note 4 completed<br>• General Operating rating note updated : I/O in VDDIO clusters can be multiplexed as analog inputs and outputs.<br>• SDADC Operating Conditions table updated<br>• ADC Csample typical value removed, max value updated 4.5pF => 3.2pF |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | Added new section PTC Characteristics |
| Electrical Characteristics 105°C (SAM C20/C21 N) | • Added new section PTC Characteristics<br>• AC Characteristics table note 4 added<br>• IDLE0 current consumption at 105°C updated in Power Consumption<br>• ADC Operating Conditions table updated and note 2 added<br>• ADC Csample typical value removed, max value updated 4.5pF => 3.2pF<br>• SDADC Operating Conditions table updated |
| Electrical Characteristics AEC - Q100 Grade 1. 125°C (SAM C20/C21 E/G/J) | • Updated the PTC Power Consumption tables with new values.<br>• Updated the Current Consumption table with new IDLE0 Current Consumption at 125°C in Power Consumption |


© 2026 Microchip Technology Inc. and its subsidiaries

| Revision History (continued) | |
|---|---|
| **Section** | **Updates** |
| Electrical Characteristics AEC - Q100 Grade 1, 125°C (SAM C20/C21 N) | Added new Sensor Load Capacitance table to PTC Characteristics. |
| Packaging Information | • Removed references to moisture sensitivity for each package<br>• Updated the Package Drawings to the latest version |

## Revision E - 06/2020

This revision included numerous typographical updates throughout the document.

| | |
|---|---|
| Ordering Information | Updated the diagram and notes to display AEC - Q100 nomenclature. |
| SAM C20/C21 N AEC - Q100 Electrical Characteristics | New electrical specifications chapter added. |
| General | • Updated all references in the document to AEC - Q100 to match industry naming conventions<br>• Updated the device name throughout the document to SAM C20/C21 |

## Revision D - 01/2020

| | |
|---|---|
| Configuration Summary | • Updated Table 1-3 SAM C21 Family Features with new values for the PTC<br>• Updated Table 1-4 SAM C20 Family Features with new values for the AC |
| Block Diagram | Updated the arrow to bridge between PORT and AHB-APB Bridge B. |
| Signal Descriptions | Updated Table 5-1 Signal Descriptions List - SAM C20/C21 with small typographical edits for VOUT and VREFA. |
| Multiplexed Signals | Updated Table 6-1, Table 6-2, Table 6-3, and Table 6-4 with new ADC-DAC/VREFA values in the REF column. |
| Processor and Architecture | Updated Table 10-3 and 10-4 in Interrupt Line Mapping with notations designating which chip the Peripheral Source applies to. |
| GCLK | • Updated Figure 16-1 Block Diagram to correct missing information<br>• Updated the GENCTRL bit for the SYNCBUSY register |
| MCLK | • Updated the AHBMASK Register to display the DIVAS and APBD bits<br>• Updated the TC4 bit of the APBCMASK Register to display the Description table<br>• Updated the APBDMASK Register with a new note on availability of features for SAMC2x “N” Series devices |
| PM | Updated the SLEEPMODE bit of the SLEEPCFG Register with new data in the table. |
| OSCCTRL | Updated the AMPGC bit of the XOSCCTRL Register with a new note. |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| SUPC | • Updated Voltage Reference System Operation with new text referencing the INTREF<br>• Updated Selecting a Voltage Reference with new text for the INTREF<br>• Updated the ACTCFG bit of the BODVDD Register to remove an erroneous reference to Sleep Mode<br>• Updated the STDBYCFG bit of the BODVDD Register with new table values |
| EIC | • Updated Initialization with new information for line 6<br>• Updated Interrupt Pin Debouncing with new text designating the availability of features for SAM C20/C21 N Variants<br>• Updated the DEBOUNCEN Register with a new note<br>• Updated the DPRESCALER Register with a new note<br>• Updated the PINSTATE Register with a new note |
| NVMCTRL | Updated the LOCK Register with new reset values. |
| PORT I/O Pin Controller | Corrected typographical error in OUTTGL reference in Features. |
| CAN | • Updated Power Management with new interrupt text<br>• Updated Sleep Mode Operation with new introductory sentence |
| CCL | • Updated Truth Table Inputs Selection for Internal Events Inputs Selection (EVENT) with C20/C21 availability text<br>• Updated the table for the INSELx bit in the LUTCTRLn Register |
| ADC | • Updated Overview with supporting text for the INTREF voltage reference<br>• Updated Features with new supporting text for INTREF<br>• Updated Analog Connections with new supporting text for INTREF<br>• Updated Reference Configuration with new supporting text for INTREF<br>• Updated the REFSEL bit for the REFCTRL Register<br>• Updated the MUXPOS bit for the INPUTCTRL Register |
| SDADC | • Removed erroneous information from Features for Conversion Range<br>• Updated the REFSEL bit for the REFCTRL Register |
| AC | • Updated the Overview with information for CMP2 and CMP3 on E and G variants<br>• Updated Features with information on the INTREF<br>• Updated the Block Diagrams to replace Bandgap with INTREF<br>• Updated Analog Connections to replace Bandgap text with INTREF<br>• Updated Principle of Operation to replace Bandgap text with INTREF<br>• Updated the table for the MUXNEG bit of the COMPCTRL Register |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| DAC | • Updated Analog Connections to replace Bandgap text with INTREF<br>• Updated the table for the REFSEL bit for the CTRLB Register |
| FREQM | • Added a new note to the Block Diagram<br>• Updated the DIVREF bit description for the CFGA Register |
| Electrical Characteristics at 85°C | • Updated Table 45-18 Decoupling Requirements in Voltage Regulator Characteristics with new values and a new note<br>• Updated Table 45-19 Operating Conditions in Analog-to-Digital Converter (ADC) Characteristics with INTREF replacing Bandgap<br>• Updated Table 45-23 Operating Conditions in Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics with the addition of a note and the removal of single-ended mode references<br>• Updated Crystal Oscillator Characteristics with a new equation and removed $C_{SHUNT}$from the diagram |
| Electrical Characteristics at 105°C | Updated Table 47-8 Operating Conditions in Sigma-Delta Analog-to-Digital Converter (SDADC) Characteristics to remove the Single-ended mode reference and added new note. |
| AEC Q100 Grade 1, Electrical Characteristics at 125°C | Updated Table 48-3 Current Consumption with new values for the Maximum under Standby. |
| Schematic Checklist | Removed pF references from Figure 50-7 Crystal Oscillator Schematic and removed pF references from Table 50-5 in Crystal Oscillator |

## Revision C - 01/2019

| | |
|---|---|
| Ordering Information | • Added a note to clarify SAM C2xN availability in 105°C<br>• Introduced AEC - Q100 Grade 1 qualified silicon revision F<br>• Clarified availability of factory programmed Bootloader for WLCSP |
| PAC | • Added missing CAN1 bit field to INTFLAGC register<br>• Corrected a typographical error for PTC bit field in INTFLAGC & STATUSC register<br>• Corrected a typographical error for RSTC bit field in STATUSA register |
| OSC32KCTRL | • RTCCTRL register was missing, and same has been added in this version |
| I/O Multiplexing and Considerations | • Corrected a typographical error for SERCOM4 and SERCOM5 in Table 6-2 |
| RTC | • Register (EVCTRL, INTENCLR, INTENSET, INTFLAG, SYNCBUSY) summary typographical error was addressed<br>• FREQCORR, DBGCTRL registers were missing in register summary, added in this version |
| SUPC | • Clarified the ENABLE bit in section 22.8.6 Voltage Regulator System (VREG) Control<br>• Corrected typo for Reset value of the Voltage Regulator System (VREG) Control register in the section 22.8.6 |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| SERCOM-USART | • Clarified Auto baud detection mode for LIN Slave in CTRLA register |
| TCC | • Corrected a typo in the Register summary - 32-bit mode for the PER & PERBUF registers has been removed |
| ADC | • Updated section 38.6.2.3 Operation<br>• Updated section 38.6.5 Interrupts |
| Electrical Characteristics at 85°C | • Typographical errors in the Electrical specification chapter addressed<br>– Table 45-32. Power Consumption<br>• Clarified SPI maximum speed information in Table 45-56<br>• ADC , SD ADC, and AC Electrical Characteristics for silicon revision F added |
| Electrical Characteristics at 105°C | Updated AC Table 46-8. Power Consumption |
| AEC-Q100 Grade 1, 125°C Electrical Characteristics | New chapter of electrical specifications for AEC - Q100 Grade qualified device. |
| Packaging | • Top marking legend information added for all packages |

## Revision B - 06/2017

| | |
|---|---|
| General | • The SAM C20 Family Data Sheet (DS60001480A) was combined with this data sheet to create this version<br>• The Errata chapter was removed. This content is now provided in a separate document. |

## Revision A - 03/2017

| | |
|---|---|
| General Updates | • Updated the document from Atmel to Microchip style and template<br>• The literature number changed from the Atmel 42365 to the Microchip DS60001479A<br>• The Data Sheet revision letter was restarted to A<br>• An ISBN number was added |
| Ordering Information | • Removed space form the ordering codes. |
| Introducing SAM C20/C21N | • 100-pin TQFP package option.<br>• More features: Eight TCs, eight SERCOMs |
| EIC – External Interrupt Controller | • Added interrupt pin debouncing for SAM C20/C21N. |
| Configurable Custom Logic (CCL) | • LUTCTRLn.INSELx: Added ALT2TC at INSELx=0xA. For the ALT2TC options the LUT 0 to 3 mapping will be TC4,TC5,TC6,TC7. The ALT2TC is only applicable for SAM C20/C21N. |
| OSC32KCTRL – 32.768 kHz Oscillators Controller | • XOSC32K.STARTUP[2:0]: Table for start-up times updated. |
| CAN - Control Area Network (SAM C21 Only) | • Updated block diagram.<br>• The CAN cannot operate in Standby sleep mode:<br>– Merged content from "Power Down (Sleep Mode)" section into Sleep Mode Operation.<br>– Updated description of Power Management .<br>– MRCFG.RUNSTDBY bit removed. |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | • Digital-to-Analog Converter (DAC) Characteristics: Updated conditions and typical numbers for power consumption.<br>• Digital Phase Locked Loop (DPLL) Characteristics: Added typical characterization numbers.<br>• 48 MHz RC Oscillator (OSC48M) Characteristics: Updated $T_{START}$ values, updated note 4 and removed the condition. |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | • Digital-to-Analog Converter (DAC) Characteristics: Updated conditions and typical numbers for power consumption.<br>• Digital Phase Locked Loop (DPLL) Characteristics: Added typical characterization numbers.<br>• 48 MHz RC Oscillator (OSC48M) Characteristics: Updated $T_{START}$ values, updated note 4 and removed the condition. |
| Electrical Characteristics 105°C (SAM C20/C21 N) | • Electrical characterization data added for SAM C20/C21. |
| Schematic Checklist | • External Reset Circuit: Updated schematic diagram and recommended pin connections. |

## 54.2. Rev KJ - 11/2016

| | |
|---|---|
| Errata SAM C20 and Errata SAM C21 | • Added DMAC errata (reference 15670). |

## 54.3. Rev J - 10/2016

| | |
|---|---|
| PAC - Peripheral Access Controller | • INTFLAGA.TSENS moved to bit position 12.<br>• STATUSA.TSENS moved to bit position 12. |

## 54.4. Rev I - 09/2016

| | |
|---|---|
| General | • Removed preliminary status from the datasheet. |
| Ordering Information | • SAM C20J/SAM C21J: Added -UUT ordering codes for SAM C20/C21J17 and SAM C20/C21J18 |
| Pinout | • Added pinout for the WLCSP56 package. |
| I/O Multiplexing and Considerations | • Multiplexed Signals: VREFB removed from the reference (REF) column. This is not an option. |
| OSCCTRL – Oscillators Controller | • 48MHz Internal Oscillator (OSC48M) Operation: Removed the sentence "Frequency selection must be done when OSC48M is disabled." |
| SUPC – Supply Controller | • Removed references to backup domain. |
| WDT – Watchdog Timer | • Removed references to backup domain. |
| RTC – Real-Time Counter | • Clock/Calendar (Mode 2): Updated description. |
| DMAC – Direct Memory Access Controller | • Sleep Mode Operation: Added information on behaviour of DMA channels with CHCTRLA.RUNSTDBY=0. |
| EIC – External Interrupt Controller | • Added interrupt pin debouncing. |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter | • LIN Host section added.<br>• CTRLA.TXPO: Row heading updated from RTS to RTS/TE.<br>• CTRLA.FORM: Added LIN Master to FORM[3:0]=0x2. Added LIN Slave to FORM[3:0]=0x4.<br>• CTRLB.LINCMD[3:0] bit group added<br>• CTRLC.GTIME: Bitfield values removed. |
| Timer/Counter (TC) | • CTRLA.ENABLE and SWRST bit description updated: Added "This bit is not enable protected." |
| ADC - Analog-to-Digital Converter | • Reference Configuration: Removed information on number of external and internal voltage references and supported voltage supply range. This information is replaced with references to the REFCTRL.REFSEL register bits and ADC characteristics for reference selection details and voltage ranges respectively. |
| DAC – Digital-to-Analog Converter (SAM C21 only) | • CTRLB.ION bit description updated: For bit value '1' the internal DAC can be used as input to the AC or ADC. |
| TSENS – Temperature Sensor | • Added example to the VALUE register. |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | • Absolute Maximum Ratings: VDD max updated from 5.5V to 6.1V.<br>• General Operating Ratings: Updated note.<br>• Injection Current: New section added.<br>• Power Consumption: Standby typical values updated and maximum values added.<br>• Analog-to-Digital Converter (ADC) Characteristics:<br>– Added Ts, sampling tile with DAC as input.<br>– Analog-to-Digital Converter (ADC) Characteristics: In the condition column REFCTRL.REFSEL is corrected to CTRLC.RESSEL.<br>– Analog Comparator Characteristics: Removed Hysteresis for COMPCTRLn.SPEED = 0x0 (low power), Updated IDDANS units from nA to µA and updated condition for IDDANA with voltage scaler disabled (COMPCTRLn.SPEED = 0x1 changed to 0x3).<br>• Digital Phase Locked Loop (DPLL) Characteristics: Updated values.<br>• 48 MHz RC Oscillator (OSC48M) Characteristics: Added note on the output frequency regarding accuracy for the WLCSP package. |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | • Power Consumption: Standby typical values updated and maximum values added.<br>• NVM Characteristics: New section added.<br>• Analog Comparator (AC) Characteristics: Updated IDDANA units from nA to µA and updated condition for IDD with voltage scaler disabled (COMPCTRLn.SPEED = 0x1 changed to 0x3).<br>• Digital Phase Locked Loop (DPLL) Characteristics: Characterization data added. |
| Packaging Information | • Added package outline drawing (POD) for 56-Ball WLCSP. |
| Schematic Checklist | • External Analog Reference Connections: Recommended pin connections column updated.<br>• External Reset Circuit: Updated description. |


© 2026 Microchip Technology Inc. and its subsidiaries

## 54.5. Rev H - 05/2016

| | |
|---|---|
| Product Mapping | AHB-APB Bridge B:<br>• DMAC base address corrected from 0x41004400 to 0x4106000.<br>• MTB base address corrected from 0x41004800 to 0x41008000.<br>• Reserved space corrected from 0x41005000 to 0x41009000. |
| Micro Trace Buffer | MTB base address corrected from 0x41006000 to 0x41008000. |
| SUPC – Supply Controller | VDD Brown-Out Detector (BODVDD): Removed references to battery backup (VBAT) and voltage monitored bit (BODVDD.VMON). |
| ADC - Analog-to-Digital Converter | Updated formula to increase the resolution by n bits in Oversampling and Decimation. |
| SDADC – Sigma-Delta Analog-to-Digital Converter (SAM C21 only) | Decimation Filter: Removed figure of spectral mask of an OSR=32. This option is not available. |
| TSENS – Temperature Sensor | • INTFLAG.OVF bit description updated.<br>• GAIN and OFFSET register bit description updated. |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | • Digital-to-Analog Converter (DAC) Characteristics: Clock and timing conversion rate conditions updated: Rload > 5kW corrected to Rload > 5kΩ.<br>• Added Temperature Sensor Characteristics. |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | Added Analog Characteristics. |

## 54.6. Rev G - 04/2015

| | |
|---|---|
| Ordering Information | Added Device Identification. |
| I/O Multiplexing and Considerations | New sections added:<br>• SERCOM I2C Pins: Information moved from the "Type" column in Table 6-2 into separate table.<br>• Updated CCL column.<br>• GPIO Clusters: Moved from Absolute Maximum Ratings.<br>• TCC Configurations: Moved from Timer/Counter for Control Applications (TCC). |
| Memories | • Updated Table 9-4.<br>• Updated Table 9-1. |
| PAC - Peripheral Access Controller | Register bit correction: INTFLAGAHB, INTFLAGA, INTFLAGB, INTFLAGC, STATUSA, STATUSB and STATUSC |
| DSU - Device Service Unit | Table 13-6 updated: MBIST is not available when the device is protected from the external address space. |
| GCLK - Generic Clock Controller | • Block Diagram: GCLK_MAIN goes into the MCLK, not the PM.<br>• Signal Description: Available signals are GCLK_IO[7:0]. |
| MCLK – Main Clock | • Updated block diagram in Selecting the Synchronous Clock Division Ratio. |
| OSCCTRL – Oscillators Controller | Added OSC48M Calibration (CAL48M) register added (only available for Rev D silicon). |
| SUPC – Supply Controller | • Updated VREF.SEL bit selection table.<br>• Removed references to BODCORE register and bit descriptions and updated description in VDDCORE Brown-Out Detector (BODCORE). |
| PM - Power Manager | • Sleep modes: Removed references to IDLE0 and IDLE1. Renamed IDLE2 to IDLE. |
| DMAC – Direct Memory Access Controller | CTRL.CRCENABLE bit added in bit position 2. |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| NVMCTRL – Nonvolatile Memory Controller | CTRLB.CACHEDIS: Updated from one bit in postion 18 to two bits in position 19:18. Updated bit description and bit value settings. |
| SERCOM SPI – SERCOM Serial Peripheral Interface | Features: Updated references to serial clock speed in master and slave operation. |
| CAN - Control Area Network (SAM C21 Only) | CREL: Updated reset value from 0x31000000 (device rev B) to 0x32100000 (device rev C and newer). |
| Timer/Counter (TC) | Added register property "Write-Synchronized" to the CCBUFx and PERBUF registers. |
| Timer/Counter for Control Applications (TCC) | • Updated number of TCC instances from one to three.<br>• Counter Operation: 'Stop Command and Event Action' split into 'Stop Command' and 'Pause Event Action'<br>• Capture Operations: Value 0 in CAPTMIN mode is captured only in down-counting mode.<br>• Ramp Operations: RAMP2C Operation added.<br>• Compare Operations: Reorganization of section.<br>• Corrected bit names in the WAVE register: CIRCCENx -> CICCENx and CIRPEREN -> CIPEREN. |
| Configurable Custom Logic (CCL) | • Number of LUTCTRL registers changed from eight to four.<br>• Number of SEQCTRL registers changed from four to two.<br>• Truth Table Inputs Selection: Updated description and figure in Analog Comparator Inputs (AC). |
| SDADC – Sigma-Delta Analog-to-Digital Converter (SAM C21 only) | • Resolution corrected from 24-bit to 16-bit.<br>• Conversion range updated from "0V to $V_{ref}$ "to "0V to 0.7x$V_{ref}$"<br>• Test Mode section removed.<br>• Updated operation formula in the following registers:<br>– OFFSETCORR<br>– GAINCORR<br>– SHIFTCORR<br>• Updated RESULT bit description in RESULT. |
| AC – Analog Comparators | COMPCTRL: SPEED bit description updated. Values 0x1 and 0x2 is reserved. |
| DAC – Digital-to-Analog Converter (SAM C21 only) | Updated DATA register: DATA bits access corrected from read/write (R/W) to write (W).TPUBSAMD-354 |
| TSENS – Temperature Sensor | Measurement: Added temperature measurement recommendation to avoid discrepancies. |
| Electrical Characteristics 85°C (SAM C20/C21 E/G/J) | • Added electrical characteristics for 85°C.<br>• GPIO Clusters moved to I/O Multiplexing and Considerations. |
| Electrical Characteristics 105°C (SAM C20/C21 E/G/J) | • Added electrical characteristics for 105°C. |
| Packaging Information | Updated package drawings to include GPC, drawing no. and revision letter. |
| Errata SAM C20 and Errata SAM C21 | • Updated revision B errata: Added Errata reference 14497, 14633 and 15342.<br>• Added revision C errata.<br>• Added revision D errata. |

## 54.7. Rev F - 02/2015

| | |
|---|---|
| Configuration Summary | Number of PTC X and Y lines updated for SAM C20/C21G and SAM C20/C21E. |
| DSU - Device Service Unit | ADDR: Added AMOD bits. |
| SUPC – Supply Controller | Debug Operation: Updated description. |


© 2026 Microchip Technology Inc. and its subsidiaries

| | |
|---|---|
| References to oscillator OSC16M removed and replaced with OSC48M: | • Clock Distribution: Block diagram updated.<br>• Block Diagram: OSCM16M replaced by OSC48M. DFLL48M removed.<br>• GENCTRLn.SRC[4:0]: Value 0x6 description updated.<br>• 48MHz Internal Oscillator (OSC48M) Operation. |
| Event System (EVSYS) | CTRLA: Note added to CTRLA.SWRST bit description. |
| DMAC – Direct Memory Access Controller | Updated description of the PRICTRL0.LVLPRIn bits. |
| Timer/Counter (TC) | Updated section Clocks: The TC bus clocks (CLK_TCx_APB) can be enabled and disabled in the *Main Clock Module (MCLK)* (not the Power Manager). |
| SDADC – Sigma-Delta Analog-to-Digital Converter (SAM C21 only) | • Block Diagram: Reference selection updated.<br>• REFCTRL: Added note to REFSEL bit description. |
| AC – Analog Comparators | Removed references to multiple level hystereseis. Levels are not available, only on or off:<br>• Features: Selectable hysteresis updated from "4-levels" or off to "on or off".<br>• Comparator Configuration: Removed references to COMPCTRLx.HYST bits.<br>• Input Hysteresis: Removed references to COMPCTRLx.HYST bits.<br>• Register Summary: Removed the COMPCTRLx.HYST bits.<br>• COMPCTRL: Removed the HYST bits. |

## 54.8. Rev E - 12/2015

| | |
|---|---|
| Configuration Summary | • Corrected memory sizes.<br>• Number of ADC channels corrected.<br>• Number of TCC instances corrected from three to one. |
| Ordering Information | • Introduced 105°C ordering codes.<br>• Corrected package type from QFN48 to TQFP48 for ATSAM C20/C21G16A-AUT. |
| DSU - Device Service Unit | Bit CTRL.CRC is write-only. |
| NVMCTRL – Nonvolatile Memory Controller | Updated description in NVM Write: Removed reference to default MANW value. This is covered in the CTRLB.MANW bit description. |
| DMAC – Direct Memory Access Controller | Added note in Sleep Mode Operation. |
| Configurable Custom Logic (CCL) | Removed oscillator related sub sections from Sequential Logic. |
| SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter | • Added RS485 to the TXPO bit description in the CTRLAregister. |

## 54.9. Rev D - 09/2015

| | |
|---|---|
| SERCOM USART - SERCOM Synchronous and Asynchronous Receiver and Transmitter | • Updated formula in the RXPLregister. |
| Errata SAMC20 and Errata SAMC21 | • Reinserted errata section which was missing from datasheet rev C. |


© 2026 Microchip Technology Inc. and its subsidiaries

## 54.10. Rev C - 09/2015

| | |
|---|---|
| General | Editorial updates. |
| DMAC – Direct Memory Access Controller | • Updated number of bits in the SWTRIGCTRL, INTSTATUS, BUSYCH and PENDCH registers (Related to number of DMA channels available). |
| PORT - I/O Pin Controller | • Functional Description: Overview diagram updated. |
| ADC - Analog-to-Digital Converter | • Block Diagram: Renamed ADC input signals from ADC to AIN.<br>• Signal Description: Renamed ADC signal to AIN |
| Peripheral Touch Controller (PTC) | • Block Diagram updated.<br>• Section Self-capacitance Sensor Arrangement updated. |

## 54.11. Rev B - 06/2015

| | |
|---|---|
| Ordering Information | • Remove carrier type Tray option. |

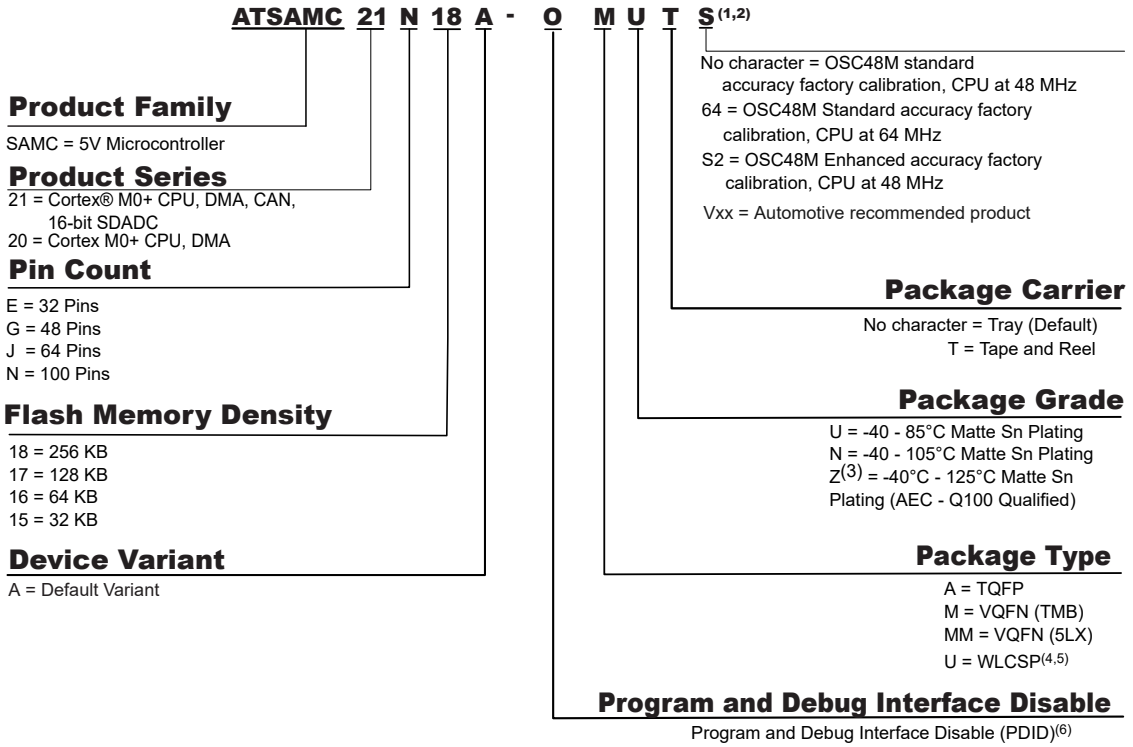
## 54.12. Rev A - 04/2015

| |
|---|
| Initial revision. |


© 2026 Microchip Technology Inc. and its subsidiaries

# 55. Product Identification System

To order or obtain information, for example, on pricing or delivery, contact Microchip: https://www.microchip.com/en-us/about/contact-us.




**Notes:**

1. Not all combinations are valid. The available ordering numbers are listed in the Configuration Summary.
2. The SAM C20/C21 N product is available for the 105°C temperature grade and AEC - Q100 Grade 1 (125°C temperature).
3. The AEC - Q100 Grade 1 qualified version is offered for SAM C20/C21 E/G/J in the TQFP and VQFN packages only, and for SAM C20/C21 N in the TQFP package only. The VQFN package will have wettable flanks, and both TQFP and VQFN packages are assembled with gold bond wires. The TSENS is not available in AEC - Q100 qualified device part numbers.
4. Devices in the WLCSP package include a factory programmed Bootloader. Contact your local Microchip sales office for additional information.
5. The WLCSP package type is available only with the package Grade U.
6. Contact a local Microchip sales office for additional information about PDID functionality.


© 2026 Microchip Technology Inc. and its subsidiaries

# Microchip Information

## Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

## Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

## Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.


© 2026 Microchip Technology Inc. and its subsidiaries

## Product Page Links

ATSAMC20N17A, ATSAMC20N18A, ATSAMC21E15A, ATSAMC21N17A, ATSAMC21N18A, ATSAMC20E15A, ATSAMC20E16A, ATSAMC20E17A, ATSAMC20E18A, ATSAMC20G15A, ATSAMC20G16A, ATSAMC20G17A, ATSAMC20G18A, ATSAMC20J15A, ATSAMC20J16A, ATSAMC20J17A, ATSAMC20J18A, ATSAMC21E16A, ATSAMC21E17A, ATSAMC21E18A, ATSAMC21G15A, ATSAMC21G16A, ATSAMC21G17A, ATSAMC21G18A, ATSAMC21J15A, ATSAMC21J16A, ATSAMC21J17A, ATSAMC21J18A


© 2026 Microchip Technology Inc. and its subsidiaries