# Highly Integrated Wireless MCU with CAN FD, Ethernet, USB, Motor Control, Graphics, Touch and Enhanced Security

## Introduction

The PIC32-BZ6 family is a high-performance, low-power, integrated wireless microcontroller unit (MCU) powered by 128 MHz ARM® Cortex®-M4F processor. The PIC32-BZ6 family combines robust security features with extensive wireless connectivity options, including Bluetooth® and IEEE 802.15.4 protocols (Thread, Proprietary, MiWi).

With 2 MB of Flash and 512 kB of RAM, the PIC32-BZ6 family provides ample memory for complex applications, while its Secure Boot feature with an immutable root of trust ensures reliable and secure operation. Equipped with a rich set of peripherals, including analog and digital signal processing, communication interfaces, Quadrature Encoder Interface (QEI), and touch sensing, the PIC32-BZ6 family can support a wide range of applications.

The PIC32-BZ6 family consists of AEC-Q100 Grade 1 (125 °C) qualified SoCs, PIC32CX-BZ6, and global regulatory certified, RF-ready modules, PIC32WM-BZ6x.


© 2025 Microchip Technology Inc. and its subsidiaries

# PIC32CX-BZ6 SoC Family Features

**Operating Conditions**

- 1.9-3.6V, -40°C to +125°C, DC to 128 MHz
  - AEC Q100 Grade 1 qualified with reduced parameters
- 1.9-3.6V, -40°C to +105°C, DC to 128 MHz
  - AEC Q100 Grade 2 qualified

**Core: 128 MHz ARM Cortex-M4F**

- 3.40 Coremark®/MHz (with IAR Compiler)
- 4 KB Combined Instruction Cache and Data Cache
- 8-Zone Memory Protection Unit (MPU)
- Thumb®-2 Instruction Set
- Digital Signal Processing Application-Specific Extension (ASE) Rev 2
- Nested Vector Interrupt Controller (NVIC)
- Embedded Trace Module (ETM) with Instruction Trace Stream
- CoreSight™ Embedded Trace Buffer (ETB)
- Trace Port Interface Unit (TPIU)
- IEEE 754-Compliant Single Precision Floating Point Unit (FPU)

**Memories**

- 64 KB ROM for Secure Boot
  - Support for asymmetric secure boot
- 3072-Bit eFuse
  - Secure boot key storage
  - Debug lock
- 2 MB On-Chip Self-Programmable Flash with:
  - Error Correction Code (ECC)
  - Prefetch module to speed up Flash accesses
  - 20k cycles endurance (100k cycles with erase retry option) and 20 years of data retention support
- 72 KB Boot Flash Memory (18 Pages, 17 Pages for Security Hash Enabled)
  - 64 KB for User boot code configuration
  - 8 KB for Flexible device configuration
- 512 KB Multi-Port Programmable QoS SRAM Main Memory
  - 256 KB of ECC RAM option
  - 32 KB of RAM space for CoreSight™ ETB debug usage when enabled
  - Up to 64 KB of SRAM can be retained in Backup mode
- Up to 4 KB of Tightly Coupled Memory (TCM)

**System**

- Power-on Reset (POR) and Brown-out Reset (BOR)
- Internal and External Clock Options
- Integrated 16 MHz Crystal Oscillator with ±30 ppm Stability and External Crystal Support
- External Interrupt Controller (EIC)
  - Up to four external interrupts
  - One non-maskable interrupt
- Extensive Debug and Trace Capabilities
  - 2-Wire Serial Wire Debug (SWD) programming and debugging interface
  - ETM trace interface pins for serial wire trace

**Supported Connectivity Standards**

- Bluetooth® Low Energy SDK Qualified against Bluetooth® Core 6.0
- IEEE 802.15.4 MAC/Phy SDK for Custom Protocol Support(1)
- Zigbee® 3.0(1)
- Thread 1.4(1)

**Power Supply**

- System-on-chip (SoC) Uses Low Dropout (LDO)
- Integrated PMU with:
  - Buck (DC-DC/switching) mode; supports high power (PWM), low power (PSM)
  - Modular Linear Differential Operator (MLDO) mode
- Integrated On-Chip 1.5V LDO Regulator for eFuse
- Integrated On-Chip 1.2V Low Dropout Regulators
- POR and BOR on 3.3V and 1.2V Rails
- Run, Idle, Standby Sleep, Deep Sleep and Extreme Deep Sleep Modes
- SleepWalking Peripherals

**2.4 GHz RF Transceiver**

- Integrated 2.4 GHz Ultra Low-Power RF Transceiver Shared Between Bluetooth Low Energy and IEEE 802.15.4 Modems and Link (MAC) Controllers
- Integrated Transceiver (TRX) Switch and Balun with One Single-Ended Radio Frequency Input/Output (RFIO) for Transmit (TX) /Receive (RX)
- Hardware Radio Arbiter with Programmable QoS:
  - Resolution: up to per packet level
  - Based on shared transceiver and antenna
- High Efficiency Switching Power Ampli ier (PA)
- Programmable Transmit Output Power Ranges from -24 dBm to +11 dBm with 1 dB Step Size
- Supported Data Rates:
  - Bluetooth Low Energy 6.0: 2 Mbps, 1 Mbps, 500 kbps and 125 kbps
  - IEEE 802.15.4: 250 kbps
  - Proprietary 2.4 GHz: 2 Mbps, 1 Mbps and 500 kbps


© 2025 Microchip Technology Inc. and its subsidiaries

**Bluetooth**

- Qualified against Bluetooth Core 6.0
- Up to +11 dBm Programmable Transmit Output Power
- Typical Receiver Power Sensitivity:
  - Bluetooth Low Energy at 2 Mbps: -95 dBm
  - Bluetooth Low Energy at 1 Mbps: -98 dBm
  - Bluetooth Low Energy at 125 Kbps: -108 dBm
  - Bluetooth Low Energy at 500 Kbps: -102 dBm
- Bluetooth Low Energy Supported Features:
  - Encrypted advertising data
  - Security level characteristic
  - Coding scheme selection
  - Connection subrating
  - Periodic advertising enhancement
  - Channel classification enhancement
  - Low energy power control
  - 2M uncoded PHY
  - Long range coded PHY
  - Channel selection algorithm #2
  - Advertising extensions, offloads CPU with hardware based scheduler
  - High duty cycle non-connectible advertising
  - Data length extensions
  - Secure connections
  - Privacy upgrades
  - Robust to interference with wideband RSSI detector
- ECDH P256 Hardware Engine for Bluetooth Pairing Link Key Generation
- AES128 Hardware Module for Real-Time Bluetooth Payload Data Encryption
- Bluetooth Qualification Test Facility (BQTF) Certification
- Supports SIG Defined/Custom Bluetooth Low Energy Profiles and Services
- Bluetooth Low Energy Profiles:
  - Bluetooth Low Energy peripheral and central roles
  - Bluetooth Low Energy APIs for application layer to implement standard or customized GATT based profiles/ services
  - Alert notification service (Apple Notification Center Service (ANCS))
  - Proximity reporter and time information
  - Multi-link and multi-role
- Bluetooth Low Energy Services:
  - Provisioning
  - Over-the-Air firmware update (also known as OTA DFU)
  - Advertisement/Beacon
  - Personalized configuration
  - Alert notification service

**IEEE 802.15.4**

- MiWi™
- Proprietary
- Thread
- PLCP Service Data Unit (PSDU) Data Rate: 250 Kbps
  - Proprietary data rates: 500 Kbps, 1 Mbps and 2 Mbps
- Programmable RX Mode
  - RX sensitivity for 250 Kbps in Continuous mode: -103 dBm
  - RX sensitivity in RPC mode: -100 dBm
  - RPC mode provides lower power consumption in RX mode to support California Green Energy Specifcation at the system level
- TX Output Power up to +11 dBm
- Hardware Assisted MAC
  - Auto acknowledge
  - Auto retry
  - Channel access back-off
  - Automated Frame Check Sequence (FCS)
  - Automatic Address iltering
- Capable of Zigbee 3.0, Thread 1.4, and other 802.15.4 Protocol Stacks(1)
- SFD Detection; Spreading; De-spreading; Framing; CRC-16 Computation
- Independent TX/RX Buffers for Improved CPU Offloading
  - 128-byte TX and 128-byte RX frame buffer

**High Performance Peripherals**

- 16-Channel Direct Memory Access Controller (DMAC)
  - Built-in Cyclic Redundancy Check (CRC) with memory CRC generation and monitors hardware support
- One QSPI
  - Execute-In-Place (XIP) support
  - Dedicated AHB memory zone

**Low-Cost Controllerless (LCC) Graphics Solutions**

- Supports RGB332 8-bit color with 480x272 TFT display
- I2C for maxTouch Control
- DMA and GPIO Pins for 8-bit Pixel Transmission and Synchronization (2)

**Cryptography**

- Standard AES Encryption and Decryption with Key Sizes of 128 bits, 192 bits and 256 bits with Hardware Accelerator
- Standard SHA Hash Algorithms, Including SHA-256 and SHA-384 with Hardware Supported
- RSA Encryption and Decryption with Key Sizes of 1024 bits and 2048 bits
- Elliptic Curve Digital Signature Algorithm (ECDSA) Using All Supported NIST Curves
- NIST 800-90B Compliant True Random Number Generator (TRNG)
- Integrated Scatter Gather DMA


© 2025 Microchip Technology Inc. and its subsidiaries

**System Peripherals**

- 32-Channel Event System
  - All channels can be connected to any event generator
  - All channels provide a pure asynchronous path
  - Twelve channels support synchronous and re-synchronous
- Six Serial Communication Interfaces (SERCOM), each Configurable to Operate as:
  - Universal Synchronous Asynchronous Receiver Transmitter (USART) with full duplex and single-wire half duplex configuration
  - ISO7816
  - I²C up to 1 MHz
  - Serial Peripheral Interface (SPI)
  - Local Interconnect Network (LIN) Commander/Responder
  - RS485
  - SPI inter-byte space
- One SERCOM Configured as I²C-Only Interface
- Ten 16-Bit Timers/Counters (TC), Each Configurable as:
  - 16-bit TC with two compare/capture channels
  - 8-bit TC with two compare/capture channels
  - 32-bit TC with two compare/capture channels by paring two TCs
- Two 24-Bit Timer/Counters for Control (TCC) with Extended Functions:
  - Up to six 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 to increase resolution up to 5 bits and reduce quantization error
- One 16-Bit Timer/Counters for Control (TCC) with Extended Functions:
  - Up to two compare channels with optional complementary output
- 34 PWM Channels: 6 Channels x 2 from 24-bit TCC, 2 Channels from 16-bit TCC, and 10 x 2 Channels from TC
- 32-Bit Real Time Counter (RTCC) with Clock/Calendar Function
  - Up to four wake-up pins with tamper detection and debouncing filter
- Watchdog Timer (WDT) with Window Mode
- Deadman Timer (DMT)
- Frequency Meter (FREQM)
- Configurable Custom Logic (CCL) with Two Look-up Table (LUT)
- One 7-Bit General Purpose Digital-to-Analog Converter (DAC)
- Two 12-Bit, up to 1 Msps Analog-to-Digital Converter (ADC) SAR Core with up to 19 Analog Channels:
  - Differential and single-ended input
  - Automatic offset and gain error compensation
- Up to Two Analog Comparator (AC) with Window Compare Function
- Up to 18 Capacitive Voltage Divider (CVD) Channels for Touch Button Support (Using Shared ADC SAR Core)
- One Temperature Sensor (Die Temperature) Built into Wireless Subsystem
- Two Controller Area Network with Flexible Data Rate (CAN-FD) Ports with Dedicated Direct Memory Access (DMA) and 16 Buffers, 16 Filters and 3 Masks
  - ISO11898-1:2015 and CAN FD 1.0 compliant, supports up to 64 data bytes
  - Arbitration bit rate up to 1 Mbps
  - Data bit rate up to 10 Mbps
- Supports USB Full-Speed and Low-Speed Compliance
  - Port with up to eight dedicated DMA channels
- Ethernet MAC (ETH)
  - Compatible with IEEE Standard 802.3
  - 10/100Mbps Reduced Media Independent Interface (RMII) with dedicated DMA
  - 802.1AS and IEEE 1588-2008 Precision Time Protocol (PTP) compliant for precision clock synchronization and TSU support
  - IEEE802.3AZ/AF/PoE compliant for energy efficiency
  - Wake on Local Area Network (LAN) support
- One Quadrature Encoder Interface (QEI)

**Oscillators**

- 16 MHz, ±30 PPM Crystal/Resonator Oscillator or External Clock (POSC) for 2.4 GHz RF Transceiver
- Shared System Phase-Locked Loop (PLL) with RF Subsystem
- 50 MHz Ethernet Phase-Locked Loop (EPLL)
- 96 MHz USB Phase-Locked Loop (UPLL)
- 32.768 kHz Ultra-Low Power Internal Oscillator (LPRC)
- Higher Accuracy 32.768 kHz, ±250 ppm Clock Options
  - 32 kHz clock derived from POSC
  - 32.768 kHz crystal/resonator oscillator (SOSC)
  - External 32.768 kHz clock source
- 8 MHz Internal RC Oscillator (FRC)

**I/O**

- Peripheral Pin Select (PPS) Support
- 23 High-Current Sink/Source on Most I/O Pins
- Con igurable Open-Drain Output on Digital I/O Pins
- Up to 54 Programmable I/O Pins

**DC Speci ication**

- Electrostatic Discharge (ESD) Protection
  - Human Body Model (HBM): 2 kV (min)
  - Charged Device Model (CDM): 500V (min)
- Boot time[3][4]: ~7.1 ms

**Package**

- PIC32CX2501BZ62132
  - 132-pin QFN
  - Size: 10 mm x 10 mm x 0.9 mm


© 2025 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. Optional by firmware  configuration.
2. Detailed description is available in DS01387A and Microchip Graphics Suite (MGS) Display Con iguration and Drivers.
3. The numbers are design targets only and subject to change after device characterization is complete.
4. The time from Reset release until control is transferred to the Flash without code authentication.

## PIC32WM-BZ6 Module Features

The following section lists the PIC32WM-BZ6 Module-related features, which complement SoC features.

**PIC32WM-BZ6 Module Variants**

- PIC32WM-BZ6204 Based on PIC32CX2501BZ62132 SoC
  - PIC32WM-BZ6204UE (u.FL Antenna with Embedded Crypto)
  - PIC32WM-BZ6204UC (u.FL Antenna with Trust&GO

**Antenna**

- External Antenna

**Clock Management**

- Integrated 16 MHz POSC

**Advanced Analog**

- Two 12-Bit ADC up to 1 Msps Conversion Rate
  - Up to 19 analog channels (PIC32WM-BZ6204)
- CVD Touch Support Using Shared ADC, with up to 18 Rx/Tx Channels for Touch Measurement
- 7-Bit General Purpose DAC

**Input/Output**

- Up to 54 GPIO Pins (PIC32WM-BZ6204)

**Package and Operating Conditions**

- PIC32WM-BZ6204 Module Package:
  - 83-pin SMD package with Shield CAN
  - Size: 15.5 mm x 20.7 mm x 2.8 mm
- Operating Conditions:
  - 1.9-3.6V for variants
  - Operating temperature: -40ºC to +85ºC

**Certifications**

- PIC32WM-BZ6204 Module Certified to FCC, ISED, CE, UKCA, MIC, KCC, NCC and SRRC Radio Regulations
- Bluetooth SIG
- RoHS and REACH Compliant


© 2025 Microchip Technology Inc. and its subsidiaries

# Table of Contents


Introduction....1

PIC32CX-BZ6 SoC Family Features....2

PIC32WM-BZ6 Module Features....5

1. Ordering Information....17
   - 1.1. PIC32CX-BZ6 SoC Ordering Information....17
   - 1.2. PIC32WM-BZ6 Module Ordering Information....17

2. Configuration Summary....19

3. PIC32CX-BZ6 SoC Description....20
   - 3.1. PIC32CX-BZ6 SoC Block Diagram....20

4. PIC32WM-BZ6 Module Description....21
   - 4.1. Basic Connection Requirement....21
   - 4.2. PIC32WM-BZ6 Module Placement Guidelines....24
   - 4.3. PIC32WM-BZ6 Module Routing Guidelines....24
   - 4.4. PIC32WM-BZ6 Module RF Considerations....25
   - 4.5. PIC32WM-BZ6 Module Antenna Considerations....25
   - 4.6. PIC32WM-BZ6 Module Reflow Profile Information....27
   - 4.7. PIC32WM-BZ6 Module Assembly Considerations....28

5. Pinout and Signal Descriptions List....29

6. I/O Ports and Peripheral Pin Select (PPS)....33
   - 6.1. Overview....33
   - 6.2. Features....33
   - 6.3. Block Diagram....33
   - 6.4. Parallel I/O (PIO) Ports....34
   - 6.5. Peripheral Pin Select (PPS)....37
   - 6.6. Peripheral Multiplexing....55
   - 6.7. Function Priority for Device Pins....56
   - 6.8. Operation in Power Saving Modes....63
   - 6.9. Results of Various Resets....63
   - 6.10. Port Register Summary....64
   - 6.11. Register Description....72
   - 6.12. Peripheral Pin Select (PPS) Input Mapping Register Summary....143
   - 6.13. Peripheral Pin Select (PPS) Output Mapping Register Summary....152
   - 6.14. Register Description....163

7. Power Subsystem....166
   - 7.1. Block Diagram....166
   - 7.2. VDD Voltage Domain Overview....167
   - 7.3. $V_{DD-AON}$ Power Domain Overview....168
   - 7.4. $V_{DDBKUPCORE}$ Power Domain....168
   - 7.5. PMU Controller....168
   - 7.6. Voltage Regulators....168




© 2025 Microchip Technology Inc. and its subsidiaries

7.7. Power Supply Modes....168
7.8. Typical Power Supply Connection for SoC....170
7.9. Typical Power Supply Connection for PIC32WM-BZ6 Module....170
7.10. Power-Up Sequence....171

8. Product Memory Mapping Overview....172
8.1. Embedded Memories....173
8.2. Physical Memory Map....173
8.3. Boot ROM....173
8.4. Flash Memory Parameters....173
8.5. eFuse Memory....174
8.6. SRAM Memory Configuration....174
8.7. Boot Flash Device Configuration Word....176
8.8. Boot Flash Code Protection Register....178

9. Processor and Architecture....179
9.1. Cortex-M4 Processor....179
9.2. Nested Vector Interrupt Controller (NVIC)....182
9.3. High-Speed Bus System....187

10. Prefetch Cache (PCHE)....190
10.1. Overview....190
10.2. Features....190
10.3. Block Diagram....190
10.4. Product Dependencies....192
10.5. Prefetch Behavior....193
10.6. Configurations....193
10.7. Predictive Prefetch Behavior....194
10.8. Coherency Support....194
10.9. Effects of Reset....194
10.10. Error Conditions....195
10.11. Register Summary....196
10.12. Register Description....196

11. Cortex M Cache Controller (CMCC)....203
11.1. Overview....203
11.2. Features....203
11.3. Block Diagram....204
11.4. Signal Description....205
11.5. Product Dependencies....205
11.6. Functional Description....205
11.7. RAM Properties....208
11.8. Register Summary....209
11.9. Register Description....209

12. Secure Boot ROM....223
12.1. Overview....223
12.2. Features....223
12.3. Functional Description....223
12.4. Register Summary....230


© 2025 Microchip Technology Inc. and its subsidiaries

12.5. Register Description ..... 230
12.6. Application Transition ..... 231

13. eFuse Controller ..... 233
13.1. Overview ..... 233
13.2. Hardware Mode ..... 233
13.3. eFuse Programming ..... 233
13.4. eFuse Auto-Loading ..... 234
13.5. Security Keys ..... 236
13.6. Counters ..... 236
13.7. Register Summary ..... 238
13.8. Register Description ..... 238

14. Security Features ..... 242
14.1. Overview ..... 242
14.2. Features ..... 242
14.3. Reference Documentation ..... 242
14.4. Interface ..... 242
14.5. Description ..... 243
14.6. Register Summary ..... 244
14.7. Register Description ..... 244

15. Flash Controller (FC) ..... 246
15.1. Overview ..... 246
15.2. Features ..... 246
15.3. Functional Block Diagram ..... 247
15.4. Product Dependencies ..... 247
15.5. Flash Memory Addressing ..... 248
15.6. Memory Configuration ..... 248
15.7. Boot Flash Memory (BFM) Partitions ..... 249
15.8. Program Flash Memory (PFM) Partitions ..... 249
15.9. Error Correcting Code (ECC) and Flash Programming ..... 250
15.10. Interrupts ..... 250
15.11. Error Detection ..... 251
15.12. NVMKEY Register Unlocking Sequence ..... 252
15.13. DWord Programming ..... 253
15.14. Quad Double Word Programming ..... 254
15.15. Row Programming ..... 255
15.16. Page Erase ..... 256
15.17. Program Flash Memory (PFM) Erase ..... 258
15.18. Device Code Protection Bit (CP) ..... 259
15.19. Effects of Various Resets ..... 259
15.20. Register Summary ..... 260
15.21. Register Description ..... 262

16. Device Service Unit (DSU) ..... 275
16.1. Overview ..... 275
16.2. Features ..... 275
16.3. DSU Block Diagram ..... 275
16.4. Signal Description ..... 275


© 2025 Microchip Technology Inc. and its subsidiaries

16.5. Product Dependencies....276
16.6. Debug Operation....277
16.7. Chip Erase....278
16.8. Programming....279
16.9. Intellectual Property Protection....280
16.10. Device Identification....282
16.11. Functional Description....283
16.12. Register Summary....286
16.13. Register Description....288

17. Clock and Reset Unit (CRU)....316
17.1. Overview....316
17.2. Features....316
17.3. Clock System....317
17.4. Resets....327
17.5. Register Summary....334
17.6. Register Description....336

18. Power Management Unit (PMU)....365
18.1. Overview....365
18.2. Features....365
18.3. Functional Description....365
18.4. Register Summary....370
18.5. Register Description....370
18.6. Register Summary....374
18.7. Register Description....374

19. Watchdog Timer (WDT)....380
19.1. Overview....380
19.2. Features....380
19.3. Block Diagram....380
19.4. Watchdog Timer Operation....380
19.5. Interrupt and Reset Generation....384
19.6. Operation in Debug and Power-Saving Modes....384
19.7. Effects of Various Resets....385
19.8. Register Summary....386
19.9. Register Description....386

20. Deadman Timer (DMT)....388
20.1. Overview....388
20.2. Features....388
20.3. Block Diagram....388
20.4. DMT Operation....389
20.5. Register Summary....392
20.6. Register Description....393

21. RAM Error Correction Code (RAMECC)....402
21.1. Overview....402
21.2. Features....402
21.3. Block Diagram....402


© 2025 Microchip Technology Inc. and its subsidiaries

21.4. Signal Description....402
21.5. Product Dependencies....403
21.6. Functional Description....404
21.7. Register Summary....405
21.8. Register Description....405

22. System Configuration and Register Locking (CFG)....412
22.1. Overview....412
22.2. Features....412
22.3. Modes of Operation....413
22.4. Locking and Unlocking the System Configuration Registers....414
22.5. NMI Events....414
22.6. Register Locking....414
22.7. Effects of Various Resets....415
22.8. Register Summary....417
22.9. Register Description....418

23. Peripheral Module Disable (PMD)....462
23.1. Overview....462
23.2. Enabling Peripherals....462
23.3. Controlling Configuration Changes....462
23.4. PMD Register Summary....463
23.5. Register Description....463

24. Peripheral Access Controller (PAC)....471
24.1. Overview....471
24.2. Features....471
24.3. Block Diagram....471
24.4. Product Dependencies....471
24.5. Functional Description....473
24.6. Register Summary....476
24.7. Register Description....477

25. Real-Time Counter and Calendar (RTCC)....504
25.1. Overview....504
25.2. Features....504
25.3. Block Diagram....505
25.4. Signal Description....506
25.5. Product Dependencies....506
25.6. Functional Description....507
25.7. Register Summary - Mode 0 - 32-Bit Counter....519
25.8. Register Description - Mode 0 - 32-Bit Counter....520
25.9. Register Summary - Mode 1 - 16-Bit Counter....545
25.10. Register Description - Mode 1 - 16-Bit Counter....546
25.11. Register Summary - Mode 2 - Clock/Calendar....569
25.12. Register Description - Mode 2 - Clock/Calendar....570

26. Direct Memory Access Controller (DMAC)....594
26.1. Overview....594
26.2. Features....594


© 2025 Microchip Technology Inc. and its subsidiaries

26.3. Block Diagram ..... 596
26.4. Signal Description ..... 596
26.5. Product Dependencies ..... 596
26.6. Functional Description ..... 597
26.7. Register Summary ..... 623
26.8. Register Description ..... 627
26.9. Register Summary - SRAM ..... 658
26.10. Register Description - SRAM ..... 658

27. External Interrupt Controller (EIC) ..... 665
27.1. Overview ..... 665
27.2. Features ..... 665
27.3. Block Diagram ..... 665
27.4. Signal Description ..... 665
27.5. Product Dependencies ..... 666
27.6. Functional Description ..... 667
27.7. Register Summary ..... 674
27.8. Register Description ..... 674

28. Configurable Custom Logic (CCL) ..... 689
28.1. Overview ..... 689
28.2. Features ..... 689
28.3. Block Diagram ..... 690
28.4. Signal Description ..... 690
28.5. Product Dependencies ..... 690
28.6. Functional Description ..... 691
28.7. Register Summary ..... 701
28.8. Register Description ..... 701

29. Frequency Meter (FREQM) ..... 706
29.1. Overview ..... 706
29.2. Features ..... 706
29.3. Block Diagram ..... 706
29.4. Signal Description ..... 706
29.5. Product Dependencies ..... 706
29.6. Functional Description ..... 708
29.7. Register Summary ..... 711
29.8. Register Description ..... 711

30. Event System (EVSYS) ..... 721
30.1. Overview ..... 721
30.2. Features ..... 721
30.3. Block Diagram ..... 722
30.4. Product Dependencies ..... 722
30.5. Functional Description ..... 723
30.6. Register Summary ..... 731
30.7. Register Description ..... 735

31. Serial Communication Interface (SERCOM) ..... 753
31.1. Overview ..... 753


© 2025 Microchip Technology Inc. and its subsidiaries

31.2. Features ... 753
31.3. Block Diagram ... 754
31.4. Signal Description ... 754
31.5. Product Dependencies ... 754
31.6. Functional Description ... 755

32. SERCOM Synchronous and Asynchronous Receiver and Transmitter (SERCOM USART) ... 760
32.1. Overview ... 760
32.2. USART Features ... 760
32.3. Block Diagram ... 761
32.4. Signal Description ... 761
32.5. Product Dependencies ... 761
32.6. Functional Description ... 763
32.7. Register Summary ... 780
32.8. Register Description ... 780

33. SERCOM Serial Peripheral Interface (SERCOM SPI) ... 806
33.1. Overview ... 806
33.2. Features ... 806
33.3. Block Diagram ... 806
33.4. Signal Description ... 807
33.5. Product Dependencies ... 807
33.6. Functional Description ... 808
33.7. Register Summary ... 819
33.8. Register Description ... 819

34. SERCOM Inter-Integrated Circuit (SERCOM $I^2C$) ... 840
34.1. Overview ... 840
34.2. Features ... 840
34.3. Block Diagram ... 841
34.4. Signal Description ... 841
34.5. Product Dependencies ... 841
34.6. Functional Description ... 843
34.7. Register Summary - I2C Client ... 861
34.8. Register Description - $I^2C$ Client ... 861
34.9. Register Summary - I2C Host ... 878
34.10. Register Description - $I^2C$ Host ... 878

35. Quad Serial Peripheral Interface (QSPI) ... 897
35.1. Overview ... 897
35.2. Features ... 897
35.3. Block Diagram ... 898
35.4. Signal Description ... 898
35.5. Product Dependencies ... 898
35.6. Functional Description ... 900
35.7. Register Summary ... 918
35.8. Register Description ... 919

36. Analog-to-Digital Converter (ADC) ... 940
36.1. Overview ... 940


© 2025 Microchip Technology Inc. and its subsidiaries

36.2. Features ... 940
36.3. Block Diagram ... 941
36.4. ADC Operation ... 941
36.5. ADC Module Configuration ... 946
36.6. Additional ADC Functions ... 952
36.7. Interrupts ... 959
36.8. Power-Saving Modes of Operation ... 961
36.9. Effects of Reset ... 963
36.10. Transfer Function ... 963
36.11. ADC Sampling Requirements ... 963
36.12. Register Summary ... 965
36.13. Register Description ... 971

37. Automated Touch Sensing using the Enhance Capacitive Voltage Divider (CVD) ... 1022
37.1. Overview ... 1022
37.2. Features ... 1022
37.3. Configuration ... 1022
37.4. Block Diagram ... 1023
37.5. CVD Module Operation ... 1023
37.6. Register Summary ... 1027
37.7. Register Description ... 1029

38. Analog Comparators (AC) ... 1070
38.1. Overview ... 1070
38.2. Features ... 1070
38.3. Block Diagram ... 1071
38.4. Product Dependencies ... 1071
38.5. Functional Description ... 1073
38.6. Register Summary ... 1083
38.7. Register Description ... 1083

39. Digital-to-Analog Converter (DAC) ... 1100
39.1. Overview ... 1100
39.2. Features ... 1100
39.3. Block Diagram ... 1100
39.4. DAC Timing Diagram ... 1100
39.5. Functional Description ... 1100
39.6. Register Summary ... 1102

40. Timer/Counter (TC) ... 1106
40.1. Overview ... 1106
40.2. Features ... 1106
40.3. Block Diagram ... 1107
40.4. Signal Description ... 1107
40.5. Product Dependencies ... 1108
40.6. Functional Description ... 1109
40.7. Register Summary - 8-bit Mode ... 1125
40.8. Register Description - 8-Bit Mode ... 1125
40.9. Register Summary - 16-bit Mode ... 1148
40.10. Register Description - 16-Bit Mode ... 1148


© 2025 Microchip Technology Inc. and its subsidiaries

40.11. Register Summary - 32-bit Mode ..... 1169
40.12. Register Description - 32-Bit Mode ..... 1169

41. Timer/Counter for Control Applications (TCC) ..... 1190
41.1. Overview ..... 1190
41.2. Features ..... 1190
41.3. Block Diagram ..... 1191
41.4. Signal Description ..... 1191
41.5. Product Dependencies ..... 1192
41.6. Functional Description ..... 1193
41.7. Register Summary ..... 1227
41.8. Register Description ..... 1229

42. 802.15.4 Bluetooth® Radio Subsystem ..... 1269
42.1. Overview ..... 1269
42.2. Features ..... 1269
42.3. Wireless Subsystem Top Level Diagram ..... 1271
42.4. Analog RF Front-End ..... 1271
42.5. Digital Front-end ..... 1271
42.6. Bluetooth Low Energy Link Controller ..... 1271
42.7. 802.15.4 Subsystem ..... 1272
42.8. Radio Arbiter ..... 1272
42.9. Register Banks ..... 1272
42.10. Coexistence Interface ..... 1272

43. Controller Area Network (CAN) ..... 1273
43.1. Overview ..... 1273
43.2. Features ..... 1273
43.3. Block Diagram ..... 1274
43.4. Signal Description ..... 1274
43.5. Peripheral Dependencies ..... 1274
43.6. Functional Description ..... 1275
43.7. Register Summary ..... 1304

44. Universal Serial Bus (USB) ..... 1382
44.1. Overview ..... 1382
44.2. Features ..... 1382
44.3. Block Diagram ..... 1383
44.4. USB OTG Control Registers ..... 1384

45. Ethernet Media Access Controller (ETH) ..... 1423
45.1. Overview ..... 1423
45.2. Features ..... 1423
45.3. Block Diagram ..... 1424
45.4. Signal Interface ..... 1424
45.5. Functional Description ..... 1425
45.6. Programming Interface ..... 1449
45.7. Register Summary ..... 1453

46. Quadrature Encoder Interface (QEI) ..... 1584


© 2025 Microchip Technology Inc. and its subsidiaries

46.1. Overview....1584
46.2. Features....1586
46.3. Block Diagram....1587
46.4. Module Description....1587
46.5. QEI Operation in Power-Saving Modes....1595
46.6. QEI Registers....1595

47. Electrical Characteristics....1615
47.1. Absolute Maximum Ratings....1615
47.2. DC Electrical Characteristics....1616
47.3. Thermal Specifications....1616
47.4. Power Supply DC Module Electrical Specifications....1616
47.5. Active Current Consumption DC Electrical Specifications....1621
47.6. Idle Current Consumption DC Electrical Specifications....1621
47.7. Standby/Sleep Current Consumption DC Electrical Specifications....1622
47.8. Deep Sleep Current Consumption DC Electrical Specifications....1623
47.9. XDS (Extreme Deep Sleep) Current Consumption DC Electrical Specifications....1624
47.10. Wake-Up Timing from Low Power Modes AC Electrical Specifications....1625
47.11. I/O PIN AC/DC Electrical Specifications....1625
47.12. Maximum Clock Frequencies Electrical Specifications....1628
47.13. External XTAL and Clock (POSC) AC Electrical Specifications....1629
47.14. XOSC32 (SOSC) AC Electrical Specifications....1631
47.15. Low Power Internal 32 kHz RC Oscillator AC Electrical Specifications....1633
47.16. FRC AC Electrical Specifications....1633
47.17. RFPLL (RF/System Phase Locked Loop) AC Electrical Specifications....1633
47.18. DAC Module Electrical Specifications....1634
47.19. ADC Electrical Specifications....1635
47.20. Analog Comparator AC Electrical Specifications....1637
47.21. SPI Module Electrical Specifications....1638
47.22. UART AC Electrical Specifications....1642
47.23. $I^2C$ Module Electrical Specifications....1642
47.24. QSPI Module Electrical Specifications....1646
47.25. CANx Module AC Electrical Specifications....1648
47.26. TCx Timer Capture Module AC Electrical Specifications....1648
47.27. TCCx Timer Capture Module AC Electrical Specifications....1649
47.28. USB AC Electrical Specifications....1650
47.29. FLASH NVM AC Electrical Specifications....1651
47.30. GMAC Electrical Specifications....1652
47.31. Frequency Meter AC Electrical Specifications....1652
47.32. Quadrature Encoder Interface AC Electrical Specifications....1653
47.33. SWD 2-Wire AC Electrical Specifications....1654
47.34. Bluetooth Low Energy RF Characteristics....1654
47.35. 802.15.4 RF Characteristics....1657

48. Packaging Information....1660
48.1. PIC32CX-BZ6 SoC Packaging Information....1660
48.2. PIC32WM-BZ6 Module Packaging Information....1664

49. Appendix A: Regulatory Approval....1668


© 2025 Microchip Technology Inc. and its subsidiaries

49.1. United States....1668
49.2. Canada....1669
49.3. Europe....1671
49.4. UKCA (UK Conformity Assessed)....1672
49.5. Other Regulatory Information....1673

50. Document Revision History....1674

Microchip Information....1675
Trademarks....1675
Legal Notice....1675
Microchip Devices Code Protection Feature....1675


© 2025 Microchip Technology Inc. and its subsidiaries

# 1. Ordering Information

This chapter provides the ordering information of the PIC32CX-BZ6 SoC and the PIC32WM-BZ6 Module.

## 1.1. PIC32CX-BZ6 SoC Ordering Information

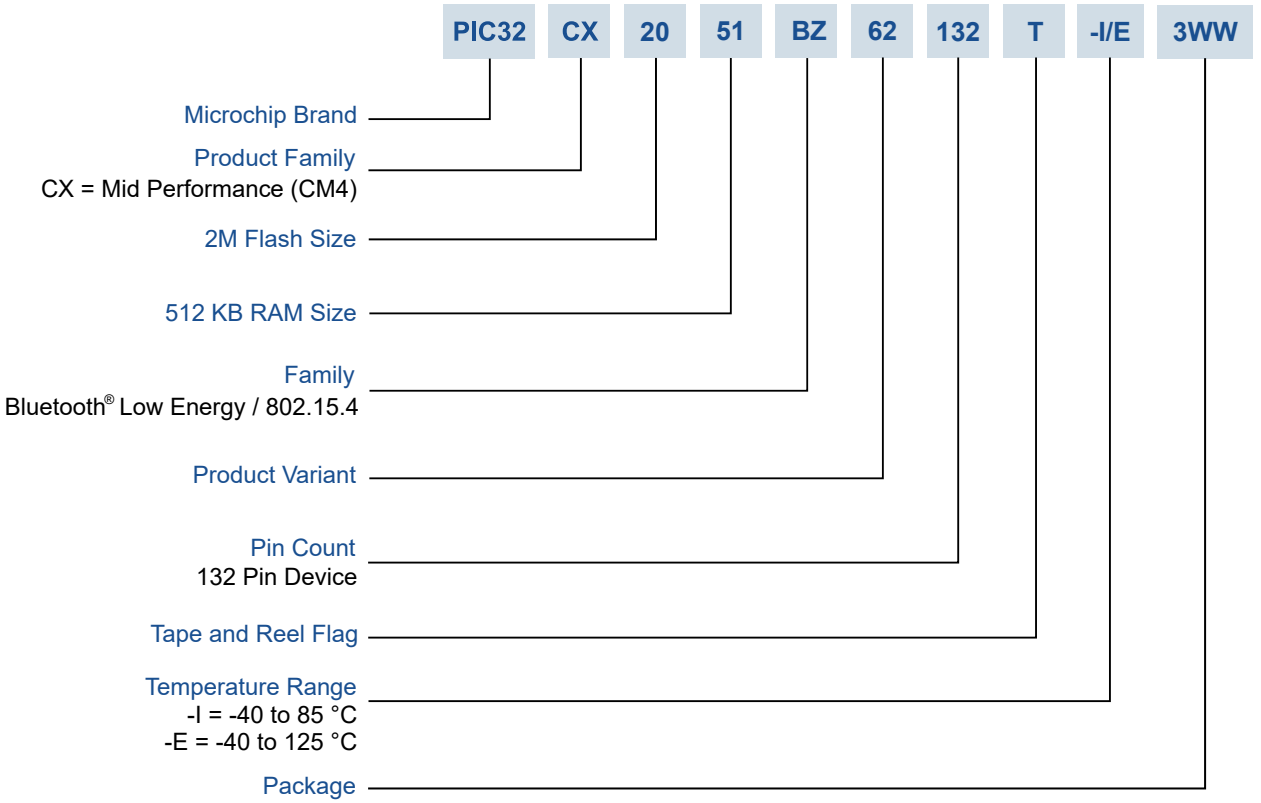
The following table describes the ordering information of the PIC32CX-BZ6 SoC.

**Table 1-1.** PIC32CX-BZ6 SoC Ordering Details

| SoC Name | Pin and Package | Description | Ordering Code |
|---|---|---|---|
| PIC32CX2051BZ62132 | 132-pin and DQFN (10 mm x 10 mm x 1 mm) | 32-bit ARM® Cortex™-M4 with MCU Bluetooth® / 802.15.4 Connectivity | PIC32CX2051BZ62132-I/3WW<br>PIC32CX2051BZ62132-E/3WW<br>PIC32CX2051BZ62132T-I/3WW<br>PIC32CX2051BZ62132T-E/3WW |

The following figure illustrates the details of the PIC32CX-BZ6 ordering information.

**Figure 1-1.** PIC32CX-BZ6 Ordering Information




## 1.2. PIC32WM-BZ6 Module Ordering Information

The following table describes the ordering information of the PIC32WM-BZ6 Module.

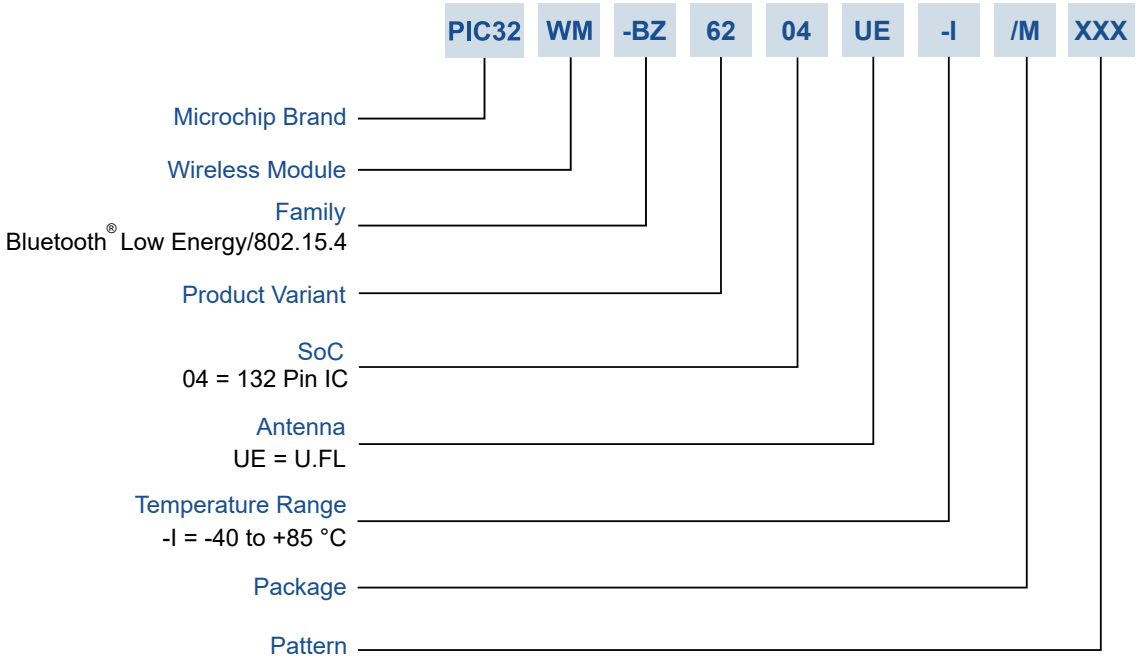

© 2025 Microchip Technology Inc. and its subsidiaries

**Table 1-2.** PIC32WM-BZ6 Module Ordering Details

| Model Name | Module SoC | Description | Regulatory Certification | Ordering Code |
|---|---|---|---|---|
| PIC32WM-BZ6204UE | PIC32CX2051BZ62132-V/ZWX | Wireless module Bluetooth LE/802.15.4 WBZ653UE-I, with U.FL connector, dimension is 19 mm x 25 mm x 2.75 mm | FCC, ISED, CE | PIC32WM-BZ6204UE-I |
| PIC32WM-BZ6204UC | PIC32CX2051BZ62132-V/ZWX | Wireless module Bluetooth LE/802.15.4 WBZ653UC-I, with U.FL connector, Trust & Go | FCC, ISED, CE | PIC32WM-BZ6204UC-I |

The following figure illustrates the details of the PIC32WM-BZ6 Module ordering information.

**Figure 1-2.** PIC32WM-BZ6 Module Ordering Information





© 2025 Microchip Technology Inc. and its subsidiaries

# 2. Configuration Summary

**Table 2-1.** PIC32CX-BZ6 and PIC32WM-BZ6 Family Features - Memory and Package

| Device | Boot ROM Memory (KB) | Program Memory (KB) | Data Memory (KB) | EFUSE (Bits) | Pins | Package |
|---|---|---|---|---|---|---|
| PIC32CX2051BZ62132 | 64 | 2048 | 512 | 3072 | 132 | DQFN |
| PIC32CX2051BZ62064 | | | | | 64 | VQFN |
| PIC32CX2051BZ66048 | | | | | 48 | QFN |

**Table 2-2.** PIC32CX-BZ6 and PIC32WM-BZ6 Family Features - Peripherals

| Device | Peripherals | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SERCOM | TC (16-bit)/Compare (Channels) | TCC (24-bit/16-bit) | QEI | QSPI | USB2.0 OTG - FS / LS | CAN-FD | EMAC (10/100) | IEEE 1588 PTP | IEEE 802.3az (EEE) | DMA Channels | RTCC | CCL/LUT | WDT | DMT | Frequency Measurement | Event System (Channels) | External Interrupt Lines | GPIO Pins |
| PIC32CX2051BZ62132 | 7 | 10 | 2/1 | 1 | Y | 1 | 2 | 1 | 1 | 1 | 16 | Y | 1/2[1] | Y | Y | Y | 32 | 4 | 54 |
| PIC32CX2051BZ62064 | | | | | | | | | | | | | | | | | | | 42 |
| PIC32CX2051BZ66048 | 5 | | | | | | | 0 | 0 | 0 | | | | | | | | | 29 |

**Note:**

1. CCL/LUT = 1/2, means 1 CCL with 2 LUT

**Table 2-3.** PIC32CX-BZ6 and PIC32WM-BZ6 Family Features - Analog, Security and Wireless

| Device | Analog | | | | | | Security | | | | | | | Wireless | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Analog Comparators/Channels | ADC 12-bit SAR Cores | ADC (Channels) | CVD | DAC | Temperature Sensor | AES | TRNG | SHA | Public Key Cryptography | ICM (SHA1/SHA224/SHA256) | Secure Hash Function | Root-of-Trust | Max TX Power (dBm) | Bluetooth® | 802.15.4 | BT/802.15.4 Co-existence (Radio Arbiter) |
| PIC32CX2051BZ62132 | Y | 2 | 19 | 18 | 1 | Y | Y | Y | Y | Y | Y | Y | Y | 10 | Y | Y | Y |
| PIC32CX2051BZ62064 | | | 14 | 13 | | | | | | | | | | | | | |
| PIC32CX2051BZ66048 | | | 9 | 8 | | | | | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

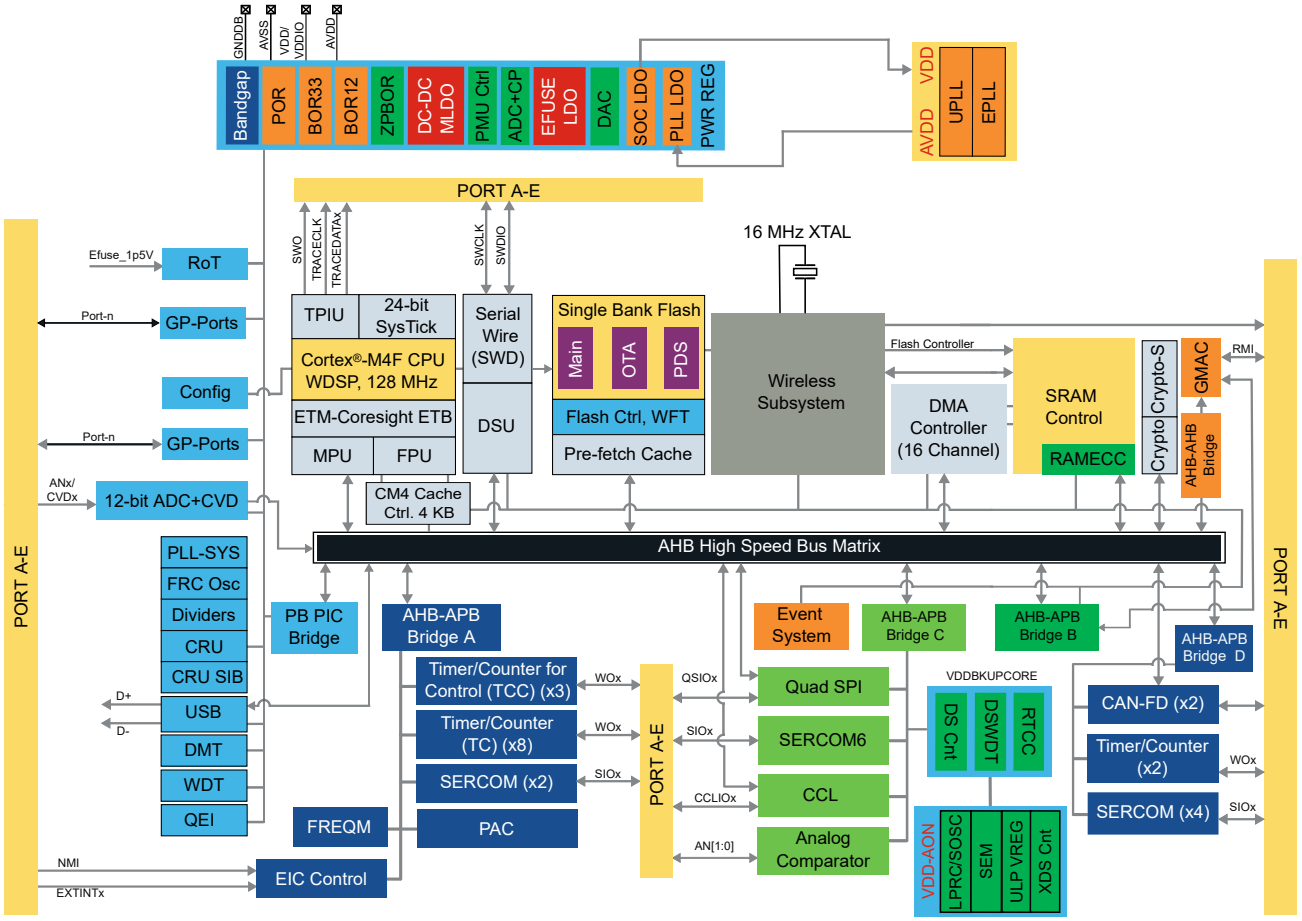
# 3. PIC32CX-BZ6 SoC Description

This chapter illustrates the block diagram of the PIC32CX-BZ6 SoC.

## 3.1. PIC32CX-BZ6 SoC Block Diagram

The following figure illustrates the block diagram of the core and peripheral modules in the PIC32CX-BZ6 SoC.

**Figure 3-1.** PIC32CX-BZ6 SoC Block Diagram





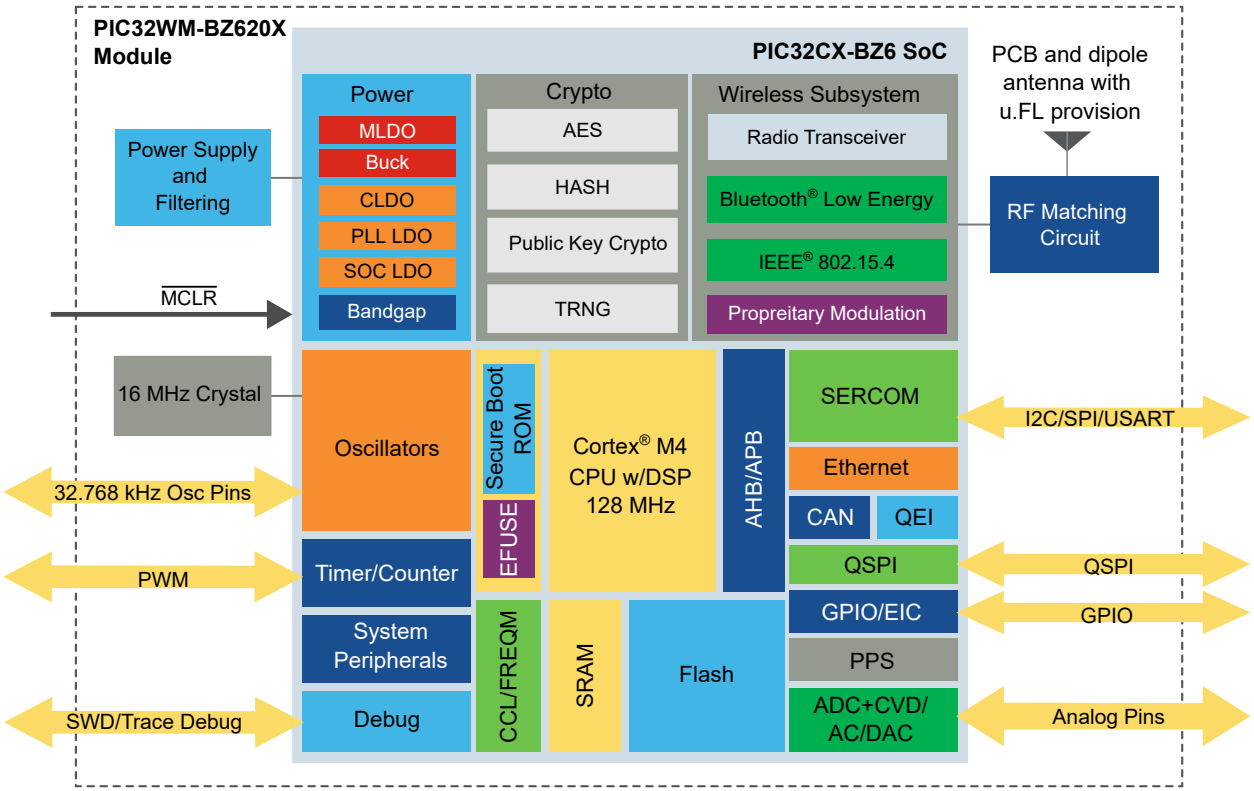
© 2025 Microchip Technology Inc. and its subsidiaries

# 4. PIC32WM-BZ6 Module Description

The PIC32WM-BZ6 modules are fully certified wireless modules built around PIC32CX-BZ6 SoC. The PIC32WM-BZ6 modules integrate a 16 MHz crystal, in addition to the circuits for power supply decoupling, RF matching and the U.FL connector for external antenna (PIC32WM-BZ6UE).

The operating voltage range for the PIC32WM-BZ6 Module is 1.9-3.6V. The following figure illustrates the PIC32WM-BZ6 Module block diagram and various peripherals supported by the module.

**Figure 4-1.** PIC32WM-BZ6 Module Block Diagram

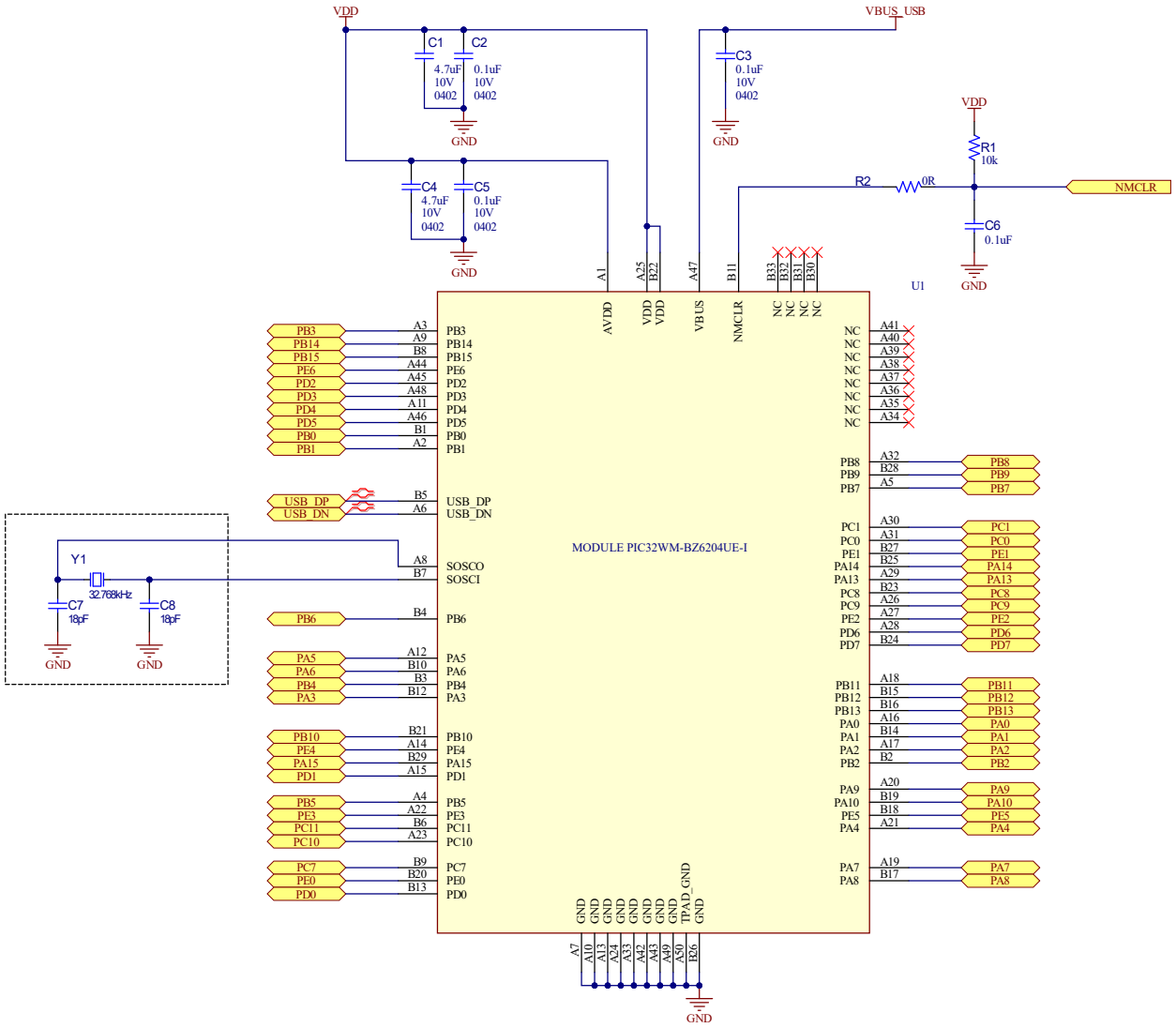



## 4.1. Basic Connection Requirement

The PIC32WM-BZ6 modules require attention to a minimal set of device pin connections before proceeding with development.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 4-2.** Module Basic Connection and Interface Diagram for PIC32WM-BZ6204 Module



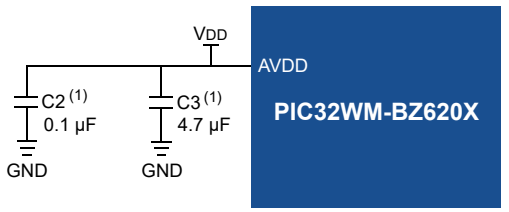

### 4.1.1. Power Pins

It is recommended that a bulk and a decoupling capacitor be added at the input supply pin ($V_{DD}$, $A_{VDD}$ and GND pins) of the PIC32WM-BZ6 Module.

- The recommendation is to have a 4.7 µF on the $A_{VDD}$ pin, 4.7 µF and a 0.1 µF on the $V_{DD}$ pin.
- The value of the capacitors are based on typical application requirements and are the minimum recommended values. Depending on the application requirement (in other words, a noisy power line or other known noise sources), the user can adjust the values of the capacitor to provide a clean supply to the module.
- Place all the capacitors close to the module power supply pins.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 4-3.** Recommended Module Power Supply Connections




**Notes:**

1. The value of the C2 and C3 capacitors can vary based on the application requirement.
2. Place the C2 and C3 capacitors close to the module pin.

### 4.1.2. Master Clear ($\overline{MCLR}$) Pin

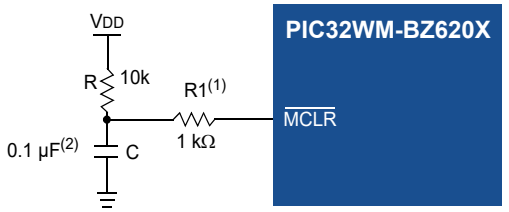
The $\overline{MCLR}$ pin provides two specific device functions:

- Device Reset
- Device programming and debugging

Pulling the $\overline{MCLR}$ pin low, generates a device Reset. Module Basic Connection and Interface Diagram illustrates a typical $\overline{MCLR}$ circuit (see *Module Basic Connection and Interface Diagrams* in the *Basic Connection Requirement* from Related Links).

The PIC32WM-BZ6 module has sufficient filtering (0.1 µF) and pull-up (10k) on the Reset line. On a typical application, there is no need for extra filtering on this pin.

**Figure 4-4.** Example of $\overline{MCLR}$ Pin Connections




**Notes:**

1. 470Ω ≤ R1 ≤ 1 kΩ limits any current flowing into $\overline{MCLR}$ from the external capacitor C, in the event of $\overline{MCLR}$ pin breakdown, due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS). Ensure that the $\overline{MCLR}$ pin $V_{IH}$ and $V_{IL}$ specifications are met without interfering with the Debug/ Programmer tools.
2. The capacitor can be sized to prevent unintentional Resets from brief glitches or to extend the device Reset period during POR.

**Related Links**

Basic Connection Requirement

### 4.1.3. SWD Lines

For SWD programming and debugging purposes, use the CM4_SWCLK, CM4_SWDIO and CM4_SWO pins. The recommendation is to use the CM4_SWCLK and CM4_SWDIO pins for the PIC32WM-BZ6 Module for SWD as the default configuration (CM4_SWO can be optional).

Keep the trace length between the SWD pins of the PIC32WM-BZ6 Module and the SWD header as short as possible. If the SWD connector is expected to experience an ESD event, a series resistor is recommended with the value in the range of a few tens of Ωs, not to exceed 100Ω.


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:** Provide an option for adding an external pull-up on SWD lines.

#### 4.1.4. Unused I/O Pins

The recommendation is not to allow the unused I/O pins to float as inputs. The user can configure them as inputs and pulled up. Alternatively, depending on the application, they can be pulled down as well.

##### 4.1.4.1. GPIO Pins/PPS Functions

Most of the PIC32WM-BZ6 Module pins can be configured as GPIOs pins or for PPS functionality. To find the functionality supported by each of these GPIOs, see *I/O Ports and Peripheral Pin Select (PPS)* from Related Links. The recommendation is to add a series resistor on the host board for all GPIOs, mainly critical high-frequency pins and clocks for EMI considerations. The value of the series resistor depends on the actual pin configuration. The user must place these resistors close to the module. Example of Host Board on Top Layer illustrates the placement of the series resistor; see the *Example of Host Board on Top Layer* figure in the *PIC32WM-BZ6 Module Routing Guidelines* from Related Links.

**Related Links**

I/O Ports and Peripheral Pin Select (PPS)

## 4.2. PIC32WM-BZ6 Module Placement Guidelines

- For the best PCB antenna performance, the PIC32WM-BZ6 Module must be placed at the edge of the host board.
- A low-impedance ground plane for the PIC32WM-BZ6 Module ensures the best radio performance (best range and lowest noise). The ground plane can be extended beyond the minimum recommendation as required for the host board EMC and noise reduction.
- Exposed GND pads on the bottom of the PIC32WM-BZ6 Module must be soldered to the host board (see *Example of Host Board on Top Layer* figure in the *PIC32WM-BZ6 Module Routing Guidelines* from Related Links).
- A PCB cutout or a copper keepout is required under RF test point. See *PIC32WM-BZ6 Module Packaging Information* from Related Links.
- Copper keepout areas are required on the top layer under voltage test points. See *PIC32WM-BZ6 Module Packaging Information* from Related Links.
- On the other hand, the entire region except the exposed ground paddle can be solder-masked.

**Related Links**

PIC32WM-BZ6 Module Routing Guidelines

## 4.3. PIC32WM-BZ6 Module Routing Guidelines

- Use the multi-layer host board for routing signals on the inner layer and the bottom layer.
- The top layer (underneath the module) of the host board must be ground with as many GND vias as possible.
- Avoid fan-out of the signals under the module or antenna area. Use a via to fan-out signals to the edge of the PIC32WM-BZ6 Module.
- For a better GND connection to the PIC32WM-BZ6 Module, solder the exposed GND pads of the PIC32WM-BZ6 Module on the host board.
- For the module GND pad, use a GND via of a minimum 10 mil (hole diameter) for good ground to all the layers and thermal conduction path.
- The recommendation is to have a series resistor on the host board for all GPIOs. Place these resistors close to the PIC32WM-BZ6 Module.
- Place the SOSC crystal (32.768 kHz) on the host board close to the PIC32WM-BZ6 Module and follow the shortest trace routing length with no vias.


© 2025 Microchip Technology Inc. and its subsidiaries

- USB differential pair signals are 90Ω impedance controlled on the PIC32WM-BZ6 Module PCB and the same must be followed on the host board.

## 4.4. PIC32WM-BZ6 Module RF Considerations

The product design, environment and application significantly affect the overall performance of the system. The product designer must ensure system-level shielding (if required) and verify the performance of the product features and applications.

The following are the guidelines to consider for optimal RF performance:

- Position the PIC32WM-BZ6 module in a noise-free RF environment; keep it far away from high-frequency clock signals and any other sources of RF energy.
- Do not shield the antenna by any metal objects.
- The power supply must be clean and noise-free.
- Ensure that the width of the traces routed to GND, VDD rails are sufficiently large for handling peak TX current consumption.
  **Note:** The PIC32WM-BZ6 module includes RF shielding on top of the board as a standard feature.

## 4.5. PIC32WM-BZ6 Module Antenna Considerations

### 4.5.1. External Antenna Placement Recommendations

The user must apply the following recommendations for the placement of the antenna and its cable:

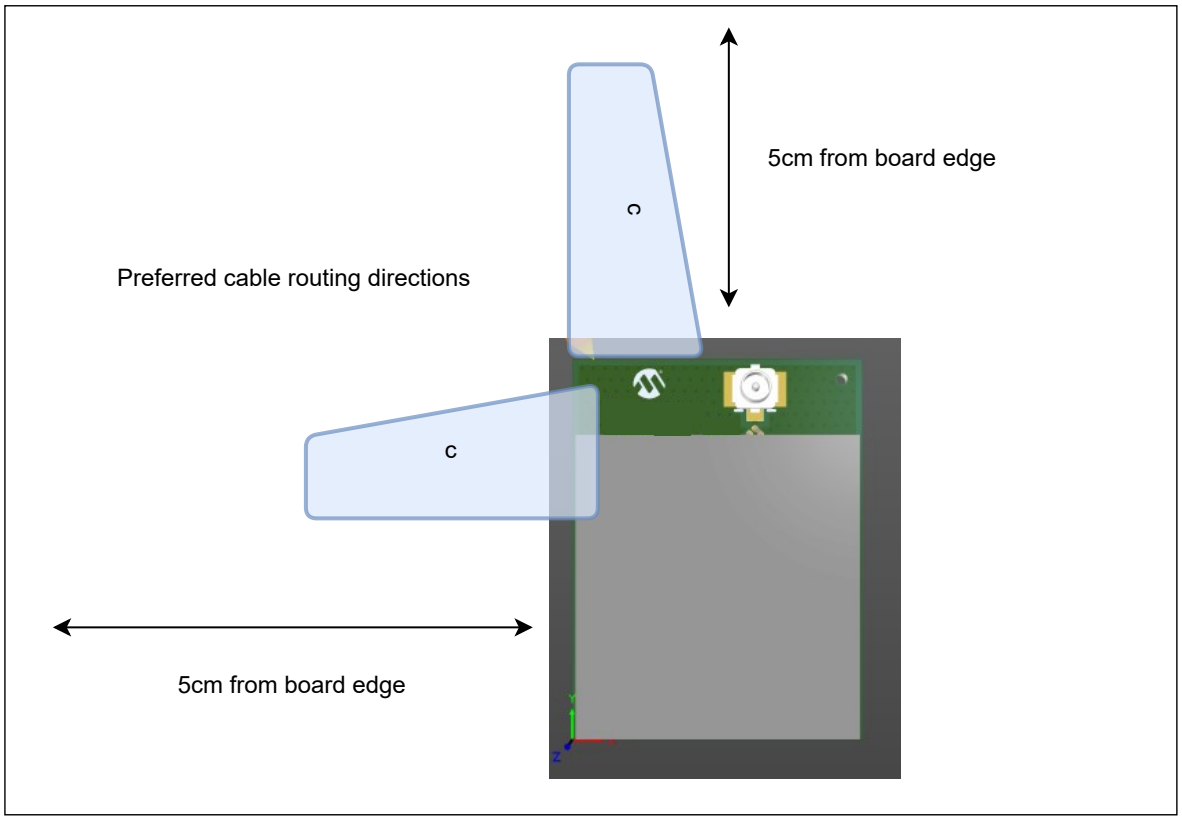
- Do not route the antenna cable over circuits generating electrical noise on the host board or alongside or underneath the module. The recommendation is to route the cable straight out of the module.
- Do not place the antenna in direct contact with or in close proximity to the plastic casing/objects. (Except when the selected antenna specifically recommends it.)
- Do not enclose the antenna within a metal shield.
- The user must keep any components capable of radiating noise, signals or harmonics in the 2.4-2.5 GHz frequency range away from the antenna, and, if feasible, provide shielding for such components. Any noise radiated from the host board in this frequency band degrades the sensitivity of the module.
- Place the antenna at a distance greater than 5 cm away from the module. The following figure illustrates the antenna keep-out area where the antenna must not be placed.

These recommendations are based on an open-air measurement and do not take into account any metal shielding of the customer end product. When a metal enclosure is used, the antenna can be located closer to the PIC32WM-BZ6 Module.

The following figure illustrates an indication on how to route the antenna cable depending on the location of the antenna with respect to the PIC32WM-BZ6 PCB. There are two possible options for the optimum routing of the cable.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 4-5.** PIC32WM-BZ6 Antenna Placement Guidelines




**Note:** These are generic guidelines and the recommendation is that the customers can check and fine-tune the antenna positioning in the final host product based on RF performance.

#### 4.5.1.1. External Antennas

The PIC32WM-BZ6204UE/PIC32WM-BZ6204UC Modules have a small surface mount U.FL connector for an external antenna connection. The choice of antenna is limited to the antenna types that the module is tested and approved for regulatory certification.

The PIC32WM-BZ6204UE/PIC32WM-BZ6204UC Modules are approved to use with the antennas listed in the following table. It is permissible to use a different antenna, provided it is the same antenna type, has the same antenna gain (equal or less than) and similar in-band and out-of-band characteristics are present (refer to antenna specification sheet for cutoff frequencies).

If other antenna types are used, the OEM installer must conduct the necessary assessments and authorize the antenna with the respective regulatory agencies and ensure compliance.

**Table 4-1.** Approved External Antenna List with Antenna Gain for PIC32WM-BZ6204UE/PIC32WM-BZ6204UC Module

| Antenna No. | Part Number | Manufacturer | Antenna Gain at 2.4G (dBi) | Antenna Type | Cable Length (mm) | Regulatory Certification | |
|---|---|---|---|---|---|---|---|
| | | | | | | FCC/ ISED(2)(3) | CE |
| 1 | ANT-2.4-LPW-125 | Linx Technologies | 2.8 | Dipole | 125 | x | x |
| 2 | RFA-02-L2H1 | Aristotle | 2 | Dipole | 150 | x | x |
| 3 | RFA-02-C2H1-D034 | Aristotle | 2 | Dipole | 150 | x | x |
| 4 | RFA-02-D3 | Aristotle | 2 | Dipole | 150 | x | x |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 4-1.** Approved External Antenna List with Antenna Gain for PIC32WM-BZ6204UE/PIC32WM-BZ6204UC Module (continued)

| Antenna No. | Part Number | Manufacturer | Antenna Gain at 2.4G (dBi) | Antenna Type | Cable Length (mm) | Regulatory Certification | |
|---|---|---|---|---|---|---|---|
| | | | | | | FCC/ ISED(2)(3) | CE |
| 5 | RFDPA870910IMAB308 | Walsin | 2.04 | Dipole | 100 | x | x |
| 6 | W1049B030 | Pulse | 2.85 | Dipole | 76 | x | x |
| 7 | RFDPA870920IMLB301 | Walsin | 1.84 | Dipole | 200 | x | x |
| 8 | RFDPA870920IMAB302 | Walsin | 1.82 | Dipole | 200 | x | x |
| 9 | RFDPA870920IMAB305 | Walsin | 1.82 | Dipole | 200 | x | x |
| 10 | RFDPA870915IMAB306 | Walsin | 1.82 | Dipole | 150 | x | x |
| 11 | FXP840.07.0055B | TAOGLAS | 3.35 | Flex PCB | 55 | x | x |
| 12 | YF0026AA | Quectel | 2.1 | Flex PCB | 100 | x | x |
| 13 | FXP70.07.0053A | TAOGLAS | 1.1 | Flex PCB | 53 | x | x |
| 14 | RFA-02-P05-D034 | Aristotle | 2 | PCB | 150 | x | x |
| 15 | RFA-02-P33-D034 | Aristotle | 2 | PCB | 150 | x | x |

**Notes:**

1. 'x' denotes the antennas covered under the certi ication.
2. If the end product using the module is designed to have an antenna port that is accessible to the end user, a unique (non-standard) antenna connector (as permissible by FCC) must be used (for example, RP (Reverse Polarity)-SMA socket).
3. If an RF coaxial cable is used between the module RF output and the enclosure, a unique (non-standard) antenna connector must be used in the enclosure wall to interface with the antenna.
4. Contact the antenna vendor for detailed antenna speci ications to review the suitability to the end product operating environment and to identify alternatives.
5. All the dipole antennas in the list above have similar cutoff frequencies.

## 4.6. PIC32WM-BZ6 Module Reflow Profile Information

The PIC32WM-BZ6 Module was assembled using the IPC/JEDEC J-STD-020 standard lead-free reflow profile. The PIC32WM-BZ6 Module can be soldered to the host board using standard leaded or lead-free solder reflow profiles. To avoid damaging the module, adhere to the following recommendations:

- For solder reflow recommendations, refer to the *AN233 Solder Reflow Recommendation Application Note* (DS00233).
- Do not exceed a peak temperature (TP) of 250°C.
- For specific reflow profile recommendations from the vendor, refer to the *Solder Paste Data Sheet*.
- Use no-clean flux solder paste.
- Do not wash as moisture can be trapped under the shield.
- Use only one flow. If the PCB requires multiple flows, apply the module on the final flow.

### 4.6.1. Cleaning

The exposed GND pad helps to self-align the module, avoiding pad misalignment. The recommendation is to use the no clean solder pastes. Ensure full drying of no-clean paste fluxes as a result of the reflow process. As per the recommendation by the solder paste vendor, this requires longer reflow profiles and/or peak temperatures toward the high end of the process window. The uncured flux residues can lead to corrosion and/or shorting in accelerated testing and possibly the field.


© 2025 Microchip Technology Inc. and its subsidiaries

## 4.7. PIC32WM-BZ6 Module Assembly Considerations

The PIC32WM-BZ6 module is assembled with an EMI shield to ensure compliance with EMI emission and immunity rules. The EMI shield is made of a tin-plated steel (SPTE) and is not hermetically sealed. Use the solutions such as IPA and similar solvents to clean this module. The user must never use cleaning solutions containing acid on the module.

### 4.7.1. Conformal Coating

The modules are not intended for use with a conformal coating, and the customer assumes all risks (such as the module reliability, performance degradation and so on) if a conformal coating is applied to the modules.


© 2025 Microchip Technology Inc. and its subsidiaries

# 5. Pinout and Signal Descriptions List

This following table provides detailed signal names classified by the peripherals along with the device pinout for the PIC32CX-BZ6 and the PIC32WM-BZ6 Module.

**Table 5-1.** Pinout and Signal Descriptions List

| SoC (PIC32CX2501BZ62132) | Module (PIC32WM-BZ6204) | Pin Name |
|---|---|---|
| A73[3] | A7, A10, A13, A24, A33, A42, A43, A49, B26 | GND |
| A1 | A34, A35, A36, A37, A38, A39, A40, A41, B30, B31, B32, B33 | No connection |
| B1 | — | No connection |
| A2 | — | No connection |
| B2 | A16 | RPA0/RTC_IN3/QSPI_DATA0/IOCA0/RPA0 |
| A3 | — | pmu_vddio/vpmu_vddc (PMU Core circuit power; 1.9-3.6V)[8] |
| B3 | — | No connection |
| A4 | B14 | RPA1/RTC_IN2/QSPI_SCK/IOCA1/RPA1 |
| A5 | A17 | RPA2/RTC_IN1/QSPI_DATA3/IOCA2/RPA2 |
| B4 | A12 | RPA5/SERCOM0_PAD0/IOCA5/RPA5 |
| A6 | B13 | RPD0/IOCD0/RPD0 |
| B5 | A15 | RPD1/IOCD1/RPD1 |
| A7 | — | FAVdd/VDDIO[8,9] |
| B6 | — | No connection |
| A8 | B10 | RPA6/SERCOM0_PAD1/IOCA6/RPA6 |
| B7 | — | No connection |
| A9 | A19 | RPA7/IOCA7/RPA7/VBUSON/SERCOM1_PAD0/TRCLK |
| B8 | — | No connection |
| A10 | B17 | RPA8/IOCA8/RPA8/CAN0_TX/SERCOM1_PAD1 |
| B9 | — | No connection |
| A11 | A20 | RPA9/IOCA9/RPA9/CAN0_RX/RTC_IN0_ALT/SERCOM1_PAD2 |
| B10 | — | No connection |
| A12 | A25, B22 | VDDIO[8,9] |
| B11 | — | No connection |
| A13 | B19 | RPA10/IOCA10/RPA10/RTC_OUT_ALT/SERCOM1_PAD3 |
| B12 | B15 | RPB12/IOCB12/RPB12/QSPI_DATA1 |
| A14 | A27 | RPE2/IOCE2/RPE2/GMAC_GCRS_DV |
| A15 | A26 | RPC9/IOCC9/RPC9/GMAC_GTXEN |
| B13 | B23 | RPC8/IOCC8/RPC8/GMAC_GRXER |
| A16 | A23 | RPC10/IOCC10/RPC10 |
| B14 | A22 | RPE3/IOCE3/RPE3 |
| A17 | — | No connection |
| B15 | — | No connection |
| A18 | — | No connection |
| A19 | — | No connection |
| B16 | — | No connection |
| A20 | — | No connection |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 5-1.** Pinout and Signal Descriptions List (continued)

| SoC (PIC32CX2501BZ62132) | Module (PIC32WM-BZ6204) | Pin Name |
|---|---|---|
| B17 | B28 | RPB9/IOCB9/RPB9/INT0/SERCOM0_PAD2/CM4_SWDIO |
| A21 | B16 | RB13/IOCB13/RPB13/RTC_EVENT/QSPI_CS |
| B18 | A21 | RPA4/USBID/RTC_OUT/SERCOM0_PAD3/IOCA4/RPA4 |
| A22 | A32 | RPB8/IOCB8/RPB8/CM4_SWCLK |
| B19 | B25 | RPA14/IOCA14/RPA14/GMAC_GRX0/I2C_SERCOM_PAD1/ANN0 |
| A23 | A29 | RPA13/GMAC_GRX1/I2C_SERCOM_PAD0/IOCA13/RPA13 |
| A24 | — | VDDIO(8,9) |
| B20 | — | VDDIO(8,9) |
| A25 | A30 | RPC1/IOCC1/RPC1/GMAC_GREFCLKOUT/SERCOM2_PAD3 |
| B21 | A31 | RPC0/IOCC0/RPC0/GMAC_GTX0/SERCOM2_PAD2 |
| A26 | B27 | RPE1/IOCE1/RPE1/GMAC_GTX1/SERCOM2_PAD1 |
| B22 | B20 | RPE0/IOCE0/RPE0/SERCOM2_PAD0 |
| A27 | A18 | RPB11/IOCB11/RPB11/ANA0/QSPI_DATA2 |
| B23 | — | No connection |
| A28 | B21 | RPB10/IOCB10/RPB10/USBOEN/AN0 |
| B24 | — | No connection |
| A29 | — | No connection |
| B25 | — | No connection |
| A30 | — | No connection |
| B26 | — | No connection |
| A31 | — | CLDO_IN (Secondary digital power supply; 1.2V ± 5%)<br>Connect this pin to CLDO_OUT pin with 1 µF capacitor to ground |
| B27 | — | No connection |
| A32 | — | xo_n |
| B28 | — | No connection |
| A33 | — | xo_p |
| B29 | — | No connection |
| A34 | — | VDD_RF (RF block power supply)<br>Connect this pin to RFLDO_OUT pin with 1 µF to ground |
| B30 | — | No connection |
| A35 | — | No connection |
| A36 | — | No connection |
| A37 | — | No connection |
| A38 | — | No connection |
| B31 | — | No connection |
| A39 | — | CLDO_OUT (Core LDO power supply; 1.2V ± 5%)<br>Connect this pin to a 1 µF decoupling capacitor |
| B32 | — | No connection |
| A40 | — | RFLDO_OUT (RF LDO power supply; 1.2V)<br>Connect this pin to VDD_RF pin with a 1 µF decoupling capacitor |
| B33 | — | No connection |
| A41 | — | CLDO_BUCK (RF/DIG CLDO supply);<br>Filtered version of PMU output; connect 1 µF decoupling capacitor |
| B34 | — | No connection |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 5-1.** Pinout and Signal Descriptions List (continued)

| SoC (PIC32CX2501BZ62132) | Module (PIC32WM-BZ6204) | Pin Name |
|---|---|---|
| A42 | — | BUCK_PA (RF PA supply);<br>Filtered version of PMU output; connect a 1 µF decoupling capacitor |
| B35 | — | No connection |
| A43 | — | RF_IO |
| B36 | — | No connection |
| A44 | — | No connection |
| B37 | A45 | RPD2/IOCD2/RPD2/CVDT0/CVDRP16/CVD16/AN16 |
| A45 | A44 | RPE6/IOCE6/RPE6/CVDT14/CVDRP14/CVD14/AN14 |
| B38 | B18 | RPE5/IOCE5/RPE5/ANB0 |
| A46 | A14 | RPE4/IOCE4/RPE4 |
| B39 | B6 | RPC11/IOCC11/RPC11 |
| A47 | — | VDDIO(8,9) |
| B40 | B4 | RPB6/IOCB6/RPB6/CVDT2/CVDR2/CVD2/AN2(5) |
| A48 | B12 | RPA3/RTC_IN0/ANN1/DACOUT/IOCA3/RPA3 |
| B41 | B2 | RPB2/IOCB2/RPB2/AC_AIN0/CVDT6/CVDRP6/CVD6/AN6 |
| A49 | B3 | RPB4/IOCB4/RPB4/CVDT8/CVDRP8/CVD8/AN8(5) |
| B42 | A48 | RPD3/IOCD3/RPD3/CVDT9/CVDRP9/CVD9/AN9 |
| A50 | A46 | RPD5/IOCD5/RPD5/CVDT11/CVDRP11/CVD11/AN11 |
| B43 | A3 | RPB3/IOCB3/RPB3/AC_AIN1/CVDT7/CVDR7/CVD7/AN7 |
| A51 | A47 | VBUS |
| B44 | — | No connection |
| A52 | A1 | AVdd |
| B45 | B29 | RPA15/IOCA15/RPA15 |
| A53 | — | No connection |
| A54 | — | No connection |
| A55 | — | No connection |
| A56 | — | No connection |
| B46 | A5 | RPB7/IOCB7/RPB7/LVDIN/CVDT3/CVDRP3/CVD3/AN3/CM4_SWO |
| A57 | A11 | RPD4/IOCD4/RPD4/CVDT10/CVDRP10/CVD10/AN10 |
| B47 | — | No connection |
| A58 | — | VDDIO/VUSB3V3(8,9) |
| B48 | B8 | RPB15/IOCB15/RPB15/CVDT16/CVDRP17/CVD17/AN17 |
| A59 | A9 | RPB14/IOCB14/RPB14/CVDT17/CVDRP18/CVD18/AN18 |
| B49 | B1 | RPB0/IOCB0/RPB0/AC_AIN2/CVDT4/CVDRP4/CVD4/AN4 |
| B50 | — | No connection |
| A60 | B5 | D+ |
| B51 | A2 | RPB1/IOCB1/RPB1/AC_AIN3/CVDT5/CVDRP5/CVD5/AN5 |
| A61 | A6 | D- |
| B52 | — | No connection |
| A62 | — | PLL_VDD<br>Connect to SOC_LDO_VDD with 100 nF to ground |
| A63 | — | SOC_LDO_VDD |
| B53 | — | No connection |
| A64 | — | SOC_LDO_VDD |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 5-1.** Pinout and Signal Descriptions List (continued)

| SoC (PIC32CX2501BZ62132) | Module (PIC32WM-BZ6204) | Pin Name |
|---|---|---|
| B54 | — | No connection |
| A65 | A28 | RPD6/IOCD6/RPD6/GMAC_GMDIO/CVDT12/CVDRP12/CVD12/AN12 |
| B55 | B24 | RPD7/IOCD7/RPD7/GMAC_GMDC/CVDT13/CVDRP13/CVD13/AN13 |
| A66 | B9 | RPC7/IOCC7/RPC7/CVDT15/CVDRP15/CVD15/AN15 |
| B56 | A4 | RPB5/IOCB5/RPB5/CVDT1/CVDRP1/CVD1/AN1(5) |
| A67 | B11 | MCLR |
| B57 | B7 | SOSCI/RPA11/RPA11(4) |
| A68 | A8 | SOSCO/RPA12/RPA12(4) |
| A69 | — | PMU_MLDO_OUT |
| B58 | — | No connection |
| A70 | — | vpmu_vddp(8) |
| B59 | — | No connection |
| A71 | — | No connection |
| B60 | — | PMU_BK_LX<br>Connect 4.7 µH shielded inductor with 10 µF and 100 nF decoupling capacitor |
| A72 | — | No connection |

**Notes:**

1. The remappable peripherals can use all GPIOs (RPAn, RPBn, RPCn, RPDn, and RPEn) via PPS.
2. All GPIOs (RPAn, RPBn, RPCn, RPDn, and RPEn) can be used for I/O Change Notification (IOCAn, IOCBn, IOCCn, IOCDn, and IOCEn).
3. Connect the metal paddle at the bottom of the device to the system ground.
4. If not using SOSC, use this pin as an input-only pin.
5. JTAG is the default functionality on these pins. It is recommended to write '0' to the CFGCON0.JTAGEN register during application initialization to use these pins for regular GPIO functionality.
6. A pull-up resistor (10K) on the SWCLK pin is critical for reliable operation.
7. For more details on power supply pin connections and the filtering components, refer to the Design Package available on the product page.
8. Provide filtered version of main supply input.
9. Connect a 100 nF decoupling capacitor to this pin.
10. These I/O pins are 5.5V tolerant: NMCLR, PA0, PA1, PA2, PA4, PA5, PA6, PA7, PA8, PA10, PA13, PA15, PB3, PB12, PB13, PC0, PC1, PC8, PC9, PC10, PC11, PD0, PD1, PE0, PE1, PE2, PE3 and PE4.

    All other I/O pins are 3.3V tolerant.


© 2025 Microchip Technology Inc. and its subsidiaries

# 6. I/O Ports and Peripheral Pin Select (PPS)

## 6.1. Overview

General purpose I/O (GPIO) pins allow the PIC32CX-BZ6 devices to monitor and control other devices. To add flexibility and functionality, some pins are multiplexed with alternate function(s). These functions depend on which peripheral features are on the device. In general, when a peripheral is functioning, do not use that pin as a general purpose I/O pin. For more details on pin multiplexing, see *GPIO Pins/PPS Functions* from Related Links. There are default priorities for each GPIO pin as well. For details on priorities, see *Function Priority for Device Pins* from Related Links.
**Note:** The fuse values stored in Boot Flash memory (NVR) can be used to alter the power-on default function for certain GPIO pin. See *System Configuration and Register Locking (CFG)* from Related Links.

**Related Links**
GPIO Pins/PPS Functions
Function Priority for Device Pins
System Configuration and Register Locking (CFG)

## 6.2. Features

The following are some of the key features of the I/O ports:

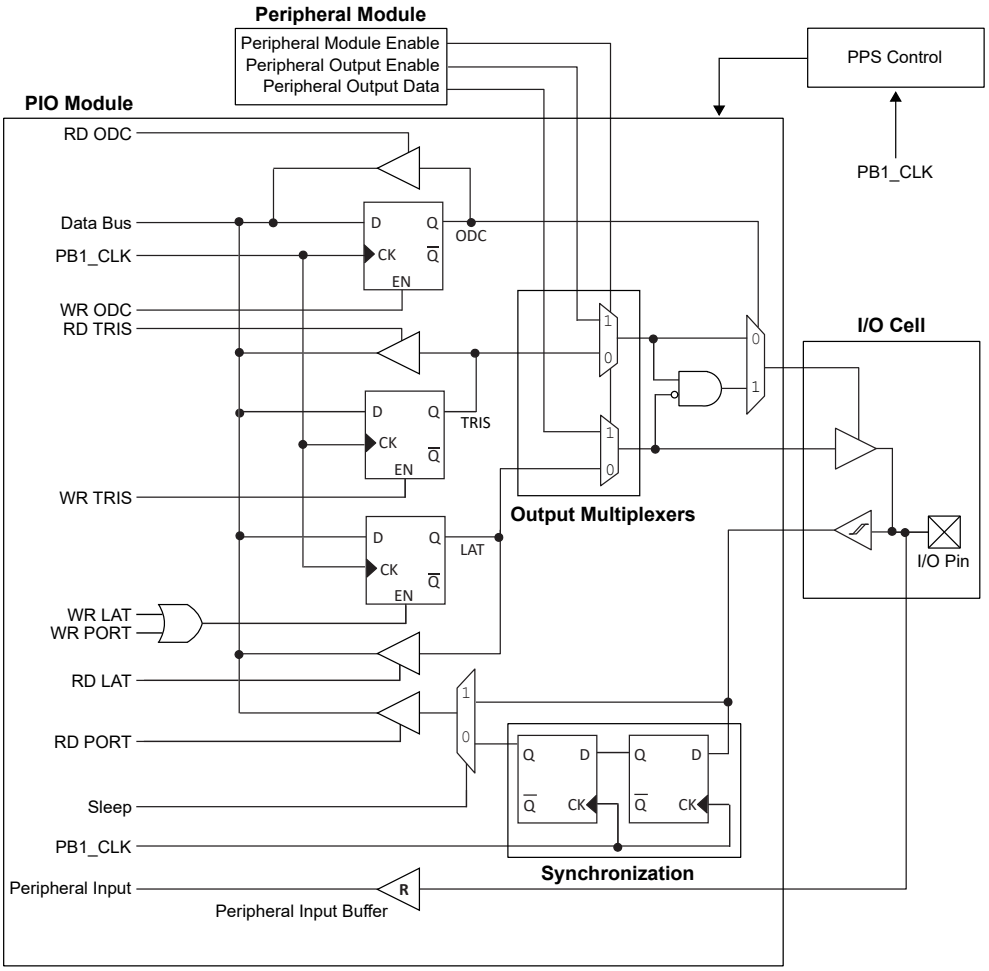
- Individual Output Pin Open-Drain Enable/Disable
- Individual Input Pin Weak Pull-Up and Pull-Down
- Monitoring Selective Inputs and Generate Interrupt After Detecting Change in Pin State
- Operation During Sleep and Idle Modes
- Fast Bit Manipulation Using CLR, SET and INV Registers
- Slew Rate Control (Not Available on All Devices)
- Flexibility for Peripheral Pin Remapping Through PPS

## 6.3. Block Diagram

The following figure illustrates a block diagram of a typical multiplexed I/O port.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 6-1.** Typical Multiplexed Port Structure Block Diagram




## 6.4. Parallel I/O (PIO) Ports

Each I/O port has 14 registers directly associated with the operation of the port. Each I/O port pin has a corresponding bit in these registers. The data direction register (TRISx) determines whether the pin is an input or an output. If the data direction bit is a '1', then the pin is an input. All port pins are defined as inputs after a Reset. Throughout this section, the letter 'x', denotes any or all port module instances. For example, TRISx represents TRISA or TRISB. Any bit and its associated data and control registers that are not valid for a particular device are disabled and read as zeros.

### 6.4.1. I/O Ports Configuration

#### 6.4.1.1. Configuring Port Functions (PORTx)

The PORTx registers allow I/O pins to be accessed:

- A write to a PORTx register writes to the corresponding LATx register (PORTx data latch). Those I/O port pin(s) configured as outputs are updated.
- A write to a PORTx register is effectively the same as a write to a LATx register.
- A read from a PORTx register reads the synchronized signal applied to the port I/O pins.

#### 6.4.1.2. Configuring Latch Functions (LATx)

The LATx registers (PORTx data latch) hold data written to port I/O pins:


© 2025 Microchip Technology Inc. and its subsidiaries

- A write to a LATx register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated.
- A read from a LATx register reads the data held in the PORTx data latch, not from the port I/O pins.

#### 6.4.1.3. Open-Drain Configuration (ODCx)

The user can individually configure each I/O pin for either normal digital output or open-drain output. This is controlled by the open-drain control register, ODCx, associated with each I/O pin. If the ODCx bit for an I/O pin is a '1', the pin acts as an open-drain output. If the ODCx bit for an I/O pin is a '0', configure the pin for a normal digital output (the ODCx bit is valid only for output pins). After a Reset, the status of all the bits of the ODCx register is set to '0'.

The maximum open-drain voltage allowed is the same as the maximum $V_{IH}$ specification. The ODCx register setting functions in all of the I/O modes, allowing the output to behave as an open-drain even if a peripheral is controlling the pin. Although, the user can achieve the same result by manipulating the corresponding LATx and TRISx bits, this procedure does not allow the peripheral to operate in the Open-drain mode (except for the default operation of the $I^2C$ pins). $I^2C$ pins are already open-drain pins; therefore, the ODCx settings do not influence the $I^2C$ pins.

#### 6.4.1.4. Configuring Analog and Digital Port Pins (ANSELx)

The ANSELx register controls the operation of the analog port pins. The port pins that are to function as analog inputs must have their corresponding ANSEL and TRISx bits set. To use port pins for I/O functionality with digital modules, such as Timers, SERCOMs and so on, the corresponding ANSELx bit must be cleared. The ANSELx register has a default value of 0xFFFF; therefore, all pins that share analog functions are, by default, analog and not digital.

If the TRISx bit is cleared (output) while the ANSELx bit is set, the digital output level ($V_{OH}$ or $V_{OL}$) is converted by an analog peripheral, such as the ADC module or the comparator module. When the PORTx register is read, all pins configured as analog input channels are read as cleared (a low-level). Pins configured as digital inputs do not convert an analog input. Analog levels on any pin defined as a digital input (including the ANx pins) can cause the input buffer to consume current that exceeds the device specifications.

#### 6.4.1.5. Configuring Tri-State Functions (TRISx)

The TRISx registers configure the data direction flow through port I/O pins. The TRISx register bits determine whether a PORTx I/O pin is an input or an output:

- If a data direction bit is '1', the corresponding I/O port pin is an input.
- If a data direction bit is '0', the corresponding I/O port pin is an output.
- A read from a TRISx register reads the last value written to that register.
- All I/O port pins are defined as inputs after a Power-on Reset (POR).

#### 6.4.1.6. I/O Port Write/Read Timing

One instruction cycle is required between a port direction change or port write operation and a read operation of the same port. Typically this instruction is an NOP.

#### 6.4.1.7. Input Change Notification (CN)

The Input Change Notification (CN) function of the I/O ports allows PIC32CX-BZ6 devices to generate interrupt requests to the processor in response to a change-of-state on selected input pins. This feature can detect input change of states even in the Sleep mode, when the clocks are disabled.

The following control registers are associated with the CN functionality of each I/O port:

- Change Notice Pull-up Enable (CNPUEx)
- Change Notice Pull-down Enable (CNPDx)
- Change Notice Control (CNCONx)


© 2025 Microchip Technology Inc. and its subsidiaries

- Change Notice Enable (CNENx/CNNEx)
- Change Notice Status (CNSTATx/CNFx) or the positive edge control

Each I/O pin also has a weak pull-up and a weak pull-down connected to it. The pull-ups act as a current source or sink source connected to the pin and eliminate the need for external resistors when push button or keypad devices are connected. The pull-ups and pull-downs are enabled separately using the CNPUEx and the CNPDx registers, which contain the control bits for each of the pins. Setting any of the control bits enables the weak pull-ups and/or pull-downs for the corresponding pins.

**Note:** Pull-ups and pull-downs on change notification pins must always be disabled when the port pin is configured as a digital output

The CNCONx registers provide change notice control.

The CNENx/CNNEx registers contain the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. CNENx enables a mismatch CN interrupt condition when EDGEDETECT is not set. When EDGEDETECT is set, CNNEx controls the negative edge while CNENx controls the positive. On devices that do not have EDGEDETECT, this CN logic acts as if EDGEDETECT is not set.

The CNSTATx/CNFx registers indicate whether a change occurred on the corresponding pin since the last read of the PORTx bit. The CNFx registers indicate the type of change that occurred.

#### 6.4.1.7.1. CN Configuration and Operation

The CN pins are configured as follows:

1. Disable the CPU interrupts.
2. Set the desired CN I/O pin as an input by setting the corresponding TRISx register bits = `1`.
3. Enable the CN Module by setting the ON bit (CNCONx[15]) = `1`.
4. Enable the individual CN input pins, and optional pull-ups or pull-downs.
5. Read the corresponding PORTx registers to clear the CN interrupt.
6. Configure the CN Interrupt Priority bits, NVIC.IP register.
7. Clear the CN Interrupt Flag bit, by setting the corresponding CLRPEND bit in the NVIC.IPCR register.
8. Configure the CN pin interrupt for a specific edge detect using the EDGEDETECT bit in the CNCONx register, and set up edge control using the CNENx/CNNEx bits.
9. Enable the CN Interrupt Enable bit by setting the corresponding enable bit in the NVIC.ISER register.
10. Enable CPU interrupts.

The CNSTATx/CNFx registers indicate whether a change occurred on the corresponding pin since the last read of the PORTx bit. The CNFx registers indicate the type. When a CN interrupt occurs in the Mismatch mode, the user must read the PORTx register associated with the CN pins. This will clear the mismatch condition and set up the CN logic to detect the next pin change. The current PORTx value can be compared to the PORTx read value obtained at the last CN interrupt or during initialization and used to determine which pin changed. The CN pins have a minimum input pulse-width specification. See *Electrical Characteristics* from Related Links.

**Related Links**

Electrical Characteristics

### 6.4.1.8. Slew Rate Control

Some I/O pins can be configured for various types of slew rate control on their associated port. This is controlled by the slew rate control bits in the SRCON1x and SRCON0x registers that are associated


© 2025 Microchip Technology Inc. and its subsidiaries

with each I/O port. The slew rate control is configured using the corresponding bit in each register, as shown in the following table.

As an example, writing 0x0001, 0x0000 to SRCON1A and SRCON0A, respectively, can enable slew rate control on the RA0 pin and sets the slew rate to the slow edge rate.

**Table 6-1.** Slew Rate Control Bit Settings[1]

| SRCON1x | SRCON0x | Description |
|---|---|---|
| 1 | 1 | Slew rate control is enabled and is set to the slowest edge rate |
| 1 | 0 | Slew rate control is enabled and is set to the slow edge rate |
| 0 | 1 | Slew rate control is enabled and is set to the medium edge rate |
| 0 | 0 | Slew rate control is disabled and is set to the fastest edge rate |

**Note:**

1. By default, all the port pins are set to the fastest edge rate.

#### 6.4.1.9. CLR, SET and INV Registers

Every I/O module register has corresponding SET, CLR and INV registers, which provide atomic bit manipulations. As the name of the registers imply, a value written to a SET, CLR or INV register effectively performs the implied operation, but only on the corresponding base register and only bits specified as '1' are modified. Bits specified as '0' are not modified. For example,

- Writing `0x0001` to the TRISASET register sets only bit 0 in the base register TRISA
- Writing `0x0020` to the PORTBCLR register clears only bit 5 in the base register PORTB
- Writing `0x9000` to the LATAINV register inverts only bits 15 and 12 in the base register LATA

Reading the SET, CLR and INV registers returns an undefined value. To see the influences of a write operation to a SET, CLR or INV register, the base register must be read instead.

A typical method to toggle an I/O pin requires a read-modify-write operation performed on a PORTx register in the software. For example, a read from a PORTx register, mask and modify the desired output bit or bits, and write the resulting value back to the PORTx register. This method is vulnerable to a read-modify-write issue where the port value may change after it is read and before the modified data can be written back, thus, changing the previous state. This method also requires more instructions.

A more efficient and atomic method uses the PORTxINV register. A write to the PORTxINV register effectively performs a read-modify-write operation on the target base register, equivalent to the software operation described previously; however, it is done in the hardware. To toggle an I/O pin using this method, a '1' is written to the corresponding bit in the PORTxINV register. This operation reads the PORTx register, inverts only those bits specified as '1', and writes the resulting value to the LATx register, thus, toggling the corresponding I/O pins all in a single atomic instruction cycle. PORTAINV = `0x0001`.

## 6.5. Peripheral Pin Select (PPS)

A major challenge in general purpose devices is providing the largest possible set of peripheral features while minimizing the conflict of features on I/O pins. The challenge is even greater on low pin-count devices. In an application where more than one peripheral needs to be assigned to a single pin, inconvenient workarounds in application code or a complete redesign may be the only option.

The PPS configuration provides an alternative to these choices by enabling peripheral set selection and their placement on a wide range of I/O pins. By increasing the pinout options available on a particular device, the users can better modify the device to their entire application, rather than trimming the application to fit the device.


© 2025 Microchip Technology Inc. and its subsidiaries

This feature operates over a fixed subset of digital I/O pins. The users may independently map the input and/or output of most digital peripherals to these I/O pins. The PPS configuration is performed in the software and generally does not require the device to be reprogrammed. The hardware safeguards that prevent accidental or spurious changes to the peripheral mapping are included once the PPS configuration is established.

### 6.5.1. Remappable Pin Groupings

The remappable pins, as well as the available input and output functions, are divided into six groups. The remappable pins of Group k can be assigned pin functions only from Group k, k = 1, 2, 3, 4, 5, 6. The pins used by each peripheral are spread across all five groups when possible to maximize flexibility.

### 6.5.2. Available Pins

The number of available pins is dependent on the particular device and its pin count. Pins that support the PPS feature include the designation “RPn” in their full pin designation, where:

- RP – Designates a remappable peripheral
- n – Remappable port number

### 6.5.3. Available Peripherals

The peripherals managed by the PPS are all digital-only peripherals. These include general serial communications (SERCOM), general purpose timer clock inputs, timer-related peripherals (input capture and output compare), interrupt-on-change inputs and reference clocks (input and output).

In comparison, some digital-only peripheral modules are never included in the PPS feature. This is because the peripheral’s function requires special I/O circuitry on a specific port and cannot be easily connected to multiple pins. These modules include $I^2C$ among others. A similar requirement excludes all modules with analog inputs, such as the ADC.

A key difference between remappable and non-remappable peripherals is that remappable peripherals are not associated with a default I/O pin. The peripheral must always be assigned to a specific I/O pin before it can be used. In contrast, non-remappable peripherals are always available on a default pin, assuming that the peripheral is active and not conflicting with another peripheral.

When a remappable peripheral is active on a given I/O pin, it takes priority over all other digital I/O. Priority is given regardless of the type of peripheral that is mapped. Remappable peripherals never take priority over any analog functions associated with the pin.

### 6.5.4. RP Register Protection

The <INPUT>R and RPxxR registers are implemented with two levels of protection:

- I/O Lock Feature – All PPS registers may only be written while CFGCON0.IOLOCK = 0; once the IOLOCK is set, the registers cannot be written.
- IOLOCK Protection – The state of the IOLOCK bit can only be changed once it is unlocked using the CFGCON0.CFGLOCK[1:0] register.

These features prevent the RP registers from being inadvertently written during normal operation because changing the pinout functionality may have detrimental system-level outcome.

### 6.5.5. Controlling PPS

The PPS features are controlled through two sets of SFRs: one to map peripheral inputs and another to map outputs. They are separately controlled; therefore, a particular peripheral’s input and output (if the peripheral has both) can be placed on any selectable function pin without constraint.

The association of a peripheral to a peripheral-selectable pin is handled in two different ways, depending on whether an input or output is mapped.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.5.5.1. Remappable Inputs

#### 6.5.5.1.1. Enabling Remappable Peripheral Inputs

With PPS, each remappable input pin function (EXTINT0, SERCOM0_PAD3 and so on) is assigned to be driven from a specific device pin by programming the corresponding <INPUT>R[4:0] register (meaning, EXTINT0R[4:0], SCOM0P3R[4:0] and so on) with a value defined in the *Input Pin Selection Group 1*, *Input Pin Selection Group 2*, *Input Pin Selection Group 3*, *Input Pin Selection Group 4*, *Input Pin Selection Group 5*, *Input Pin Selection Group 6* tables and *[pin name]R* register. See these tables in the *Remappable Input Example* and *[pin name]R* from Related Links.

Assigning a remappable input pin function does not automatically enable the digital input buffer on the pin. The buffer must be enabled for each remap pin (RPx) by disabling all higher priority pin functions on that pin. In general, all functions other than GPIO are considered higher priority than remap pins. For the list and priority of pin functions on each pin, see *Function Priority for Device Pins* from Related Links.

When transitioning the pin usage from one peripheral to another, it is mandatory to make sure the pins are not configured for any other peripheral. Therefore, to avoid glitching outputs, the user is responsible for turning off the appropriate peripherals before remapping the pin functions associated with that peripheral. On Reset, all inputs are mapped to a default value and all outputs are disabled; therefore, the mapping may be safely performed after any device Reset. After Reset, all inputs are mapped to a default value and all outputs are disabled, the user can safely perform the mapping after any device Reset.

**Related Links**

Remappable Input Example

[pin name]R

Peripheral Pin Select Input Register

Function Priority for Device Pins

#### 6.5.5.1.2. Remappable Input Priority

The user can select only one single pin for any of the remappable peripheral inputs. There is no need for priority encoding for RP inputs.

**Note:** A remappable input function does not have any control over the output of the RPx pin.

In this way, it is possible to drive a remappable output function on an RPx pin and a completely different remappable input function on the same pin. This can be useful, for instance, in driving an EVSYS output back into a Timer clock or gating input by assigning both functions to the same remap pin.

**Note:** To allow flexibility on a different pin variant, repeat the same input functions in multiple groups. Therefore, to differentiate between them, the 'Off' code is provided in the *Input Pin Selection Group* tables. See these tables in the *Remappable Input Example* from Related Links.

For proper operation, the software must ensure that the unused group register offset of a (repeated) input function is programmed to `5'h0`. Failure to do so leads to unknown behavior.

**Related Links**

Remappable Input Example

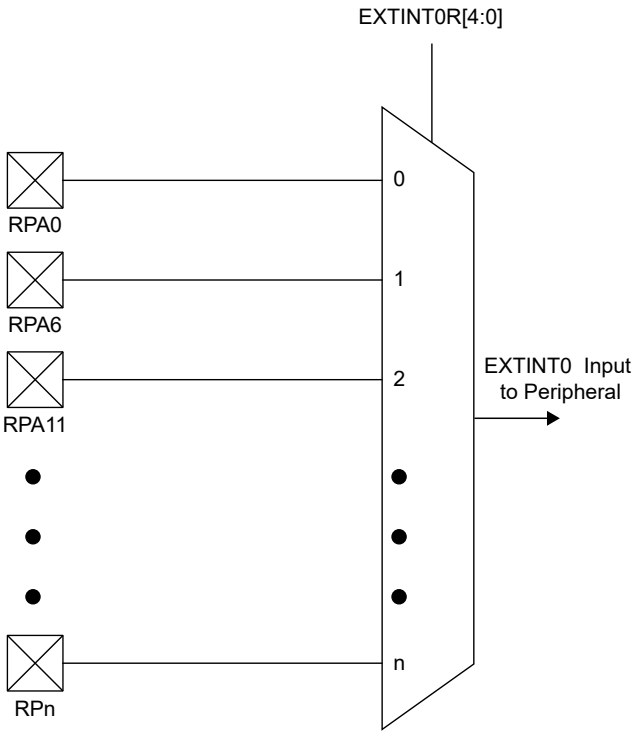
#### 6.5.5.1.3. Remappable Input Example

The following figure illustrates the remappable pin selection for the EXTINT0 input. In order to remap the EXTINT0 input to a particular pin, program the EXTINT0R remap register. EXTINT0 is in Group 1; therefore, it can be mapped to any pin that is in Group 1 (RPA0, RPA6, RPA11, RPB0, RPB6, RPB11 and so on).


© 2025 Microchip Technology Inc. and its subsidiaries

To map EXTINT0 to RPB0, program the value 4 (5'b00100) into the EXINTR0R SFR register. See the *Input Pin Selection Group 1* table in the *Input Mapping in PIC32CX-BZ6 Family of Devices* from Related Links.

**Figure 6-2.** EXTINT0 Remappable Pin Selection




**Note:** For input only, PPS functionality does not have priority over TRISx settings. Therefore, when configuring the RPn pin for input, the user must configure the corresponding bit in the TRISx register for input (set to '1').

**Related Links**

Input Mapping in PIC32CX-BZ6 Family of Devices

### 6.5.5.1.4. Input Mapping in PIC32CX-BZ6 Family of Devices

The following tables provide input mapping in PIC32CX-BZ6 family of devices


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-2.** Input Pin Selection Group 1

| Peripheral Pin (pin name) | [*pin name*]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| EXTINT0 | EXTINT0R | EXTINT0R[4:0] | 0 0000 = Off |
| SERCOM0/PAD3 | SCOM0P3R | SCOM0P3R[4:0] | 0 0001 = RPA0 |
| SERCOM1/PAD2 | SCOM1P2R | SCOM1P2R[4:0] | 0 0010 = RPA6 |
| SERCOM2/PAD3 | SCOM2P3R | SCOM2P3R[4:0] | 0 0011 = RPA11 |
| SERCOM3/PAD2 | SCOM3P2R | SCOM3P2R[4:0] | 0 0100 = RPB0 |
| SERCOM4/PAD3 | SCOM4P3R | SCOM4P3R[4:0] | 0 0101 = RPB6 |
| SERCOM5/PAD2 | SCOM5P2R | SCOM5P2R[4:0] | 0 0110 = RPB11 |
| QSPI_DATA1 | QSPI_DATA1R | QSPI_DATA1R[4:0] | 0 0111 = RPC0 |
| CCL/IN0 | CCLIN0R | CCLIN0R[4:0] | 0 1000 = RPD6 |
| CCL/IN3 | CCLIN3R | CCLIN3R[4:0] | 0 1001 = RPE0 |
| TC0/WO0 | TC0WO0R | TC0WO0R[4:0] | 0 1010 = RPE6 |
| TC1/WO0 | TC1WO0R | TC1WO0R[4:0] | 0 1011 = RPB1 |
| TC2/WO0 | TC2WO0R | TC2WO0R[4:0] | 0 1100 = RPC9 |
| TC3/WO0 | TC3WO0R | TC3WO0R[4:0] | 0 1101 = RPD3 |
| TC4/WO0 | TC4WO0R | TC4WO0R[4:0] | 0 1110 = RPE4 |
| TC5/WO0 | TC5WO0R | TC5WO0R[4:0] | 0 1111 = RPA3 |
| TC6/WO0 | TC6WO0R | TC6WO0R[4:0] | 1 0000 = RPA7 |
| TC7/WO0 | TC7WO0R | TC7WO0R[4:0] | 1 0001 = RPB4 |
| TC8/WO0 | TC8WO0R | TC8WO0R[4:0] | 1 0010 = RPB8 |
| QEI/HOME0 | QEIHOME0R | QEIHOME0R[4:0] | 1 0011 = RPB12 |
| AC/CMPTEN | ACCMPTENR | ACCMPTENR[4:0] | 1 0100 = RPA2 |
| | | | 1 0101 = RPA10 |
| | | | 1 0110 = RPA14 |
| | | | 1 0111 = RPB3 |
| | | | 1 1000 = RPB7 |
| | | | 1 1001 = RPA9 |
| | | | 1 1010 – 1 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-3.** Input Pin Selection Group 2

| Peripheral Pin (pin name) | [*pin name*]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| EXTINT1 | EXTINT1R | EXTINT1R[4:0] | 0 0000 = Off |
| SERCOM0/PAD0 | SCOM0P0R | SCOM0P0R[4:0] | 0 0001 = RPA1 |
| SERCOM1/PAD3 | SCOM1P3R | SCOM1P3R[4:0] | 0 0010 = RPA7 |
| SERCOM2/PAD0 | SCOM2P0R | SCOM2P0R[4:0] | 0 0011 = RPA12 |
| SERCOM3/PAD3 | SCOM3P3R | SCOM3P3R[4:0] | 0 0100 = RPB1 |
| SERCOM4/PAD0 | SCOM4P0R | SCOM4P0R[4:0] | 0 0101 = RPB7 |
| SERCOM5/PAD3 | SCOM5P3R | SCOM5P3R[4:0] | 0 0110 = RPC1 |
| QSPI_DATA2 | QSPI_DATA2R | QSPI_DATA2R[4:0] | 0 0111 = RPC7 |
| CCL/IN1 | CCLIN1R | CCLIN1R[4:0] | 0 1000 = RPD7 |
| CCL/IN4 | CCLIN4R | CCLIN4R[4:0] | 0 1001 = RPE1 |
| TC0/WO0 | TC0WO0R | TC0WO0R[4:0] | 0 1010 = RPA3 |
| TC1/WO1 | TC1WO1R | TC1WO1R[4:0] | 0 1011 = RPB2 |
| TC2/WO1 | TC2WO1R | TC2WO1R[4:0] | 0 1100 = RPC0 |
| TC3/WO1 | TC3WO1R | TC3WO1R[4:0] | 0 1101 = RPD4 |
| TC4/WO1 | TC4WO1R | TC4WO1R[4:0] | 0 1110 = RPE6 |
| TC5/WO1 | TC5WO1R | TC5WO1R[4:0] | 0 1111 = RPA4 |
| TC6/WO1 | TC6WO1R | TC6WO1R[4:0] | 1 0000 = RPA8 |
| TC7/WO1 | TC7WO1R | TC7WO1R[4:0] | 1 0001 = RPB12 |
| TC8/WO1 | TC8WO1R | TC8WO1R[4:0] | 1 0010 = RPB5 |
| TC9/WO0 | TC9WO0R | TC9WO0R[4:0] | 1 0011 = RPB9 |
| QEI/HOME0 | QEIHOME0R | QEIHOME0R[4:0] | 1 0100 = RPB13 |
| QEI/INDX0 | QEIINDX0R | QEIINDX0R[4:0] | 1 0101 = RPB0 |
| | | | 1 0110 = RPB4 |
| | | | 1 0111 = RPB8 |
| | | | 1 1000 = RPA0 |
| | | | 1 1001 = RPA11 |
| | | | 1 1010 – 1 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-4.** Input Pin Selection Group 3

| Peripheral Pin (pin name) | [*pin name*]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| EXTINT2 | EXTINT2R | EXTINT2R[4:0] | 0 0000 = Off |
| SERCOM1/PAD0 | SCOM1P0R | SCOM1P0R[4:0] | 0 0001 = RPA2 |
| SERCOM3/PAD0 | SCOM3P0R | SCOM3P0R[4:0] | 0 0010 = RPA8 |
| SERCOM5/PAD0 | SCOM5P0R | SCOM5P0R[4:0] | 0 0011 = RPA13 |
| QSPI_DATA3 | QSPI_DATA3R | QSPI_DATA3R[4:0] | 0 0100 = RPB2 |
| CCL/IN2 | CCLIN2R | CCLIN2R[4:0] | 0 0101 = RPB8 |
| CCL/IN5 | CCLIN5R | CCLIN5R[4:0] | 0 0110 = RPB13 |
| TC0/WO1 | TC0WO1R | TC0WO1R[4:0] | 0 0111 = RPC8 |
| TC2/WO0 | TC2WO0R | TC2WO0R[4:0] | 0 1000 = RPD2 |
| TC3/WO0 | TC3WO0R | TC3WO0R[4:0] | 0 1001 = RPE2 |
| TC4/WO0 | TC4WO0R | TC4WO0R[4:0] | 0 1010 = RPA4 |
| TC5/WO0 | TC5WO0R | TC5WO0R[4:0] | 0 1011 = RPB3 |
| TC6/WO0 | TC6WO0R | TC6WO0R[4:0] | 0 1100 = RPC1 |
| TC7/WO0 | TC7WO0R | TC7WO0R[4:0] | 0 1101 = RPD6 |
| TC8/WO0 | TC8WO0R | TC8WO0R[4:0] | 0 1110 = RPE0 |
| TC9/WO1 | TC9WO1R | TC9WO1R[4:0] | 0 1111 = RPA5 |
| QEI/INDX0 | QEIINDX0R | QEIINDX0R[4:0] | 1 0000 = RPA9 |
| QEI/QEA0 | QEIQEA0R | QEIQEA0R[4:0] | 1 0001 = RPB6 |
| CAN1/RX | CAN1RXR | CAN1RXR[4:0] | 1 0010 = RPB10 |
| | | | 1 0011 = RPA0 |
| | | | 1 0100 = RPB1 |
| | | | 1 0101 = RPB5 |
| | | | 1 0110 = RPB9 |
| | | | 1 0111 = RPA1 |
| | | | 1 1000 = RPA12 |
| | | | 1 1001 = Off |
| | | | 1 1010 – 1 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-5.** Input Pin Selection Group 4

| Peripheral Pin (pin name) | [*pin name*]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| EXTINT3 | EXTINT3R | EXTINT3R[4:0] | 0 0000 = Off |
| NMI | NMIR | NMIR[4:0] | 0 0001 = RPA3 |
| SERCOM0/PAD2 | SCOM0P2R | SCOM0P2R[4:0] | 0 0010 = RPA9 |
| SERCOM2/PAD2 | SCOM2P2R | SCOM2P2R[4:0] | 0 0011 = RPA14 |
| SERCOM4/PAD2 | SCOM4P2R | SCOM4P2R[4:0] | 0 0100 = RPB3 |
| QSPI_DATA0 | QSPI_DATA0R | QSPI_DATA0R[4:0] | 0 0101 = RPC9 |
| TC0/WO1 | TC0WO1R | TC0WO1R[4:0] | 0 0110 = RPD3 |
| TC2/WO1 | TC2WO1R | TC2WO1R[4:0] | 0 0111 = RPE3 |
| TC3/WO1 | TC3WO1R | TC3WO1R[4:0] | 0 1000 = RPA0 |
| TC4/WO1 | TC4WO1R | TC4WO1R[4:0] | 0 1001 = RPB4 |
| TC5/WO1 | TC5WO1R | TC5WO1R[4:0] | 0 1010 = RPC7 |
| TC6/WO1 | TC6WO1R | TC6WO1R[4:0] | 0 1011 = RPD7 |
| TC7/WO1 | TC7WO1R | TC7WO1R[4:0] | 0 1100 = RPE1 |
| TC9/WO0 | TC9WO0R | TC9WO0R[4:0] | 0 1101 = RPA6 |
| QEI/QEA0 | QEIQEA0R | QEIQEA0R[4:0] | 0 1110 = RPA10 |
| QEI/QEB0 | QEIQEB0R | QEIQEB0R[4:0] | 0 1111 = RPB7 |
| COEX/WLAN_ACTIVE | COEXWLANACTR | COEXWLANACTR[4:0] | 1 0000 = RPB11 |
| | | | 1 0001 = RPA1 |
| | | | 1 0010 = RPA5 |
| | | | 1 0011 = RPA13 |
| | | | 1 0100 = RPB2 |
| | | | 1 0101 = RPB6 |
| | | | 1 0110 = RPB10 |
| | | | 1 0111 = RPA8 |
| | | | 1 1000 = RPA2 |
| | | | 1 1001 = Off |
| | | | 1 1010 – 1 1111 = off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-6.** Input Pin Selection Group 5

| Peripheral Pin (pin name) | [*pin name*]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| TC8/WO1 | TC8WO1R | TC8WO1R[4:0] | 0 0000 = Off |
| TC9/WO1 | TC9WO1R | TC9WO1R[4:0] | 0 0001 = RPA4 |
| QEI/QEB0 | QEIQEB0R | QEIQEB0R[4:0] | 0 0010 = RPA10 |
| | | | 0 0011 = RPB4 |
| | | | 0 0100 = RPB10 |
| | | | 0 0101 = RPD4 |
| | | | 0 0110 = RPE4 |
| | | | 0 0111 = RPA1 |
| | | | 0 1000 = RPB0 |
| | | | 0 1001 = RPC8 |
| | | | 0 1010 = RPD2 |
| | | | 0 1011 = RPE2 |
| | | | 0 1100 = RPC10 |
| | | | 0 1101 = RPC11 |
| | | | 0 1110 = RPA15 |
| | | | 0 1111 = RPB14 |
| | | | 1 0000 = RPB15 |
| | | | 1 0001 = RPD0 |
| | | | 1 0010 = RPD1 |
| | | | 1 0011 = Off |
| | | | 1 0100 = Off |
| | | | 1 0101 = Off |
| | | | 1 0110 = Off |
| | | | 1 0111 = Off |
| | | | 1 1000 = Off |
| | | | 1 1001 = Off |
| | | | 1 1010 – 1 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-7.** Input Pin Selection Group 6

| Peripheral Pin (pin name) | [pin name]R SFR | [*pin name*]R bits | [*pin name*]R Value to RPn Pin Selection |
|---|---|---|---|
| SERCOM0/PAD1 | SCOM0P1R | SCOM0P1R[4:0] | 0 0000 = Off |
| SERCOM1/PAD1 | SCOM1P1R | SCOM1P1R[4:0] | 0 0001 = RPA5 |
| SERCOM2/PAD1 | SCOM2P1R | SCOM2P1R[4:0] | 0 0010 = RPB5 |
| SERCOM3/PAD1 | SCOM3P1R | SCOM3P1R[4:0] | 0 0011 = RPB9 |
| SERCOM4/PAD1 | SCOM4P1R | SCOM4P1R[4:0] | 0 0100 = RPD5 |
| SERCOM5/PAD1 | SCOM5P1R | SCOM5P1R[4:0] | 0 0101 = RPE5 |
| REFI | REFIR | REFIR[4:0] | 0 0110 = RPA3 |
| | | | 0 0111 = RPA6 |
| | | | 0 1000 = RPA7 |
| | | | 0 1001 = RPB6 |
| | | | 0 1010 = RPB8 |
| | | | 0 1011 = Off |
| | | | 0 1100 = Off |
| | | | 0 1101 = Off |
| | | | 0 1110 = Off |
| | | | 0 1111 = Off |
| | | | 1 0000 = Off |
| | | | 1 0001 = Off |
| | | | 1 0010 = Off |
| | | | 1 0011 = Off |
| | | | 1 0100 = Off |
| | | | 1 0101 = Off |
| | | | 1 0110 = Off |
| | | | 1 0111 = Off |
| | | | 1 1000 = Off |
| | | | 1 1001 = Off |
| | | | 1 1010 – 1 1111 = Off |

### 6.5.5.2. Remappable Output

The remappable pin output assigns a peripheral output function to an output pin. When the group for the output pin is identified, see the following table, which shows peripheral output functions and their group.

Each remappable output can be programmed to an output function that is from its same output group number. For example, if RPA0 is part of Group 2, then the user can program it to have any Group 2 output function on its pin. Therefore, for a given output peripheral signal, the user must first choose which remappable pin to use, choose a group number for that pin, then program the control registers for that pin. For example, RPA<0-10, 13, 14> G<1, 2, 3, 4> R or RPB<0-13> G<1, 2, 3, 4>R. See *Remappable Output Pin Configuration – Group 1*, *Remappable Output Pin Configuration – Group 2*, *Remappable Output Pin Configuration – Group 3*, *Remappable Output Pin Configuration – Group 4*, *Remappable Output Pin Configuration – Group 5* and *Remappable Output Pin Configuration – Group 6* tables in the *Output Mapping in PIC32CX-BZ6 Family of Devices* from Related Links.

The user must follow the rules for which group belong to which pin such that multiple peripherals are not driving the same pin from different groups. For instance, pin RPA0 (PA0) as an output belongs to Group 2 and Group 3. If the peripheral driving the signal to RPA0 is coming from Group 2, the software must ensure that all Group 3 signals for RPA0 are disabled with an OFF value in the corresponding RPA0G3R control register.


© 2025 Microchip Technology Inc. and its subsidiaries

A null output is associated with the output register Reset value of '0'. By default, do this to ensure that remappable outputs remain disconnected from all output pins.

**Related Links**

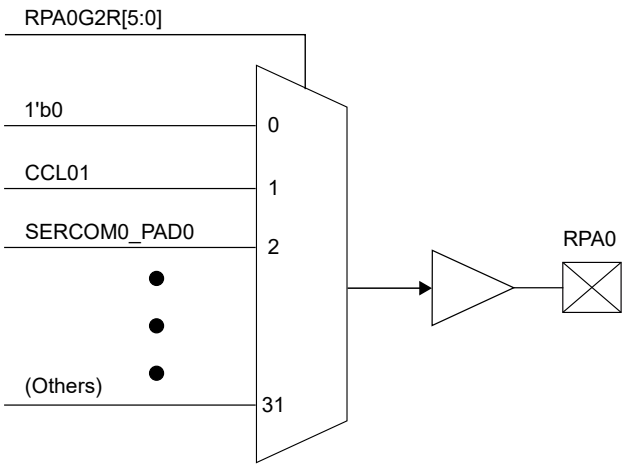
Output Mapping in PIC32CX-BZ6 Family of Devices

#### 6.5.5.2.1. Pin Output RP Registers

Register *RPnR* shows the RP Remap Register format for output functions. See *RPnR* from Related Links. Each RP pin has a 6-bit field that can be assigned to the desired output function. For a complete list of output function values and their associated register names, see *Remappable Output Pin Configuration – Group 1*, *Remappable Output Pin Configuration – Group 2*, *Remappable Output Pin Configuration – Group 3*, *Remappable Output Pin Configuration – Group 4*, *Remappable Output Pin Configuration – Group 5* and *Remappable Output Pin Configuration – Group 6* tables in the *Output Mapping in PIC32CX-BZ6 Family of Devices* from Related Links.

When transitioning the pin usage from one peripheral to another, it is mandatory to make sure the pins are not configured for any other peripheral. Therefore, to avoid glitching outputs, the user is responsible for turning off the appropriate peripherals before remapping the pin functions associated with that peripheral. After Reset, map all inputs to a default value and disable all outputs. The user must perform the mapping after any device Reset.

**Figure 6-3.** Example Multiplexing of Remappable Output Signal for RPA0 (Map Output Function to Pin)




**Related Links**

RPnR

Peripheral Pin Select Output Register

Output Mapping in PIC32CX-BZ6 Family of Devices

#### 6.5.5.2.2. Output Mapping in PIC32CX-BZ6 Family of Devices

The following tables provide output mapping in PIC32CX-BZ6 family of devices.

**Table 6-8.** PPS Output Groups

| Group 1 | Group 2 | Group 3 | Group 4 | Group 5 | Group 6 |
|---|---|---|---|---|---|
| Off | Off | Off | Off | Off | Off |
| SERCOM0_PAD3 | CCLO1 | CCLO0 | CCLO1 | CCLO0 | SERCOM0_PAD1 |
| SERCOM0_PAD2 | SERCOM0_PAD0 | SERCOM0_PAD0 | SERCOM0_PAD2 | SERCOM1_PAD3 | SERCOM1_PAD1 |
| SERCOM1_PAD0 | SERCOM0_PAD3 | SERCOM0_PAD3 | SERCOM0_PAD0 | SERCOM2_PAD0 | SERCOM2_PAD1 |
| SERCOM1_PAD2 | SERCOM0_PAD2 | SERCOM1_PAD2 | SERCOM1_PAD3 | TCC0_WO4 | SERCOM3_PAD1 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-8.** PPS Output Groups (continued)

| Group 1 | Group 2 | Group 3 | Group 4 | Group 5 | Group 6 |
|---|---|---|---|---|---|
| SERCOM2_PAD3 | SERCOM1_PAD3 | SERCOM1_PAD0 | SERCOM1_PAD0 | TCC0_WO2 | SERCOM4_PAD1 |
| SERCOM2_PAD2 | SERCOM1_PAD2 | SERCOM1_PAD3 | SERCOM2_PAD2 | TCC0_WO1 | SERCOM5_PAD1 |
| SERCOM3_PAD0 | SERCOM2_PAD0 | SERCOM2_PAD0 | SERCOM2_PAD0 | TCC1_WO4 | QSPI_SCK |
| SERCOM3_PAD2 | SERCOM2_PAD3 | SERCOM2_PAD3 | SERCOM3_PAD3 | TCC1_WO2 | REFO1 |
| SERCOM4_PAD3 | SERCOM2_PAD2 | SERCOM3_PAD2 | SERCOM3_PAD0 | TCC1_WO1 | REFO2 |
| SERCOM4_PAD2 | SERCOM3_PAD3 | SERCOM3_PAD0 | SERCOM4_PAD2 | TC8_WO1 | REFO3 |
| SERCOM5_PAD0 | SERCOM3_PAD2 | SERCOM3_PAD3 | SERCOM4_PAD0 | TC9_WO1 | REFO4 |
| SERCOM5_PAD2 | SERCOM4_PAD0 | SERCOM4_PAD0 | SERCOM5_PAD3 | QSPI_CS | Off |
| TCC0_WO0 | SERCOM4_PAD3 | SERCOM4_PAD3 | SERCOM5_PAD0 | QEICCMP0 | Off |
| TCC0_WO4 | SERCOM4_PAD2 | SERCOM5_PAD2 | TCC0_WO3 | GMAC_TSUCOMP | Off |
| TCC1_WO0 | SERCOM5_PAD3 | SERCOM5_PAD0 | TCC0_WO1 | TSTBUS3 | Off |
| TCC1_WO4 | SERCOM5_PAD2 | SERCOM5_PAD3 | TCC0_WO5 | TSTBUS7 | Off |
| TCC2_WO0 | TCC0_WO1 | TCC0_WO2 | TCC1_WO3 | TSTBUS11 | Off |
| TC0_WO1 | TCC0_WO5 | TCC0_WO0 | TCC1_WO1 | Off | Off |
| TC1_WO0 | TCC0_WO3 | TCC0_WO4 | TCC1_WO5 | Off | Off |
| TC2_WO0 | TCC1_WO1 | TCC1_WO2 | TCC2_WO1 | Off | Off |
| TC3_WO0 | TCC1_WO5 | TCC1_WO0 | TC0_WO0 | Off | Off |
| TC4_WO0 | TCC1_WO3 | TCC1_WO4 | TC1_WO1 | Off | Off |
| TC5_WO0 | TCC2_WO1 | TCC2_WO0 | TC2_WO1 | Off | Off |
| TC6_WO0 | TC0_WO1 | TC0_WO0 | TC3_WO1 | Off | Off |
| TC7_WO0 | TC1_WO1 | TC1_WO0 | TC4_WO1 | Off | Off |
| TC8_WO0 | TC2_WO1 | TC2_WO0 | TC5_WO1 | Off | Off |
| QSPI_CS | TC3_WO1 | TC3_WO0 | TC6_WO1 | Off | Off |
| QSPI_DATA3 | TC4_WO1 | TC4_WO0 | TC7_WO1 | Off | Off |
| QSPI_DATA2 | TC5_WO1 | TC5_WO0 | TC9_WO0 | Off | Off |
| QSPI_DATA1 | TC6_WO1 | TC6_WO0 | QSPI_CS | Off | Off |
| QEICCMP0 | TC7_WO1 | TC7_WO0 | QSPI_DATA2 | Off | Off |
| TSTBUS0 | TC8_WO1 | TC8_WO0 | QSPI_DATA1 | Off | Off |
| TSTBUS4 | TC9_WO0 | TC9_WO1 | QSPI_DATA0 | Off | Off |
| TSTBUS8 | QSPI_CS | QSPI_CS | TSTBUS2 | Off | Off |
| FECTRL1 | QSPI_DATA0 | QSPI_DATA1 | TSTBUS6 | Off | Off |
| FECTRL2 | QSPI_DATA3 | QSPI_DATA0 | AC_CMP1 | Off | Off |
| COEX_BT_STATE | QSPI_DATA2 | QSPI_DATA3 | AC_CMPTOUT | Off | Off |
| Off | TSTBUS5 | TSTBUS1 | CAN1_TX | Off | Off |
| Off | TSTBUS9 | TSTBUS10 | FECTRL3 | Off | Off |
| Off | FECTRL0 | AC_CMP0 | FECTRL5 | Off | Off |
| Off | COEX_RF_ACT | AC_CMPTRDY | Off | Off | Off |
| Off | Off | FECTRL4 | Off | Off | Off |
| Off | Off | Off | Off | Off | Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-9.** Remappable Output Pin Configuration – Group 1

| RPn Port Pin | RPnG1R SFR | RPnG1R Bits | RPnG1R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA0 | RPA0G1R | RPA0G1R[5:0] | 00 0000 = Off |
| RPA2 | RPA2G1R | RPA2G1R[5:0] | 00 0001 = SERCOM0_PAD3 |
| RPA3 | RPA3G1R | RPA3G1R[5:0] | 00 0010 = SERCOM0_PAD2 |
| RPA6 | RPA6G1R | RPA6G1R[5:0] | 00 0011 = SERCOM1_PAD0 |
| RPA7 | RPA7G1R | RPA7G1R[5:0] | 00 0100 = SERCOM1_PAD2 |
| RPA9 | RPA9G1R | RPA9G1R[5:0] | 00 0101 = SERCOM2_PAD3 |
| RPA10 | RPA10G1R | RPA10G1R[5:0] | 00 0110 = SERCOM2_PAD2 |
| RPA14 | RPA14G1R | RPA14G1R[5:0] | 00 0111 = SERCOM3_PAD0 |
| RPB0 | RPB0G1R | RPB0G1R[5:0] | 00 1000 = SERCOM3_PAD2 |
| RPB1 | RPB1G1R | RPB1G1R[5:0] | 00 1001 = SERCOM4_PAD3 |
| RPB3 | RPB3G1R | RPB3G1R[5:0] | 00 1010 = SERCOM4_PAD2 |
| RPB4 | RPB4G1R | RPB4G1R[5:0] | 00 1011 = SERCOM5_PAD0 |
| RPB6 | RPB6G1R | RPB6G1R[5:0] | 00 1100 = SERCOM5_PAD2 |
| RPB7 | RPB7G1R | RPB7G1R[5:0] | 00 1101 = TCC0_WO0 |
| RPB8 | RPB8G1R | RPB8G1R[5:0] | 00 1110 = TCC0_WO4 |
| RPB11 | RPB11G1R | RPB11G1R[5:0] | 00 1111 = TCC1_WO0 |
| RPB12 | RPB12G1R | RPB12G1R[5:0] | 01 0000 = TCC1_WO4 |
| RPC0 | RPC0G1R | RPC0G1R[5:0] | 01 0001 = TCC2_WO0 |
| RPC9 | RPC9G1R | RPC9G1R[5:0] | 01 0010 = TC0_WO1 |
| RPD3 | RPD3G1R | RPD3G1R[5:0] | 01 0011 = TC1_WO0 |
| RPD6 | RPD6G1R | RPD6G1R[5:0] | 01 0100 = TC2_WO0 |
| RPE0 | RPE0G1R | RPE0G1R[5:0] | 01 0101 = TC3_WO0 |
| RPE4 | RPE4G1R | RPE4G1R[5:0] | 01 0110 = TC4_WO0 |
| RPE6 | RPE6G1R | RPE6G1R[5:0] | 01 0111 = TC5_WO0 |
| | | | 01 1000 = TC6_WO0 |
| | | | 01 1001 = TC7_WO0 |
| | | | 01 1010 = TC8_WO0 |
| | | | 01 1011 = QSPI_CS |
| | | | 01 1100 = QSPI_DATA3 |
| | | | 01 1101 = QSPI_DATA2 |
| | | | 01 1110 = QSPI_DATA1 |
| | | | 01 1111 = QEICCMP0 |
| | | | 10 0000 = TSTBUS0 |
| | | | 10 0001 = TSTBUS4 |
| | | | 10 0010 = TSTBUS8 |
| | | | 10 0011 = FECTRL1 |
| | | | 10 0100 = FECTRL2 |
| | | | 10 0101 = COEX_BT_STATE |
| | | | 10 0110 = Off |
| | | | 10 0111 = Off |
| | | | 10 1000 = Off |
| | | | 10 1001 = Off |
| | | | 10 1010 = Off |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-10.** Remappable Output Pin Configuration – Group 2

| RPn Port Pin | RPnG2R SFR | RPnG2R Bits | RPnG2R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA0 | RPA0G2R | RPA0G2R[5:0] | 00 0000 = Off |
| RPA1 | RPA1G2R | RPA1G2R[5:0] | 00 0001 = CCLO1 |
| RPA3 | RPA3G2R | RPA3G2R[5:0] | 00 0010 = SERCOM0_PAD0 |
| RPA4 | RPA4G2R | RPA4G2R[5:0] | 00 0011 = SERCOM0_PAD3 |
| RPA7 | RPA7G2R | RPA7G2R[5:0] | 00 0100 = SERCOM0_PAD2 |
| RPA8 | RPA8G2R | RPA8G2R[5:0] | 00 0101 = SERCOM1_PAD3 |
| RPB0 | RPB0G2R | RPB0G2R[5:0] | 00 0110 = SERCOM1_PAD2 |
| RPB1 | RPB1G2R | RPB1G2R[5:0] | 00 0111 = SERCOM2_PAD0 |
| RPB2 | RPB2G2R | RPB2G2R[5:0] | 00 1000 = SERCOM2_PAD3 |
| RPB4 | RPB4G2R | RPB4G2R[5:0] | 00 1001 = SERCOM2_PAD2 |
| RPB5 | RPB5G2R | RPB5G2R[5:0] | 00 1010 = SERCOM3_PAD3 |
| RPB7 | RPB7G2R | RPB7G2R[5:0] | 00 1011 = SERCOM3_PAD2 |
| RPB8 | RPB8G2R | RPB8G2R[5:0] | 00 1100 = SERCOM4_PAD0 |
| RPB9 | RPB9G2R | RPB9G2R[5:0] | 00 1101 = SERCOM4_PAD3 |
| RPB12 | RPB12G2R | RPB12G2R[5:0] | 00 1110 = SERCOM4_PAD2 |
| RPB13 | RPB13G2R | RPB13G2R[5:0] | 00 1111 = SERCOM5_PAD3 |
| RPC0 | RPC0G2R | RPC0G2R[5:0] | 01 0000 = SERCOM5_PAD2 |
| RPC1 | RPC1G2R | RPC1G2R[5:0] | 01 0001 = TCC0_WO1 |
| RPC7 | RPC7G2R | RPC7G2R[5:0] | 01 0010 = TCC0_WO5 |
| RPD4 | RPD4G2R | RPD4G2R[5:0] | 01 0011 = TCC0_WO3 |
| RPD7 | RPD7G2R | RPD7G2R[5:0] | 01 0100 = TCC1_WO1 |
| RPE1 | RPE1G2R | RPE1G2R[5:0] | 01 0101 = TCC1_WO5 |
| RPE6 | RPE6R | RPE6R[5:0] | 01 0110 = TCC1_WO3 |
| | | | 01 0111 = TCC2_WO1 |
| | | | 01 1000 = TC0_WO1 |
| | | | 01 1001 = TC1_WO1 |
| | | | 01 1010 = TC2_WO1 |
| | | | 01 1011 = TC3_WO1 |
| | | | 01 1100 = TC4_WO1 |
| | | | 01 1101 = TC5_WO1 |
| | | | 01 1110 = TC6_WO1 |
| | | | 01 1111 = TC7_WO1 |
| | | | 10 0000 = TC8_WO1 |
| | | | 10 0001 = TC9_WO0 |
| | | | 10 0010 = QSPI_CS |
| | | | 10 0011 = QSPI_DATA0 |
| | | | 10 0100 = QSPI_DATA3 |
| | | | 10 0101 = QSPI_DATA2 |
| | | | 10 0110 = TSTBUS5 |
| | | | 10 0111 = TSTBUS9 |
| | | | 10 1000 = FECTRL0 |
| | | | 10 1001 = COEX_RF_ACT |
| | | | 10 1010 = Off |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-11.** Remappable Output Pin Configuration – Group 3

| RPn Port Pin | RPnG3R SFR | RPnG3R Bits | RPnG3R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA0 | RPA0G3R | RPA0G3R[5:0] | 00 0000 = Off |
| RPA1 | RPA1G3R | RPA1G3R[5:0] | 00 0001 = CCLO0 |
| RPA2 | RPA2G3R | RPA2G3R[5:0] | 00 0010 = SERCOM0_PAD0 |
| RPA4 | RPA4G3R | RPA4G3R[5:0] | 00 0011 = SERCOM0_PAD3 |
| RPA5 | RPA5G3R | RPA5G3R[5:0] | 00 0100 = SERCOM1_PAD2 |
| RPA8 | RPA8G3R | RPA8G3R[5:0] | 00 0101 = SERCOM1_PAD0 |
| RPA9 | RPA9G3R | RPA9G3R[5:0] | 00 0110 = SERCOM1_PAD3 |
| RPA13 | RPA13G3R | RPA13G3R[5:0] | 00 0111 = SERCOM2_PAD0 |
| RPB1 | RPB1G3R | RPB1G3R[5:0] | 00 1000 = SERCOM2_PAD3 |
| RPB2 | RPB2G3R | RPB2G3R[5:0] | 00 1001 = SERCOM3_PAD2 |
| RPB3 | RPB3G3R | RPB3G3R[5:0] | 00 1010 = SERCOM3_PAD0 |
| RPB5 | RPB5G3R | RPB5G3R[5:0] | 00 1011 = SERCOM3_PAD3 |
| RPB6 | RPB6G3R | RPB6G3R[5:0] | 00 1100 = SERCOM4_PAD0 |
| RPB8 | RPB8G3R | RPB8G3R[5:0] | 00 1101 = SERCOM4_PAD3 |
| RPB9 | RPB9G3R | RPB9G3R[5:0] | 00 1110 = SERCOM5_PAD2 |
| RPB10 | RPB10G3R | RPB10G3R[5:0] | 00 1111 = SERCOM5_PAD0 |
| RPB13 | RPB13G3R | RPB13G3R[5:0] | 01 0000 = SERCOM5_PAD3 |
| RPC1 | RPC1G3R | RPC1G3R[5:0] | 01 0001 = TCC0_WO2 |
| RPC8 | RPC8G3R | RPC8G3R[5:0] | 01 0010 = TCC0_WO0 |
| RPD2 | RPD2G3R | RPD2G3R[5:0] | 01 0011 = TCC0_WO4 |
| RPD6 | RPD6G3R | RPD6G3R[5:0] | 01 0100 = TCC1_WO2 |
| RPE0 | RPE0G3R | RPE0G3R[5:0] | 01 0101 = TCC1_WO0 |
| RPE2 | RPE2G3R | RPE2G3R[5:0] | 01 0110 = TCC1_WO4 |
| | | | 01 0111 = TCC2_WO0 |
| | | | 01 1000 = TC0_WO0 |
| | | | 01 1001 = TC1_WO0 |
| | | | 01 1010 = TC2_WO0 |
| | | | 01 1011 = TC3_WO0 |
| | | | 01 1100 = TC4_WO0 |
| | | | 01 1101 = TC5_WO0 |
| | | | 01 1110 = TC6_WO0 |
| | | | 01 1111 = TC7_WO0 |
| | | | 10 0000 = TC8_WO0 |
| | | | 10 0001 = TC9_WO1 |
| | | | 10 0010 = QSPI_CS |
| | | | 10 0011 = QSPI_DATA1 |
| | | | 10 0100 = QSPI_DATA0 |
| | | | 10 0101 = QSPI_DATA3 |
| | | | 10 0110 = TSTBUS1 |
| | | | 10 0111 = TSTBUS10 |
| | | | 10 1000 = AC_CMP0 |
| | | | 10 1001 = AC_CMPTRDY |
| | | | 10 1010 = FECTRL4 |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-12.** Remappable Output Pin Configuration – Group 4

| RPn Port Pin | RPnG4R SFR | RPnG4R Bits | RPnG4R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA0 | RPA0G4R | RPA0G4R[5:0] | 00 0000 = Off |
| RPA1 | RPA1G4R | RPA1G4R[5:0] | 00 0001 = CCLO1 |
| RPA2 | RPA2G4R | RPA2G4R[5:0] | 00 0010 = SERCOM0_PAD2 |
| RPA3 | RPA3G4R | RPA3G4R[5:0] | 00 0011 = SERCOM0_PAD0 |
| RPA5 | RPA5G4R | RPA5G4R[5:0] | 00 0100 = SERCOM1_PAD3 |
| RPA6 | RPA6G4R | RPA6G4R[5:0] | 00 0101 = SERCOM1_PAD0 |
| RPA8 | RPA8G4R | RPA8G4R[5:0] | 00 0110 = SERCOM2_PAD2 |
| RPA9 | RPA9G4R | RPA9G4R[5:0] | 00 0111 = SERCOM2_PAD0 |
| RPA10 | RPA10G4R | RPA10G4R[5:0] | 00 1000 = SERCOM3_PAD3 |
| RPA13 | RPA13G4R | RPA13G4R[5:0] | 00 1001 = SERCOM3_PAD0 |
| RPA14 | RPA14G4R | RPA14G4R[5:0] | 00 1010 =SERCOM4_PAD2 |
| RPB2 | RPB2G4R | RPB2G4R[5:0] | 00 1011 = SERCOM4_PAD0 |
| RPB3 | RPB3G4R | RPB3G4R[5:0] | 00 1100 = SERCOM5_PAD3 |
| RPB4 | RPB4G4R | RPB4G4R[5:0] | 00 1101 = SERCOM5_PAD0 |
| RPB6 | RPB6G4R | RPB6G4R[5:0] | 00 1110 = TCC0_WO3 |
| RPB7 | RPB7G4R | RPB7G4R[5:0] | 00 1111 = TCC0_WO1 |
| RPB10 | RPB10G4R | RPB10G4R[5:0] | 01 0000 = TCC0_WO5 |
| RPB11 | RPB11G4R | RPB11G4R[5:0] | 01 0001 = TCC1_WO3 |
| RPC7 | RPC7G4R | RPC7G4R[5:0] | 01 0010 = TCC1_WO1 |
| RPC9 | RPC9G4R | RPC9G4R[5:0] | 01 0011 = TCC1_WO5 |
| RPD3 | RPD3G4R | RPD3G4R[5:0] | 01 0100 = TCC2_WO1 |
| RPD7 | RPD7G4R | RPD7G4R[5:0] | 01 0101 = TC0_WO0 |
| RPE1 | RPE1G4R | RPE1G4R[5:0] | 01 0110 = TC1_WO1 |
| RPE3 | RPE3G4R | RPE3G4R[5:0] | 01 0111 = TC2_WO1 |
| | | | 01 1000 = TC3_WO1 |
| | | | 01 1001 = TC4_WO1 |
| | | | 01 1010 = TC5_WO1 |
| | | | 01 1011 = TC6_WO1 |
| | | | 01 1100 = TC7_WO1 |
| | | | 01 1101 = TC9_WO0 |
| | | | 01 1110 = QSPI_CS |
| | | | 01 1111 = QSPI_DATA2 |
| | | | 10 0000 = QSPI_DATA1 |
| | | | 10 0001 = QSPI_DATA0 |
| | | | 10 0010 = TSTBUS2 |
| | | | 10 0011 = TSTBUS6 |
| | | | 10 0100 = AC_CMP1 |
| | | | 10 0101 = AC_CMPTOUT |
| | | | 10 0110 = CAN1_TX |
| | | | 10 0111 = FECTRL3 |
| | | | 10 1000 = FECTRL5 |
| | | | 10 1001 = Off |
| | | | 10 1010 = Off |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-13.** Remappable Output Pin Configuration – Group 5

| RPn Port Pin | RPnG5R SFR | RPnG5R Bits | RPnG5R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA1 | RPA1G5R | RPA1G5R[5:0] | 00 0000 = Off |
| RPA4 | RPA4G5R | RPA4G5R[5:0] | 00 0001 = CCLO0 |
| RPA10 | RPA10G5R | RPA10G5R[5:0] | 00 0010 = SERCOM1_PAD3 |
| RPA15 | RPA15G5R | RPA15G5R[5:0] | 00 0011 = SERCOM2_PAD0 |
| RPB0 | RPB0G5R | RPB0G5R[5:0] | 00 0100 = TCC0_WO4 |
| RPB4 | RPB4G5R | RPB4G5R[5:0] | 00 0101 = TCC0_WO2 |
| RPB10 | RPB10G5R | RPB10G5R[5:0] | 00 0110 = TCC0_WO1 |
| RPB14 | RPB14G5R | RPB14G5R[5:0] | 00 0111 = TCC1_WO4 |
| RPB15 | RPB15G5R | RPB15G5R[5:0] | 00 1000 = TCC1_WO2 |
| RPC8 | RPC8G5R | RPC8G5R[5:0] | 00 1001 = TCC1_WO1 |
| RPC10 | RPC10G5R | RPC10G5R[5:0] | 00 1010 = TC8_WO1 |
| RPC11 | RPC11G5R | RPC11G5R[5:0] | 00 1011 = TC9_WO1 |
| RPD0 | RPD0G5R | RPD0G5R[5:0] | 00 1100 = QSPI_CS |
| RPD1 | RPD1G5R | RPD1G5R[5:0] | 00 1101 = QEICCMP0 |
| RPD2 | RPD2G5R | RPD2G5R[5:0] | 00 1110 = GMAC_TSUCOMP |
| RPD4 | RPD4G5R | RPD4G5R[5:0] | 00 1111 = TSTBUS3 |
| RPE2 | RPE2G5R | RPE2G5R[5:0] | 01 0000 = TSTBUS7 |
| RPE4 | RPE4G5R | RPE4G5R[5:0] | 01 0001 = TSTBUS11 |
| | | | 01 0010 = Off |
| | | | 01 0011 = Off |
| | | | 01 0100 = Off |
| | | | 01 0101 = Off |
| | | | 01 0110 = Off |
| | | | 01 0111 = Off |
| | | | 01 1000 = Off |
| | | | 01 1001 = Off |
| | | | 01 1010 = Off |
| | | | 01 1011 = Off |
| | | | 01 1100 = Off |
| | | | 01 1101 = Off |
| | | | 01 1110 = Off |
| | | | 01 1111 = Off |
| | | | 10 0000 = Off |
| | | | 10 0001 = Off |
| | | | 10 0010 = Off |
| | | | 10 0011 = Off |
| | | | 10 0100 = Off |
| | | | 10 0101 = Off |
| | | | 10 0110 = Off |
| | | | 10 0111 = Off |
| | | | 10 1000 = Off |
| | | | 10 1001 = Off |
| | | | 10 1010 = Off |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-14.** Remappable Output Pin Configuration – Group 6

| RPn Port Pin | RPnG6R SFR | RPnG6R Bits | RPnG6R Value to Peripheral Pin Selection |
|---|---|---|---|
| RPA3 | RPA3G6R | RPA3G6R[5:0] | 00 0000 = Off |
| RPA5 | RPA5G6R | RPA5G6R[5:0] | 00 0001 = SERCOM0_PAD1 |
| RPA6 | RPA6G6R | RPA6G6R[5:0] | 00 0010 = SERCOM1_PAD1 |
| RPA7 | RPA7G6R | RPA7G6R[5:0] | 00 0011 = SERCOM2_PAD1 |
| RPB5 | RPB5G6R | RPB5G6R[5:0] | 00 0100 = SERCOM3_PAD1 |
| RPB6 | RPB6G6R | RPB6G6R[5:0] | 00 0101 = SERCOM4_PAD1 |
| RPB8 | RPB8G6R | RPB8G6R[5:0] | 00 0110 = SERCOM5_PAD1 |
| RPB9 | RPB9G6R | RPB9G6R[5:0] | 00 0111 = QSPI_SCK |
| RPD5 | RPD5G6R | RPD5G6R[5:0] | 00 1000 = REFO1 |
| RPE5 | RPE5G6R | RPE5G6R[5:0] | 00 1001 = REFO2 |
| | | | 00 1010 = REFO3 |
| | | | 00 1011 = REFO4 |
| | | | 00 1100 = Off |
| | | | 00 1101 = Off |
| | | | 00 1110 = Off |
| | | | 00 1111 = Off |
| | | | 01 0000 = Off |
| | | | 01 0001 = Off |
| | | | 01 0010 = Off |
| | | | 01 0011 = Off |
| | | | 01 0100 = Off |
| | | | 01 0101 = Off |
| | | | 01 0110 = Off |
| | | | 01 0111 = Off |
| | | | 01 1000 = Off |
| | | | 01 1001 = Off |
| | | | 01 1010 = Off |
| | | | 01 1011 = Off |
| | | | 01 1100 = Off |
| | | | 01 1101 = Off |
| | | | 01 1110 = Off |
| | | | 01 1111 = Off |
| | | | 10 0000 = Off |
| | | | 10 0001 = Off |
| | | | 10 0010 = Off |
| | | | 10 0011 = Off |
| | | | 10 0100 = Off |
| | | | 10 0101 = Off |
| | | | 10 0110 = Off |
| | | | 10 0111 = Off |
| | | | 10 1000 = Off |
| | | | 10 1001 = Off |
| | | | 10 1010 = Off |
| | | | 10 1011 – 11 1111 = Off |


© 2025 Microchip Technology Inc. and its subsidiaries

## 6.6. Peripheral Multiplexing

Many pins also support one or more peripheral modules. When configured to operate with a peripheral, a pin may not be used for general input or output. In many cases, a pin must still be configured for input or output, although some peripherals override the TRISx configuration. The typical Multiplexed Port Structure Block Diagram illustrates how ports are shared with other peripherals and the associated I/O pin to which they are connected. See *Typical Multiplexed Port Structure Block Diagram* in *Block Diagram* from Related Links. Multiple peripheral functions may be multiplexed on each I/O pin. The priority of the peripheral function depends on the order of the pin descriptions. See *Function Priority for Device Pins* from Related Links.

**Note:** The TRISx register bit can control the output of a pin or, in some cases, by the peripheral itself.

**Related Links**

Block Diagram

Function Priority for Device Pins

### 6.6.1. Multiplexed Digital Input Peripheral

The following conditions are characteristics of a multiplexed digital input peripheral:

- Peripheral does not control the TRISx register. Some peripherals require the pin be configured as an input by setting the corresponding TRISx bit = `1`.
- Peripheral input path is independent of I/O input path and uses an input buffer that is dependent on the peripheral.
- PORTx register data input path is not affected and can read the pin value.

### 6.6.2. Multiplexed Digital Output Peripheral

The following conditions are characteristics of a multiplexed digital output peripheral:

- The peripheral controls the output data. Some peripherals require the pin to be configured as an output by setting the corresponding TRISx bit = `0`.
- If a peripheral pin has an automatic tri-state feature, the peripheral can tri-state the pin.
- The peripheral can affect the pin output driver type. For example, drive strength, slew rate and so on.
- PORTx register output data has no effect.

### 6.6.3. Multiplexing Digital Bidirectional Peripheral

The following conditions are characteristics of a multiplexed digital bidirectional peripheral:

- The peripheral automatically configures the pin as an output but not as an input. Some peripherals require the pin to be configured as an input by setting the corresponding TRISx bit = `1`.
- Peripherals control output data.
- The pin output driver type can be affected by the peripheral (for example, drive strength, slew rate and so on).
- The PORTx register data input path is not affected and can read the pin value.
- PORTx register output data has no effect.

### 6.6.4. Multiplexing Analog Input Peripheral

The following condition is characteristic of a multiplexed analog input peripheral:

- All digital port input buffers are disabled.
- PORTx registers read '`0`' to prevent crowbar current.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.6.5. Multiplexing Analog Output Peripheral

The following conditions are characteristics of a multiplexed analog output peripheral:

- All digital port input buffers are disabled.
- PORTx registers read '0' to prevent crowbar current.
- Analog output is driven onto the pin, independent of the associated TRISx setting.

## 6.7. Function Priority for Device Pins

The device pins have an associated priority order in which functionality is exhibited on each pin. This priority order impacts the availability of PPS functionality. For example, if enabling SERCOM0, choose the outputs to be High Speed mode in the DEVCFG1 fuses (bit 11), give priority to pins PB9, PA4, PA5 and PA6 and use them as SERCOM0 pins instead of GPIO/PPS pins. The following table provides details for the priority in which functions are brought out on each device pin. Top entry is higher priority and bottom entry is lower priority for each specific pin.

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6),

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pa0 | QSPI_DATA0 | QSPI |
| | RTC_IN3 | RTCC |
| | RPA0 | PPS |
| | IOCA0 | Change Notification |
| | RA0 | GPIO |
| pa1 | QSPI_SCK | QSPI |
| | RTC_IN2 | RTCC |
| | RPA1 | PPS |
| | IOCA1 | Change Notification |
| | RA1 | GPIO |
| pa2 | QSPI_DATA3 | QSPI |
| | RTC_IN1 | RTCC |
| | RPA2 | PPS |
| | IOCA2 | Change Notification |
| | RA2 | GPIO |
| pa3 | TRD2 | Trace (Debug) |
| | SCLKI | Secondary Oscillator - Digital |
| | DACOUT | DAC |
| | ANN1 | ADC (Differential) |
| | RTC_IN0 | RTCC |
| | RPA3 | PPS |
| | IOCA3 | Change Notification |
| | RA3 | GPIO |
| pa4 | SERCOM0_PAD3 | SERCOM0 |
| | RTC_OUT | RTCC |
| | USBID | USB |
| | RPA4 | PPS |
| | IOCA4 | Change Notification |
| | RA4 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pa5 | SERCOM0_PAD0 | SERCOM0 |
| | RPA5 | PPS |
| | IOCA5 | Change Notification |
| | RA5 | GPIO |
| pa6 | TRD3 | Trace |
| | SERCOM0_PAD1 | SERCOM0 |
| | RPA6 | PPS |
| | IOCA6 | Change Notification |
| | RA6 | GPIO |
| pa7 | TRCLK | Trace(Debug) |
| | SERCOM1_PAD0 | SERCOM1 |
| | VBUSON | USB Charge Pump |
| | RPA7 | PPS |
| | IOCA7 | Change Notification |
| | RA7 | GPIO |
| pa8 | SERCOM1_PAD1 | SERCOM1 |
| | CAN0_TX | Enhanced CAN |
| | RPA8 | PPS |
| | IOCA8 | Change Notification |
| | RA8 | GPIO |
| pa9 | SERCOM1_PAD2 | SERCOM1 |
| | RTC_IN0_ALT | RTCC |
| | CAN0_RX | Enhanced CAN |
| | RPA9 | PPS |
| | IOCA9 | Change Notification |
| | RA9 | GPIO |
| pa10 | SERCOM1_PAD3 | SERCOM1 |
| | RTC_OUT_ALT | RTCC |
| | RPA10 | PPS |
| | IOCA10 | Change Notification |
| | RA10 | GPIO |
| pa11 | SOSCI | Secondary Oscillator - Analog |
| | RPA11 | PPS |
| | RA11 | GPIO |
| pa12 | SOSCO | Secondary Oscillator - Analog |
| | RPA12 | PPS |
| | RA12 | GPIO |
| pa13 | I2C_SERCOM_PAD0 | I2C |
| | GMAC_GRX1 | Ethernet |
| | RPA13 | PPS |
| | IOCA13 | Change Notification |
| | RA13 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pa14 | ANN0 | ADC (Differential) |
| | I2C_SERCOM_PAD1 | I2C |
| | GMAC_GRX0 | Ethernet |
| | RPA14 | PPS |
| | IOCA14 | Change Notification |
| | RA14 | GPIO |
| pa15 | RPA15 | PPS |
| | IOCA15 | Change Notification |
| | RA15 | GPIO |
| pb0 | AN4 | ADC |
| | CVD4 | CVD |
| | CVDR4 | CVD |
| | CVDT4 | CVD |
| | AC_AIN2 | Analog Comparator |
| | RPB0 | PPS |
| | IOCB0 | Change Notification |
| | RB0 | GPIO |
| pb1 | AN5 | ADC |
| | CVD5 | CVD |
| | CVDR5 | CVD |
| | CVDT5 | CVD |
| | AC_AIN3 | Analog Comparator |
| | RPB1 | PPS |
| | IOCB1 | Change Notification |
| | RB1 | GPIO |
| pb2 | AN6 | ADC |
| | CVD6 | CVD |
| | CVDR6 | CVD |
| | CVDT6 | CVD |
| | AC_AIN0 | Analog Comparator |
| | RPB2 | PPS |
| | IOCB2 | Change Notification |
| | RB2 | GPIO |
| pb3 | AN7 | ADC |
| | CVD7 | CVD |
| | CVDR7 | CVD |
| | CVDT7 | CVD |
| | AC_AIN1 | Analog Comparator |
| | RPB3 | PPS |
| | IOCB3 | Change Notification |
| | RB3 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pb4 | AN8 | ADC |
| | CVD8 | CVD |
| | CVDR8 | CVD |
| | CVDT8 | CVD |
| | RPB4 | PPS |
| | IOCB4 | Change Notification |
| | RB4 | GPIO |
| pb5 | TRD0 | Trace (Debug) |
| | AN1 | ADC |
| | CVD1 | CVD |
| | CVDR1 | CVD |
| | CVDT1 | CVD |
| | RPB5 | PPS |
| | IOCB5 | Change Notification |
| | RB5 | GPIO |
| pb6 | TMS | JTAG |
| | TRD1 | Trace |
| | AN2 | ADC |
| | CVD2 | CVD |
| | CVDR2 | CVD |
| | CVDT2 | CVD |
| | RPB6 | PPS |
| | IOCB6 | Change Notification |
| | RB6 | GPIO |
| pb7 | SWO | Debug |
| | AN3 | ADC |
| | CVD3 | CVD |
| | CVDR3 | CVD |
| | CVDT3 | CVD |
| | LVDIN | LVD Voltage Reference |
| | RPB7 | PPS |
| | IOCB7 | Change Notification |
| | RB7 | GPIO |
| pb8 | SWCLK | Debug |
| | RPB8 | PPS |
| | IOCB8 | Change Notification |
| | RB8 | GPIO |
| pb9 | CM4_SWDIO | Debug |
| | SERCOM0_PAD2 | SERCOM0 |
| | INT0 | Wake-up interrupt |
| | RPB9 | PPS |
| | IOCB9 | Change Notification |
| | RB9 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pb10 | AN0 | ADC |
| | USBOEN | USB |
| | RPB10 | PPS |
| | IOCB10 | Change Notification |
| | RB10 | GPIO |
| pb11 | QSPI_DATA2 | QSPI |
| | ANA0 | ADC (Alternate) |
| | RPB11 | PPS |
| | IOCB11 | Change Notification |
| | RB11 | GPIO |
| pb12 | QSPI_DATA1 | QSPI |
| | RPB12 | PPS |
| | IOCB12 | Change Notification |
| | RB12 | GPIO |
| pb13 | QSPI_CS | QSPI |
| | RTC_EVENT | RTCC |
| | RPB13 | PPS |
| | IOCB13 | Change Notification |
| | RB13 | GPIO |
| pb14 | AN18 | ADC |
| | CVD18 | CVD |
| | CVDR18 | CVD |
| | CVDT17 | CVD |
| | RPB14 | PPS |
| | IOCB14 | Change Notification |
| | RB14 | GPIO |
| pb15 | AN17 | ADC |
| | CVD17 | CVD |
| | CVDR17 | CVD |
| | CVDT16 | CVD |
| | RPB15 | PPS |
| | IOCB15 | Change Notification |
| | RB15 | GPIO |
| pc0 | SERCOM2_PAD2 | SERCOM2 |
| | GMAC_GTX0 | Ethernet |
| | RPC0 | PPS |
| | IOCC0 | Change Notification |
| | RC0 | GPIO |
| pc1 | SERCOM2_PAD3 | SERCOM2 |
| | GMAC_GREFCLKOUT | Ethernet |
| | RPC1 | PPS |
| | IOCC1 | Change Notification |
| | RC1 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pc7 | AN15 | ADC |
| | CVD15 | CVD |
| | CVDR15 | CVD |
| | CVDT15 | CVD |
| | RPC7 | PPS |
| | IOCC7 | Change Notification |
| | RC7 | GPIO |
| pc8 | GMAC_GRXER | Ethernet |
| | RPC8 | PPS |
| | IOCC8 | Change Notification |
| | RC8 | GPIO |
| pc9 | GMAC_GTXEN | Ethernet |
| | RPC9 | PPS |
| | IOCC9 | Change Notification |
| | RC9 | GPIO |
| pc10 | RPC10 | PPS |
| | IOCC10 | Change Notification |
| | RC10 | GPIO |
| pc11 | RPC11 | PPS |
| | IOCC11 | Change Notification |
| | RC11 | GPIO |
| pd0 | RPD0 | PPS |
| | IOCD0 | Change Notification |
| | RD0 | GPIO |
| pd1 | RPD1 | PPS |
| | IOCD1 | Change Notification |
| | RD1 | GPIO |
| pd2 | AN16 | ADC |
| | CVD16 | CVD |
| | CVDR16 | CVD |
| | CVDT0 | CVD |
| | RPD2 | PPS |
| | IOCD2 | Change Notification |
| | RD2 | GPIO |
| pd3 | AN9 | ADC |
| | CVD9 | CVD |
| | CVDR9 | CVD |
| | CVDT9 | CVD |
| | RPD3 | PPS |
| | IOCD3 | Change Notification |
| | RD3 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pd4 | AN10 | ADC |
| | CVD10 | CVD |
| | CVDR10 | CVD |
| | CVDT10 | CVD |
| | RPD4 | PPS |
| | IOCD4 | Change Notification |
| | RD4 | GPIO |
| pd5 | AN11 | ADC |
| | CVD11 | CVD |
| | CVDR11 | CVD |
| | CVDT11 | CVD |
| | RPD5 | PPS |
| | IOCD5 | Change Notification |
| | RD5 | GPIO |
| pd6 | AN12 | ADC |
| | CVD12 | CVD |
| | CVDR12 | CVD |
| | CVDT12 | CVD |
| | GMAC_GMDIO | Ethernet |
| | RPD6 | PPS |
| | IOCD6 | Change Notification |
| | RD6 | GPIO |
| pd7 | AN13 | ADC |
| | CVD13 | CVD |
| | CVDR13 | CVD |
| | CVDT13 | CVD |
| | GMAC_GMDC | Ethernet |
| | RPD7 | PPS |
| | IOCD7 | Change Notification |
| | RD7 | GPIO |
| pe0 | SERCOM2_PAD0 | SERCOM2 |
| | RPE0 | PPS |
| | IOCE0 | Change Notification |
| | RE0 | GPIO |
| pe1 | SERCOM2_PAD1 | SERCOM2 |
| | GMAC_GTX1 | Ethernet |
| | RPE1 | PPS |
| | IOCE1 | Change Notification |
| | RE1 | GPIO |
| pe2 | GMAC_GCRS_DV | Ethernet |
| | RPE2 | PPS |
| | IOCE2 | Change Notification |
| | RE2 | GPIO |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-15.** Priority for Device Pins PAn (n = 0-15), PBn (n = 0-15), PCn (n = 0-10), PDn (n = 0-7), PEn (n = 0-6), (continued)

| Pin Name | Function In Priority Order | Reference Peripheral |
|---|---|---|
| pe3 | RPE3 | PPS |
| | IOCE3 | Change Notification |
| | RE3 | GPIO |
| pe4 | RPE4 | PPS |
| | IOCE4 | Change Notification |
| | RE4 | GPIO |
| pe5 | ANB0 | ADC (Alternate) |
| | RPE5 | PPS |
| | IOCE5 | Change Notification |
| | RE5 | GPIO |
| pe6 | AN14 | ADC |
| | CVD14 | CVD |
| | CVDR14 | CVD |
| | CVDT14 | CVD |
| | RPE6 | PPS |
| | IOCE6 | Change Notification |
| | RE6 | GPIO |

## 6.8. Operation in Power Saving Modes

### 6.8.1. I/O Port Operation in Sleep Mode

As the device enters Sleep mode, the system clock is disabled; however, the CN module continues to operate. If one of the enabled CN pins changes state, the corresponding Interrupt flag is set in NVIC, the device wakes from Sleep (or Idle) mode, then executes the CN interrupt service routine.

### 6.8.2. I/O Port Operation in Idle Mode

As the device enters Idle mode, the system clock sources remain functional.

## 6.9. Results of Various Resets

**Table 6-16.** Results of Resets Available

| Reset Name | Description |
|---|---|
| Device Reset | All I/O registers are forced to their reset states upon a device Reset. |
| Power-on Reset (PoR) | All I/O registers are forced to their reset states upon a Power-on Reset (POR). |
| Watchdog Reset | All I/O registers are unchanged upon a Watchdog Reset. |


© 2025 Microchip Technology Inc. and its subsidiaries

## 6.10. Port Register Summary

See *PORT A* module in the *Product Memory Mapping Overview* from Related Links for base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at their virtual address plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | ANSELA | 7:0 | | | | | ANSA3 | | | |
| | | 15:8 | | ANSA14 | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | TRISA | 7:0 | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx |
| | | 15:8 | TRISAx | TRISAx | TRISAx | | | TRISAx | TRISAx | TRISAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | PORTA | 7:0 | RAx | RAx | RAx | RAx | RAx | RAx | RAx | RAx |
| | | 15:8 | RAx | RAx | RAx | RAx | RAx | RAx | RAx | RAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | LATA | 7:0 | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx |
| | | 15:8 | LATAx | LATAx | LATAx | | | LATAx | LATAx | LATAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x34 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | ODCA | 7:0 | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx |
| | | 15:8 | ODCAx | ODCAx | ODCAx | | | ODCAx | ODCAx | ODCAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x44 ... 0x4F | Reserved | | | | | | | | | |
| 0x50 | CNPUA | 7:0 | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx |
| | | 15:8 | CNPUAx | CNPUAx | CNPUAx | | | CNPUAx | CNPUAx | CNPUAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x54 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | CNPDA | 7:0 | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx |
| | | 15:8 | CNPDAx | CNPDAx | CNPDAx | | | CNPDAx | CNPDAx | CNPDAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x64 ... 0x6F | Reserved | | | | | | | | | |
| 0x70 | CNCONA | 7:0 | | | | | | | | |
| | | 15:8 | ON | FRZ | SIDL | | EDGEDETECT | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x74 ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | CNENA | 7:0 | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx |
| | | 15:8 | CNENAx | CNENAx | CNENAx | | | CNENAx | CNENAx | CNENAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x84 ... 0x8F | Reserved | | | | | | | | | |
| 0x90 | CNSTATA | 7:0 | CNSTATAx | CNSTATAx | CNSTATAx | CNSTATAx | CNSTATAx | CNSTATAx | CNSTATAx | CNSTATAx |
| | | 15:8 | CNSTATAx | CNSTATAx | CNSTATAx | | | CNSTATAx | CNSTATAx | CNSTATAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x94 ... 0x9F | Reserved | | | | | | | | | |
| 0xA0 | CNNEA | 7:0 | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx |
| | | 15:8 | CNNEAx | CNNEAx | CNNEAx | | | CNNEAx | CNNEAx | CNNEAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xA4 ... 0xAF | Reserved | | | | | | | | | |
| 0xB0 | CNFA | 7:0 | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx |
| | | 15:8 | CNFAx | CNFAx | CNFAx | | | CNFAx | CNFAx | CNFAx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xB4 ... 0xBF | Reserved | | | | | | | | | |
| 0xC0 | SRCON0A | 7:0 | SR0Ax | SR0Ax | SR0Ax | SR0Ax | | SR0Ax | SR0Ax | SR0Ax |
| | | 15:8 | | SR0Ax | SR0Ax | | | SR0Ax | SR0Ax | SR0Ax |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xC4 ... 0xCF | Reserved | | | | | | | | | |
| 0xD0 | SRCON1A | 7:0 | SR1Ax | SR1Ax | SR1Ax | SR1Ax | | SR1Ax | SR1Ax | SR1Ax |
| | | 15:8 | | SR1Ax | SR1Ax | | | SR1Ax | SR1Ax | SR1Ax |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xD4 ... 0xFF | Reserved | | | | | | | | | |
| 0x0100 | ANSELB | 7:0 | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx |
| | | 15:8 | ANSBx | ANSBx | | | ANSBx | ANSBx | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0104 ... 0x010F | Reserved | | | | | | | | | |
| 0x0110 | TRISB | 7:0 | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx |
| | | 15:8 | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0114 ... 0x011F | Reserved | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0120 | PORTB | 7:0 | RBx | RBx | RBx | RBx | RBx | RBx | RBx | RBx |
| | | 15:8 | RBx | RBx | RBx | RBx | RBx | RBx | RBx | RBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0124 ... 0x012F | Reserved | | | | | | | | | |
| 0x0130 | LATB | 7:0 | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx |
| | | 15:8 | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0134 ... 0x013F | Reserved | | | | | | | | | |
| 0x0140 | ODCB | 7:0 | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx |
| | | 15:8 | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0144 ... 0x014F | Reserved | | | | | | | | | |
| 0x0150 | CNPUB | 7:0 | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx |
| | | 15:8 | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0154 ... 0x015F | Reserved | | | | | | | | | |
| 0x0160 | CNPDB | 7:0 | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx |
| | | 15:8 | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0164 ... 0x016F | Reserved | | | | | | | | | |
| 0x0170 | CNCONB | 7:0 | | | | | | | | |
| | | 15:8 | ON | FRZ | SIDL | | EDGEDETECT | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0174 ... 0x017F | Reserved | | | | | | | | | |
| 0x0180 | CNENB | 7:0 | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx |
| | | 15:8 | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0184 ... 0x018F | Reserved | | | | | | | | | |
| 0x0190 | CNSTATB | 7:0 | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx |
| | | 15:8 | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0194 ... 0x019F | Reserved | | | | | | | | | |
| 0x01A0 | CNNEB | 7:0 | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx |
| | | 15:8 | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x01A4 ... 0x01AF | Reserved | | | | | | | | | |
| 0x01B0 | CNFB | 7:0 | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx |
| | | 15:8 | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01B4 ... 0x01BF | Reserved | | | | | | | | | |
| 0x01C0 | SRCON0B | 7:0 | | | | | | | | |
| | | 15:8 | | | SR0Bx | SR0Bx | SR0Bx | | SR0Bx | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01C4 ... 0x01CF | Reserved | | | | | | | | | |
| 0x01D0 | SRCON1B | 7:0 | | | | | | | | |
| | | 15:8 | | | SR1Bx | SR1Bx | SR1Bx | | SR1Bx | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x01D4 ... 0x01FF | Reserved | | | | | | | | | |
| 0x0200 | ANSELC | 7:0 | ANSCx | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0204 ... 0x020F | Reserved | | | | | | | | | |
| 0x0210 | TRISC | 7:0 | TRISCx | | | | | | TRISCx | TRISCx |
| | | 15:8 | | | | | TRISCx | TRISCx | TRISCx | TRISCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0214 ... 0x021F | Reserved | | | | | | | | | |
| 0x0220 | PORTC | 7:0 | RCx | | | | | | RCx | RCx |
| | | 15:8 | | | | | RCx | RCx | RCx | RCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0224 ... 0x022F | Reserved | | | | | | | | | |
| 0x0230 | LATC | 7:0 | LATCx | | | | | | LATCx | LATCx |
| | | 15:8 | | | | | LATCx | LATCx | LATCx | LATCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0234 ... 0x023F | Reserved | | | | | | | | | |
| 0x0240 | ODCC | 7:0 | OSCCx | | | | | | OSCCx | OSCCx |
| | | 15:8 | | | | | OSCCx | OSCCx | OSCCx | OSCCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0244 ... 0x024F | Reserved | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0250 | CNPUC | 7:0 | CNPUCx | | | | | | CNPUCx | CNPUCx |
| | | 15:8 | | | | | CNPUCx | CNPUCx | CNPUCx | CNPUCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0254 ... 0x025F | Reserved | | | | | | | | | |
| 0x0260 | CNPDC | 7:0 | CNPDCx | | | | | | CNPDCx | CNPDCx |
| | | 15:8 | | | | | CNPDCx | CNPDCx | CNPDCx | CNPDCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0264 ... 0x026F | Reserved | | | | | | | | | |
| 0x0270 | CNCONC | 7:0 | | | | | | | | |
| | | 15:8 | ON | FRZ | SIDL | | EDGEDETECT | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0274 ... 0x027F | Reserved | | | | | | | | | |
| 0x0280 | CNENC | 7:0 | CNENCx | | | | | | CNENCx | CNENCx |
| | | 15:8 | | | | | CNENCx | CNENCx | CNENCx | CNENCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0284 ... 0x028F | Reserved | | | | | | | | | |
| 0x0290 | CNSTATC | 7:0 | CNSTATCx | | | | | | CNSTATCx | CNSTATCx |
| | | 15:8 | | | | | CNSTATCx | CNSTATCx | CNSTATCx | CNSTATCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0294 ... 0x029F | Reserved | | | | | | | | | |
| 0x02A0 | CNNEC | 7:0 | CNNECx | | | | | | CNNECx | CNNECx |
| | | 15:8 | | | | | CNNECx | CNNECx | CNNECx | CNNECx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x02A4 ... 0x02AF | Reserved | | | | | | | | | |
| 0x02B0 | CNFC | 7:0 | CNFCx | | | | | | CNFCx | CNFCx |
| | | 15:8 | | | | | CNFCx | CNFCx | CNFCx | CNFCx |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x02B4 ... 0x02BF | Reserved | | | | | | | | | |
| 0x02C0 | SRCON0C | 7:0 | | | | | | | SR0Cx | SR0Cx |
| | | 15:8 | | | | | | | SR0Cx | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x02C4 ... 0x02CF | Reserved | | | | | | | | | |
| 0x02D0 | SRCON1C | 7:0 | | | | | | | SR1Cx | SR1Cx |
| | | 15:8 | | | | | | | SR1Cx | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x02D4 ... 0x02FF | Reserved | | | | | | | | | |
| 0x0300 | ANSELD | 7:0 | ANSDx | ANSDx | ANSDx | ANSDx | ANSDx | ANSDx | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0304 ... 0x030F | Reserved | | | | | | | | | |
| 0x0310 | TRISD | 7:0 | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0314 ... 0x031F | Reserved | | | | | | | | | |
| 0x0320 | PORTD | 7:0 | RDx | RDx | RDx | RDx | RDx | RDx | RDx | RDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0324 ... 0x032F | Reserved | | | | | | | | | |
| 0x0330 | LATD | 7:0 | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0334 ... 0x033F | Reserved | | | | | | | | | |
| 0x0340 | ODCD | 7:0 | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0344 ... 0x034F | Reserved | | | | | | | | | |
| 0x0350 | CNPUD | 7:0 | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0354 ... 0x035F | Reserved | | | | | | | | | |
| 0x0360 | CNPDD | 7:0 | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0364 ... 0x036F | Reserved | | | | | | | | | |
| 0x0370 | CNCOND | 7:0 | | | | | | | | |
| | | 15:8 | ON | FRZ | SIDL | | EDGEDETECT | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0374 ... 0x037F | Reserved | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0380 | CNEND | 7:0 | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0384 ... 0x038F | Reserved | | | | | | | | | |
| 0x0390 | CNSTATD | 7:0 | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0394 ... 0x039F | Reserved | | | | | | | | | |
| 0x03A0 | CNNED | 7:0 | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x03A4 ... 0x03AF | Reserved | | | | | | | | | |
| 0x03B0 | CNFD | 7:0 | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x03B4 ... 0x03BF | Reserved | | | | | | | | | |
| 0x03C0 | SRCON0D | 7:0 | | | | | | | SR0Dx | SR0Dx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x03C4 ... 0x03CF | Reserved | | | | | | | | | |
| 0x03D0 | SRCON1D | 7:0 | | | | | | | SR1Dx | SR1Dx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x03D4 ... 0x03FF | Reserved | | | | | | | | | |
| 0x0400 | ANSELE | 7:0 | | ANSEx | ANSEx | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0404 ... 0x040F | Reserved | | | | | | | | | |
| 0x0410 | TRISE | 7:0 | | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0414 ... 0x041F | Reserved | | | | | | | | | |
| 0x0420 | PORTE | 7:0 | | REx | REx | REx | REx | REx | REx | REx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x0424 ... 0x042F | Reserved | | | | | | | | | |
| 0x0430 | LATE | 7:0 | | LATEx | LATEx | LATEx | LATEx | LATEx | LATEx | LATEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0434 ... 0x043F | Reserved | | | | | | | | | |
| 0x0440 | ODCE | 7:0 | | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0444 ... 0x044F | Reserved | | | | | | | | | |
| 0x0450 | CNPUE | 7:0 | | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0454 ... 0x045F | Reserved | | | | | | | | | |
| 0x0460 | CNPDE | 7:0 | | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0464 ... 0x046F | Reserved | | | | | | | | | |
| 0x0470 | CNCONE | 7:0 | | | | | | | | |
| | | 15:8 | ON | FRZ | SIDL | | EDGEDETECT | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0474 ... 0x047F | Reserved | | | | | | | | | |
| 0x0480 | CNENE | 7:0 | | CNENEx | CNENEx | CNENEx | CNENEx | CNENEx | CNENEx | CNENEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0484 ... 0x048F | Reserved | | | | | | | | | |
| 0x0490 | CNSTATE | 7:0 | | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0494 ... 0x049F | Reserved | | | | | | | | | |
| 0x04A0 | CNNEE | 7:0 | | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04A4 ... 0x04AF | Reserved | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Port Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x04B0 | CNFE | 7:0 | | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04B4 ... 0x04BF | Reserved | | | | | | | | | |
| 0x04C0 | SRCON0E | 7:0 | | | | | | | SR0Ex | SR0Ex |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04C4 ... 0x04CF | Reserved | | | | | | | | | |
| 0x04D0 | SRCON1E | 7:0 | | | | | | | SR1Ex | SR1Ex |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 6.11. 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.

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.1. Analog Select Register for PortA

**Name:** ANSELA
**Offset:** 0x00
**Reset:** 0x8
**Property:** -

The ANSELA register controls the operation of the analog portA pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | ANSA14 | | | | | | |
| Access | | R/W | | | | | | |
| Reset | | 0 | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ANSA3 | | | |
| Access | | | | | R/W | | | |
| Reset | | | | | 0 | | | |

**Bit 14 – ANSA14** Analog Select for PA14
Configures the PA14 as an analog input when this bit is set to '1'.

**Bit 3 – ANSA3** Analog Select for PA3
Configures the PA3 as an analog input when this bit is set to '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.2. Tri-state Functions for PortA

**Name:** TRISA
**Offset:** 0x10
**Reset:** 0x0
**Property:** -

The TRISA register configures the data direction flow through port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | TRISAx | TRISAx | TRISAx | | | TRISAx | TRISAx | TRISAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx | TRISAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – TRISAx** (x = 0 to 15; x = 0 for bit0 mapped to PA0, ... x = 15 for bit15 mapped to PA15) Tri-state pins for PortA

The tri-state data direction bit configures the selected I/O pin of Port A as an input or output.

| Value | Description |
|---|---|
| 1 | Configures the I/O as input |
| 0 | Configures the I/O as output |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.3. Port Pin Data for PortA

**Name:** PORTA
**Offset:** 0x20
**Reset:** 0x0
**Property:** -

A write to a PORTA register writes to the corresponding LATA register (PORTA data latch). Those I/O port pin(s) configured as outputs are updated. A write to a PORTA register is effectively the same as a write to a LATA register. A read from a PORTA register reads the synchronized signal applied to the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | RAx | RAx | RAx | RAx | RAx | RAx | RAx | RAx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | RAx | RAx | RAx | RAx | RAx | RAx | RAx | RAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – RAx** (x = 0 to 15; x = 0 for bit0 mapped to PA0, ... x = 15 for bit15 mapped to PA15) Port pin configuration for PortA


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.4. Latch Functions for PortA

**Name:** LATA
**Offset:** 0x30
**Reset:** 0x0
**Property:** -

The LATA register (PORTA data latch) holds data written to port I/O pins. A write to a LATA register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated. A read from a LATA register reads the data held in the PORTA data latch, not from the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | LATAx | LATAx | LATAx | | | LATAx | LATAx | LATAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx | LATAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – LATAx** (x = 0 to 15; x = 0 for bit0 mapped to PA0, ... x = 15 for bit15 mapped to PA15) Port pin configuration for PortA


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.5. Open-Drain Configuration for PortA

**Name:** ODCA
**Offset:** 0x40
**Reset:** 0x0
**Property:** -

This register configures each I/O pin individually for either normal digital output or open-drain output, associated with each I/O pin.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ODCAx | ODCAx | ODCAx | | | ODCAx | ODCAx | ODCAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx | ODCAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – ODCAx** (x = 0 to 15; x = 0 for bit0 mapped to PA0, ... x = 15 for bit15 mapped to PA15 Open-Drain Configuration for PortA

**Note:** After a Reset, the status of all the bits of the ODCA register is set to '0'.

| Value | Description |
|---|---|
| 1 | Configures an I/O pin as an open-drain output. |
| 0 | Configures an I/O pin as an normal digital output. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.6. Pull-up Configuration for PortA

**Name:** CNPUA
**Offset:** 0x50
**Reset:** 0x0
**Property:** -

This register enables the weak internal pull-ups connected with an I/O pin when any of the control bits is set.
**Note:** If configuring the port pin as a digital output, the user must always disable this 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPUAx | CNPUAx | CNPUAx | | | CNPUAx | CNPUAx | CNPUAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx | CNPUAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNPUAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Pull-up configuration for PortA

| Value | Description |
|---|---|
| 1 | Enables the weak pull-up associated with an I/O pin. |
| 0 | Disables the weak pull-up associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.7. Pull-down Control Register Configuration for PortA

**Name:** CNPDA
**Offset:** 0x60
**Reset:** 0x0
**Property:** -

This register enables the weak pull-down connected with an I/O pin when any of the control bits is set.
**Note:** If configuring the port pin as a digital output, the user must always disable this 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPDAx | CNPDAx | CNPDAx | | | CNPDAx | CNPDAx | CNPDAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx | CNPDAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNPDAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Pull-down configuration for PortA

| Value | Description |
|---|---|
| 1 | Enables the weak pull-down associated with an I/O pin. |
| 0 | Disables the weak pull-down associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.8. Change Notice Control for PortA

**Name:** CNCONA
**Offset:** 0x70
**Reset:** 0x0
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | | EDGEDETECT | | | |
| Access | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | | 0 | | | |

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

**Bit 15 – ON** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Change Notice is enabled |
| 0 | Change Notice is disabled |

**Bit 14 – FRZ** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Freezes the module operation when in the Debug mode |
| 0 | Continues the module operation when in the Debug mode |

**Bit 13 – SIDL** Stop in the Idle mode bit

| Value | Description |
|---|---|
| 1 | Discontinues the module operation when device enters the Idle mode |
| 0 | Continues the module operation even in the Idle mode |

**Bit 11 – EDGEDETECT** Change Notification Style bit

| Value | Description |
|---|---|
| 1 | Edge Style. Detects edge transitions. This is associated with CNENA (positive edge)/CNNEA (negative edge)/ CNFA. |
| 0 | Mismatch Style. Detects change from last PortA read. This is associated with CNENA/CNSTATA. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.9. Change Notice Enable for PortA

**Name:** CNENA
**Offset:** 0x80
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When EDGEDETECT is set, CNENA controls the positive edge. CNENA enables a mismatch CN interrupt condition when EDGEDETECT is not 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 |
|---|---|---|---|---|---|---|---|---|
| | CNENAx | CNENAx | CNENAx | | | CNENAx | CNENAx | CNENAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx | CNENAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNENAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Change Notice Enable for PortA

| Value | Description |
|---|---|
| 1 | Enables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.10. Change Notice Status for PortA

**Name:** CNSTATA
**Offset:** 0x90
**Reset:** 0x0
**Property:** -

This register indicates whether a change occurred on the corresponding pin since the last read of the PortA bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNSTATAx | CNSTATAx | CNSTATAx | | | CNSTATAx | CNSTATAx | CNSTATAx |
| Access | R | R | R | | | R | R | R |
| Reset | 0 | 0 | 0 | | | 0 | 0 | 0 |

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

**Bits 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNSTATAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Change Notice Status for PortA

'1' indicates change occurred in an I/O pin.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.11. Change Notice Enable for PortA

**Name:** CNNEA
**Offset:** 0xA0
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When EDGEDETECT is set, CNNEA controls the negative edge.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNNEAx | CNNEAx | CNNEAx | | | CNNEAx | CNNEAx | CNNEAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx | CNNEAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNNEAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Change Notice Enable for PortA

| Value | Description |
|---|---|
| 1 | Enables a negative edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a negative edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.12. Change Notice Flag for PortA

**Name:** CNFA
**Offset:** 0xB0
**Reset:** 0x0
**Property:** -

This register indicates edge-detect style change occurred on the corresponding pin since the last read of the PortA bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNFAx | CNFAx | CNFAx | | | CNFAx | CNFAx | CNFAx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx | CNFAx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,13,14,15 – CNFAx** (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15; x = 0 for bit0 mapped to PA0, ... x = 10 for bit 10, x = 13 for bit13, ... x = 15 for bit 15 mapped to PA15) Change Notice Flag for PortA
When CNCONAx = 1, CNFAx stores the occurrence of the CN event until cleared by the software.
When CNCONAx = 0, CNFAx Reads '0'.

| Value | Description |
|---|---|
| 1 | An Enabled Edge Event occurred on pin PORTAx |
| 0 | An Enabled Edge Event did not occur on pin PORTAx |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.13. Slew Rate Control 0 for PortA

**Name:** SRCON0A
**Offset:** 0xC0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port A.
**Note:** To configure the slew rate, the user must also configure the SRCON1A register associated with Port A. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | SR0Ax | SR0Ax | | | SR0Ax | SR0Ax | SR0Ax |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | SR0Ax | SR0Ax | SR0Ax | SR0Ax | | SR0Ax | SR0Ax | SR0Ax |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bits 0,1,2,4,5,6,7,8,9,10,13,14 – SR0Ax** (x = 0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 13, 14; x = 0 for bit0 mapped to PA0, ... x = 2 for bit 2, x = 4 for bit4, ... x = 10 for bit 10, x = 13 for bit13, ... x = 14 for bit 14 mapped to PA14) Slew Rate Control 0 for PortA.

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.14. Slew Rate Control 1 for PortA

**Name:** SRCON1A
**Offset:** 0xD0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port A.
**Note:** To configure the slew rate, the user must also configure the SRCON0A register associated with Port A. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | SR1Ax | SR1Ax | | | SR1Ax | SR1Ax | SR1Ax |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | SR1Ax | SR1Ax | SR1Ax | SR1Ax | | SR1Ax | SR1Ax | SR1Ax |
| Access | R/W | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |

**Bits 0,1,2,4,5,6,7,8,9,10,13,14 – SR1Ax** (x = 0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 13, 14; x = 0 for bit0, ... x = 2 for bit 2, x = 4 for bit4, ... x = 10 for bit10, x = 13 for bit13, ... x = 14 for bit14) Slew Rate Control 1 for PortA

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.15. Analog Select Register for PortB

**Name:** ANSELB
**Offset:** 0x100
**Reset:** 0x8
**Property:** -

The ANSELB register controls the operation of the analog PortB pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ANSBx | ANSBx | | | ANSBx | ANSBx | | |
| Access | R/W | R/W | | | R/W | R/W | | |
| Reset | 0 | 0 | | | 0 | 0 | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx | ANSBx |
| Access | R/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 0,1,2,3,4,5,6,7,10,11,14,15 – ANSBx** (x = 0,1,2,3,4,5,6,7,10,11,14,15; x = 0 for bit0, ... x = 15 for bit15)
Analog Select for PB

Configures the PB as an analog input when this bit is set to '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.16. Tri-state Functions for PortB

**Name:** TRISB
**Offset:** 0x110
**Reset:** 0x0
**Property:** -

The TRISB register configures the data direction flow through port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx | TRISBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – TRISBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Tri-state pins for PortB

The tri-state data direction bit configures the selected I/O pin of Port B as an input or output.

| Value | Description |
|---|---|
| 1 | Configures the I/O as input |
| 0 | Configures the I/O as output |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.17. Port Pin Data for PortB

**Name:** PORTB
**Offset:** 0x120
**Reset:** 0x0
**Property:** -

A write to a PORTB register writes to the corresponding LATB register (PORTB data latch). Those I/O port pin(s) configured as outputs are updated. A write to a PORTB register is effectively the same as a write to a LATB register. A read from a PORTB register reads the synchronized signal applied to the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | RBx | RBx | RBx | RBx | RBx | RBx | RBx | RBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | RBx | RBx | RBx | RBx | RBx | RBx | RBx | RBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – RBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) PortB configuration


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.18. Latch Functions for PortB

**Name:** LATB
**Offset:** 0x130
**Reset:** 0x0
**Property:** -

The LATB register (PORTB data latch) holds data written to port I/O pins. A write to a LATB register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated. A read from a LATB register reads the data held in the PORTB data latch, not from the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx | LATBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – LATBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Latch configuration for PortB


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.19. Open-Drain Configuration for PortB

**Name:** ODCB
**Offset:** 0x140
**Reset:** 0x0
**Property:** -

This register configures each I/O pin individually for either normal digital output or open-drain output, associated with each I/O pin.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx | ODCBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – ODCBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Open-Drain Configuration for PortB

**Note:** After a Reset, the status of all the bits of the ODCB register is set to '0'.

| Value | Description |
|---|---|
| 1 | Configures an I/O pin as an open-drain output. |
| 0 | Configures an I/O pin as an normal digital output. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.20. Change Notice Pull-up for PortB

**Name:** CNPUB
**Offset:** 0x150
**Reset:** 0x0
**Property:** -

This register enables the weak internal pull-ups connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx | CNPUBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNPUBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Pull-up configuration for PortB

| Value | Description |
|---|---|
| 1 | Enables the weak pull-up associated with an I/O pin. |
| 0 | Disables the weak pull-up associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.21. Change Notice Pull-down for PortB

**Name:** CNPDB
**Offset:** 0x160
**Reset:** 0x0
**Property:** -

This register enables the weak pull-down connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx | CNPDBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNPDBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Pull-down configuration for PortB

| Value | Description |
|---|---|
| 1 | Enables the weak pull-down associated with an I/O pin. |
| 0 | Disables the weak pull-down associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.22. Change Notice Control for PortB

**Name:** CNCONB
**Offset:** 0x170
**Reset:** 0x0
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | | EDGEDETECT | | | |
| Access | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | | 0 | | | |

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

**Bit 15 – ON** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Change Notice is enabled |
| 0 | Change Notice is disabled |

**Bit 14 – FRZ** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Freezes the module operation when in the Debug mode |
| 0 | Continues the module operation when in the Debug mode |

**Bit 13 – SIDL** Stop in the Idle mode bit

| Value | Description |
|---|---|
| 1 | Discontinues the module operation when device enters the Idle mode |
| 0 | Continues the module operation even in the Idle mode |

**Bit 11 – EDGEDETECT** Change Notification Style bit

| Value | Description |
|---|---|
| 1 | Edge Style. Detects edge transitions. This is associated with CNENA (positive edge)/CNNEA (negative edge)/ CNFA. |
| 0 | Mismatch Style. Detects change from last PortA read. This is associated with CNENA/CNSTATA. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.23. Change Notice Enable for PortB

**Name:** CNENB
**Offset:** 0x180
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNENB controls the positive edge. CNENB enables a mismatch CN interrupt condition when CNSTYLE is not 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 |
|---|---|---|---|---|---|---|---|---|
| | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx | CNENBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNENBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Enable for PortB

| Value | Description |
|---|---|
| 1 | Enables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.24. Change Notice Status for PortB

**Name:** CNSTATB
**Offset:** 0x190
**Reset:** 0x0
**Property:** -

This register indicates whether a change occurred on the corresponding pin since the last read of the PortB bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx | CNSTATBx |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNSTATBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Status for PortB
'1' indicates change occurred in an I/O pin.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.25. Change Notice Enable for PortB

**Name:** CNNEB
**Offset:** 0x1A0
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When EDGEDETECT is set, CNNEB controls the negative edge.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx | CNNEBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNNEBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Enable for PortB

| Value | Description |
|---|---|
| 1 | Enables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.26. Change Notice Flag for PortB

**Name:** CNFB
**Offset:** 0x1B0
**Reset:** 0x0
**Property:** -

This register indicates the edge-detect style change occurred on the corresponding pin since the last read of the PortB bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx | CNFBx |
| Access | R/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 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 – CNFBx** (x = 0 to 15; x = 0 for bit0 mapped to PB0, ... x = 15 for bit15 mapped to PB15) Change Notice Flag for PortB

When CNCONBx = 1, CNFBx stores the occurrence of the CN event until cleared by the software.
When CNCONBx = 0, CNFBx Reads '0'.

| Value | Description |
|---|---|
| 1 | An Enabled Edge Event occurred on pin PORTBx |
| 0 | An Enabled Edge Event did not occur on pin PORTBx |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.27. Slew Rate Control 0 for PortB

**Name:** SRCON0B
**Offset:** 0x1C0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port B.
**Note:** To configure the slew rate, the user must also configure the SRCON1B register associated with PortB. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | SR0Bx | SR0Bx | SR0Bx | | SR0Bx | |
| Access | | | R/W | R/W | R/W | | R/W | |
| Reset | | | 0 | 0 | 0 | | 0 | |

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

**Bits 9,11,12,13 – SR0Bx** (x = 9, 11, 12, 13; x = 9 for bit9 mapped to PB9, ... x = 13 for bit13 mapped to PB13)
Slew Rate Control 0 for PortB

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.28. Slew Rate Control 1 for PortB

**Name:** SRCON1B
**Offset:** 0x1D0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port B.
**Note:** To configure the slew rate, the user must also configure the SRCON0B register associated with Port B. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | SR1Bx | SR1Bx | SR1Bx | | SR1Bx | |
| Access | | | R/W | R/W | R/W | | R/W | |
| Reset | | | 0 | 0 | 0 | | 0 | |

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

**Bits 9,11,12,13 – SR1Bx** (x = 9, 11, 12, 13; x = 9 for bit9 mapped to PB9, ... x = 13 for bit13 mapped to PB13)
Slew Rate Control 1 for PortB

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.29. Analog Select Register for PortC

**Name:** ANSELC
**Offset:** 0x200
**Reset:** 0x8
**Property:** -

The ANSELC register controls the operation of the analog PortC pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ANSCx | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

**Bit 7 – ANSCx** (x = 7 for bit7) Analog Select for PC
Configures the PC as an analog input when this bit is set to '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.30. Tri-state Functions for PortC

**Name:** TRISC
**Offset:** 0x210
**Reset:** 0x0
**Property:** -

The TRISC register configures the data direction flow through port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TRISCx | TRISCx | TRISCx | TRISCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – TRISCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Tri-state pins for PortC

The tri-state data direction bit configures the selected I/O pin of Port C as an input or output.

| Value | Description |
|---|---|
| 1 | Configures the I/O as input |
| 0 | Configures the I/O as output |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.31. Port Pin Data for PortC

**Name:** PORTC
**Offset:** 0x220
**Reset:** 0x0
**Property:** -

A write to a PORTC register writes to the corresponding LATC register (PORTC data latch). Those I/O port pin(s) configured as outputs are updated. A write to a PORTC register is effectively the same as a write to a LATC register. A read from a PORTC register reads the synchronized signal applied to the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | RCx | RCx | RCx | RCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – RCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) PortC configuration


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.32. Latch Functions for PortC

**Name:** LATC
**Offset:** 0x230
**Reset:** 0x0
**Property:** -

The LATC register (PORTC data latch) holds data written to port I/O pins. A write to a LATC register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated. A read from a LATC register reads the data held in the PORTC data latch, not from the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LATCx | LATCx | LATCx | LATCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – LATCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Latch configuration for PortC


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.33. Open-Drain Configuration for PortC

**Name:** ODCC
**Offset:** 0x240
**Reset:** 0x0
**Property:** -

This register configures each I/O pin individually for either normal digital output or open-drain output, associated with each I/O pin.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | OSCCx | OSCCx | OSCCx | OSCCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – OSCCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Open-Drain Configuration for PortC

**Note:** After a Reset, the status of all the bits of the ODCC register is set to '0'.

| Value | Description |
|---|---|
| 1 | Configures an I/O pin as an open-drain output. |
| 0 | Configures an I/O pin as an normal digital output. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.34. Change Notice Pull-up for PortC

**Name:** CNPUC
**Offset:** 0x250
**Reset:** 0x0
**Property:** -

This register enables the weak internal pull-ups connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNPUCx | CNPUCx | CNPUCx | CNPUCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNPUCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Pull-up configuration for PortC

| Value | Description |
|---|---|
| 1 | Enables the weak pull-up associated with an I/O pin. |
| 0 | Disables the weak pull-up associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.35. Change Notice Pull-down for PortC

**Name:** CNPDC
**Offset:** 0x260
**Reset:** 0x0
**Property:** -

This register enables the weak pull-down connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNPDCx | CNPDCx | CNPDCx | CNPDCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNPDCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Pull-down configuration for PortC

| Value | Description |
|---|---|
| 1 | Enables the weak pull-down associated with an I/O pin. |
| 0 | Disables the weak pull-down associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

## 6.11.36. Change Notice Control for PortC

**Name:** CNCONC
**Offset:** 0x270
**Reset:** 0x0
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | | EDGEDETECT | | | |
| Access | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | | 0 | | | |

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

**Bit 15 – ON** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Change Notice is enabled |
| 0 | Change Notice is disabled |

**Bit 14 – FRZ** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Freezes the module operation when in the Debug mode |
| 0 | Continues the module operation when in the Debug mode |

**Bit 13 – SIDL** Stop in the Idle mode bit

| Value | Description |
|---|---|
| 1 | Discontinues the module operation when device enters the Idle mode |
| 0 | Continues the module operation even in the Idle mode |

**Bit 11 – EDGEDETECT** Change Notification Style bit

| Value | Description |
|---|---|
| 1 | Edge Style. Detects edge transitions. This is associated with CNENA (positive edge)/CNNEA (negative edge)/ CNFA. |
| 0 | Mismatch Style. Detects change from last PortA read. This is associated with CNENA/CNSTATA. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.37. Change Notice Enable for PortC

**Name:** CNENC
**Offset:** 0x280
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNENC controls the positive edge. CNENC enables a mismatch CN interrupt condition when CNSTYLE is not 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNENCx | CNENCx | CNENCx | CNENCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNENCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Enable for PortC

| Value | Description |
|---|---|
| 1 | Enables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.38. Change Notice Status for PortC

**Name:** CNSTATC
**Offset:** 0x290
**Reset:** 0x0
**Property:** -

This register indicates whether a change occurred on the corresponding pin since the last read of the PortC bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNSTATCx | CNSTATCx | CNSTATCx | CNSTATCx |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNSTATCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Status for PortC
'1' indicates change occurred in an I/O pin.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.39. Change Notice Enable for PortC

**Name:** CNNEC
**Offset:** 0x2A0
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNNEC controls the negative edge.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNNECx | CNNECx | CNNECx | CNNECx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNNECx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Enable for PortC

| Value | Description |
|---|---|
| 1 | Enables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.40. Change Notice Flag for PortC

**Name:** CNFC
**Offset:** 0x2B0
**Reset:** 0x0
**Property:** -

This register indicates the edge-detect style change occurred on the corresponding pin since the last read of the PortC bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CNFCx | CNFCx | CNFCx | CNFCx |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 0,1,7,8,9,10,11 – CNFCx** (x = 0, 1, 7, 8, 9, 10, 11; x = 0 for bit0 mapped to PC0, ... x = 11 for bit11 mapped to PC11) Change Notice Flag for PortC
When CNCONCx = 1, CNFCx stores the occurrence of the CN event until cleared by the software.
When CNCONCx = 0, CNFCx Reads '0'.

| Value | Description |
|---|---|
| 1 | An Enabled Edge Event occurred on pin PORTCx |
| 0 | An Enabled Edge Event did not occur on pin PORTCx |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.41. Slew Rate Control 0 for PortC

**Name:** SRCON0C
**Offset:** 0x2C0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port C.
**Note:** To configure the slew rate, the user must also configure the SRCON1C register associated with PortC. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR0Cx | |
| Access | | | | | | | R/W | |
| Reset | | | | | | | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR0Cx | SR0Cx |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1,9 – SR0Cx** (x = 0, 1, 9; x = 0 for bit0 mapped to PC0, ... x = 9 for bit9 mapped to PC9) Slew Rate Control 0 for PortC

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.42. Slew Rate Control 1 for PortC

**Name:** SRCON1C
**Offset:** 0x2D0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port C.
**Note:** To configure the slew rate, the user must also configure the SRCON0C register associated with Port C. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1,9 – SR1Cx** (x = 0, 1, 9; x = 0 for bit0 mapped to PC0, ... x = 9 for bit9 mapped to PC9) Slew Rate Control 1 for PortC

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.43. Analog Select Register for PortD

**Name:** ANSELD
**Offset:** 0x300
**Reset:** 0x8
**Property:** -

The ANSELD register controls the operation of the analog PortD pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ANSDx | ANSDx | ANSDx | ANSDx | ANSDx | ANSDx | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | | |

**Bits 2,3,4,5,6,7 – ANSDx** (x = 2 to 7; x = 2 for bit2, ... x = 7 for bit7) Analog Select for PD
Configures the PC as an analog input when this bit is set to '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.44. Tri-state Functions for PortD

**Name:** TRISD
**Offset:** 0x310
**Reset:** 0x0
**Property:** -

The TRISD register configures the data direction flow through port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx | TRISDx |
| Access | R/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 – TRISDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Tri-state pins for PortD

The tri-state data direction bit configures the selected I/O pin of Port D as an input or output.

| Value | Description |
|---|---|
| 1 | Configures the I/O as input |
| 0 | Configures the I/O as output |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.45. Port Pin Data for PortD

**Name:** PORTD
**Offset:** 0x320
**Reset:** 0x0
**Property:** -

A write to a PORTD register writes to the corresponding LATD register (PORTD data latch). Those I/O port pin(s) configured as outputs are updated. A write to a PORTD register is effectively the same as a write to a LATD register. A read from a PORTD register reads the synchronized signal applied to the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | RDx | RDx | RDx | RDx | RDx | RDx | RDx | RDx |
| Access | R/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 – RDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7) PortD configuration


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.46. Latch Functions for PortD

**Name:** LATD
**Offset:** 0x330
**Reset:** 0x0
**Property:** -

The LATD register (PORTD data latch) holds data written to port I/O pins. A write to a LATD register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated. A read from a LATD register reads the data held in the PORTD data latch, not from the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx | LATDx |
| Access | R/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 – LATDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7) Latch configuration for PortD


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.47. Open-Drain Configuration for PortD

**Name:** ODCD
**Offset:** 0x340
**Reset:** 0x0
**Property:** -

This register configures each I/O pin individually for either normal digital output or open-drain output, associated with each I/O pin.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx | ODCDx |
| Access | R/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 – ODCDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Open-Drain Configuration for PortD

**Note:** After a Reset, the status of all the bits of the ODCD register is set to '0'.

| Value | Description |
|---|---|
| 1 | Configures an I/O pin as an open-drain output. |
| 0 | Configures an I/O pin as an normal digital output. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.48. Change Notice Pull-up for PortD

**Name:** CNPUD
**Offset:** 0x350
**Reset:** 0x0
**Property:** -

This register enables the weak internal pull-ups connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx | CNPUDx |
| Access | R/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 – CNPUDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Pull-up configuration for PortD

| Value | Description |
|---|---|
| 1 | Enables the weak pull-up associated with an I/O pin. |
| 0 | Disables the weak pull-up associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.49. Change Notice Pull-down for PortD

**Name:** CNPDD
**Offset:** 0x360
**Reset:** 0x0
**Property:** -

This register enables the weak pull-down connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx | CNPDDx |
| Access | R/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 – CNPDDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Pull-down configuration for PortD

| Value | Description |
|---|---|
| 1 | Enables the weak pull-down associated with an I/O pin. |
| 0 | Disables the weak pull-down associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.50. Change Notice Control for PortD

**Name:** CNCOND
**Offset:** 0x370
**Reset:** 0x0
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | | EDGEDETECT | | | |
| Access | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | | 0 | | | |

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

**Bit 15 – ON** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Change Notice is enabled |
| 0 | Change Notice is disabled |

**Bit 14 – FRZ** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Freezes the module operation when in the Debug mode |
| 0 | Continues the module operation when in the Debug mode |

**Bit 13 – SIDL** Stop in the Idle mode bit

| Value | Description |
|---|---|
| 1 | Discontinues the module operation when device enters the Idle mode |
| 0 | Continues the module operation even in the Idle mode |

**Bit 11 – EDGEDETECT** Change Notification Style bit

| Value | Description |
|---|---|
| 1 | Edge Style. Detects edge transitions. This is associated with CNENA (positive edge)/CNNEA (negative edge)/ CNFA. |
| 0 | Mismatch Style. Detects change from last PortA read. This is associated with CNENA/CNSTATA. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.51. Change Notice Enable for PortD

**Name:** CNEND
**Offset:** 0x380
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNEND controls the positive edge. CNEND enables a mismatch CN interrupt condition when CNSTYLE is not 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx | CNENDx |
| Access | R/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 – CNENDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Enable for PortD

| Value | Description |
|---|---|
| 1 | Enables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.52. Change Notice Status for PortD

**Name:** CNSTATD
**Offset:** 0x390
**Reset:** 0x0
**Property:** -

This register indicates whether a change occurred on the corresponding pin since the last read of the PortD bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx | CNSTATDx |
| 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 – CNSTATDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Status for PortD
'1' indicates change occurred in an I/O pin.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.53. Change Notice Enable for PortD

**Name:** CNNED
**Offset:** 0x3A0
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNNED controls the negative edge.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx | CNNEDx |
| Access | R/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 – CNNEDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Enable for PortD

| Value | Description |
|---|---|
| 1 | Enables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.54. Change Notice Flag for PortD

**Name:** CNFD
**Offset:** 0x3B0
**Reset:** 0x0
**Property:** -

This register indicates the edge-detect style change occurred on the corresponding pin since the last read of the PortD bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx | CNFDx |
| Access | R/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 – CNFDx** (x = 0 to 7; x = 0 for bit0 mapped to PD0, ... x = 7 for bit7 mapped to PD7)
Change Notice Flag for PortD

When CNCONDx = 1, CNFDx stores the occurrence of the CN event until cleared by the software.
When CNCONDx = 0, CNFDx Reads '0'.

| Value | Description |
|---|---|
| 1 | An Enabled Edge Event occurred on pin PORTDx |
| 0 | An Enabled Edge Event did not occur on pin PORTDx |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.55. Slew Rate Control 0 for PortD

**Name:** SRCON0D
**Offset:** 0x3C0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with PortD.
**Note:** To configure the slew rate, the user must also configure the SRCON1D register associated with PortD. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR0Dx | SR0Dx |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1 – SR0Dx** (x = 0, 1; x = 0 for bit0 mapped to PC0, ... x = 1 for bit1 mapped to PC1) Slew Rate Control 0 for PortD

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.56. Slew Rate Control 1 for PortD

**Name:** SRCON1D
**Offset:** 0x3D0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with Port D.
**Note:** To configure the slew rate, the user must also configure the SRCON0D register associated with PortD. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR1Dx | SR1Dx |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1 – SR1Dx** (x = 0, 1; x = 0 for bit0 mapped to PC0, ... x = 1 for bit1 mapped to PC1) Slew Rate Control 1 for PortD

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.57. Analog Select Register for PortE

**Name:** ANSELE
**Offset:** 0x400
**Reset:** 0x8
**Property:** -

The ANSELE register controls the operation of the analog PortE pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | ANSEx | ANSEx | | | | | |
| Access | | R/W | R/W | | | | | |
| Reset | | 0 | 0 | | | | | |

**Bits 5,6 – ANSEx** (x = 5 to 6; x = 5 for bit5, ... x = 6 for bit6) Analog Select for PE
Configures the PE as an analog input when this bit is set to '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.58. Tri-state Functions for PortE

**Name:** TRISE
**Offset:** 0x410
**Reset:** 0x0
**Property:** -

The TRISE register configures the data direction flow through port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx | TRISEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – TRISEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Tri-state pins for PortE

The tri-state data direction bit configures the selected I/O pin of Port E as an input or output.

| Value | Description |
|---|---|
| 1 | Configures the I/O as input |
| 0 | Configures the I/O as output |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.59. Port Pin Data for PortE

**Name:** PORTE
**Offset:** 0x420
**Reset:** 0x0
**Property:** -

A write to a PORTE register writes to the corresponding LATE register (PORTE data latch). Those I/O port pin(s) configured as outputs are updated. A write to a PORTE register is effectively the same as a write to a LATE register. A read from a PORTE register reads the synchronized signal applied to the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | REx | REx | REx | REx | REx | REx | REx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – REx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) PortE configuration


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.60. Latch Functions for PortE

**Name:** LATE
**Offset:** 0x430
**Reset:** 0x0
**Property:** -

The LATE register (PORTE data latch) holds data written to port I/O pins. A write to a LATE register latches data to corresponding port I/O pins. Those I/O port pins configured as outputs are updated. A read from a LATE register reads the data held in the PORTE data latch, not from the port I/O pins.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | LATEx | LATEx | LATEx | LATEx | LATEx | LATEx | LATEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – LATEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Latch configuration for PortE


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.61. Open-Drain Configuration for PortE

**Name:** ODCE
**Offset:** 0x440
**Reset:** 0x0
**Property:** -

This register configures each I/O pin individually for either normal digital output or open-drain output, associated with each I/O pin.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx | ODCEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – ODCEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6)
Open-Drain Configuration for PortE

**Note:** After a Reset, the status of all the bits of the ODCE register is set to '0'.

| Value | Description |
|---|---|
| 1 | Configures an I/O pin as an open-drain output. |
| 0 | Configures an I/O pin as an normal digital output. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.62. Change Notice Pull-up for PortE

**Name:** CNPUE
**Offset:** 0x450
**Reset:** 0x0
**Property:** -

This register enables the weak internal pull-ups connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx | CNPUEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – CNPUEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Change Notice Pull-up configuration for PortE

| Value | Description |
|---|---|
| 1 | Enables the weak pull-up associated with an I/O pin. |
| 0 | Disables the weak pull-up associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.63. Change Notice Pull-down for PortE

**Name:** CNPDE
**Offset:** 0x460
**Reset:** 0x0
**Property:** -

This register enables the weak pull-down connected with an I/O pin when any of the control bits is set.
**Note:** This register must always be disabled when the port pin is configured as a digital output.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx | CNPDEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – CNPDEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Change Notice Pull-down configuration for PortE

| Value | Description |
|---|---|
| 1 | Enables the weak pull-down associated with an I/O pin. |
| 0 | Disables the weak pull-down associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.64. Change Notice Control for PortE

**Name:** CNCONE
**Offset:** 0x470
**Reset:** 0x0
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | | EDGEDETECT | | | |
| Access | R/W | R/W | R/W | | R/W | | | |
| Reset | 0 | 0 | 0 | | 0 | | | |

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

**Bit 15 – ON** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Change Notice is enabled |
| 0 | Change Notice is disabled |

**Bit 14 – FRZ** Change Notice (CN) Control ON bit

| Value | Description |
|---|---|
| 1 | Freezes the module operation when in the Debug mode |
| 0 | Continues the module operation when in the Debug mode |

**Bit 13 – SIDL** Stop in the Idle mode bit

| Value | Description |
|---|---|
| 1 | Discontinues the module operation when device enters the Idle mode |
| 0 | Continues the module operation even in the Idle mode |

**Bit 11 – EDGEDETECT** Change Notification Style bit

| Value | Description |
|---|---|
| 1 | Edge Style. Detects edge transitions. This is associated with CNENA (positive edge)/CNNEA (negative edge)/ CNFA. |
| 0 | Mismatch Style. Detects change from last PortA read. This is associated with CNENA/CNSTATA. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.65. Change Notice Enable for PortE

**Name:** CNENE
**Offset:** 0x480
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNENE controls the positive edge. CNENE enables a mismatch CN interrupt condition when CNSTYLE is not 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

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

**Bits 0,1,2,3,4,5,6 – CNENEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Change Notice Enable for PortE

| Value | Description |
|---|---|
| 1 | Enables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/positive edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.66. Change Notice Status for PortE

**Name:** CNSTATE
**Offset:** 0x490
**Reset:** 0x0
**Property:** -

This register indicates whether a change occurred on the corresponding pin since the last read of the PortE bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx | CNSTATEx |
| Access | | R | R | R | R | R | R | R |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – CNSTATEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6)
Change Notice Status for PortE
'1' indicates change occurred in an I/O pin.


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.67. Change Notice Enable for PortE

**Name:** CNNEE
**Offset:** 0x4A0
**Reset:** 0x0
**Property:** -

This register contains the CN interrupt enable control bits for each of the input pins. Setting any of these bits enables a CN interrupt for the corresponding pins. When CNSTYLE is set, CNNEE controls the negative edge.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx | CNNEEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – CNNEEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Change Notice Enable for PortE

| Value | Description |
|---|---|
| 1 | Enables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |
| 0 | Disables a mismatch/negative edge CN interrupt condition associated with an I/O pin. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.68. Change Notice Flag for PortE

**Name:** CNFE
**Offset:** 0x4B0
**Reset:** 0x0
**Property:** -

This register indicates the edge-detect style change occurred on the corresponding pin since the last read of the PortE bit.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx | CNFEx |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 0,1,2,3,4,5,6 – CNFEx** (x = 0 to 6; x = 0 for bit0 mapped to PE0, ... x = 6 for bit6 mapped to PE6) Change Notice Flag for PortE

When CNCONEx = 1, CNFEx stores the occurrence of the CN event until cleared by the software.
When CNCONEx = 0, CNFEx Reads '0'.

| Value | Description |
|---|---|
| 1 | An Enabled Edge Event occurred on pin PORTEx |
| 0 | An Enabled Edge Event did not occur on pin PORTEx |


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.69. Slew Rate Control 0 for PortE

**Name:** SRCON0E
**Offset:** 0x4C0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with PortE.
**Note:** To configure the slew rate, the user must also configure the SRCON1E register associated with PortD. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR0Ex | SR0Ex |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1 – SR0Ex** (x = 0, 1; x = 0 for bit0 mapped to PC0, ... x = 1 for bit1 mapped to PC1) Slew Rate Control 0 for PortE

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.11.70. Slew Rate Control 1 for PortE

**Name:** SRCON1E
**Offset:** 0x4D0
**Reset:** 0x0
**Property:** -

This register configures the slew rate control bits associated with PortE.
**Note:** To configure the slew rate, the user must also configure the SRCON0D register associated with PortE. See *Slew Rate Control Bit Settings* table in the *Slew Rate Control* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SR1Ex | SR1Ex |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 0,1 – SR1Ex** (x = 0, 1; x = 0 for bit0 mapped to PC0, ... x = 1 for bit1 mapped to PC1) Slew Rate Control 1 for PortE

**Related Links**
Slew Rate Control


© 2025 Microchip Technology Inc. and its subsidiaries

## 6.12. Peripheral Pin Select (PPS) Input Mapping Register Summary

See the *PPS* module in the *Product Memory Mapping Overview* from Related Links for base address.

**Table 6-17.** Peripheral Pin Select Input Registers

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x000 | EXTINT0R | 7:0 | — | — | — | EXTINT0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x004 | EXTINT1R | 7:0 | — | — | — | EXTINT1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x008 | EXTINT2R | 7:0 | — | — | — | EXTINT2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x00C | EXTINT3R | 7:0 | — | — | — | EXTINT3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | | | | |
| 0x03C | NMIR | 7:0 | — | — | — | NMIR[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x040 | SCOM0P0R | 7:0 | — | — | — | SCOM0P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x044 | SCOM0P1R | 7:0 | — | — | — | SCOM0P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x048 | SCOM0P2R | 7:0 | — | — | — | SCOM0P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x04C | SCOM0P3R | 7:0 | — | — | — | SCOM0P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x050 | SCOM1P0R | 7:0 | — | — | — | SCOM1P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x054 | SCOM1P1R | 7:0 | — | — | — | SCOM1P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x058 | SCOM1P2R | 7:0 | — | — | — | SCOM1P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x05C | SCOM1P3R | 7:0 | — | — | — | SCOM1P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x060 | SCOM2P0R | 7:0 | — | — | — | SCOM2P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x064 | SCOM2P1R | 7:0 | — | — | — | SCOM2P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x068 | SCOM2P2R | 7:0 | — | — | — | SCOM2P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x06C | SCOM2P3R | 7:0 | — | — | — | SCOM2P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x070 | SCOM3P0R | 7:0 | — | — | — | SCOM3P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x074 | SCOM3P1R | 7:0 | — | — | — | SCOM3P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x078 | SCOM3P2R | 7:0 | — | — | — | SCOM3P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x07C | SCOM3P3R | 7:0 | — | — | — | SCOM3P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x080 | SCOM4P0R | 7:0 | — | — | — | SCOM4P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x084 | SCOM4P1R | 7:0 | — | — | — | SCOM4P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x088 | SCOM4P2R | 7:0 | — | — | — | SCOM4P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x08C | SCOM4P3R | 7:0 | — | — | — | SCOM4P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x090 | SCOM5P0R | 7:0 | — | — | — | SCOM5P0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x094 | SCOM5P1R | 7:0 | — | — | — | SCOM5P1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x098 | SCOM5P2R | 7:0 | — | — | — | SCOM5P2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x09C | SCOM5P3R | 7:0 | — | — | — | SCOM5P3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0A0 | QD0R | 7:0 | — | — | — | QD0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0A4 | QD1R | 7:0 | — | — | — | QD1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0A8 | QD2R | 7:0 | — | — | — | QD2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x0AC | QD3R | 7:0 | — | — | — | QD3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0B0 | REFIR | 7:0 | — | — | — | REFIR[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0B4 | CCLIN0R | 7:0 | — | — | — | CCLIN0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0B8 | CCLIN1R | 7:0 | — | — | — | CCLIN1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0BC | CCLIN2R | 7:0 | — | — | — | CCLIN2R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0C0 | CCLIN3R | 7:0 | — | — | — | CCLIN3R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0C4 | CCLIN4R | 7:0 | — | — | — | CCLIN4R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0C8 | CCLIN5R | 7:0 | — | — | — | CCLIN5R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0CC | TC0WO0R | 7:0 | — | — | — | TC0WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0D0 | TC0WO0R | 7:0 | — | — | — | TC0WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0D4 | TC0WO1R | 7:0 | — | — | — | TC0WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x0D8 | TC0WO1R | 7:0 | — | — | — | TC0WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0DC | TC1WO0R | 7:0 | — | — | — | TC1WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0E4 | TC1WO1R | 7:0 | — | — | — | TC1WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0EC | TC2WO0R | 7:0 | — | — | — | TC2WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0F0 | TC2WO0R | 7:0 | — | — | — | TC2WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0F4 | TC2WO1R | 7:0 | — | — | — | TC2WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0F8 | TC2WO1R | 7:0 | — | — | — | TC2WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x0FC | TC3WO0R | 7:0 | — | — | — | TC3WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x100 | TC3WO0R | 7:0 | — | — | — | TC3WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x104 | TC3WO1R | 7:0 | — | — | — | TC3WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x108 | TC3WO1R | 7:0 | — | — | — | TC3WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x10C | TC4WO0R | 7:0 | — | — | — | TC4WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x110 | TC4WO0R | 7:0 | — | — | — | TC4WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x114 | TC4WO1R | 7:0 | — | — | — | TC4WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x118 | TC4WO1R | 7:0 | — | — | — | TC4WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x11C | TC5WO0R | 7:0 | — | — | — | TC5WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x120 | TC5WO0R | 7:0 | — | — | — | TC5WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x124 | TC5WO1R | 7:0 | — | — | — | TC5WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x128 | TC5WO1R | 7:0 | — | — | — | TC5WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x12C | TC6WO0R | 7:0 | — | — | — | TC6WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x130 | TC6WO0R | 7:0 | — | — | — | TC6WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x134 | TC6WO1R | 7:0 | — | — | — | TC6WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x138 | TC6WO1R | 7:0 | — | — | — | TC6WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x13C | TC7WO0R | 7:0 | — | — | — | TC7WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x140 | TC7WO0R | 7:0 | — | — | — | TC7WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x144 | TC7WO1R | 7:0 | — | — | — | TC7WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x148 | TC7WO1R | 7:0 | — | — | — | TC7WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x14C | TC8WO0R | 7:0 | — | — | — | TC8WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x150 | TC8WO0R | 7:0 | — | — | — | TC8WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x154 | TC8WO1R | 7:0 | — | — | — | TC8WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x158 | TC8WO1R | 7:0 | — | — | — | TC8WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x15C | TC9WO0R | 7:0 | — | — | — | TC9WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x160 | TC9WO0R | 7:0 | — | — | — | TC9WO0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x164 | TC9WO1R | 7:0 | — | — | — | TC9WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x168 | TC9WO1R | 7:0 | — | — | — | TC9WO1R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x16C | QEIHOME0R | 7:0 | — | — | — | QEIHOME0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x170 | QEIHOME0R | 7:0 | — | — | — | QEIHOME0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x174 | QEIINDX0R | 7:0 | — | — | — | QEIINDX0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x178 | QEIINDX0R | 7:0 | — | — | — | QEIINDX0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x17C | QEIQEA0R | 7:0 | — | — | — | QEIQEA0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x180 | QEIQEA0R | 7:0 | — | — | — | QEIQEA0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x184 | QEIQEB0R | 7:0 | — | — | — | QEIQEB0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x188 | QEIQEB0R | 7:0 | — | — | — | QEIQEB0R[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x18C | ACCMPTENR | 7:0 | — | — | — | ACCMPTENR[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-17.** Peripheral Pin Select Input Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x190 | CAN1RXR | 7:0 | — | — | — | CAN1RXR[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x194 | COEXWLANACTR | 7:0 | — | — | — | COEXWLANACTR[4:0] | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |

**Related Links**

Product Memory Mapping Overview


© 2025 Microchip Technology Inc. and its subsidiaries

## 6.13. Peripheral Pin Select (PPS) Output Mapping Register Summary

See the *PPS* module in the *Product Memory Mapping Overview* from Related Links for base address.

**Table 6-18.** Peripheral Pin Select Output Registers

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x200 | RPA0G1R | 7:0 | — | — | RPA0G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x204 | RPA0G2R | 7:0 | — | — | RPA0G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x208 | RPA0G3R | 7:0 | — | — | RPA0G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x20C | RPA0G4R | 7:0 | — | — | RPA0G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | | | | |
| 0x210 | RPA1G2R | 7:0 | — | — | RPA1G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x214 | RPA1G3R | 7:0 | — | — | RPA1G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x218 | RPA1G4R | 7:0 | — | — | RPA1G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x21C | RPA1G5R | 7:0 | — | — | RPA1G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x220 | RRPA2G1R | 7:0 | — | — | RPA2G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x224 | RPA2G3R | 7:0 | — | — | RPA2G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x228 | RPA2G4R | 7:0 | — | — | RPA2G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x22C | RPA3G1R | 7:0 | — | — | RPA3G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x230 | RPA3G2R | 7:0 | — | — | RPA3G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x234 | RPA3G4R | 7:0 | — | — | RPA3G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x238 | RPA3G6R | 7:0 | — | — | RPA3G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x23C | RPA4G2R | 7:0 | — | — | RPA4G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x240 | RPA4G3R | 7:0 | — | — | RPA4G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x244 | RPA4G5R | 7:0 | — | — | RPA4G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x248 | RPA5G3R | 7:0 | — | — | RPA5G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x24C | RPA5G4R | 7:0 | — | — | RPA5G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x250 | RPA5G6R | 7:0 | — | — | RPA5G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x254 | RPA6G1R | 7:0 | — | — | RPA6G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x258 | RPA6G4R | 7:0 | — | — | RPA6G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x25C | RPA6G6R | 7:0 | — | — | RPA6G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x260 | RPA7G1R | 7:0 | — | — | RPA7G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x264 | RPA7G2R | 7:0 | — | — | RPA7G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x268 | RPA7G6R | 7:0 | — | — | RPA7G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x26C | RPA8G2R | 7:0 | — | — | RPA8G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x270 | RPA8G3R | 7:0 | — | — | RPA8G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x274 | RPA8G4R | 7:0 | — | — | RPA8G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x278 | RPA9G1R | 7:0 | — | — | RPA9G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x27C | RPA9G3R | 7:0 | — | — | RPA9G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x280 | RPA9G4R | 7:0 | — | — | RPA9G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x284 | RPA10G1R | 7:0 | — | — | RPA10G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x288 | RPA10G4R | 7:0 | — | — | RPA10G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x28C | RPA10G5R | 7:0 | — | — | RPA10G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2A0 | RPA13G3R | 7:0 | — | — | RPA13G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2A4 | RPA13G4R | 7:0 | — | — | RPA13G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2A8 | RPA14G1R | 7:0 | — | — | RPA14G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2AC | RPA14G4R | 7:0 | — | — | RPA14G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2B0 | RPA15G5R | 7:0 | — | — | RPA15G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2B4 | RPB0G1R | 7:0 | — | — | RPB0G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2B8 | RPB0G2R | 7:0 | — | — | RPB0G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x2BC | RPB0G5R | 7:0 | — | — | RPB0G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2C0 | RPB1G1R | 7:0 | — | — | RPB1G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2C4 | RPB1G2R | 7:0 | — | — | RPB1G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2C8 | RPB1G3R | 7:0 | — | — | RPB1G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2CC | RPB2G2R | 7:0 | — | — | RPB2G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2D0 | RPB2G3R | 7:0 | — | — | RPB2G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2D4 | RPB2G4R | 7:0 | — | — | RPB2G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2D8 | RPB3G1R | 7:0 | — | — | RPB3G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2DC | RPB3G3R | 7:0 | — | — | RPB3G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2EO | RPB3G4R | 7:0 | — | — | RPB3G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2E4 | RPB4G1R | 7:0 | — | — | RPB4G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x2E8 | RPB4G2R | 7:0 | — | — | RPB4G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2EC | RPB4G4R | 7:0 | — | — | RPB4G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2F0 | RPB4G5R | 7:0 | — | — | RPB4G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2F4 | RPB5G2R | 7:0 | — | — | RPB5G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2F8 | RPB5G3R | 7:0 | — | — | RPB5G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x2FC | RPB5G6R | 7:0 | — | — | RPB5G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x300 | RPB6G1R | 7:0 | — | — | RPB6G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x304 | RPB6G3R | 7:0 | — | — | RPB6G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x308 | RPB6G4R | 7:0 | — | — | RPB6G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x30C | RPB6G6R | 7:0 | — | — | RPB6G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x310 | RPB7G1R | 7:0 | — | — | RPB7G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x314 | RPB7G2R | 7:0 | — | — | RPB7G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x318 | RPB7G4R | 7:0 | — | — | RPB7G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x31C | RPB8G1R | 7:0 | — | — | RPB8G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x320 | RPB8G2R | 7:0 | — | — | RPB8G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x324 | RPB8G3R | 7:0 | — | — | RPB8G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x328 | RPB8G6R | 7:0 | — | — | RPB8G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x32C | RPB9G2R | 7:0 | — | — | RPB9G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x330 | RPB9G3R | 7:0 | — | — | RPB9G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x334 | RPB9G6R | 7:0 | — | — | RPB9G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x338 | RPB10G3R | 7:0 | — | — | RPB10G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x33C | RPB10G4R | 7:0 | — | — | RPB10G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x340 | RPB10G5R | 7:0 | — | — | RPB10G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x344 | RPB11G1R | 7:0 | — | — | RPB11G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x348 | RPB11G4R | 7:0 | — | — | RPB11G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x34C | RPB12G1R | 7:0 | — | — | RPB12G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x350 | RPB12G2R | 7:0 | — | — | RPB12G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x354 | RPB13G2R | 7:0 | — | — | RPB13G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x358 | RPB13G3R | 7:0 | — | — | RPB13G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x35C | RPB14G5R | 7:0 | — | — | RPB14G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x360 | RPB15G5R | 7:0 | — | — | RPB15G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x364 | RPC0G1R | 7:0 | — | — | RPC0G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x368 | RPC0G2R | 7:0 | — | — | RPC0G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x36C | RPC1G2R | 7:0 | — | — | RPC1G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x370 | RPC1G3R | 7:0 | — | — | RPC1G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x388 | RPC7G2R | 7:0 | — | — | RPC7G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x38C | RPC7G4R | 7:0 | — | — | RPC7G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x390 | RPC8G3R | 7:0 | — | — | RPC8G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x394 | RPC8G5R | 7:0 | — | — | RPC8G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x398 | RPC9G1R | 7:0 | — | — | RPC9G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x39C | RPC9G4R | 7:0 | — | — | RPC9G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3A0 | RPC10G5R | 7:0 | — | — | RPC10G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3A4 | RPC11G5R | 7:0 | — | — | RPC11G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3B8 | RPD0G5R | 7:0 | — | — | RPD0G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x3BC | RPD1G5R | 7:0 | — | — | RPD1G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3C0 | RPD2G3R | 7:0 | — | — | RPD2G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3C4 | RPD2G5R | 7:0 | — | — | RPD2G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3C8 | RPD3G1R | 7:0 | — | — | RPD3G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3CC | RPD3G4R | 7:0 | — | — | RPD3G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3D0 | RPD4G2R | 7:0 | — | — | RPD4G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3D4 | RPD4G5R | 7:0 | — | — | RPD4G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3D8 | RPD5G6R | 7:0 | — | — | RPD5G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3DC | RPD6G1R | 7:0 | — | — | RPD6G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3E0 | RPD6G3R | 7:0 | — | — | RPD6G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x3E4 | RPD7G2R | 7:0 | — | — | RPD7G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x3E8 | RPD7G4R | 7:0 | — | — | RPD7G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x40C | RPE0G1R | 7:0 | — | — | RPE0G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x410 | RPE0G3R | 7:0 | — | — | RPE0G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x414 | RPE1G2R | 7:0 | — | — | RPE1G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x418 | RPE1G4R | 7:0 | — | — | RPE1G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x41C | RPE2G3R | 7:0 | — | — | RPE2G3R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x420 | RPE2G5R | 7:0 | — | — | RPE2G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x424 | RPE3G4R | 7:0 | — | — | RPE3G4R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x428 | RPE4G1R | 7:0 | — | — | RPE4G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x42C | RPE4G5R | 7:0 | — | — | RPE4G5R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x430 | RPE5G6R | 7:0 | — | — | RPE5G6R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 6-18.** Peripheral Pin Select Output Registers (continued)

| Offset | Name | Bit Position | Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0x434 | RPE6G1R | 7:0 | — | — | RPE6G1R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |
| 0x438 | RPE6G2R | 7:0 | — | — | RPE6G2R[5:0] | | | | | |
| | | 15:8 | — | — | — | — | — | — | — | — |
| | | 23:16 | — | — | — | — | — | — | — | — |
| | | 31:24 | — | — | — | — | — | — | — | — |

**Related Links**

Product Memory Mapping Overview

## 6.14. 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 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.

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as ‘0’
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.14.1. Peripheral Pin Select Input Register

**Name:** *[pin name]*R
**Offset:** See the following Note
**Reset:** 0x00
**Property:** -

**Notes:**

1. For offset address, see *Peripheral Pin Select Input Registers* table in the *Peripheral Pin Select (PPS) Input Mapping Register Summary* from Related Links.
2. The user can only change the register values if the IOLOCK configuration bit (CFGCON0.IOLOCK) = 0.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | [pin name]R[4:0] | | | | |
| Access | | | | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
| Reset | | | | 0 | 0 | 0 | 0 | 0 |

**Bits 4:0 – *[pin name]*R[4:0]** Peripheral Pin Select Input bits
Where [*pin name*] refers to the pins that are used to configure peripheral input mapping. See *Input Pin Selection Group 1*, *Input Pin Selection Group 2*, *Input Pin Selection Group 3*, *Input Pin Selection Group 4* and *Input Pin Selection Group 5* tables in the *Input Mapping in PIC32CX-BZ6 Family of Devices* for input pin selection values from Related Links.
**Note:** This field is only writable when CFGCON0.IOLOCK = 0.

**Related Links**
Peripheral Pin Select (PPS) Input Mapping Register Summary
Input Mapping in PIC32CX-BZ6 Family of Devices


© 2025 Microchip Technology Inc. and its subsidiaries

### 6.14.2. Peripheral Pin Select Output Register

**Name:** RPnR
**Offset:** See the following Note
**Reset:** 0x0
**Property:** -

**Notes:**

1. For the offset address, see the *Peripheral Pin Select Output Registers* table in the *Peripheral Pin Select (PPS) Output Mapping Register Summary* from Related Links.
2. The user can only change the register values if the IOLOCK Configuration bit (CFGCON0.IOLOCK) = 0.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | RPnR[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 – RPnR[5:0]** Peripheral Pin Select Output Register
Output bits. For output pin selection values, see *Remappable Output Pin Configuration – Group 1*, *Remappable Output Pin Configuration – Group 2*, *Remappable Output Pin Configuration – Group 3*, *Remappable Output Pin Configuration – Group 4*, *Remappable Output Pin Configuration – Group 5* and *Remappable Output Pin Configuration – Group 6* tables in the *Output Mapping in PIC32CX-BZ6 Family of Devices* from Related Links.
**Note:** This field is only writable, when CFGCON0.IOLOCK = 0.

**Related Links**
Peripheral Pin Select (PPS) Output Mapping Register Summary
Output Mapping in PIC32CX-BZ6 Family of Devices

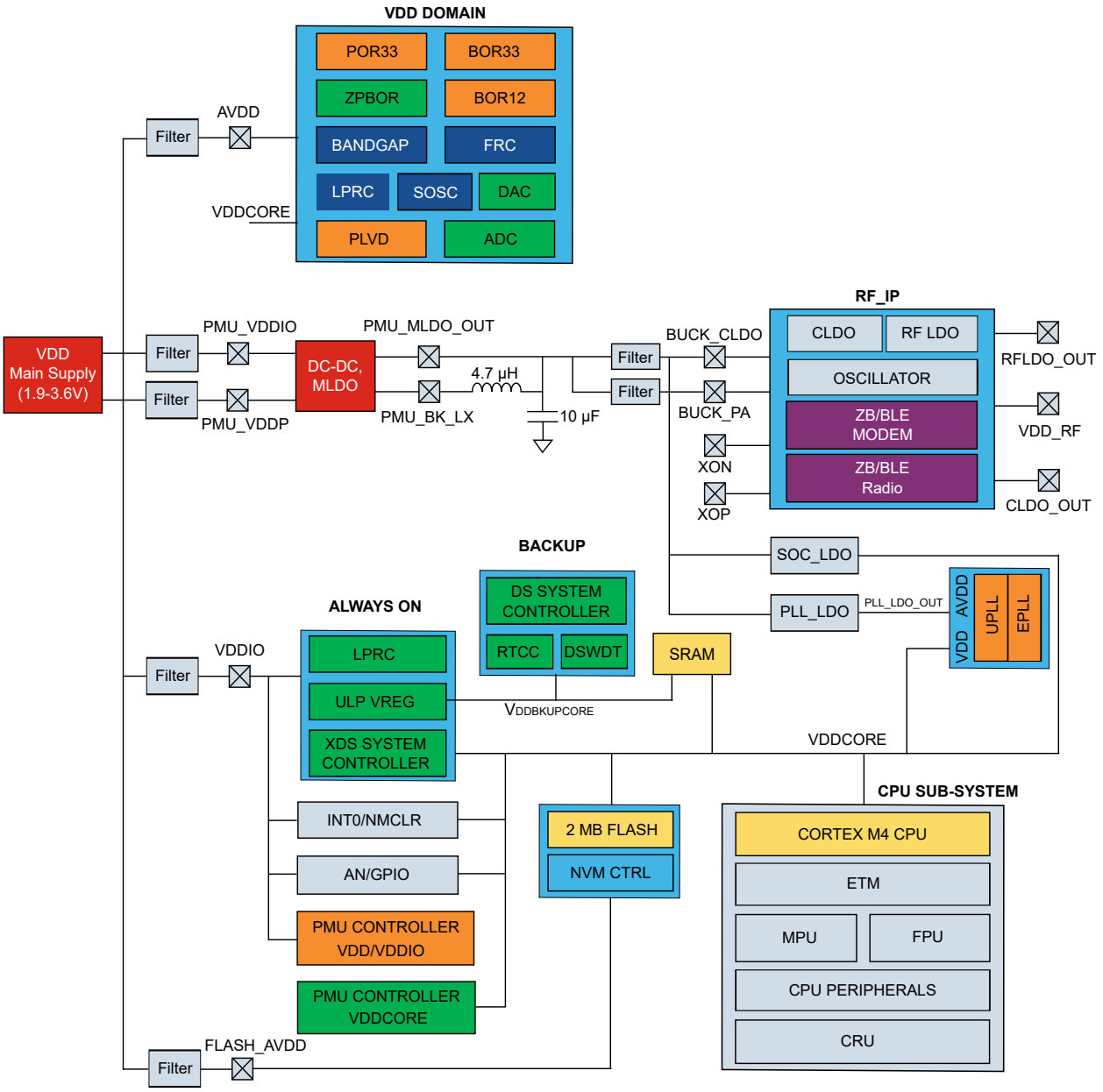

© 2025 Microchip Technology Inc. and its subsidiaries

# 7. Power Subsystem

## 7.1. Block Diagram

The following figure shows the detailed view of the power subsystem on the PIC32CX-BZ6 device.

**Figure 7-1.** Power Subsystem Block Diagram




The power domains of PIC32CX-BZ6 SoC are as follows:

- VDD – 1.9-3.6V, Main Supply powering VDDIO, FLASH_VDD, AVDD, PMU_VDDIO, PMU_VDDP and VUSB3V3. All other supplies are derived from VDD with or without filtering.
  - VDDIO
    - 1.9-3.6V, powering the Always ON (AON ), PMU Controller, AN/GPIO, INT0/NMCLR, BKUP


© 2025 Microchip Technology Inc. and its subsidiaries

  - FLASH_AVDD
    - 1.9V to 3.6V, filtered version of VDD (powering the Flash)
  - AVDD
    - 1.9V to 3.6V, filtered version of VDD for system analog functionality
  - PMU_VDDIO
    - 1.9-3.6V, filtered version of VDD (powering the PMU sub-system)
  - PMU_VDDP
    - 1.9-3.6V, filtered version of VDD (powering the PMU sub-system)
  - VUSB3V3
    - 3-3.6V, filtered version of VDD (powering USB transceiver supply)
- GND
  - Common GND for digital, analog and RF sub-systems

Other power supply pins as follows:

- CLDO_OUT, RF LDO OUT (1.2V ± 5%)
  - Output pin for the internal voltage regulator for decoupling, do not use this pin as an external power supply source
- CLDO_IN, VDD_RF
  - Loop back path for the CLDO and RF LDO power after external filtering
- BUCK_PA, BUCK_CLDO, PLL_VDD, SOC_LDO_VDD
  - 1.35V input for all the regulators in the SOC
  - Powered with 1.35V ± 5% from the combination of DC-DC, MLDO and external board filtering
- PMU_BK_LX
  - Pin for connecting the inductor for the internal switching regulator
- PMU_MLDO_OUT (1.35V ± 3.7%)
  - 1.35V PMU output pin. This is the shared output pin for both MLDO and the DC-DC converter
  - MLDO_OUT powers the internal LDO's in the Bluetooth/802.15.4 subsystem

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

## 7.2. VDD Voltage Domain Overview

The PIC32CX-BZ6 Power System VDD Integration Block (SIB) consists of the following modules:

- Power-on Reset (POR) – This module holds all the components in their inactive state until VDD reaches a stable operating voltage. POR ensures that the supply voltage is sufficient for proper operation of all other analog modules (PD_AVDD) in the Power SIB module.
- Bandgap (BG) – This module provides a stable reference voltage for Brown-out Reset, ADC, Flash, Comparators and low-voltage detect. The ADC, Flash and Comparators are outside the Power SIB module.
- Single core voltage regulator (SOC_LDO) based architecture fed with 1.35V from combination of DC-DC and MLDO and external board filtering.
- Feed the voltage regulator powering RFIP with 1.35V ± 5% from a combination of DC-DC and MLDO and external board filtering.
- CLDO has its own BOR to monitor power status.
- Brown-out Reset (BOR) – Use the BOR module to monitor the VDD supply voltage. This module provides a more accurate trip point but is only enabled when the POR event is inactive and the bandgap reference voltage is enabled and ready. Use this module to ensure that the supply voltage is above the minimum operating voltage needed for program memory reads to be valid.


© 2025 Microchip Technology Inc. and its subsidiaries

- Zero-power BOR (ZPBOR) – Use this low-power BOR during Deep Sleep operation. The user can only enable the ZPBOR when DSZPBOR configuration bit is '1'.
- Flash Low-Voltage Detect (LVD) uses PLVD.
- Master Clear Filter (MCLRF) – NMCLR generates a device Reset request based on the state of a device input pin. To minimize the effects of noise and to avoid unwanted Reset conditions, the MCLRF function filters the input pin to assure a specific pulse duration of the low input pulse.
  **Note:** Pulse width needs to be greater 400 ns to pass through MCLRF.
- Programmable Low-Voltage Detect of VDD (PLVD) consists of the following sub-modules:
  - LVD comparator
  - Resistor ladder
  - Analog voltage switch
  - LVD control ($V_{DDCORE\ DOMAIN}$)

## 7.3. $V_{DD-AON}$ Power Domain Overview

The PIC32CX-BZ6 $V_{DD-AON}$ power domain block consists of the following modules:

- SOSC's analog component acts as a secondary oscillator and uses a low-power 32.768 kHz crystal oscillator for accurate time keeping.
- For context saving, use the Extreme Deep Sleep (XDS) system controller with the semaphore.
- The Deep Sleep Regulator (ULPVREG) is an ultra-low-power regulator that provides power during deep sleep modes and/or retention power to the rest of the system in various modes of operation. $V_{DDBKUPCORE}$ is the voltage output.
- Low Power RC oscillator (LPRC) operates at a nominal frequency of 32.768 KHz.

## 7.4. $V_{DDBKUPCORE}$ Power Domain

The PIC32CX-BZ6 $V_{DDBKUPCORE}$ domain consists of the following modules:

- Real Time Clock Calender (RTCC)
- Deep Sleep WDT (DSWDT)
- Deep Sleep System Controller (DSCTRL)
- 32.768 kHz Oscillator Controller (digital) (SOSC_DIG)

## 7.5. PMU Controller

The PMU controller acts as a control unit to monitor/program the BUCK/MLDO and provides unified control to various LDOs present on the PIC32CX-BZ6 device.

## 7.6. Voltage Regulators

The following voltage regulators are available in power subsystem:

- VREG (DC-DC/MLDO):
  - MLDO mode – Linear voltage regulator generating 1.35V for powering CLDO and RF LDO; operates from 1.9-3.6V
  - DC-DC – Switching voltage regulator generating 1.35V; operates from 2.3-3.6V
- ULP_VREG – Ultra-low-power voltage regulator for operation in back-up mode
- CLDO, RF LDO – Powering the different blocks of the RF subsystem
- SOC_LDO – Powering the VDDCORE of the PIC32CX-BZ6
- PLL_LDO – Supplies 1.2V analog for EPLL(Ethernet PLL) and UPLL (USB PLL)

## 7.7. Power Supply Modes

The PIC32CX-BZ6 device supports a single power supply from 1.9-3.6V. The IO supply cannot be decoupled from the main supply. The same voltage must be applied to VDDx, PMU_VDDIO,


© 2025 Microchip Technology Inc. and its subsidiaries

VPMU_VDDC and AVDD with different levels of filtering. The system has the following different modes:

- RUN mode – The PIC32CX-BZ6 device automatically gets into RUN mode upon power-up. This is the default state of the device.
  - MLDO mode – Provide a soft start-up by using the MLDO, limiting the charging current. The MLDO also helps to extend the supply voltage range down to 1.9V below Buck BOR. This mode does not require external inductor.
  - BUCK mode – The most efficient mode when the CPU and peripherals are running. In this mode, the DC-DC converter powers the SoC. This mode requires external LC-filtering and appropriate decoupling on-board before supplying power to other blocks.
    BUCK mode has the following two operating modes:
    - PWM (Pulse Width Modulation) mode – Buck can deliver the highest output current with good efficiency. The internal switching clock in this mode is 1-2 MHz. In PWM mode, it is expected to reach an efficiency of 85%.
    - PSM (Pulse Skipping Mode) – This mode is recommended when the load current demand is low. The PSM mode is a type of frequency modulation scheme with an efficiency of up to 80%.
- Low-power modes – The PIC32CX-BZ6 supports various low-power modes; Sleep, Deep-Sleep (DS) and Extreme Deep Sleep (XDS). See *Power Management Unit (PMU)* from Related Links for more details on how to transition from RUN mode to low-power modes.
- IDLE mode – See *Power Management Unit (PMU)* from Related Links.

The on-the-fly software can do the selection between Switching (BUCK) mode and Linear (MLDO) mode. Be sure to design the power supply according to the type of mode in use.

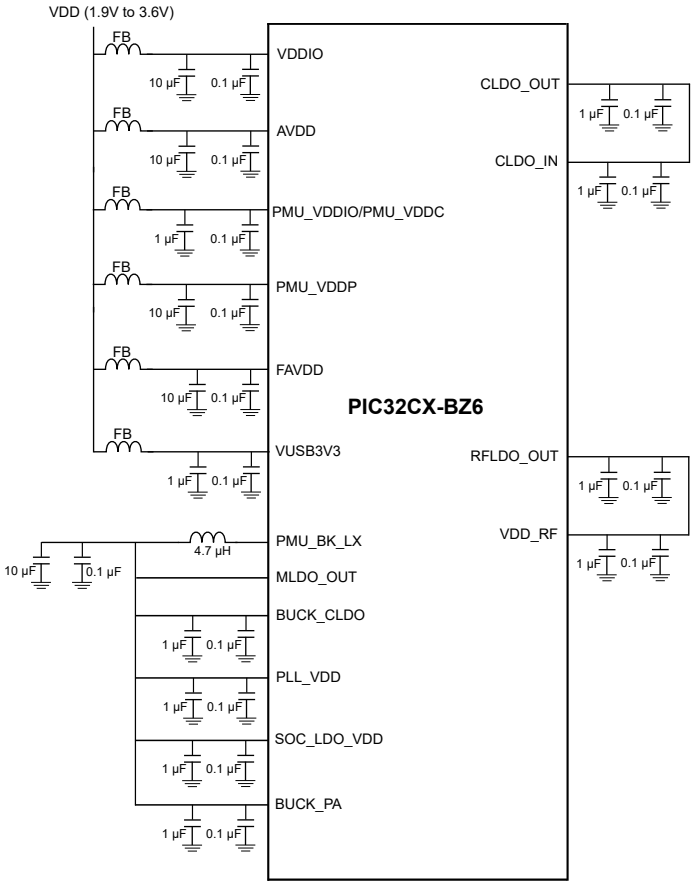
**Related Links**

Power Management Unit (PMU)


© 2025 Microchip Technology Inc. and its subsidiaries

## 7.8. Typical Power Supply Connection for SoC

**Figure 7-2.** Typical Power Supply Connection for SoC

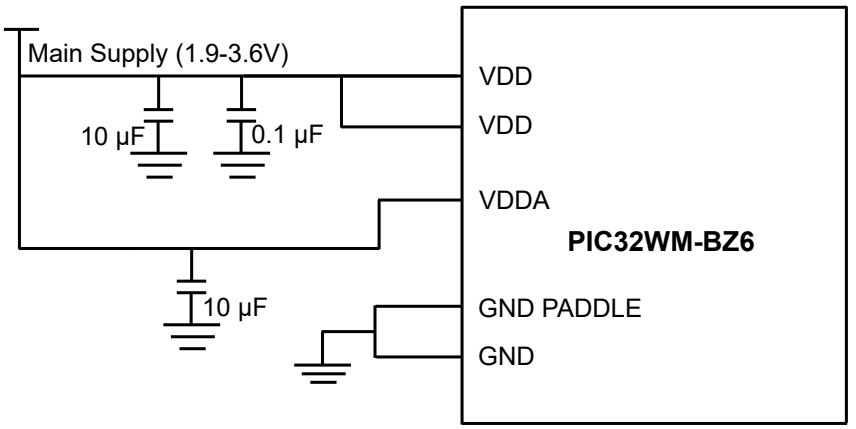



## 7.9. Typical Power Supply Connection for PIC32WM-BZ6 Module

The PIC32WM-BZ6 modules require only a single power supply on the VDD pins of the module.

- VDD ranges from 1.9-3.6V for non-ECC PIC32WM-BZ6 modules.

**Figure 7-3.** PIC32WM-BZ6 Module Schematics with VDD and Optional Bulk Capacitors





© 2025 Microchip Technology Inc. and its subsidiaries

## 7.10. Power-Up Sequence

Characteristics of power-up sequence are as follows:

- The VDD/AVDD domains must rise at the same time.
- At Power-on Reset, the PIC32CX-BZ6 operates in the MLDO mode.
- The LDOs start with their default settings and VDDCORE is powered-up.
- The RF block is maintained in the Sleep mode during the power-up time.
- The PMU controller switches the MLDO mode to DC-DC mode based on device settings in the Flash BCFG/TCFG area.

### 7.10.1. Starting of Voltage Regulators

The characteristics of power-up voltage regulators are as follows:

- On power-up, the internal regulator starts in MLDO mode.
- After MLDO boots up, the CLDO, SoC LDO and PLL LDO start gets initialized.
- Start the code execution.
- The RF system is maintained in Sleep mode during the power-up time.

### 7.10.2. Starting-up of Crystals

The characteristics of power-up crystals are as follows:

- The power-up of the SoC happens with the internal oscillators.
- After the power-up, the boot ROM code will either run on internal FRC oscillator only or turn on POSC+PLL and switch to 128 MHz, depending on seccfg[1:0]. After execution of boot ROM code, the user software can request to switch on the SOSC and the POSC crystals, if necessary.

### 7.10.3. BOR and POR

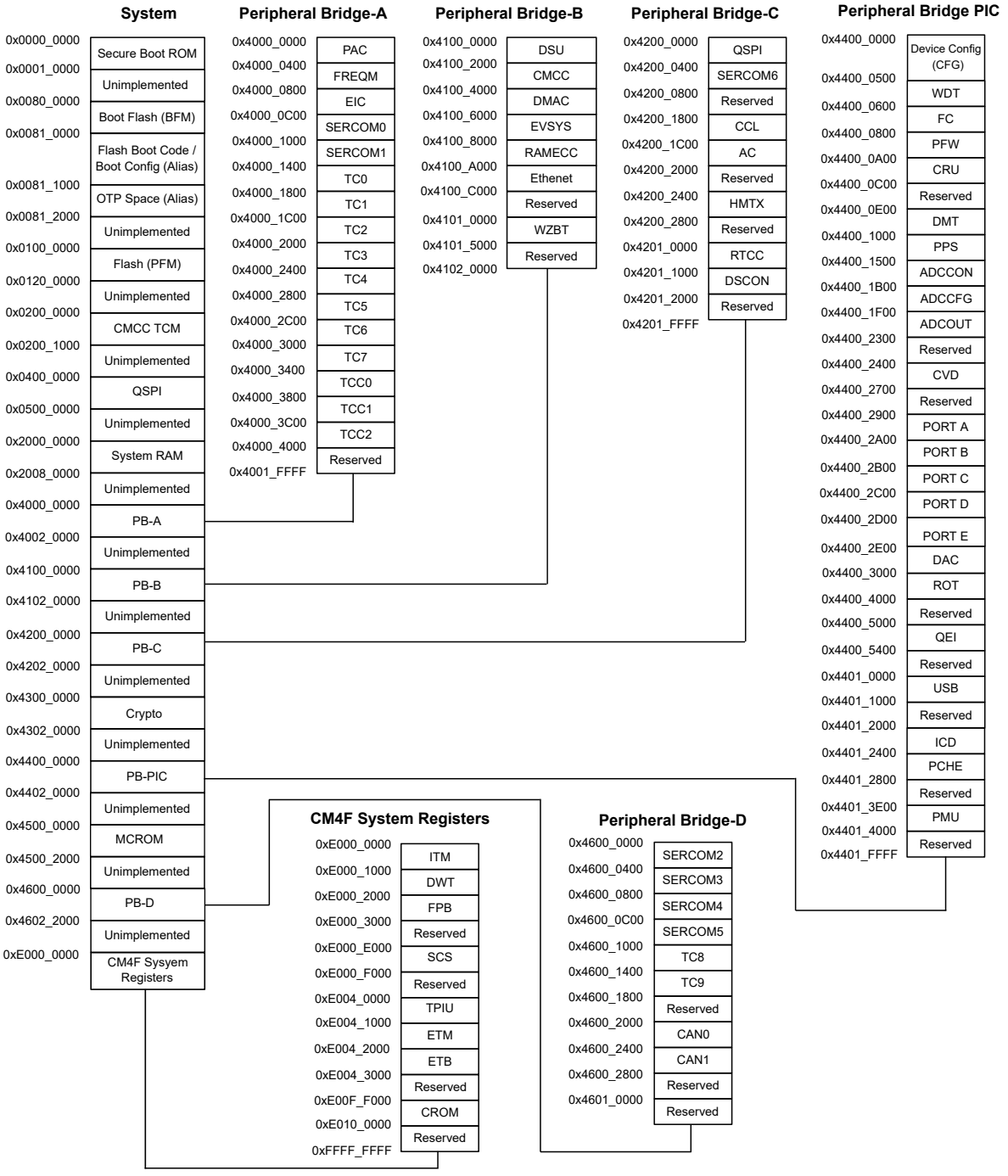
The Brown-out Reset (BOR) monitors the VDD supply voltage. On detection of a brown-out condition, the BOR re-arms the POR. In this device, the minimum BOR trip point is the voltage below which I/O is considered un-trusted; thus, it generates a Reset. There are three classes of BOR in the PIC32CX-BZ6 and the PIC32WM-BZ6 Module:

- BOR3.3 to monitor VDD
- BOR1.2 to monitor VDDCORE and CLDO
- ZPBOR monitors VDD during the Deep Sleep and Extreme Deep Sleep mode if enabled


© 2025 Microchip Technology Inc. and its subsidiaries

# 8. Product Memory Mapping Overview

**Figure 8-1.** Product Mapping





© 2025 Microchip Technology Inc. and its subsidiaries

**Notes:**

- Access attempts to any unimplemented memory location generate a bus error.
- CFGCON1.QSCHE_EN controls the QSPI (XIP) space “cacheable and bufferable” attribute.
- The MCROM is the microcode crypto ROM associated with the BA457. Only the BA457 has read permissions to MCROM.
- The DAP derives the base address of the components from CoreSight ROM (CROM) entry values.
- Component base address = CROM base address + CROM entry value
- For more details on each component register space, refer to the *Cortex-M4 Technical Reference Manual r0p0* (CM4F) documentation.

## 8.1. Embedded Memories

- Internal ROM for secure boot
- Internal high-speed Flash
- Internal high-speed RAM with retention capability in the low power modes
- eFuse One-Time-Programmable memory secure boot key storage

## 8.2. 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.

**Table 8-1.** Physical Memory Map

| Memory | Start Address | Size |
|---|---|---|
| | | PIC32CX2051BZ6/PIC32WM-BZ6 |
| Boot ROM | 0x00000000 | 64 KB |
| Boot Flash | 0x00800000 | 64 KB |
| Embedded Program Flash | 0x01000000 | 2048 KB |
| Embedded SRAM | 0x20000000 | 512 KB |
| Peripheral Bridge A | 0x40000000 | 128 KB |
| Peripheral Bridge B | 0x41000000 | |
| Peripheral Bridge C | 0x42000000 | |
| Peripheral Bridge PIC | 0x44000000 | |
| Peripheral Bridge D | 0x46000000 | |
| eFuse | — | 3072 bits |

## 8.3. Boot ROM

A 64-KB ROM is dedicated for the secure boot firmware as a part of Root of Trust (RoT) macro. On a POR, secure boot firmware, which actually authenticates the rest of the program image in the Flash, is always run. The RoT macro stores the keys and credentials required for code authentication that are a part of the eFuses.

## 8.4. Flash Memory Parameters

A single page contains 4K bytes, which is applicable to all the device part numbers listed in the configuration summary chapter (see *Configuration Summary* from Related Links).

The number of pages available in a device part number will vary depending on the available maximum Flash memory size.

**Equation 8-1.** Calculating Flash Memory

$$Number of Pages = \frac{Flash Size(Bytes)}{Page Size(Bytes)}$$


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**
Configuration Summary

## 8.5. eFuse Memory

An eFuse is OTP memory that exists as a part of the RoT macro to facilitate key and other required credential storage needed by secure boot.

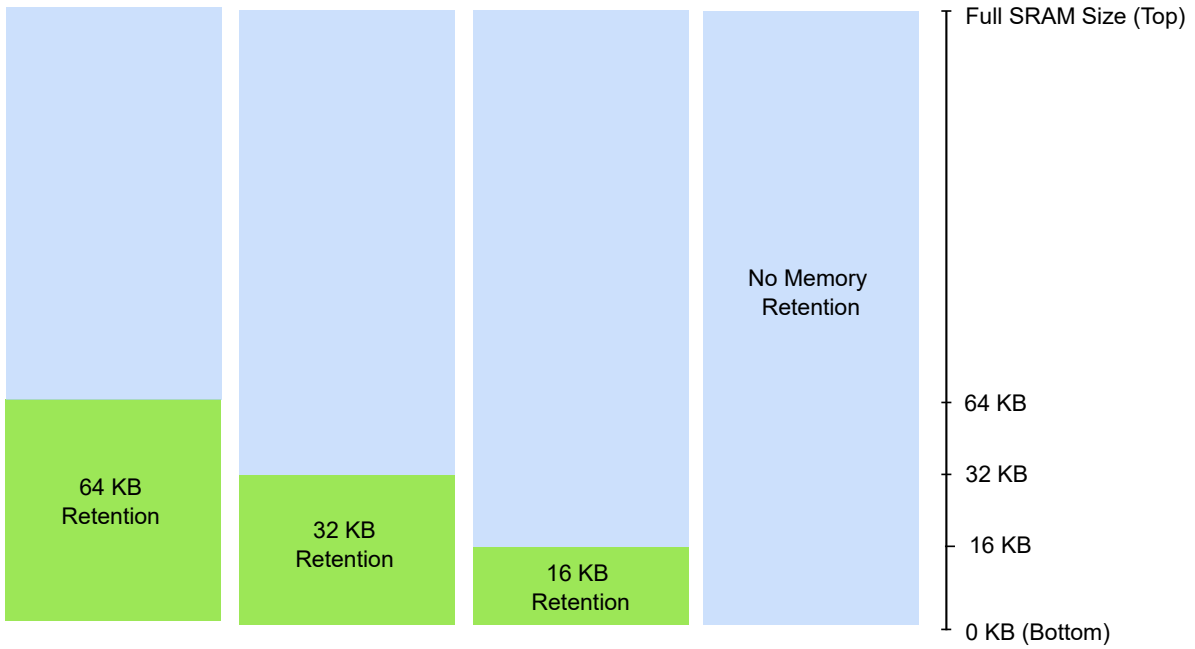
## 8.6. SRAM Memory Configuration

### Retention

Depending on the application and power budget needs, part of the system memory can be retained in the Deep Sleep mode. The amount of the SRAM retained in this mode is software selectable, by writing the WCMSIZ register in the PMU module, up to 64 KB of SRAM.

By default, no retention is selected.

**Figure 8-2.** Retention Options




### RAM Error Correction

For safety applications, the PIC32CX-BZ6 family embeds Error Correction Codes (ECC) to detect and correct single bit errors or to enable dual error detection for the system memory. The ECC is software selectable through the CFGCON0.FRECCDIS bit in the boot Flash device configuration. By default, ECC is disabled.

ECC can be applied only for 256 KB of SRAM. When enabled, half the memory will be reserved to store the ECC, and will not be available for the application.

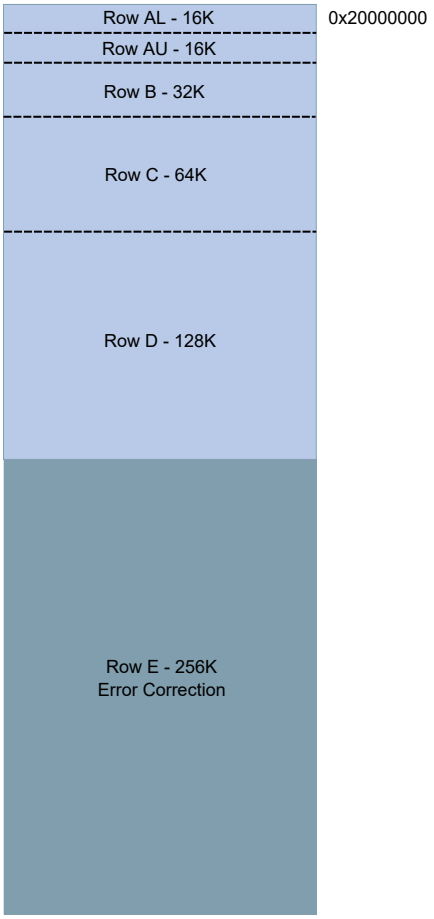
Therefore, when ECC is enabled, usable System RAM is 256KB (512KB/2) for the main 512K data RAM variant. Row E is used as ECC memory, if enabled. ECC support for row AL/AU/B/C/D can be selected using the WCMSIZ.SRAM1_SIZ bits and CFGCON0.FRECCDIS. bit.

ECC (Error Correcting Code) is activated when both WCMSIZ.SRAM1SIZ bits are set to '0' and the CFGCON0.FRECCDIS bit is '0'. This configuration does not enable Embedded Trace Buffer (ETB) Coresight, but it does support memory rows AL/AU/B/C/D. It is important to note that enabling ECC is independent of the ETB Coresight status. ECC can be utilized across these rows provided that the system has 224 KB of available memory and the 32 kb ETB Coresight feature is enabled. However,


© 2025 Microchip Technology Inc. and its subsidiaries

ECC should only be enabled if the application is not using the Memory Retention feature during Deep Sleep, which corresponds to WCMSIZ.SRAM1SIZ being '00'. This ensures that ECC functionality does not interfere with memory retention requirements

**Figure 8-3.** Memory with RAM Error Correction




**Note:** ECC is not possible if the user enables SRAM retention.

### CoreSight ETB Connection

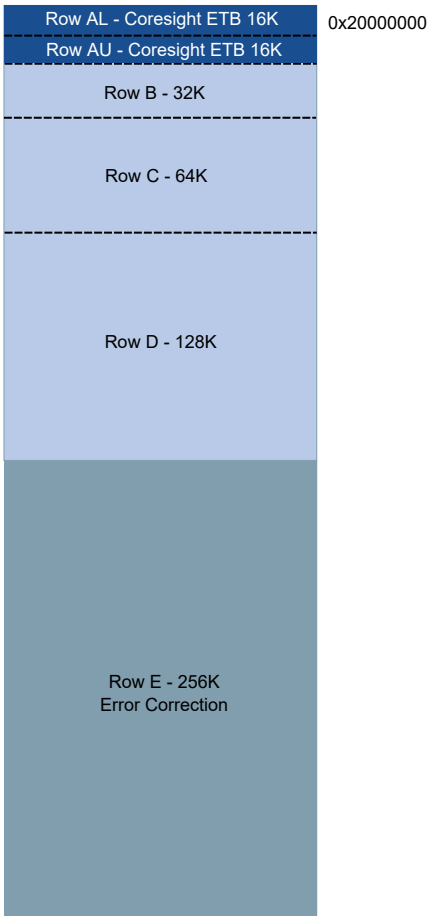
When CoreSight ETB is enabled, a 32 KB system memory space is reserved for CoreSight (Embedded Trace Buffer) ETB debug usage. Therefore, when CoreSight ETB is enabled,

- For 512 KB data RAM variants:
  Usable system RAM is 480 KB (512-32 KB)
- For 256 KB data RAM variants:
  Usable system RAM is 224 KB (256-32 KB).

The following figure illustrates an example where both ECC and CoreSight ETB are enabled.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 8-4.** Memory with ECC and CoreSight ETB




## 8.7. Boot Flash Device Configuration Word

The PIC32CX-BZ6 device provides several user-writable configuration registers related to the configuration and operation of the system. The device configuration words are programmed in boot Flash memory (NVR pages) and get loaded on equivalent registers after the device Reset. The following table shows the device configuration words in Boot Flash.

**Table 8-2.** Boot Flash and Device Configuration Word

| Physical Address | Register Name | Bit Range | 31:0 |
|---|---|---|---|
| 0x00810E88 | ALTFUSERID | 31:0 | See *USER_ID* in the *CFG Register Summary* from Related Links |
| 0x00810E8C | ALTDEVCFG4 | 31:0 | See *CFGCON4(L)* in the *CFG Register Summary* from Related Links |
| 0x00810E90 | ALTDEVCFG2 | 31:0 | See *CFGCON2(L)* in the *CFG Register Summary* from Related Links |
| 0x00810E94 | ALTDEVCFG1 | 31:0 | See *CFGCON1(L)* in the *CFG Register Summary* from Related Links |
| 0x00810E98 | ALTDEVCFG0 | 31:0 | See *CFGCON0(L)* in the *CFG Register Summary* from Related Links |
| 0x00810E9C | ALTFBCFG0 | 31:0 | See *BCFG0* in the *CFG Register Summary* from Related Links |
| 0x00810F88 | FUSERID | 31:0 | See *USER_ID* in the *CFG Register Summary* from Related Links |
| 0x00810F8C | DEVCFG4 | 31:0 | See *CFGCON4(L)* in the *CFG Register Summary* from Related Links |
| 0x00810F90 | DEVCFG2 | 31:0 | See *CFGCON2(L)* in the *CFG Register Summary* from Related Links |
| 0x00810F94 | DEVCFG1 | 31:0 | See *CFGCON1(L)* in the *CFG Register Summary* from Related Links |
| 0x00810F98 | DEVCFG0 | 31:0 | See *CFGCON0(L)* in the *CFG Register Summary* from Related Links |
| 0x00810F9C | FBCFG0 | 31:0 | See *BCFG0* in the *CFG Register Summary* from Related Links |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 8-2.** Boot Flash and Device Configuration Word (continued)

| Physical Address | Register Name | Bit Range | 31:0 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00810FBC | FCPN0 | 7:0 | — | | | | | | | |
| | | 15:8 | — | | | | | | | |
| | | 23:16 | — | | | | | | | |
| | | 31:24 | — | — | — | CP | — | — | — | — |

**Related Links**

Register Summary


© 2025 Microchip Technology Inc. and its subsidiaries

## 8.8. Boot Flash Code Protection Register

**Name:** FCPN0
**Offset:** 0x00805FBC
**Reset:** 0x00000000
**Property:** —

**Note:** Offset is an absolute address of this register.

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

| 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 28 – CP** Flash (BFM, PFM) Code Protect

**Note:** The value of this bit is the inverse polarity of the value read from the BCFG0 register.

| Value | Description |
|---|---|
| 0 | Protection is enabled |
| 1 | Protection is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

# 9. Processor and Architecture

## 9.1. Cortex-M4 Processor

The PIC32CX-BZ6 implements the ARM Cortex-M4 processor is High performance 32-bit processor with security and secure boot feature. It offers the following significant benefits to developers:

- Outstanding processing performance combined with fast interrupt handling
- Enhanced system debug with extensive breakpoint and trace capabilities
- Efficient processor core, system and memories
- Ultra-low-power consumption with integrated sleep modes
- Platform security robustness with integrated Memory Protection Unit (MPU)

The implemented ARM Cortex-M4 is revision r0p1.

For additional information, refer to http://www.arm.com.

The Cortex-M4 processor is built on a high-performance processor core with a 3-stage pipeline Harvard architecture; making it ideal for demanding embedded applications. The processor delivers exceptional power efficiency through an efficient instruction set and extensively optimized design, providing high-end processing hardware including IEEE 754-compliant single-precision floating-point computation, a range of single-cycle and SIMD multiplication and multiply-with-accumulate capabilities, saturating arithmetic and dedicated hardware division.

To facilitate the design of cost-sensitive devices, the Cortex-M4 processor implements tightly-coupled system components that reduce processor area while significantly improving interrupt handling and system debug capabilities. The Cortex-M4 processor implements a version of the Thumb instruction set based on Thumb-2 technology, ensuring high code density and reduced program memory requirements. The Cortex-M4 instruction set provides the exceptional performance expected of a modern 32-bit architecture, with the high code density of 8-bit and 16-bit microcontrollers.

The Cortex-M4 processor closely integrates a configurable Nested Vector Interrupt Controller (NVIC) to deliver industry-leading interrupt performance. The NVIC includes a Non-Maskable interrupt (NMI) and provides up to 8 interrupt priority levels. The tight integration of the processor core and NVIC provides fast execution of Interrupt Service Routines (ISRs), dramatically reducing the interrupt latency. This is achieved through the hardware stacking of registers and the ability to suspend load-multiple and store-multiple operations. Interrupt handlers do not require wrapping in assembler code, removing any code overhead from the ISRs. A tail-chain optimization also significantly reduces the overhead when switching from one ISR to another.

To optimize low-power designs, the NVIC integrates with the sleep modes that include a deep sleep function that enables the entire device to be rapidly powered down while still retaining program state.

### 9.1.1. System Level Interface

The Cortex-M4 processor provides multiple interfaces using AMBA technology to provide high-speed, low-latency memory accesses. It supports unaligned data accesses and implements atomic bit manipulation that enables faster peripheral controls, system spinlocks and thread-safe Boolean data handling.

The Cortex-M4 processor has an MPU that provides:

- Fine grain memory control
- Enabling applications to utilize multiple privilege levels
- Separating and protecting code, data and stack on a task-by-task basis


© 2025 Microchip Technology Inc. and its subsidiaries

These requirements are becoming critical in many embedded applications, such as automotive sector.

### 9.1.2. Integrated Configurable Debug

The Cortex-M4 processor implements a complete hardware debug solution. This provides high system visibility of the processor and memory through a 2-pin Serial Wire Debug (SWD) port that is ideal for microcontrollers and other small package devices.

For system trace, the processor integrates an Instrumentation Trace Macrocell (ITM) alongside data watchpoints and a profiling unit. The Embedded Trace Macrocell (ETM) delivers unrivaled instruction trace capture in an area far smaller than traditional trace units, enabling many low cost MCUs to implement full instruction trace for the first time.

To enable simple and cost-effective profiling of the system events these generate, a stream of software-generated messages, data trace and profiling information is exported over three different ways:

- Output off chip using the TPIU, through a single pin, called Serial Wire Viewer (SWV). Limited to ITM system trace
- Output off chip using the TPIU, through a 4-bit pin interface. Bandwidth is limited.
- Internally stored in RAM, using the CoreSight ETB. Bandwidth is, then, optimal but capacity is limited.

The Flash Patch and Breakpoint Unit (FPB) provide up to eight hardware breakpoint comparators that debuggers can use. The comparators in the FPB also provide remap functions of up to eight words in the program code in the code memory region. This enables applications stored on a non-erasable, ROM-based microcontroller to be patched if a small programmable memory, for example, Flash, is available in the device. During initialization, the application in ROM detects, from the programmable memory, whether there is a requirement for a patch or not. If there is a requirement for a patch, the application programs the FPB to remap a number of addresses. When those addresses are accessed, the accesses are redirected to a remap table specified in the FPB configuration, which means the program in the non-modifiable ROM can be patched.

### 9.1.3. Cortex-M4 Processor Features and Configuration

- The Thumb instruction set combines high code density with 32-bit performance
- IEEE 754-compliant single-precision Floating Point Unit (FPU)
- Integrated sleep modes for low power consumption
- Fast code execution permits slower processor clock or increases Sleep mode time
- Hardware division and fast digital-signal-processing orientated multiply accumulate
- Saturating arithmetic for signal processing
- Deterministic, high-performance interrupt handling for time-critical applications
- Memory Protection Unit (MPU) for safety-critical applications
- Extensive debug and trace capabilities: Serial wire debug and serial wire trace reduce the number of pins required for debugging, tracing and code profiling.

**Table 9-1.** Cortex-M4 Processor Features and Configuration

| Features | PIC32CX-BZ6 Configuration |
|---|---|
| Interrupts | 57 |
| Number of priority bits | 3 = eight levels of priority |
| Data endianness | Little-endian |
| SysTick Timer calibration value | 0x80000000 |
| MPU | Present |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-1.** Cortex-M4 Processor Features and Configuration (continued)

| Features | PIC32CX-BZ6 Configuration |
|---|---|
| Debug support level | 3 = Full debug plus DWT data matching |
| Trace support level | 2 = Full trace. Standard trace plus ETM |
| JTAG | Not present |
| Bit Banding | Not present |
| FPU | Present |

### 9.1.4. Cortex-M4 Core Peripherals

**Table 9-2.** Cortex-M4 Core Peripherals

| Cortex-M4 Core Peripherals | Description |
|---|---|
| Nested Vectored Interrupt Controller | The Nested Vector Interrupt Controller is an embedded interrupt controller that supports low latency interrupt processing. |
| System Control Block | The System Control Block (SCB) is the programmer's model interface to the processor. It provides system implementation information and system control, including configuration, control and reporting of system exceptions. For more details, refer to the *Cortex-M4 Technical Reference Manual*. |
| System Timer | The System Timer (SysTick) is a 24-bit countdown timer. Use this as a Real-Time Operating System (RTOS) tick timer or as a simple counter. The SysTick timer runs on the processor clock and it does not decrement when the processor is halted for debugging. For more details, refer to the *Cortex-M4 Technical Reference Manual*. |
| Memory Protection Unit | The Memory Protection Unit (MPU) improves system reliability by defining the memory attributes for different memory regions. It provides up to eight different regions and an optional predefined background region. For more details, refer to the *Cortex-M4 Technical Reference Manual*. |
| Floating-Point Unit | The Floating Point Unit (FPU) provides IEEE 754-compliant operations on single-precision, 32-bit, floating-point values. For more details, refer to the *Cortex-M4 Technical Reference Manual*. |

### 9.1.5. Cortex-M4 Address Map

**Table 9-3.** Cortex-M4 Address Map

| Address | Core Peripheral |
|---|---|
| 0xE000E000-0xE000E00F | System Control Space (SCS) |
| 0xE000E010-0xE000E01F | System timer |
| 0xE000E100-0xE000E4EF | Nested vectored interrupt controller |
| 0xE000ED00-0xE000ED3F | System control block |
| 0xE000ED90-0xE000ED93 | MPU type register |
| 0xE000ED94-0xE000EDBB | Memory protection unit |
| 0xE000EF00-0xE000EF03 | Nested vectored interrupt controller |
| 0xE000EF34-0xE000EF47 | Floating point unit |


© 2025 Microchip Technology Inc. and its subsidiaries

## 9.2. Nested Vector Interrupt Controller (NVIC)

### 9.2.1. Overview

The Nested Vectored Interrupt Controller (NVIC) in the PIC32CX-BZ6 family devices supports 57 interrupts with eight different priority levels. For more details, refer to the *Cortex-M4 Technical Reference Manual* (www.arm.com).

### 9.2.2. Interrupt Line Mapping

The following table provides details about each of the interrupt lines that is connected to one peripheral instance. Each peripheral can have one or more interrupt flags, located in the peripheral's Interrupt Flag Status and Clear (INTFLAG) register.

An interrupt flag is set when the interrupt condition occurs. Each interrupt in the peripheral can be individually enabled by configuring it in the peripheral's Interrupt Enable register.

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

Depending on their criticality, the interrupt requests for one peripheral are either ORed together on system level, generating one interrupt, or directly connected to NVIC interrupt lines (see the following table).

An interrupt request sets 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 9-4.** NVIC Interrupt Mapping

| Module | Source | NVIC Line | Remark |
|---|---|---|---|
| CRU Subsystem | NMI | NMI | Non-Maskable interrupt |
| RTC | PER | 0 | Prescalar |
| | CMP | | Compare |
| | TAMPER | | Tamper |
| | OVF | | Overflow |
| EIC | EXTINT | 1 | External interrupt |
| FREQM | DONE | 2 | Measurement done |
| Flash Subsystem | Flash Controller | 3 | Flash controller |
| | PFW | | Program Flash Write |
| | PCACHE | | Page Cache |
| PORT-A | PortA Input Change Interrupt | 4 | PortA input change interrupt |
| PORT-B | PortB Input Change Interrupt | 5 | PortB input change interrupt |
| PORT-C | PortC Input Change Interrupt | 6 | PortC input change interrupt |
| PORT-D | PortD Input Change Interrupt | 7 | PortD input change interrupt |
| PORT-E | PortE Input Change Interrupt | 8 | PortE input change interrupt |
| DMAC | SUSP 0..3 | 9 | Channel suspend |
| | TCMPL 0..3 | | Transfer complete |
| | TERR 0..3 | | Transfer error |
| | SUSP 4..15 | 10 | Channel suspend |
| | TCMPL 4..15 | | Transfer complete |
| | TERR 4..15 | | Transfer error |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-4.** NVIC Interrupt Mapping (continued)

| Module | Source | NVIC Line | Remark |
|---|---|---|---|
| EVSYS | EVD 0..3 | 11 | Event Detected Channel n interrupt |
| | OVR 0..3 | | Overrun Channel n interrupt |
| | EVD 4..11 | 12 | Error |
| | OVR 4..11 | | Overrun Channel n interrupt |
| PAC | ERR | 13 | Error |
| RAMECC | SINGLEE-0 | 14 | Single bit error |
| | DUALE-1 | | Dual bit error |
| SERCOM0<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 15 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| SERCOM1<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 16 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| SERCOM2<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 17 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| SERCOM3<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 18 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-4.** NVIC Interrupt Mapping (continued)

| Module | Source | NVIC Line | Remark |
|---|---|---|---|
| SERCOM4<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 19 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| SERCOM5<br>Order: USART, I2CM, I2CS, SPI | DRE, MB, PERC, DRE | 20 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| SERCOM6<br>Order: I2CM, I2CS | MB, PERC | 21 | Data Register Empty, Host on Bus, Stop Received |
| | TXC, SB, AMATCH, TXC | | Transmit Complete, Device on Bus, Address Match |
| | RXC, -, DRDY, RXC | | Receive Complete, Data Ready |
| | RXS, -, -, SSL | | Receive Start, Device Select Low |
| | CTSIC, -, -, - | | Clear to Send Input Change |
| | RXBRK, -, -, - | | Receive Break |
| | ERR, ERR, ERR, ERR | | Error |
| TCC0 | CNT | 22 | Count |
| | DFS | | Debug Fault State |
| | ERR | | Capture Overflow Error |
| | FAULTA | | Recoverable Fault A |
| | FAULTB | | Recoverable Fault B |
| | FAULT0 | | Non-Recoverable Fault 0 |
| | FAULT1 | | Non-Recoverable Fault 1 |
| | OVF | | Overflow/Underflow |
| | TRG | | Retrigger event |
| | UFS | | Non-Recoverable Update Fault |
| | MC | | Match or Capture |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-4.** NVIC Interrupt Mapping (continued)

| Module | Source | NVIC Line | Remark |
|---|---|---|---|
| TCC1 | CNT | 23 | Count |
| | DFS | | Debug Fault State |
| | ERR | | Capture Overflow Error |
| | FAULTA | | Recoverable Fault A |
| | FAULTB | | Recoverable Fault B |
| | FAULT0 | | Non-Recoverable Fault 0 |
| | FAULT1 | | Non-Recoverable Fault 1 |
| | OVF | | Overflow/Underflow |
| | TRG | | Retrigger event |
| | UFS | | Non-Recoverable Update Fault |
| | MC | | Match or Capture |
| TCC2 | CNT | 24 | Count |
| | DFS | | Debug Fault State |
| | ERR | | Capture Overflow Error |
| | FAULTA | | Recoverable Fault A |
| | FAULTB | | Recoverable Fault B |
| | FAULT0 | | Non-Recoverable Fault 0 |
| | FAULT1 | | Non-Recoverable Fault 1 |
| | OVF | | Overflow/Underflow |
| | TRG | | Retrigger event |
| | UFS | | Non-Recoverable Update Fault |
| | MC | | Match or Capture |
| TC0 | OVF | 25 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC1 | OVF | 26 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC2 | OVF | 27 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC3 | OVF | 28 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC4 | OVF | 29 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC5 | OVF | 30 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC6 | OVF | 31 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-4.** NVIC Interrupt Mapping (continued)

| Module | Source | NVIC Line | Remark |
|---|---|---|---|
| TC7 | OVF | 32 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC8 | OVF | 33 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| TC9 | OVF | 34 | Overflow/Underflow |
| | ERR | | Error |
| | MC | | Match or Capture |
| ADCTRL | GIRQ | 35 | Global |
| | DIRQ0 | | Digital comparator |
| | DIRQ1 | | Digital comparator |
| | AIRQ0 | | Digital filter |
| | AIRQ1 | | Digital filter |
| | FLT | 36 | Fault |
| | EOS_RDY | 37 | End-of-screen ready |
| | FCC | | First class channel BVMI DMA group |
| | BGVR_RDY | 38 | RDY ADC analog circuit |
| AC | COMP0 | 39 | Change in comparator0 status |
| | COMP1 | | Change in comparator1 status |
| | WIN_0 | | Change in window0 status |
| Crypto | INT0 | 40 | Crypto Host |
| | INT1 | 41 | TRNG |
| QSPI | LINE | 42 | QSPI |
| Wireless SIB | ZB_INT0 | 43 | 802.15.4 interrupt |
| | BT_INT0 | 44 | Bluetooth interrupt |
| | BT_INT1 | 45 | Bluetooth interrupt |
| | ARBITER | 46 | Arbiter |
| | CLKI_WAKEUP_NMI | 47 | Clock input with a wake-up trigger |
| CVD | CVD | 48 | CVD event |
| Crypto | INT2 | 49 | Crypto interrupt2 |
| QEI | upbs_event[1] | 50 | QEI interrupt |
| CAN0 | LINE0 | 51 | CAN 0 LINE0 |
| | LINE1 | | CAN 0 LINE1 |
| | ERROR | | Error |
| CAN1 | LINE0 | 52 | CAN1 LINE0 |
| | LINE1 | | CAN1 LINE1 |
| | ERROR | | Error |
| ETH | eth_intreq_q | 53 | Ethernet interrupt signal synchronous to APB clock |
| USB | usbcore1_interrupt | 54 | USB core 1 interrupt |
| Wireless SIB | pll_locked_out | 55 | PLL lock for Boot |
| | pll_locked_out | 56 | PLL lock for User |
| BLE Stack(1) | Firmware Interrupt only | 57 | Firmware interrupt only |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-4.** NVIC Interrupt Mapping (continued)

| Module | Source | NVIC Line | Remark |
|---|---|---|---|

**Notes:**

1. The last interrupt index is a software only interrupt intended to be used as a software interrupt by the Bluetooth LE stack. It will be used as BT_LC_IRQ by the stack. This does not have a associated hardware interrupt.
2. "-" indicates that there is no source for that module.

## 9.3. High-Speed Bus System

The high-speed bus system matrix connects a multitude of initiator logic cores/IPs to a multitude of target logic cores/IPs, supporting AHB2/APB2 buses.

### 9.3.1. Features

High-Speed Bus Matrix has the following features:

- AMBA Advanced High-performance Bus (AHB Lite)-compliant interfaces
- Symmetric crossbar bus switch implementation
- Allows concurrent accesses from different Initiator to different target
- 32-bit data bus
- APB compliant user interface


© 2025 Microchip Technology Inc. and its subsidiaries

## 9.3.2. Configuration

**Figure 9-1.** High-Speed Bus Matrix Inter-connectivity

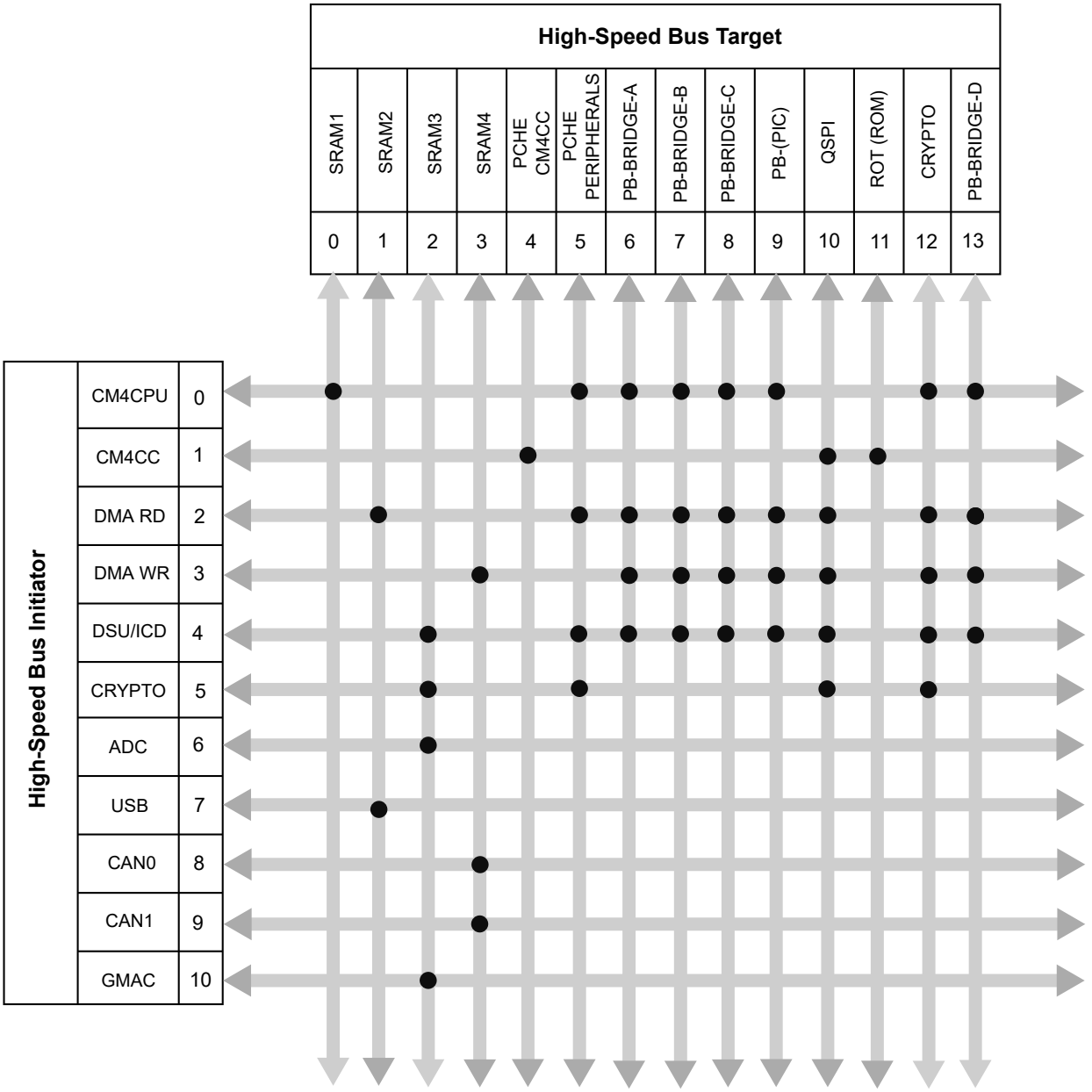



**Table 9-5.** High Speed Bus Matrix Initiator

| High-Speed Bus Matrix Initiator | Initiator ID |
|---|---|
| CM4CPU - Cortex-M4 System Bus | 0 |
| CM4CC - Cortex-M4 CMCC Bus | 1 |
| DMA RD – DMA-Read | 2 |
| DMA-WR – DMA-Write | 3 |
| DSU/ICD (Test mode only) – Device Service Unit/In-Chip Debugger | 4 |
| CRYPTO | 5 |
| ADC | 6 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 9-5.** High Speed Bus Matrix Initiator (continued)

| High-Speed Bus Matrix Initiator | Initiator ID |
|---|---|
| USB | 7 |
| CAN0 | 8 |
| CAN1 | 9 |
| GMAC | 10 |

**Table 9-6.** High-Speed Bus Matrix Target

| High-Speed Bus Matrix Target | Target ID |
|---|---|
| SRAM1 – SRAM Port 1 | 0 |
| SRAM2 – SRAM Port 2 | 1 |
| SRAM3 – SRAM Port 3 | 2 |
| SRAM4 – SRAM Port 4 | 3 |
| PCHE – Prefetch Cache of CM4CC | 4 |
| PCHE – Prefetch Cache of Peripherals | 5 |
| PB-BRIDGE-A – Peripheral Bridge A | 6 |
| PB-BRIDGE-B – Peripheral Bridge B | 7 |
| PB-BRIDGE-C – Peripheral Bridge C | 8 |
| PB-(PIC) – Peripheral Bridge D | 9 |
| QSPI – Quad SPI Interface | 10 |
| ROT – Root of Trust | 11 |
| CRYPTO | 12 |
| PB-BRIDGE-D – Peripheral Bridge D | 13 |


© 2025 Microchip Technology Inc. and its subsidiaries

# 10. Prefetch Cache (PCHE)

## 10.1. Overview

The prefetch cache is a performance-enhancing module included in the PIC32CX-BZ6 devices, along with the L1 cache (Cortex M Cache Controller) to the Cortex-M4 CPU.

## 10.2. Features

The Prefetch module increases the system performance for most of the applications.

The Prefetch module includes the following features:

- Fully Associative Lines For:
  - Four lines for CPU instructions cache
  - Two lines for CPU data cache
  - Two lines for peripheral data cache
- 32-Byte Cache Lines and 256-Bits Parallel Memory Fetch
- Configurable Predictive Prefetch for CPU Instructions Cache
- Error Detection and Correction

## 10.3. Block Diagram

When running the prefetch module at high-clock rates, insert the Wait states into Program Flash Memory (PFM) read transactions to meet the access time of the PFM. The user can hide the Wait states to the core by prefetching and storing the instructions in a temporary holding area that the CPU can access quickly. Although, the data path to the CPU is 32 bits wide, the data path to the PFM is 256 bits wide. This wide data path provides the same bandwidth to the CPU as a 32-bit path running at eight times the frequency.

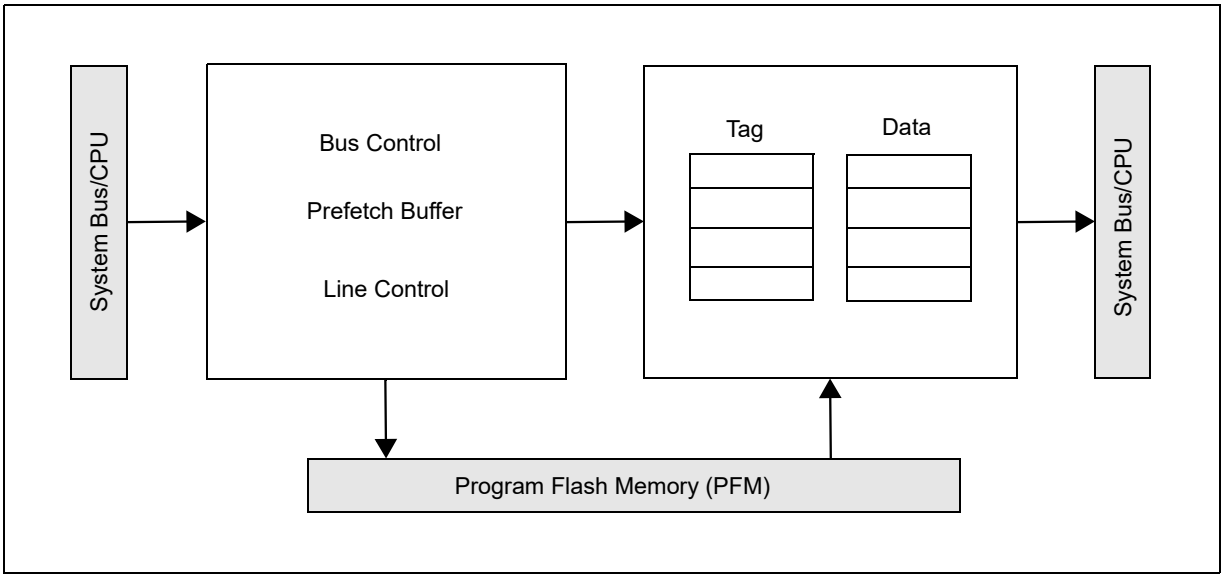
The prefetch module holds a subset of PFM in temporary holding spaces known as lines. Each line contains a tag and data field. In general, the lines hold a copy of what is currently in memory to make instructions or data available to the CPU without the Wait states.

The CPU or a peripheral can request the data located in the PFM. If the requested data is not currently stored in the prefetch module line, a read is performed to the PFM at the correct address, and the data is supplied to the prefetch module and to the CPU or peripheral. If the requested data is stored in the prefetch module and is valid, the data is supplied to the CPU or peripheral without Wait states.

The following figure illustrates a block diagram of the prefetch module. Logically, the prefetch module fits between the system bus module and the PFM.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 10-1.** Prefetch Cache Block Diagram




### 10.3.1. Line Organization

The Prefetch module consists of two arrays, data and tag, each of which hold four lines. A data array consists of program instructions, program data or peripheral data. Address matches are based on the physical address, not the virtual address.

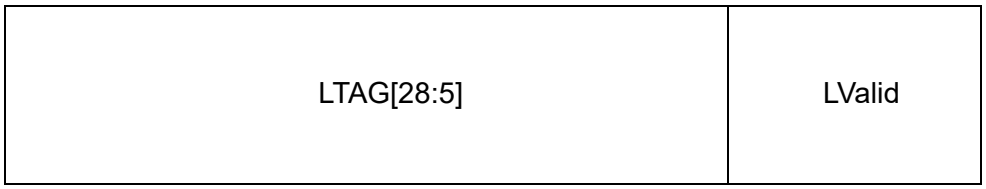
Each line in the tag array contains the following information:

- Tag – Physical address of the data held in the data line
- Valid bit

Each line in the data array, contains 32 bytes of data. Depending on the line, the data can be CPU instructions, CPU data or peripheral data.

The following figures illustrate the organization of a line.

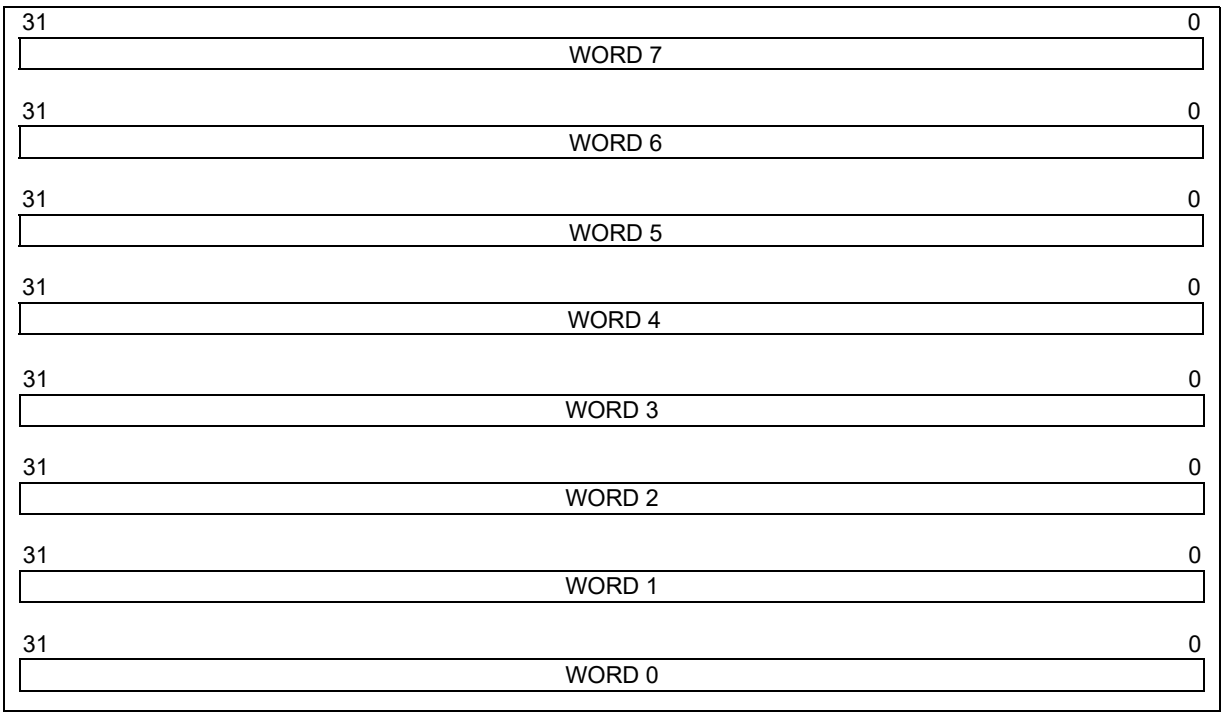
**Figure 10-2.** Tag Line





© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 10-3.** Data Line




## 10.4. Product Dependencies

Not applicable.

### 10.4.1. I/O Lines

Not applicable.

### 10.4.2. Power Management

#### 10.4.2.1. Standby Sleep Mode

When the device enters Standby Sleep mode, the Prefetch module is disabled and placed into a Low-power state where no clocking occurs in the module.

#### 10.4.2.2. Idle Mode

When the device enters Idle mode, iCache, Prefetch and dCache clocks are internally gated-off, the aCache (peripheral data) clock remains functional for peripheral accesses and the CPU stops executing code. Any outstanding prefetch completes before the Prefetch module stops its clock through automatic clock gating.

### 10.4.3. Clocks

The PCHE interfaces with the CPU through the AHB (SYS_CLK).

### 10.4.4. DMA

Not applicable.

### 10.4.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the PCACHE interrupt(s) requires the NVIC interrupt controller to be configured first.


© 2025 Microchip Technology Inc. and its subsidiaries

### 10.4.6. Events

Not applicable.

### 10.4.7. Debug Operation

The behavior of the Prefetch module is unaltered in the Debug mode.

### 10.4.8. Register Access Protection

Not applicable.

### 10.4.9. Analog Connections

Not applicable.

## 10.5. Prefetch Behavior

The prefetch module complements an L1 CPU (CMCC) cache rather than replacing it. Four 256-bit (32-byte) lines hold instructions, two 256-bit (32-byte) lines hold CPU data and two 256-bit (32-byte) lines hold peripheral data from the PFM. The prefetch module uses the Wait state's value from the PFMWS[3:0] bits (CHECON[3:0]) and Address Wait state ADRWS bit (CHECON[8]) to determine how long it must wait for Flash access when it reads instructions or data from the PFM.

If the instructions or data already reside in the prefetch module line, the prefetch module returns the instruction or data in '0' Wait states. For CPU instructions, if predictive prefetch is enabled and the code is 100% linear, the prefetch module provides instructions back to the CPU with the Wait states only on the first instruction of the prefetch module line.

If the CPU accesses uncacheable addresses, it bypasses the cache. During the bypass, the prefetch module accesses the PFM for every instruction, incurring an address setup time defined by ADRWS and a Flash access time as defined by PFMWS bits. Therefore, the total Flash wait states is a sum of ADRWS and PFMWS. The Bypass mode is also forced for a cache if its associated I/D/A CHEEN bit (CHECON) is zero.

To allow caching for I and/or D caches, set the I and/or D *CHEEN bit to '1'. To enable a cache, set the ACHEEN bit to '1'.

## 10.6. Configurations

The CHECON register controls the general configurations available for accelerating the instruction and data accesses to the Flash memory system.

The Prefetch module implements the following general options:

- The PFMWS[3:0] bits (CHECON[3:0]) control the number of system clock cycles required to access the PFM. The total Flash Wait states is a sum of ADRWS and PFMWS.
- The PREFEN[1:0] bits (CHECON[5:4]) control the predictive and prefetched instruction, which allows the cache controller to fetch the next 32-byte aligned set of instructions.
- The PFMSECEN bit (CHECON[7]) controls the Prefetch module that generates an interrupt event on a specific count of single-bit errors corrected by the Flash Error Correction Code (ECC).
- The ADRWS bit (CHECON[8]) controls the number of system clock cycles required for address setup to PFM.
- The CHEPERF bit (CHECON[12]) controls the gathering statistics of the CPU instruction cache.
- The ICHECOH bit (CHECON[16]) controls the auto invalidate for the CPU instruction cache.
- The DCHECOH bit (CHECON[17]) controls the auto invalidate for the CPU data cache.
- The ACHECOH bit (CHECON[18]) controls the auto invalidate for the peripheral data cache.
- The ICHEINV bit (CHECON[20]) controls the manual invalidate for the CPU instruction cache.
- The DCHEINV bit (CHECON[21]) controls the manual invalidate for the CPU data cache.


© 2025 Microchip Technology Inc. and its subsidiaries

- The ACHEINV bit (CHECON[22]) controls the manual invalidate for the peripheral data cache.
- The ICHEEN bit (CHECON[24]) controls the CPU instruction cache enable.
- The DCHEEN bit (CHECON[25]) controls the CPU data cache enable.
- The ACHEEN bit (CHECON[26]) controls the peripheral data cache enable.

## 10.7. Predictive Prefetch Behavior

When the user configures the module for predictive prefetch, the prefetch module predicts the next line address, fetches the instruction, then, stores it in the prefetch buffer. If the requested instruction is not in a prefetch module line and the read address matches the predicted address, the content of the prefetch buffer is loaded in the prefetch module line while simultaneously returning the critical word to the read initiator.

On enabling the predictive prefetch, the prefetch function starts predicting based on the first address read to the PFM. When the user places the first line in the prefetch module, the module increments the address to the next 32-byte aligned address and starts a PFM access.

The predictive prefetches, like all PFM read accesses, are never aborted. If a new address request does not match the predicted address, a new PFM access occurs after the current access finishes. The PREFEN [1:0] bits (CHECON[5:4]) can start a predictive prefetch. This allows the cache controller to speculatively fetch the next 32-byte aligned set of instructions. The predictive prefetch feature is available only for CPU instruction but not for CPU data and peripheral.

If the selected system clock speed is sufficiently low enough to access the Flash at zero Wait states, the predictive prefetch is detrimental and may be disabled.

## 10.8. Coherency Support

When a PFM programming event causes flash programming initiated by the Flash controller, the prefetch module invalidates all lines and the contents of the prefetch buffer. If a transaction is in progress, the invalidation occurs after completion. When programming or erasing a Flash page, a read of that Flash page causes the transaction to stall until the erase or program event completes.

The prefetch module provides two methods for coherency control:

- Auto invalidate via I/D/A CHECOH
- Manual invalidate via I/D/A CHEINV

The user can choose to auto invalidate the each/any cache on a PFM programming event by setting *CHECOH = `1`. This is the safest option. However, the user has the option to never auto invalidate each/any cache by setting *CHECOH = `0`.

In addition to using *CHECOH, use *CHEINV as an alternate invalidate method to invalidate each/any cache manually. If using *CHEINV to manually invalidate each/any cache due to a PFM programming event, stop all instruction/data fetches from the desired Flash, set *CHEINV, wait for it to clear and, then, start the programming sequence. When using *CHEINV to invalidate each/any cache for reasons other than programming, it can be set at any time but only takes effect after any pending transactions complete.

## 10.9. Effects of Reset

### 10.9.1. On Reset

Upon a device Reset, the following occurs:

- All lines are invalidated
- All tag bits are cleared

### 10.9.2. After Reset

The module operates as per the values in the CHECON register. See the *CHECON* register from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**

CHECON

CHECON - Prefetch Module Control Register

## 10.10. Error Conditions

The prefetch module handles and reports information about two error types:

- ECC Double-bit Error Detected (DED)
- ECC Single-bit Error Corrected (SEC)

The user can enable and disable the ECC Error detection logic using the configuration bits, ECCCTL[1:0] (CFGCON0/DEVCFG0[29:28]).

The ECC logic increases the read access delay from the PFM. Depending on the frequency of the system clock, the Wait states can be different between ECC-enabled and ECC-disabled.

**Note:** ECC errors are captured for predictive prefetch reads of the PFM. However, do not report those errors until, and unless, the system starts using that data.

### 10.10.1. ECC Double-bit Error Detected (DED)

A read from the Flash memory that results in a PFM ECC DED causes the Prefetch module to return a bus exception error to the initiator. If that initiator is the CPU, it recognizes the bus exception error, prevents the instruction from executing, or read data from loading and generates an exception using the bus exception error vector.

When an ECC DED error occurs, the PFMDED bit (CHESTAT[27]) is set. The exception handling code can, then, check this bit to determine whether the PFM ECC DED event is causing an exception or not. The exception handler clears this bit in software.

**Note:** CPU instructions or data prefetched from the PFM is always loaded into the Prefetch module, even if a DED error is generated. The Prefetch module line containing the DED data is tagged as valid until the line is replaced.

### 10.10.2. ECC Single Error Corrected (SEC)

A PFM ECC SEC event is not a critical error and, as such, is reported through an interrupt. The user has the option to enable or disable this interrupt through the PFMSECEN bit (CHECON[7]). The data in the prefetch module is correct, and no further ECC events are generated for addresses that hit the data line as long as that data is in the prefetch module.

Each read that returns from the PFM with an ECC SEC status causes the PFMSECCNT[7:0] bits (CHESTAT[7:0]) to decrement by one. If PFMSECCNT[7:0] is zero and a PFM ECC SEC event occurs, the PFMSEC bit (CHESTAT[26]) is set and generates an interrupt. Therefore, the PFMSECCNT[7:0] bits must be set to the number of PFM ECC SEC events desired for an interrupt minus 1. For example, to generate an interrupt after five PFM ECC SEC events, PFMSECCNT[7:0] must be set to four ('00000100'). The prefetch module does not reload the PFMSECCNT[7:0] bits when it reaches '0'. Software must write the desired count each time it services the PFMSEC interrupt.

Software can generate an ECC SEC interrupt by setting the PFMSECEN bit, then setting the PFMSEC bit. If the PFMSEC bit is already set when PFMSECEN is set, the prefetch module generates an ECC SEC interrupt. The ECC SEC interrupt persists as long as the PFMSECEN and PFMSEC bits remain set.


© 2025 Microchip Technology Inc. and its subsidiaries

## 10.11. Register Summary

See the *PCHE* module in the *Product Memory Mapping Overview* from Related Links for base address.

**Note:** CHECON and CHESTAT registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CHECON | 7:0 | PFMSECEN | | PREFEN[1:0] | | PFMWS[3:0] | | | |
| | | 15:8 | | | | CHEPERF | | | | ADRWS |
| | | 23:16 | | ACHEINV | DCHEINV | ICHEINV | | ACHECOH | DCHECOH | ICHECOH |
| | | 31:24 | | | | | | ACHEEN | DCHEEN | ICHEEN |
| 0x04 ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | CHESTAT | 7:0 | PFMSECCNT[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | PFMDED | PFMSEC | | |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | CHEHIT | 7:0 | CHEHIT[7:0] | | | | | | | |
| | | 15:8 | CHEHIT[15:8] | | | | | | | |
| | | 23:16 | CHEHIT[23:16] | | | | | | | |
| | | 31:24 | CHEHIT[31:24] | | | | | | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | CHEMIS | 7:0 | CHEMIS[7:0] | | | | | | | |
| | | 15:8 | CHEMIS[15:8] | | | | | | | |
| | | 23:16 | CHEMIS[23:16] | | | | | | | |
| | | 31:24 | CHEMIS[31:24] | | | | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 10.12. Register Description

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown


© 2025 Microchip Technology Inc. and its subsidiaries

### 10.12.1. CHECON - Prefetch Module Control Register

**Name:** CHECON
**Offset:** 0x00
**Reset:** 0x0700010F
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | ACHEEN | DCHEEN | ICHEEN |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | ACHEINV | DCHEINV | ICHEINV | | ACHECOH | DCHECOH | ICHECOH |
| Access | | R/S/HC | R/S/HC | R/S/HC | | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | CHEPERF | | | | ADRWS |
| Access | | | | R/W | | | | R/W |
| Reset | | | | 0 | | | | 1 |

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

**Bit 26 – ACHEEN** Peripheral Data Cache Enable bit

| Value | Description |
|---|---|
| 1 | Caching enabled |
| 0 | Caching disabled (and all lines invalidated) |

**Bit 25 – DCHEEN** Data Cache Enable bit

| Value | Description |
|---|---|
| 1 | Caching enabled |
| 0 | Caching disabled (and all lines invalidated) |

**Bit 24 – ICHEEN** Instruction Data Cache Enable bit

| Value | Description |
|---|---|
| 1 | Caching enabled |
| 0 | Caching disabled (and all lines invalidated) |

**Bit 22 – ACHEINV** Manual Invalidate Control for Peripheral Data Cache
**Note:** Hardware auto clears this bit when cache invalidate completes. Bits may clear at different times.

| Value | Description |
|---|---|
| 1 | Force invalidate cache/invalidate busy |
| 0 | Cache invalidation follows ACHECOH/invalid complete |

**Bit 21 – DCHEINV** Manual Invalidate Control for Data Cache
**Note:** Hardware auto clears this bit when cache invalidate completes. Bits may clear at different times.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Force invalidate cache/invalidate busy |
| 0 | Cache invalidation follows DCHECOH/invalid complete |

**Bit 20 – ICHEINV** Manual Invalidate Control for Instruction Cache

**Notes:**

- Predictive Prefetch Buffer (PFB) is included with iCache invalidate.
- Hardware auto clears this bit when cache invalidate completes. Bits may clear at different times.

| Value | Description |
|---|---|
| 1 | Force invalidate cache/invalidate busy |
| 0 | Cache invalidation follows ICHECOH/invalid complete |

**Bit 18 – ACHECOH** Auto Cache Coherency Control for Peripheral Data Cache

**Note:** ACHECOH must be stable before initiation of programming to ensure correct invalidation of data.

| Value | Description |
|---|---|
| 1 | Auto invalidate cache on a programming event |
| 0 | No auto invalidated cache on a programming event |

**Bit 17 – DCHECOH** Auto Cache Coherency Control for Data Cache

**Note:** DCHECOH must be stable before initiation of programming to ensure correct invalidation of data.

| Value | Description |
|---|---|
| 1 | Auto invalidate cache on a programming event |
| 0 | No auto invalidated cache on a programming event |

**Bit 16 – ICHECOH** Auto Cache Coherency Control for Instruction Cache

**Note:** ICHECOH must be stable before initiation of programming to ensure correct invalidation of data.

| Value | Description |
|---|---|
| 1 | Auto invalidate cache on a programming event |
| 0 | No auto invalidated cache on a programming event |

**Bit 12 – CHEPERF** Cache Performance Counters Enable

**Note:** Performance counters are reset on 0 to 1 transition of this bit.

| Value | Description |
|---|---|
| 1 | Enable performance counters |
| 0 | Disable performance counters |

**Bit 8 – ADRWS** Address Wait State Enable

Total Flash wait states are ADRWS + PFMWS.

**Note:** CPU hang is observed when CHECON.ADRWS configuration switches from '1' to '0' and a Flash read access. To Avoid CPU hangs, execute the CHECON configuration from SRAM or Boot ROM during system initialization until the configuration change is done, then resume execution from Flash after configuration is set.

| Value | Description |
|---|---|
| 1 | Add 1 address Wait state - Allowing for higher clock frequencies |
| 0 | Add 0 address Wait states - Allowing for higher performance at lower clock frequencies |

**Bit 7 – PFMSECEN** Flash Single-bit Error Corrected (SEC) Interrupt Enable bit


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Generate an interrupt when PFMSEC is set |
| 0 | Do not generate an interrupt when PFMSEC is set |

**Bits 5:4 – PREFEN[1:0]** Instruction Predictive Prefetch Enable

| Value | Description |
|---|---|
| 01 | Instruction predictive prefetch enabled for cacheable regions only |
| 00 | Instruction predictive prefetch disabled |

**Note:** Other values are unavailable.

**Bits 3:0 – PFMWS[3:0]** PFM Access Time Defined in Terms of SYSCLK Wait States bits
Total Flash Wait states are ADRWS + PFMWS.

| Value | Description |
|---|---|
| 1111 | Fifteen Wait states |
| 1110 | Fourteen Wait states |
| ... | |
| 0001 | One Wait state |
| 0000 | Zero Wait state |

**Notes:**

- This is not the Wait state seen by the CPU.
- For the Wait states-to-SYSCLK relationship, see *Electrical Characteristics* from Related Links.

**Related Links**
Electrical Characteristics


© 2025 Microchip Technology Inc. and its subsidiaries

### 10.12.2. CHESTAT - Prefetch Module Status Register

**Name:** CHESTAT
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | PFMDED | PFMSEC | | |
| Access | | | | | HS | HS | | |
| Reset | | | | | 0 | 0 | | |

| 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 |
|---|---|---|---|---|---|---|---|---|
| | PFMSECCNT[7:0] | | | | | | | |
| Access | HS | HC | R/W | X | X | X | X | X |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 27 – PFMDED** Flash Double-bit Error Detected (DED) Status bit
This bit is set in hardware and can only be cleared (set to '0') in software.

| Value | Description |
|---|---|
| 1 | A DED error has occurred |
| 0 | A DED error has not occurred |

**Bit 26 – PFMSEC** Flash Single-bit Error Corrected (SEC) Status bit
**Note:** The PCACHE interrupt event reports the error event to the CPU. See *Nested Vector Interrupt Controller (NVIC)* from Related Links.

| Value | Description |
|---|---|
| 1 | A SEC error occurred when PFMSECCNT[7:0] equals to zero |
| 0 | A SEC error has not occurred |

**Bits 7:0 – PFMSECCNT[7:0]** Flash SEC Count bits
The count value decreases by '1' each time an SEC error occurs. Holds at zero. When an SEC error occurs when PFMSECCNT[7:0] is zero, the PFMSEC status bit is set. If PFMSECEN is also set, a prefetch module interrupt event is generated.

**Related Links**
Nested Vector Interrupt Controller (NVIC)


© 2025 Microchip Technology Inc. and its subsidiaries

### 10.12.3. CHEHIT – Prefetch Module Hit Statistics Register

**Name:** CHEHIT
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** -

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

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

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

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

**Bits 31:0 – CHEHIT[31:0]** Instruction Cache Hit Count bits
When CHECON.CHEPERF = 1, CHEHIT increments once per iCache or Predictive Prefetch Buffer (PFB) hit.
**Note:** CHEHIT is Reset on the '0' to '1' transition of CHECON.CHEPERF.


© 2025 Microchip Technology Inc. and its subsidiaries

### 10.12.4. CHEMIS – Prefetch Module Miss Statistics Register

**Name:** CHEMIS
**Offset:** 0x30
**Reset:** 0x00000000
**Property:** -

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

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

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

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

**Bits 31:0 – CHEMIS[31:0]** Instruction Cache Miss Count bits
When CHECON.CHEPERF = 1, CHEMIS increments once per iCache or Predictive Prefetch Buffer (PFB) miss.
**Note:** CHEMIS is Reset on the '0' to '1' transition of CHECON.CHEPERF.


© 2025 Microchip Technology Inc. and its subsidiaries

# 11. Cortex M Cache Controller (CMCC)

## 11.1. Overview

The Cortex M Cache Controller provides an L1 cache to the Cortex M CPU. The CMCC sits transparently between the CPU and the cache leading to improved performance.

The CMCC interfaces with the CPU through the AHB and is connected to the APB bus interface for its configuration.

## 11.2. Features

- Physically Addressed and Physically Tagged
- L1 Data and Instruction Cache Set to 4 KB
- L1 Cache Line Size Set to 16 Bytes
- L1 Cache Integrates 32-Bit Bus Host Interface
- Unified 4-Way Set Associative Cache Architecture
- Lock-Down Feature, Which Allows Cached to be Locked Per Way
- Write Through Cache Operations, Read Allocate
- Configurable as Data and Instruction Tightly Coupled Memory (TCM)
- Round Robin Victim Selection Policy
- Event Monitoring, with One Programmable 32-Bit Counter
- Cache Interface Includes Cache Maintenance Operations Registers


© 2025 Microchip Technology Inc. and its subsidiaries

## 11.3. Block Diagram

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

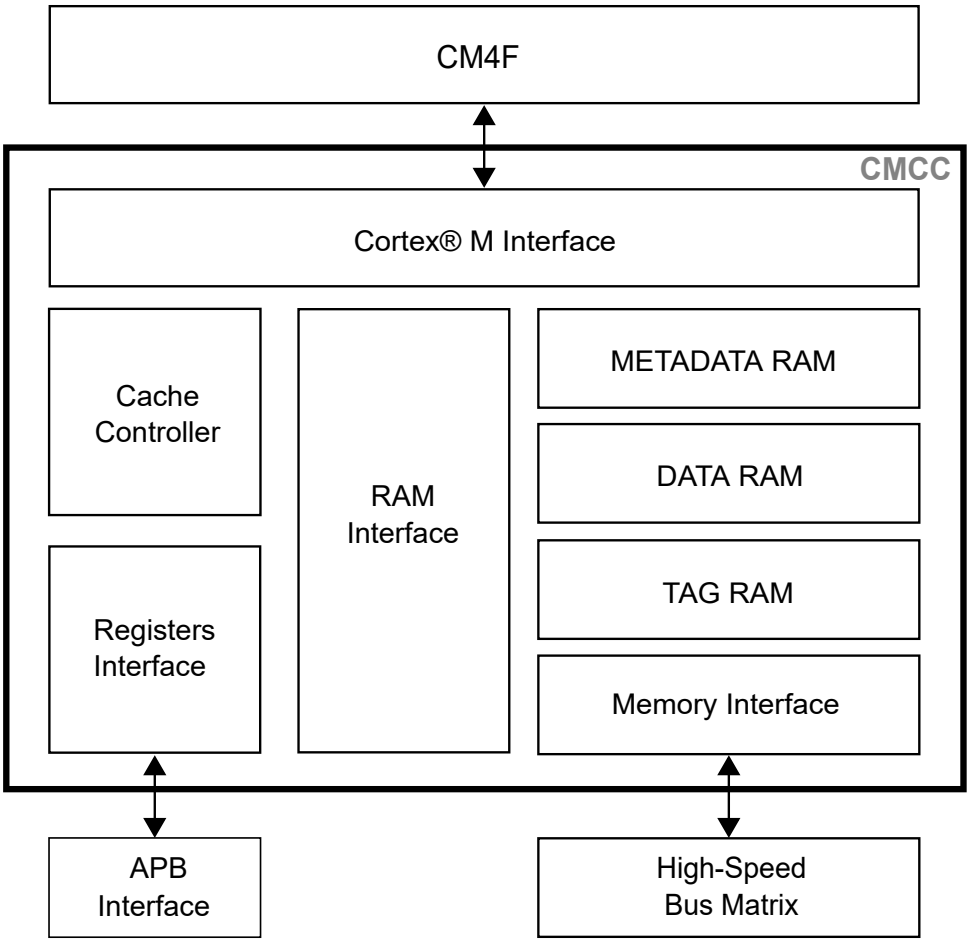
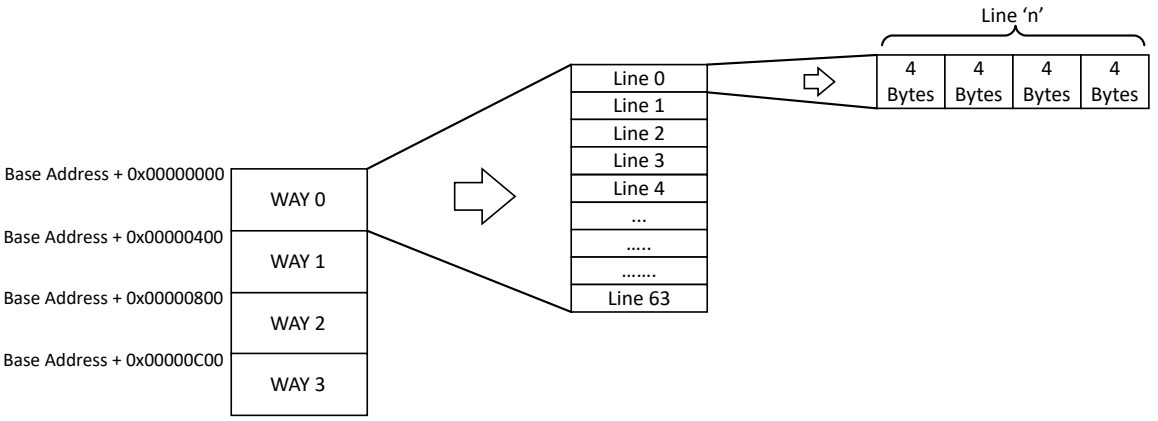



**Figure 11-2.** CMCC Organization





© 2025 Microchip Technology Inc. and its subsidiaries

## 11.4. Signal Description

Not applicable.

## 11.5. Product Dependencies

Not applicable.

### 11.5.1. I/O Lines

Not applicable.

### 11.5.2. Power Management

The CMCC will continue to function as long as the CPU is not sleeping and the CMCC is enabled.

### 11.5.3. Clocks

The CMCC interfaces with the CPU through the AHB (SYS_CLK) and is connected to the APB bus (PB2_CLK) interface for its configuration.

### 11.5.4. DMA

Not applicable.

### 11.5.5. Interrupts

Not applicable.

### 11.5.6. Events

Not applicable.

### 11.5.7. Debug Operation

When the CPU is halted in debug mode, the CMCC is halted. Any read access by the debugger in cached zones are not cached.

### 11.5.8. Register Access Protection

Not applicable.

### 11.5.9. Analog Connections

Not applicable.

## 11.6. Functional Description

### 11.6.1. Principle of Operation

### 11.6.2. Initialization and Normal Operation

On reset, the cache controller data entries are all invalidated and the cache is disabled. The cache is transparent to processor operations. The user can activate the cache controller by using its configuration registers. The configuration interface is memory mapped in the APB bus.

Use the following sequence to enable the cache controller:

- Verify that the CMCC is disabled, reading the value of the SR.CSTS.
- Enable the CMCC by writing '1' in CTRL.CEN. The MODULE is disabled by writing a '0' in CTRL.CEN.

### 11.6.3. Change Cache Size

It is possible to change the cache size by writing to the Cache Size Configured By Software bits in the Cache Configuration register (CFG.CSIZESW).

Use the following sequence to change the cache size:


© 2025 Microchip Technology Inc. and its subsidiaries

- Disable the CMCC controller by writing a zero to the Cache Controller Enable bit in the Cache Control register (CTRL.CEN = 0).
- Check the Cache Controller Status bit in the Cache Status register to verify that the CMCC is successfully disabled (SR.CSTS = 0).
- Change CFG.CSIZESW to its new value.
- Enable the CMCC by writing CTRL.CEN = 1.

### 11.6.4. Data Cache Disable

The instructions alone can be cached by disabling the data cache, as described in the following steps:

1. Disable the cache controller by writing a '0' to CTRL.CEN.
2. Check SR.CSTS to verify that the CMCC is successfully disabled.
3. Write CFG.DCDIS = 1.
4. Enable the CMCC by writing CTRL.CEN = 1.

### 11.6.5. Instruction Cache Disable

The data alone can be cached by disabling the instruction cache, as described in the following steps:

1. Disable the cache controller by writing CTRL.CEN = 0.
2. Check SR.CSTS to verify that the CMCC is successfully disabled.
3. Write CFG.ICDIS = 1.
4. Enable the CMCC by writing CTRL.CEN = 1.

### 11.6.6. Cache Load and Lock

It is possible to lock a specific way for code optimization by writing the Lock Way register (LCKWAY.LCKWAY). The CMCC does not update the locked way as part of cache operations.

The load and lock mechanism can be implemented to use cache memory in a deterministic way. Follow these steps to load and lock a way:

1. Disable cache controller by clearing the CTRL.CEN bit.
2. Invalidate the desired WAY line by line. This resets the round robin algorithm of the invalidated line, which becomes eligible for the next load operation.
3. Disable the instruction cache, but keep the data cache enabled.
4. Enable the cache by setting the CTRL.CEN bit.
5. Place the respective piece of code and/or data to the corresponding WAY due to simple LOAD operations. Loading the piece of code and/or data forces the cache to refill the previous invalidated line in the right way. Validate only the first byte. The cache automatically refills the complete line.
6. Lock the specific WAY by setting LCKWAY.LCKWAY[3:0].
7. Re-enable the instruction cache. The locked WAY is now loaded and ready to operate. The user can use the remaining WAYS as I-cache or D-cache as per the requirement.

### 11.6.7. Tightly Coupled Memory

Users can use a part of the cache as Tightly Coupled Memory (TCM). The Cache Size Configuration determines the cache size by Software bits in the Cache Configuration register (CFG.CSIZESW). The relation between cache and TCM is as given below:

TCM size = (Maximum cache size - Configured cache size)


© 2025 Microchip Technology Inc. and its subsidiaries

The user can obtain the TCM start address from the product memory mapping. The cache memory starts first from the address followed by the TCM memory. Size of the Way is fixed and the number of ways varies according to the available size for the cache memory. See *Product Memory Mapping Overview* from Related Links.

**Table 11-1.** TCM Sizes

| Maximum Cache | Configured Cache | TCM Size |
|---|---|---|
| 4 KB | 4 KB | 0 KB |
| 4 KB | 2 KB | 2 KB |
| 4 KB | 1 KB | 3 KB |
| 4 KB | 0 KB | 4 KB |

The TCM is also accessible in its maximum size in case of disabling the CMCC. The TCM does not need to be locked to operate.

**Note:** Writing into the cache DATA RAM region through the CPU can overwrite the valid cache lines. This can result in data corruption when the cache controller is accessing the data for cache transactions. Access the DATA RAM region only after configuring it as TCM.

**Related Links**

Product Memory Mapping Overview

### 11.6.8. Cache Maintenance

#### 11.6.8.1. Cache Invalidate by Line Operation

If issuing an invalidate by line command, the CMCC resets the valid bit information of the decoded cache line. As the line is no longer valid, the replacement counter points to that line.

- Disable the cache controller by writing a zero to the Cache Controller Enable bit in the Cache Control register (CTRL.CEN).
- Check SR.CSTS to verify that the CMCC is successfully disabled.
- Perform an invalidate by line by writing the set `{index,way}` in the Cache Maintenance 1 register (MAINT1.INDEX, MAINT1.WAY).
- Enable the CMCC by writing '1' to CTRL.CEN.

#### 11.6.8.2. Cache Invalidate All Operation

Use the following sequence to invalidate all cache entries.

- Disable the cache controller by writing '0' to the Cache Enable bit in the Cache Control register (CTRL.CEN).
- Check SR.CSTS to verify that the CMCC is successfully disabled.
- Perform a full invalidate operation by writing '1' to the Cache Controller Invalidate All bit in the Cache Maintenance 0 register (MAINT0.INVALL).
- Enable the CMCC by writing '1' to CTRL.CEN.

### 11.6.9. Cache Performance Monitoring

The Cortex M cache controller includes a programmable monitor/32-bit counter. The user can configure the monitor to count the number of clock cycles, the number of data hit or the number of instructions hit.

It is important to know that the Cortex-M4 processor prefetches instructions ahead of execution. It performs only 32-bit read access on the instruction bus, which means:

- One arm instruction is fetched per bus access
- Two thumb instructions are fetched per bus access


© 2025 Microchip Technology Inc. and its subsidiaries

As a consequence, two thumb instructions (for example, NOP) need one bus access, which results in the hit counter incrementing by 1.

Use the following sequence to activate the counter:

- Configure the monitor counter by writing the MCFG.MODE.
  - CYCLE_COUNT is used to increment the counter along with the program counter to count the number of cycles.
  - IHIT_COUNT is the instruction hit counter, which increments the counter when there is a hit for the instruction in the cache.
  - DHIT_COUNT is the data hit counter, which increments the counter when there is a hit for the data in the cache.
- Enable the counter by writing a '1' to the Cache Controller Monitor Enable bit in the Cache Monitor Enable register (MEN.MENABLE).
- If required, reset the counter by writing a '1' to the Cache Controller Software Reset bit in the Cache Monitor Control register (MCTRL.SWRST).
- Check the value of the monitor counter by reading the MSR.EVENT_CNT bit field.

## 11.7. RAM Properties

The following table shows the different access properties of the three RAM blocks, according to the different modes described in the previous chapters.

**Table 11-2.** Access to RAM

| Access Condition | Data RAM | Tag RAM | Metadata RAM |
|---|---|---|---|
| CPU access when CMCC DISABLED | Read/Write | no Read/Write - hardfault | no Read/Write - hardfault |
| CPU access when CMCC ENABLED | CACHE section configured: Read/Write(1)<br>TCM section configured: Read/Write | no Read/Write - hardfault | no Read/Write - hardfault |
| Debugger access when CMCC DISABLED | Read/Write | Read/Write | Read/Write |
| Debugger access when CMCC ENABLED | CACHE section configured: Read/Write(1)<br>TCM section configured: R/W | no Read/Write | no Read/Write |

**Note:**

1. A write operation in this zone can corrupt the coherency of the cache. There is a need for an invalidate operation.


© 2025 Microchip Technology Inc. and its subsidiaries

## 11.8. Register Summary

See *CMCC* module in the *Product Memory Mapping Overview* from Related Links for base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | TYPE | 7:0 | LCKDOWN | WAYNUM[1:0] | | RRP | LRUP | RANDP | GCLK | AP |
| | | 15:8 | | | CLSIZE[2:0] | | | CSIZE[2:0] | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | CFG | 7:0 | | CSIZESW[2:0] | | | | DCDIS | ICDIS | GCLKDIS |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | CTRL | 7:0 | | | | | | | | CEN |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0C | SR | 7:0 | | | | | | | | CSTS |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x10 | LCKWAY | 7:0 | | | | | LCKWAY[3:0] | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | MAINT0 | 7:0 | | | | | | | | INVALL |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 | MAINT1 | 7:0 | INDEX[3:0] | | | | | | | |
| | | 15:8 | | | | | INDEX[7:4] | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | WAY[3:0] | | | | | | | |
| 0x28 | MCFG | 7:0 | | | | | | | MODE[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x2C | MEN | 7:0 | | | | | | | | MENABLE |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x30 | MCTRL | 7:0 | | | | | | | | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x34 | MSR | 7:0 | EVENT_CNT[7:0] | | | | | | | |
| | | 15:8 | EVENT_CNT[15:8] | | | | | | | |
| | | 23:16 | EVENT_CNT[23:16] | | | | | | | |
| | | 31:24 | EVENT_CNT[31:24] | | | | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 11.9. 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.


© 2025 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.

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.

Some registers are enable-protected, meaning they can only be written in case of disabling the peripheral. The enable-protected property denotes the enable protection in each individual register description.

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.1. Cache Type

**Name:** TYPE
**Offset:** 0x00
**Reset:** 0x000012D2
**Property:** R

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | CLSIZE[2:0] | | | CSIZE[2:0] | | |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 1 | 0 | 0 | 1 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | LCKDOWN | WAYNUM[1:0] | | RRP | LRUP | RANDP | GCLK | AP |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |

**Bits 13:11 – CLSIZE[2:0]** Cache Line Size
This field configures the cache line size. 0x02 is the value read for PIC32CX-BZ6 devices, as cache line size is 16 bytes.

| Value | Name | Description |
|---|---|---|
| 0x0 | CLSIZE_4B | Cache line size is 4 bytes |
| 0x1 | CLSIZE_8B | Cache line size is 8 bytes |
| 0x2 | CLSIZE_16B | Cache line size is 16 bytes |
| 0x3 | CLSIZE_32B | Cache line size is 32 bytes |
| 0x4 | CLSIZE_64B | Cache line size is 64 bytes |
| 0x5 | CLSIZE_128B | Cache line size is 128 bytes |
| 0x6-0x7 | — | Reserved |

**Bits 10:8 – CSIZE[2:0]** Cache Size
This bit field configures the cache size. 0x02 is the value read for PIC32CX-BZ6 devices, as cache size is 4 KB.

| Value | Name | Description |
|---|---|---|
| 0x0 | CSIZE_1KB | Cache size is 1 KB |
| 0x1 | CSIZE_2KB | Cache size is 2 KB |
| 0x2 | CSIZE_4KB | Cache size is 4 KB |
| 0x3 | CSIZE_8KB | Cache size is 8 KB |
| 0x4 | CSIZE_16KB | Cache size is 16 KB |
| 0x5 | CSIZE_32KB | Cache size is 32 KB |
| 0x6 | CSIZE_64KB | Cache size is 64 KB |
| 0x7 | — | Reserved |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 7 – LCKDOWN** Lock Down Supported

| Value | Description |
|---|---|
| 0 | Lockdown is not supported |
| 1 | Lockdown is supported |

**Bits 6:5 – WAYNUM[1:0]** Number of Way

This bit field configures the mapping of the cache. 0x02 is the value read for PIC32CX-BZ6 devices, as the device supports 4-Way set associative.

| Value | Name | Description |
|---|---|---|
| 0x0 | DMAPPED | Direct mapped cache |
| 0x1 | ARCH2WAY | 2-WAY set associative |
| 0x2 | ARCH4WAY | 4-WAY set associative |
| 0x3 | ARCH8WAY | 8-WAY set associative |

**Bit 4 – RRP** Round Robin Policy Supported

| Value | Description |
|---|---|
| 0 | Round robin policy is not supported |
| 1 | Round robin policy is supported |

**Bit 3 – LRUP** Least Recently Used Policy Supported

| Value | Description |
|---|---|
| 0 | Least recently used policy is not supported |
| 1 | Least recently used policy is supported |

**Bit 2 – RANDP** Random Selection Policy Supported

| Value | Description |
|---|---|
| 0 | Random victim selection is not supported |
| 1 | Random victim selection is supported |

**Bit 1 – GCLK** Dynamic Clock Gating

| Value | Description |
|---|---|
| 0 | Cache controller does not support clock gating |
| 1 | Cache controller uses dynamic clock gating |

**Bit 0 – AP** Access Port Access Allowed

| Value | Description |
|---|---|
| 0 | Access port access is disabled |
| 1 | Access port access is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.2. Cache Configuration

**Name:** CFG
**Offset:** 0x04
**Reset:** 0x00000020
**Property:** R/W

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | CSIZESW[2:0] | | | | DCDIS | ICDIS | GCLKDIS |
| Access | | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | | 0 | 1 | 0 | | 0 | 0 | 0 |

**Bits 6:4 – CSIZESW[2:0]** Cache Size Configured by Software
This field configures the cache size.

| Value | Name | Description |
|---|---|---|
| 0x0 | CONF_CSIZE_1KB | The Cache Size is configured to 1 KB |
| 0x1 | CONF_CSIZE_2KB | The Cache Size is configured to 2 KB |
| 0x2 | CONF_CSIZE_4KB | The Cache Size is configured to 4 KB |
| 0x3 | — | Reserved |

**Bit 2 – DCDIS** Data Cache Disable
Writing a '0' to this bit enables data caching.
Writing a '1' to this bit disables data caching.

**Bit 1 – ICDIS** Instruction Cache Disable
Writing a '0' to this bit enables instruction caching.
Writing a '1' to this bit disables instruction caching.

**Bit 0 – GCLKDIS** GCLK Dynamic Clock Gating
Writing a '0' to this bit disables the Dynamic Clock Gating feature.
Writing a '1' to this bit enables the Dynamic Clock Gating feature.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.3. Cache Control

**Name:** CTRL
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** Write-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 | | | | | | | | |

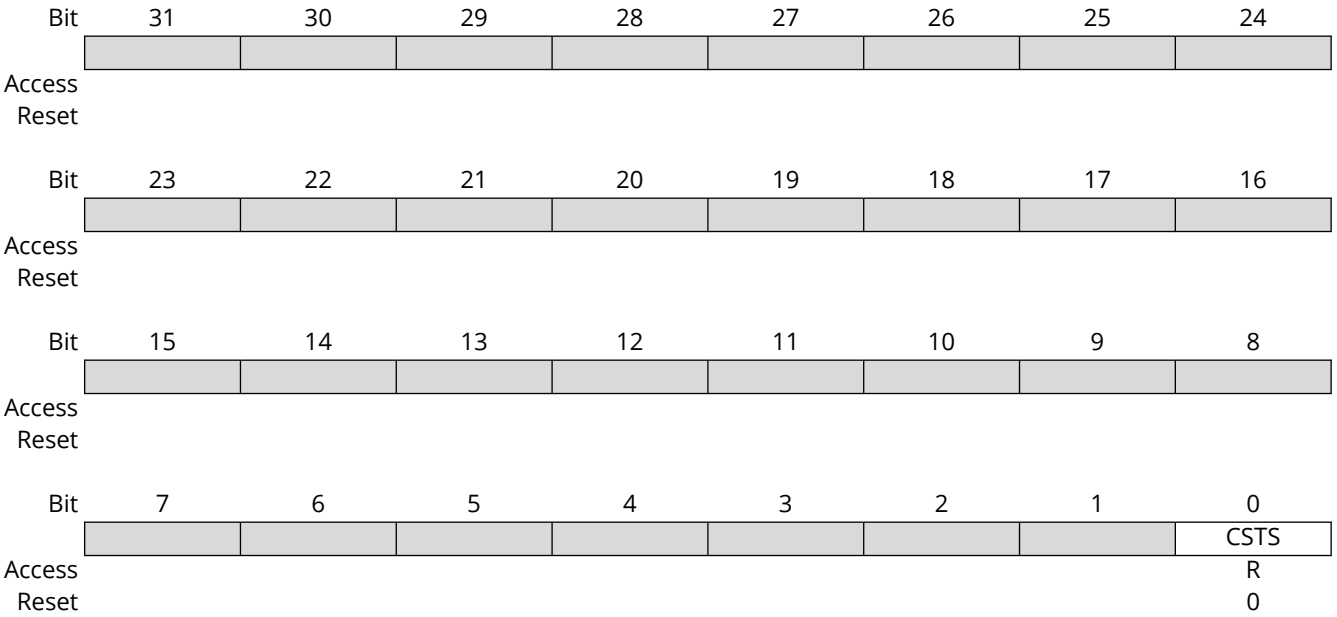
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | CEN |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – CEN** Cache Controller Enable
Writing a '0' to this bit disables the CMCC.
Writing a '1' to this bit enables the CMCC.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.4. Cache Status

**Name:** SR
**Offset:** 0x0C
**Reset:** 0x00000000
**Property:** Read-only




**Bit 0 – CSTS** Cache Controller Status
Writing to this bit has no effect.
Reading '0' shows CMCC is disabled.
Reading '1' shows CMCC is enabled.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.5. Cache Lock per Way

**Name:** LCKWAY
**Offset:** 0x10
**Reset:** 0x00000000
**Property:** Read/Write

| 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 | | | | | | | | |

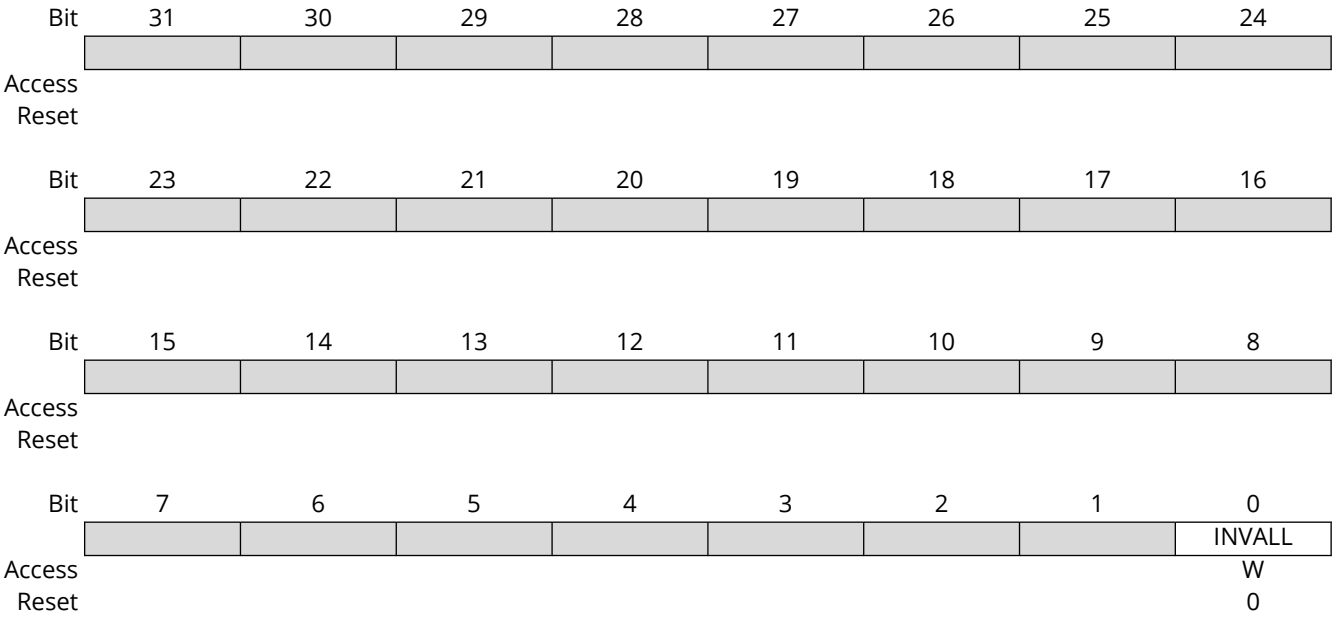
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | LCKWAY[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 3:0 – LCKWAY[3:0]** Lockdown Way Register
This field selects which way is locked.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.6. Cache Maintenance 0

**Name:** MAINT0
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** Write-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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | INVALL |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – INVALL** Cache Controller Invalidate All
Writing a '0' to this bit has no effect.
Writing a '1' to this bit invalidates all cache entries.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.7. Cache Maintenance 1

**Name:** MAINT1
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** Write-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | WAY[3:0] | | | | | | | |
| Access | W | W | W | W | | | | |
| Reset | 0 | 0 | 0 | 0 | | | | |

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

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | INDEX[7:4] | | | |
| Access | | | | | W | W | W | W |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bits 31:28 – WAY[3:0]** Invalidate Way

| Value | Name | Description |
|---|---|---|
| 0x0 | WAY0 | Way 0 is selection for index invalidation |
| 0x1 | WAY1 | Way 1 is selection for index invalidation |
| 0x2 | WAY2 | Way 2 is selection for index invalidation |
| 0x3 | WAY3 | Way 3 is selection for index invalidation |
| 0x4-0xF | | Reserved |

**Bits 11:4 – INDEX[7:0]** Invalidate Index
This field selects the index value for invalidation


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.8. Cache Monitor Configuration

**Name:** MCFG
**Offset:** 0x28
**Reset:** 0x00000000
**Property:** Read/Write

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | MODE[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bits 1:0 – MODE[1:0]** Cache Controller Monitor Counter Mode
This field selects the type of data monitored.

| Value | Name | Description |
|---|---|---|
| 0x0 | CYCLE_COUNT | Cycle counter |
| 0x1 | IHIT_COUNT | Instruction hit counter |
| 0x2 | DHIT_COUNT | Data hit counter |
| 0x3 | — | Reserved |


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.9. Cache Monitor Enable

**Name:** MEN
**Offset:** 0x2C
**Reset:** 0x00000000
**Property:** Read/Write

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | MENABLE |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bit 0 – MENABLE** Cache Controller Monitor Enable
Writing a '0' to this bit disables the monitor counter.
Writing a '1' to this bit enables the monitor counter.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.10. Cache Monitor Control

**Name:** MCTRL
**Offset:** 0x30
**Reset:** 0x00000000
**Property:** Write-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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SWRST |
| Access | | | | | | | | W |
| Reset | | | | | | | | 0 |

**Bit 0 – SWRST** Cache Controller Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets the event counter register.


© 2025 Microchip Technology Inc. and its subsidiaries

### 11.9.11. Cache Monitor Status

**Name:** MSR
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** Read-only

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | EVENT_CNT[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 |
|---|---|---|---|---|---|---|---|---|
| | EVENT_CNT[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 |
|---|---|---|---|---|---|---|---|---|
| | EVENT_CNT[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 |
|---|---|---|---|---|---|---|---|---|
| | EVENT_CNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – EVENT_CNT[31:0]** Monitor Event Counter
This field indicates the Monitor Event Counter value.


© 2025 Microchip Technology Inc. and its subsidiaries

# 12. Secure Boot ROM

## 12.1. Overview

The boot ROM ensures the integrity of the device at boot.

An immutable boot sequence (Secure Boot Code) is implemented into a ROM called the secure boot ROM. The secure boot ROM manages system initialization, firmware, data authentication and necessary system configurations.

For secure boot support on the PIC32CX-BZ6, a 64K ROM is dedicated to the secure boot firmware. Keys, unique device ID, storage key and secure boot key required for code authentication are stored as a part of eFuses. On every reset, secure boot firmware authenticates the rest of the program image in the Flash when necessary.

**Note:** The boot ROM will either run on internal FRC oscillator or turn on POSC+PLL and switch to 128 MHz, depending on seccfg[1:0]. In addition, the boot ROM has extra security features, such as device integrity checks, memory/peripherals security attributions, and secure boot, which can be executed before jumping to the Flash in the Secure state when user configured the device in secured mode.

## 12.2. Features

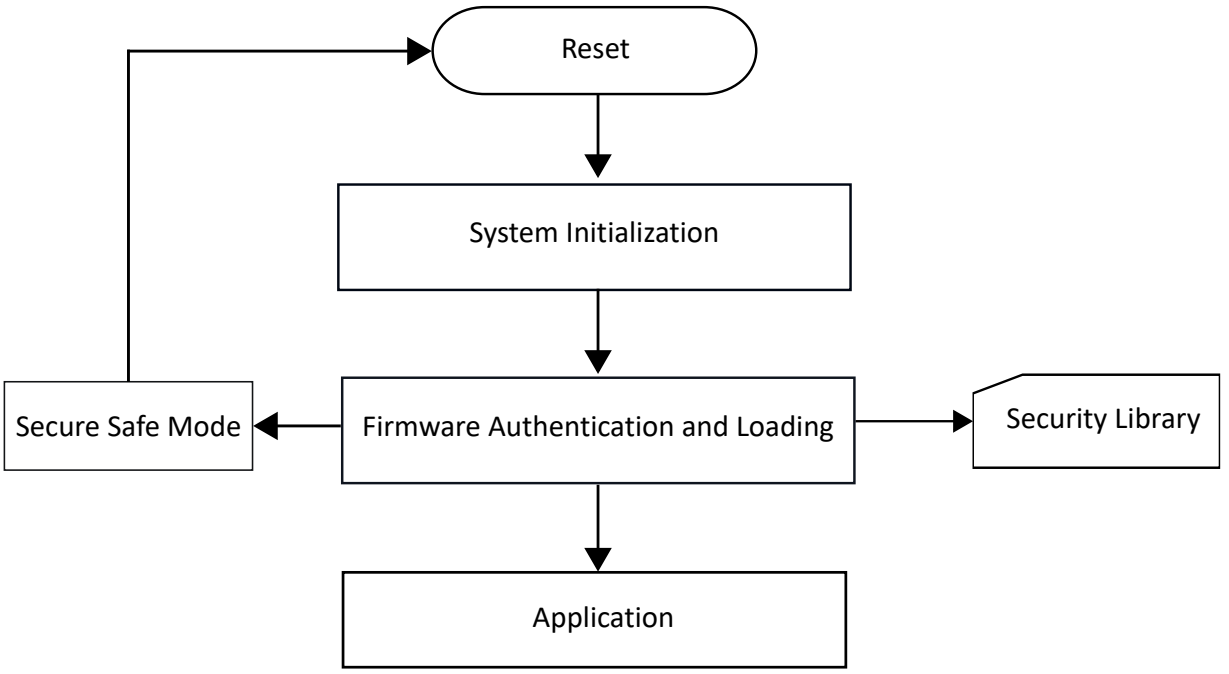
- Immutable Boot Support with ROM
- Firmware Code Authentication
- Support for Immutable Keys
- Microchip or Customer Programmable Secure Boot Key
- Support for Secure Execution Environment
- Supports Anti-Rollback
- Support for Firmware Readable Life Cycle Counter
- Defined System Boot State
- Support Fast Boot to Reduce Boot Time Using Dirty-bits Mechanism

## 12.3. Functional Description


© 2025 Microchip Technology Inc. and its subsidiaries

### 12.3.1. Boot ROM Flow

**Figure 12-1.** Boot ROM Flow




When the device comes out of reset, Boot ROM firmware starts with system initialization. After the completion of system initialization, the bootstrap code checks for a valid firmware image and executes. If the device does not find any valid image, it jumps to Secure Safe mode. For validating the firmware image, the bootstrap code uses the security library depending on the firmware images authentication scheme specified for that firmware.

Firmware programming is handled by Device Firmware Update (DFU). DSU supports the Debug mode. Also, it implements a CoreSight™ Debug ROM that provides device identification, as well as identification of other debug components within the system. See *Device Service Unit (DSU)* from Related Links.

The following options are valid images for a secured and unsecured PIC32CX-BZ6 device:

- An image with valid structure (defined in Firmware Image Format ) and without using any authentication scheme
- An image with a valid structure (defined in Firmware Image Format ) with an authentication scheme supported by the device. For supporting the authentication scheme using the public key cryptography, EFUSE must have a valid SECURE_BOOT_KEY.
- If the device is unsecured (SECURE_BOOT_KEY is invalid or device secured state is not set), the bootstrap code will look for valid images (valid sequence number, header and so on). Root of trust may not be possible in this scenario.
- If the device is secured (with SECURE_BOOT_KEY! = 0), the valid image must always use the `ECDSA p384 + SHA 384 or ECDSA p256 + SHA 256` for authentication and execute only trusted code.

**Related Links**

Device Service Unit (DSU)


© 2025 Microchip Technology Inc. and its subsidiaries

#### 12.3.1.1. DSU Mode Entry

Immediately after resetting, the hardware performs a read operation of the register bits of code protect and Device Secured State bits to determine the Device Security state. Depending on the Security state, the ROM firmware needs to perform specific actions.

#### 12.3.1.2. System Initialization

All the necessary hardware modules that the Boot ROM firmware uses will be initialized – System clock, Interrupts and Jumping to application firmware from Boot ROM code.

### 12.3.2. Firmware Authentication and Loading

The firmware image validation block is the primary block that ensures the integrity/authenticity of the images that are run on a secured device.

#### 12.3.2.1. Firmware Image Format

The firmware image comes with a metadata header, metadata payload and metadata footer that gives the ROM firmware information about location of the firmware image, authentication information, sequence number and more. The user can only execute the application (firmware image) from embedded Flash.

The following table provides details about the contents of each firmware image.

**Table 12-1.** Metadata Storage Structure for 272-Bit Width Flash

| Offset | Byte 0 | Byte 1 | Byte 2 | Byte 3 |
|---|---|---|---|---|
| 0x00 | 0x00 | | | |
| 0x04 | 0x00 | | | |
| 0x08 | 0x00 | | | |
| 0x0C | 0x00 | | | |
| 0x10 | 0x00 | | | |
| 0x14 | 0x00 | | | |
| 0x18 | “MCHP” | | | |
| 0x1C | 0x00 | | | |
| 0x20 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x24 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x28 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x2C | 0xFFFF_FFFF/0x00(1) | | | |
| 0x30 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x34 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x38 | 0xFFFF_FFFF/0x00(1) | | | |
| 0x3C | SEQ_NUM | | | |
| 0x40 | MD_REV (0x03) | CONT_IDX | MD_AUTH_MTHD | MD_AUTH_KEY |
| 0x44 | PL_DEC_MTHD | PL_DEC_KEY | PL_LEN | |
| ... | PAYLOAD... | | | |
| ... | ... | | | |
| ... | MD_SIG | | | |
| “MCHP” | | The indicator that specifies this is a Microchip metadata | | |
| MD_REV | | Reversion of the metadata definition. This field is 0x03 indicating this is V3 | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 12-1.** Metadata Storage Structure for 272-Bit Width Flash (continued)

| Offset | Byte 0 | Byte 1 | Byte 2 | Byte 3 |
|---|---|---|---|---|
| CONT_IDX | | Contain Indicator<br>• 0: Boot Config, Device Cal, or RF Cfg/Cal.<br>• 1: Firmware Image.<br>• 2: Encapsulated Firmware Image<br>• 3: Variable<br>• 4: Stack Data<br>• 5: App Data. Reserved. Application Storage Format should be defined by application developer<br>• 6 - 255: Reserved | | |
| MD_AUTH_MTHD | | Metadata Authentication or Integrity Method:<br>• 0: None.<br>• 1: CRC16.<br>• 2: ECDSA p256 + SHA2-256<br>• 3: ECDSA p384 + SHA2-384<br>• 4: Reserved | | |
| MD_AUTH_KEY | | Metadata Authentication Key Index:<br>• 0: Key #0<br>• 1: Key #1 | | |
| PL_DEC_MTHD | | Payload Decrypt Method:<br>• 0: Plain<br>• 1: AES<br>**Note:** The current implementation always sets `PL_DEC_MTHD` to 0. | | |
| PL_DEC_KEY | | Payload Decrypt Key Index:<br>• 0: Key #0<br>• 1: Key #1<br>• ... | | |
| PL_LEN | | The Length of PAYLOAD:<br>• 0: 0 byte<br>• 1: 1 byte<br>• ... | | |
| PAYLOAD | | Payload Data | | |
| MD_SIG | | The signature of metadata authentication.<br>The signature length is authentication method dependent. | | |

**Note:**

1. To support ECC, these bytes are 0xFFFF_FFFF before SEQ_NUM is programmed; and they will be programmed to 0x00 when programming SEQ_NUM.

**Table 12-2.** Firmware Images Offsets for 2 MB Flash Variant

| | Embedded Flash Variant | |
|---|---|---|
| | Address | Size (bytes) |
| Image 0 metadata | 0x0080_0000 | 512 |
| Image 0 | 0x0080_0200 | Max: 60928 |
| Image 1 Metadata | 0x0080_8000 | 512 |
| Image 1 | 0x0080_8200 | Max: 28160 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 12-2.** Firmware Images Offsets for 2 MB Flash Variant (continued)

| Embedded Flash Variant | | |
|---|---|---|
| | Address | Size (bytes) |
| Image 2 Metadata | 0x0100_0000 | 512 |
| Image 2 | 0x0100_0200 | Max: 2096640 |
| Image 3 Metadata | 0x0110_0000 | 512 |
| Image 3 | 0x0110_0200 | Max: 1048064 |
| Image 4 Metadata | NA | NA |
| Image 4 | NA | NA |
| Image 5 Metadata | NA | NA |
| Image 5 | NA | NA |

**Table 12-3.** Firmware Images Offsets for 1 MB Flash Variant

| Embedded Flash Variant | | |
|---|---|---|
| | Address | Size (bytes) |
| Image 0 metadata | 0x0080_0000 | 512 |
| Image 0 | 0x0080_0200 | Max: 60928 |
| Image 1 Metadata | 0x0080_8000 | 512 |
| Image 1 | 0x0080_8200 | Max: 28160 |
| Image 2 Metadata | 0x0100_0000 | 512 |
| Image 2 | 0x0100_0200 | Max: 1048064 |
| Image 3 Metadata | 0x0108_0000 | 512 |
| Image 3 | 0x0108_0200 | Max: 523776 |
| Image 4 Metadata | NA | NA |
| Image 4 | NA | NA |
| Image 5 Metadata | NA | NA |
| Image 5 | NA | NA |

For an embedded Flash-only device, the ROM firmware checks for a valid image in three memory locations, see Table 12-3.

#### 12.3.2.2. Firmware Image Validation – Secured Mode

The following are the main blocks in the process of image validation:

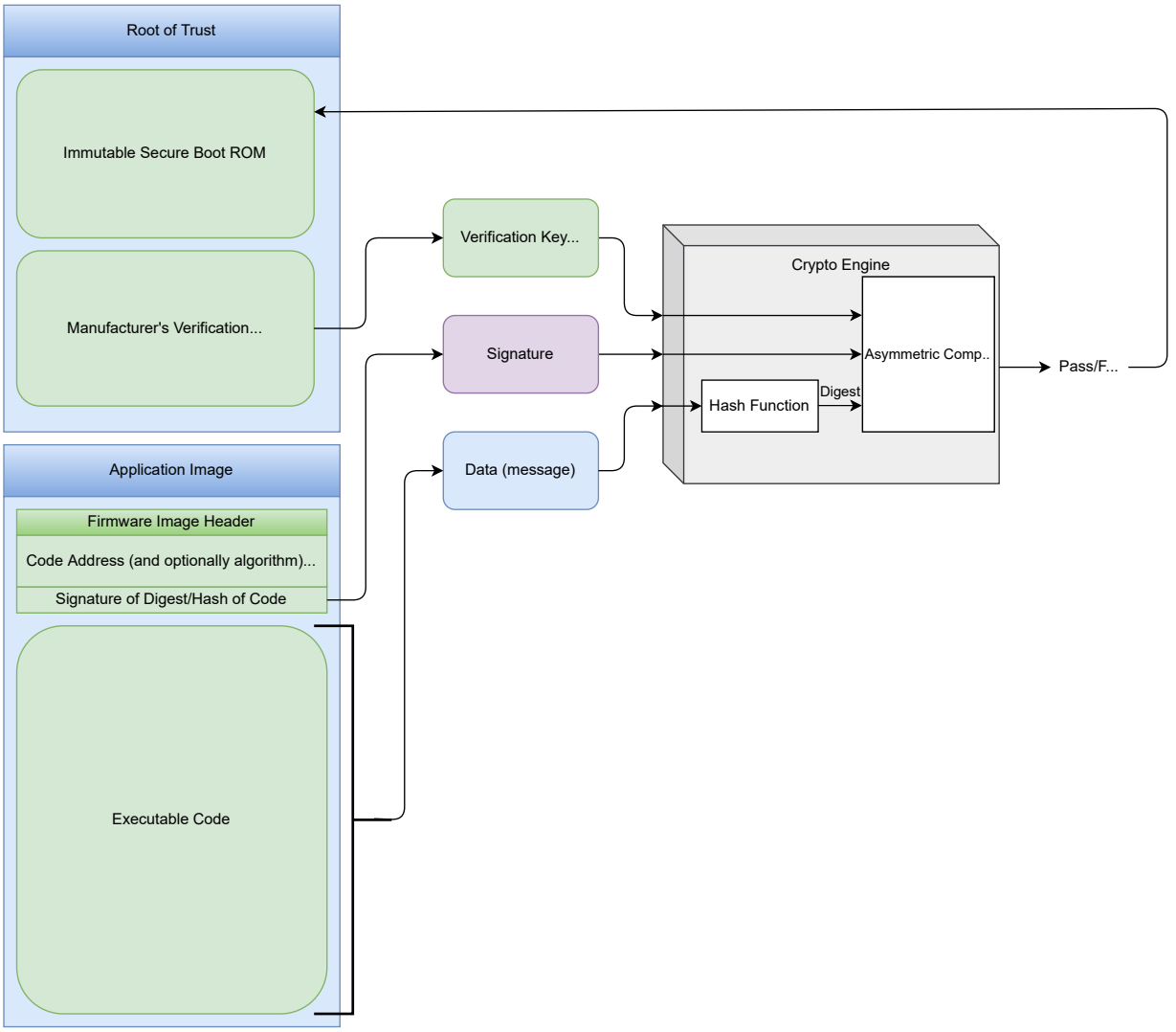
1. READ_CFG – Read device configuration (Device id, secure boot key, anti-rollback counter, life cycle counter). Secure elements and device id information can be accessed from eFuse memory.
   a. Device id – Device identifier for each device
   b. Secure boot key – Bootstrap code evaluates only images that use the secure boot key.
   c. Anti-rollback counter – This counter is used to keep track of the firmware version. The values are written by firmware in the eFuse region.
   d. Life cycle counter – This counter is used by firmware to track the changes in the life cycle of the device.
2. LOOKUP_METADATA – Search for valid firmware image metadata with the lowest sequence number.
3. AUTHENTICATE_IMAGE – Authenticate the image based on the security configuration of the device (eFuse) and as indicated by its metadata. The firmware image header dictates the authentication scheme. Secure boot is done on successful authentication of the firmware.


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:** In order to reduce the boot time while the system is waking up from deep sleep mode, the fast boot is introduced. A fast boot attempt will be tried first on the image. A hash code is generated from the firmware image, then encrypted with AES-ECB. The encrypted hash will be compared to that saved in the last page of boot Flash. If comparison result is positive, then Application Transition process starts. Otherwise, full authentication will be performed.

4. To support Fast Boot, Dirty-bits are introduced, to further help the secure boot code understands a device's activities. The Dirty-bits is managed by the secure boot code, and it gets set by the Flash controller once there is a programming or erasing event. The secure boot code clears the Dirty-bits during system boot once a firmware image passes the full firmware authentication.
5. RUN_APPLICATION – If the valid image is in embedded flash and is available in its destination address space, jump to the DST_ADDR as indicated by its metadata. If the device does not find any valid image it jumps to Secure safe mode.
6. SECURE_SAFE_MODE – Secure Safe mode is a `while (1) loop` in the firmware, where the device is waiting for a valid application image to execute using the events like Power-on Reset or a programming tool to put the device into the image Lookup state to validate it.

**Figure 12-2.** Secure Boot Firmware Validation





© 2025 Microchip Technology Inc. and its subsidiaries

#### 12.3.2.3. Firmware Image Validation – Unsecured Mode

In the Unsecured mode, the device is not checked for the authentication scheme that needs SECURE_BOOT_KEY.

If the device is unsecured (SECURE_BOOT_KEY is invalid or device secured state is not set), the bootstrap code looks for valid images (valid sequence number, header and more). Root of trust cannot be possible in this scenario.

#### 12.3.2.4. Firmware Loading

Firmware metadata can be stored in embedded Flash, RAMs or external storage. The firmware loading is done by the secondary boot loader. The firmware loading depends on the firmware metadata.

The secure boot code loads the primary firmware image into the SRAMs and authenticates it on the SRAM directly. After loading the firmware from the source, it relinquishes control of it.

### 12.3.3. Device Protection

Code Protect and Device secured state bits to determine the device security state and accessibility. The ROM firmware accessibility is defined based on the device state.

**Table 12-4.** Device Protection

| | No Code Protect | | | | Code Protect | | | |
|---|---|---|---|---|---|---|---|---|
| | Unsecured Device | | Secured Device | | Unsecured Device | | Secured Device | |
| | Boot Key = 0 | Boot Key != 0 | Boot Key = 0 | Boot Key != 0 | Boot Key = 0 | Boot Key ! = 0 | Boot Key = 0 | Boot Key != 0 |
| Usage | Device with no security/ code protect | Debug of secured device | Device with no security, lock program interfaces | Secured device | Device with no security | No intended usage | Device with no security, lock program interfaces | Secured device/code |
| Firmware Authentication | No | Optional | No | Mandatory | No | Optional | No | Mandatory |
| Debug Support | Open | Open | Secure debug entry required | Secure debug entry required | Secure debug entry required | Secure debug entry required | Secure debug entry required | Secure debug entry required |
| Serial Programming | All resources | All resources | None | Flash controller RoT | Chip erase only | Chip erase only | Chip erase only | Chip erase only |
| DFU | Present | Present | Locked | Present | Present | Present | Locked | Present |
| OTA | Present | Present | Present | Present | Present | Present | Present | Present |


© 2025 Microchip Technology Inc. and its subsidiaries

## 12.4. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00<br>...<br>0x0BFF | Reserved | | | | | | | | | |
| 0x0C00 | SEC_BOOT | 7:0 | | | | | | | | |
| | | 15:8 | BOOT_STATUS(2)[7:0] | | | | | | | |
| | | 23:16 | | | | | | | SEC_BOOT_DONE(1)[1:0] | |
| | | 31:24 | | | | | | | | |

## 12.5. Register Description


© 2025 Microchip Technology Inc. and its subsidiaries

### 12.5.1. SEC_BOOT – Secure Boot Register

**Name:** SEC_BOOT
**Offset:** 0xC00
**Reset:** 0x00000000
**Property:** -

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SEC_BOOT_DONE(1)[1:0] | |
| Access | | | | | | | R/S | R/S |
| Reset | | | | | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_STATUS(2)[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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bits 17:16 – SEC_BOOT_DONE(1)[1:0]** Bits to Indicate that Secure Boot is Done
FW can only set these bits. FW can never clear these bits.
These bits drive sec_boot_done output.

| Value | Description |
|---|---|
| 00 | Secure boot is done |
| 01 | Secure boot is not done |
| 10 | Secure boot is not done |
| 11 | Secure boot is done |

**Bits 15:8 – BOOT_STATUS(2)[7:0]** Firmware Managed Bits to Indicate Secure Boot Status
The 8-bit code is written to this field to indicate the secure boot status - like authentication success or failure or any other kind of indication. The code and its corresponding message is to be determined by the firmware. Refer to Application Transition for the Boot Status firmware definition.
This field can be reset only on POR reset.
This field drives boot_status[7:0] output from the macro.

**Notes:**

1. These register bits are reset on **any** device reset.
2. These register bits are reset **only on** POR reset.

## 12.6. Application Transition

After loading and authenticating a valid image, the ROM firmware must perform several additional setup steps before transitioning to the application.

1. Remap interrupt and exception vectors to the application space addresses.
2. Set the SEC_BOOT_DONE configuration register bits.


© 2025 Microchip Technology Inc. and its subsidiaries

3. Jump to application code at FW_IMG_DST_ADDR.

The ROM firmware programs the SECBOOTDONE bits to 0b11 when the firmware image is validated and before switching to the application. The BOOTSTATUS bits will be programmed with the firmware at each step and will indicate firmware status while trying to load the firmware.

**Table 12-5.** BOOT_STATUS[4:2] – Primary Status Codes

| Code | Value | Description |
|---|---|---|
| **DSU Mode** | | |
| TMODE_PE | 0x02 | System is in Parallel Execution Mode |
| TMOD_LOCK_DOWN | 0x03 | System is in Test Lock Down Mode; check for secondary status code. |
| **Secure Boot** | | |
| SYS_FW_AUTH | 0x04 | System is currently authenticating the firmware; do nothing. |
| SYS_FW_LOAD | 0x05 | System is loading or programming the firmware image; do nothing. |
| SYS_APP_MODE | 0x06 | System is in application mode; a firmware image is loaded and is running. |
| SYS_LOCK_DOWN | 0x07 | System is Locked down; check secondary status. |

**Table 12-6.** BOOT_STATUS[1:0] – Secondary Status Codes

| Code | Value | Description |
|---|---|---|
| **TMOD_LOCK_DOWN** | | |
| TMOD_8_10 | 0x00 | System is in TMOD 8 or TMOD 10. Proceed with Factory reset. |
| ERR_BAD_TMOD | 0x01 | System does not support this test mode; reset the device. |
| **SYS_LOCK_DOWN** | | |
| SYS_LD_ERR_FW_AUTH_FAIL | 0x00 | Reset the device and program a valid firmware image. |
| SYS_LD_ERR_NO_FW_IMG | 0x01 | Reset the device and program a valid firmware image. |


© 2025 Microchip Technology Inc. and its subsidiaries

# 13. eFuse Controller

## 13.1. Overview

The PIC32CX-BZ6 devices contain a programmable eFuse memory controller, which is used to store various parameters like a user programmable boot key, anti-rollback counter, life cycle counter and secure configuration details.

The eFuse controller is required to perform two modes of operation:

- Program Mode - Programming of eFuse panel
- Read Mode - Reading of eFuse panel

## 13.2. Hardware Mode

The Hardware (HW) mode is essentially using a state machine (FSM) within the eFuse Controller to perform the program and read operations. All control of the eFuse panel is done via the eFuse controller.

The user can program using the following registers/register bits:

- EFUSE_RWDATA.ADDR
- EFUSE_RWDATA.DATA
- EFUSE_CON.PGM_1BIT
- EFUSE_CON.EN_PGM
- EFUSE_CON.PGM_MODE

For fuse reading, the user can initiate fuse loading by the following registers/register bits:

- EFUSE_RWDATA.ADDR
- EFUSE_CON.EN_LD
- EFUSE_CON.EN_LD_ALL

By default, the eFuse controller starts in hardware mode so that it is automatically initialized. Read of the fuses can be performed after a device reset.

**Note:** To allow the use of the eFuse contents in determining the device configuration, the eFuse controller will automatically perform an initial read of the fuses after a POR (Power-On Reset) reset.

The EFUSE_RWDATA register is used to program data into the fuse panels, while the eFuse read operation leads to writing into System Registers.

## 13.3. eFuse Programming

In the eFuse programming operations, the eFuse controller can perform bit programming or byte programming determined by EFUS_CON.PGM_1BIT. In PIC32CX-BZ6, the programming voltage gets supplied by the eFuse LDO.

### 13.3.1. eFuse Programming Sequence

For eFuse programming operations, the eFuse controller will program 1-bit or 8 bits together at a time.

1. Write EFUSE_CON.PGM_MODE = `0`.
2. Write EFUSE_CON.EN_LD = 0, and EFUSE_CON.EN_LD_ALL = `0`.
3. Write EFUSE_CON.EN_OTP_LDO = `1`. This enables the PMU OTP LDO output to 1.5V.
4. Write the EFUSE_RWDATA register with the offset address of eFuse and the data to be written.
   a. For single bit write operation: write eFuse offset address on EFUSE_RWDATA.ADDR[11:0] and data on EFUSE_RWDATA.DATA[0].


© 2025 Microchip Technology Inc. and its subsidiaries

b. For 8-bit write operation: write eFuse offset address on EFUSE_RWDATA.ADDR[11:3] and data on EFUSE_RWDATA.DATA[7:0].
The above options are controlled by EFUSE_CON. PGM_1BIT register bit as in Step 5.2.

5. All the below steps to be done in a single step:
   a. Set EFUSE_CON.PGM_MODE register bit.
   b. Set EFUSE_CON. PGM_1BIT register bit for 1-bit write operation. Clear the bit for 8-bit write operation.
   c. Set EFUSE_CON.EN_PGM register bit.
6. The eFuse controller clears the EFUSE_CON.EN_PGM register bit when the eFuse programming is complete.
7. Repeat the steps 4 to 5 until all the values are programmed.
8. Write EFUSE_CON.EN_OTP_LDO = 0 to switch off PMU OTP LDO.

**Note:** Lifecycle counter and anti-rollback programming values to be programed in 1- bit operation mode.

### 13.3.2. eFuse Reading Sequence

The read from the eFuse panel happens at boot time and the holding registers are loaded with the eFuse values. Therefore, the memory map of the eFuse is in 1:1 correspondence with the memory map of the holding registers. If required to verify the values in eFuse, reset the device after the eFuse programming sequence is done. The modified eFuse values are read from the eFuse and loaded on holding registers at boot-up. Read Holding Registers (similar to the process for the peripheral registers), then verify these registers.

## 13.4. eFuse Auto-Loading

In the time of POR, the eFuse controller loads the eFuse values from the eFuse panel into the holding registers.

### 13.4.1. Holding Registers

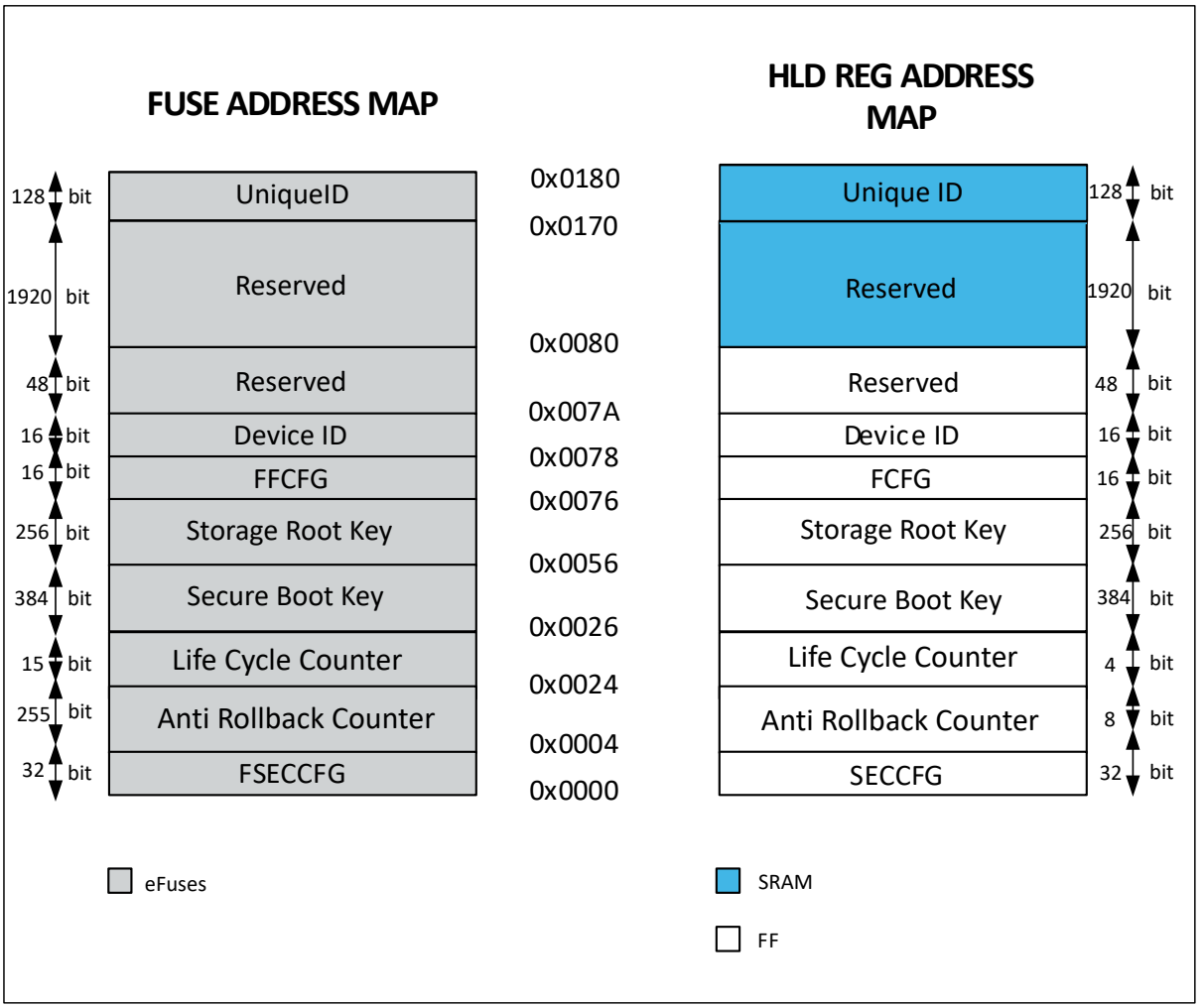
The RoT Holding Registers store eFuse values that auto-loaded on a POR or when the EN_LD_ALL/ EN_LD register bits are set to perform a manual load operation.

The eFuse blocks are used to implement the following holding registers:

- Unique ID
- 16-bit device ID
- 256-bit Storage Root Key (SRK)
- 384-bit public boot key
- 15-bit lifecycle counter
- 255-bit anti-rollback counter
- Security/RoT configuration


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 13-1.** Memory Map of the Fuse Panels




The holding registers are governed by their read and write access permissions. The holding registers are loaded from their corresponding fuses if any of the following conditions are met:

- After a POR during the auto-load state
- Explicit FW setting of EN_LD_ALL or EN_LD bits
- After fuse programming if EN_LD_ALL or EN_LD bits are set in the same cycle as EN_PGM being set

The above scheme minimizes the power and boot time associated with an eFuse load operation.

Holding registers can also be written by firmware dictated by the write locks as specified in the following table.

**Table 13-1.** Device Elements Provisioning

| Secure Element | Size in Bits | Provisioner | To Program Lock |
|---|---|---|---|
| Unique ID | 128 | MCHP | Yes |
| Device ID | 16 | MCHP | Yes |
| SECCFG | 32 | Code signer (MCHP, 3rd Parties, Customers) | No |
| Anti Rollback Counter | 255 | Code signer (MCHP, 3rd Parties, Customers) | No |
| Life Cycle Counter | 15 | Code signer (MCHP, 3rd Parties, Customers) | No |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 13-1.** Device Elements Provisioning (continued)

| Secure Element | Size in Bits | Provisioner | To Program Lock |
|---|---|---|---|
| Secure Boot Key | 384 | Code signer (MCHP, 3rd Parties, Customers) | Yes |
| Storage Root Key | 256 | MCHP | Yes |

All the holding registers are reset by POR. All the regular SFRs in the macro are reset by a system reset signal unless specifically mentioned otherwise.

## 13.5. Security Keys

### 13.5.1. Secure Boot Key

Secure boot key is the public key part of the private/public key pair that is used for code authentication. It can be 256 bits or 384 bits depends on the authentication algorithm. If the secure boot key has all values set to '0', it is considered invalid and no secure boot key is programmed.

### 13.5.2. Storage Root Key

The storage root key is also called a local storage key. This is a 256-bit (symmetric crypto) secret encryption key used to store important data secretly. It is a programmable key that is randomly generated for each device in the factory.

## 13.6. Counters

There are two thermometer counters in this PIC32CX-BZ6 device as follows:

- Lifecycle counter
- Anti-rollback counter

### 13.6.1. Life Cycle Counter

This is a 15-bit thermometer counter that is decoded into a 4-bit value. This counter is used by firmware to track the changes in the lifecycle of the device. These fuses are programmable by firmware. The holding registers for this counter hold the decoded value from the fuses.

The possible states are as follows:

- BLANK
- FACTORY
- CUSTOMER

An example of decoding the thermometer counter can be seen in Figure 13-2.

### 13.6.2. Anti-Rollback Counter

This is a 256-bit thermometer counter that is decoded into an 8-bit value. This counter is used to keep track of the firmware version. These fuses are programmable by firmware. The holding registers for this counter hold the decoded value from the fuses. The holding register is not writable and is only readable by firmware.

The firmware image major revision number is greater than or equal to the value of the anti-rollback counter. Also, the counter is used to indicate the minimum acceptable firmware revision.

The following figure provides an example for the decode of the thermometer counter.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 13-2.** 4-bit Thermometer Counter

| Counter | Decode |
|---|---|
| 0000 | 0 |
| 0001 | 1 |
| 0011 | 2 |
| 0111 | 3 |
| 1111 | 4 |
| 0010 | 2 |
| 0100 | 3 |
| 0101 | 3 |
| 0110 | 3 |
| 1000 | 4 |
| 1001 | 4 |
| 1010 | 4 |
| 1110 | 4 |
| 1011 | 4 |
| 1100 | 4 |
| 1101 | 4 |
| 1110 | 4 |

**Notes:**

- Rows in green indicate the correct flow of the increment steps of the counters.
- However, if the user programs an incorrect value to the counter, the decode results are shown in rows of orange.


© 2025 Microchip Technology Inc. and its subsidiaries

## 13.7. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00<br>...<br>0x0C03 | Reserved | | | | | | | | | |
| 0x0C04 | EFUSE_RWDATA | 7:0 | DATA[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ADDR[7:0] | | | | | | | |
| | | 31:24 | | | | | ADDR[11:8] | | | |
| 0x0C08 | EFUSE_CON | 7:0 | EN_PGM | EN_LD_ALL | EN_LD | | | | PGM_MODE | PGM_1BIT |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | EN_OTP_LDO |
| | | 31:24 | | | | | | | | |

## 13.8. Register Description


© 2025 Microchip Technology Inc. and its subsidiaries

### 13.8.1. EFUSE_CON – eFuse Configuration Register

**Name:** EFUSE_CON
**Offset:** 0xC08
**Reset:** 0x00000000
**Property:** -

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | EN_OTP_LDO |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

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

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EN_PGM | EN_LD_ALL | EN_LD | | | | PGM_MODE | PGM_1BIT |
| Access | R/W/HC | R/W/HC | R/W/HC | | | | R/W | R/W |
| Reset | 0 | 0 | 0 | | | | 0 | 0 |

**Bit 16 – EN_OTP_LDO** Enable OTP LDO

The separate OTP LDO gets automatically enabled for programming operations. If longer settling times are desired or many values need to be programmed consecutively, the OTP LDO can be enabled manually.

**Note:** While the OTP LDO is enabled manually, it is not possible to start any loading operation. To load newly programmed values to the holding registers, EN_OTP_LDO needs to be set to 0 first, before starting the load using EN_LD_ALL/EN_LD.

| Value | Description |
|---|---|
| 1 | Enable OTP LDO |
| 0 | Disable OTP LDO |

**Bit 7 – EN_PGM** eFuse Programming Start Bit

**Notes:**

- This bit has no effect when PGM_MODE = 0.
- This bit will be automatically cleared by hardware when the programming operation is complete.
- When the EN_PGM bit is set, the EFUSE_CON & EFUSE_RWDATA registers may not be changed until the EN_PGM is cleared.
- EN_PGM, PGM_MODE and PGM_1BIT can all be written in the same cycle.

| Value | Description |
|---|---|
| 1 | Start eFuse Programming operation |
| 0 | eFuse Programming operation has completed |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 6 – EN_LD_ALL** eFuse Panel Read Start Bit for Loading into the Holding Registers

**Notes:**

- This bit will be automatically cleared by hardware when the read operation is complete.
- This bit resets to '1', so that an initial auto-load of the fuse values is performed after a device reset. This allows the value of the fuse to be used in determining functionality like device pinout, memory configurations, etc. When this auto-load is completed, the hardware clears this register bit.
- The user can set this bit after programming the fuse for the fuses to be loaded into the holding register. After this bit is cleared, the user can read the fuses (holding registers) to verify the programmed value.
- When the EN_LD_ALL bit is set, the EFUSE_CON & EFUSE_RWDATA registers cannot be changed until EN_LD_ALL is cleared.

| Value | Description |
|---|---|
| 1 | Start eFuse Read operation for entire eFuse panel |
| 0 | eFuse Read operation has completed |

**Bit 5 – EN_LD** eFuse Word Read Start Bit for Loading the Fuse Byte Pointed by ADDR Field into the Holding Register

**Notes:**

- This bit will be automatically cleared by hardware when the read operation is complete.
- When

  the EN_LD bit is set, the EFUSE_CON & EFUSE_RWDATA registers cannot be changed until EN_LD is cleared.

| Value | Description |
|---|---|
| 1 | Start eFuse Read operation for specified eFuse word as addressed in EFUSE_RWDATA register |
| 0 | eFuse Read operation has completed |

**Bit 1 – PGM_MODE** eFuse Programming Mode Enable Bit

**Note:** EN_PGM, PGM_MODE and PGM_1BIT can all be written in the same cycle.

| Value | Description |
|---|---|
| 1 | eFuse Programming enabled |
| 0 | eFuse Programming disabled |

**Bit 0 – PGM_1BIT** eFuse CTRL to Program 1 Bit at a Time. Valid only when EN_PGM is Set

**Note:** EN_PGM, PGM_MODE and PGM_1BIT can all be written in the same cycle.

| Value | Description |
|---|---|
| 1 | eFuse controller will program EFUSE_RWDATA.DATA[0] to the address in EFUSE_RWDATA.ADDR[11:0] |
| 0 | eFuse controller will program EFUSE_RWDATA.DATA[7:0] to the address in EFUSE_RWDATA.ADDR[11:3] |

**Note:**

1. EN_PGM, PGM_MODE, PGM_1BIT, EN_LD_ALL, EN_LD - All the bits can be written in the same cycle (as per requirement). Combinational setting of configuration bits for programming and read operation is not allowed when OTP LDO is enabled manually with EN_OTP_LDO.


© 2025 Microchip Technology Inc. and its subsidiaries

### 13.8.2. EFUSE_RWDATA – eFuse Read/Write Data Register

**Name:** EFUSE_RWDATA
**Offset:** 0xC04
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | 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 |

| 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 27:16 – ADDR[11:0]** eFuse Program/Read Address
When EN_PGM = 1 & PGM_1bit = 0, **ADDR[11:3]** is used as eFuse Program Address.
When EN_PGM = 1 & PGM_1bit = 1, **ADDR[11:0]** is used as eFuse Program Address.
When EN_LD_ALL = 0 & EN_LD = 1, **ADDR[11:3]** is used as eFuse Sense to Read Address.
When EN_LD_ALL = 1, **ADDR[11:0]** is not used.

**Bits 7:0 – DATA[7:0]** eFuse Program (Write) Data
When EN_PGM = 1 & PGM_1bit = 0, **DATA[7:0]** is used as eFuse Program Data.
When EN_PGM = 1 & PGM_1bit = 1, **DATA[0]** is used as eFuse Program Data.
When EN_PGM = 0, **DATA[7:0]** is not used.


© 2025 Microchip Technology Inc. and its subsidiaries

# 14. Security Features

## 14.1. Overview

This device includes a set of components that can support a high level of system security to the device.

## 14.2. Features

The key features of the module are:

- High-Performance Cryptographic Accelerators
- Authentication, Using Public Key Algorithms
- Integrity, Using Secure Hash Algorithms
- Privacy, Using Symmetric Encryption (AES)
- Entropy, Using a True Random Number Generator
- Public Key, RSA Encryption, and Decryption with Key Sizes of 1024 Bits, 2048 Bits, 3072 Bits and 4096 Bits
- ECC, Crypto Element with Protected Hardware-Based Key Storage
- Passing the Storage Root Key from the Root of Trust Macro into the Security Module
- Integrated Scatter-Gather DMA
- AHB Host and Client Interfaces

## 14.3. Reference Documentation

- American National Standards Institute, "*Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography*", X9.63-2011, December 2011
- American National Standards Institute, "*Public Key Cryptography for the Financial Servic3es Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*", X9.62-2005, November 2005
- International Standards Organization, "*Information Technology - Security techniques - Cryptographic techniques based on elliptic curves -- Part 2: Digital Signatures*", ISO/IEC 15946-2, December 2002
- National Institute of Standards and Technology, "*Secure Hash Standard (SHS)*", FIPS Pub 180-4, March 2012
- National Institute of Standards and Technology, "*Digital Signature Standard (DSS)*", FIPS Pub 186-3, June 2009
- National Institute of Standards and Technology, "*Advanced Encryption Standard (AES)*", FIPS Pub 197, November 2001
- National Institute of Standards and Technology, "*Recommendation for Block Cipher Modes of Operation*", FIPS SP 800-38A, 2001
- RSA Laboratories, "*PKCS#1 v2.2: RSA Cryptography Standard*", October 2012

## 14.4. Interface

This block is designed to be accessed internally via a registered host interface.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 14-1.** I/O Block Diagram




## 14.5. Description

The security hardware incorporates the following functions:

### 14.5.1. Symmetric Encryption/Decryption

Standard AES encryption and decryption with key sizes of 128 bits, 192 bits and 256 bits are supported with a hardware accelerator. AES modes that can be configured include Electronic Code Block (ECB).

### 14.5.2. Cryptographic Hashing

Standard SHA hash algorithms, including SHA-256 and SHA-384, are supported by hardware.

### 14.5.3. Public Key Cryptographic Engine

A large variety of public key algorithms are supported directly in hardware. These include:

- RSA encryption and decryption with key sizes of 1024 bits and 2048 bits
- Elliptic Curve point multiply with all standard NIST curves using either binary fields or prime fields
- Elliptic Curve point multiply with Curve25519
- The Elliptic Curve Digital Signature Algorithm (ECDSA) using all supported NIST curves
- The Elliptic Curve Korean Certificate-based Digital Signature Algorithm (EC-KCDSA) using all supported NIST curves
- The Edwards-curve Digital Signature Algorithm (EdDSA) using Curve25519

### 14.5.4. True Random Number Generator

A true Random Number Generator includes a module provided for the pre-calculation of random bits. This block has a Health Check function included with it.

### 14.5.5. Cryptographic API

The Boot ROM includes an API for direct software access to cryptographic functions. API functions for Hashing and AES include a DMA interface so the operations can function on large blocks of SRAM with a single call.


© 2025 Microchip Technology Inc. and its subsidiaries

## 14.6. Register Summary

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CRYPTOCON | 7:0 | | RUNSTDBY | | | | | ENABLE | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

## 14.7. Register Description

There are no registers directly accessible to the application in this block. Users must use the APIs to use this block. For the list of APIs, see *Secure Boot ROM* from Related Links. Apart from the crypto module, enable and disable the register.

**Related Links**

Secure Boot ROM


© 2025 Microchip Technology Inc. and its subsidiaries

### 14.7.1. CRYPTOCON – Crypto Control Register

**Name:** CRYPTOCON
**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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

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

| 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 bit

| Value | Description |
|---|---|
| 1 | If enabled, the clock to security module remains enabled in Idle mode |
| 0 | Security module clock disabled in Idle mode |

**Bit 1 – ENABLE** Module Enable Bit

| Value | Description |
|---|---|
| 1 | Crypto module is enabled. Security module clock is enabled |
| 0 | Crypto module is disabled. Security module clock is disabled |

**Bit 0 – SWRST** Software Reset

Write '1' to reset registers and internal state. Writing '0' has no effect. SWRST stays high until the reset completes.

| Value | Description |
|---|---|
| 1 | Reset registers and internal state |
| 0 | No effect |


© 2025 Microchip Technology Inc. and its subsidiaries

# 15. Flash Controller (FC)

## 15.1. Overview

The PIC32CX-BZ6 devices contain a single bank of Flash memory with their Program Flash Memory (PFM) partition and Boot Flash Memory (BFM) partition for storing user code or non-volatile data. The user can use the Flash controller to access the Flash memory. For commands and configuration of the Flash controller, use the peripheral bus interface.

## 15.2. Features

### Flash Controller

1. PB-Bridge-D Interface that Provides Access to the Flash Controller Registers
2. AHB Initiator for Bus Hosted Reads the Row Programming Data from SRAM
3. Write Protect for Program Flash (PFM)
   a. Single page protection resolution
   b. Protect < Address
   c. Protect ≥ Address
4. Individual Page Write Protection for Boot Flash (BFM)
5. Error Correction Code Support
6. Supports Chip and Page Erase
7. Supports Double Word, Quad Double Word and Row Program Options
8. Supports Flash Erase/Retry to Increase Retention and Endurance

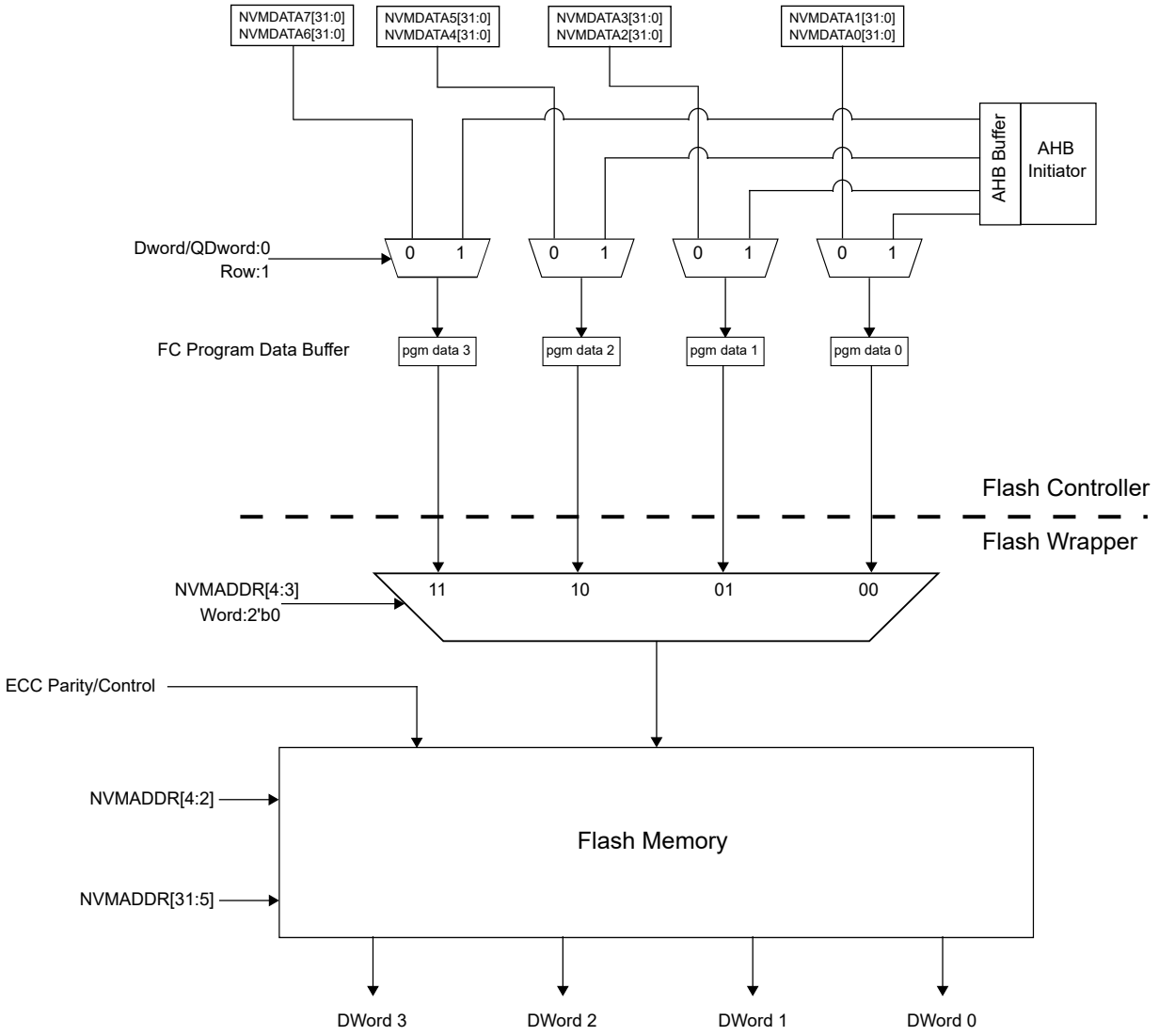
### Flash Memory

1. 256-Bit Wide Flash Memory Access
2. 4 Kbytes Page Size
3. Row Size is 1 KB (256 Word)
4. Flash-Based One-Time-Programmable (OTP) Page. The Flash controller allows the Flash memory to be accessed through the following methods:
   a. Run-Time Self-Programming (RTSP)
   b. Serial Wire Debug (SWD) programming using DSU
5. Flash Word is 256 Bits


© 2025 Microchip Technology Inc. and its subsidiaries

## 15.3. Functional Block Diagram

**Figure 15-1.** Flash Memory Block Diagram




## 15.4. Product Dependencies

Not applicable.

### 15.4.1. I/O Lines

Not applicable.

### 15.4.2. Power Management

The Flash Controller does not operate in Power-saving/Low-power/Sleep modes. If a WAIT instruction is encountered when programming, the CPU stops execution (stall), waits for the programming operation to complete, then enters the Power-saving/Low-power mode.

### 15.4.3. Clocks

AHB (SYS_CLK) initiator for bus manager reads of row programming data from system FlexRAM. Use the PB1_CLK bus clock for control register access.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.4.4. DMA

Not applicable.

### 15.4.5. Interrupts

The interrupt request line is connected to the interrupt controller. Using the Flash controller interrupt(s) requires the NVIC interrupt controller to be configured first.

### 15.4.6. Events

Not applicable.

### 15.4.7. Debug Operation

Programming operations continues to completion if the processor execution is halted in Debug mode.

### 15.4.8. Register Access Protection

Not applicable.

### 15.4.9. Analog Connections

Not applicable.

## 15.5. Flash Memory Addressing

Flash memory addressing uses physical addresses only. For more information on addressing, see *Product Memory Mapping Overview* from Related Links.

**Related Links**

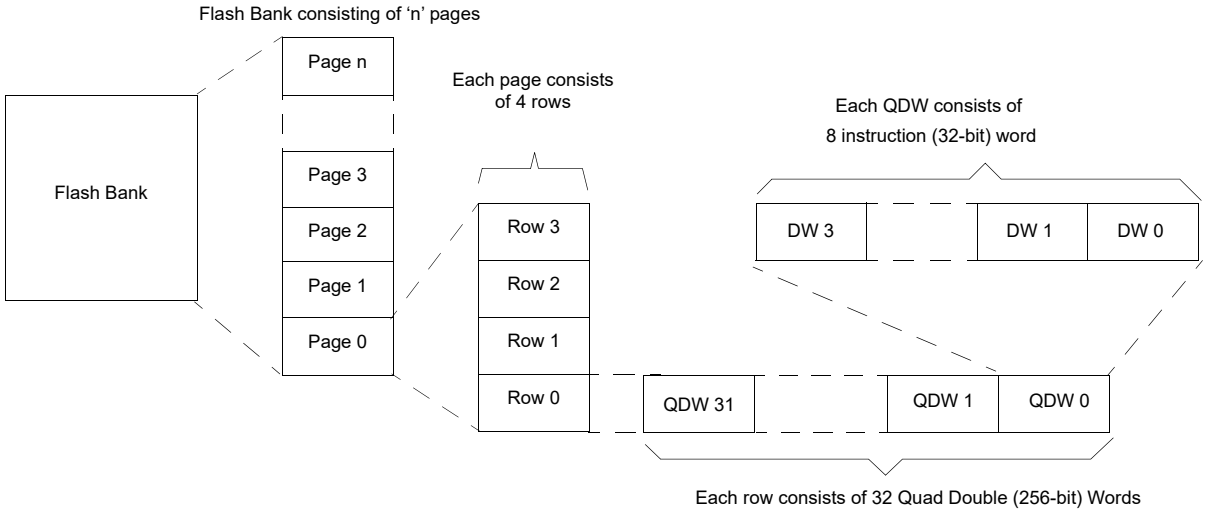
Product Memory Mapping Overview

## 15.6. Memory Configuration

### 15.6.1. Flash Memory Construction

Flash memory is divided into pages. A page is the smallest unit of memory that can be erased at one time. Each page of memory is segmented into four rows. A row is the largest unit of memory that can be programmed at one time. A row consists of 32 Quad Double Words (QDW). Each Quad Double Word consists of four instruction Double Words (DW). Flash memory can be programmed in rows, Quad Double Word (256-bit) or DWord (64-bit) units.

**Figure 15-2.** Flash Construction





© 2025 Microchip Technology Inc. and its subsidiaries

### 15.6.2. Flash Memory Organization

The device Flash memory is divided into two logical Flash partitions:

1. Main Program Flash Memory (PFM)
2. Boot/Configuration Flash Memory (BFM)
   a. Boot Flash
   b. Device/Boot Configuration – Device and boot configuration data
   c. OTP – User system calibration data

The following table provides details about each Flash section having different protection status.

**Table 15-1.** Protection Status

| Flash Partition | Memory Region | Write Protection | Erase Protection | Chip Erase through DSU |
|---|---|---|---|---|
| BFM | Boot Flash (16 pages) | Yes.<br>Page-wise Configurable | Yes.<br>Page-wise Configurable | Erased |
| | Device/Boot Configuration (1 page) | Yes.<br>Configurable | Yes.<br>Configurable | Erased |
| | One Time Programmable (OTP) (1 page) | Yes.<br>Configurable | Always Erase protected.<br>Can not be erased | Not Erased |
| PFM | Program Flash | Yes.<br>Configurable | Yes.<br>Configurable | Erased |

## 15.7. Boot Flash Memory (BFM) Partitions

### 15.7.1. BFM Write Protection

Pages in the BFM regions can be protected individually using bits in the NVMLBWP register. LBWP[15:0] are associated with Boot Flash pages 0 to 15. LBWP[16] is associated with the Device/Boot Configuration page, and LBWP[17] is associated with the One Time Programmable (OTP) page.

At Reset, all pages are in a write-protected state and must be disabled prior to performing any programming operations on the BFM regions. There is also an unlock bit, ULOCK(NVMLBWP[31]), that is set at Reset and can be cleared by the user software. When cleared, changes to write protection for that region can no longer be made. When cleared, the ULOCK bit can only be set by a Reset.

The user can only change the NVMLBWP write-protect register by following the unlock sequence. See *NVMKEY Register Unlocking Sequence* from Related Links.

**Related Links**

NVMKEY Register Unlocking Sequence

## 15.8. Program Flash Memory (PFM) Partitions

### 15.8.1. PFM Write Protection

Write protection for the PFM region is implemented by pages and defined by the NVMPWPLT and NVMPWPGTE registers. The NVMPWP* registers define an area within the program space (PFM) that is write-protected. This write-protected address resolves to Flash page boundaries; therefore, the user can ignore the 12 LSBs for a 4 KB page Flash of any address written to the NVMPWP* registers. The size of the Flash determines the width of each NVMPWP* address register. Use the NVMPWPLT register to set the Program Flash pages lower than the provided address as write-protected. Use the NVMPWPLT register to set the Program Flash pages greater than or equal to the provided address as write-protected. Therefore, a value of all `0`s in the NVMPWPLT register and all `1`s in the NVMPWPGTE register results in no region of Flash being write-protected (default state at Reset).


© 2025 Microchip Technology Inc. and its subsidiaries

There is also an unlock bit, ULOCK (NVMPWPLT [31] and NVMPWPGTE[31]), that is set at Reset and can be cleared by the user software. After clearing, the user can no longer make changes to the write-protection of the PFM, including the ULOCK bit. The NVMPWPLT and NVMPWPGTE write-protected register can only be changed by following the unlock sequence. See *NVMKEY Register Unlocking Sequence* from Related Links.

**Related Links**

NVMKEY Register Unlocking Sequence

## 15.9. Error Correcting Code (ECC) and Flash Programming

The PIC32CX-BZ6 devices incorporate Error Correcting Code (ECC) features that detect and correct errors resulting in extended Flash memory life. For more details on this feature, see *Prefetch Cache* from Related Links.

ECC is implemented in 256-bit Quad Flash Words or 64-bit DWORD. As a result, when programming Flash memory on a device with ECC, the programming operation must be, at minimum, four instruction Words or in groups of four instruction Words. This is the reason that the Quad Double Word programming command exists and why row programming always programs multiples of four Words.

For a given software application, ECC can be enabled at all times, disabled at all times or dynamically enabled using the ECCCTL Configuration bits in the CFGCON0 register. When ECC is enabled at all times, the DWORD NVMOP programming command does not function and the Quad Double Word is the smallest unit of memory that can be programmed. If disabling or enabling the ECC dynamically, both the DWORD and Quad Double Word programming NVMOP commands are functional and the programming method used determines how ECC is handled.

If dynamic ECC and if the memory was programmed with the DWORD command, ECC is turned-off for that Word, and, when it is read, no error correction is performed. If the memory was programmed with the Quad Double Word or row programming commands, ECC data is written and tested for errors (and corrected if needed) when read. The following table describes the different ECC scenarios.

**Table 15-2.** ECC Programming Summary

| ECCCTL Setting | Programming Operation | | | Data Read |
|---|---|---|---|---|
| | DWord | Quad Double Word | Row Write | |
| Disabled | Allowed | Allowed | Allowed | ECC is never applied on a Flash read |
| Enabled | Not allowed | Allowed | Allowed | ECC is applied on every Flash Word read |
| Dynamic | Allowed but when used, the programmed Word is flagged to NOT USE ECC | Writes ECC data and flags programmed Words to USE ECC | Writes ECC data and flags programmed Words to USE ECC | ECC is only applied on Words that are flagged to USE ECC |
| **Note:** When using dynamic ECC, the user must program all the non-ECC locations with the 64-bit Word programming command and program all the ECC-enabled locations with a 256-bit Double Quad Double Word or row programming command. Divisions between ECC and non-ECC memory must be on even Quad Word boundaries (address bits 0 through 3 are equal to '0'). | | | | |

**Related Links**

Prefetch Cache (PCHE)

## 15.10. Interrupts

An interrupt is generated when the Flash controller clears the WR bit upon completion of a Flash program or erase operation. The interrupt event causes a CPU interrupt if it was configured and enabled in the NVIC. For the vector mapping table, see *Nested Vector Interrupt Controller (NVIC)* from


© 2025 Microchip Technology Inc. and its subsidiaries

Related Links. The interrupt occurs regardless of the outcome of the program or erase operation, successful or unsuccessful. The only exception is the No Operation (NOP) programming operation (NVMOP = 0), which is used to manually clear the error flags and does not create an interrupt event on completion but does clear the WR bit.

The Flash Controller interrupts are not persistent, and, therefore, there is no need for an additional step to clear the cause or source of the interrupt.

After configuring the interrupt controller, the Flash event causes the CPU to jump to the vector assigned to the Flash event. The CPU starts executing the code at the vector address. The user software at this vector address must perform the required operations and, then, exit.

**Related Links**

Nested Vector Interrupt Controller (NVIC)

### 15.10.1. Interrupts and CPU Stalling

The CPU cannot fetch the code from the same Flash bank, either BFM or PFM, that is the target of the programming operation. When attempting this operation, the CPU ceases to execute code (stall) while the programming operation is in progress. CPU code execution does not resume until the programming operation is complete, and, when this occurs, any pending interrupts, including those from the Flash Controller, are processed in order of priority.

**Note:** Code that is already loaded into the processor cache continues to execute up to the point where an attempt is made to fetch code or data from the same Flash bank as the active programming operation. At this point the CPU stops.

The user can avoid the stalling of the CPU by placing any needed executable code in SRAM during Flash programming.

## 15.11. Error Detection

The NVMCON register includes two bits for detecting error conditions during a program or erase operation. They are Low-Voltage Detect Error, LVDERR bit (NVMCON[12]), and Write Error, WRERR bit (NVMCON[13]).

The WRERR is set each time the WR bit (NVMCON[15]) is set, initiating a programming operation. When the Flash operation is complete, indicated by hardware clearing the value of the WR bit (WR bit is set to '0'), hardware updates the value in the WRERR bit to indicate if an error occurred. Firmware must check the value of the WR bit to see if the Flash operation completed before checking the value of the WRERR bit. When the WRERR bit is set, any future attempt to initiate programming or erase operation is ignored. WRERR must be cleared before commencing Flash program or erase operations.

The LVDERR bit is set when a Brown-out Reset (BOR) occurs during a programming operation. The only Reset that clears the LVDERR bit is a Power-on Reset (POR). Other Reset types do not affect the LVDERR bit. When the LVDERR bit is set, any attempt to initiate programming or erase operation is ignored. Clear the bit before commencing Flash program or erase operations.

In the software, manually clear both the WRERR and LVDERR bits by initiating a Flash operation (setting WR) referred to as NOP (0x00) (see the NVMOP bit fields).

**Note:** Executing the NVMOP NOP command clears WRERR, LVDERR and WR bits but does not generate an interrupt event on completion.

**Table 15-3.** Programming Error Cause and Effects

| Cause of Error | Effect on Programming Erase Operation | Indication |
|---|---|---|
| A low-voltage event occurred during a programming sequence. | The last programming or erase operation might not have completed. | LVDERR = 1, WRERR = 1 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 15-3.** Programming Error Cause and Effects (continued)

| Cause of Error | Effect on Programming Erase Operation | Indication |
|---|---|---|
| A non-POR Reset occurred during programming. | Programming or erase operation is aborted. | WRERR = 1 |
| Attempt to program or erase a page out of the Flash memory range. | Erase or programming operation is not initiated. | WRERR = 1 |
| Attempt to erase or program a write-protected PFM page. | Erase or programming operation is not initiated. | WRERR = 1 |
| Attempt to erase or program a write-protected BFM page. | Operation occurs, but the page is not programmed or erased. | WRERR = 0 |
| Bus host error or row programming data underrun error during programming. | Programming or erase operation is aborted. | WRERR = 1 |

## 15.12. NVMKEY Register Unlocking Sequence

Important register settings that can compromise the Flash memory if inadvertently changed are protected by a register-unlocking sequence. The user can implement this feature using the NVMKEY register. The NVMKEY register is a write-only register that is used to implement an unlock sequence to help prevent accidental writes or erasures of the Flash memory.

In some instances, the operation is also dependent on the setting of the WREN bit (NVMCON[14]) (see the following table).

**Table 15-4.** NVMKEY Register Unlocking and WREN

| Operation | WREN Setting | Unlock Sequence Required |
|---|---|---|
| Changing value of NVMOP[3:0] (NVMCON[3:0]) | 0 | No |
| Setting WR (NVMCON[15]) to start a write or erase operation | 1 | Yes |
| Changing any fields in the NVMPWP* register | — | Yes |
| Changing any fields in the NVMLBWP register | — | Yes |

The user must follow the following steps in the exact order as shown to enable writes to registers that require this unlock sequence:

1. Write 0x00000000 to NVMKEY.
2. Write 0xAA996655 to NVMKEY.
3. Write 0x556699AA to NVMKEY.
4. Write the value to the register NVMCON, NVMCON2, NVMPWP* or NVMLBWP requiring the unlock sequence.

When using the unlock sequence to set or clear bits in the NVMCON register (see step 4). The user must execute steps 2 through 4 without any other activity on the peripheral bus that is in use by the Flash Controller. Disable the interrupts and DMA transfers that access the same peripheral bus as the Flash Controller. In addition, the operation in step 4 must be atomic. Use the Set, Clear and Invert registers, where applicable, for the target register in step 4.

The following code shows code written in the C language to initiate an NVM Operation (NVMOP) command. In this particular example, the WR bit is being set in the NVMCON register and, therefore, must include the unlock sequence.


© 2025 Microchip Technology Inc. and its subsidiaries

**Initiate NVM Operation (System Unlock Sequence Example)**:

```
void NVMInitiateOperation(void)
{
  // Disable Interrupts
  asm volatile("di%0" : "=r"(int_status));
  uint32_t globalInterruptState= __get_PRIMASK();
// Disable Interrupts
__disable_irq();
  NVMKEY = 0x0;
  NVMKEY = 0xAA996655;
  NVMKEY = 0x556699AA;
  NVMCONSET = 1 << 15;// must be an atomic instruction

  // Restore Interrupts
  __set_PRIMASK(globalInterruptState);
}
```

**Note:** After writing the unlock codes to the NVMKEY register, the next activity on the same peripheral bus as the Flash Controller resets the lock. As a result, the user can only use the atomic operations. Use of the NVMCONSET register sets the WR bit in a single instruction without changing other bits in the register. Using NVMCONbits.WR = 1 fails as this line of code compiles to a read-modify-write sequence.

## 15.13. DWord Programming

The smallest block of data that can be programmed in a single operation is one Flash write Word (64-bit). The data to be programmed must be written to the NVAMDATA0 and NVMDATA1 registers, and the address of the Word must be loaded into the NVMADDR register before initiating the programming sequence. The instruction Word at the physical location pointed to by the NVMADDR register is, then, programmed. Programming occurs on 64-bit Dword boundaries; therefore, the user can ignore the bits '0' and '1' of the NVMADDR register.

If programming a Word, erase it before programming again, even if changing a bit from an erased '1' state to a '0' state.

Word programming only succeeds if the target address is in a page that is not write-protected. Programming to a write-protected PFM page fails and results in the WRERR bit being set in the NVMCON register. Programming a write-protected BFM page fails but does not set the WRERR bit.

A programming sequence consists of the following steps:

1. Write two 32-bit data to be programmed to the NVMDATA0 and NVMDATA1 register.
2. Load the NVMADDR register with the address to be programmed.
3. Set the WREN bit = 1 and NVMOP bits = 1 in the NVMCON register. This defines and enables the programming operation.
4. Initiate the programming operation. (See *NVMKEY Register Unlocking Sequence* from Related Links.)
5. Monitor the WR bit of the NVMCON register to flag completion of the operation.
6. Clear the WREN bit in the NVMCON register.
7. Check for errors and process accordingly.

The following code shows code for Double Word programming, where a value of NVMDATA0 = 0x11111111 and NVMDATA1 = 0x22222222 are programmed into address 0x1008000 and 0x1008004.

**Word Programming Code Example**:

```
…
// Set up Address and Data Registers
  NVMADDR= 0x1008000;          // physical address
  NVMDATA0 = 0x11111111;  //value written to 0x1008000
  NVMDATA1 = 0x22222222; //value written to 0x1008004
```


© 2025 Microchip Technology Inc. and its subsidiaries

```
// set the operation, assumes WREN = 0
  NVMCONbits.NVMOP = 0x1;         // NVMOP for Word programming

// Enable Flash for write operation and set the NVMOP
  NVMCONbits.WREN = 1;

// Start programming
  NVMInitiateOperation();               // see Initiate NVM Operation (Unlock Sequence
Example)

// Wait for WR bit to clear
  while (NVMCONbits.WR);

// Disable future Flash Write/Erase operations
  NVMCONbits.WREN = 0;

// Check Error Status
  if(NVMCON & 0x3000)            // mask for WRERR and LVDERR
  {
        // process errors
    }
…
```

**Related Links**

NVMKEY Register Unlocking Sequence

## 15.14. Quad Double Word Programming

The process for Quad Double Word programming is identical to Word programming except that all eight of the NVMDATAx registers are used. The value of the NVMDATA0 register is programmed at address NVMADDR and so on. Refer to the following code example for details.

Always perform the Quad Double Word programming on a Quad Double Word boundary so the user can ignore the NVMADDR address bits 3 through 0.

Quad Double Word programming only succeeds, if the target address is in a page that is not write-protected. If programming a Quad Double Word, erase it before programming any word in it again, even if changing a bit from an erased '1' state to a '0' state.

The value of 0x11111111 is programmed into location 0x1008000 and so on. Refer to the following code example for details.

The following example code provides details about the Quad Double Word programming.

**Quad Double Word Programming Code Example**:

```
…

// Set up Address and Data Registers
    NVMADDR  = 0x1008000;      // physical address
    NVMDATA0 = 0x11111111;     // value written to 0x1008000
    NVMDATA1 = 0x22222222;     // value written to 0x1008004
    NVMDATA2 = 0x33333333;     // value written to 0x1008008
    NVMDATA3 = 0x44444444;     // value written to 0x100800C
    NVMDATA4 = 0x55555555      // value written to 0x1008010
    NVMDATA5 = 0x66666666      // value written to 0x1008014
    NVMDATA6 = 0x77777777      // value written to 0x1008018
    NVMDATA7 = 0x88888888      // value written to 0x100801C

// set the operation, assumes WREN = 0
    NVMCONbits.NVMOP = 0x2;    // NVMOP for Quad Double Word programming

// Enable Flash for write operation and set the NVMOP
    NVMCONbits.WREN = 1;

// Start programming
    NVMInitiateOperation();    // see Initiate NVM Operation (Unlock Sequence Example)

// Wait for WR bit to clear
    while(NVMCON & NVMCON_WR);

// Disable future Flash Write/Erase operations
```


© 2025 Microchip Technology Inc. and its subsidiaries

```
    NVMCONbits.WREN = 0;

// Check Error Status
    if(NVMCON & 0x3000)    // mask for WRERR and LVDERR bits
```

## 15.15. Row Programming

The largest block of data that can be programmed is a row.

Unlike Word and Quad Word Programming where the SFR memory stores the data source, in Row programming, the SRAM stores the source data. The NVMSRCADDR register is a pointer to the physical location of the source data for Row programming.

Like other Non-Volatile Memory (NVM) programming commands, the NVMADDR register points to the target address of the operation. Row programming always occurs on row boundaries with the row size of 1024. The user can ignore the bits 0 through 9 of the NVMADDR register.

Row Word programming only succeeds if the target address is in a page that is not write-protected. If row programming, erase it before programming any word in it again, even if changing a bit from an erased '1' state to a '0' state.

Array `rowbuff` is populated with data and programmed into a row located at physical address `0x1008000`.

**Notes:**

- When assigning the value to the NVMSRCADDR register, convert it to a physical address.
- Concurrent access from the Wireless Subsystem and Flash controller using row programming (NVM_RowWrite API) to FlexRAM can cause corrupt data. While using row programming, the Wireless Subsystem must be Inactive. Otherwise, use the quad-word programming (NVM_QuadWordWrite API) method.

**Row Programming Code Example**:

```
…

unsigned long rowbuff[256]; // example is for a 256 Word row size.
int x;                      // loop counter

// put some data in the source buffer
      for (x = 0; x < (sizeof(rowbuff) * sizeof (int)); x++)
            ((char *)rowbuff)[x] = x;

// set destination row address
     NVMADDR = 0x1008000;             // row physical address

// set source address. Must be converted to a physical address.
     NVMSRCADDR = (unsigned int)((int)rowbuff & 0x1FFFFF);

// define Flash operation
      NVMCONbits.NVMOP = 0x3;            // NVMOP for Row programming

// Enable Flash Write
      NVMCONbits.WREN = 1;

// commence programming
      NVMInitiateOperation();            // see Initiate NVM Operation (Unlock Sequence
Example)

// Wait for WR bit to clear
      while(NVMCONbits.WR);

// Disable future Flash Write/Erase operations
       NVMCONbits.WREN = 0;

// Check Error Status
       if(NVMCON & 0x3000)            // mask for WRERR and LVDERR bits
      {
         // process errors
       }
…
```


© 2025 Microchip Technology Inc. and its subsidiaries

## 15.16. Page Erase

A Page Erase performs an erase of a single page of either PFM or BFM.

The page to be erased is selected using the NVMADDR register. Pages are always erased on page boundaries; therefore, for a device with an instruction Word page size of 4096, the user can ignore the bits 0 through 11 of the NVMADDR register.

A Page Erase only succeeds, if the target address is a page that is not write-protected. Erasing a write-protected page fails and result in the WRERR bit being set in the NVMCON register.

The following code shows the code for a single Page Erase operation at address 0x1008000.

**Page Erase Code Example**:

```
…

// set destination page address
    NVMADDR = 0x1008000;     // page physical address

// define Flash operation
    NVMCONbits.NVMOP = 0x4;    // NVMOP for Page Erase

// Enable Flash Write
    NVMCONbits.WREN = 1;

// commence programming
    NVMInitiateOperation();    // see Initiate NVM Operation (Unlock Sequence Example)

// Wait for WR bit to clear
    while(NVMCONbits.WR);

// Disable future Flash Write/Erase operations
    NVMCONbits.WREN = 0;

// Check Error Status
    if(NVMCON & 0x3000)    // mask for WRERR and LVDERR bits
    {
        // process errors
    }

…
```

### 15.16.1. Page Erase Retry

Page Erase Retry is a method to improve the life of a Flash by attempting to erase again if the Page Erase was not successful. The user can only use the Erase Retry for a Page Erase.

Page Erase Retry works by increasing the voltage used on the Flash when erasing. Initially, the minimum voltage necessary is applied by setting the RETRY[1:0] bits (NVMCON2[9:8]) = `01`. If the page erase is not successful, the voltage may be increased by incrementing the setting of the RETRY[1:0] bits from `01` to `10` to `11` and finally to `00`.

**Note:** Each Flash page, as it ages and wears, can have different voltage requirements; therefore, a higher setting on one Flash page does not indicate that the same setting must be used on all pages.

The maximum voltage for Page Erase is used when the RETRY[1:0] bits = `00`. If Page Erase is not successful after 7 trials, the Flash for that page, or the Words that did not erase, must be considered non-functional.

Together with the normal Page Erase controls, Page Erase Retry also uses the WS[4:0], CREAD1, VREAD1 and RETRY[1:0] bits in the NVMCON2 register. The ERS[3:0] bits (NVMCON2[31:28]) are for the benefit of software performing the programming sequence in the event that a drop in power causes a BOR event but not a POR event.

Perform the following steps to set up a Page Erase Retry:

1. Set the NVMADDR register with the address of the page to be erased.
2. Execute the write unlock sequence.


© 2025 Microchip Technology Inc. and its subsidiaries

3. Save the value of the NVMCON2 register.
4. Do the following in the NVMCON2 register:
   a. Set the ERS[3:0] bits as desired.
   b. Set the WS[4:0] bits per the description.
   c. Set the VREAD1 bit to '1'.
   d. Set the CREAD1 bit to '1'.
   e. Set the RETRY[1:0] bits to '01', '10', '11', '00'.
5. Run the unlock sequence using the Page Erase command to start the sequence.
6. Wait for the WR bit (NVMCON[15]) to be cleared by hardware.
7. Clear the WREN bit (NVMCON[14]).
8. Verify the erase using the CPU. To shorten the verify time, use CREAD1 = 1 to perform a hardware compare to logic '1' of each bit in the Flash Word including ECC. A successful compare yields a read of 0x00000001 in the lowest addressed Word in a Flash Word (256 bits). This is the Compare Word. All other Words are 0x00010000. If any bit is logic '0', all Words in the Flash Word read 0x00000000. Remember to increment the address by the number of bytes in a Flash Word between reads.
9. If all Compare Words verify correctly, the Page Erase Retry process is complete. Go to step 11.
10. If a Compare Word yields a read of 0x00000000, perform steps 4 through 9 up to six more times with the following change to step 4:
    a. Increment the RETRY[1:0] bits by one if the bit has not already reached the '00' setting.
    b. Maintain all other fields.
11. Restore the value of the NVMCON2 register, which was saved in step 3.

**Notes:**

1. When CREAD1 is set to '1' to perform hardware compare, the compare happens on the entire Flash panel (PFM, BFM). Hence, execute the code that performs this page erase retry operation from SRAM.
2. When the VREAD1 = 1, the Flash uses the WS[3:0] bits for the Flash access wait state generation to the panel selected by NVMADDR. Software is responsible for writing the VREAD1 bit back to '0' when both erase and verify is complete.
3. The device configuration boot page (the page containing the DEVCFGx values) does not support Page Erase Retry.

The following code provides code for a single page erase operation at address 0x1008000, in case of Page Erase Retry.

**Page Erase Retry Code Example**:

```
uint32_tsaveNVMCON2;
uint32_t*cmpPtr;
uint8_terased;
uint8_ttryCount;

// set destination page address
    NVMADDR = 0x1008000;    // Page physical address

// define flash operation
    NVMCONbits.NVMOP = 0x4;    // NVMOP for Page Erase

// Unlock sequence
    NVMKEY = 0x0;
    NVMKEY = 0xAA996655;
    NVMKEY = 0x556699AA;

// save NVMCON2
    saveNVMCON2 = NVMCON2;
```


© 2025 Microchip Technology Inc. and its subsidiaries

```
// set up Page Erase Retry
    NVMCON2bits.ERS = 0;    // Stage 0 - SW use only
    NVMCON2bits.VREAD1 = 1;
    NVMCON2bits.CREAD1 = 1;
    NVMCON2bits.RETRY = 0b01;

    tryCount = 0;        // Up to 4 attempts

    do {
        tryCount++;

        // commence programming
        NVMInitiateOperation();

        // Wait for WR bit to clear
        while(NVMCONbits.WR);

        // Turn off WREN
        NVMCONbits.WREN = 0;

        // Check that the page was erased
        erased = 1;
        cmpPtr = (uint32_t *)NVMADDR;
        erased &= (*cmpPtr == 0x00000001);
        cmpPtr++;
        erased &= (*cmpPtr == 0x00010000);
        cmpPtr++;
        erased &= (*cmpPtr == 0x00010000);
        cmpPtr++;
        erased &= (*cmpPtr == 0x00010000);

        if (!erased) {
            // Erase failed. Try with different settings.
            NVMCON2bits.RETRY++;

            NVMCONbits.NVMOP = 0x4;
            NVMCONbits.WREN = 1;
        }
    } while (!erased && (tryCount < 4));

// Restore settings
    NVMCON2 = saveNVMCON2;
```

## 15.17. Program Flash Memory (PFM) Erase

Program Flash memory can be erased entirely. All three discrete NVMOP values, 0111, 0110, 0101, do the same operation of erase of entire Flash. When erasing the entire PFM area, in case of RTSP (Run Time Self Programming), the code must be executing from BFM. When erasing the entire PFM area, PFM write-protection must be completely disabled.

The following code shows code for erasing the entire Flash bank.

**Program Flash Erase Code Example**:

```
…

// define Flash operation
    NVMCONbits.NVMOP = 0x7;        // NVMOP for entire PFM erase

// Enable Flash Write
    NVMCONbits.WREN = 1;

// commence programming
    NVMInitiateOperation();        // see Initiate NVM Operation (Unlock Sequence Example)

// Wait for WR bit to clear
    while(NVMCONbits.WR);

// Disable future Flash Write/Erase operations
    NVMCONbits.WREN = 0;

// Check Error Status
    if(NVMCON & 0x3000)        // mask for WRERR and LVDERR bits
    {
         // process errors
```


© 2025 Microchip Technology Inc. and its subsidiaries

```
        }
...
```

## 15.18. Device Code Protection Bit (CP)

The PIC32CX-BZ6 family of devices features code protection, which, when the user enables it, prevents reading of the Flash memory by an external programming device (SWD through DSU).

If enabling code protection, the user can only disable it by erasing the device with the `Chip Erase` command through an external programmer. See *Device Service Unit (DSU)* from Related Links.

When programming a device that opts to utilize code protection, the external programming device must perform verification prior to enabling code protection. Enabling code protection must be the last step of the programming process. For the location of the code protection enable bits, refer to the *PIC32CX-BZ6 Programming Specification*. Additionally, see *System Configuration and Register Locking (CFG)* from Related Links.

**Related Links**

Device Service Unit (DSU)

System Configuration and Register Locking (CFG)

## 15.19. Effects of Various Resets

Device Resets, other than a Power-on Reset (POR), reset the entire contents of the NVMPWP and NVMLBWP registers. All other register content persists through a non-POR reset.

All Flash Controller registers are forced to their reset states upon a POR.


© 2025 Microchip Technology Inc. and its subsidiaries

## 15.20. Register Summary

See *FC* module in the *Product Memory Mapping Overview* from Related Links for the base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | NVMCON | 7:0 | | | | | NVMOP[3:0] | | | |
| | | 15:8 | WR | WREN | WRERR | LVDERR | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | NVMCON2 | 7:0 | | | | | | | | NVMPREPG |
| | | 15:8 | | | CREAD1 | VREAD1 | | | RETRY[1:0] | |
| | | 23:16 | | | | WS[4:0] | | | | |
| | | 31:24 | ERS[3:0] | | | | | | | SLEEP |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | NVMKEY | 7:0 | NVMKEY[7:0] | | | | | | | |
| | | 15:8 | NVMKEY[15:8] | | | | | | | |
| | | 23:16 | NVMKEY[23:16] | | | | | | | |
| | | 31:24 | NVMKEY[31:24] | | | | | | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | NVMADDR | 7:0 | NVMADDR[7:0] | | | | | | | |
| | | 15:8 | NVMADDR[15:8] | | | | | | | |
| | | 23:16 | NVMADDR[23:16] | | | | | | | |
| | | 31:24 | NVMADDR[31:24] | | | | | | | |
| 0x34 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | NVMDATA0 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0x44 ... 0x4F | Reserved | | | | | | | | | |
| 0x50 | NVMDATA1 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0x54 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | NVMDATA2 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0x64 ... 0x6F | Reserved | | | | | | | | | |
| 0x70 | NVMDATA3 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x74 ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | NVMDATA4 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0x84 ... 0x8F | Reserved | | | | | | | | | |
| 0x90 | NVMDATA5 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0x94 ... 0x9F | Reserved | | | | | | | | | |
| 0xA0 | NVMDATA6 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0xA4 ... 0xAF | Reserved | | | | | | | | | |
| 0xB0 | NVMDATA7 | 7:0 | NVMDATA[7:0] | | | | | | | |
| | | 15:8 | NVMDATA[15:8] | | | | | | | |
| | | 23:16 | NVMDATA[23:16] | | | | | | | |
| | | 31:24 | NVMDATA[31:24] | | | | | | | |
| 0xB4 ... 0xBF | Reserved | | | | | | | | | |
| 0xC0 | NVMSRCADDR | 7:0 | NVMSRCADDR[7:0] | | | | | | | |
| | | 15:8 | NVMSRCADDR[15:8] | | | | | | | |
| | | 23:16 | NVMSRCADDR[23:16] | | | | | | | |
| | | 31:24 | NVMSRCADDR[31:24] | | | | | | | |
| 0xC4 ... 0xCF | Reserved | | | | | | | | | |
| 0xD0 | NVMPWPLT | 7:0 | PWPLT[7:0] | | | | | | | |
| | | 15:8 | PWPLT[15:8] | | | | | | | |
| | | 23:16 | PWPLT[23:16] | | | | | | | |
| | | 31:24 | ULOCK | | | | | | | |
| 0xD4 ... 0xDF | Reserved | | | | | | | | | |
| 0xE0 | NVMPWPGTE | 7:0 | PWPGTE[7:0] | | | | | | | |
| | | 15:8 | PWPGTE[15:8] | | | | | | | |
| | | 23:16 | PWPGTE[23:16] | | | | | | | |
| | | 31:24 | ULOCK | | | | | | | |
| 0xE4 ... 0xEF | Reserved | | | | | | | | | |
| 0xF0 | NVMLBWP | 7:0 | LBWP[7:0] | | | | | | | |
| | | 15:8 | LBWP[15:8] | | | | | | | |
| | | 23:16 | | | | | | | LBWP[17:16] | |
| | | 31:24 | ULOCK | | | | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview


© 2025 Microchip Technology Inc. and its subsidiaries

## 15.21. Register Description

The following NVM control registers control the Flash program, erase and write protection operations:

- NVMCON: Programming Control Register
  - This register is the control register for Flash program/erase operations. The following are the uses of this register:
    - Selects the operation to be performed
    - Initiates the operation
    - Provides status of the result after completing the operation
- NVMCON2: Programming Control2 Register
  - This register is the control and status register for Flash program/erase operations.
- NVMKEY: Programming Unlock Register
  - This is a write-only register that helps to or that helps the user to implement an unlock sequence to help prevent accidental writes/erasures of Flash memory and write permission settings.
- NVMADDR: Flash Address Register
  - This register stores the physical target address for row, Quad Double Word and Single Double Word programming as well as page erasing.
- NVMDATAx: Flash Program Data Register (x = 0-3)
  - These registers hold the data to be programmed during Flash Word program operations.
- NVMSRCADDR: Source Data Address Register
  - This register points to the physical address of the data to be programmed when executing a row program operation.
- NVMPWPLT: Flash Program Write Protect Lower Register
  - This register sets the program flash pages lower than provided address as a write protected.
- NVMPWPGTE: Flash Program Write Protect Greater Register
  - This register sets the program flash pages greater than provided address as a write protected.
- NVMLBWP: Flash Boot Write Protect Register
  - This register sets the boot flash partition pages as a write protected.

The following is the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.1. NVMCON - Programming Control Register

**Name:** NVMCON
**Offset:** 0x00
**Reset:** 0x00000000
**Property:** -

**Notes:**

1. See *NVMPWPLT* and *NVMPWPGTE* for program space write protection conditions for the operation used in NVMOP.
   See *NVMLBWP* for NVR space write protection conditions for the operation used in NVMOP.
2. This operation is only allowed if test_detect_active is asserted else no operation occurs. No interrupt generated, WRERR not set.
   Only available to internal (CPU) sources. All NVR and PWP protection mechanisms are ignored.
3. This operation is only allowed if test_private_mode is asserted else no operation occurs. No interrupt generated, WRERR not set. Only available to internal (CPU) sources. All NVR and PWP protection mechanisms are ignored.
4. If `cfg_flash_eccctl[1:0]=2'b00`, this operation no operation occurs but does not affect WRERR or LVDERR.
5. SRF is included in this erase. SRF substitution addresses stored in test flash are not affected by this erase.
6. When Code Protected these NVMOP commands are dis-allowed and status does not change. (This protects against modifying user CP/SIGN settings or corrupting user code.)
7. Behaves as R-0, when PFM_NUM_PANELS=`1` or `cfg_pfm_num_panels = 2'b10 or 2'b01`.
8. These bits are only reset by a POR and are not affected by other Reset sources.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | WR | WREN | WRERR | LVDERR | | | | |
| Access | R/S/HC | R/W | R/HS/HC | R/HS/HC | | | | |
| Reset | 0 | 0 | 0 | 0 | | | | |

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

**Bit 15 – WR** Write Control Bit

**Note:** This field can only be modified when WREN = `1`, TEMP = `1` and the NVMKEY unlock sequence is satisfied.

| Value | Description |
|---|---|
| `1` | Initiate a Flash operation. Hardware clears this bit when the operation completes |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Flash operation complete or inactive |

**Bit 14 – WREN** Write Enable Bit

| Value | Description |
|---|---|
| 1 | Enables writes to WR |
| 0 | Disables writes to WR |

**Bit 13 – WRERR** Write Error Bit(8)

**Note:** Cleared by setting NVMOP == 0000b and initiating a Flash operation (WR)

| Value | Description |
|---|---|
| 1 | Program or erase sequence did not complete successfully |
| 0 | Program or erase sequence completed normally |

**Bit 12 – LVDERR** Low Voltage Detect Error Bit(8)

The error is only captured for programming/erase operations (when NVMWR = 1).

**Note:** Cleared by setting NVMOP == 0000b and initiating a Flash operation (NVMWR)

| Value | Description |
|---|---|
| 1 | Low voltage is detected (possible data corruption if WRERR is set) |
| 0 | Voltage level is within the acceptable range for programming |

**Bits 3:0 – NVMOP[3:0]** NVM Operation Bits(8)

These bits are only writable when WREN = 0.

| Value | Description |
|---|---|
| 1111 | Reserved |
| 1110(2) | Chip Erase Operation: Erases PFM, BFM (except configuration page) when accessed through SWD interface only |
| . . . | ... |
| 1000 | Reserved |
| 0111(5,6) | Program erase operation: Erase all program Flash memory (PFM). All pages in this region must be unprotected for the erase operation to proceed. |
| 0110(5,6) | Upper Program Flash Memory Erase Operation: Erases only the upper mapped region of the program Flash memory. All pages in this region must be unprotected for the erase operation to proceed. In the PIC32CX-BZ6, which has a single bank of Flash memory, this operation functions the same as when NVMOP is set to 0111. |
| 0101(5,6) | Lower Program Flash Memory Erase Operation: Erases only the lower mapped region of the program Flash memory. All pages in this region must be unprotected for the erase operation to proceed. In the PIC32CX-BZ6, which has a single bank of Flash memory, this operation functions the same as when NVMOP is set to 0111. |
| 0100(1,6) | Page erase operation: Erases the page selected by NVMADDR if the page is not write-protected |
| 0011(1,6) | Row program operation: Programs row selected by NVMADDR if the page is not write-protected |
| 0010(1,6) | Quad Double Word (128-bit) program operation: Programs the 128-bit Flash Word selected by NVMADDR if the page is not write-protected |
| 0001(1,4,6) | Double Word program operation: Programs the Word selected by NVMADDR if the page is not write-protected(2) |
| 0000 | Clear the status of WRERR and LVDERR |


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.2. NVMCON2 - Programming Control 2 Register

**Name:** NVMCON2
**Offset:** 0x10
**Reset:** 0x011F4000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ERS[3:0] | | | | | | | SLEEP |
| Access | R/W | R/W | R/W | R/W | | | | R/W |
| Reset | 0 | 0 | 0 | 0 | | | | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | WS[4:0] | | | | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | CREAD1 | VREAD1 | | | RETRY[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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | NVMPREPG |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

**Bits 31:28 – ERS[3:0]** Erase Retry State Bit
The software uses these bits to track the software state of the erase retry procedure in the event of a system Reset ($\overline{MCLR}$) or Brown-out Reset (BOR) event.
**Note:** These bits are only reset by a POR and are not affected by other Reset sources.

**Bit 24 – SLEEP** NVM Power Down in Sleep Mode Bit
**Note:** This field can only be modified when the NVMKEY unlock sequence is satisfied.

| Value | Description |
|---|---|
| 1 | Configures Flash for power down when the system is in Sleep mode |
| 0 | Configures Flash for standby when the system is in Sleep mode |

**Bits 20:16 – WS[4:0]** Flash Access Wait State Control for NVMVREAD1 = 1
**Notes:**
1. When VREAD1 = 1, WS[4:0] only affects the memory containing NVMADDR[31:0].
2. The NVMKEY unlock sequence must be satisfied to modify this field.

| Value | Description |
|---|---|
| 11111 | 31 wait states (32 total system clocks) |
| 11110 | 30 wait states (31 total system clocks) |
| ... | |
| 00010 | 2 wait states (3 total system clocks) |
| 00001 | 1 wait state (2 total system clocks) |
| 00000 | 0 wait state (1 total system clock) |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 13 – CREAD1** Compare Read of Logic 1 Bit

Compare Read 1 causes all bits in a Flash Word (including ECC if it exists) to be evaluated during the read. If all bits are 1, the lowest Word in the Flash Word evaluates to 0x0000_0001, all other Words are 0x0001_0000. If any bit is 0, the read evaluates to 0x0000_0000 for all Words in the Flash Word.

**Notes:**

1. When using erase retry in an ECC Flash system, use CREAD1 = 1.
2. To modify this field, satisfy the NVMKEY unlock sequence.

| Value | Description |
|---|---|
| 1 | Compare read is enabled, only if NVMVREAD1 = 1 |
| 0 | Compare read is disabled |

**Bit 12 – VREAD1** Verify Read of logic 1 Control bit

**Notes:**

1. When VREAD1 = 1, Flash Wait state control is from WS[] for the memory containing NVMADDR[].
2. Using erase retry and verify read procedure increases the life of the Flash memory.
3. This field becomes modifiable only when NVMCON.WR = 0 and the NVMKEY unlock sequence is satisfied.

| Value | Description |
|---|---|
| 1 | Selects erase retry procedure with Verify Read |
| 0 | Selects single erase without Verify Read |

**Bits 9:8 – RETRY[1:0]** Erase Retry Control bit, only used when VREAD1 = 1

**Notes:**

1. This field becomes modifiable only when NVMCON.WR = 0.
2. These bits are only reset by a POR and are not affected by other Reset sources.

| Value | Description |
|---|---|
| 00 | Erase strength for last retry cycle |
| 11 | Erase strength for third retry cycle |
| 10 | Erase strength for second retry cycle |
| 01 | Erase strength for first retry cycle |

**Bit 0 – NVMPREPG** NVM Pre-Program Control Bit

**Note:** This field can only be modified when NVMCON.NVMWR = 0.

| Value | Description |
|---|---|
| 1 | Program Operations include Pre-Program step |
| 0 | Program Operations exclude Pre-Program step |


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.3. NVMKEY – Programming Unlock Register

**Name:** NVMKEY
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** -

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

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

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | NVMKEY[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMKEY[7:0] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – NVMKEY[31:0]** Unlock Register bits
These bits are write-only and read '0' on any read.
**Note:** This register is used as part of the unlock sequence to prevent inadvertent writes to the program Flash.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.4. NVMADDR – Flash Address Register

**Name:** NVMADDR
**Offset:** 0x30
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NVMADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMADDR[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 – NVMADDR[31:0]** Flash (Word) Address bits
**Note:** These bits are only reset by a POR and are not affected by other Reset sources.

**Table 15-5.** Flash (Word) Address Bits

| NVMOP | Flash Address Bits |
|---|---|
| Page Erase | • Address identifies the page to erase<br>• Any address within a 4 Kbytes page boundary will cause the page to be erased |
| Row program | • Address identifies the row to program<br>• The value of the address must be aligned to a row boundary |
| Double Word Program | 64-bit Word on 256-bit flash system<br>• Address identifies the Double Word to program<br>• NVMADDR[2:0] bits are ignored<br>• Must be aligned to a Double Word boundary |
| Quad Double Word Program | 256-bit Word on 256-bit flash system)<br>• Address identifies the Quad Double Word to program<br>• NVMADDR[4:0] bits are ignored<br>• Must be aligned to a Quad Double Word boundary |

**Notes:**

1. Hardware prevents writes to this register when NVMCON.WR = 1.
2. For all other NVMOP[3:0] bit settings, the Flash address is ignored. For additional information on these bits, see the *NVMCON* register from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**

NVMCON

NVMCON - Programming Control Register


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.5. NVMDATAx – Flash Program Data Register x

**Name:** NVMDATAx
**Offset:** 0x40 + x*0x10 [x=0..7]
**Reset:** 0x00000000
**Property:** -

**Note:** These bits are only reset by a POR and are not affected by other Reset sources.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NVMDATA[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMDATA[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMDATA[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMDATA[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 – NVMDATA[31:0]** Flash Programming Data x bits (x=0...7)
The value in this register is written to Flash when a program operation is commanded.

- Single Double Word Program (64-bit)
  - Writes NVMDATA0 to the target Flash address defined in NVMADDR]31:3]
  - Writes NVMDATA1 to the target Flash address defined in NVMADDR[31:3]
- Quad Double Word Program (256-bit)
  - Writes NVMDATA0 to NVMADDR[31:5], and bits [4:2]=000
  - Writes NVMDATA1 to NVMADDR[31:5], and bits [4:2]=001
  - Writes NVMDATA2 to NVMADDR[31:5], and bits [4:2]=010
  - Writes NVMDATA3 to NVMADDR[31:5], and bits [4:2]=011
  - Writes NVMDATA4 to NVMADDR[31:5], and bits [4:2]=100
  - Writes NVMDATA5 to NVMADDR[31:5], and bits [4:2]=101
  - Writes NVMDATA6 to NVMADDR[31:5], and bits [4:2]=110
  - Writes NVMDATA7 to NVMADDR[31:5], and bits [4:2]=111

**Note:**

- Hardware prevents writes to this register when NVMCON.NVMWR = 1.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.6. NVMSRCADDR – Source Data Address Register

**Name:** NVMSRCADDR
**Offset:** 0xC0
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NVMSRCADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMSRCADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMSRCADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | NVMSRCADDR[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 – NVMSRCADDR[31:0]** Source Data (Word) Address bits
This is the system physical Word address of the data (in DRM) to be programmed into the Flash when NVMCON.NVMOP is set to row programming.
**Notes:**

1. Hardware prevents writes to this register when NVMCON.NVMWR = 1.
2. These bits are only reset by a POR and are not affected by other Reset sources.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.7. NVMPWPLT – Flash Program Write Protect Lower Register

**Name:** NVMPWPLT
**Offset:** 0xD0
**Reset:** 0x80000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ULOCK | | | | | | | |
| Access | R/C | | | | | | | |
| Reset | 1 | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PWPLT[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 |
|---|---|---|---|---|---|---|---|---|
| | PWPLT[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 |
|---|---|---|---|---|---|---|---|---|
| | PWPLT[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 – ULOCK** NVMPWPLT Register Unlock Bit

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied.
2. This field can be cleared at the same time as writing to PWPLT[23:0].

| Value | Description |
|---|---|
| 1 | NVMPWPLT register is not locked and can be modified |
| 0 | NVMPWPLT register is locked and cannot be modified |

**Bits 23:0 – PWPLT[23:0]** Flash Program Write Protect Less Than Address

Pages at Flash addresses less than this value are write-protected.

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied, and ULOCK = 1.
2. This is a byte address force to align to page boundaries.
3. These bits are only reset by a POR and are not affected by other Reset sources.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.8. NVMPWPGTE – Flash Program Write Protect Greater Register

**Name:** NVMPWPGTE
**Offset:** 0xE0
**Reset:** 0x80FFFFFF
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ULOCK | | | | | | | |
| Access | R/C | | | | | | | |
| Reset | 1 | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | PWPGTE[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 |
|---|---|---|---|---|---|---|---|---|
| | PWPGTE[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 |
|---|---|---|---|---|---|---|---|---|
| | PWPGTE[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 31 – ULOCK** NVMPWPGTE Register Unlock bit

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied.
2. This field can be cleared at the same time as writing to PWPGTE[23:0].

| Value | Description |
|---|---|
| 1 | NVMPWPGTE register is not locked and can be modified |
| 0 | NVMPWPGTE register is locked and cannot be modified |

**Bits 23:0 – PWPGTE[23:0]** Flash Program Write Protect Address

Pages at Flash addresses greater than or equal to this value are write-protected.

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied and ULOCK = 1.
2. This is a byte address forced to align to page boundaries.
3. These bits are only reset by a POR and are not affected by other Reset sources.


© 2025 Microchip Technology Inc. and its subsidiaries

### 15.21.9. NVMLBWP - Flash Boot (Page) Write Protect Register

**Name:** NVMLBWP
**Offset:** 0xF0
**Reset:** 0x80FFFFFF
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ULOCK | | | | | | | |
| Access | R/C | | | | | | | |
| Reset | 1 | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | LBWP[17:16] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | LBWP[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 |
|---|---|---|---|---|---|---|---|---|
| | LBWP[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 31 – ULOCK** Boot (Page) Write Protect (LBWPn) Unlock Bit

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied.
2. This field can be cleared at the same time as writing to LBWP[msb:lsb].

| Value | Description |
|---|---|
| 1 | LBWPn bits are not locked and can be modified |
| 0 | LBWPn bits are locked and cannot be modified |

**Bits 17:0 – LBWP[17:0]** Boot Pages Write Protect Bits

- Bit [15:0] – Provides erase and write protection for Boot Flash pages 0 to 15.
- Bit 16 – Provides erase and write protection for the Device/Boot Configuration page.
- Bit 17 – Indicates whether the protection settings are One Time Programmable (OTP).
  **Note:** The OTP page is always erase protected and its associated LBWP bit is only for write protection.

**Notes:**

1. This field can only be modified when the NVMKEY unlock sequence is satisfied and ULOCK = 1.
2. The OTP page is always erase protected, and its associated LBWP bit is only for write protection.
3. These bits are only reset by a POR and are not affected by other Reset sources.

| Value | Description |
|---|---|
| 1 | Erase and write protection for lower boot page n is enabled |
| 0 | Erase and write protection for lower boot page n is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

# 16. Device Service Unit (DSU)

## 16.1. Overview

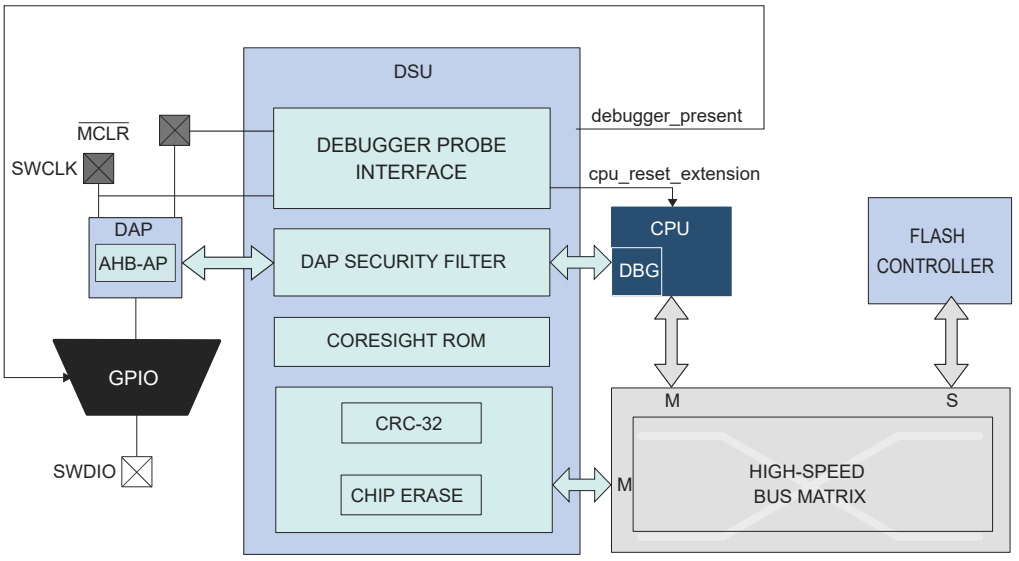
The Device Service Unit (DSU) provides a means of detecting debugger probes. It enables the ARM Debug Access Port (DAP) to have control over multiplexed debug pads and CPU Reset. The DSU also provides system-level services to debug adapters in an ARM debug system. It implements a CoreSight debug ROM 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 are unavailable when the Code Protect bit and SECCFG.DEBUG_LCK bit protect the device.

## 16.2. Features

- 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
- Debug Access Port Security Filter

## 16.3. DSU Block Diagram

**Figure 16-1.** DSU Block Diagram




## 16.4. Signal Description

The DSU uses three signals to function.


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 16-1.** Signal Description

| Signal Name | Type | Description |
|---|---|---|
| $\overline{\text{MCLR}}$ | Digital input | External Reset pin |
| CM4_SWCLK | Digital input | Software clock pin |
| CM4_SWDIO | Digital I/O | Software bidirectional data pin |

## 16.5. Product Dependencies

The following sections describe how the other parts of the system must be configured correctly to use this peripheral.

### 16.5.1. 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. See *Debugger Probe Detection* from Related Links. The Hot-Plugging feature depends on the GPIO configuration. If the user changes the SWCLK pin function in the port, the Hot-Plugging feature is not disabled. Hot-Plugging is disabled with the CFGCON0.HPLUGDIS bit, which is enabled by default. Therefore, to use the SWCLK pin for GPIO functions, disable it by setting CFGCON0.HPLUGDIS = 1.

**Related Links**

Debugger Probe Detection

### 16.5.2. Power Management

The DSU continues to operate in any Sleep mode (Standby Sleep, Idle) where the selected source clock is running.

### 16.5.3. Clocks

The DSU bus clocks (CLK_DSU_APB (SYS_CLK) and CLK_DSU_AHB (SYS_CLK)) can be enabled and disabled by the CRU.

### 16.5.4. DMA

The DMA request lines are connected to the DMA Controller (DMAC). The user must first configure the DMAC to use DMA requests with this peripheral. See *Direct Memory Access Controller (DMAC)* from Related Links.

The CFG.DCCDMALEVEL bit field must be configured depending on the DMA channels access modes (read or write for DCC0 and DCC1).

**Related Links**

Direct Memory Access Controller (DMAC)

### 16.5.5. Interrupts

Not applicable.

### 16.5.6. Events

Not applicable.

### 16.5.7. 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)


© 2025 Microchip Technology Inc. and its subsidiaries

**Notes:**

- Optional write protection is indicated by the PAC Write Protection property in the register description.
- If halting the CPU in the Debug mode, all write protection is automatically disabled. Write protection does not apply for accesses through an external debugger.

### 16.5.8. Analog Connections

Not applicable.

## 16.6. Debug Operation

### 16.6.1. 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

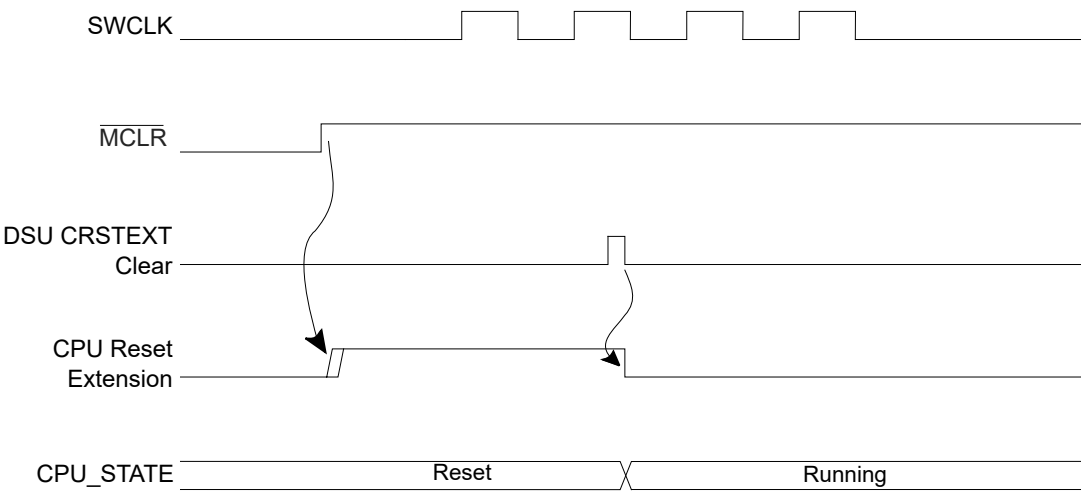
For more details on the ARM debug components, refer to the *ARM Debug Interface v5 Architecture Specification*.

Debug using SWD will be blocked on secured devices (SECCFG.DEBUG_LCK).

### 16.6.2. CPU Reset Extension

The CPU Reset extension refers to the extension of the Reset phase of the CPU core after releasing the external Reset. This ensures that the CPU is not executing code at start-up while a debugger is connected to the system. The debugger is detected on a $\overline{\text{MCLR}}$ release event when SWCLK is low. At start-up, the SWCLK is internally pulled up to avoid false detection of a debugger if the SWCLK pin is left 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 Code Protect bit (FCPN0.CP) or the SECCFG.DEBUG_LCK bit is protecting the device. Trying to do so sets the Protection Error bit (PERR) of the Status A register (STATUSA.PERR).

**Figure 16-2.** Typical CPU Reset Extension Set and Clear Timing Diagram





© 2025 Microchip Technology Inc. and its subsidiaries

### 16.6.3. Debugger Probe Detection

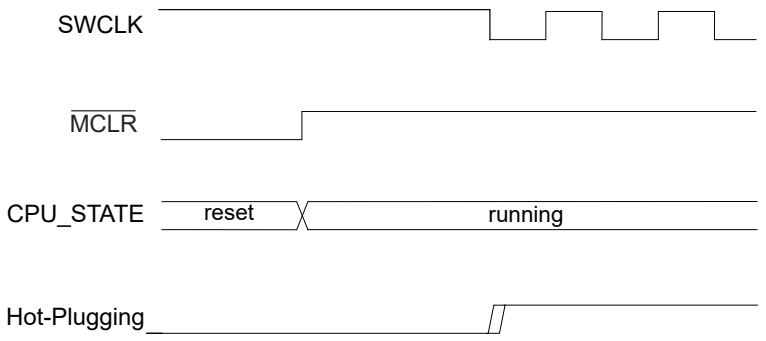
#### 16.6.3.1. Cold-Plugging

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.

#### 16.6.3.2. 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 $\overline{\text{MCLR}}$ 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 changing the SWCLK pin function in the port, the Hot-Plugging feature is not disabled. Hot-Plugging is disabled with the CFGCON0.HPLUGDIS bit, which is enabled by default. Therefore, to use the SWCLK pin for GPIO functions, it must be disabled by setting CFGCON0.HPLUGDIS = 1. The availability of the Hot-Plugging feature can be read from the Hot-Plugging Enable bit of the Status B register (STATUSB.HPE).

**Figure 16-3.** Hot-Plugging Detection Timing Diagram




The presence of a debugger probe is detected when either Hot-Plugging or Cold-Plugging is detected. After the detection, the Debugger Present bit of the Status B register (STATUSB.DBGPRES) is set. For security reasons, the Hot-Plugging is not available when the Code Protect bit (FCPN0.CP) or SECCFG.DEBUG_LCK bit is protecting the device.

In this detection, the user must correctly power the pads. Thus, at cold start-up, this detection cannot be done until POR is released. If the device is protected using Code Protect bit (FCPN0.CP) or SECCFG.DEBUG_LCK bit, 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 deasserting the external Reset before POR release, the user must retry the above procedure until it gets connected to the device.

## 16.7. Chip Erase

Chip erase consists of removing all sensitive information stored in the chip and clearing the Code Protect bit. Therefore, this erases all volatile memories and the Flash memory.
**Note:** The OTP Boot flash memory page will not be erased.

When the device is protected using the FCPN0.CP or SECCFG.DEBUG_LCK bit, the debugger must first Reset the device to be detected. This ensures that internal registers are reset after removing the Protected state. The user can trigger the chip erase operation 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). After issuing, the module clears volatile memories prior to erasing the Flash array. To ensure the completion of the chip erase operation, 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, the recommendation is to issue a Chip Erase command after a Cold-Plugging procedure to ensure that the device is in a known and safe state.


© 2025 Microchip Technology Inc. and its subsidiaries

The following is the recommended sequence:

1. Issue the Cold-Plugging procedure (See *Cold-Plugging* from Related Links), then the device performs the following:
   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 (excluding OTP)
   c. Erases the Lock Row and Code Protect bit protection
3. Check for completion by polling STATUSA.DONE (read as '1' when completed).
4. Reset the device to let the Flash Controller update the fuses.

**Related Links**
Cold-Plugging

## 16.8. Programming

Programming the Flash or RAM memories is only possible when the device is not protected by the Code Protect bit.

**Important:** If securing the device using SECCFG.DEBUG_LCK, row programming is not supported because SRAM memory is not accessible by the external debugger. Either Word programming or Quad Word programming is possible based on the CFGCON0.ECCCTL setting.

The programming procedure is as follows:

1. At power-up, $\overline{\text{MCLR}}$ 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 (See *Power-on Reset (POR)* electrical characteristics from Related Links). The system is going to be in this Static state until the internally regulated supplies have reached a safe Operating state.
2. The power management 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.
3. The debugger maintains a low-level on SWCLK. $\overline{\text{MCLR}}$ 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. The user must issue a Chip Erase command to ensure the Flash is fully erased before programming.
7. Programming is available through the AHB-AP. For more details, refer to the *PIC32CX-BZ6Programming Specification*. Additionally, see *Flash Controller* from Related Links.
   **Note:** Programming of Boot Flash Memory (BFM) pages is allowed only when the bit 3 of DSUEXT.TESTMODE at address 0x410001FC is set to '1'.
8. Completing the operation, the chip can be restarted either by asserting $\overline{\text{MCLR}}$, toggling power or writing a '1' to the Status A register CPU Reset Phase Extension bit (STATUSA.CRSTEXT). Ensure that the SWCLK pin is high when releasing $\overline{\text{MCLR}}$ to prevent extending the CPU Reset.


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**
Power-on Reset (POR)
Flash Controller (FC)

## 16.9. Intellectual Property Protection

Intellectual property protection consists of restricting access to internal memories from external tools when the device is protected, and the user can accomplish this by setting the Code Protect bit or DEBUG_LCK bit.

The following are the two protection mechanisms in the PIC32CX-BZ6 devices:

1. Code Protect – If enabling the Code Protect bit by configuring FCPN0.CP in Boot Flash memory, the device is locked from programming and debugging. Only chip erase can retrieve the device to the normal programming and debugging condition.
   **Notes:**
   a. It is not possible to enter Debug Mode when Code Protect is active.
   b. Programming the Flash is only possible when the device is not code protected.
   c. If the device is already code protected, issue chip erase procedure to make the device un-protected and programming ready.
   d. Issue the Chip-Erase command by writing a '1' to DSU.CTRL.CE. The device then:
      - Clears the system volatile memories only when code protect is enabled.
      - Erases the whole Flash array (excluding OTP, TCFG).
2. Secured Device – DEBUG_LCK bits in eFuse (SECCFG.DEBUG_LCK in Root of Trust) determines if the device is locked for debugging. If the DEBUG_LCK bits are non-zero, device is a secured device. Securing of the device implies:
   a. The user cannot execute any unauthenticated firmware.
   b. The debug features of the device are not available and are locked down.
   c. Device programming through SWD interface is still available. Debugger can be plugged in only through the Cold-Plugging procedure. The Hot-Plugging feature is not available (see *Cold-Plugging* and *Hot-Plugging* from Related Links).
   d. The DEBUG_LCK bits are in eFuse (one time programmable memory); therefore, when locked, the device is permanently locked for debug unlike the Code Protect mechanism, which can be cleared on a chip erase.

**Important:** On a secured device (non-zero value of SECCFG.DEBUG_LCK), if the boot key is zero, it implies that secure boot code need not authorize the firmware code. Therefore, programming through an external debugger is disabled on a device, which is secured, and secure boot key is zero.

When the device is protected, read/write accesses using the AHB-AP is limited to the DSU address range and DSU commands are restricted.

The DSU implements a security filter that monitors the AHB transactions generated by the ARM AHB-AP 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. For more details, refer to the *ARM Debug Interface v5 Architecture Specification* on www.arm.com.

The user can access the DSU 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


© 2025 Microchip Technology Inc. and its subsidiaries

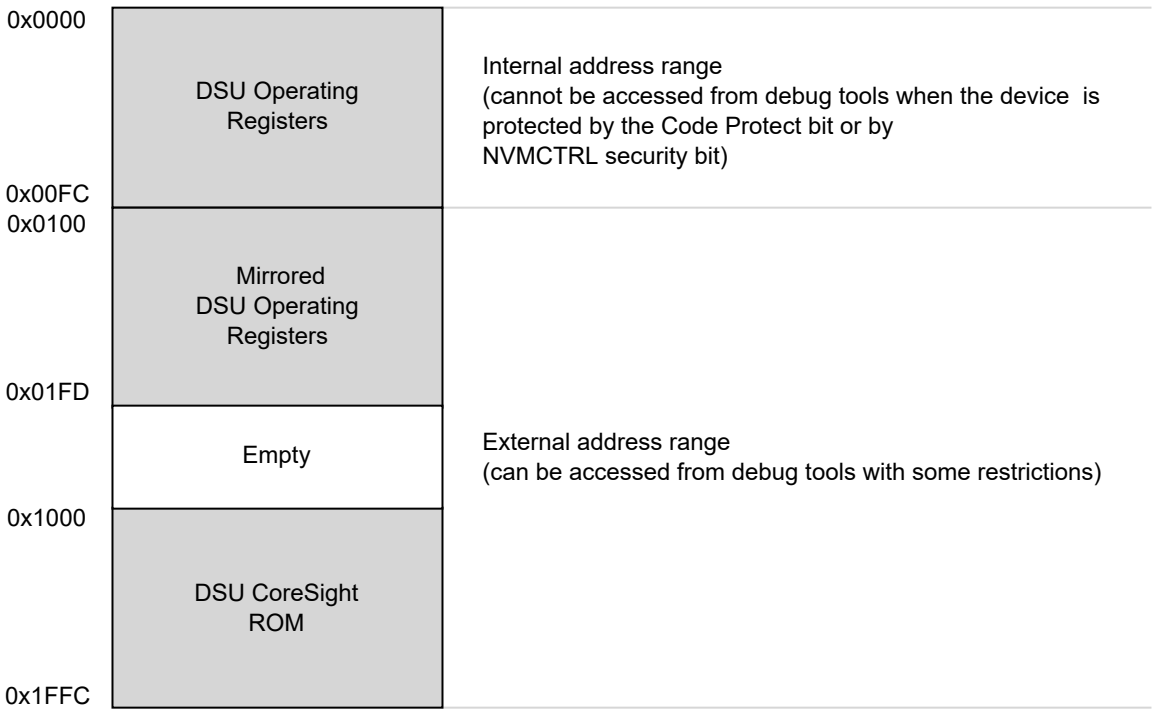
For security reasons, DSU features have limitations when using a debug adapter. To differentiate external accesses from internal ones, the first 0x100 bytes of the DSU register map were mirrored at offset 0x100:

- 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, see Table 16-2.

**Figure 16-4.** APB Memory Mapping




When the device is protected, some features that are not activated by APB transactions are unavailable.

**Table 16-2.** Feature Availability Under Protection

| Feature | Code Protected and Unsecured | Code Protected and Secured | Not Code Protected and Unsecured | Not Code Protected and Secured |
|---|---|---|---|---|
| CPU Reset Extension | Yes | Yes | Yes | Yes |
| Clear CPU Reset Extension | No | No | Yes | No |
| Debugger Cold-Plugging | Yes | Yes | Yes | Yes |
| MEM-AP access during Cold-Plugging | No | No | Yes | Yes |
| Debugger Hot-Plugging | No | No | Yes | No |


© 2025 Microchip Technology Inc. and its subsidiaries

**Notes:**

- Code Protected means FCPN0.CP of the Flash fuse configuration bit is set.
- Secured means SECCFG.DEBUG_LCK bits of the eFuse are non zero values.

**Related Links**

Cold-Plugging

Hot-Plugging

## 16.10. Device Identification

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

### 16.10.1. 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 debug access port. The CoreSight ROM implements a 64-bit conceptual ID composed from the PID0 to PID7 CoreSight ROM table registers. The following figure illustrates the conceptual 64-bit peripheral ID.

**Figure 16-5.** Conceptual 64-bit Peripheral ID

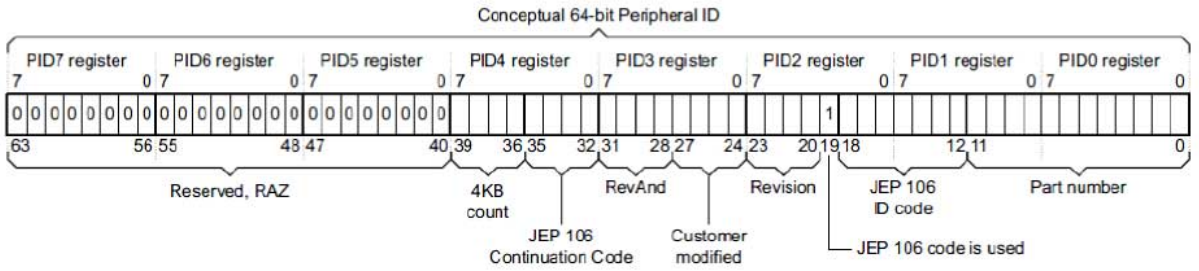



**Table 16-3.** Conceptual 64-Bit Peripheral ID Bit Descriptions

| Field | Size | Description | Location |
|---|---|---|---|
| JEP-106 CC code | 4 | Microchip continuation code: 0x0 | PID4 |
| JEP-106 ID code | 7 | Microchip 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 details, refer to the *ARM Debug Interface Version 5 Architecture Specification*.

### 16.10.2. Chip Identification Method

The following table shows how the DSU DIS register identifies the device:

**Table 16-4.** DSU DID Encoding

| Field | Size | Value | Comments |
|---|---|---|---|
| Revision | 4 bits | 0x0 | Immutable Field (0x0=Rev-A0) |
| Family | 5 bits | 0b00000 | Family[4:0] |
| Series | 6 bits | 0b000001 | Series[5:0] |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 16-4.** DSU DID Encoding (continued)

| Field | Size | Value | Comments |
|---|---|---|---|
| Die | 8 bits | 0xA8 | Immutable Field = Mask ID [7:0] |
| DEVSEL | 8 bits | RoT eFuse (device_id) | VSEL[7:0] determines product variants.<br>0x00 is 132 pins variants |

## 16.11. Functional Description

### 16.11.1. Principle of Operation

The DSU provides memory services, such as CRC32 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 completion of the current operation. Hence, the user must wait for the STATUSA.DONE bit to be set prior to issuing another one.

### 16.11.2. Basic Operation

#### 16.11.2.1. Initialization

The module is enabled by enabling its clocks. See *Clock and Reset Unit (CRU)* from Related Links. The DSU registers can be PAC write-protected. See *Peripheral Access Controller (PAC)* from Related Links.

**Related Links**

Clock and Reset Unit (CRU)

Peripheral Access Controller (PAC)

#### 16.11.2.2. Operation From a Debug Adapter

Debug adapters must access the DSU registers in the external address range 0x100-0x2000. If the device is protected by the Code Protect bit, accessing the first 0x100 bytes causes the system to return an error. See *Intellectual Property Protection* from Related Links.

When the device debug is locked through Secure Device, other than DSU external address range, Flash Controller registers, eFuse controller registers and Flash Memory addresses are placed under MEM-AP permissible address range to allow programming.

**Table 16-5.** Permissible Address Range for External Debugger when Secured Device

| Parameter | Description | Value |
|---|---|---|
| PERM_ADDR_START1 | Permissible address range start | 0x4400_0600 |
| PERM_ADDR_END1 | Permissible address range end | 0x4400_07FF |
| PERM_ADDR_START2 | Second permissible address range start | 0x4400_3000 |
| PERM_ADDR_END2 | Second permissible address range end | 0x4400_3BFF |
| FLASH_ADDR_START | Flash address start | 0x0080_0000 |
| FLASH_ADDR_END | Flash address end | 0x011F FFFF |
| DSU_MBIST_IMPLEMENTED | MBIST Controller Enabled | 1'b0 |

**Related Links**

Intellectual Property Protection

#### 16.11.2.3. Operation From the CPU

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


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**
Intellectual Property Protection

### 16.11.3. 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 the following table)

**Table 16-6.** 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. DATA forced to 0xFFFFFFFF before calculation (no seed). |
| 1-3 | Reserved | — |

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

#### 16.11.3.1. 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). Ensure both are Word-aligned.

The initial value used for the CRC32 calculation must be written to the Data register (DATA). In general, this value is 0xFFFFFFFF, but it can be, for example, the result of generating a common CRC32 of separate memory blocks through previous CRC32 calculation.

After completion, 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 the starting point for subsequent CRC32 calculations.

If the device is in Protected state by the Code Protect or SECCFG.DEBUG_LCK security bit, it is only possible to calculate the CRC32 of the whole flash array when operated from the external address space. In most cases, this area is the entire onboard non-volatile memory. The Address, Length and Data registers are forced to predefined values when the CRC32 operation is started, and values written by the user are ignored. This allows the user to verify the contents of a protected device.

The actual test starts 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).

#### 16.11.3.2. Interpreting the Results

The user must monitor the Status A register. After completing the operation, 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.

### 16.11.4. 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 Code Protect bit or SECCFG.DEBUG_LCK 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.


© 2025 Microchip Technology Inc. and its subsidiaries

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 is 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.

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

**Table 16-7.** Available Features when Operated from External Address Range and the Device is Protected

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


© 2025 Microchip Technology Inc. and its subsidiaries

## 16.12. Register Summary

See *DSU* module in the *Product Memory Mapping Overview* from Related Links for base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRL | 7:0 | | | | CE | | | | 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[7:0] | | | | | | | |
| | | 23:16 | FAMILY[0] | | SERIES[5:0] | | | | | |
| | | 31:24 | REVISION[3:0] | | | | FAMILY[4:1] | | | |
| 0x1C | CFG | 7:0 | | | | ETBRAMEN | DCCDMALEVEL[1:0] | | LQOS[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 ... 0x37 | Reserved | | | | | | | | | |
| 0x38 | UUID0 | 7:0 | UUID[7:0] | | | | | | | |
| | | 15:8 | UUID[15:8] | | | | | | | |
| | | 23:16 | UUID[23:16] | | | | | | | |
| | | 31:24 | UUID[31:24] | | | | | | | |
| 0x3C | UUID1 | 7:0 | UUID[7:0] | | | | | | | |
| | | 15:8 | UUID[15:8] | | | | | | | |
| | | 23:16 | UUID[23:16] | | | | | | | |
| | | 31:24 | UUID[31:24] | | | | | | | |
| 0x40 | UUID2 | 7:0 | UUID[7:0] | | | | | | | |
| | | 15:8 | UUID[15:8] | | | | | | | |
| | | 23:16 | UUID[23:16] | | | | | | | |
| | | 31:24 | UUID[31:24] | | | | | | | |
| 0x44 | UUID3 | 7:0 | UUID[7:0] | | | | | | | |
| | | 15:8 | UUID[15:8] | | | | | | | |
| | | 23:16 | UUID[23:16] | | | | | | | |
| | | 31:24 | UUID[31:24] | | | | | | | |
| 0x48 | SECCFG | 7:0 | DEBUG_LCK[1:0] | | UUID_LCK[1:0] | | | | | |
| | | 15:8 | | | | | BOOT_KEY_LCK[1:0] | | ROOT_KEY_LCK[1:0] | |
| | | 23:16 | | | | | | | | ADD_BOOT_KEY |
| | | 31:24 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x4C | CTR_STAT | 7:0 | ROLLBACK_CTR[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x50 | BOOT_STATUS | 7:0 | BOOT_STATUS[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x54 | BOOT_KEY0 | 7:0 | BOOT_KEY[7:0] | | | | | | | |
| | | 15:8 | BOOT_KEY[15:8] | | | | | | | |
| | | 23:16 | BOOT_KEY[23:16] | | | | | | | |
| | | 31:24 | BOOT_KEY[31:24] | | | | | | | |
| ... | | | | | | | | | | |
| 0x80 | BOOT_KEY11 | 7:0 | BOOT_KEY[7:0] | | | | | | | |
| | | 15:8 | BOOT_KEY[15:8] | | | | | | | |
| | | 23:16 | BOOT_KEY[23:16] | | | | | | | |
| | | 31:24 | BOOT_KEY[31:24] | | | | | | | |
| 0x84 ... 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 | | | | | | | | | |
| 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 | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 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 | | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 16.13. 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. See *Register Access Protection* from Related Links.

**Related Links**

Register Access Protection


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.1. Control

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

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | CE | | | | SWRST |
| Access | | | | W | | | | W |
| Reset | | | | 0 | | | | 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 0 – SWRST** Software Reset
Writing a '0' to this bit has no effect.
Writing a '1' to this bit resets the module.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.2. Status A

**Name:** STATUSA
**Offset:** 0x01
**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.3. Status B

**Name:** STATUSB
**Offset:** 0x02
**Reset:** 0x0x
**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 | | | | 0 | 0 | 0 | x | x |

**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-on Reset or an external Reset can set it again.

**Bits 2, 3 – DCCD** Debug Communication Channel x Dirty
Writing a '0' to this bit has no effect.
Writing a '1' to this bit has no effect.
This bit is set when DCC is written.
This bit is cleared when DCC 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 code protected and FCPN0.CP bit in Devconfig Boot Flash memory is set to enable code protection. See *FCPN0* from Related Links.
This bit is never cleared.

**Related Links**
FCPN0
Boot Flash Code Protection Register


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.4. Address

**Name:** ADDR
**Offset:** 0x04
**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 (see *32-bit Cyclic Redundancy Check (CRC32)* from Related Links).

**Related Links**
32-bit Cyclic Redundancy Check (CRC32)


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.5. Length

**Name:** LENGTH
**Offset:** 0x08
**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.6. Data

**Name:** DATA
**Offset:** 0x0C
**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.7. Debug Communication Channel x

**Name:** DCC
**Offset:** 0x10 + n*0x04 [n=0..1]
**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.8. Device Identification

**Name:** DID
**Offset:** 0x18
**Property:** PAC Write Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | REVISION[3:0] | | | | FAMILY[4:1] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | r | r | r | r | 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[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | d | d | d | d | d | d | d | d |

| 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 – REVISION[3:0]** Revision Number
Identifies the die revision number. 0x0 = rev.A, 0x1 = rev.B and so on.
**Note:** The device variant (last letter of the ordering number) is independent of the die revision.

**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 15:8 – DIE[7:0]** Die Number
Identifies the die family. 0xA8 for PIC32CX-BZ6 family of devices.

**Bits 7:0 – DEVSEL[7:0]** Device Selection
This bit field identifies a device within a product family and product series. The value corresponds to the Flash memory density, pin count and device variant parts of the ordering code. See *DSU DID Encoding* table in *Chip Identification Method* from Related Links for the DEVSEL[7:0] values of different variants.

**Related Links**
Chip Identification Method


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.9. Configuration

**Name:** CFG
**Offset:** 0x1C
**Reset:** 0x00000002
**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 |
|---|---|---|---|---|---|---|---|---|
| | | | | ETBRAMEN | DCCDMALEVEL[1:0] | | LQOS[1:0] | |
| Access | | | | R/W | R/W | R/W | R/W | R/W |
| Reset | | | | 0 | 0 | 0 | 1 | 0 |

**Bit 4 – ETBRAMEN** Trace Control
ETB Ram Enable Writing a one to this bit will reserve the first 32 KB of the RAM for the trace ETB ram buffer.

**Bits 3:2 – DCCDMALEVEL[1:0]** DMA Trigger Level

| Value | Description |
|---|---|
| 0x0 | DMA trigger rises when DCC is empty |
| 0x1 | DMA trigger rises when DCC is full |
| 0x2 – 0x3 | Reserved |

**Bits 1:0 – LQOS[1:0]** Latency Quality Of Service
These bits define the priority access during the memory access.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.10. Unique Identifier Register n

**Name:** UUID
**Offset:** 0x38 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** -

These registers contain the unique identifier of the device. It is stored in eFuse memory in Root of Trust module, which is directly driven in this register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | UUID[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 |
|---|---|---|---|---|---|---|---|---|
| | UUID[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 |
|---|---|---|---|---|---|---|---|---|
| | UUID[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 |
|---|---|---|---|---|---|---|---|---|
| | UUID[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – UUID[31:0]** Unique Identifier bits
These bits provide the unique identifier value. The four 32-bit UUID registers contain the 128-bit UUID.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.11. Secure Configuration

**Name:** SECCFG
**Offset:** 0x48
**Reset:** 0x00000000
**Property:** -

This register contains the secure configuration setting of the device. It is stored in eFuses memory in Root of Trust module, which is directly driven in this register.

**Note:**
*_LCK bits in this register refer to the program locks of corresponding eFuses.

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | ADD_BOOT_KEY |
| Access | | | | | | | | R |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | BOOT_KEY_LCK[1:0] | | ROOT_KEY_LCK[1:0] | |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

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

**Bit 16 – ADD_BOOT_KEY** Additional Boot Key
This bit is the LSB of the Y co-ordinate in the compressed Secure Boot Key.

**Bits 11:10 – BOOT_KEY_LCK[1:0]** Lock Bits for Secure Boot Key

| Value | Description |
|---|---|
| 11 | Secure boot key is locked and cannot be programmed |
| 10 | Secure boot key is locked and cannot be programmed |
| 01 | Secure boot key is locked and cannot be programmed |
| 00 | Secure boot key is not locked |

**Bits 9:8 – ROOT_KEY_LCK[1:0]** Lock Bits for Storage Root Key

| Value | Description |
|---|---|
| 11 | Storage root key is locked and cannot be programmed |
| 10 | Storage root key is locked and cannot be programmed |
| 01 | Storage root key is locked and cannot be programmed |
| 00 | Storage root key is not locked |

**Bits 7:6 – DEBUG_LCK[1:0]** Lock Bits for Debug


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 11 | Debug is locked. Not possible to debug. |
| 10 | Debug is locked. Not possible to debug. |
| 01 | Debug is locked. Not possible to debug. |
| 00 | Debug is not locked. |

**Bits 5:4 – UUID_LCK[1:0]** Programming Lock Bits for Unique ID Fuses

| Value | Description |
|---|---|
| 11 | Unique ID is locked and cannot be programmed |
| 10 | Unique ID is locked and cannot be programmed |
| 01 | Unique ID is locked and cannot be programmed |
| 00 | Unique ID is not locked |


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.12. Rollback Counter Status

**Name:** CTR_STAT
**Offset:** 0x4C
**Reset:** 0x0000009C
**Property:** -

This register has Rollback Counter information. It is stored in eFuse memory in Root of Trust module which is directly driven in this 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

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

**Bits 7:0 – ROLLBACK_CTR[7:0]** Rollback Counter status
This bit contains the Rollback Counter status bit.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.13. Boot Status

**Name:** BOOT_STATUS
**Offset:** 0x50
**Reset:** 0x0
**Property:** -

This register reflects the SEC_BOOT.BOOT_STATUS bits in Root of Trust module. The secure boot firmware in ROM manages the SEC_BOOT.BOOT_STATUS bits to indicate secure boot status.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_STATUS[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 7:0 – BOOT_STATUS[7:0]**
These bits hold 8-bit code which is written to SEC_BOOT.BOOT_STATUS bits to indicate the secure boot status, such as authentication success, failure or any other indication.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.14. Secure Boot Key n

**Name:** BOOT_KEY
**Offset:** 0x54 + n*0x04 [n=0..11]
**Reset:** 0x00000000
**Property:** -

These registers read the secure boot key. The secure boot key is stored in eFuse in the Root of Trust module, which is directly driven in this register. The twelve 32-bit BOOT_KEY registers contain the 384-bit secure public boot key.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_KEY[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 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_KEY[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 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_KEY[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 |
|---|---|---|---|---|---|---|---|---|
| | BOOT_KEY[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – BOOT_KEY[31:0]** Secure Boot key bits
These bits provide the secure boot key.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.15. CoreSight ROM Table Entry x

**Name:** ENTRY
**Offset:** 0x1000 + n*0x04 [n=0..1]
**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 code protected indicating that the entry is not present.
This bit is cleared at power-up if the device is not code protected indicating that the entry is present.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.16. CoreSight ROM Table End

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

**Table 16-8.** Register Bit Attribute Legend

| Symbol | Description | Symbol | Description | Symbol | Description |
|---|---|---|---|---|---|
| **R** | Readable bit | **HC** | Cleared by Hardware | (Grey cell) | Unimplemented |
| **W** | Writable bit | **HS** | Set by Hardware | **X** | Bit is unknown at Reset |
| **K** | Write to clear | **S** | Software settable bit | — | — |

| 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.17. 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 code protected, indicating that the system memory is accessible from a debug adapter.
This bit is cleared at power-up if the device is code protected, indicating that the system memory is not accessible from a debug adapter.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.18. 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]** 4 KB Count
These bits will always return zero when read, indicating that this debug component occupies one 4 KB block.

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


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.19. Peripheral Identification 0

**Name:** PID0
**Offset:** 0x1FE0
**Reset:** 0x000000D0
**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 |
|---|---|---|---|---|---|---|---|---|
| | PARTNBL[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |

**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.20. Peripheral Identification 1

**Name:** PID1
**Offset:** 0x1FE4
**Reset:** 0x0000009C
**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 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |

**Bits 7:4 – JEPIDCL[3:0]** Low Part of the JEP-106 Identity Code
These bits will always return 0x9 when read (JEP-106 identity code is 0x29).

**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.21. Peripheral Identification 2

**Name:** PID2
**Offset:** 0x1FE8
**Reset:** 0x0000000A
**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 | 1 | 0 |

**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 '1' 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 0x2 when read, (JEP-106 identity code is 0x29).


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.22. Peripheral Identification 3

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

**Table 16-9.** Register Bit Attribute Legend

| Symbol | Description | Symbol | Description | Symbol | Description |
|---|---|---|---|---|---|
| **R** | Readable bit | **HC** | Cleared by Hardware | (Grey cell) | Unimplemented |
| **W** | Writable bit | **HS** | Set by Hardware | **X** | Bit is unknown at Reset |
| **K** | Write to clear | **S** | Software settable bit | — | — |

| 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 |
|---|---|---|---|---|---|---|---|---|
| | REVAND[3:0] | | | | CUSMOD[3:0] | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.23. Component Identification 0

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

**Table 16-10.** Register Bit Attribute Legend

| Symbol | Description | Symbol | Description | Symbol | Description |
|---|---|---|---|---|---|
| **R** | Readable bit | **HC** | Cleared by Hardware | (Grey cell) | Unimplemented |
| **W** | Writable bit | **HS** | Set by Hardware | **X** | Bit is unknown at Reset |
| **K** | Write to clear | **S** | Software settable bit | — | — |

| 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.24. 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 (For more details, 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 0x0 when read.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.25. Component Identification 2

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

**Table 16-11.** Register Bit Attribute Legend

| Symbol | Description | Symbol | Description | Symbol | Description |
|---|---|---|---|---|---|
| **R** | Readable bit | **HC** | Cleared by Hardware | (Grey cell) | Unimplemented |
| **W** | Writable bit | **HS** | Set by Hardware | **X** | Bit is unknown at Reset |
| **K** | Write to clear | **S** | Software settable bit | — | — |

| 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 16.13.26. Component Identification 3

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

**Table 16-12.** Register Bit Attribute Legend

| Symbol | Description | Symbol | Description | Symbol | Description |
|---|---|---|---|---|---|
| **R** | Readable bit | **HC** | Cleared by Hardware | (Grey cell) | Unimplemented |
| **W** | Writable bit | **HS** | Set by Hardware | **X** | Bit is unknown at Reset |
| **K** | Write to clear | **S** | Software settable bit | — | — |

| 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.


© 2025 Microchip Technology Inc. and its subsidiaries

# 17. Clock and Reset Unit (CRU)

## 17.1. Overview

The Clock and Reset Unit (CRU) provides both clocking and Reset functions. This section describes the clocking and Reset functionality, summarizes the clock distribution and terminology in the PIC32CX-BZ6 device. The CRU handles the clock control to provide system clocks and interface peripheral clocks. The clock is distributed to a different peripheral through peripheral-specific configuration. The CRU controls switching and synchronization of clock sources.

## 17.2. Features

The Clock and Reset Unit has the following features:

- Supports the Following as System Clock Sources:
  - 16 MHz Primary Crystal Oscillator (POSC)
  - 8 MHz Fast RC Oscillator (FRC)
  - 32 kHz Low Power RC Oscillator (LPRC)
  - 32.768 kHz Secondary Crystal Oscillator (SOSC)
  - 128 MHz System Phase-Locked Loop (RFPLLPGM MHz)
  - USB Phase-Locked Loop (UPLL) Clock
  - Ethernet Phase-Locked Loop (EPLL) Clock
- Provides Control Registers for PLL
- Provides Glitch-Free Clock Switching Between Various Clock Sources
- Post Dividers on Processor Clock Generator to Slow Down System Clock for Power Save
- A Fail Safe Clock Monitor that Detects Clock Failure and Provides Automatic Switching to the FRC
- Provides Control Registers for the User Interface of Clocks and Resets
- Provides Configuration Bits for Oscillator Selection and Calibration of On-Chip Oscillators
- Provides Control Registers to Generate a Reference Clock Output
- Provide Resets for the System
- Provides NMI for the System
- Multiple PB Clock (Peripheral Clock) Dividers
- One System Clock, SYS_CLK, which Almost All Clocks Used Throughout the System are Derived from
- Five Peripheral Clocks, Created by Independent Integer Dividers of the SYS_CLK:
  - PB1_CLK: PB-PIC and PB-Bridge-A
  - PB2_CLK: PB-Bridge-B and PB-Bridge-C
  - PB3_CLK: DS/XDS Bus Clock
  - PB4_CLK: PB-Bridge-D
  - PB5_CLK: PB-Bridge-B (Wireless ZBT only)
- Six Reference Output Clocks (REFO1 – REFO6) with the Following Clock Sources:
  - 96 MHz USB PLL (UPLL)
  - 50 MHz Ethernet PLL (EPLL)
  - System clock (SYS_CLK)
  - PB1 Bus Clock (PB1_CLK)


© 2025 Microchip Technology Inc. and its subsidiaries

  - 16 MHz Primary Crystal Oscillator (POSC)
  - 8 MHz Fast RC Oscillator (FRC)
  - 32 kHz Low Power RC Oscillator (LPRC)
  - 32.768 kHz Secondary Crystal Oscillator (SOSC)
  - 128 MHz system PLL (RFPLL PGM MHz), SPLL_CLK1
  - REFI pin
- Provides Clock Source for Backup Core for Sleep Operations

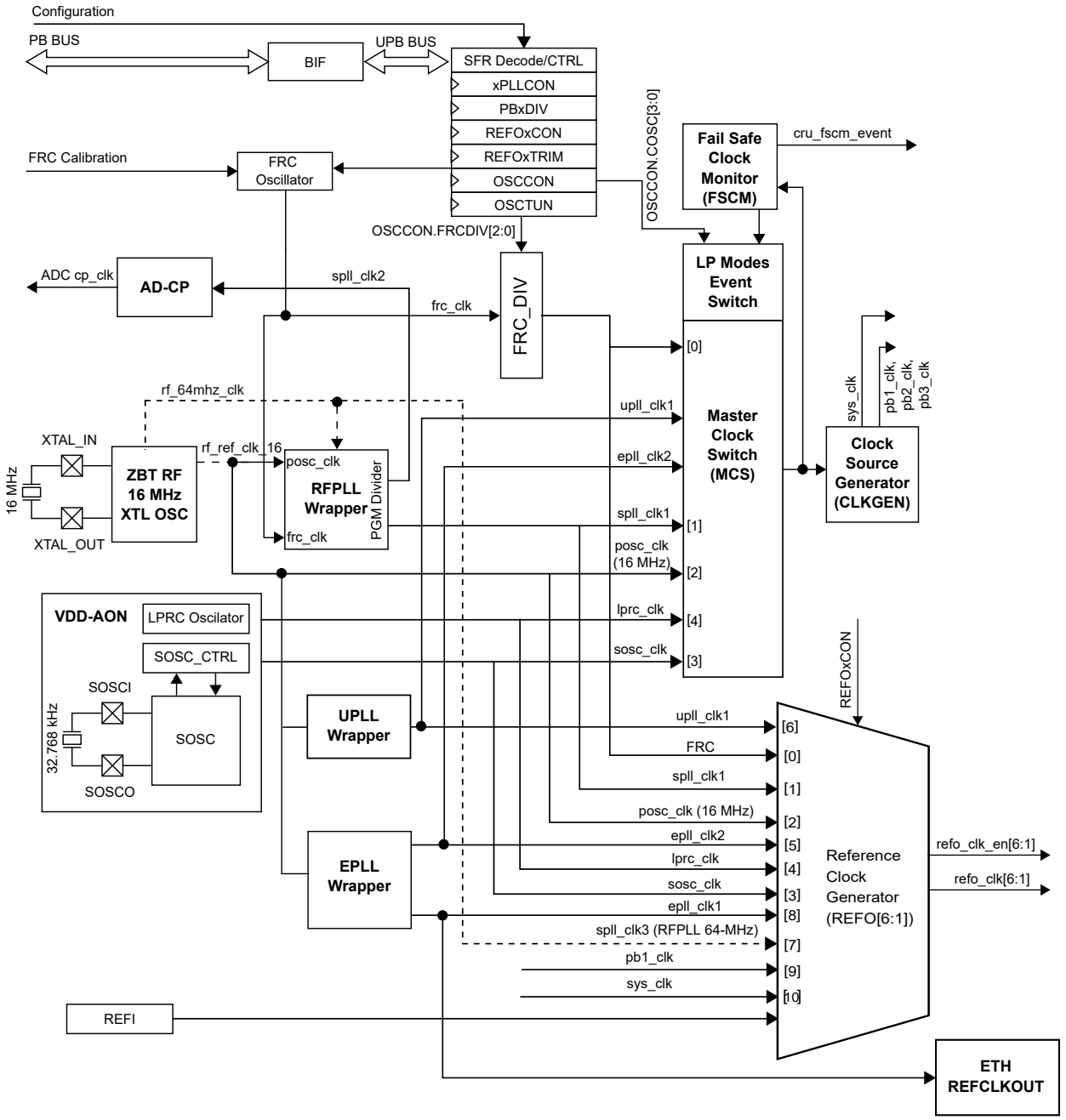
## 17.3. Clock System

### 17.3.1. Block Diagram

The clock system along with the PMD provides gated clock output for all peripheral buses. See *Peripheral Module Disable* from Related Links. The following figure illustrates the clock system block diagram.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 17-1.** Clock System Block Diagram





© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 17-2.** RFPLL Wrapper

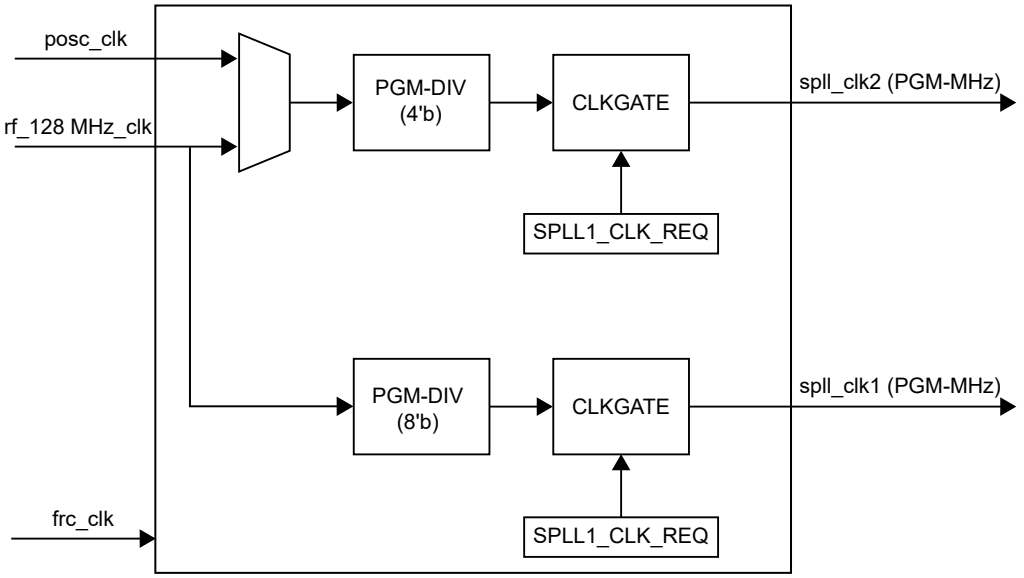
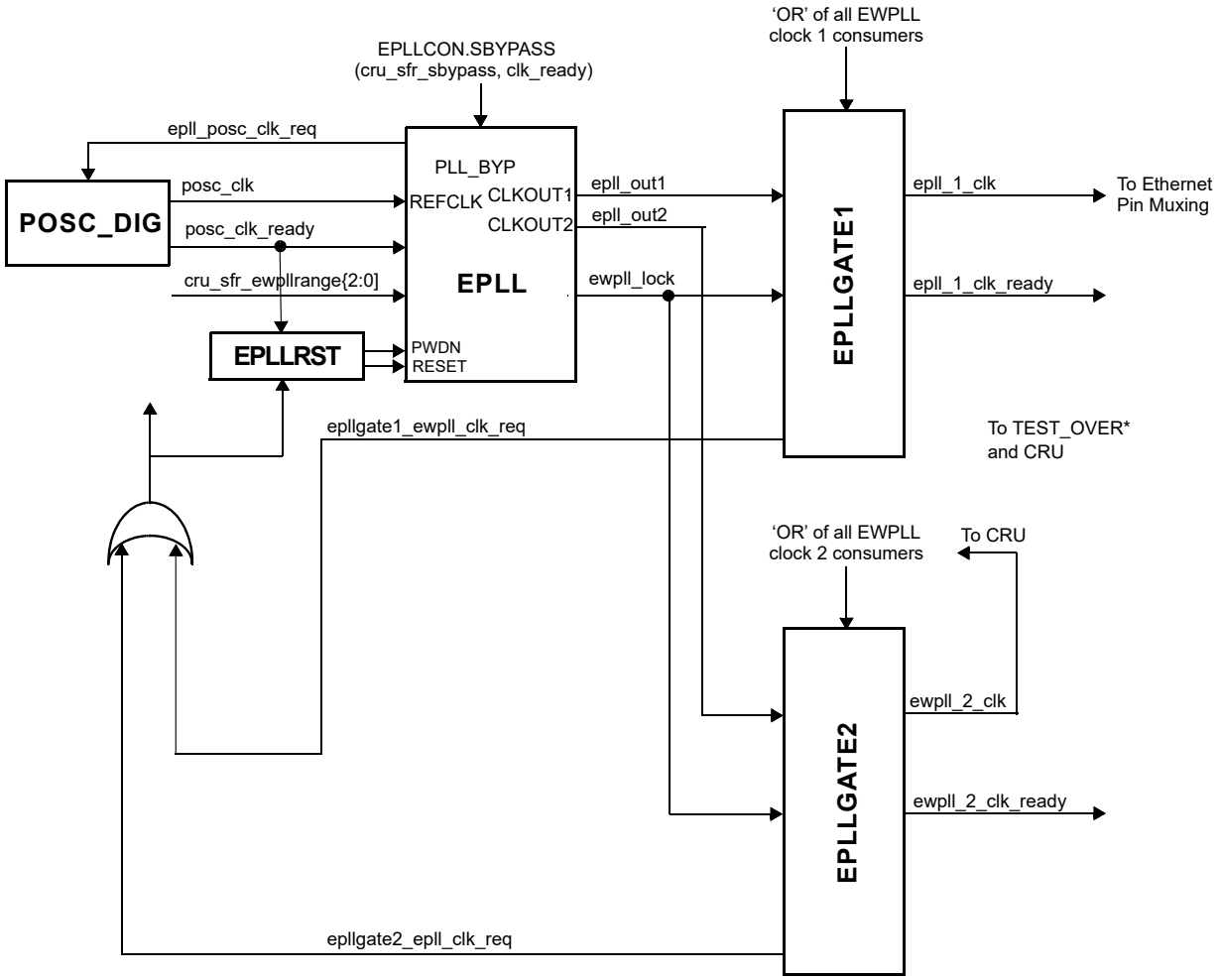



**Figure 17-3.** Ethernet PLL Subsystem





© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 17-4.** USB PLL Subsystem

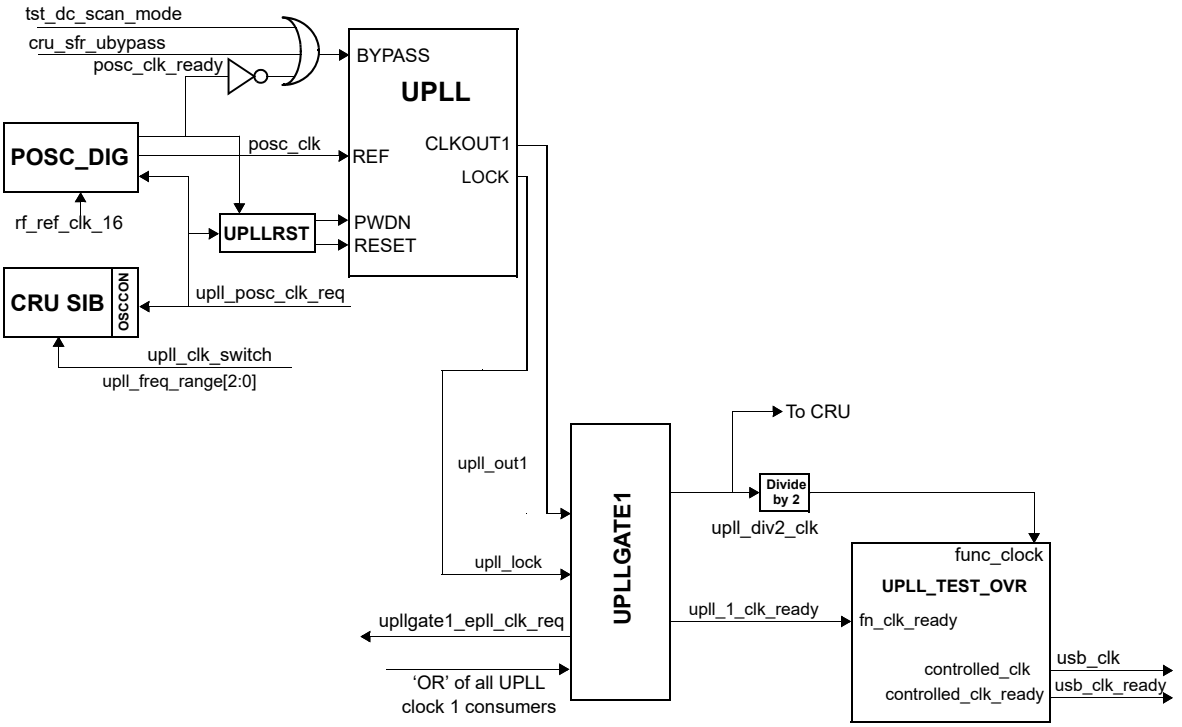
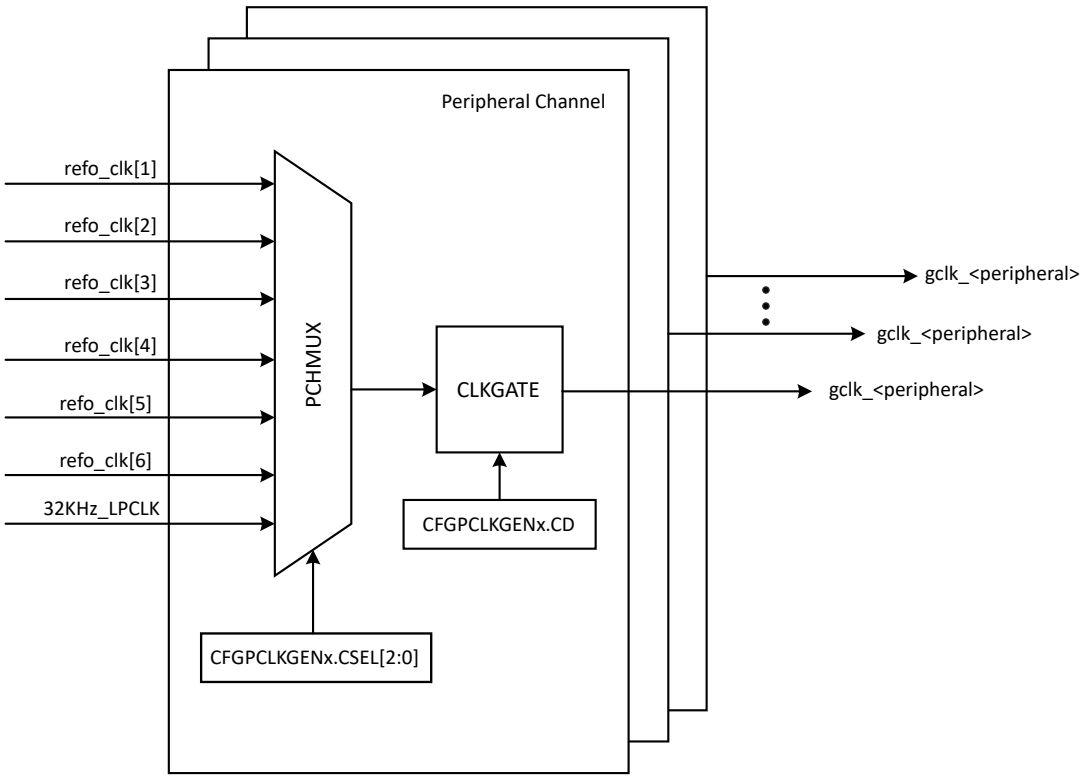


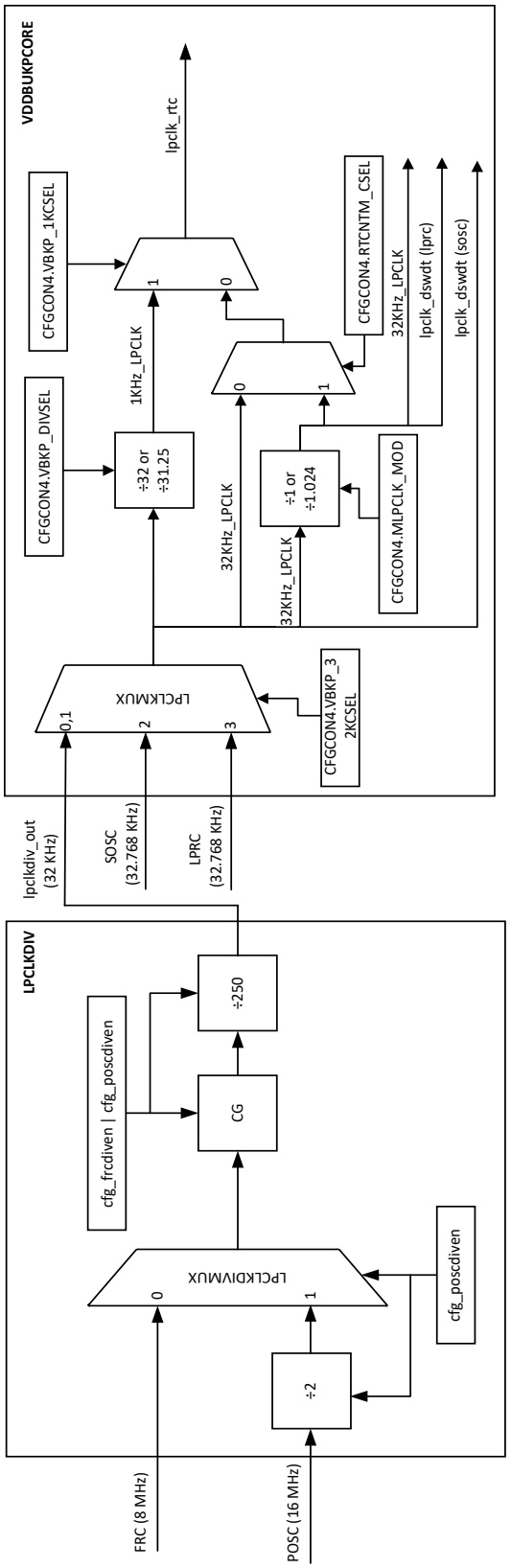

**Figure 17-5.** Peripheral Clock Generation (GCLK)





© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 17-6.** Low Power Clock Generation (LPCLK)





© 2025 Microchip Technology Inc. and its subsidiaries

The CRU Master Clock Switch (MCS) selects which input clock is fed to the CLKGEN Synchronous Clock Generator. The CLKGEN generates and controls the synchronous clocks on the system. This includes the CPU, bus clocks (APB and AHB) and the synchronous (to the CPU) user interfaces of the peripherals. It contains prescalers for the CPU and bus clocks.

**Related Links**

Peripheral Module Disable (PMD)

## 17.3.2. Oscillators

### 17.3.2.1. Fast RC Oscillator (FRC)

The on-chip 8 MHz Fast RC Oscillator (FRC) is a fast, with precise frequency, internal RC oscillator. The FRC oscillator is accurate to provide the clock frequency, which is necessary to maintain the baud rate tolerance for serial data transmissions. Power-on Reset (POR) sets OSCCON.NOSC[3:0] = 0000; hence, the device starts with FRC when powered up.

The oscillator module provides a 6-bit wide user tuning adjustment using the OSCTRIM.TUN[5:0] bits.

#### 17.3.2.1.1. Enabling the FRC

The FRC oscillator is powered in the following conditions:

- When OSCCON.NOSC[3:0] = 4'b0000 or a fail-safe clock monitor is enabled and a clock fail is detected, forcing a switch to FRC.
- When SPLL_CLK2 of ADC PMU Controller, Flash controller and 32KHz_LPCLK (32 KHz) request it as the source.

#### 17.3.2.1.2. Frequency Tuning in User Mode

In addition to the factory calibration, the user can tune base frequency in their application. This frequency tuning capability allows the user to deviate from the factory-calibrated frequency. The user can tune the frequency by writing to the OSCTRIM.TUN[5:0] register bits.

### 17.3.2.2. Low Power RC Oscillator (LPRC)

The Low Power Internal RC Oscillator (LPRC) operates at a nominal frequency of 32.768 kHz.
**Note:** The LPRC is not a 50% duty cycle clock; however, it maintains an average frequency over a number of base clocks.

The LPRC can be used as both a source for the system clock and a reference for Backup core modules. These modules include the Deep Sleep Watchdog (DSWDT), clock monitor circuits and other modules that require a 32 kHz reference clock.

### 17.3.2.3. Primary Oscillator (POSC)

A 16 MHz ±30 ppm crystal/resonator oscillator or external clock is the primary oscillator for the 2.4 GHz RF transceiver. This is the input clock source for the system PLL, providing up to a 128 MHz clock.

### 17.3.2.4. Secondary Oscillator (SOSC)

The Secondary Oscillator (SOSC) is a low-power 32.768 kHz crystal oscillator, which provides accurate time keeping.

The following are the features of the secondary oscillator:

- 32.768 kHz Operation
- Provides System Clock (SYS_CLK) Output
- Provides Source to CRU or LPCLKGEN on Request
- Provides Clock for the Low Power Mode


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.3.3. System and Peripheral Bus Clock Generation (CLKGEN)

The CLKGEN module generates and controls the 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 prescalers for the CPU and bus clocks.

There are two types of clocks generated by this module, called system clocks and peripheral clocks:

- The system clock (SYS_CLK used by the CPU supports components such as memory subsystems and fast peripherals.
- The peripheral bus clocks (
  PB1_CLK, PB2_CLK, PB3_CLK, PB4_CLK and PB5_CLK
  ) are used to clock slow peripheral devices attached to the pb_bus.

The PBx_CLK outputs are based on the SYS_CLK frequency with a fixed divisor. The divisor is determined by the value of the PBxDIV registers.

The system and peripheral bus clocks are stopped when in Sleep mode. The clocks are restarted by disabling the sleep enable.

### 17.3.4. FRC Divider (FRCDIV)

The OSCCON.FRCDIV[2:0] register enables the division of the Fast RC (FRC) oscillator, allowing it to be used as a system clock and REFO clock through divider settings. The divisor is configured for eight divider selections: /1, /2, /4, /8, /16, /32, /64, /256.

### 17.3.5. RFPLL Wrapper

The output of RFPLL wrapper is used as a system clock or REFO clock source. Selection of the system clock source is performed with the OSCCON.NOSC[3:0] register field.

The RFPLL wrapper generates two clocks:

- SPLL_CLK1 (PGM MHz)
  - Clock frequencies = 128 MHz/(1-255) frequency choices
  - Clock ready indication
- SPLL_CLK2 (PGM MHz)
  - Clock frequencies = 128 MHz/(1-15) and optional clock disable option
  - Clock ready indication

Clocks are produced only when the consumer generates a request; CRU is the consumer for SPLL_CLK1 (SPLL_CLK1 is chosen through OSCCON.NOSC[3:0]) and ADC charge pump is the consumer for SPLL_CLK2, (SPLLCON.SPLL_BYP[1:0]). Along with the clocks, individual clock ready is also generated, which indicates that clocks are ready for consumption.

### 17.3.6. Ethernet PLL (EPLL)

The Ethernet PLL provides 50 MHz for the Ethernet PHY, which operates at 100 Mbit with two bits between the digital and PHY layers. The output should be 50 MHz when this PLL is used as the Ethernet clock source.

The clock output frequency can be obtained through EPLL feedback divider, reference frequency divider and the first post divider. VCO is the frequency before EPLL post divider and the value is calculated as:

*VCO = Ref. Clock * EPLL Feedback Divider / Reference Frequency Divider*

The EPLL output is then given by:

*EPLL OUT = VCO / First Post Divider*

There are two first post divider which corresponds to EPLL OUT1 and EPLL OUT2.


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 17-1.** Configuration Parameters and Output Frequencies for EPLL

| Ref. Clock | Reference Frequency Divider | EPLL Feedback Divider | VCO (MHz) | Post Divider 1/2 | EPLL OUT 1/2 (MHz) | Closed Loop Bandwidth | Bandwidth Select |
|---|---|---|---|---|---|---|---|
| 16 | 1 | 100 | 1600 | 32 | 50 | 1 MHz | 010 |
| 16 | 1 | 75 | 1200 | 24 | 50 | 1 MHz | 010 |
| 16 | 2 | 200 | 1600 | 32 | 50 | 400 KHz | 001 |
| 16 | 2 | 150 | 1200 | 24 | 50 | 400 KHz | 001 |
| 16 | 2 | 100 | 800 | 16 | 50 | 400 KHz | 001 |
| 16 | 4 | 400 | 1600 | 32 | 50 | 400 KHz | 001 |
| 16 | 4 | 300 | 1200 | 24 | 50 | 400 KHz | 001 |
| 16 | 4 | 200 | 800 | 16 | 50 | 400 KHz | 001 |

Where,

- EPLL Feedback Divider is EPLLCON.EPLLREFDIV[5:0]
- Reference Frequency Divider is EPLLCON.EPLLFBDIV[9:0]
- Post Divider 1 is EPLLCON.EPLLPOSTDIV1[5:0]
- Post Divider 2 is APLLCON.EPLLPOSTDIV2[5:0]
- Bandwidth select is EPLLBSWSEL[2:0]

### 17.3.7. USB PLL

The USB PLL (UPLL) provides the 96 MHz double sampling clock (2*48 MHz) for the Full Speed USB module. It also manages power dissipation by turning the USB clock source on or off. The usb_clock output is enabled when the USB module is active, not suspended, and not in Sleep mode. The output should be 96 MHz when this PLL is used as the USB clock source. The FRC cannot be used as the reference clock source for the UPLL.

The UPLL output can be obtained through the UPLL feedback divider, reference frequency divider, and the first post-divider. The VCO frequency is the frequency before the UPLL post-divider, calculated as:

*VCO = Ref. Clock * UPLL Feedback Divider / Reference frequency Divider*

The UPLL output is then given by:

*UPLL OUT = VCO / First Post Divider*

**Table 17-2.** Configuration Parameters and Output Frequencies for UPLL

| Ref. Clock | Reference Frequency Divider | UPLL Feedback Divider | VCO (MHz) | Post Divider | UPLL OUT (MHz) | CLBW | Bandwidth Select |
|---|---|---|---|---|---|---|---|
| 16 | 1 | 96 | 1536 | 16 | 96 | 1 MHz | 010 |
| 16 | 1 | 72 | 1152 | 12 | 96 | 1 MHz | 010 |
| 16 | 2 | 192 | 1536 | 16 | 96 | 400 KHz | 001 |
| 16 | 2 | 144 | 1152 | 12 | 96 | 400 KHz | 001 |
| 16 | 2 | 384 | 1536 | 16 | 96 | 400 KHz | 001 |
| 16 | 4 | 288 | 1152 | 12 | 96 | 400 KHz | 001 |

Where,

- UPLL Feedback Divider is UPLLCON.UPLLFBDIV[9:0]
- Reference Frequency Divider is UPLLCON.UPLLREFDIV[5:0]
- First Post Divider is UPLLCON.UPLLPOSTDIV1[5:0]


© 2025 Microchip Technology Inc. and its subsidiaries

- Bandwidth select is UPLLBSWSEL[2:0]

### 17.3.8. Reference Clock (REFO_CLK) Generation

The reference clock generator module uses multiple clock sources as input source and generates six different reference clock outputs.

The REFOxCON registers are used to configure the input clock source and divisor.

The clock sources for the reference clock generator:

- System clock (SYS_CLK)
- PB1 bus clock (PB1_CLK)
- 16 MHz Primary Crystal Oscillator (POSC)
- 8 MHz Fast RC Oscillator (FRC)
- 32 kHz Low Power RC Oscillator (LPRC)
- 32.768 kHz Secondary Crystal Oscillator (SOSC)
- 128 MHz system PLL (RFPLL PGM MHz, SPLL_CLK1, SPLL_CLK3)
- 96 MHz USB PLL (UPLL)
- 50 MHz Ethernet PLL (EPLL)
- REFI pin
  - External clock input (REFI) is provided on anyone of the supported I/O pins. For details on supported input pins, see *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.

The following table lists the clock's source mapping for both the CLKGEN generator and REFO_CLK generator.

**Table 17-3.** CRU Source and Output Clock Mapping

| Clock Source | CLKGEN Selection (OSCCON.NOSC[3:0]) | REFO Selection (REFOxCON.ROSEL[3:0]) |
|---|---|---|
| FRC | 0000 | 0000 |
| SPLL_CLK1 | 0001 | 0001 |
| POSC (16 MHz) | 0010 | 0010 |
| SOSC | 0011 | 0011 |
| LPRC | 0100 | 0100 |
| SPLL_CLK3 (RFPLL, 128 MHz) | — | 0111 |
| PB1_CLK | — | 1001 |
| SYS_CLK | — | 1010 |
| REFI Pin | — | 1011 |
| EPLL_CLK2 | 0101 | 0101 |
| UPLL_CLK1 | 0110 | 0110 |
| EPLL_CLK1 | — | 1000 |

**Related Links**

I/O Ports and Peripheral Pin Select (PPS)

### 17.3.9. Peripheral Clock Generation (GCLK)

All six reference clock generator outputs are given as input for GCLK generator module for peripheral clock generation. The GCLK module provides selection among following clocks:

- REFO1 to REFO6 clocks
  **Note:** Only REFO1 – REFO4 can get routed to chip IOs.
- Low-power clock (32KHz_LPCLK) (either LPRC, SOSC or 32 KHz clock derived from POSC/FRC)


© 2025 Microchip Technology Inc. and its subsidiaries

The GCLK generator provides the Generic Clocks (GCLK_<Periperhal>) for system peripherals via peripheral channels. There are a total of 26 peripheral channels with the mapping as shown in the following table. The peripheral channels are fixed configuration and mapped in the CFGPCLKGENx register. CFGPCLKGENx dictates the peripheral clock selection and enables the clock for a specific peripheral. For details about the position in CFGPCLKGENx, see *Register Summary* of the *System Configuration and Register Locking (CFG)* from Related Links.

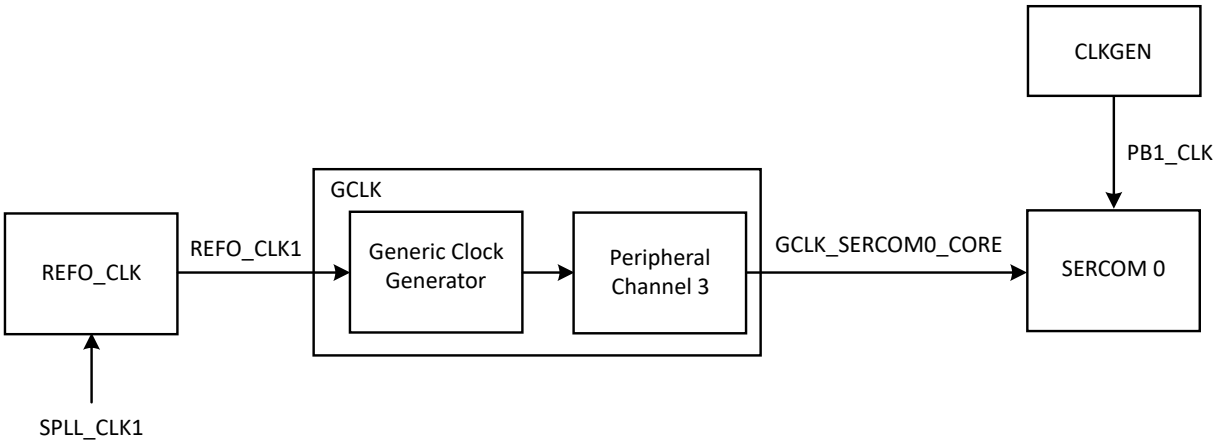
**Table 17-4.** Peripheral Clock Generation

| Peripheral Clock | Channel Index |
|---|---|
| GCLK_EIC, GCLK_CCL | 0 |
| GCLK_FREQM_MSR | 1 |
| GCLK_FREQM_REF | 2 |
| GCLK_SERCOM2_CORE | 3 |
| GCLK_SERCOM0_CORE, GCLK_SERCOM1_CORE | 4 |
| GCLK_SERCOM2_CORE | 4 |
| GCLK_SERCOM2_CORE | 5 |
| GCLK_TC0 | 6 |
| GCLK_TC1 | 7 |
| GCLK_TC2, GCLK_TC3 | 8 |
| GCLK_TC4, GCLK_TC5 | 9 |
| GCLK_TC6, GCLK_TC7 | 10 |
| GCLK_TC8, GCLK_TC9 | 11 |
| GCLK_EVSYS_CH_0 | 12 |
| GCLK_EVSYS_CH_1 | 13 |
| GCLK_EVSYS_CH_2 | 14 |
| GCLK_EVSYS_CH_3 | 15 |
| GCLK_EVSYS_CH_4 | 16 |
| GCLK_EVSYS_CH_5 | 17 |
| GCLK_EVSYS_CH_6 | 18 |
| GCLK_EVSYS_CH_7 | 19 |
| GCLK_EVSYS_CH_8 | 20 |
| GCLK_EVSYS_CH_9 | 21 |
| GCLK_EVSYS_CH_10 | 22 |
| GCLK_EVSYS_CH_11 | 23 |
| GCLK_TCC0 | 24 |
| GCLK_TCC1, GCLK_TCC2 | 25 |
| GCLK_AC | 26 |
| GCLK_CM4_TRACE | 27 |
| GCLK_CAN0 | 28 |
| GCLK_CAN1 | 29 |
| GCLK_ETH_TSU | 30 |
| GCLK_CVD | 31 |

The following figure illustrates an example, where SPLL_CLK1 clocks the SERCOM0. The SPLL_CLK1 is input to the REFO generator. The Generic Clock Generator uses the REFO_CLK1 as its clock source and feeds into Peripheral Channel 3. The Generic Clock channel 3, also called GCLK_SERCOM0_CORE, is connected to SERCOM0. The SERCOM0 interface is clocked by PB1_CLK bus clock.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 17-7.** Example of SERCOM0 Clock




**Related Links**

System Configuration and Register Locking (CFG)

### 17.3.10. LPCLK Divider

The low-power clock divider module provides the clock source for low-power domain (VDDBUKUPCORE) modules. There are two sources of sleep clock sources, such as 32 KHz and 32.768 KHz clock. These clock sources are either from LPRC, SOSC or derived from POSC/FRC modules, such as RTCC, which requires 32.768 KHz in the RTC mode; whereas, the Bluetooth link controller requires a 32 KHz clock to maintain the Bluetooth clock in the Standby Sleep mode. Therefore, if the LPCLK source is 32 KHz, the RTC divider must be 31.25, which the user can select using CFGCON4.VBKP_DIVSEL. If the LPCLK source is 32.768 KHz, program CFGCON4.MLPCLK_MOD to divide it by 1.024. See *Power Management Unit (PMU)* for low power mode from Related Links.

**Related Links**

Power Management Unit (PMU)

### 17.3.11. Start-Up Considerations

The presence of hardware NVR configuration fuses on the PIC32CX-BZ6 device allows the system configuration fuses to be ready upon exiting Reset. The following start-up conditions exist:

- On any device Reset, no start-up time is required to transfer configuration values from the NVR memory into the configuration holding registers.
- When

  the device is active, the user can change the primary system clock source from FRC to SPLL by using the OSCCON register.

### 17.3.12. Fail-Safe Clock Monitor (FSCM)

The clock system includes a Fail-Safe Clock Monitor (FSCM). The FSCM monitors the SYS_CLK for continuous operation. If the SYS_CLK fails, it switches the SYS_CLK over to the FRC oscillator and triggers an NMI. The FRC is an untuned 8 MHz oscillator that drives the SYS_CLK during an FSCM event. If executing the NMI, software can restart the main oscillator or shut down the system.

The SYS_CLK and the FSCM halt in the Sleep modes to prevent FSCM detection.

## 17.4. Resets

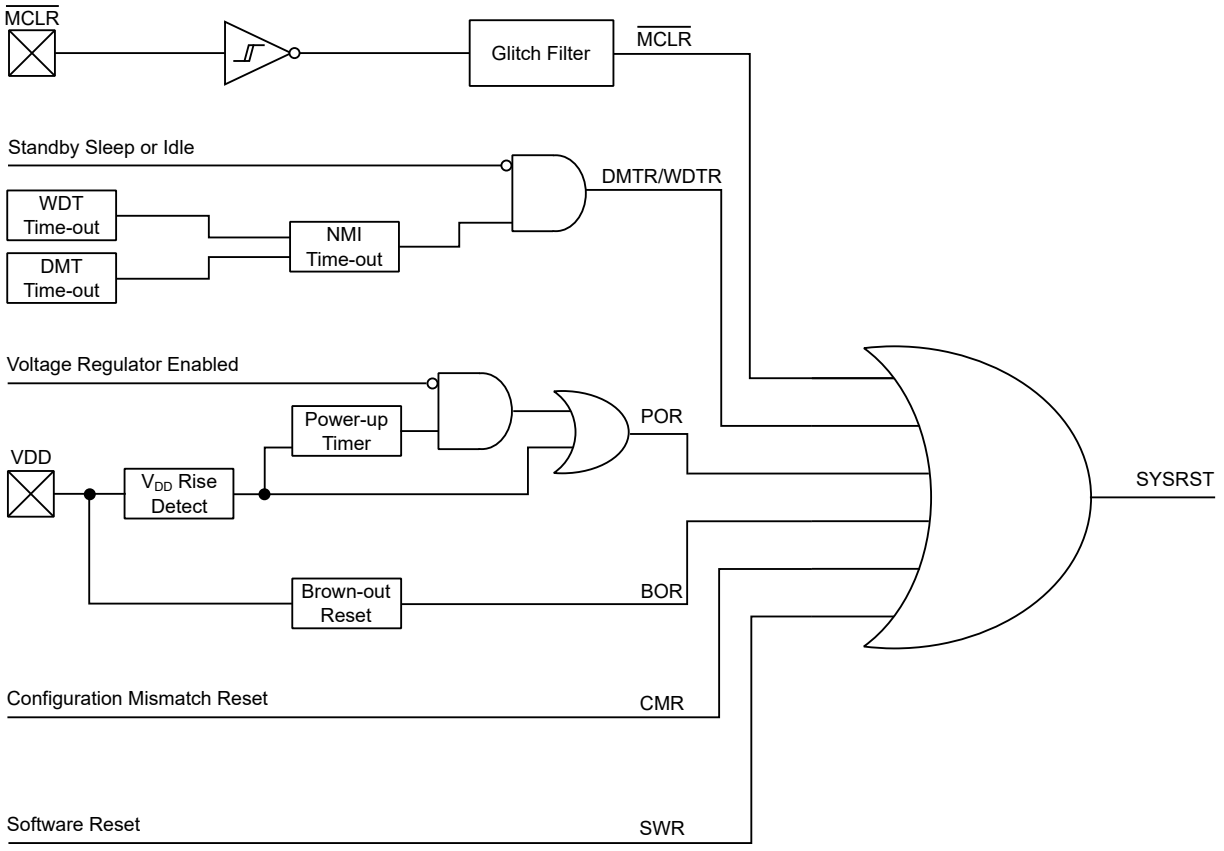
The Reset module combines all Reset sources and controls the device Master Reset signal, SYSRST. The device Reset sources are as follows:


© 2025 Microchip Technology Inc. and its subsidiaries

- Power-on Reset ($V_{dd}$, I/O, or POR)
- Brown-out Reset (BOR/ZPBOR)
- Master Clear Reset ($\overline{MCLR}$)
- Watchdog Timer Reset (NMI Counter)
- Dead Man Timer Reset (NMI Counter)
- Software Reset (SWR)
- Configuration Mismatch Reset (CMR)

A simplified block diagram of the Reset module is shown in the following figure. Any active source of Reset will make the system Reset (SYSRST) signal active. Many registers associated with the CPU and peripherals are forced to a known Reset state.

**Figure 17-8.** CRU-System Reset Block Diagram




### 17.4.1. Control Registers

Most types of device resets set its corresponding status bits in the RCON register to indicate the type of Reset. The one exception is for the Non-Maskable Interrupt (NMI) time-out Reset. A POR clears all RCON bits, except the BOR and POR bits (RCON[1:0]), which are set. The user software can set or clear any of the bits at any time during code execution. The RCON bits serve only as status bits. Setting a Reset status bit in software does not allow system Reset.

The RCON register has other bits associated with the Watchdog Timer (WDT) and device power-saving states. For more information on the function of these bits, see *Using the RCON Status Bits* from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

The RSWRST control register has only one bit, SWRST. This bit is used to force a software Reset condition.

The system clock begins after a delay equal to the duration of the value of RNMICON.NMICNT as it is decremented to zero. During this interval, the program can clear the WDT or DMT flag bits, if desired, to avoid a Reset. If the Active flag is not cleared, the device resets at the end of the interval. The RNMICON.NMICNT value can be set to zero for no delay and up to 255 SYS_CLK cycles.

The user can also trigger the NMI interrupt by setting the RNMICON.SWNMI bit in software or if the RNMICON.CF bit is set by the FSCM. But these do not begin the countdown and do not automatically lead to a Reset.

The Resets module consists of the following Special Function Registers (SFRs):

- RCON - Reset Control Register
- RSWRST - Software Reset Register
- RNMICON - Non-maskable Interrupt (NMI) Control Register

**Related Links**
Using the RCON Status Bits

## 17.4.2. Modes of Operation

### 17.4.2.1. System Reset (SYSRST)

The internal System Reset (SYSRST) can be generated from multiple Reset sources, such as:

- Power-on Reset (POR)
- Brown-out Reset (BOR/ZPBOR)
- Master Clear Reset ($\overline{\text{MCLR}}$)(EXTR)
- Watchdog Time-out Reset (WDTO)
- Deadman Timer Reset (DMTR)
- Software Reset (SWR)
- Configuration Mismatch Reset (CMR)

#### 17.4.2.1.1. Power-on Reset (POR)

A power-on event generates an internal POR pulse when a VDD rise is detected above VPOR. The device supply voltage characteristics must meet the specified starting voltage and rise rate requirements to generate the POR pulse. In particular, VDD must fall below VPOR before initiating a new POR. For more information on the VPOR and VDD rise-rate specifications, see *Electrical Characteristics* from Related Links.

This device has an on-chip internal voltage regulator and its power-on delay is designated as TPU. For more information on the TPU specification, see *Electrical Characteristics* from Related Links.

When the POR event expires, but the device Reset is still asserted while the device configuration settings load and the clock oscillator sources configure, the clock monitoring circuitry waits for the oscillator source to become stable. The clock source of this device when exiting from Reset is always OSCCON.NOSC.

After these delays expire, the System Reset, SYSRST, is de-asserted. Before allowing the CPU to start code execution, eight system clock cycles (SYS_CLK) are required before deasserting the synchronized Reset to the CPU core. When the device is active, the user can change the primary system clock source from FRC to SPLL by using the OSCCON register

The power-on event sets the BOR and POR status bits (RCON[1:0]).

For more information on the values of the delay parameters, see *Electrical Characteristics* from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:** When the device exits the Reset condition (begins normal operation), the device operating parameters (voltage, frequency, temperature and so on) must be within their operating ranges, otherwise, the device will not function correctly.

**Related Links**

Electrical Characteristics

#### 17.4.2.1.2. Master Clear Reset (EXTR)

Whenever the master clear pin ($\overline{MCLR}$) is driven low, the Reset event is synchronized with the system clock, SYS_CLK, before asserting the SYSRST, provided the input pulse on $\overline{MCLR}$ is longer than a certain minimum width, as specified in the Electrical Specifications.

The $\overline{MCLR}$ pin provides a filter to minimize the effects of noise and to avoid unwanted Reset events. The RCON.EXTR status bit is set to indicate the $\overline{MCLR}$ Reset.

EXTR is not a true POR. The user can configure the $\overline{MCLR}$ pin to generate a POR event by configuring the CFGCON1.SMCLR bit rather than an EXTR event.

#### 17.4.2.1.3. Software Reset (SWR)

This device does not provide a specific `RESET` instruction; however, a device Reset can be performed in software (software Reset (SWR)) by executing an SWR command sequence. The SWR acts like an $\overline{MCLR}$ Reset. The SWR sequence requires the system unlock sequence to be executed before writing the RSWRST.SWRST bit.

An SWR is performed as follows:

1. Write the system unlock sequence.
2. Set the SWRST bit (RSWRST[0]) = `1`.
3. Read the RSWRST register.

Setting the SWRST bit (RSWRST[0]) will arm the SWR. The subsequent read of the RSWRST register triggers the SWR, which will occur on the next clock cycle following the read operation. To ensure no other user code is executed before the Reset event occurs, it is recommended that four `NOP` instructions or a `while(1)` statement is placed after the `READ` instruction.

The SWR Status bit (RCON[6]) is set to indicate the SWR.

#### 17.4.2.1.4. Watchdog Timer Reset (WDTR)

A WDTR event is synchronized with the system clock (SYS_CLK), before asserting the SYSRST.

The RNMICON.WDTR flag is going to be set if there is a WDT event when in the CPU Run mode. The device Reset happens after the NMI counter expires and RCON.WDTO flag is set.

A WDT timeout during the Standby Sleep or Idle mode is going to wake up the processor. The WDTR flag will be set if there is a WDT event. The RNMICON.WDTS flag will be set if there is a WDT event during the Standby Sleep/Idle mode (WDTS). The WDTS flag triggers the NMI interrupt, but does not start the NMI counter, nor cause a device Reset. There is no affect on the RCON.WDTO bit. See *Power Management Unit (PMU)* for power modes from Related Links.

**Related Links**

Power Management Unit (PMU)

#### 17.4.2.1.5. Brown-out Reset (BOR)

This device has a simple Brown-out Reset (BOR) capability. If the voltage supplied to the regulator is inadequate to maintain a regulated level, the regulator Reset circuitry generates a BOR event, which is synchronized with the system clock, SYS_CLK, before asserting the SYSRST. The BOR flag bit (RCON[1]) captures this event. For more information, see *Electrical Characteristics* from Related Links.

**Related Links**

Electrical Characteristics


© 2025 Microchip Technology Inc. and its subsidiaries

#### 17.4.2.1.6. Configuration Mismatch Reset (CMR)

To maintain the integrity of the stored configuration values, all device Configuration bits are loaded and implemented as a complementary set of bits. As the Configuration Words are being loaded, for each bit loaded as '1', a complementary value of '0' is stored into its corresponding background Word location and vice versa. The bit pairs are compared every time the Configuration Words are loaded, including in Standby Sleep mode. During this comparison, if the Configuration bit values are not opposite to each other, a configuration mismatch event is generated, which causes a device Reset.

If a device Reset occurs as a result of a configuration mismatch, the CMR Status bit (RCON[9]) is set.

#### 17.4.2.1.7. Deadman Timer Reset (DMTR)

A Deadman Timer Reset (DMTR) is generated when the DMT count has expired.

The primary function of the DMTR is to reset the processor in the event of a software malfunction. The DMT is a free-running instruction fetch timer, which is clocked whenever an instruction fetch occurs until a count match occurs. Instructions are not fetched when the processor is in Standby Sleep mode.

The DMT consists of a 32-bit counter with a time-out count match value as specified by the CFGCON2.DMTCNT bits in the Configuration register.

In general, a DMTR is useful in mission-critical and safety-critical applications, where it must detect any single failure of the software functionality and sequencing. For more information on the DMTR, see *Deadman Timer (DMT)* from Related Links.

**Related Links**

Deadman Timer (DMT)

### 17.4.2.2. Non-Maskable Interrupt (NMI)

The NMI timer provides a delay between DMT or WDT events and a device Reset. The delay set in the System Clock counts from 0-255 in the NMICNT[15:0] bits (RNMICON[15:0]). If these bits are set to '0', there is no delay between the RNMICON.DMTO or RNMICON.WDTR flag and a device Reset. If set to a non-zero value, the NMI interrupt has that number of system clocks to clear flags or save data for debugging purposes.

If the corresponding NMI flag in RNMICON is not cleared before the counter reaches zero, then a device Reset will be issued. If the corresponding NMI flag in RNMICON is cleared before the counter reaches zero, then the counter is stopped, then reloaded with the NMICNT value again, then it waits for another NMI event to occur. In this case, a device Reset is not asserted and the software can return from this interrupt.

The RNMICON.DMTO flag will be set if there is a DMT event. The device will be reset after the NMI counter expires.

The RNMICON.WDTR flag will be set if there is a WDT event. The device will be reset after the NMI counter expires.

The RNMICON.WDTS flag will be set if there is a WDT event during Standby Sleep mode. The RNMICON.WDTS flag triggers the NMI interrupt but does not start the NMI counter nor cause a Reset.

The Fail-Safe Clock Monitor (FSCM) sets the RNMICON.CF bit in case of a clock failure. The CF flag triggers the NMI interrupt but does not start the timer nor cause a Reset.

The RNMICON.SWNMI bit can be set in software to cause an NMI interrupt but does not start the NMI counter nor cause a Reset.


© 2025 Microchip Technology Inc. and its subsidiaries

#### 17.4.2.3. Determining the Source of Device Reset

After a device Reset, examine the RCON register to confirm the source of the Reset. Clear all Reset Status bits in the RCON register after reading them to ensure the RCON value provides meaningful results after the next device Reset.

### 17.4.3. Effects of Various Resets

The Reset value of the Reset Control register, RCON, depends on the type of device Reset, as indicated in the following table.

**Table 17-5.** Status Bits, Their Significance and Initialization Condition for RCON Register(1)

| Condition | EXTR | SWR | WDTO | DMTO | STANDBY SLEEP(2) | IDLE (2) | CMR | BOR | POR |
|---|---|---|---|---|---|---|---|---|---|
| Power-on Reset or $\overline{MCLR}$ set as POR | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Brown-out Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | * |
| $\overline{MCLR}$ Reset during the Run mode | 1 | * | * | * | * | * | * | * | * |
| $\overline{MCLR}$ Reset during the Idle mode | 1 | * | * | * | * | 1 | * | * | * |
| $\overline{MCLR}$ Reset during the Standby Sleep mode | 1 | * | * | * | 1 | * | * | * | * |
| Software Reset command | * | 1 | * | * | * | * | * | * | * |
| Configuration Word mismatch Reset | * | * | * | * | * | * | 1 | * | * |
| WDT Time-out Reset during the Run mode and NMI counter expires | * | * | 1 | * | * | * | * | * | * |
| WDT Time-out Reset during the Idle mode | * | * | * | * | * | 1 | * | * | * |
| WDT Time-out Reset during the Standby Sleep mode | * | * | * | * | 1 | * | * | * | * |
| DMT Time-out Reset and NMI counter expires | * | * | * | 1 | * | * | * | * | * |
| Interrupt exit from the Idle mode | * | * | * | * | * | 1 | * | * | * |
| Interrupt exit from the Standby Sleep mode | * | * | * | * | 1 | * | * | * | * |

1. Legends: * = unchanged
2. The device enters the Standby Sleep or Idle states when it executes a WAIT (WFI) instruction along with the sleep sequence. See *Power Management Unit (PMU)* for Sleep/Idle modes from Related Links.

**Related Links**

Power Management Unit (PMU)

#### 17.4.3.1. Special Function Register (SFR) Reset States

Most of the SFRs associated with the CPU and peripherals are reset to a particular value at a device Reset. This also applies to a WDT/DMT NMI condition, which is treated as a full device Reset by the CPU and peripherals.

The Reset value for the Reset Control register, RCON, is depending on the type of device Reset, see *Status Bits, Their Significance and Initialization Condition for RCON Register* table in *Effects of Various Resets* from Related Links.

**Related Links**

Effects of Various Resets


© 2025 Microchip Technology Inc. and its subsidiaries

#### 17.4.3.2. Configuration Word Register Reset States

All Reset conditions force the configuration settings to be reloaded. The POR and BOR reset all the Configuration Word registers before loading the configuration settings. For all other Reset conditions, the Configuration Word registers are not Reset prior to being reloaded.

#### 17.4.3.3. Using the RCON Status Bits

The user software can read the RCON register after any system Reset to determine the cause of the Reset. The following table provides a summary of the Reset flag bit operation.

**Note:** Clear the status bits in the RCON register after reading them so that the next RCON register value after a device Reset is meaningful.

**Table 17-6.** Reset Flag Bit Operation

| Flag Bit | Set By | Cleared By |
|---|---|---|
| POR (RCON[0]) | POR | User Software |
| BOR (RCON[1]) | POR, BOR | User Software |
| IDLE (RCON[2]) | WAIT instruction | User Software, POR, BOR |
| STANDBY SLEEP (RCON[3]) | WAIT instruction | User Software, POR, BOR |
| WDTO (RCON[4]) | WDT timeout and NMI counter expires | User Software, POR, BOR |
| DMTO (RCON[5]) | DMT timeout and NMI counter expires | User Software, POR, BOR |
| SWR (RCON[6]) | Software Reset Command | User Software, POR, BOR |
| EXTR (RCON[7]) | $\overline{\text{MCLR}}$ Reset | User Software, POR, BOR |
| CMR (RCON[9]) | Configuration mismatch Reset | User Software, POR, BOR |
| BCFGFAIL (RCON[26]) | Non-recoverable error in primary and alternate Configuration Words | User Software, POR, BOR |
| BCFGERR (RCON[27]) | Recoverable error in primary Configuration Words | User Software, POR, BOR |

### 17.4.4. CRU Clock Configuration Registers

The register summary table shows the mapping of the registers in memory, as well as the details of the bit fields in each register. Each register has an associated SET/CLR/INV function register with the suffix appended to the register name, for example: <reg>SET, <reg>CLR, <reg>INV.


© 2025 Microchip Technology Inc. and its subsidiaries

## 17.5. Register Summary

See *CRU* module in the *Product Memory Mapping Overview* from Related Links for base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at their virtual addresses plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | OSCCON | 7:0 | CLKLOCK | | | SLPEN | CF | | SOSCEN | OSWEN |
| | | 15:8 | COSC[3:0] | | | | NOSC[3:0] | | | |
| | | 23:16 | DRMEN | | 2SPDSLP | | | | | |
| | | 31:24 | | | | | | FRCDIV[2:0] | | |
| 0x04 | OSCTRIM | 7:0 | | | TUN[5:0] | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | SPLLCON | 7:0 | | | | SPLLFLOCK | SPLLPWDN | | | |
| | | 15:8 | SPLL1POSTDIV1[7:0] | | | | | | | |
| | | 23:16 | | | | | SPLL2POSTDIV2[3:0] | | | |
| | | 31:24 | SPLL_BYP[1:0] | | | | | | | |
| 0x0C | UPLLCON | 7:0 | UPLLPOSTDIV1[3:0] | | | | UPLLPWDN | UPLLBSWSEL[2:0] | | |
| | | 15:8 | UPLLFBDIV[3:0] | | | | UPLLRST | UPLLFLOCK | UPLLPOSTDIV1[5:4] | |
| | | 23:16 | UPLLREFDIV[1:0] | | UPLLFBDIV[9:4] | | | | | |
| | | 31:24 | UPLL_BYP | | | | UPLLREFDIV[5:2] | | | |
| 0x10 | EPLLCON | 7:0 | EPLLPOSTDIV1[3:0] | | | | EPLLPWDN | EPLLBSWSEL[2:0] | | |
| | | 15:8 | EPLLFBDIV[3:0] | | | | EPLLRST | EPLLFLOCK | EPLLPOSTDIV1[5:4] | |
| | | 23:16 | EPLLREFDIV[1:0] | | EPLLFBDIV[9:4] | | | | | |
| | | 31:24 | EPLL_BYP | | | ECLKOUTEN | EPLLREFDIV[5:2] | | | |
| 0x14 | APLLCON | 7:0 | | | EPLLPOSTDIV2[5:0] | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | RCON | 7:0 | EXTR | SWR | DMTO | WDTO | SLEEP | IDLE | BOR | POR |
| | | 15:8 | | | | | | DPSLP | CMR | |
| | | 23:16 | | | | | | | VBPOR | VBAT |
| | | 31:24 | POR_IO | POR_CORE | | | BCFGERR | BCFGFAIL | NVMLTA | NVMEOL |
| 0x1C | RSWRST | 7:0 | | | | | | | | SWRST |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 | RNMICON | 7:0 | NMICNT[7:0] | | | | | | | |
| | | 15:8 | NMICNT[15:8] | | | | | | | |
| | | 23:16 | SWNMI | | | | EXT | PLVD | CF | WDTS |
| | | 31:24 | | | | | | | DMTO | WDTR |
| 0x24 ... 0x27 | Reserved | | | | | | | | | |
| 0x28 | REFO1CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 0x2C | REFO1TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x30 | REFO2CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x34 | REFO2TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x38 | REFO3CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 0x3C | REFO3TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x40 | REFO4CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 0x44 | REFO4TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x48 | REFO5CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 0x4C | REFO5TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x50 | REFO6CON | 7:0 | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| | | 15:8 | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| | | 23:16 | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| | | 31:24 | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 0x54 | REFO6TRIM | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | ROTRIM8 | | | | | | | |
| | | 31:24 | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| 0x58 | PB1DIV | 7:0 | | | | | PB1DIV[6:0] | | | |
| | | 15:8 | PB1DIVON | | | | PB1DIVRDY | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x5C | PB2DIV | 7:0 | | | | | PB2DIV[6:0] | | | |
| | | 15:8 | PB2DIVON | | | | PB2DIVRDY | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x60 | PB3DIV | 7:0 | | | | | PB3DIV[6:0] | | | |
| | | 15:8 | PB3DIVON | | | | PB3DIVRDY | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x64 | PB4DIV | 7:0 | | | | | PB4DIV[6:0] | | | |
| | | 15:8 | PB4DIVON | | | | PB4DIVRDY | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x68 | PB5DIV | 7:0 | | | | | PB5DIV[6:0] | | | |
| | | 15:8 | PB5DIVON | | | | PB5DIVRDY | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x6C | SLEWCON | 7:0 | | | | | | SLW_UP | SLW_DN | SLW_BUSY |
| | | 15:8 | | | | | | SLW_DIV[2:0] | | |
| | | 23:16 | | | | | SYS_DIV[3:0] | | | |
| | | 31:24 | | | | | SLW_DELAY[3:0] | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x70 | CLKSTAT | 7:0 | SPLLRDY | UPLLRDY | EPLL2RDY | LPRCRDY | SOSCRDY | POSCRDY | SPLL1RDY | FRCRDY |
| | | 15:8 | | | | | | PB1RDY | SYSRDY | EPLL1RDY |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x74 ... 0x77 | Reserved | | | | | | | | | |
| 0x78 | CLK_DIAG | 7:0 | | UPLL_STOP | EPLL_STOP | SPLL1_STOP | LPRC_STOP | FRC_STOP | SOSC_STOP | POSC_STOP |
| | | 15:8 | | | | | | | | |
| | | 23:16 | NMICTR7 | NMICTR6 | NMICTR5 | NMICTR4 | NMICTR3 | NMICTR2 | NMICTR1 | NMICTR0 |
| | | 31:24 | NMICTR15 | NMICTR14 | NMICTR13 | NMICTR12 | NMICTR11 | NMICTR10 | NMICTR9 | NMICTR8 |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 17.6. 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.

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.

Some registers are enable-protected, meaning the user can only write them by disabling the peripheral. Enable protection is denoted by the Enable-Protected property in each individual register description.

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.1. CRU Oscillator Control

**Name:** OSCCON
**Offset:** 0x00
**Reset:** 0x00200003

**Note:** Perform the system unlock sequence before writing this register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | FRCDIV[2:0] | | |
| Access | | | | | | R/W/L | R/W/L | R/W/L |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DRMEN | | 2SPDSLP | | | | | |
| Access | R/W/L | | R/W/L | | | | | |
| Reset | 0 | | 1 | | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COSC[3:0] | | | | NOSC[3:0] | | | |
| Access | R | R | R | R | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CLKLOCK | | | SLPEN | CF | | SOSCEN | OSWEN |
| Access | R/W/L | | | R/W/L | R/W/HS/L | | R/W/L | R/W/HC/L |
| Reset | 0 | | | 0 | 0 | | 1 | 1 |

**Bits 26:24 – FRCDIV[2:0]** Fast RC Clock Divider bits

| Value | Description |
|---|---|
| 000 | FRC is divided by 1 (default value) |
| 001 | FRC is divided by 2 |
| 010 | FRC is divided by 4 |
| 011 | FRC is divided by 8 |
| 100 | FRC is divided by 16 |
| 101 | FRC is divided by 32 |
| 110 | FRC is divided by 64 |
| 111 | FRC is divided by 256 |

**Bit 23 – DRMEN** Enable Dream Mode bit

| Value | Description |
|---|---|
| 1 | When the SLEEP/WAIT (WFI) instruction is executed and SLPEN = 1, DMA transfer complete causes the device to enter the Sleep mode. |
| 0 | DMA transfer has no effect |

**Bit 21 – 2SPDSLP** 2-Speed start-up enabled in the Standby Sleep mode bit
**Note:** CFGCON2.WAKE2SPD specifies the default Reset value.

| Value | Description |
|---|---|
| 1 | When the device exits the Standby Sleep mode, the SYS_CLK is going to be from FRC until the selected clock is ready. |
| 0 | When the device exits the Standby Sleep mode, the SYS_CLK is going to be from the selected clock. |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 15:12 – COSC[3:0]** Current Oscillator Selection bits (Read-only)

**Notes:**

- Default value on reset is 4'b0000
- Loaded with NOSC[3:0] at the completion of a successful clock switch
- Set to FRC value (0000) when FSCM detects a failure and switches clock to FRC

| Value | Description |
|---|---|
| 0000 | Fast RC oscillator (FRC) divided by OSCCON.FRCDIV |
| 0001 | System PLL Clock-1 (SPLL_CLK1 module) (input clock is 128 MHz from RFPLL wrapper and divider set by SPLLCON) |
| 0010 | Primary Oscillator (POSC) |
| 0011 | Secondary Oscillator (SOSC) |
| 0100 | Low Power RC Oscillator (LPRC) |
| 0101 | Ethernet PLL Clock-2 (EPLL Module) (input clock and divider set by APLLCON) |
| 0110 | USB PLL Clock (UPLL Module) (input clock and divider set by UPLLCON) |
| 0111-1111 | Reserved for future use |

**Bits 11:8 – NOSC[3:0]** New Oscillator Selection bits

**Note:** Default value on reset is 4'b0000.

| Value | Description |
|---|---|
| 0000 | Fast RC oscillator (FRC) divided by OSCCON.FRCDIV |
| 0001 | System PLL Clock-1 (SPLL_CLK1 module) (input clock is 128 MHz from RFPLL wrapper and divider set by SPLLCON) |
| 0010 | Primary Oscillator (POSC) |
| 0011 | Secondary Oscillator (SOSC) |
| 0100 | Low-Power RC Oscillator (LPRC) |
| 0101 | Ethernet PLL Clock-2 (EPLL Module) (input clock and divider set by APLLCON) |
| 0110 | USB PLL Clock (UPLL Module) (input clock and divider set by UPLLCON) |
| 0111-1111 | Reserved for future use |

**Bit 7 – CLKLOCK** Clock Lock Enabled bit

**Notes:**

- When set, this bit can only be cleared via a device Reset.
- When active, this bit prevents writes to the following registers: NOSC[3:0] and OSWEN.

| Value | Description |
|---|---|
| 1 | All clock and PLL configuration registers are locked.<br>These include OSCCON, OSCTRIM, SPLLCON, PBxDIV |
| 0 | Clock and PLL selection registers are not locked; configurations can be modified. |

**Bit 4 – SLPEN** Enable Sleep Mode bit

| Value | Description |
|---|---|
| 1 | When a WAIT Instruction is executed, the device enters the Standby Sleep mode. |
| 0 | When a WAIT instruction is executed, the device enters the IDLE mode. |

**Bit 3 – CF** Clock Fail Detect bit (Read/Writable/Clearable by application)

**Notes:**

- Writing '1' to this bit initiates the clock-switching sequence by the clock switch state machine.
- A Reset occurs when the clock switch state machine initiates a valid clock switching sequence.
- This bit is set when the clock fail event is detected.

| Value | Description |
|---|---|
| 1 | FSCM detected a clock failure |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | FSCM did not detected a clock failure |

**Bit 1 – SOSCEN** Low Power Secondary Oscillator Enable bit

**Note:** Set the value specified by the SOSCEN configuration bits in CFGCON2.SOSCSEL on RESET.

| Value | Description |
|---|---|
| 1 | Enable Secondary Oscillator |
| 0 | Disable Secondary Oscillator |

**Bit 0 – OSWEN** Oscillator Switch Enable bit

**Notes:**

- Writing '0' to this bit has no effect.
- Hardware clears this bit after a successful clock switch.
- Hardware clears this bit after a redundant clock switch (NOSC = COSC).
- Hardware clears this bit after FSCM switches the oscillator to the fail-safe clock source.

| Value | Description |
|---|---|
| 1 | Requests the oscillator switch to select as specified by NOSC[3:0] bits |
| 0 | Oscillator switch is complete |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.2. CRU Oscillator Trimming

**Name:** OSCTRIM
**Offset:** 0x4
**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 |
|---|---|---|---|---|---|---|---|---|
| | | | TUN[5:0] | | | | | |
| Access | | | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 5:0 – TUN[5:0]** Internal Fast RC (FRC) Oscillator Tuning bits
This bit field specifies the user tuning capability for the internal fast RC oscillator.
**Note:** The system unlock sequence must be done before this register can be written.

| Value | Description |
|---|---|
| 011111 | Maximum Frequency |
| 011110 | — |
| ... | — |
| 000001 | — |
| 000000 | Center Frequency, oscillator is running at calibrated frequency |
| 111111 | — |
| 111110 | — |
| ... | — |
| 100001 | — |
| 100000 | Minimum Frequency |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.3. SPLL Control

**Name:** SPLLCON
**Offset:** 0x8
**Reset:** 0xC0010028

**Note:** Perform the system unlock sequence before this register is written.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SPLL_BYP[1:0] | | | | | | | |
| Access | R/W/L | R/W/L | | | | | | |
| Reset | 1 | 1 | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SPLL2POSTDIV2[3:0] | | | |
| Access | | | | | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | | | | 0 | 0 | 0 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SPLL1POSTDIV1[7:0] | | | | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | SPLLFLOCK | SPLLPWDN | | | |
| Access | | | | R/W/L | R/W/L | | | |
| Reset | | | | 0 | 1 | | | |

**Bits 31:30 – SPLL_BYP[1:0]** SPLL Bypass; SPLL_CLK2 clock source selection

**Notes:**

- Dictates the clock source for ADC CP (Analog-to-Digital Converter Charge Pump) (SPLL_CLK2) Clock generation only
- Preselect the clock sources and keep them ready before the need of ADC CP arrives. Failure to do so results in loss of clock for one or two cycles when ADC CP is enabled.

| Value | Description |
|---|---|
| 00 | RFPLL Clock is the clock source for ADC CP Clock Generation. |
| x1 | FRC is used as clock source for ADC CP Clock Generation. |
| 10 | POSC is used as clock source for ADC CP Clock Generation. |

**Bits 19:16 – SPLL2POSTDIV2[3:0]** ADC-CP (SPLL2) Post Divide Value

| Value | Description |
|---|---|
| 1 ≤ SPLLPOSTDIV2 ≤ 15 | Divide by SPLL2POSTDIV2 |
| 0 | No Clock; Clock disabled |

**Bits 15:8 – SPLL1POSTDIV1[7:0]** SPLL1 Post Divide Value

| Value | Description |
|---|---|
| 2 ≤ SPLLPOSTDIV ≤ 255 | Divide by SPLL1POSTDIV1 |
| 0,1 | Divide by 1 |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 4 – SPLLFLOCK** System PLL Force Lock

| Value | Description |
|---|---|
| 1 | Force the SPLL lock signal to be asserted |
| 0 | Do not force the SPLL lock signal to be asserted |

**Bit 3 – SPLLPWDN** PLL Power Down Register bit

| Value | Description |
|---|---|
| 1 | PLL is powered down |
| 0 | PLL is active |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.4. UPLL Control

**Name:** UPLLCON
**Offset:** 0xC
**Reset:** 0x00

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | UPLL_BYP | | | | UPLLREFDIV[5:2] | | | |
| Access | R/W/L | | | | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 1 | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | UPLLREFDIV[1:0] | | UPLLFBDIV[9:4] | | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | UPLLFBDIV[3:0] | | | | UPLLRST | UPLLFLOCK | UPLLPOSTDIV1[5:4] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | UPLLPOSTDIV1[3:0] | | | | UPLLPWDN | UPLLBSWSEL[2:0] | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

**Bit 31 – UPLL_BYP** UPLL Bypass
When this bit is set, the input clock REF bypasses PLL to PLLOUTx.
**Note:** It is recommended to setup SPLL first before setting up other PLLs, especially when using SYSPLL for generating main system clock.

**Bits 27:22 – UPLLREFDIV[5:0]** Reference Frequency Divide

| Value | Description |
|---|---|
| 1 ≤ UPLLDIVR ≤ 63 | Divide by UPLLDIVR |
| 0 | Not used |

**Bits 21:12 – UPLLFBDIV[9:0]** PLL Feedback Divider

| Value | Description |
|---|---|
| 16 ≤ UPLLFBDIV ≤ 1023 | Divide by UPLLDIVR |
| 0 to 15 | Not used |

**Bit 11 – UPLLRST** USB PLL Reset

| Value | Description |
|---|---|
| 1 | Assert the reset to the UPLL |
| 0 | De-assert the reset to the UPLL |

**Bit 10 – UPLLFLOCK** USB PLL Force Lock


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Force the UPLL lock signal to be asserted |
| 0 | Do not force the UPLL lock signal to be asserted |

**Bits 9:4 – UPLLPOSTDIV1[5:0]** First Post Divide Value

| Value | Description |
|---|---|
| 1 ≤ UPLLPOSTDIV1 ≤ 63, | Divide by UPLLPOSTDIV1 |
| 0 | Not used |

**Bit 3 – UPLLPWDN** PLL Power Down Register bit

| Value | Description |
|---|---|
| 1 | PLL is powered down |
| 0 | PLL is active |

**Bits 2:0 – UPLLBSWSEL[2:0]** PLL Bandwidth Select

Use the frequency range that matches the PLL closed loop bandwidth as based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.5. EPLL Control

**Name:** EPLLCON
**Offset:** 0x10
**Reset:** 0x00

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | EPLL_BYP | | | ECLKOUTEN | EPLLREFDIV[5:2] | | | |
| Access | R/W/L | | | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 1 | | | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EPLLREFDIV[1:0] | | EPLLFBDIV[9:4] | | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EPLLFBDIV[3:0] | | | | EPLLRST | EPLLFLOCK | EPLLPOSTDIV1[5:4] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EPLLPOSTDIV1[3:0] | | | | EPLLPWDN | EPLLBSWSEL[2:0] | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

**Bit 31 – EPLL_BYP** EPLL Bypass
When this bit is set, the input clock REF bypasses PLL to PLLOUTx.
**Note:** It is recommended to setup SPLL first before setting up other PLLs, especially when using SYSPLL for generating main system clock.

**Bit 28 – ECLKOUTEN** Ethernet Clock Out pin Enable Bit

| Value | Description |
|---|---|
| 1 | ETH_CLK_OUT Pin is enabled |
| 0 | ETH_CLK_OUT Pin is disabled |

**Bits 27:22 – EPLLREFDIV[5:0]** Reference Frequency Divide

| Value | Description |
|---|---|
| 1 ≤ EPLLDIVR ≤ 63 | Divide by EPLLDIVR |
| 0 | Not used |

**Bits 21:12 – EPLLFBDIV[9:0]** PLL Feedback Divider

| Value | Description |
|---|---|
| 16 ≤ EPLLFBDIV ≤ 1023 | Divide by EPLLDIVR |
| 0 to 15 | Not used |

**Bit 11 – EPLLRST** EPLL Reset


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Assert the reset to the EPLL |
| 0 | De-assert the reset to the EPLL |

**Bit 10 – EPLLFLOCK** EPLL Force Lock

| Value | Description |
|---|---|
| 1 | Force the EPLL lock signal to be asserted |
| 0 | Do not force the EPLL lock signal to be asserted |

**Bits 9:4 – EPLLPOSTDIV1[5:0]** First Post Divide Value

| Value | Description |
|---|---|
| 1 ≤ EPLLPOSTDIV1 ≤ 63, | Divide by EPLLPOSTDIV1 |
| 0 | Not used |

**Bit 3 – EPLLPWDN** EPLL Power Down Register bit

| Value | Description |
|---|---|
| 1 | EPLL is powered down |
| 0 | EPLL is active |

**Bits 2:0 – EPLLBSWSEL[2:0]** PLL Bandwidth Select

Use the frequency range that matches the PLL closed loop bandwidth as based on the reference frequency divided by REFDIV to be set to allow the PLL loop filter to work with the post-reference divider frequency.


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.6. Auxiliary PLL Control

**Name:** APLLCON
**Offset:** 0x14
**Reset:** 0xC0010028

**Note:** The system unlock sequence must be done before this register can be written.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | EPLLPOSTDIV2[5:0] | | | | | |
| Access | | | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 1 |

**Bits 5:0 – EPLLPOSTDIV2[5:0]** EPLL Post Divider
EPLL Post Divider bits for controlling second PLL clock output.

| Value | Description |
|---|---|
| 1 ≤ xPLLPOSTDIV2 ≤ 63 | Divide by xPLLPOSTDIV2 |
| 0 | Not used |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.7. Reset Control Register

**Name:** RCON
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | POR_IO | POR_CORE | | | BCFGERR | BCFGFAIL | NVMLTA | NVMEOL |
| Access | R/W/HS | R/W/HS | | | R/W/HS | R/W/HS | R/W/HS | R/W/HS |
| Reset | 0 | 0 | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | VBPOR | VBAT |
| Access | | | | | | | R/W/HS | R/W/HS |
| Reset | | | | | | | 0 | 0 |

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

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EXTR | SWR | DMTO | WDTO | SLEEP | IDLE | BOR | POR |
| Access | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS | R/W/HS |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 31 – POR_IO** I/O Voltage POR Flag bit

- 1 = A Power-on Reset has occurred due to I/O voltage.
- 0 = A Power-on Reset has not occurred due to I/O voltage.

This bit is set by hardware at detection of an I/O POR event. User software must clear this bit to view the next detection.
**Note:** Writing '1' to this bit does not cause POR_IO reset.

**Bit 30 – POR_CORE** Core Voltage POR Flag bit

- 1 = A Power-on Reset has occurred due to core voltage.
- 0 = A Power-on Reset has not occurred due to core voltage.

This bit is set by hardware at detection of a core POR event. User software must clear this bit to view the next detection.
**Note:** Writing '1' to this bit does not cause POR_IO reset.

**Bit 27 – BCFGERR** BCFG Error Flag bit

- 0 = A BCFG error has not occurred.
- 1 = A BCFG error has occurred.

This bit is set when a primary BCFG value has an error but the secondary BCFG value is valid and used.

**Bit 26 – BCFGFAIL** BCFG Failure Flag bit

- 0 = A BCFG failure has not occurred.


© 2025 Microchip Technology Inc. and its subsidiaries

- 1 = A BCFG failure has occurred.

This bit is set when both the Primary and Secondary BCFG values has an unrecoverable error. Only the default values are in effect.

**Bit 25 – NVMLTA** NVM Life Time Alert Flag bit

- 0 = A NVM LTA error has not occurred.
- 1 = A NVM LTA error has occurred.

This bit is set due to charge leakage, and the NVM (Flash) is nearing EOL.

**Bit 24 – NVMEOL** NVM End of Life Flag bit

- 0 = A NVM EOL failure has not occurred.
- 1 = A NVM EOL failure has occurred.

This bit may not be visible to the user because the part does not come out of Reset if the bit is asserted.

**Bit 17 – VBPOR** VBPOR Mode Flag bit

- 1= A VBAT Domain POR has occurred.
- 0 = A VBAT Domain POR has not occurred.

**Notes:**

- User may write this bit to '1'. Does not cause a VBPOR event (used for testing only).
- The actual register is not implemented as part of this module; bit location only reflects the state of the lpwr_vbpor_status input.

**Bit 16 – VBAT** VBAT Mode Flag bit

- 1= A POR exit from VBAT has occurred. A true POR must be established with the valid VBAT voltage level on the VBAT pin.
- 0 = A POR exit from VBAT has not occurred.

**Bit 10 – DPSLP** Deep Sleep Mode Flag bit

- 1 = Deep Sleep mode has occurred.
- 0 = Deep Sleep mode has not occurred.

This bit is set by hardware at the time of entry into the Deep Sleep mode. User software must clear this bit to view next detection.

**Bit 9 – CMR** Configuration Mismatch Reset Flag bit

- 1 = A CMR event has occurred.
- 0 = A CMR event has not occurred.

**Note:** Writing '1' to this bit does not cause Mismatch Reset.

**Bit 7 – EXTR** External Reset $\overline{\text{MCLR}}$ Status bit

- 1 = A Master Clear (pin) Reset has occurred.
- 0 = A Master Clear (pin) Reset has not occurred.

**Note:** Writing '1' to this bit does not cause a ($\overline{\text{MCLR}}$).

**Bit 6 – SWR** Software Reset Flag bit

- 1 = A SWR has occurred.


© 2025 Microchip Technology Inc. and its subsidiaries

- 0 = A SWR has not occurred.

**Note:** Writing '1' to this bit does not cause SWR.

**Bit 5 – DMTO** Deadman Timer Time-out Flag bit

- 1 = DMT Time-out has occurred and caused a Reset.
- 0 = DMT Time-out has not occurred.

**Note:** Writing '1' to this bit does not cause DMT Reset.

**Bit 4 – WDTO** Watchdog Timer Time-Out Flag bit

- 1 = WDT Time-out has occurred and caused a Reset.
- 0 = WDT Time-out has not occurred.

**Note:** Writing '1' to this bit does not cause WDTR.

**Bit 3 – SLEEP** Wake from Standby Sleep Flag bit

- 1 = Device has been in Standby Sleep mode.
- 0 = Device has not been in Standby Sleep mode.

**Note:** Writing '1' to this bit does not invoke the Standby Sleep mode.

**Bit 2 – IDLE** Wake from Idle Flag bit

- 1 = Device was in Idle mode.
- 0 = Device was not in Idle mode.

**Note:** Writing '1' to this bit does not invoke the Idle mode.

**Bit 1 – BOR** BOR Flag bit

- 1 = A BOR occurred.
- 0 = A BOR did not occur.

This bit is set by hardware at detection of a BOR event. User software must clear this bit to view the next detection.
**Note:** Writing '1' to this bit does not cause a BOR.

**Bit 0 – POR** POR Flag bit

- 1 = A Power-on Reset occurred.
- 0 = A Power-on Reset did not occur.

This bit is set by hardware at detection of a POR event. User software must clear this bit to view the next detection.
**Note:** Writing '1' to this bit does not cause a POR.


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.8. Software Reset Register

**Name:** RSWRST
**Offset:** 0x1C
**Reset:** 0x00000000
**Property:** -

**Note:** The system unlock sequence must be done before writing this register. For more details, see *Register Locking* from Related Links.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | SWRST |
| Access | | | | | | | | W/HC |
| Reset | | | | | | | | 0 |

**Bit 0 – SWRST** Software Reset Trigger bit
'1' = Enable SWR event. A subsequent read of this register triggers the system Reset sequence. The system unlock sequence must be done before the bit can be written. This bit always reads a value of logic '0'.

**Related Links**
Register Locking


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.9. NMI Control Register

**Name:** RNMICON
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** -

**Note:** The system unlock sequence must be done before writing this register. See *System Configuration and Register Locking (CFG)* from Related Links.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DMTO | WDTR |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SWNMI | | | | EXT | PLVD | CF | WDTS |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | NMICNT[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 |
|---|---|---|---|---|---|---|---|---|
| | NMICNT[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 25 – DMTO** Deadman Timer Time-Out Flag bit (this causes a Reset when NMICNT expires)

- 1 = DMT Time-out occurred and caused an NMI.
- 0 = DMT Time-out did not occur.

**Note:** Writing '1' to this bit cause a user-initiated DMT NMI event and NMICNT start.

**Bit 24 – WDTR** Watchdog Timer Time-Out in Run Flag bit

- 1 = WDT Time-out occurred and caused an NMI (this may cause a Reset if NMICNT expires).
- 0 = WDT Time-out did not occur.

**Note:** Writing '1' to this bit cause a user initiated WDT NMI event and NMICNT start.

**Bit 23 – SWNMI** Software NMI Trigger bit

- 1 = Writing '1' to this bit generates an NMI.
- 0 = Writing '0' to this bit has no effect.

**Bit 19 – EXT** External / Generic NMI Event bit

- 1 = A general NMI event was detected and caused an NMI.
- 0 = A general NMI event was not detected.

**Note:** Writing '1' to this bit cause a user initiated EXT NMI event.


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 18 – PLVD** Programmable Low Voltage Detect Event bit

- 1 = PLVD detected a low voltage condition and caused an NMI.
- 0 = PLVD did not detect a low voltage condition.

**Note:** Writing '1' to this bit causes a user-initiated PLVD NMI event.

**Bit 17 – CF** Clock Fail Detect bit (Read/Clear-able by application)

- 1 = FSCM detected clock failure and caused an NMI.
- 0 = FSCM did not detect clock failure.

**Note:** Writing '1' to this bit causes a user-initiated clock failure NMI event but does not cause a clock switch.

**Bit 16 – WDTS** Watch-Dog Timer Time-out in Standby Sleep Flag bit

- 1 = WDT Time-out occurred during the Standby Sleep mode and caused a wake-up from sleep.
- 0 = WDT Time-out did not occur during the Standby Sleep mode.

**Note:** Writing '1' to this bit causes a user-initiated WDT NMI event.

**Bits 15:0 – NMICNT[15:0]** NMI Reset counter value bit

This bit field specifies the reload value used by the NMI Reset counter. The following events generate an NMI event and/or reset when the NMI_CNT expires on:

- WDT event
- DMT event

| Values | Description |
|---|---|
| 0x0000 | No delay between NMI assertion and device Reset event. |
| 0x0001 | — |
| 0x0002 | — |
| ..... | — |
| ..... | — |
| ..... | — |
| 0xFFFE | — |
| 0xFFFF | Number of SYSCLK cycles that the software has to clear the NMI event before a device Reset is performed. If the NMI event is cleared before the counter reaches zero, no device Reset is asserted. |

**Related Links**

System Configuration and Register Locking (CFG)


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.10. Reference Oscillator x Control

**Name:** REFOxCON
**Offset:** 0x28 + (x-1)*0x08 [x=1..6]
**Reset:** 0x00000000

**Notes:**

- Do not write REFOCON.ROSEL while the REFOCON.ACTIVE bit is '1'. This results in undefined behavior.
- Do not write REFOCON when REFOCON.ON != REFOCON.ACTIVE. This results in undefined behavior.
- This register can always be accessed regardless of the SYSKEY value.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | RODIV14 | RODIV13 | RODIV12 | RODIV11 | RODIV10 | RODIV9 | RODIV8 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | RODIV7 | RODIV6 | RODIV5 | RODIV4 | RODIV3 | RODIV2 | RODIV1 | RODIV0 |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | ON | FRZ | SIDL | OE | RSLP | | DIVSW_EN | ACTIVE |
| Access | R/W | R/W | R/W | R/W | R/W | | HC/ R/W | HS/HC/ R |
| Reset | 0 | 0 | 0 | 0 | 0 | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ROSEL3 | ROSEL2 | ROSEL1 | ROSEL0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30 – RODIV** Reference Clock Divider bits
Specifies 1/2 period of reference clock in the source clocks.
For example, period of REFO_CLK = [Reference source * 2] * RODIV.

| Value | Description |
|---|---|
| 0x7FFF | REFOx clock is Base clock frequency divided by 65,534 (32,767 *2) |
| 0x7FFE | REFOx clock is Base clock frequency divided by 65,532 (32,766 * 2) |
| ... | |
| ... | |
| ... | |
| 0x0003 | REFOx clock is Base clock frequency divided by 6 (3*2) |
| 0x0002 | REFOx clock is Base clock frequency divided by 4 (2*2) |
| 0x0001 | REFOx clock is Base clock frequency divided by 2 (1*2) |
| 0x0000 | REFOx clock is same frequency as Base clock (no divider) |

**Bit 15 – ON** Output Enable bit

| Value | Description |
|---|---|
| 1 | Enables the Reference Oscillator Module |
| 0 | Disables the Reference Oscillator Module |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 14 – FRZ** Freeze in Debug mode bit

| Value | Description |
|---|---|
| 1 | When emulator is in the Debug mode, module freezes operation |
| 0 | When emulator is in the Debug mode, module continues operation |

**Bit 13 – SIDL** Peripheral Stop in Idle Mode bit

| Value | Description |
|---|---|
| 1 | Discontinue module operation when device enters the Idle mode |
| 0 | Continues module operation in the Idle mode |

**Bit 12 – OE** Reference Clock Output Enable bit

| Value | Description |
|---|---|
| 1 | Reference clock is driven out on REFOx pin |
| 0 | Reference clock is not driven out on REFOx pin |

**Bit 11 – RSLP** Reference Oscillator Run in Standby Sleep bit
**Note:** This bit is ignored when ROSEL[3:0] = (0000 or 0001).

| Value | Description |
|---|---|
| 1 | Reference Oscillator output continues to run in Standby Sleep |
| 0 | Reference Oscillator output is disabled in Standby Sleep |

**Bit 9 – DIVSW_EN** Clock RODIV/ROTRIM switch enabled

| Value | Description |
|---|---|
| 1 | Clock Divider Switching is in progress |
| 0 | Clock Divider Switch is completed |

**Bit 8 – ACTIVE** Reference Clock Request Status bit

| Value | Description |
|---|---|
| 1 | Reference clock request is active (User must not update this REFOCON register) |
| 0 | Reference clock request is not active (User can update this REFOCON register) |

**Bits 0, 1, 2, 3 – ROSEL** Reference Clock Source Select bits
Select one of various clock sources to be used as the reference clock.

| Value | Description |
|---|---|
| 1100-1111 | Reserved |
| 1011 | REFI Pin |
| 1010 | System clock, SYS_CLK (reference clock reflects any device clock switching) |
| 1001 | Peripheral clock, PB1_CLK (reference clock reflects any peripheral clock switching) |
| 1000 | Ethernet PLL (Clock-1) |
| 0111 | System PLL (Clock-3), SPLL_CLK3 |
| 0110 | USB PLL (Clock-1) |
| 0101 | Ethernet PLL (Clock-2) |
| 0100 | LPRC |
| 0011 | SOSC |
| 0010 | POSC |
| 0001 | System PLL (Clock-1), SPLL_CLK1 |
| 0000 | FRC |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.11. Reference Oscillator x Trim

**Name:** REFOxTRIM
**Offset:** 0x2C + (x-1)*0x08 [x=1..6]
**Reset:** 0x00000000

**Notes:**

- Do not write REFOxTRIM when REFOxCON.ON != REFOxCON.ACTIVE. This results in undefined behavior.
- This register can always be accessed regardless of the SYSKEY value.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | ROTRIM0 | ROTRIM1 | ROTRIM2 | ROTRIM3 | ROTRIM4 | ROTRIM5 | ROTRIM6 | ROTRIM7 |
| Access | R/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 |
|---|---|---|---|---|---|---|---|---|
| | ROTRIM8 | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

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

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

**Bits 23, 24, 25, 26, 27, 28, 29, 30, 31 – ROTRIM** Trim bits - Provides fractional additive to RODIV value for 1/2 period of REFOx clock

**Note:** ROTRIM values greater than zero are only valid when RODIV values are greater than 0.

| Value | Description |
|---|---|
| 0000_0000_0 | 0/512 (0.0) divisor is added to RODIV value |
| 0000_0000_1 | 1/512 (0.001953125) divisor is added to RODIV value |
| 0000_0001_0 | 2/512 (0.00390625) divisor is added to RODIV value |
| ... | ... |
| ... | ... |
| 100000000 | 256/512 (0.5000) divisor is added to RODIV value |
| ... | ... |
| ... | ... |
| 1111_1111_0 | 510/512 (0.99609375) divisor is added to RODIV value |
| 1111_1111_1 | 511/512 (0.998046875) divisor is added to RODIV value |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.12. PBx Clock Divisor Control

**Name:** PBxDIV
**Offset:** 0x58 + (x-1)*0x04 [x=1..5]
**Reset:** 0x00008800

**Note:** Perform the system unlock sequence before this register can be written. The Reset value for PB3DIV[6:0] is 0x09.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | PBxDIVON | | | | PBxDIVRDY | | | |
| Access | R | | | | R | | | |
| Reset | 1 | | | | 1 | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | PBxDIV[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 – PBxDIVON** (x=1 to 5) Output Enable bit

| Value | Description |
|---|---|
| 1 | Enables PBx Output clock |
| 0 | Disables PBx Output clock<br>**Note:** Do not write PB1DIV.PB1DIVON bit as '0', as the CRU system uses this clock, and it is one of the source for REFO. |

**Bit 11 – PBxDIVRDY** (x=1 to 5) PBx Peripheral Clock Divisor Ready

| Value | Description |
|---|---|
| 1 | Indicates the PB clock divisor logic is not switching divisors and the PBxDIV may be written. |
| 0 | Indicates the PB clock divisor logic is currently switching values and the PBxDIV cannot be written. |

**Bits 6:0 – PBxDIV[6:0]** (x=1 to 5) PBx Peripheral Clock Divisor Control value

| Value | Description |
|---|---|
| 000_0000 | Divide by 1 PBx Clock same frequency as SYS_CLK |
| 000_0001 | Divide by 2 PBx Clock is 1/2 of SYS_CLK |
| 000_0010 | Divide by 3 PBx Clock is 1/3 of SYS_CLK |
| 000_0011 | Divide by 4 PBx Clock is 1/4 of SYS_CLK |
| . . . | ... |
| . . . | ... |
| 000_1111 | Divide by 16 PBx Clock is 1/16 of SYS_CLK |
| 001_0000 | Divide by 17 PBx Clock is 1/17 of SYS_CLK |
| . . . | ... |
| . . . | ... |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 111_1110 | Divide by 127 PBx Clock is 1/127 of SYS_CLK |
| 111_1111 | Divide by 128 PBx Clock is 1/128 of SYS_CLK |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.13. Slew Rate Control for Clock Switching

**Name:** SLEWCON
**Offset:** 0x6C
**Reset:** 0x00000000

**Notes:**
- Perform the system unlock sequence before this register is written.
- Updates to this register do not take affect until OSCCON.OSWEN is set.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SLW_DELAY[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | c | c | c | c |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | SYS_DIV[3:0] | | | |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | c | c | c | c |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SLW_DIV[2:0] | | |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | c | c | c |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | SLW_UP | SLW_DN | SLW_BUSY |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | c | c | c |

**Bits 27:24 – SLW_DELAY[3:0]** Number of clocks generated at each slew step for a clock switch
**Note:** The input, cfg_slewcon_sel[] defines the reset value of this register field.

| Value | Description |
|---|---|
| 0000 | 1 clock is generated at each slew step |
| 0001 | 2 clocks is generated at each slew step |
| . . . | ... |
| 1111 | 16 clocks are generated at each slew step |

**Bits 19:16 – SYS_DIV[3:0]** PBx Peripheral Clock Divisor Control value

| Value | Description |
|---|---|
| 0000 | Divide by 1 - SYS_CLK_OUT same frequency as SYS_CLK source - Default |
| 0001 | Divide by 2 - SYS_CLK_OUT is 1/2 of SYS_CLK source |
| 0010 | Divide by 3 - SYS_CLK_OUT is 1/3 of SYS_CLK source |
| . . . | ... |
| 1111 | Divide by 16 - SYS_CLK_OUT is 1/16 of SYS_CLK source |

**Bits 10:8 – SLW_DIV[2:0]** Divisor steps are used when doing slewed clock switches
**Note:** Each Divisor step lasts for four clocks

| Value | Description |
|---|---|
| 000 | No Divisor is used |
| 001 | Divide by 2 ($2^1$), then no divisor is used |
| 010 | Divide by 4 ($2^2$), then by 2, then no divisor is used |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 011 | Divide by 8 ($2^3$), then by 4, then by 2, then no divisor is used |
| 100 | Divide by 16 ($2^4$), then by 8, then by 4, then by 2, then no divisor is used |
| . . . | ... |
| 111 | Divide by 128 ($2^7$), then by 64, then by 32, then by 16, then by 8, then by 4, then by 2, then no divisor is used |

**Bit 2 – SLW_UP** Enables clock slew for switching up to faster clocks

| Value | Description |
|---|---|
| 0 | Disables Clock Slewing |
| 1 | Enables Clock Slewing on a clock switch or exits from the Standby Sleep mode |

**Bit 1 – SLW_DN** Enables clock slew for switching down to slower clocks

| Value | Description |
|---|---|
| 0 | Disables Clock Slewing |
| 1 | Enables Clock Slewing on a clock switch |

**Bit 0 – SLW_BUSY** Clock Switch Slewing Active Status Bit-Read-Only

| Value | Description |
|---|---|
| 0 | Clock Switch has reached its final value |
| 1 | Clock frequency is being actively slewed |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.14. Clock Status

**Name:** CLKSTAT
**Offset:** 0x70
**Reset:** 0x00000000

**Note:** The corresponding RDY bits are updated only after clock switch request is initiated via OSCCON.NOSC[3:0].

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | PB1RDY | SYSRDY | EPLL1RDY |
| Access | | | | | | R/HS/HC | R/HS/HC | R/HS/HC |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SPLLRDY | UPLLRDY | EPLL2RDY | LPRCRDY | SOSCRDY | POSCRDY | SPLL1RDY | FRCRDY |
| Access | R/HS/HC | R/HS/HC | R/HS/HC | R/HS/HC | R/HS/HC | R/HS/HC | R/HS/HC | R/HS/HC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 10 – PB1RDY** PB1 Clock Ready Status value

| Value | Description |
|---|---|
| 1 | PB1 clock is stable and ready |
| 0 | PB1 clock is not stable and not ready |

**Bit 9 – SYSRDY** System Clock Ready Status value

| Value | Description |
|---|---|
| 1 | System clock is stable and ready |
| 0 | System clock is not stable and not ready |

**Bit 8 – EPLL1RDY** Ethernet PLL1 Ready Status value

| Value | Description |
|---|---|
| 1 | EPLL is stable and ready |
| 0 | EPLL is not stable and not ready |

**Bit 7 – SPLLRDY** System PLL Ready Status value

| Value | Description |
|---|---|
| 1 | System PLL is stable and ready |
| 0 | System PLL is not stable and not ready |

**Bit 6 – UPLLRDY** USB PLL Ready Status value


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | USB PLL is stable and ready |
| 0 | USB PLL is not stable and not ready |

**Bit 5 – EPLL2RDY** Ethernet PLL2 Ready Status value

| Value | Description |
|---|---|
| 1 | EPLL is stable and ready |
| 0 | EPLL is not stable and not ready |

**Bit 4 – LPRCRDY** LPRC Ready Status value

| Value | Description |
|---|---|
| 1 | LPRC is stable and ready |
| 0 | LPRC is not stable and not ready |

**Bit 3 – SOSCRDY** SOSC Ready Status value

| Value | Description |
|---|---|
| 1 | SOSC is stable and ready |
| 0 | SOSC is not stable and not ready |

**Bit 2 – POSCRDY** Primary Oscillator Ready Status value

| Value | Description |
|---|---|
| 1 | POSC is stable and ready |
| 0 | POSC is not stable and not ready |

**Bit 1 – SPLL1RDY** System PLL (Clock-1) Ready Status value

| Value | Description |
|---|---|
| 1 | SPLL_CLK1 is stable and ready |
| 0 | SPLL_CLK1 is not stable and not ready |

**Bit 0 – FRCRDY** FRC Ready Status value

| Value | Description |
|---|---|
| 1 | FRC is stable and ready |
| 0 | FRC is not stable and not ready |


© 2025 Microchip Technology Inc. and its subsidiaries

### 17.6.15. User Clock Diagnostics Control

**Name:** CLK_DIAG
**Offset:** 0x78
**Reset:** 0x00000000

**Note:** The system unlock sequence must be done before this register can be written.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | NMICTR15 | NMICTR14 | NMICTR13 | NMICTR12 | NMICTR11 | NMICTR10 | NMICTR9 | NMICTR8 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | NMICTR7 | NMICTR6 | NMICTR5 | NMICTR4 | NMICTR3 | NMICTR2 | NMICTR1 | NMICTR0 |
| 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 | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | UPLL_STOP | EPLL_STOP | SPLL1_STOP | LPRC_STOP | FRC_STOP | SOSC_STOP | POSC_STOP |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 – NMICTR** Internal Value of Internal NMI Counter

This field reflects the actual value of internal NMI counter.

**Bit 6 – UPLL_STOP** UPLL Clock Stop Control value

**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | UPLL clock source runs as normal |
| 1 | UPLL clock source is stopped |

**Bit 5 – EPLL_STOP** EPLL Clock Stop Control value

**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | EPLL clock source runs as normal |
| 1 | EPLL clock source is stopped |

**Bit 4 – SPLL1_STOP** SPLL Clock Stop Control value

**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | SPLL_CLK1 clock source runs as normal |
| 1 | SPLL_CLK1 clock source is stopped |

**Bit 3 – LPRC_STOP** LPRC Clock Stop Control value

**Note:** This gating logic is outside the CRU module.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | LPRC clock source runs as normal |
| 1 | LPRC clock source is stopped |

**Bit 2 – FRC_STOP** FRC Clock Stop Control value
**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | FRC clock source runs as normal |
| 1 | FRC clock source is stopped |

**Bit 1 – SOSC_STOP** SOSC Clock Stop Control value
**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | SOSC clock source runs as normal |
| 1 | SOSC clock source is stopped |

**Bit 0 – POSC_STOP** POSC Clock Stop Control value
**Note:** This gating logic is outside the CRU module.

| Value | Description |
|---|---|
| 0 | POSC clock source runs as normal |
| 1 | POSC clock source is stopped |


© 2025 Microchip Technology Inc. and its subsidiaries

# 18. Power Management Unit (PMU)

## 18.1. Overview

This section describes the Power Management Unit (PMU) in the PIC32CX-BZ6 wireless SoC with the various power saving modes. The PMU controls the sleep modes and the power domain gating of the device. Various sleep modes are provided to fit power consumption requirements. This allows the PMU to disable unused modules to save power. In the Active mode, the CPU is executing application code. When the device enters the Sleep mode, program execution is stopped and some modules and clock domains are automatically switched off according to the Sleep mode. The application code helps the device to choose the type and timing of the Sleep mode to enter. Interrupts from enabled peripherals and all enabled Reset sources can restore the device from the Sleep mode to Active mode.

**Note:** The PMU is a complex power controller that requires specific configuration and handling by the software for correct, safe operation of the SoC. The SDK and operational stacks provided by Microchip handle the start-up and operation of the PMU. Therefore, Microchip highly recommends using this software framework for all application development on the device.

## 18.2. Features

- Low Power Modes: Idle, Standby Sleep, Deep Sleep, Extreme Deep Sleep
- Sleep Walking/Dream Available in the Standby Sleep Mode
- FlexRAM (SRAM) Retention in the Standby Sleep, and Deep Sleep Mode
- I/O Lines Retention in the Deep Sleep Mode

## 18.3. Functional Description

### 18.3.1. Power Modes

The power modes and power management of different modules of the PIC32CX-BZ6 are shown in the following table.

**Table 18-1.** Power Modes and Power Management Features

| Functions | Device Power Modes | | | | | |
|---|---|---|---|---|---|---|
| | Run | Idle | SleepWalking/Dream(1) | Standby Sleep | Deep Sleep | Extreme Deep Sleep |
| CPU | ON | Clock Gated | Clock Gated | Clock Gated | OFF | OFF |
| Peripherals | ON | Idle | On Demand | Clock Gated | OFF | OFF |
| Flash | ON | Idle | On Demand | Clock Gated | OFF | OFF |
| FlexRAM | ON | Idle | On Demand | Retention Mode | OFF | OFF |
| Radio | ON | Idle | On Demand | Clock Gated | OFF | OFF |
| DSWDT | ON | ON | ON | ON | ON | OFF |
| RTCC | ON | ON | ON | ON | ON | OFF |
| FlexRAM (SRAM, retention part) | ON | Idle | On Demand | Retention Mode | Retention Mode (On Demand) | OFF |
| XTAL(16 MHz) | ON | ON | ON | ON | OFF | OFF |
| FRC | ON | ON | On Demand | OFF | OFF | OFF |
| LPRC | ON | ON | ON | ON | ON | OFF |
| SPLL | ON | ON | On Demand | OFF | OFF | OFF |
| SOSC | ON | ON | ON | ON | ON | OFF |


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:**

1. Dream (Sleep Walking) is not a mode by itself but is a companion mode to the Standby Sleep mode. This mode cannot be entered directly through a software command.

All transitions from the Run state to any of the low-power states is simply initiated by the WFI command from the CPU, however, prior to initiating the WFI command.

#### 18.3.1.1. Operation

All power-saving modes are controlled by sub-systems: XDS/DS Controller, CRU, PMU Controller and Wireless Subsystem. To enter into the different low power modes, the software performs the following actions:

- Disabling all interrupts
- Setting up the DSCON.DSEN, OSCCON.SLPEN, OSCCON.DRMEN and Wireless Subsystem Sleep mode control bits
- Set the Wireless Subsystem into the Low Power mode
- Enable the appropriate wake-up events
- Checking for any pending interrupts and, if present, abort Deep Sleep mode and service the interrupt
- If no pending interrupts, issue a SLEEP/WFI command from the CPU to get into the appropriate Low Power mode

The Low-Power modes entry and exit commands and wake-up resources are shown in the following table.

**Table 18-2.** Low-Power Saving Modes Entry and Exit Commands and Wake-up Resources

| Device Power Modes | Entry Commands | Wake-up Resources |
|---|---|---|
| Run | — | — |
| Idle | WFI Instruction + ~OSCCON.SLPEN | IRQ, Reset, Others(1) |
| Dream | OSCCON.DREAM + Event + Standby Sleep mode | IRQ, Reset, Others |
| Standby Sleep | ~DSCON.DSEN+OSCCON.SLPEN +Wireless Sleep followed by WFI | IRQ, Reset, Others |
| Deep Sleep | DSCON.DSEN + {RTCC (Enabled) or DSWDT (Enabled)} + Wireless Sleep followed by WFI | INT0, RTC0, DSWDT, Reset |
| Extreme Deep Sleep | DSCON.DSEN + {RTCC (Disabled) and DSWDT (Disabled) and INT0 (Enabled)} + Wireless Sleep followed by WFI | INT0, Reset |

**Note:**

1. Others = System Wake-up (Dream), WDT (Timeout Event), DMT (Timeout Event), PLVD Event, PMU Event, External NMI/INT, DSU/ICD Debug Event, CPU Debug Event.

#### 18.3.1.2. Run Mode

In the Run mode, the CPU is actively executing code. This mode provides normal operation of the processor and all peripherals that are currently enabled. On power-up, the device automatically gets into the Run mode. There are no special instructions required to get into the Run mode

#### 18.3.1.3. Idle Mode

In the Idle mode, all active peripherals can be clocked but the CPU core is clock gated off and no code is executed. This mode can be useful for applications that only require the processor to run when an interrupt occurs.

The device enters the Idle mode, when the OSCCON.SLPEN bit is low and the CPU executes a WFI instruction.


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:** When exiting from Standby Sleep mode, the CRU transitions the system to the Idle mode before transitioning to the RUN mode. This transition to the Idle mode is used to support the Dream mode and always occurs regardless of the CRU transitioning to the Dream mode or the Run mode. Therefore, when exiting the Standby Sleep mode, both the RCON.SLEEP and RCON.IDLE bits are going to be set.

#### 18.3.1.4. Dream Mode

In the Dream mode (or Sleep Walking mode), the CPU is clock gated but peripherals can be turned on on-demand by events related to those peripherals. No code is executed.

When the DRMEN bit in the OSCCON register is set, it allows the DMA controller to switch between the Idle mode and the Standby Sleep mode. When the OSCCON.SLPEN bit is set and the OSCCON.DRMEN bit is set, the CRU monitors the DMAC to ensure all transfers are complete before going into the Standby Sleep mode.

If OSCCON.SLPEN = '1', OSCCON.DRMEN = '1' and peripheral clock requests are active, the CRU goes into the Idle mode until all peripheral clock requests are non-active; at which time the CRU goes into the Standby Sleep mode.

If OSCCON.SLPEN is not set, the DRMEN bit has no affect as the DMA clocks are still running in the Idle mode.

If the CRU recognizes a wake/interrupt event whose priority wakes the DMA but not the CPU, the CRU transitions to the Idle mode. Therefore, the DMA can perform the needed operations and, when the DMA is finished, the CRU go backs to the Standby Sleep mode. During this time, the CPU is still asleep.

If the wake event is such that the CPU must handle the event, the whole system exits the Standby Sleep mode and transitions back to the Run mode.

#### 18.3.1.5. Standby Sleep Mode

In the Standby Sleep mode, the FlexRAM (SRAM) is in the Retention mode while the CPU and most peripherals are clock gated off and no code is executed.

**Standby Sleep Entry**

The Standby Sleep mode is entered when DSCON.DSEN is clear and the OSCCON.SLPEN bit is set and the CPU executes a WFI instruction. This causes the device clocks to be held low ('0').

Entry into the Standby Sleep mode from any other mode does not require a clock switch. This is due to the fact that when the device is in the Standby Sleep mode, no clocks are required.

**Note:** If software writes to the CRU SFR before going into the Standby Sleep mode, it is recommended to perform read on the SFR to flush the write before executing the WFI instruction that initiates the Standby Sleep mode.

**Standby Sleep Exit**

The device comes out of Standby Sleep mode and starts running with the FRC as the clock source and performs an automatic clock switch to the selected clock source.

#### 18.3.1.6. Deep Sleep Mode

In the Deep Sleep mode, the FlexRAM (SRAM) operates in Retention mode (on demand), while the CPU and peripherals are turned OFF, and no code is executed. For available peripherals that can act as wake-up sources, see *Low-Power Saving Modes Entry and Exit Commands and Wake-up Resources* table in *Operation* from Related Links.

**Deep Sleep Mode Entry**

The Deep Sleep mode is entered by performing the following steps:

1. Disable all interrupts.
2. Set Wireless Subsystem into the Low-Power mode.


© 2025 Microchip Technology Inc. and its subsidiaries

3. Set the DSEN bit in the DSCON register.
4. Enable the Deep Sleep wake-up source (See *Low-Power Saving Modes Entry and Exit Commands and Wake-up Resources* table in *Operation* from Related Links.).
5. Check for any pending interrupts and if present, abort Deep Sleep mode and service the interrupt.
6. If there are no pending interrupts, then issue a SLEEP/WFI command from the CPU.

To minimize the chance that Deep Sleep is spuriously entered, the SLEEP/WFI command must be issued as the next instruction following the setting of the DSCON.DSEN bit. This sequence can still be interrupted by interrupts and other system latencies but does not prevent the Deep Sleep mode being entered once the SLEEP/WFI command is executed. The DSEN bit is then automatically cleared when exiting the Deep Sleep mode.

**Note:** The DSWSRC register clears automatically when the DSEN bit is set, regardless of whether the Deep Sleep mode is actually entered or not. Therefore, software must read this entire register after exiting the Deep Sleep mode and before re-enabling the Deep Sleep mode.

**Deep Sleep Mode Exit**

The Deep Sleep mode exits on any of the following events:

1. Device exits Deep Sleep due to a wake-up event.
   a. POR event (de-assertion) on the VDD supply.
   b. DSWDT time-out (if DSWDT is enabled). When the DSWDT timer times out, the Deep Sleep mode will be
      exited.
   c. RTCC alarm (if RTCC is enabled).
   d. Assertion (0) of the ($\overline{\text{MCLR}}$) pin.
   e. Assertion of the INT0 pin (if the interrupt was enabled before the Deep Sleep mode was entered). The polarity configuration (refer to CFGCON0.INT0P) is used to determine the assertion level (`0` or `1`) of the pin that causes an exit from the Deep Sleep mode.
      **Note:** Any interrupts pending when entering the Deep Sleep mode are cleared, and exiting from the Deep Sleep mode requires a change on the INT0 pin while in the Deep Sleep mode.
2. The DSEN bit is automatically cleared.
3. Read the Deep Sleep Status bit and clear it.
4. Read the DSSEMA1 registers (optional).
5. When
   all state-related configuration is complete, clear the DSSR bit in the DSCON register.
6. Device releases all held logic and/or I/Os. Until this occurs, the control and data bits for the I/Os have no effects on the actual I/O state.
7. Software resumes normal operation.

**Related Links**

Operation

#### 18.3.1.6.1. Enabling Retention RAM in the Deep Sleep Mode

There is no separate backup SRAM available for the PIC32CX-BZ6 family of devices. Any context saving needs to be done in the data FlexRAM (SRAM). Depending on the size requirement of the data to be retained while in the Deep Sleep mode, the right size FlexRAM (SRAM) may be selected in the WCMSIZ register. See *WCMSIZ* from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**

WCMSIZ

WCMSIZ Register

#### 18.3.1.6.2. Software Restore

Even though exiting from the Deep Sleep mode looks like a POR to the device, it is possible to save the connected state of the device in FlexRAM (SRAM) memory. A mechanism, called “software restore”, allows software to indicate when the re-configuration of the device has finished, and it is safe to release the state of this logic and/or I/Os. This is done by clearing the DSCON.DSSR bit.

#### 18.3.1.6.3. Zero-Power BOR (ZPBOR)

The ZPBOR is a low-power BOR used during the Deep Sleep operation. The ZPBOR is enabled only when the CFGCON4.DSZPBOREN configuration bit is set as ‘1’. See *CFGCON4(L)* from Related Links.

**Related Links**

CFGCON4(L)

Configuration Control Register 4

#### 18.3.1.6.4. Deep Sleep Watchdog Timer (DSWDT)

The Deep Sleep Watchdog Timer is a configurable timer with a time-out range of 1 ms to over 24 days. The DSWDT is configured using the CFGCON4.DSWDTEN, CFGCON4.DSWDTPS[3:0] and CFGCON4.DSWDTLPRC. See *CFGCON4(L)* from Related Links. The DSWDT is a separate timer from the device’s WDT that is used in the Run mode. The device’s WDT does not have to be enabled for the DSWDT to function. Entry into the Deep Sleep mode automatically resets the DSWDT.

**Related Links**

CFGCON4(L)

Configuration Control Register 4

### 18.3.1.7. Extreme Deep Sleep (XDS) Mode

In the Extreme Deep Sleep mode, INT0, which is the external interrupt pin, alone is active and can wake up the device. Exiting XDS is equivalent to POR. The RCON register provides the status whether it is a normal power up or exiting from XDS.

**Extreme Deep Sleep Mode Entry**

1. Disable all the interrupts except INT0 (as desired).
2. Disable RTCC, DSWDT and WCM Retention.
3. Set the DSEN bit in the DSCON register.
4. Check for the pending interrupts, then, if present, abort the Extreme Deep Sleep mode and service the interrupt.
5. If there are no pending interrupts, issue a WFI command from the CPU.
   To minimize the chance of entering the Extreme Deep Sleep mode spuriously, it is recommended that the WFI command be issued as the next instruction following the setting of the DSCON.DSEN bit. This sequence can still be interrupted by interrupts and other system latencies, but it does not prevent the system entering the Extreme Deep Sleep mode once the WFI command is executed.

   The DSEN bit is, then, automatically cleared when exiting the Extreme Deep Sleep mode.

**Extreme Deep Sleep Mode Exit**

The Extreme Deep Sleep mode exits on any of the following events:

1. POR event on the VDD supply.
2. Assertion of the ($\overline{\text{MCLR}}$) pin or INT0 pin.


© 2025 Microchip Technology Inc. and its subsidiaries

## 18.4. Register Summary

See *PMU* module in the *Product Memory Mapping Overview* from Related Links for base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00<br>...<br>0x53 | Reserved | | | | | | | | | |
| 0x54 | WCMCFG | 7:0 | | | | | | | | |
| | | 15:8 | | SRAM2_CFG[2:0] | | | | SRAM1_CFG[2:0] | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x58<br>...<br>0x5F | Reserved | | | | | | | | | |
| 0x60 | WCMSIZ | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | SRAM1_SIZE[1:0] | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 18.5. 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.

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.

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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 18.5.1. WCMCFG Register

**Name:** WCMCFG
**Offset:** 0x54
**Reset:** 0x00001100
**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 |
|---|---|---|---|---|---|---|---|---|
| | | SRAM2_CFG[2:0] | | | | SRAM1_CFG[2:0] | | |
| Access | | R/W | R/W | R/W | | R/W | R/W | R/W |
| Reset | | 0 | 0 | 1 | | 0 | 0 | 1 |

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

**Bits 14:12 – SRAM2_CFG[2:0]** CMCC Memory Configuration in the Standby Sleep Mode

**Notes:**

1. This field is only writable when CFGCON0.PMULOCK = 0.
2. These bits are only applicable for controlling the state of memory in the Standby Sleep mode.
3. Memories cannot be completely turned off dynamically in the PIC32CX-BZ6 (unless in DS and 1.2V core supply is OFF). Therefore, option '000' is reserved.
4. The power consumption of these modes are: ON > NAP ~ RET+NAP.

| Value | Description |
|---|---|
| 1xx | CMCCRAM in the ON mode |
| 011 | CMCCRAM in the NAP mode |
| 001 | CMCCRAM in the RET + NAP mode |
| 000 | Reserved |

**Bits 10:8 – SRAM1_CFG[2:0]** FLEXRAM (SRAM) Configuration in the Standby Sleep Mode

**Notes:**

1. This field is only writable when CFGCON0.PMULOCK = 0.
2. These bits are only applicable for controlling the state of memory in the Standby Sleep mode.
3. Memories cannot be completely turned off dynamically in the PIC32CX-BZ6 (unless in DS and 1.2V core supply is OFF). Therefore, option '000' is reserved.
4. The power consumption of these modes are : ON > NAP ~ RET+NAP.

| Value | Description |
|---|---|
| 1xx | FLEXRAM in the ON mode |
| 011 | FLEXRAM in the NAP mode |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 001 | FLEXRAM in the RET + NAP mode |
| 000 | Reserved |


© 2025 Microchip Technology Inc. and its subsidiaries

### 18.5.2. WCMSIZ Register

**Name:** WCMSIZ
**Offset:** 0x60
**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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | SRAM1_SIZE[1:0] | |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

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

**Bits 9:8 – SRAM1_SIZE[1:0]** Flex RAM Retention Size Configuration in the Deep Sleep Mode
**Note:** This field is only writable when CFGCON0.PMULOCK = 0.

| Value | Description |
|---|---|
| 11 | 64K Flex RAM is available in retention |
| 10 | 32K Flex RAM is available in retention |
| 01 | 16K Flex RAM is available in retention |
| 00 | Flex RAM is completely powered OFF in the Deep Sleep mode |


© 2025 Microchip Technology Inc. and its subsidiaries

## 18.6. Register Summary

See *DSCON* module in the *Product Memory Mapping Overview* from Related Links for base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | DSCON | 7:0 | | | | | | | ZPBOR | DSSR |
| | | 15:8 | DSEN | | XSEMAEN | RTCPWREQ | | | | RTCCWDIS |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 | DSWSRC | 7:0 | FAULT | | | DSWDT | RTCC | MCLR | | |
| | | 15:8 | | | | | | | | INT0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x08 | DSSEMA1 | 7:0 | DSSEMA1[7:0] | | | | | | | |
| | | 15:8 | DSSEMA1[15:8] | | | | | | | |
| | | 23:16 | DSSEMA1[23:16] | | | | | | | |
| | | 31:24 | DSSEMA1[31:24] | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 18.7. 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.

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.

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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 18.7.1. DS Control

**Name:** DSCON
**Offset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | DSEN | | XSEMAEN | RTCPWREQ | | | | RTCCWDIS |
| Access | R/W/HC | | R/W | R/W | | | | R/W |
| Reset | 0 | | 0 | 0 | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | ZPBOR | DSSR |
| Access | | | | | | | R/W/C/HS | R/C/HS/HC |
| Reset | | | | | | | 0 | 0 |

**Bit 15 – DSEN** Deep Sleep Enable bit

| Value | Description |
|---|---|
| 0 | Enters the Standby Sleep mode on a WFI command |
| 1 | Enters the Deep Sleep mode on a WFI command |

**Bit 13 – XSEMAEN** XSEMA General Purpose Registers Enable bit

| Value | Description |
|---|---|
| 0 | No general purpose register retention in the Deep Sleep mode |
| 1 | Enables the general purpose register retention in the Deep Sleep mode |

**Bit 12 – RTCPWREQ** RTCC Module Disable bit
To enable RTCC function, RTCC module level registers must be programmed in addition to the DSCON.RTCPWREQ bit.

| Value | Description |
|---|---|
| 0 | Enables the RTCC module |
| 1 | Disables the RTCC module |

**Bit 8 – RTCCWDIS** RTCC Wake-up Disable bit

| Value | Description |
|---|---|
| 0 | Enables the wake-up from RTCC |
| 1 | Disables the wake-up from RTCC |

**Bit 1 – ZPBOR** Deep Sleep BOR Event Status bit


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | CFGCON4.DSZPBOREN is disabled or VDD did not drop below the DSBOR threshold during the Deep Sleep mode |
| 1 | CFGCON4.DSZPBOREN is enabled and VDD dropped below the DSBOR threshold during the Deep Sleep mode<br>**Note:** Unlike all other events, a DSBOR event does not cause a wake-up from the Deep Sleep mode. This bit is present only as a status bit. |

**Bit 0 – DSSR** I/O pin State Release bit

| Value | Description |
|---|---|
| 0 | Release I/O pins and allow their respective TRIS and LAT bits to control their states |
| 1 | Upon waking from Deep Sleep, the I/O pins maintain their previous states (Not user settable).<br>**Note:** The DSSR register bit is readable and can be cleared (but not set) by the software.<br>This bit is automatically set when entering the Deep Sleep mode. While exiting the Deep Sleep mode, due to any wake-up source other than $\overline{\text{MCLR}}$, when all state-related configuration is complete, the software must clear the DSSR bit. When the DSSR bit is cleared, the I/Os will be controlled by their I/O registers. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 18.7.2. DSWSRC

**Name:** DSWSRC
**Offset:** 0x04
**Reset:** 0x00000000

The DSWSRC register is only writable by software when DSCON.DSSR = 0.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | INT0 |
| Access | | | | | | | | R/W/C/HS |
| Reset | | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | FAULT | | | DSWDT | RTCC | MCLR | | |
| Access | R/W/C/HS | | | R/W/C/HS | R/W/C/HS | R/W/C/HS | | |
| Reset | 0 | | | 0 | 0 | 0 | | |

**Bit 8 – INT0** Deep Sleep Interrupt-on-change #0 bit

| Value | Description |
|---|---|
| 0 | Interrupt-on-change #0 (INT0) is not asserted during the Deep Sleep |
| 1 | Interrupt-on-change #0 (INT0) is asserted during the Deep Sleep |

**Bit 7 – FAULT** Deep Sleep Fault Detected bit

| Value | Description |
|---|---|
| 0 | No fault is detected during deep sleep |
| 1 | A fault on one or more fault detect keeper cells is detected during deep sleep |

**Bit 4 – DSWDT** Deep Sleep Watchdog Timer Time-out bit

| Value | Description |
|---|---|
| 0 | DSWDT does not time out during the Deep Sleep |
| 1 | DSWDT timed out during the Deep Sleep |

**Bit 3 – RTCC** Deep Sleep Real Time Clock and Calendar Alarm bit

| Value | Description |
|---|---|
| 0 | Deep Sleep RTC and calendar does not trigger an alarm during the Deep Sleep |
| 1 | Deep Sleep RTC and calendar triggers an alarm during the Deep Sleep |

**Bit 2 – $\overline{\text{MCLR}}$** Deep Sleep MCLR Event bit

| Value | Description |
|---|---|
| 0 | ($\overline{\text{MCLR}}$) pin is not active or is active but not asserted during the Deep Sleep |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | ($\overline{\text{MCLR}}$) pin is active and is asserted during the Deep Sleep |


© 2025 Microchip Technology Inc. and its subsidiaries

### 18.7.3. DSSEMA1 Register

**Name:** DSSEMA1
**Offset:** 0x08
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DSSEMA1[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 |
|---|---|---|---|---|---|---|---|---|
| | DSSEMA1[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 |
|---|---|---|---|---|---|---|---|---|
| | DSSEMA1[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 |
|---|---|---|---|---|---|---|---|---|
| | DSSEMA1[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 – DSSEMA1[31:0]** Deep Sleep Persistent General Purpose Register bits
The contents of the DSSEMA1 register are retained, even in the Deep Sleep mode. The DSSEMA1 is disabled by default in the Deep Sleep mode but can be enabled with the XSEMAEN bit (DSCON[13]). All register bits are reset only in the case of a VDD POR event outside of the Deep Sleep mode.


© 2025 Microchip Technology Inc. and its subsidiaries

# 19. Watchdog Timer (WDT)

## 19.1. Overview

The Watchdog Timer (WDT) can be used to detect system software malfunctions by resetting the device if the WDT is not cleared periodically in software. The user can configure the WDT in the Windowed mode or non-Windowed mode. Various WDT time-out periods can be selected using the WDT postscaler. The WDT can also be used to wake the device from the Standby Sleep or Idle mode.

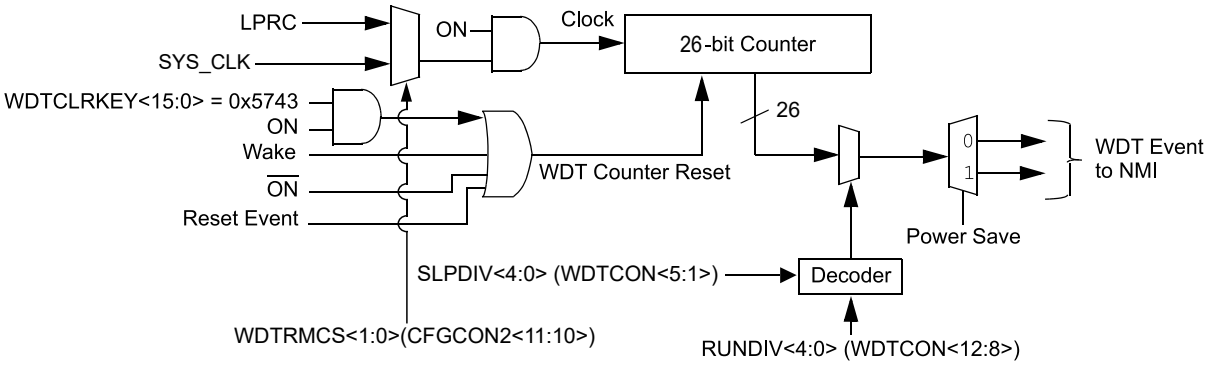
## 19.2. Features

- Configuration Using Fuses (DEVCFG) or Software Controlled
- Up to 32 Configurable Time-Out Periods
- Can Wake the Device from the Standby Sleep or Idle Mode
- Independent Run and the Standby Sleep Mode Counters
- WDT May Use Alternate Clock Source and Postscaler for Run Mode Counter
- Independent 5-Bit Postscalers for Run and the Standby Sleep Mode Counters
- Hardware and Software Enabled
- Two Clock Sources
- Windowed WDT

**Note:** When the CPU is running on the same clock or clock frequency as the WDT, the lowest pre-scale values may not allow the CPU to have enough time to reset the WDT before it expires.

## 19.3. Block Diagram

**Figure 19-1.** Block Diagram




## 19.4. Watchdog Timer Operation

The primary function of the WDT is to reset the processor in the event of a software malfunction or to wake up the processor in the event of a time-out while in the Standby Sleep mode.

If enabled, the WDT increments until it overflows or times out. A WDT time-out will force a device Reset, except during the Standby Sleep or Idle mode. To prevent a WDT time-out Reset, the user application must periodically clear the WDT by writing a keyword 0x5743 to the WDTCLRKEY[15:0] bits (WDTCON[31:16]) through a single 16-bit write.

**Related Links**

WDTCON

WDTCON - Watchdog Timer Control Register


© 2025 Microchip Technology Inc. and its subsidiaries

### 19.4.1. Modes of Operation

The WDT has two modes of operation:

- Non-Windowed
- Programmable Windowed

The Programmable Windowed mode can be enabled by setting the Watchdog Window Enable (WDTWINEN) bit (WDTCON[0]). In Programmable Windowed mode, software can clear the WDT only when the counter is in its final window before a period match occurs. There are four window size options. This window is active when the timer counter is greater than a predetermined value for each option. Any attempts to clear the WDT when the window is not active causes a device Reset. In Non-Windowed mode, software can clear the WDT anytime before the period match occurs.

### 19.4.2. Enabling and Disabling the WDT

The WDT is enabled or disabled by the device configuration or controlled through software by writing to the WDTCON register. See *WDTCON* from Related Links for more details.

**Related Links**

WDTCON

WDTCON - Watchdog Timer Control Register

### 19.4.3. Device Configuration Controlled WDT

If the DEVCFG2.WDTEN Configuration bit is set, the WDT is always enabled. The ON control bit (WDTCON[15]) reflects this by reading a '1'. In this mode, the ON bit cannot be cleared in software. The DEVCFG2.WDTEN Configuration bit will not be cleared by any form of Reset. To disable the WDT, the configuration must be rewritten to the device.

**Note:** The WDT is enabled by default on an unprogrammed device.

The DEVCFG2.WINDIS Configuration bit can be used to enable or disable the Programmable Windowed mode. The window size for the WDT Programmable Windowed mode can be configured using the DEVCFG2.WINSZ Configuration bits.

### 19.4.4. Software Controlled WDT

If the DEVCFG2.WDTEN Configuration bit is '0', the WDT module can be enabled or disabled (the default condition) by software. In this mode, the ON bit (WDTCON[15]) reflects the status of the WDT under software control. A '1' indicates the WDT module is enabled and a '0' indicates it is disabled. If the DEVCFG2.WINDIS Configuration bit is '0', the WDT Programmable Windowed mode can be enabled or disabled by software. The Programmable Windowed mode can be configured using the WDTWINEN bit (WDTCON[0]). A '1' indicates that Programmable Windowed mode is enabled and '0' indicates it is disabled. The window sizes can be configured by setting the DEVCFG2.WINSZ Configuration bits only and cannot be set in software.

The WDT is enabled in software by setting the Watchdog Timer control bit, ON (WDTCON[15]). The ON control bit is cleared on any device Reset. The bit is not cleared upon a wake from Standby Sleep or exit from Idle mode. The software WDT option allows the user to enable the WDT for critical code segments and to disable the WDT during noncritical segments for maximum power savings. This bit can also be used to disable the WDT while the device is awake to eliminate the need for WDT servicing, then re-enable it before the device is put into the Idle mode or Standby Sleep mode to wake the device at a later time.

#### 19.4.4.1. Watchdog Timer Programmable Window

The window size is determined by the Configuration bits, DEVCFG2.WINSZ and WDTPS. In the Programmable Windowed mode (WDTCON.WDTWINEN = 1), the WDT must be cleared based on the setting of the Window Size Configuration bits (DEVCFG2.WINSZ[1:0]), see the following figure. These bit settings are:


© 2025 Microchip Technology Inc. and its subsidiaries

- 11
  = WDT window is 25% of the WDT period
- 10
  = WDT window is 37.5% of the WDT period
- 01
  = WDT window is 50% of the WDT period
- 00
  = WDT window is 75% of the WDT period

If the WDT is cleared before the allowed window, a system Reset is immediately generated. See *Clock and Reset Unit (CRU)* from Related Links for more information on which type of Reset occurs.

The Windowed mode is useful for resetting the device during unexpected quick or slow execution of a critical portion of the code.

**Figure 19-2.** Programmable Windowed WDT

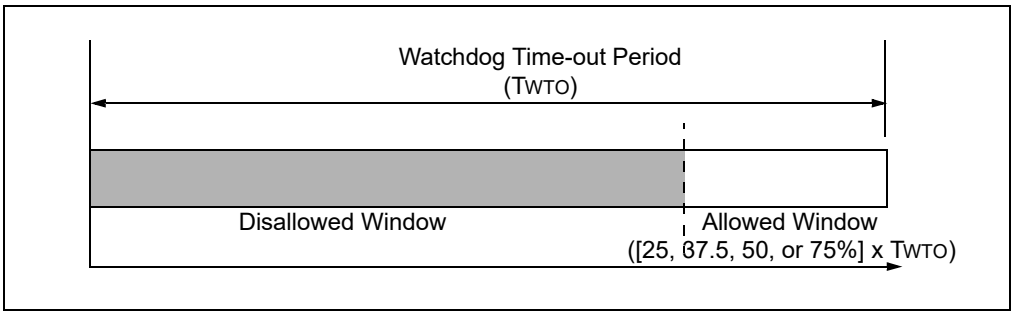
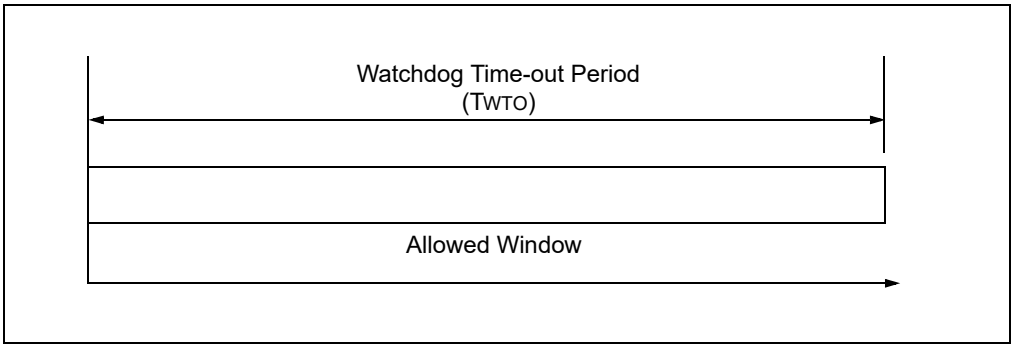



**Figure 19-3.** Non-Windowed WDT




**Related Links**
Clock and Reset Unit (CRU)

### 19.4.5. WDT Operation in Power-Saving Modes

The WDT if enabled, will continue operation in the Standby Sleep mode or Idle mode. The WDT module may be used to wake the device from Standby Sleep mode or Idle mode. When the WDT times out in a Standby Sleep mode or Idle mode, a Non-Maskable Interrupt (NMI) is generated and the WDTO bit (RCON[4]) is set. The NMI vectors execution to the CPU start-up address, but does not Reset registers or peripherals. If the device is in Standby Sleep, the SLEEP status bit (RCON[3]) will also be set. If the device is in Idle, the IDLE status bit (RCON[2]) will also be set. These bits allow the start-up code to determine the cause of the wake-up.


© 2025 Microchip Technology Inc. and its subsidiaries

### 19.4.6. WDT NMI Reset Delay

It is possible to program a delay time between a WDT event and a device Reset using the NMI reset counter. See *Resets* from Related Links for more details.

**Related Links**
Resets

### 19.4.7. Resetting the WDT

The following events will Reset both internal WDT counters:

- Disabling the WDT via the ON bit on any device Reset
- Any counter value greater than the selected WDT period

The following event will Reset the Run Mode Counter:

- Detection of a correct write value (0x5743) to the WDTCLRKEY[15:0] bits (WDTCON[31:16])

The following event will Reset the Standby Sleep Mode Counter:

- Exiting from Idle or Standby Sleep due to WDT event

**Note:** The WDT is not Reset when the device enters a Power-Saving mode. The WDT module must be serviced prior to entering a Power-Saving mode.

### 19.4.8. WDT Period Selection

The Sleep mode counter always uses the 32 kHz LPRC clock source. The Run mode counter uses the clock source selected by the DEVCFG2.WDTRMCS[1:0] Configuration Bits in fuses to be either the 32 kHz LPRC clock or the SYS_CLK.
**Note:** The WDT module time-out period is directly related to the frequency of the LPRC Oscillator when clock source is 32 kHz LPRC. The frequency of the LPRC Oscillator varies as a function of the device operating voltage and temperature. See *Electrical Characteristics* from Related Links for LPRC clock frequency specifications.

**Related Links**
Electrical Characteristics

### 19.4.9. WDT Postscalers

The WDT has a 5-bit postscaler to create a wide variety of time-out periods. This postscaler provides 1:1 through 1:1048576 divider ratios (see the following table). Time-out periods that range between 1 ms and 1048.576 seconds (nominal) can be achieved using the postscaler.

**Table 19-1.** WDT Time-out Period versus Postscaler Settings(1),(2)

| WDTPSR[4:0]/WDTPSS[4:0] | Postscaler Ratio | Time-out Period (Non-windowed Mode) | Time-out Period (Programmable Windowed Mode)(3) |
|---|---|---|---|
| 00000 | 1:1 | 1 ms | 0.75 ms |
| 00001 | 1:2 | 2 ms | 1.5 ms |
| 00010 | 1:4 | 4 ms | 3 ms |
| 00011 | 1:8 | 8 ms | 6 ms |
| 00100 | 1:16 | 16 ms | 12 ms |
| 00101 | 1:32 | 32 ms | 24 ms |
| 00110 | 1:64 | 64 ms | 48 ms |
| 00111 | 1:128 | 128 ms | 96 ms |
| 01000 | 1:256 | 256 ms | 192 ms |
| 01001 | 1:512 | 512 ms | 384 ms |
| 01010 | 1:1024 | 1.024s | 0.768s |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 19-1.** WDT Time-out Period versus Postscaler Settings[1],[2] (continued)

| WDTPSR[4:0]/WDTPSS[4:0] | Postscaler Ratio | Time-out Period (Non-windowed Mode) | Time-out Period (Programmable Windowed Mode)[3] |
|---|---|---|---|
| 01011 | 1:2048 | 2.048s | 1.536s |
| 01100 | 1:4096 | 4.096s | 3.072s |
| 01101 | 1:8192 | 8.192s | 6.144s |
| 01110 | 1:16384 | 16.384s | 12.228s |
| 01111 | 1:32768 | 32.768s | 24.576s |
| 10000 | 1:65536 | 65.536s | 49.152s |
| 10001 | 1:131072 | 131.072s | 98.304s |
| 10010 | 1:262144 | 262.144s | 196.608s |
| 10011 | 1:524288 | 524.288s | 393.216s |
| 10100 | 1:1045876 | 1048.576s | 786.432s |

**Notes:**

1. All other combinations result in operation as if the postscaler was set to '10100'.
2. The periods listed are based on a 32 kHz (nominal) input clock.
3. In this case, DEVCFG2.WINSZ = 00. The WDT window is 75% of the selected WDT period.

The settings are chosen using the DEVCFG2.WDTPSR[4:0] inputs for the Run mode counter and the DEVCFG1.WDTPSS[4:0] inputs for the Standby Sleep mode counter. The time-out period of the WDT is calculated as shown in the following equation:

$$WDTPeriod = 1\ ms \bullet 2^{Postscaler}$$

## 19.5. Interrupt and Reset Generation

The NMI timer provides a delay between WDT events and a device Reset. Set the delay in the System Clock counts from 0 to 255 in the NMICNT[15:0] bits (RNMICON[15:0]). If these bits are set to zero, there is no delay between the WDTO flag and a device Reset. If set to a non-zero value, the NMI interrupt has that number of system clocks to clear flags or save data for debugging purposes.

If the corresponding NMI flag (RNMICON.WDTR) is not cleared in RNMICON before the counter reaches zero, a device Reset is issued.

If the corresponding NMI flag in RNMICON is cleared before the counter reaches zero, the counter is stopped, then reloaded with the NMICNT value again and waits for another NMI event to occur. A device Reset will not be asserted in this case, and software will be able to return from this NMI interrupt.

The WDTS flag is set if there is a WDT event during the Standby Sleep/Idle mode. The WDTS flag will wake the CPU from Standby Sleep/Idle mode, but will not start the NMI counter, nor cause a Reset.

To detect a WDT Reset, the WDTO bit (RCON[4]), SLEEP bit (RCON[3]) and IDLE bit (RCON[2]) must be tested. If the WDTO bit is '1', the event was due to a WDT time-out. The SLEEP and IDLE bits can, then, be tested to determine if the WDT event occurred while the device was awake or if it was in the Standby Sleep or Idle mode.

## 19.6. Operation in Debug and Power-Saving Modes

### 19.6.1. WDT Operation in Power-Saving Modes

The WDT can be used to wake the device from the Standby Sleep or Idle modes. The WDT continues to operate in the Power-Saving modes. A time-out can, then, be used to wake the device. This allows the device to remain in Standby Sleep mode until the WDT expires or another interrupt wakes the device.

If the device does not re-enter the Standby Sleep or Idle mode following a wake-up, the WDT must be disabled or periodically serviced to prevent a device Reset.


© 2025 Microchip Technology Inc. and its subsidiaries

### 19.6.2. WDT Operation in Standby Sleep Mode

The WDT, if enabled, continues operation in the Standby Sleep mode. The WDT may be used to wake the device from the Sleep mode. When the WDT times out in Sleep, an NMI is generated and the WDTO bit (RCON[4]) is set. The NMI vectors execution to the CPU start-up address but does not Reset registers or peripherals. The Standby Sleep status bit (RCON[3]) is set indicating the device was in the Standby Sleep mode. These bits allow the start-up code to determine the cause of the wake-up.

### 19.6.3. WDT Operation in Idle Mode

The WDT, if enabled, continues operation in the Idle mode. The WDT may be used to wake the device from the Idle mode. When the WDT times out in Idle, an NMI is generated and the WDTO bit (RCON[4]) is set. The NMI vectors execution to the CPU start-up address but does not Reset registers or peripherals. The IDLE status bit (RCON[2]) is set, indicating the device was in the Idle mode. These bits allow the start-up code to determine the cause of the wake-up.

### 19.6.4. Time Delays During Wake-up

The delay between a WDT time-out and the beginning of code execution depends on the Power-Saving mode.

There is a time delay between the WDT event in the Standby Sleep mode and the beginning of code execution. The duration of this delay consists of the start-up time for the oscillator in use and the PWRT delay, if it is enabled. Unlike a wake-up from the Standby Sleep mode, there are no time delays associated with wake-up from the Idle mode. The system clock is running during the Idle mode; therefore, no start-up delays are required at wake-up.

### 19.6.5. WDT Operation in Debug Mode

The WDT is always suspended in the Debug mode, and therefore does not time-out.

## 19.7. Effects of Various Resets

Any form of device Reset clears the WDT. The Reset returns the WDTCON register to the default value and the WDT is disabled unless it is enabled by the device configuration.

**Note:** After a device Reset, the WDT ON bit (WDTCON[15]) reflects the state of the DEVCFG2.WDTEN.


© 2025 Microchip Technology Inc. and its subsidiaries

## 19.8. Register Summary

See *WDT* module in the *Product Memory Mapping Overview* from Related Links for the base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | WDTCON | 7:0 | | | SLPDIV[4:0] | | | | | WDTWINEN |
| | | 15:8 | ON | | | RUNDIV[4:0] | | | | |
| | | 23:16 | WDTCLRKEY[7:0] | | | | | | | |
| | | 31:24 | WDTCLRKEY[15:8] | | | | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 19.9. Register Description

The following are the list of conventions available in the register description:

- – R = Readable bit
- – W = Writable bit
- – U = Unimplemented bit, read as '0'
- – -n = Value at POR
- – 1 = Bit is set
- – 0 = Bit is cleared
- – x = Bit is unknown
- y = Values set from Configuration bits on POR
- Reset values are shown in hexadecimal.


© 2025 Microchip Technology Inc. and its subsidiaries

### 19.9.1. WDTCON - Watchdog Timer Control Register

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

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | WDTCLRKEY[15:8] | | | | | | | |
| Access | W | W | W | W | W | W | W | W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ON | | | RUNDIV[4:0] | | | | |
| Access | R/W | | | R | R | R | R | R |
| Reset | y | | | y | y | y | y | y |

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

**Bits 31:16 – WDTCLRKEY[15:0]** Watchdog Timer Clear Key bits
To clear the WDT to prevent a time-out, software must write the value 0x5743 to this location using a single 16-bit write. Anything other than a 16-bit write will not reset the WDT. You must use a 16-bit write for the WDTCLRKEY[15:0] bits.

**Bit 15 – ON** Watchdog Timer Enable bit
**Note:**
1. This bit only has control when the WDTEN bit (DEVCFG2/CFGCON2[23]) = 0.

| Value | Description |
|---|---|
| 1 | WDT is enabled |
| 0 | WDT is disabled |

**Bits 12:8 – RUNDIV[4:0]** Watchdog Timer Postscaler Run Counter Value bits
On Reset, these bits are set to the values of the WDTPSR[4:0] Configuration bits in CFGCON2.

**Bits 5:1 – SLPDIV[4:0]** Watchdog Timer Postscaler Sleep Counter Value bits
On Reset, these bits are set to the values of the WDTPSS[4:0] Configuration bits in CFGCON1.

**Bit 0 – WDTWINEN** Watchdog Timer Window Enable bit

| Value | Description |
|---|---|
| 1 | Enable windowed WDT |
| 0 | Disable windowed WDT |


© 2025 Microchip Technology Inc. and its subsidiaries

# 20. Deadman Timer (DMT)

## 20.1. Overview

The Deadman Timer (DMT) module is designed to enable users to be able to monitor the health of their application software by requiring periodic timer interrupts within a user-specified timing window. The DMT module is a synchronous counter and, when enabled, counts the instructions fetched and causes a system Reset if the DMT counter is not cleared within a set number of instructions. The DMT is typically connected to the system clock that drives the processor. The user specifies the timer time-out value and a mask value that specifies the range of the window, which is the range of counts that is not considered for the comparison event.

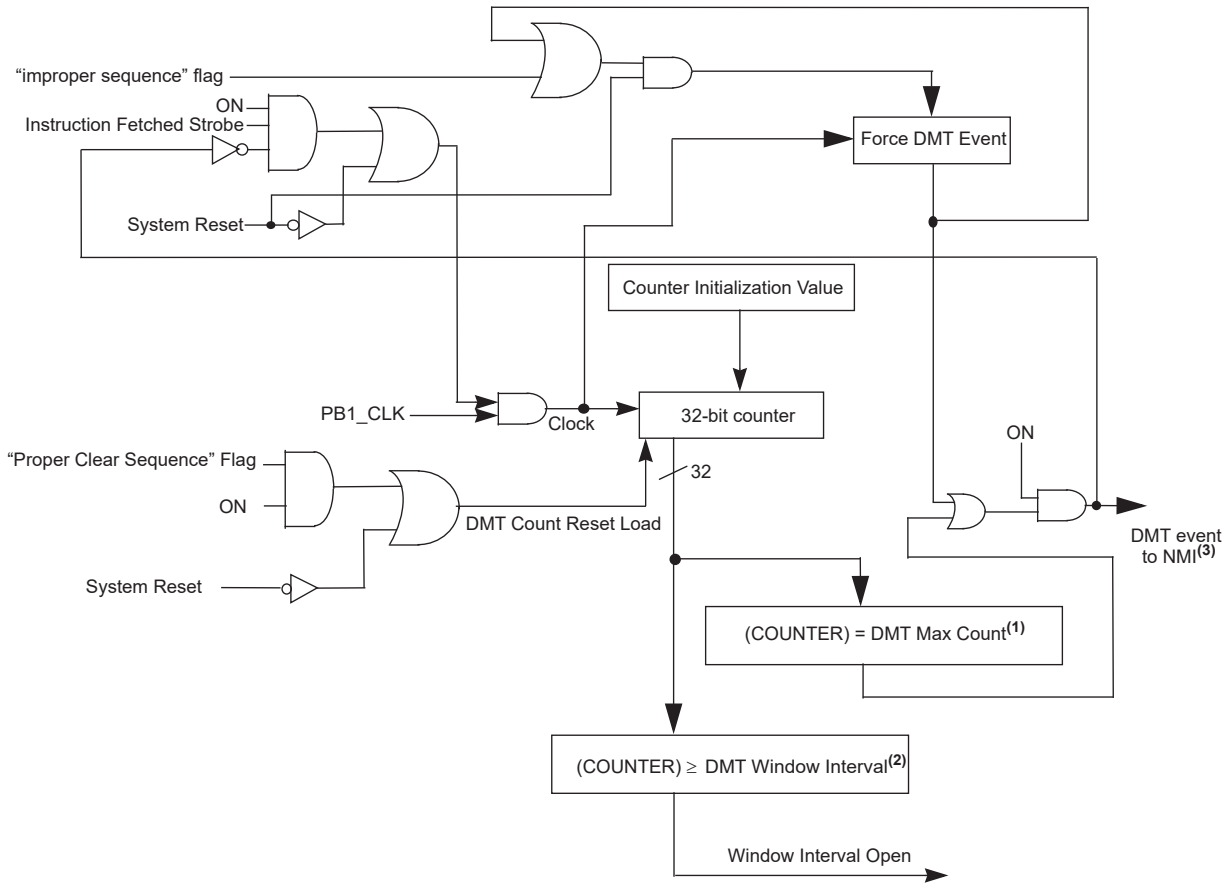
## 20.2. Features

- 32-bit Configurable Count-limit Based on Counting Instructions Fetched
- Hardware- or Software-enabled Control
- User-configurable Time-out Period or Instruction Count
- Two Instruction Sequence to Clear Timer
- 32-bit Configurable Window to Clear Timer

## 20.3. Block Diagram

The following figure illustrates the block diagram of the Deadman Timer.

**Figure 20-1.** Deadman Timer Block Diagram





© 2025 Microchip Technology Inc. and its subsidiaries

**Notes:**

1. The DMT Max Count is controlled by the DMTCNT[4:0] bits in the CFGCON2 register “Maximum = 2^31”.
2. The DMT Window Interval is controlled by the DMTINTV[2:0] bits in the CFGCON2 register.
3. For more details, see *Resets* from Related Links.

**Related Links**

Resets

## 20.4. DMT Operation

### 20.4.1. Mode of Operation

The primary function of the DMT module is to Reset the processor in the event of a software malfunction. The DMT module, which works on the system clock, is a free running instruction fetch timer that is clocked whenever an instruction fetch occurs until a count match occurs. The instructions are not fetched when the processor is in the Standby Sleep mode.

The DMT module consists of a 32-bit counter, the read-only DMTCNT register with a time-out count match value as specified by the 32-bit DMT count configuration fuse bits CFGCON2.DMTCNT[4:0]. Whenever the count match occurs, a DMT Reset event will occur and the DMTEVENT bit in DMTSTAT register is set.

A DMT module is typically used in mission-critical and safety-critical applications, where any failure of the software functionality and sequencing must be detected.

### 20.4.2. Enabling and Disabling the DMT Module

Because of the nature of the DMT, the PMD register bit is not provided to enable/disable the module. The DMT module can be enabled or disabled by the DMT ENABLE (DMTEN) bit in the Configuration Control Register 2 (CFGCON2) fuse register or it can be enabled through software by writing to the Deadman Timer Control (DMTCON) register. When the DMT is enabled, it may not be disabled without a device Reset.

If the DMTEN Configuration bit in the CFGCON2 fuse register is set, the DMT is always enabled. The ON CONTROL bit (DMTCON[15]) in the DMTCON register will reflect this by reading a ‘1’. In this mode, the ON bit (DMTCON[15]) cannot be cleared in the software. To disable the DMT, the DMTEN Configuration bit must be cleared in the CFGCON2 fuse register. When DMTEN is cleared to ‘0’ in the CFGCON2, the DMT is disabled in hardware.

Software can enable the DMT by setting the ON bit in the DMTCON register. However, for software control, the DMTEN Configuration bit in the CFGCON2 fuse register must be set to ‘0’.

### 20.4.3. DMT Count Windowed Interval

The DMT module has the Windowed Operation mode. The DMT INTERVAL (DMTINV[2:0]) bits in the CFGCON2 Fuse register sets the window interval value. The PSINTV[31:0] bits in the DMT Interval Post Status register (DMTPSINTV) allows the software to read the DMT window interval value. That means this register reads the value that is written to the DMT INTERVAL (DMTINV[2:0]) bits in the CFGCON2 Fuse register.

In the Windowed mode, software can clear the DMT only when the counter is in its final window before a count match occurs. That is, only when the DMT counter value is greater than or equal to the value written to the window interval value, can the DMT clear sequence be executed in the DMT module. If the DMT is cleared before the allowed window, a DMT Reset event is immediately generated.


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.4.4. DMT Count Selection

The DMT count is set by the DMT COUNT (DMTCNT[4:0]) Configuration bits in the CFGCON2 Fuse register. The current DMT count value can be obtained by reading the DMTCNT register.

The PSCNT[31:0] bits in the DMT COUNT POST STATUS (DMTPSCNT) register allow the software to read the maximum count selected for the DMT. The PSCNTx bit values are the values that are initially written to the DMTCNTx bits in the CFGCON2 Fuse register. Whenever the DMT event occurs, the user can always compare to see whether the current counter value in the DMTCNT register is equal to the value of the DMTPSCNT register, which holds the maximum count value.

Whenever the DMT current counter value in DMTCNT reaches the value of the DMTPSINTV register, the window interval opens permitting the user to execute the DMT clear sequence. The open window interval is indicated by the WINOPN bit in DMTSTAT register.

The UPRCNT[15:0] bits in the DMT HOLD (DMTHOLDREG) register holds the value of the last read DMT upper count values whenever DMTCNT is last read.

### 20.4.5. DMT Operation in Power-Saving Modes

As the DMT module is only incremented by instruction fetches, the count value will not change when the core is inactive. The DMT module remains inactive in the Standby Sleep and Idle modes. As soon as the device wakes-up from Standby Sleep or Idle, the DMT counter starts incrementing again for every instruction fetch.

### 20.4.6. Resetting the DMT

The DMT can be reset in two ways: one way is after a system Reset, and another way is by writing an ordered sequence to the DMT PRE-CLEAR (DMTPRECLR) register and DMT CLEAR (DMTCLR) register in a specific two-step sequence.

Clearing the DMT counter value requires the following sequence of operations:

1. The STEP1[7:0] bits in the DMTPRECLR register must be written as '01000000' (0x40). This action sets the "Enable for Clearing" state, which enables the DMT to be cleared by step 2.
2. The STEP2[7:0] bits in the DMTCLR register must be written as '00001000' (0x08). This can only be done if preceded by step 1 and if the DMT is in the open window interval.

When these values are written, following are cleared to zero:

- DMTCNT counter
- DMTPRECLR register
- DMTCLR register
- DMTSTAT register

If any value other than 0x40 is written to the STEP1x bits, the BAD1 bit in the DMTSTAT register will be set, and it causes a DMT event to occur. Any value other than 0x08 written to the STEP2x bits will cause the BAD2 bit to be set in the DMTSTAT register. Also, if step 2 is not preceded by step 1 or step 2 is not carried out in the open window interval, it causes the BAD2 flag to be set. Immediately, a DMT event will occur. In both these cases, the DMTEVENT bit in the DMTSTAT register will be set. Refer to the flowchart illustrated in the following figure.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 20-2.** Flowchart for Clearing the DMT





© 2025 Microchip Technology Inc. and its subsidiaries

## 20.5. Register Summary

See the *DMT* module in the *Product Memory Mapping Overview* from Related Links for the base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | DMTCON | 7:0 | | | | | | | | |
| | | 15:8 | ON | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x04 ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | DMTPRECLR | 7:0 | | | | | | | | |
| | | 15:8 | STEP1[7:0] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | DMTCLR | 7:0 | STEP2[7:0] | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 ... 0x2F | Reserved | | | | | | | | | |
| 0x30 | DMTSTAT | 7:0 | BAD1 | BAD2 | DMT_EVENT | | | | | WINOPN |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x34 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | DMTCNT | 7:0 | COUNTER[7:0] | | | | | | | |
| | | 15:8 | COUNTER[15:8] | | | | | | | |
| | | 23:16 | COUNTER[23:16] | | | | | | | |
| | | 31:24 | COUNTER[31:24] | | | | | | | |
| 0x44 ... 0x4F | Reserved | | | | | | | | | |
| 0x50 | DMTHOLDREG | 7:0 | UPRCNT[7:0] | | | | | | | |
| | | 15:8 | UPRCNT[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x54 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | DMTPSCNT | 7:0 | PSCNT[7:0] | | | | | | | |
| | | 15:8 | PSCNT[15:8] | | | | | | | |
| | | 23:16 | PSCNT[23:16] | | | | | | | |
| | | 31:24 | PSCNT[31:24] | | | | | | | |
| 0x64 ... 0x6F | Reserved | | | | | | | | | |
| 0x70 | DMTPSINTV | 7:0 | PSINTV[7:0] | | | | | | | |
| | | 15:8 | PSINTV[15:8] | | | | | | | |
| | | 23:16 | PSINTV[23:16] | | | | | | | |
| | | 31:24 | PSINTV[31:24] | | | | | | | |

**Related Links**

Product Memory Mapping Overview


© 2025 Microchip Technology Inc. and its subsidiaries

## 20.6. 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.

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.

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.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at their virtual addresses plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links

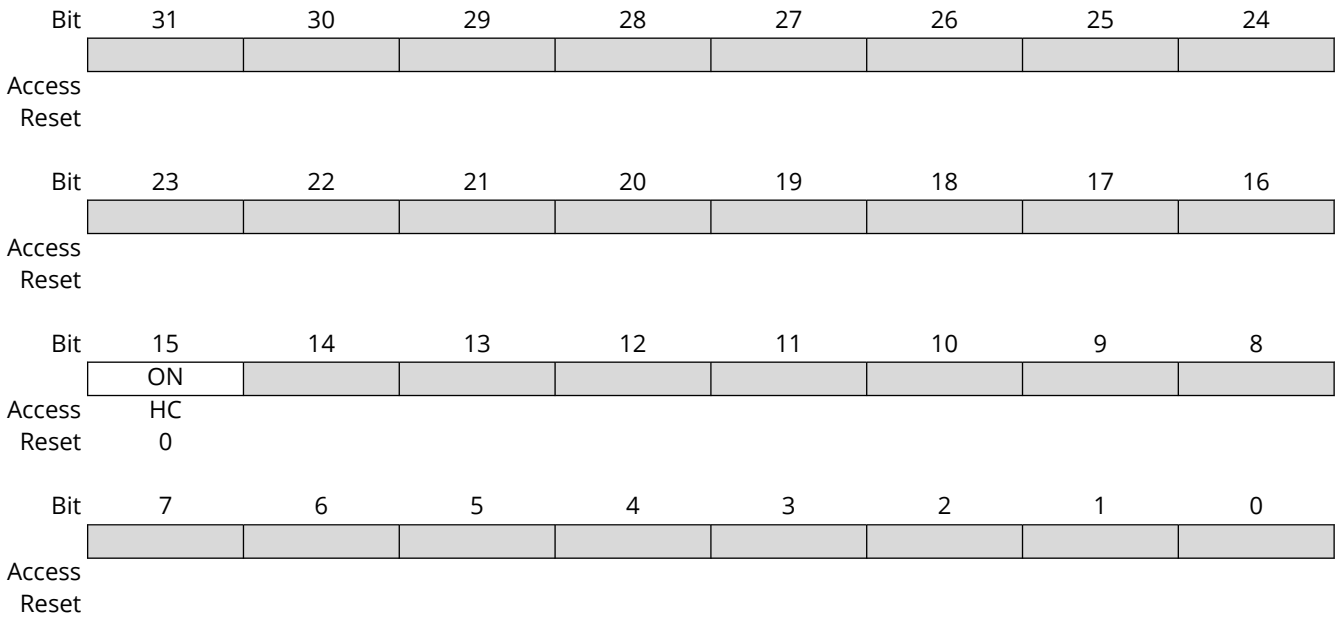
**Related Links**

CLR, SET and INV Registers


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.1. Deadman Timer Control

**Name:** DMTCON
**Offset:** 0x00
**Reset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | ON | | | | | | | |
| Access | HC | | | | | | | |
| Reset | 0 | | | | | | | |

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

**Bit 15 – ON** On bit
The ON bit reflects the status of the configuration fuse CFGCON2.DMTEN, if the fuse is set.

| Value | Description |
|---|---|
| 1 | Enables the Deadman Timer if the event configuration fuse is not enabled. |
| 0 | The DMT disabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.2. Deadman Timer Preclear

**Name:** DMTPRECLR
**Offset:** 0x10
**Reset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | STEP1[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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bits 15:8 – STEP1[7:0]** Pre-Clear Enable bit
This bit pre-clears the DMT when Write Pattern is set as following:

| Value | Description |
|---|---|
| 01000000 | Enables the Deadman Timer Pre-Clear (STEP 1). |
| All other write patterns | Sets DMTSTAT.BAD1 flag to '1'.<br>**Note:** Bits 15:8 are cleared when a DMT Reset event occurs. STEP1 is also cleared if DMTCLR.STEP2 is loaded with the correct value in the correct sequence. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.3. Deadman Timer Clear

**Name:** DMTCLR
**Offset:** 0x20
**Reset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | STEP2[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 – STEP2[7:0]** Clear Timer bit
This bit clears the DMT when Write Pattern is set as following:

| Value | Description |
|---|---|
| 00001000 | Clears DMTPRECLR.STEP1, DMTCLR.STEP2 and the Dead Man Timer if and only if preceded by the correct loading of Pre-Clear Enable (STEP1) in the correct sequence. The write to the DMTCLR.STEP2 field may be verified by reading DMTCNT and observing the counter being Reset. |
| all other write patterns | The DMTSTAT.BAD2 flag is set to '1', the value in the DMTPRECLR.STEP1 remains unchanged and the new value being written to DMTCLR.STEP2 is captured.<br>**Note:** These bits 7:0 are also cleared when a DMT Reset event occurs. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.4. Deadman Timer Status

**Name:** DMTSTAT
**Offset:** 0x30
**Reset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | BAD1 | BAD2 | DMT_EVENT | | | | | WINOPN |
| Access | R | R | R | | | | | R |
| Reset | 0 | 0 | 0 | | | | | 0 |

**Bit 7 – BAD1**
When an incorrect DMTPRECLR.STEP1 value is detected, this bit is set. It is cleared by a Reset.

**Bit 6 – BAD2**
When an incorrect value of DMTCLR.STEP2 is detected, this bit is set. It is cleared by a Reset.

**Bit 5 – DMT_EVENT**
This bit is set when the Deadman timer event is detected (counter expired or bad STEP1[7:0] or STEP2[7:0] value is entered prior to the counter increment). This bit remains set and is cleared only by a Reset.

**Bit 0 – WINOPN** Deadman Timer Clear Window bit.
A value of '1' indicates that a STEP2 clear action can take place, and if this clearing action occurs as part of a correct sequence of actions, the DMT counter will be cleared.


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.5. Deadman Timer Count

**Name:** DMTCNT
**Offset:** 0x40
**Reset:** 0x00
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | COUNTER[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNTER[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNTER[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNTER[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – COUNTER[31:0]** Read Current Contents of DMT Counter
This bit reads the available contents of DMT counter.


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.6. Deadman Timer Count Holding Register

**Name:** DMTHOLDREG
**Offset:** 0x50
**Reset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | UPRCNT[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 |
|---|---|---|---|---|---|---|---|---|
| | UPRCNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – UPRCNT[15:0]**

It is the content of DMTCNT.COUNTER[31:16] when the counter was last read to ensure a synchronous snapshot of the counter. This register is initialized to '0' on reset and is only loaded when the DMTCNT register is read.


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.7. Post Status Configure DMT Count Status Register

**Name:** DMTPSCNT
**Offset:** 0x60
**Reset:** 0x00
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PSCNT[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 |
|---|---|---|---|---|---|---|---|---|
| | PSCNT[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 |
|---|---|---|---|---|---|---|---|---|
| | PSCNT[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 |
|---|---|---|---|---|---|---|---|---|
| | PSCNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – PSCNT[31:0]** DMT Instruction Count Value Configuration Fuse Status bits
This bit always reflects the value of CFGCON2.DMTCNT.


© 2025 Microchip Technology Inc. and its subsidiaries

### 20.6.8. Post Status Configure DMT Interval Status Register

**Name:** DMTPSINTV
**Offset:** 0x70
**Reset:** 0x00
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | PSINTV[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 |
|---|---|---|---|---|---|---|---|---|
| | PSINTV[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 |
|---|---|---|---|---|---|---|---|---|
| | PSINTV[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 |
|---|---|---|---|---|---|---|---|---|
| | PSINTV[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – PSINTV[31:0]** DMT Window Interval Configuration Status bits.
This bit reflects the value of CFGCON2.DMTINTV.


© 2025 Microchip Technology Inc. and its subsidiaries

# 21. RAM Error Correction Code (RAMECC)

## 21.1. Overview

For safety applications, the PIC32CX-BZ6 family can embed error correction codes (ECC) to detect and correct single bit errors or to enable dual error detection in SRAM. As discussed in the Memories chapter, when the RAMECC is enabled, the top half of the SRAM memory will be reserved to store error correction codes and will not be available for the application. See *SRAM Memory Configuration* from Related Links.

ECC calculation is software-selectable through the CFGCON0.FRECCDIS bit in the Boot Flash Configuration. For additional information, see *System Configuration and Register Locking (CFG)* from Related Links.

**Related Links**

SRAM Memory Configuration

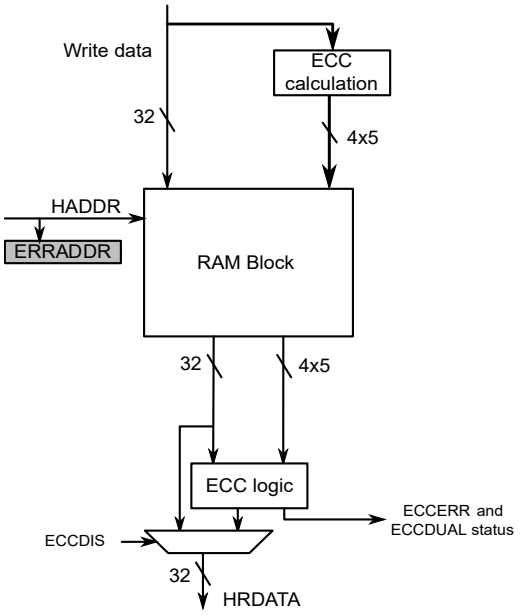
System Configuration and Register Locking (CFG)

## 21.2. Features

- Single Bit Correction and Dual Bit Detection
- Error Interrupt
- Operates Idle and Standby Sleep Mode
- Interrupts Generated by RAMECC Can be Used to Wake-up the Device from the Standby Sleep Mode

## 21.3. Block Diagram

**Figure 21-1.** RAMECC Block Diagram




## 21.4. Signal Description

Not applicable.


© 2025 Microchip Technology Inc. and its subsidiaries

## 21.5. Product Dependencies

The following sections describe how the other parts of the system must be configured correctly to use this peripheral.

### 21.5.1. I/O Lines

Not applicable.

### 21.5.2. Power Management

The RAMECC continues to operate in any Sleep mode (Standby Sleep, Idle) where the selected source clock is running. The RAMECC's interrupts can be used to wake up the device from sleep modes. See *Power Management Unit (PMU)* from Related Links for details on the different sleep modes.

**Related Links**

Power Management Unit (PMU)

### 21.5.3. DMA

Not applicable.

### 21.5.4. Interrupts

The interrupt request line is connected to the interrupt controller. Using the RAMECC interrupt(s) requires the interrupt controller to be configured first.

### 21.5.5. Events

Not applicable.

### 21.5.6. Debug Operation

When the CPU is halted in the Debug mode, the RAMECC corrects and logs ECC errors based on the following table.

**Table 21-1.** ECC Debug Operation

| DBGCTRL.ECCELOG | DBGCTRL.ECCDIS | Description |
|---|---|---|
| 0 | 0 | ECC errors from debugger reads are corrected but not logged in INTFLAG |
| 1 | 0 | ECC errors from debugger reads are corrected and logged in INTFLAG |
| X | 1 | ECC errors from debugger reads are not corrected or logged in INTFLAG |

If the RAMECC 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 are optionally write protected by the PAC, except the following registers:

- Interrupt Flag Status and Clear (INTFLAG) register
- Status (STATUS) register

Write protection is denoted by the Write-Protected property in the register description.

Write protection does not apply to accesses through an external debugger, see *Peripheral Access Controller (PAC)* from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**
Peripheral Access Controller (PAC)

## 21.6. Functional Description

### 21.6.1. Interrupts

The RAMECC has the following interrupt sources:

- Dual Bit Error (DUALE) – Indicates that a dual bit error was detected
- Single Bit Error (SINGLEE) – Indicates that a single-bit error was detected

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.

An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the ERRADDR register is read, the interrupt is disabled or the RAMECC is Reset.

All interrupt requests from the peripheral are OR'ed together at the 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.

### 21.6.2. Principle of Operation

ECC is implemented to detect and correct errors that may arise in the RAM arrays. The ECC logic is capable of double error detection and single error correction per 8-bit byte.

Upon single bit error detection, the Single Bit Error interrupt flag is raised (INTFLAG.SINGLEE). If a dual error is detected, the Dual Error interrupt flag (INTFLAG.DUALE) is raised. When the first error is detected, the ERRADDR register is frozen with the failing address and remains frozen until INTFLAG.DUALE and INTFLAG.SINGLEE are cleared. If a dual-bit error occurs while INTFLAG.SINGLEE is set, the ERRADDR register is updated with the dual bit error information and INTFLAG.DUALE is also set.

The INTFLAG.SINGLEE and INTFLAG.DUALE bits are both cleared on the ERRADDR read.

The block diagram illustrates the ECC interface. When ECC is disabled (CTRLA.ECCDIS=1), the ECC field in RAM is left unchanged on writes. On reads, ECC errors are not corrected or flagged.


© 2025 Microchip Technology Inc. and its subsidiaries

## 21.7. Register Summary

See the *RAMECC* module in the *Product Memory Mapping Overview* from Related Links for the base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | INTENCLR | 7:0 | | | | | | | DUALE | SINGLEE |
| 0x01 | INTENSET | 7:0 | | | | | | | DUALE | SINGLEE |
| 0x02 | INTFLAG | 7:0 | | | | | | | DUALE | SINGLEE |
| 0x03 | STATUS | 7:0 | | | | | | | | ECCDIS |
| 0x04 | ERRADDR | 7:0 | ERRADDR[7:0] | | | | | | | |
| | | 15:8 | ERRADDR[15:8] | | | | | | | |
| | | 23:16 | | | | | | | ERRADDR[17:16] | |
| | | 31:24 | | | | | | | | |
| 0x08 ... 0x0E | Reserved | | | | | | | | | |
| 0x0F | DBGCTRL | 7:0 | | | | | | | ECCELOG | ECCDIS |

**Related Links**

Product Memory Mapping Overview

## 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.

Some registers are optionally write protected by the PAC. Optional PAC write protection is denoted by the PAC Write-Protection property in each individual register description (see *Register Access Protection* from Related Links).

**Related Links**

Register Access Protection


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.1. Interrupt Enable Clear

**Name:** INTENCLR
**Offset:** 0x00
**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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DUALE | SINGLEE |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – DUALE** Dual Bit Error Interrupt Enable Clear
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Dual Bit Error Interrupt Enable bit, which disables the Dual Bit Error interrupt.

| Value | Description |
|---|---|
| 0 | The Dual Bit Error interrupt is disabled. |
| 1 | The Dual Bit Error interrupt is enabled. |

**Bit 0 – SINGLEE** Single Bit Error Interrupt Enable Clear
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Single Bit Error Interrupt Enable bit, which disables the Single Bit Error interrupt.

| Value | Description |
|---|---|
| 0 | The Single Bit Error interrupt is disabled. |
| 1 | The Single Bit Error interrupt is enabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.2. Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x01
**Reset:** 0x00
**Property:** Write-Protected

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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DUALE | SINGLEE |
| Access | | | | | | | R/W | R/W |
| Reset | | | | | | | 0 | 0 |

**Bit 1 – DUALE** Dual Bit Error Interrupt Enable Set
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will set the Dual Bit Error Interrupt Enable bit, which enables the Dual Bit Error interrupt.

| Value | Description |
|---|---|
| 0 | The Dual Bit Error interrupt is disabled. |
| 1 | The Dual Bit Error interrupt is enabled. |

**Bit 0 – SINGLEE** Single Bit Error Interrupt Enable Set
Writing a '0' to this bit has no effect.
Writing a '1' to this bit will clear the Single Bit Error Interrupt Enable bit, which disables the Single Bit Error interrupt.

| Value | Description |
|---|---|
| 0 | The Single Bit Error interrupt is disabled. |
| 1 | The Single Bit Error interrupt is enabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.3. Interrupt Flag Status and Clear

**Name:** INTFLAG
**Offset:** 0x02
**Reset:** 0x00

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

**Bit 1 – DUALE** Dual Bit ECC Error Interrupt
This flag is set on the occurrence of a dual bit ECC error. Writing a '0' to this bit has no effect. Reading the ECCADDR register will clear the Dual Bit Error interrupt flag.

| Value | Description |
|---|---|
| 0 | No dual bit errors have been received since the last clear. |
| 1 | At least one dual bit error has occurred since the last clear. |

**Bit 0 – SINGLEE** Single Bit ECC Error Interrupt
This flag is set on the occurrence of a single bit ECC error. Writing a '0' to this bit has no effect. Reading the ECCADDR register will clear the Single Bit Error interrupt flag.

| Value | Description |
|---|---|
| 0 | No errors have been received since the last clear. |
| 1 | At least one single bit error has occurred since the last clear. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.4. Status

**Name:** STATUS
**Offset:** 0x03
**Reset:** 0x00
**Property:** Read-only

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

**Bit 0 – ECCDIS** ECC Disable

This bit is fuse-updated at start-up based on the DEVCFG0/CFGCON0.FRECCDIS bit in the Boot Flash device configuration. When enabled, the calculated ECC is written to SRAM along with data. ECC correction and detection is enabled for reads.

| Value | Description |
|---|---|
| 0 | ECC detection and correction is enabled. |
| 1 | ECC detection and correction is disabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.5. Error Address

**Name:** ERRADDR
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** Read-only

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

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

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | ERRADDR[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 |
|---|---|---|---|---|---|---|---|---|
| | ERRADDR[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 17:0 – ERRADDR[17:0]** ECC Error Address

The RAM address offset from RAM start that caused an ECC error. If a single-bit error is followed by a dual-bit error, this register is updated with the address of the dual-bit error, otherwise, it stalls on the first error occurrence. This register reads as '0' unless INTFLAG.SINGLEE and/or INTFLAG.DUALE are '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 21.8.6. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0F
**Reset:** 0x00
**Property:** PAC Write-Protection

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

**Bit 1 – ECCELOG** ECC Error Log
When DBGCTRL.ECCDIS=0, this bit controls whether ECC errors are logged in the INTFLAG register.
When DBGCTRL.ECCDIS=1, this bit has no meaning.

| Value | Description |
|---|---|
| 0 | ECC errors for debugger reads are not logged. |
| 1 | ECC errors for debugger reads are logged if DBGCTRL.ECCDIS=0. |

**Bit 0 – ECCDIS** ECC Disable
By default, ECC errors during debugger reads are corrected and logged based on DBGCTRL.ECCELOG. Setting this bit disables ECC correction and logging.

| Value | Description |
|---|---|
| 0 | ECC errors are corrected for debugger reads and logged based on DBGCTRL.ECCELOG. |
| 1 | ECC errors are masked for debugger reads. |


© 2025 Microchip Technology Inc. and its subsidiaries

# 22. System Configuration and Register Locking (CFG)

## 22.1. Overview

A PIC32CX-BZ6 family device includes several non-volatile (programmable) configuration words that define the device's behavior. The device configuration words are located in the Boot Flash device config memory. These configuration words are loaded on equivalent system configuration registers after the device Reset. The write access to the system configuration registers is controlled through locking registers.

## 22.2. Features

This PIC32CX-BZ6 device provides several user-writable configuration registers related to the configuration and operation of the system.

- Permission Group Configuration Register (CFGPG) Defines the Permission Group
- System Key Register (SYSKEY) Defines the System Key
- Configuration Control Register 0 (CFGCON0(L)) Provides Control, Selection and Locking for Various Features of the Device
  **Note:** The registers marked with (L) are loadable from Flash.
  - PPS register locking
  - PMD register locking
  - CFGPG register locking
  - Config register locking
  - PMU controller register locking
  - Trace port enable, JTAG enable, SWO enable
  - Flash, SRAM ECC control
  - RTCC, AC alternate pinout selection
  - USB Functions
  - INT0 control
  - SMBus3 enable
- Configuration Control Register 1 (CFGCON1(L)) Provides Control, Selection and Locking for Various Features of the Device
  - QSPI DDR mode clock and address space cache enable
  - High-speed SERCOM, QSPI enable
  - WDT Sleep mode prescale configuration
  - CAN pins enable/disable
  - Debug port configuration and enable
  - USB DP DM rise/fall trim
  - AC1, AC0, CCL_OE output enable
  - MCLR configuration
- Configuration Control Register 2 (CFGCON2(L)) Provides Control, Selection and Locking for Various Features of the Device
  - DMT enable and configuration
  - WDT enable and configuration
  - Clock monitoring and control


© 2025 Microchip Technology Inc. and its subsidiaries

  - Oscillator enable and configuration
  - Two-speed start-up enabled in Sleep mode bit
- Configuration Control Register 4 (CFGCON4(L)) Provides Control, Selection and Locking for Various Features of the Device
  - Deep sleep modules control
  - SOSC configuration control
  - RTCC event control and clock select
- CFGI2C Register provides control for I2C slew rate control, I2C delay enable, and SMBus enable
- CFGCLKGENx Registers Provides Clock Selection for Peripherals
- User Unique ID Register (USERID(L)) Provides the End User with a 16-Bit ID Field that Can be Read Out Directly Through the SWD Interface Via the USERID SWD Instruction

## 22.3. Modes of Operation

### 22.3.1. System Configuration Words

The device configuration words programmed in Boot Flash memory (NVR pages) get loaded on equivalent registers after the device Reset. The following table shows the mapping between the Boot Flash memory region and loading registers. Registers marked with (L) are loadable from Flash, and they can be controlled by software after the boot with the correct unlock sequence. After programming the configuration words, the user must Reset the device to ensure the configuration data is reloaded with the new programmed values.

**Table 22-1.** Device Configuration in Flash vs Register

| Device Configuration (Flash) | Physical Address in Flash | Reloaded Register |
|---|---|---|
| FBCFG0 | 0x0081_0F9C | BCFG0 (0x4400_0200) |
| FBCFG1/DEVCFG0 | 0x0081_0F98 | CFGCON0(L) (0x4400_0000) |
| FBCFG2/DEVCFG1 | 0x0081_0F94 | CFGCON1(L) (0x4400_0010) |
| FBCFG3/DEVCFG2 | 0x0081_0F90 | CFGCON2(L) (0x4400_0020) |
| FBCFG4/DEVCFG4 | 0x0081_0F8C | CFGCON4(L) (0x4400_0040) |
| FBCFG5/FUSERID | 0x0081_0F88 | USERID(L) (0x4400_00C0) |

Other than device configurations in the Boot Flash region, there are some more system configuration registers. They are run-time programmable and do not have associated Flash region.

- CFGPGQOSx (x = 1, 2) - These registers defines the permission group settings for various bus hosts on the device bus matrix.
- CFGI2C - This register dictates I2C slew rate and delay control, SMBus enable.
- CFGPCLKGENx (x = 1, 2, 3, and 4) – These registers dictate the peripheral clock selection and enable the clock for the specific peripheral. See *Clock and Reset Unit (CRU)* from Related Links for more details.

**Related Links**

Clock and Reset Unit (CRU)

#### 22.3.1.1. System Configuration Register Protection

To ensure data integrity of each system configuration word, a comparison is continuously made between each configuration bit and its stored complement. If a mismatch is detected, a Configuration Mismatch Reset is generated causing a device Reset.

### 22.3.2. Alternate System Configuration Words

In the PIC32CX-BZ6 family of devices, the configuration words select various device configurations and are located at physical addresses from 0x00810F80 (FBCFG7).


© 2025 Microchip Technology Inc. and its subsidiaries

If an unrecoverable ECC error occurs when reading the configuration words, the alternate configuration words are used to configure the device from Boot Flash memory. This configuration can be identical to the primary configuration words or different to operate in another condition. The alternate configuration words are located at physical addresses from 0x00810E80 (ALTFBCFG7). To flag that an ECC error has occurred, the BCFGERR (RCON[27]) bit is set.

If uncorrectable ECC errors are found in both primary and alternate words, the BCFGFAIL (RCON[26]) bit is set and the default configuration is used.

After programming the configuration words, the user application must reset the device to ensure the configuration data is reloaded with the new programmed values.

## 22.4. Locking and Unlocking the System Configuration Registers

Write access to the system configuration registers is controlled via the CFGCON0.CFGLOCK[1:0] register bits.

## 22.5. NMI Events

The only system configuration that gets reset on an NMI event are CPUPG bits in the CFGPGQOS register. This allows application firmware to pass control back to the bootloader and re-enable reads of all configuration words from Boot Flash NVR pages if reads of the Boot Flash pages were disabled using group permissions.

## 22.6. Register Locking

Several modules contain registers that are protected from errant code causing unwanted changes by the system lock feature. When the system lock is in effect, system lock-protected registers are not writable. The system lock feature protects registers that are system-critical. The PIC32CX-BZ6 provides different methods of register level locking:

### 22.6.1. One-Way Lock

This mechanism provides a one-way lock (when locked, only a Reset can unlock) using the CFGCON0.IOLOCK, CFGCON0.PMDLOCK, CFGCON0.PMULOCK and CFGCON0.PGLOCK register bits. This method includes protection for the following registers:

- All PPS registers using CFGCON0.IOLOCK
- All PMD registers using CFGCON0.PMDLOCK
- CFGPGQOS register using CFGCON0.PGLOCK
- CFGPCLKGENx register using CFGCON0.PGLOCK
- All PMU registers using CFGCON0.PMULOCK

### 22.6.2. One-Way or Two-Way Lock (Software Selectable)

If CFGCON0.CFGLOCK[1:0] is '10', it locks the registers, but it is also possible to unlock the registers by writing '00' to CFGCON0.CFGLOCK[1:0]. If CFGCON0.CFGLOCK[1:0] is '11', it locks the registers, and they are no longer possible to unlock as CFGCON0.CFGLOCK[1:0] bits are also locked. Only a system Reset can unlock them. This mechanism provides one-way or two-way lock (software selectable) using the CFGCON0.CFGLOCK[1:0] register bits. This method applies to the following registers:

- BCFG0
- CFGCON0
- CFGCON1
- CFGCON2
- CFGCON4
- USER_ID


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.6.3. Two-Way Lock and Unlock

Each module that uses the system lock feature describes register bits and functions, which are affected by this system lock feature. A specific sequence of writes to the SYSKEY register unlock the access to those register bits and features.

This locking method provides a two-way (locking and unlocking) write lock of system-critical registers. It includes protection for the following registers:

- CRU.OSCCON
- CRU.OSCTRM
- CRU.SPLLCON
- CRU.RSWRST
- CRU.RNMICON
- CRU.PB1DIV
- CRU.PB2DIV
- CRU.PB3DIV
- CRU.PB4DIV
- CRU.SLEWCON
- CRU.CLK_DIAG

#### 22.6.3.1. Unlock Requirements

The unlock sequence must be atomic. If any other peripheral bus access occurs on the same peripheral bus where SYSKEY resides during the unlock attempt sequence, the unlock fails. Therefore, turn OFF all bus initiators, such as DMA, and disable interrupts.

#### 22.6.3.2. Unlock Sequence

The following steps to be followed to unlock the CFG registers. The unlock sequencer looks for steps 3 and 4 to be atomic. For this sequence, atomic means that there is no other activity on the peripheral bus between steps 3 and 4. Step 2 is only needed to ensure that the sequence starts from a known locked state.

1. Suspend all other peripheral bus accesses.
2. Write SYSKEY = 0x00000000
3. Write SYSKEY = 0xAA996655
4. Write SYSKEY = 0x556699AA

#### 22.6.3.3. Lock Sequence

When the system is unlocked, any write to the SYSKEY register causes the system lock to become active.

#### 22.6.3.4. Lock/Unlock Indication

The SYSKEY register read value indicates the status of the unlock sequence. A value of 0x00000000 indicates the system is still locked. A value of 0x00000001 indicates the sequence succeeded and the system is unlocked.

## 22.7. Effects of Various Resets

The configuration data is reloaded from the corresponding Boot Flash memory configuration words on the following types of Reset:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- External Reset (NMCLR)


© 2025 Microchip Technology Inc. and its subsidiaries

- Configuration Mismatch Reset (CM)
- Watchdog Timer Reset (WDTR)
- Software Reset (SWR)
- NMI Time-out Reset (NMITR)


© 2025 Microchip Technology Inc. and its subsidiaries

## 22.8. Register Summary

See *CFG* module in the *Product Memory Mapping Overview* from Related Links for the base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CFGCON0(L) | 7:0 | CPENFILT | | | ADCPOVR | JTAGEN | TROEN | SWOEN | |
| | | 15:8 | CFGLOCK[1:0] | | IOLOCK | PMDLOCK | PGLOCK | PMULOCK | RTCOUT_ALTEN | RTCIN0_ALTEN |
| | | 23:16 | SMBUS3EN2 | VBUSONIO | USBIDIO | HPLUGDIS | UPLLHWMD | EPLLHWMD | USBSSEN | |
| | | 31:24 | | FRECCDIS | ECCCTL[1:0] | | CMEN | INT0P | INT0E | PCM |
| 0x04 ... 0x0F | Reserved | | | | | | | | | |
| 0x10 | CFGCON1(L) | 7:0 | ZBTWKSYS | CCL_OE | TRCEN | | | CAN0_PIN_EN | | |
| | | 15:8 | QSCHE_EN | SMCLR | SERCOM_HSEN[2:0] | | | QSPI_HSEN | CMP1_OE | CMP0_OE |
| | | 23:16 | USBDPRTRIM[1:0] | | USBDPFTRIM[1:0] | | USBDMRTRIM[1:0] | | USBDMFTRIM[1:0] | |
| | | 31:24 | CAN1_PIN_EN | CLKZBREF | QSPIDDRM | WDTPSS[4:0] | | | | |
| 0x14 ... 0x1F | Reserved | | | | | | | | | |
| 0x20 | CFGCON2(L) | 7:0 | PMUTEST_VDD_EN | | DMTINTV[2:0] | | | ACMP_CYCLE[2:0] | | |
| | | 15:8 | FSCMEN | CKSWEN | WAKE2SPD | SOSCSEL | WDTRMCS[1:0] | | POSCMD[1:0] | |
| | | 23:16 | WDTEN | WINDIS | WDTSPGM | WDTPSR[4:0] | | | | |
| | | 31:24 | DMTEN | DMTCNT[4:0] | | | | | WINSZ[1:0] | |
| 0x24 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | CFGCON4(L) | 7:0 | SOSC_CFG[7:0] | | | | | | | |
| | | 15:8 | MLPCLK_MOD | VBKP_DIVSEL | VBKP_32KCSEL[1:0] | | VBKP_1KCSEL | RTCEVENT_EN | RTCEVENTSEL[1:0] | |
| | | 23:16 | DSWDTPS[2:0] | | | DSZPBOREN | CPEN_DLY[2:0] | | | RTCEVTYPE |
| | | 31:24 | RTCNTM_CSEL | LPOSCEN | | DSBITEN | DSWDTEN | DSWDTLPRC | DSWDTPS[4:3] | |
| 0x44 ... 0x4F | Reserved | | | | | | | | | |
| 0x50 | CFGPGQOS1 | 7:0 | | | | | CPUQOS[1:0] | | CPUPG[1:0] | |
| | | 15:8 | | | | | | | DMAPG[1:0] | |
| | | 23:16 | CRYPTOQOS[1:0] | | CRYPTOPG[1:0] | | | | | |
| | | 31:24 | WISIBQOS[1:0] | | FCQOS[1:0] | | | | DSUPG[1:0] | |
| 0x54 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | CFGPGQOS2 | 7:0 | USBQOS[1:0] | | USBPG[1:0] | | ADCQOS[1:0] | | ADCPG[1:0] | |
| | | 15:8 | CAN1QOS[1:0] | | CAN1PG[1:0] | | CAN0QOS[1:0] | | CAN0PG[1:0] | |
| | | 23:16 | | | | | ETHQOS[1:0] | | ETHPG[1:0] | |
| | | 31:24 | | | | | | | | |
| 0x64 ... 0x6F | Reserved | | | | | | | | | |
| 0x70 | CFGI2C | 7:0 | | SMBUSEN6 | | | | SMBUSEN2 | SMBUSEN1 | SMBUSEN0 |
| | | 15:8 | | I2CDSEL6 | | | | I2CDSEL2 | I2CDSEL1 | I2CDSEL0 |
| | | 23:16 | | SLRCTRL6 | | | | SLRCTRL2 | SLRCTRL1 | SLRCTRL0 |
| | | 31:24 | | SLRTEN6 | | | | SLRTEN2 | SLRTEN1 | SLRTEN0 |
| 0x74 ... 0x7F | Reserved | | | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x80 | CFGPCLKGEN1 | 7:0 | FREQMRCD | FREQMRCSEL[2:0] | | | EICCD | EICCSEL[2:0] | | |
| | | 15:8 | SERCOM012CD | SERCOM012CSEL[2:0] | | | FREQMMCD | FREQMMCSEL[2:0] | | |
| | | 23:16 | TCC12CD | TCC12CSEL[2:0] | | | SERCOM6CD | SERCOM6CSEL[2:0] | | |
| | | 31:24 | CM4TCD | CM4TCSEL[2:0] | | | SERCOM345CD | SERCOM345CSEL[2:0] | | |
| 0x84 ... 0x8F | Reserved | | | | | | | | | |
| 0x90 | CFGPCLKGEN2 | 7:0 | EVSYSC2CD | EVSYSC2SEL[2:0] | | | EVSYSC1CD | EVSYSC1SEL[2:0] | | |
| | | 15:8 | EVSYSC4CD | EVSYSC4SEL[2:0] | | | EVSYSC3CD | EVSYSC3SEL[2:0] | | |
| | | 23:16 | EVSYSC6CD | EVSYSC6SEL[2:0] | | | EVSYSC5CD | EVSYSC5SEL[2:0] | | |
| | | 31:24 | EVSYSC8CD | EVSYSC8SEL[2:0] | | | EVSYSC7CD | EVSYSC7SEL[2:0] | | |
| 0x94 ... 0x9F | Reserved | | | | | | | | | |
| 0xA0 | CFGPCLKGEN3 | 7:0 | EVSYSC10CD | EVSYSC10SEL[2:0] | | | EVSYSC9CD | EVSYSC9SEL[2:0] | | |
| | | 15:8 | EVSYSC12CD | EVSYSC12SEL[2:0] | | | EVSYSC11CD | EVSYSC11SEL[2:0] | | |
| | | 23:16 | TCC0CD | TCC0CSEL[2:0] | | | ACCD | ACCSEL[2:0] | | |
| | | 31:24 | CVDCD | CVDCSEL[2:0] | | | ETHD | ETHCLKSEL[2:0] | | |
| 0xA4 ... 0xAF | Reserved | | | | | | | | | |
| 0xB0 | CFGPCLKGEN4 | 7:0 | TC1CD | TC1CSEL[2:0] | | | TC0CD | TC0CSEL[2:0] | | |
| | | 15:8 | TC45CD | TC45CSEL[2:0] | | | TC23CD | TC23CSEL[2:0] | | |
| | | 23:16 | TC89CD | TC89CSEL[2:0] | | | TC67CD | TC67CSEL[2:0] | | |
| | | 31:24 | CAN1CD | CAN1CSEL[2:0] | | | CAN0CD | CAN0SEL[2:0] | | |
| 0xB4 ... 0xBF | Reserved | | | | | | | | | |
| 0xC0 | USER_ID | 7:0 | USER_ID[7:0] | | | | | | | |
| | | 15:8 | USER_ID[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xC4 ... 0xCF | Reserved | | | | | | | | | |
| 0xD0 | SYSKEY | 7:0 | SYSKEY[7:0] | | | | | | | |
| | | 15:8 | SYSKEY[15:8] | | | | | | | |
| | | 23:16 | SYSKEY[23:16] | | | | | | | |
| | | 31:24 | SYSKEY[31:24] | | | | | | | |
| 0xD4 ... 0x01FF | Reserved | | | | | | | | | |
| 0x0200 | BCFG0 | 7:0 | | | | | | | PCSCMODE | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | BINFOVALID0 | | SIGN | CP | | | | |

**Related Links**

CLR, SET and INV Registers

Product Memory Mapping Overview

## 22.9. 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.


© 2025 Microchip Technology Inc. and its subsidiaries

Some registers are optionally write protected by the PAC. Optional PAC write protection is denoted by the PAC Write-Protection property in each individual register description.

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.

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.

The following are the list of conventions available in the register description:

- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as '0'
- -n = Value at POR
- 1 = Bit is set
- 0 = Bit is cleared
- x = Bit is unknown
- HS = Hardware Set
- HC = Hardware Cleared


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.1. Configuration Control Register 0

**Name:** CFGCON0(L)
**Offset:** 0x00
**Reset:** 0x7960000b
**Property:** -

The CFGLOCK[1:0] register bits are writable only when CFGLOCK[0] = 0.

The IOLOCK, PMDLOCK and PGLOCK register bits can only be cleared on a system Reset. Thereafter, these bits are writable using CFGLOCK.

This register is loaded with trusted data from FBCFG1/DEVCFG0 during the pre-boot period. Trusted data from Flash means that when there is no BCFG* fail status and BINFOVALID = 0 during Flash, the configuration word reads. If accompanied by the fail status BCFGFAIL (RCON[26]) or blank/erase indication, Reset values (described in the following register description) are retained and new values from FBCFG1 are not loaded.

Under all conditions, Flash loading is omitted for the following bits in the CFGCON0 register:

- IOLOCK
- CFGLOCK[1:0]
- PMDLOC
- PGLOCK
- PMULOCK
- JTAGEN
- HPLUGDIS

Hence, writing these bits in Boot Flash does not have an effect on the configuration register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | FRECCDIS | ECCCTL[1:0] | | CMEN | INT0P | INT0E | PCM |
| Access | | R/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | 1 | 1 | 1 | 1 | 0 | 0 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | SMBUS3EN2 | VBUSONIO | USBIDIO | HPLUGDIS | UPLLHWMD | EPLLHWMD | USBSSEN | |
| Access | R/W/L | R/W/L | R/W/L | R/W | R/W/L | R/W/L | R/W/L | |
| Reset | 0 | 1 | 1 | 0 | 0 | 0 | 0 | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CFGLOCK[1:0] | | IOLOCK | PMDLOCK | PGLOCK | PMULOCK | RTCOUT_ALTEN | RTCIN0_ALTEN |
| Access | R/W/L | R/W/L | R/S/L | R/S/L | R/S/L | R/S/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CPENFILT | | | ADCPOVR | JTAGEN | TROEN | SWOEN | |
| Access | R/W/L | | | R/W/L | R/W/L | R/W/L | R/W/L | |
| Reset | 0 | | | 0 | 1 | 0 | 1 | |

**Bit 30 – FRECCDIS** Flex RAM (SRAM) ECC Control
**Note:** Only a read-only fuse bit sets the initialization value of RAMECC Control. The true RAMECC override is available in the RAMECC module.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | ECC is disabled |
| 0 | ECC is enabled |

**Bits 29:28 – ECCCTL[1:0]** Flash ECC Control

**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 11 | ECC and dynamically ECC are disabled |
| 10 | ECC and dynamically ECC are disabled |
| 01 | Dynamically ECC is enabled |
| 00 | ECC is enabled (NVMCON.NVMOP[3:0] != 1 (Word programming)) |

**Bit 27 – CMEN** SPA/DPA Counter measures Enable

| Value | Description |
|---|---|
| 1 | SPA/DPA Counter measure is enabled |
| 0 | SPA/DPA Counter measure is disabled |

**Bit 26 – INT0P** INT0 Polarity

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | INT0 Polarity (High) |
| 0 | INT0 Polarity (Low) |

**Bit 25 – INT0E** INT0 Enable

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | INT0 is enabled |
| 0 | INT0 is disabled |

**Bit 24 – PCM** PCHE I/D Cacheable Mode

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Always enabled from outside. Can be further enabled/disabled by PCHE SFR registers. |
| 0 | The cache-ability is controlled by the CPU via HPROT[3] of ARM protection control bus. |

**Bit 23 – SMBUS3EN2** SMBus3 Enable for SERCOM2

| I2C (TWI) Enabled? | SMBUS3EN | SMBUSEN | PAD0/PAD1 Interface |
|---|---|---|---|
| No | 0/1 | 0/1 | ST |
| Yes | 0 | 0 | I2C |
| Yes | 0 | 1 | SMBus |
| Yes | 1 | 0/1 | SMBus3 |

**Notes:**

- This bit does not enable functional SMBus3 features, only SMBus3 tolerant pad functionality. It is intended to provide a mechanism for implementing a 1.8V I2C or SMBus interface using external 1.8V pull-ups.
- This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | SMBus3 tolerant pads are enabled |
| 0 | SMBus3 tolerant pads are disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 22 – VBUSONIO** USB VBUS_ON Selection bit
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | The USB module controls the VBUS_ON pin |
| 0 | The Port function controls the VBUS_ON pin |

**Bit 21 – USBIDIO** USB USBID Selection bit
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | The USB module controls the USBID pin |
| 0 | The Port function controls the USBID pin |

**Bit 20 – HPLUGDIS** Hot Plugging Disable (outside fuse loading)
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Hot plugging is disabled |
| 0 | Hot plugging is enabled |

**Bit 19 – UPLLHWMD** UWPLL HW Control Mode for PWDN and RESET
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | PWDN and RESET are generated by hardware |
| 0 | PWDN and RESET are generated by software using corresponding PLLCON register bits |

**Bit 18 – EPLLHWMD** EWPLL HW Control Mode for PWDN and RESET
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | PWDN and RESET are generated by hardware |
| 0 | PWDN and RESET are generated by software using corresponding PLLCON register bits |

**Bit 17 – USBSSEN** USB Suspend Sleep Enable
Enables features for USB PHY FREF clock shutdown in the SUSPEND mode.
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | USB FREF clock is shut down when the Suspend mode is active |
| 0 | USB FREF clock continues to run when the Suspend mode is active |

**Bits 15:14 – CFGLOCK[1:0]** Configuration Register Lock
**Note:** These bits are only writable when CFGLOCK[1:0] is '00' or '10'.

| Value | Description |
|---|---|
| 11 | All NVR memory self-writes, Boot Configuration (BCFG0) and System Configuration registers (CFG* and USER_ID) are locked and cannot be written. CFGLOCK value cannot be changed. |
| 10 | All NVR memory self-writes, Boot Configuration (BCFG0) and System Configuration registers (CFG* and USER_ID) are locked and cannot be written. CFGLOCK value can be changed. |
| 01 | Reserved for future use |
| 00 | All NVR memory self-writes, Boot Configuration (BCFG0) and System Configuration registers (CFG* and USER_ID) are not locked and can be written. CFGLOCK value can be changed. |

**Bit 13 – IOLOCK** I/O Lock
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | I/O Remap SFR bits are locked and cannot be modified |
| 0 | I/O Remap SFR bits are not locked and can be modified |

**Bit 12 – PMDLOCK** Peripheral Module Disable (PMD) Lock
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | PMDx SFR bits are locked and cannot be modified |
| 0 | PMDx SFR bits are not locked and can be modified |

**Bit 11 – PGLOCK** Permission Group Lock
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | CFGPG SFR bits are locked and cannot be modified |
| 0 | CFGPG SFR bits are not locked and can be modified |

**Bit 10 – PMULOCK** PMU Controller Register Lock
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | PMU* SFR bits are locked and cannot be modified |
| 0 | PMU* SFR bits are not locked and can be modified |

**Bit 9 – RTCOUT_ALTEN** RTCOUT Alternate Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | RTC/OUT is available on PA10 |
| 0 | RTC/OUT is available on PA4 |

**Bit 8 – RTCIN0_ALTEN** RTCIN0 Alternate Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | RTC_IN0 is available on PA9 |
| 0 | RTC_IN0 is available on PA3 |

**Bit 7 – CPENFILT** ADC CP Filter Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | ADC CP filter is enabled |
| 0 | ADC CP filter is disabled |

**Bit 4 – ADCPOVR** ADC Charge Pump Override
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Overridden (Software controlled) |
| 0 | Hardware controlled |

**Bit 3 – JTAGEN** JTAG Enable
**Note:** JTAG functionality is not available in the PIC32CX-BZ6 devices. The default value of this bit is '1'. It is recommended to write '0' to this bit during Application initialization to use JTAG pins for regular GPIO functionality. For pin details, see *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 2 – TROEN** Trace Output Enable

**Notes:**

- When CFGCON1.TRCEN = 0, the value of this bit is ignored but has the effect of being '0'.
- This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Start Trace Clock and enable Trace Outputs (Trace probe must be present) |
| 0 | Stop Trace Clock and disable Trace Outputs |

**Bit 1 – SWOEN** SWO Enable on two-wire Debug interface

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | SWO is enabled |
| 0 | SWO is disabled |

**Related Links**

I/O Ports and Peripheral Pin Select (PPS)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.2. Configuration Control Register 1

**Name:** CFGCON1(L)
**Offset:** 0x10
**Reset:** 0x1f00403B
**Property:** -

This register is loaded with trusted data from FBCFG2/DEVCFG1 during the pre-boot period.

Trusted data from Flash means that when there is no BCFG* fail status during Flash, configuration word reads. If accompanied by fail status BCFGFAIL (RCON[26]) or blank/erase indication, Reset values (described in the following register description) are retained, and new values from FBCFG2 are not loaded.

Under all conditions, Flash loading is omitted for the ZBTWKSYS bit in CFGCON1 register. Hence, writing this bit in Boot Flash will not have an effect on the configuration register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CAN1_PIN_EN | CLKZBREF | QSPIDDRM | WDTPSS[4:0] | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | USBDPRTRIM[1:0] | | USBDPFTRIM[1:0] | | USBDMRTRIM[1:0] | | USBDMFTRIM[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | QSCHE_EN | SMCLR | SERCOM_HSEN[2:0] | | | QSPI_HSEN | CMP1_OE | CMP0_OE |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ZBTWKSYS | CCL_OE | TRCEN | | | CAN0_PIN_EN | | |
| Access | R/W/L | R/W/L | R/W/L | | | R/W/L | | |
| Reset | 0 | 0 | 1 | | | 0 | | |

**Bit 31 – CAN1_PIN_EN** CAN1 Pin Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | CAN1 pins are enabled |
| 0 | CAN1 pins are disabled |

**Bit 30 – CLKZBREF** External Reference Clock ZB Enable
The external reference clock output from the 802.15.4 wireless subsystem on the REFO1 pin, which is configurable through PPS
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enable clk_zb_to_ext on PPS.REFO1 |
| 0 | No clk_zb_to_ext on PPS.REFO1, PPS.REFO1 remains unchanged |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 29 – QSPIDDRM** QSPI Double Data Rate (DDR) Mode Clock Enable

**Notes:**

- When using the QSPI DDR mode, the System Clock (SYS_CLK) must be <= 32 MHz.
- This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | QSPI DDR mode clock is enabled |
| 0 | QSPI DDR mode clock is disabled |

**Bits 28:24 – WDTPSS[4:0]** Watchdog Timer Post-scale Select Sleep bits

**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 10100 | 1:1048576 |
| 10011 | 1:524288 |
| 10010 | 1:262144 |
| 10001 | 1:131072 |
| 10000 | 1:65536 |
| 01111 | 1:32768 |
| 01110 | 1:16384 |
| 01101 | 1:8192 |
| 01100 | 1:4096 |
| 01011 | 1:2048 |
| 01010 | 1:1024 |
| 01001 | 1:512 |
| 01000 | 1:256 |
| 00111 | 1:128 |
| 00110 | 1:64 |
| 00101 | 1:32 |
| 00100 | 1:16 |
| 00011 | 1:8 |
| 00010 | 1:4 |
| 00001 | 1:2 |
| 00000 | 1:1 |

**Bits 23:22 – USBDPRTRIM[1:0]** USB DP Rise Trim fuse bits

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

**Bits 21:20 – USBDPFTRIM[1:0]** USB DP Fall Trim fuse bits

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

**Bits 19:18 – USBDMRTRIM[1:0]** USB DM Rise Trim fuse bits

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

**Bits 17:16 – USBDMFTRIM[1:0]** USB DM Fall Trim fuse bits

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

**Bit 15 – QSCHE_EN** QSPI Address Space Cache Attribute

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Cache attribute is enabled |
| 0 | Caching is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 14 – SMCLR** Selects CRU handling of NMCLR Control
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Does not reset all device NMCLR Reset states |
| 0 | NMCLR external Reset causes a faux POR |

**Bits 13:11 – SERCOM_HSEN[2:0]** SERCOMx (Direct) Enable, Bit 13=SERCOM2, Bit 12=SERCOM1, Bit 11=SERCOM0
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enables SERCOMx via Direct Mode (High-Speed) |
| 0 | Enables SERCOMx via PPS |

**Bit 10 – QSPI_HSEN** QSPI (Direct) Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enables QSPI via Direct Mode (High-Speed) |
| 0 | Enables QSPI via PPS |

**Bit 9 – CMP1_OE** Analog Comparator-1 Output Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | AC1 output is enabled |
| 0 | AC1 output is disabled |

**Bit 8 – CMP0_OE** Analog Comparator-0 Output Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | AC0 output is enabled |
| 0 | AC0 output is disabled |

**Bit 7 – ZBTWKSYS** ZBT Subsystem External Wake-up source
**Notes:**

- Write-only bit, with read-as zero; when '1' is written, creates one pulse on the ZBT subsystem.external_NMI0 pin. This enables external system wake-up to ZBT subsystem. This allows CPU and ZBT subsystem wake-up/sleep to be independent of each other.
- Flash fuse loading is excluded for this bit.

**Bit 6 – CCL_OE** CCL Pads (via PPS) Output Enable
**Note:** This bit is only writable when CFGLOCK[1:0] = 00.

| Value | Description |
|---|---|
| 1 | CCL pads (via PPS) output is enabled |
| 0 | CCL pads (via PPS) output is disabled |

**Bit 5 – TRCEN** Trace Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Trace features in the CPU are enabled |
| 0 | Trace features in the CPU are disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 2 – CAN0_PIN_EN** CAN0 Pin Enable

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | CAN0 pins are enabled |
| 0 | CAN0 pins are disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.3. Configuration Control Register 2

**Name:** CFGCON2(L)
**Offset:** 0x20
**Reset:** 0x7f7fff38
**Property:** -

This register is loaded with trusted data from FBCFG3/DEVCFG2 during the pre-boot period.

Trusted data from Flash means that when there is no BCFG* fail status during Flash, configuration word reads. If accompanied by fail status BCFGFAIL (RCON[26]) or blank/erase indication, Reset values (described in the following register description) are retained and new values from FBCFG3 are not loaded.

Under all conditions, Flash loading is omitted for POSCMD[1:0] bits in CFGCON2 register. Hence, writing these bits in Boot Flash must not have an effect on the configuration register.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | DMTEN | DMTCNT[4:0] | | | | | WINSZ[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | WDTEN | WINDIS | WDTSPGM | WDTPSR[4:0] | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FSCMEN | CKSWEN | WAKE2SPD | SOSCSEL | WDTRMCS[1:0] | | POSCMD[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PMUTEST_VDD_EN | | DMTINTV[2:0] | | | ACMP_CYCLE[2:0] | | |
| Access | R/W/L | | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | | 1 | 1 | 1 | 0 | 0 | 0 |

**Bit 31 – DMTEN** Dead Man Timer Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | DMT is enabled always and DMTCON.ON bit does not have control |
| 0 | DMT disabled (control is placed on the DMTCON.ON bit) |

**Bits 30:26 – DMTCNT[4:0]** Dead Man Timer Count Select
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 00000 | Counter value is 2^8 for DMTPSCNT[31:0] |
| 00001 | Counter value is 2^9 for DMTPSCNT[31:0] |
| . . . | ... |
| 10100 | Counter value is 2^28 for DMTPSCNT[31:0] |
| 10101 | Counter value is 2^29 for DMTPSCNT[31:0] |
| 10110 | Counter value is 2^30 for DMTPSCNT[31:0] |
| 10111 | Counter value is 2^31 for DMTPSCNT[31:0] |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 11000 – 11111 | Reserved |

**Bits 25:24 – WINSZ[1:0]** Watchdog Timer Window Size
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 00 | Window size is 75% |
| 01 | Window size is 50% |
| 10 | Window size is 37.5% |
| 11 | Window size is 25% |

**Bit 23 – WDTEN** Watchdog Timer Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | WDT is enabled always and WDTCON.ON bit does not have control |
| 0 | WDT is disabled (control is placed on the WDTCON.ON bit) |

**Bit 22 – WINDIS** Windowed Watchdog Timer Disable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Standard WDT selected; windowed WDT disabled |
| 0 | Windowed WDT enabled |

**Bit 21 – WDTSPGM** Watchdog Timer Stop during Flash Programming
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | The WDT stops during NVR programming |
| 0 | The WDT runs during NVR programming |

**Bits 20:16 – WDTPSR[4:0]** Watchdog Timer Post-scale Select Run bits
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 10100 | 1:1048576 |
| 10011 | 1:524288 |
| 10010 | 1:262144 |
| 10001 | 1:131072 |
| 10000 | 1:65536 |
| 01111 | 1:32768 |
| 01110 | 1:16384 |
| 01101 | 1:8192 |
| 01100 | 1:4096 |
| 01011 | 1:2048 |
| 01010 | 1:1024 |
| 01001 | 1:512 |
| 01000 | 1:256 |
| 00111 | 1:128 |
| 00110 | 1:64 |
| 00101 | 1:32 |
| 00100 | 1:16 |
| 00011 | 1:8 |
| 00010 | 1:4 |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 00001 | 1:2 |
| 00000 | 1:1 |

**Bit 15 – FSCMEN** Fail-Safe Clock Monitor Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | FSCM enabled |
| 0 | FSCM disabled |

**Bit 14 – CKSWEN** Software Clock Switching Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Software clock switching enabled |
| 0 | Software clock switching disabled |

**Bit 13 – WAKE2SPD** Two-Speed Start-up Enabled in the Sleep mode
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | When the device EXITS Sleep mode, the SYS_CLK is from FRC until the selected clock is ready. |
| 0 | Not applicable. |

**Bit 12 – SOSCSEL** SOSC Selection Configuration
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Crystal (SOSCI/SOSCO) mode |
| 0 | Digital (SCLKI) mode |

**Bits 11:10 – WDTRMCS[1:0]** WDT RUN Mode Clock Select
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 11 | LPRC |
| 10 | Reserved |
| 01 | Reserved |
| 00 | WDT PB Clock (PB1_CLK) |

**Bits 9:8 – POSCMD[1:0]** Primary Oscillator Configuration
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 11 | Primary Oscillator mode is disabled |
| 10 | Reserved |
| 01 | Reserved |
| 00 | Primary Oscillator mode is selected |

**Bit 7 – PMUTEST_VDD_EN** PMU Test Output or VDD/2 Enable via ADC IE12
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | PMU Test output monitor is enabled |
| 0 | VDD/2 Monitor is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 5:3 – DMTINTV[2:0]** Dead Man Timer Count Window Interval
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 000 | Window/Interval value is zero for DMTPSINTV[31:0] - Windowed mode is disabled |
| 001 | Window/Interval value is 1/2 Counter value for DMTPSINTV[31:0] |
| 010 | Window/Interval value is 3/4 Counter value for DMTPSINTV[31:0] |
| 011 | Window/Interval value is 7/8 Counter value for DMTPSINTV[31:0] |
| 100 | Window/Interval value is 15/16 Counter value for DMTPSINTV[31:0] |
| 101 | Window/Interval value is 31/32 Counter value for DMTPSINTV[31:0] |
| 110 | Window/Interval value is 63/64 Counter value for DMTPSINTV[31:0] |
| 111 | Window/Interval value is 127/128 Counter value for DMTPSINTV[31:0] |

**Bits 2:0 – ACMP_CYCLE[2:0]** AC Comparator Result Wait Cycles
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| n | Wait for 32 µs* ACMP_CYCLE[2:0]+1 cycles to generate comparator done indication |


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.4. Configuration Control Register 4

**Name:** CFGCON4(L)
**Offset:** 0x40
**Reset:** 0x840e4000
**Property:** -

This register is loaded with trusted data from FBCFG4/DEVCFG4 during the pre-boot period.

Trusted data from Flash means that when there is no BCFG* fail status during Flash, configuration word reads. If accompanied by fail status BCFGFAIL (RCON[26]) or blank/erase indication, Reset values (described in the following register description) are retained and new values from FBCFG4 are not loaded.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | RTCNTM_CSEL | LPOSCEN | | DSBITEN | DSWDTEN | DSWDTLPRC | DSWDTPS[4:3] | |
| Access | R/W/L | R/W/L | | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 1 | 0 | | 0 | 0 | 1 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | DSWDTPS[2:0] | | | DSZPBOREN | CPEN_DLY[2:0] | | | RTCEVTYPE |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | MLPCLK_MOD | VBKP_DIVSEL | VBKP_32KCSEL[1:0] | | VBKP_1KCSEL | RTCEVENT_EN | RTCEVENTSEL[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

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

**Bit 31 – RTCNTM_CSEL** RTCC Counter Mode Clock Select
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Raw 32 KHz clock |
| 0 | Processed 32 KHz clock |

**Bit 30 – LPOSCEN** Low Power (Secondary) Oscillator Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enable Low Power (Secondary) Oscillator, also at Reset |
| 0 | Disable Low Power (Secondary) Oscillator |

**Bit 28 – DSBITEN** Deep Sleep Bit Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

**Note:** When waking up from Deep Sleep Modes, while DSCON_DSSR is set, the DSBITEN should not be changed by software, doing so may trigger a spurious Configuration Mismatch fault.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Enable DS bit in DSCON |
| 0 | Disable DS bit in DSCON |

**Bit 27 – DSWDTEN** Deep Sleep Watchdog Timer Enable

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enable DSWDT during deep sleep |
| 0 | Disable DSWDT during deep sleep |

**Bit 26 – DSWDTLPRC** Deep Sleep Watchdog Timer Reference Clock Select

**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Select LPRC as DSWDT reference clock |
| 0 | Select SOSC as DSWDT reference clock |

**Bits 25:21 – DSWDTPS[4:0]** Deep Sleep Watchdog Timer Postscale Select

The DS WDT prescaler is 32; this creates an approximate base time unit of 1 ms.

**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 11111 | $1:2^{36}$ (25.7 days) |
| 11110 | $1:2^{35}$ (12.8 days) |
| 11101 | $1:2^{34}$ (6.4 days) |
| 11100 | $1:2^{33}$ (77.0 hours) |
| 11011 | $1:2^{32}$ (38.5 hours) |
| 11010 | $1:2^{31}$ (19.2 hours) |
| 11001 | $1:2^{30}$ (9.6 hours) |
| 11000 | $1:2^{29}$ (4.8 hours) |
| 10111 | $1:2^{28}$ (2.4 hours) |
| 10110 | $1:2^{27}$ (72.2 minutes) |
| 10101 | $1:2^{26}$ (36.1 minutes) |
| 10100 | $1:2^{25}$ (18.0 minutes) |
| 10011 | $1:2^{24}$ (9.0 minutes) |
| 10010 | $1:2^{23}$ (4.5 minutes) |
| 10001 | $1:2^{22}$ (135.3 s) |
| 10000 | $1:2^{21}$ (67.7 s) |
| 01111 | $1:2^{20}$ (33.825 s) |
| 01110 | $1:2^{19}$ (16.912 s) |
| 01101 | $1:2^{18}$ (8.456 s) |
| 01100 | $1:2^{17}$ (4.228 s) |
| 01011 | 1:65536 (2.114 s) |
| 01010 | 1:32768 (1.057 s) |
| 01001 | 1:16384 (528.5 ms) |
| 01000 | 1:8192 (264.3 ms) |
| 00111 | 1:4096 (132.1 ms) |
| 00110 | 1:2048 (66.1 ms) |
| 00101 | 1:1024 (33 ms) |
| 00100 | 1:512 (16.5 ms) |
| 00011 | 1:256 (8.3 ms) |
| 00010 | 1:128 (4.1 ms) |
| 00001 | 1:64 (2.1 ms) |
| 00000 | 1:32 (1 ms) |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 20 – DSZPBOREN** Deep Sleep Zero-Power BOR Enable
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enable ZPBOR during deep sleep |
| 0 | Disable ZPBOR during deep sleep |

**Bits 19:17 – CPEN_DLY[2:0]** Charge-pump Ready Digital Delay (Safety Delay to Analog CP Ready)
n = (n+1) LPRC Clock Cycle Delay
**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

**Bit 16 – RTCEVTYPE** RTCC Event Type
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | RTC_EVENT |
| 0 | RTC_OUT |

**Bit 15 – MLPCLK_MOD** LPCLK Modifier in Counter/Delay Mode
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Divide-by 1.024 (Recommended when LPCLK = 32.768 KHz) |
| 0 | Divide-by 1 (Recommended when LPCLK = 32 KHz) |

**Bit 14 – VBKP_DIVSEL** VDDBUKPCORE LPCLK Clock Divider Selection
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Divide by 31.25 (Recommended when LPCLK = 32 KHz) |
| 0 | Divide-by 32 (Recommended when LPCLK = 32.768 KHz) |

**Bits 13:12 – VBKP_32KCSEL[1:0]** VDDBUKPCORE 32 KHz Clock Source Selection
**Notes:**

- When '00' or '01', the Deep Sleep mode is entered and it falls back to '11' before entering Deep Sleep. Any change of clock source results in gaps in the LPCLK output.
- These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 11 | LPRC |
| 10 | SOSC |
| 01 | POSC |
| 00 | FRC |

**Bit 11 – VBKP_1KCSEL** VDDBUKPCORE LPCLK Clock Selection
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Divide by 32 or 31.25 clock depending on VBKP_DIVSEL |
| 0 | 32 KHz low power clock |

**Bit 10 – RTCEVENT_EN** Output Enable for RTCC Event Output
**Note:** This bit is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 1 | Enables RTCC-Event output |
| 0 | Disables RTCC-Event output |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 9:8 – RTCEVENTSEL[1:0]** RTCC Event Selection

**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 00 | 1-Second clock |
| 01 | Alarm pulse |
| 1x | 32 KHz clock |

**Bits 7:0 – SOSC_CFG[7:0]** SOSC Configuration Bits

Gain configuration for SOSC Oscillator:
G3>G2>G1>G0

- 11 = Gain is G3
- 10 = Gain is G2
- 01 = Gain is G1
- 00 = Gain is G0

**Note:** These bits are only writable when CFGLOCK[1:0] is '00'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.5. Permission Group Configuration Register 1

**Name:** CFGPGQOS1
**Offset:** 0x50
**Reset:** 0xe040004c
**Property:** -

All bits in this register are writable only when CFGCON0.PGLOCK = 0.

There is no Flash location for this register because the purpose of this register to provide software based protection mechanism to device memory mapped region.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | WISIBQOS[1:0] | | FCQOS[1:0] | | | | DSUPG[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | | | R/W/L | R/W/L |
| Reset | 1 | 1 | 1 | 0 | | | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | CRYPTOQOS[1:0] | | CRYPTOPG[1:0] | | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | | | | |
| Reset | 0 | 1 | 0 | 0 | | | | |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | DMAPG[1:0] | |
| Access | | | | | | | R/W/L | R/W/L |
| Reset | | | | | | | 0 | 0 |

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

**Bits 31:30 – WISIBQOS[1:0]** Wireless SIB QOS Control Bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 29:28 – FCQOS[1:0]** FC Controller QOS Control Bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 25:24 – DSUPG[1:0]** DSU Permission Group
The DSU bus host has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- DSUPG[1:0] == `2'b11` : Initiator is assigned to Permission Group 3
- DSUPG[1:0] == `2'b10` : Initiator is assigned to Permission Group 2


© 2025 Microchip Technology Inc. and its subsidiaries

- DSUPG[1:0] == 2'b01 : Initiator is assigned to Permission Group 1
- DSUPG[1:0] == 2'b00 : Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 23:22 – CRYPTOQOS[1:0]** Crypto QOS Control Bits

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 21:20 – CRYPTOPG[1:0]** Crypto Permission Group

The Crypto bus host has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- CRYPTOPG[1:0] == 2'b11 : Initiator is assigned to Permission Group 3
- CRYPTOPG[1:0] == 2'b10 : Initiator is assigned to Permission Group 2
- CRYPTOPG[1:0] == 2'b01 : Initiator is assigned to Permission Group 1
- CRYPTOPG[1:0] == 2'b00 : Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 9:8 – DMAPG[1:0]** DMA (Rd/Wr) Permission Group

The DMA bus host has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- DMAPG[1:0] == 2'b11 : Initiator is assigned to Permission Group 3
- DMAPG[1:0] == 2'b10 : Initiator is assigned to Permission Group 2
- DMAPG[1:0] == 2'b01 : Initiator is assigned to Permission Group 1
- DMAPG[1:0] == 2'b00 : Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 3:2 – CPUQOS[1:0]** CPU I/D and System Bus QOS Control Bits

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 1:0 – CPUPG[1:0]** CPU (Code) Permission Group

The CPU Bus host has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- CPUPG[1:0] == 2'b11 : Initiator is assigned to Permission Group 3
- CPUPG[1:0] == 2'b10 : Initiator is assigned to Permission Group 2
- CPUPG[1:0] == 2'b01 : Initiator is assigned to Permission Group 1
- CPUPG[1:0] == 2'b00 : Initiator is assigned to Permission Group 0

**Note:** CPUPG[1:0] automatically reverts to 2'b00 when the CPU acknowledges entering into an NMI exception.


© 2025 Microchip Technology Inc. and its subsidiaries

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.6. Permission Group Configuration Register 2

**Name:** CFGPGQOS2
**Offset:** 0x60
**Reset:** 0x00084444
**Property:** -

All bits in this register are writable only when CFGCON0.PGLOCK = 0.

There is no Flash location for this register because the purpose of this register to provide software based protection mechanism to device memory mapped region.

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | ETHQOS[1:0] | | ETHPG[1:0] | |
| Access | | | | | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | | | | 1 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CAN1QOS[1:0] | | CAN1PG[1:0] | | CAN0QOS[1:0] | | CAN0PG[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | USBQOS[1:0] | | USBPG[1:0] | | ADCQOS[1:0] | | ADCPG[1:0] | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |

**Bits 19:18 – ETHQOS[1:0]** ETH QOS Control bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 17:16 – ETHPG[1:0]** ETH Permission Group
The Ethernet bus master has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- ETHPG[1:0] == 2'b11: Initiator is assigned to Permission Group 3
- ETHPG[1:0] == 2'b10: Initiator is assigned to Permission Group 2
- ETHPG[1:0] == 2'b01: Initiator is assigned to Permission Group 1
- ETHPG[1:0] == 2'b00: Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 15:14 – CAN1QOS[1:0]** CAN1 QOS Control bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 13:12 – CAN1PG[1:0]** CAN1 Permission Group
The CAN1 bus master has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- CAN1PG[1:0] == 2'b11: Initiator is assigned to Permission Group 3
- CAN1PG[1:0] == 2'b10: Initiator is assigned to Permission Group 2
- CAN1PG[1:0] == 2'b01: Initiator is assigned to Permission Group 1
- CAN1PG[1:0] == 2'b00: Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 11:10 – CAN0QOS[1:0]** CAN0 QOS Control bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 9:8 – CAN0PG[1:0]** CAN0 Permission Group
The CAN0 bus master has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- CAN0PG[1:0] == 2'b11: Initiator is assigned to Permission Group 3
- CAN0PG[1:0] == 2'b10: Initiator is assigned to Permission Group 2
- CAN0PG[1:0] == 2'b01: Initiator is assigned to Permission Group 1
- CAN0PG[1:0] == 2'b00: Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

**Bits 7:6 – USBQOS[1:0]** USB QOS Control bits
**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 5:4 – USBPG[1:0]** USB Permission Group
The USB bus master has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- USBPG[1:0] == 2'b11: Initiator is assigned to Permission Group 3
- USBPG[1:0] == 2'b10: Initiator is assigned to Permission Group 2
- USBPG[1:0] == 2'b01: Initiator is assigned to Permission Group 1
- USBPG[1:0] == 2'b00: Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 3:2 – ADCQOS[1:0]** ADC QOS Control bits

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.

| Value | Description |
|---|---|
| 00 | Disable; Background |
| 01 | Low; Sensitive bandwidth |
| 10 | Medium; Sensitive latency |
| 11 | High; Critical latency |

**Bits 1:0 – ADCPG[1:0]** ADC Permission Group

The ADC bus master has access to Access Controlled memory regions in the Bus Structure's Permission Groups SFRs.

- ADCPG[1:0] == 2'b11: Initiator is assigned to Permission Group 3
- ADCPG[1:0] == 2'b10: Initiator is assigned to Permission Group 2
- ADCPG[1:0] == 2'b01: Initiator is assigned to Permission Group 1
- ADCPG[1:0] == 2'b00: Initiator is assigned to Permission Group 0

**Note:** This field is only writable when CFGCON0.PGLOCK = 0.


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.7. I2C Configuration

**Name:** CFGI2C
**Offset:** 0x70
**Reset:** 0x00000000
**Property:** -

The CFGI2C dictates the peripheral clock selection described in the Clock and Reset Unit chapter. See *Clock and Reset Unit (CRU)* from Related Links.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | SLRTEN6 | | | | SLRTEN2 | SLRTEN1 | SLRTEN0 |
| Access | | R/W/L | | | | R/W/L | R/W/L | R/W/L |
| Reset | | 0 | | | | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | SLRCTRL6 | | | | SLRCTRL2 | SLRCTRL1 | SLRCTRL0 |
| Access | | R/W/L | | | | R/W/L | R/W/L | R/W/L |
| Reset | | 0 | | | | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | I2CDSEL6 | | | | I2CDSEL2 | I2CDSEL1 | I2CDSEL0 |
| Access | | R/W/L | | | | R/W/L | R/W/L | R/W/L |
| Reset | | 0 | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | SMBUSEN6 | | | | SMBUSEN2 | SMBUSEN1 | SMBUSEN0 |
| Access | | R/W/L | | | | R/W/L | R/W/L | R/W/L |
| Reset | | 0 | | | | 0 | 0 | 0 |

**Bit 30 – SLRTEN6** I2C Slew Rate Enable for SERCOM6

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C Slew Rate is disabled |
| 1 | I2C Slew Rate is enabled |

**Bit 26 – SLRTEN2** I2C Slew Rate Enable for SERCOM2

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C Slew Rate is disabled |
| 1 | I2C Slew Rate is enabled |

**Bit 25 – SLRTEN1** I2C Slew Rate Enable for SERCOM1

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C Slew Rate is disabled |
| 1 | I2C Slew Rate is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 24 – SLRTEN0** I2C Slew Rate Enable for SERCOM0

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C Slew Rate is disabled |
| 1 | I2C Slew Rate is enabled |

**Bit 22 – SLRCTRL6** Slew Rate Control for SERCOM6

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | Slew Rate Control is controlled via GPIO registers |
| 1 | Slew Rate Control is controlled via SERCOM registers |

**Bit 18 – SLRCTRL2** Slew Rate Control for SERCOM2

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | Slew Rate Control is controlled via GPIO registers |
| 1 | Slew Rate Control is controlled via SERCOM registers |

**Bit 17 – SLRCTRL1** Slew Rate Control for SERCOM1

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | Slew Rate Control is controlled via GPIO registers |
| 1 | Slew Rate Control is controlled via SERCOM registers |

**Bit 16 – SLRCTRL0** Slew Rate Control for SERCOM0

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | Slew Rate Control is controlled via GPIO registers |
| 1 | Slew Rate Control is controlled via SERCOM registers |

**Bit 14 – I2CDSEL6** I2C Delay Select for SERCOM6

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C delay is disabled |
| 1 | I2C delay is enabled |

**Bit 10 – I2CDSEL2** I2C Delay Select for SERCOM2

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C delay is disabled |
| 1 | I2C delay is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 9 – I2CDSEL1** I2C Delay Select for SERCOM1

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C delay is disabled |
| 1 | I2C delay is enabled |

**Bit 8 – I2CDSEL0** I2C Delay Select for SERCOM0

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | I2C delay is disabled |
| 1 | I2C delay is enabled |

**Bit 6 – SMBUSEN6** SMBus Enable for SERCOM6

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | SMBus is disabled |
| 1 | SMBus is enabled |

**Bit 2 – SMBUSEN2** SMBus Enable for SERCOM2

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | SMBus is disabled |
| 1 | SMBus is enabled |

**Bit 1 – SMBUSEN1** SMBus Enable for SERCOM1

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | SMBus is disabled |
| 1 | SMBus is enabled |

**Bit 0 – SMBUSEN0** SMBus Enable for SERCOM0

**Note:**

- This field is only writable when CFGLOCK[1:0] is '00'.

| Value | Description |
|---|---|
| 0 | SMBus is disabled |
| 1 | SMBus is enabled |

**Related Links**

Clock and Reset Unit (CRU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.8. Peripheral Clock Generator 1

**Name:** CFGPCLKGEN1
**Offset:** 0x80
**Reset:** 0x00000000
**Property:** -

The CFGPCLKGEN1 dictates the peripheral clock selection described in the Clock and Reset Unit chapter. See *Clock and Reset Unit (CRU)* from Related Links.

There is no Flash location for this register because the purpose of this register is to provide application-based peripheral clocking selection. This is best handled in the application software drivers.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CM4TCD | CM4TCSEL[2:0] | | | SERCOM345CD | SERCOM345CSEL[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 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TCC12CD | TCC12CSEL[2:0] | | | SERCOM6CD | SERCOM6CSEL[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 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | SERCOM012CD | SERCOM012CSEL[2:0] | | | FREQMMCD | FREQMMCSEL[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 |
|---|---|---|---|---|---|---|---|---|
| | FREQMRCD | FREQMRCSEL[2:0] | | | EICCD | EICCSEL[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 31 – CM4TCD** CM4_Trace Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 30:28 – CM4TCSEL[2:0]** CM4_Trace Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 27 – SERCOM345CD** SERCOM3, SERCOM4 and SERCOM5 Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Clock is enabled |

**Bits 26:24 – SERCOM345CSEL[2:0]** SERCOM3, SERCOM4 and SERCOM5 Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 23 – TCC12CD** TCC1 and TCC2 Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 22:20 – TCC12CSEL[2:0]** TCC1 and TCC2 Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 19 – SERCOM6CD** SERCOM6 Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 18:16 – SERCOM6CSEL[2:0]** SERCOM6 Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 15 – SERCOM012CD** SERCOM0, SERCOM1 and SERCOM2 Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 14:12 – SERCOM012CSEL[2:0]** SERCOM0, SERCOM1 and SERCOM2 Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 11 – FREQMMCD** FREQM Measurement Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 10:8 – FREQMMCSEL[2:0]** FREQM Measurement Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 7 – FREQMRCD** FREQM Reference Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 6:4 – FREQMRCSEL[2:0]** FREQM Reference Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 3 – EICCD** EIC Peripheral Clock Disable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 2:0 – EICCSEL[2:0]** EIC Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Related Links**
Clock and Reset Unit (CRU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.9. Peripheral Clock Generator 2

**Name:** CFGPCLKGEN2
**Offset:** 0x90
**Reset:** 0x00000000
**Property:** -

The CFGPCLKGEN2 dictates the peripheral clock selection described in the Clock and Reset Unit chapter. See *Clock and Reset Unit (CRU)* from Related Links.

There is no Flash location for this register because the purpose of this register is to provide an application-based peripheral clocking selection. This is best handled in the application software drivers.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC8CD | EVSYSC8SEL[2:0] | | | EVSYSC7CD | EVSYSC7SEL[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 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC6CD | EVSYSC6SEL[2:0] | | | EVSYSC5CD | EVSYSC5SEL[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 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC4CD | EVSYSC4SEL[2:0] | | | EVSYSC3CD | EVSYSC3SEL[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 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC2CD | EVSYSC2SEL[2:0] | | | EVSYSC1CD | EVSYSC1SEL[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 31 – EVSYSC8CD** EVSYS Channel 8 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 30:28 – EVSYSC8SEL[2:0]** EVSYS Channel 8 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 27 – EVSYSC7CD** EVSYS Channel 7 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 26:24 – EVSYSC7SEL[2:0]** EVSYS Channel 7 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 23 – EVSYSC6CD** EVSYS Channel 6 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 22:20 – EVSYSC6SEL[2:0]** EVSYS Channel 6 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 19 – EVSYSC5CD** EVSYS Channel 5 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 18:16 – EVSYSC5SEL[2:0]** EVSYS Channel 5 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 15 – EVSYSC4CD** EVSYS Channel 4 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 14:12 – EVSYSC4SEL[2:0]** EVSYS Channel 4 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 11 – EVSYSC3CD** EVSYS Channel 3 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Clock is enabled |

**Bits 10:8 – EVSYSC3SEL[2:0]** EVSYS Channel 3 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 7 – EVSYSC2CD** EVSYS Channel 2 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 6:4 – EVSYSC2SEL[2:0]** EVSYS Channel 2 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 3 – EVSYSC1CD** EVSYS Channel 1 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 2:0 – EVSYSC1SEL[2:0]** EVSYS Channel 1 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Related Links**
Clock and Reset Unit (CRU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.10. Peripheral Clock Generator 3

**Name:** CFGPCLKGEN3
**Offset:** 0xA0
**Reset:** 0x00000000
**Property:** -

The CFGPCLKGEN3 dictates the peripheral clock selection described in the Clock and Reset Unit chapter. See *Clock and Reset Unit (CRU)* from Related Links.

There is no Flash location for this register because the purpose of this register is to provide an application-based peripheral clocking selection. This is best handled in the application software drivers.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CVDCD | CVDCSEL[2:0] | | | ETHD | ETHCLKSEL[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 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TCC0CD | TCC0CSEL[2:0] | | | ACCD | ACCSEL[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 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC12CD | EVSYSC12SEL[2:0] | | | EVSYSC11CD | EVSYSC11SEL[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 |
|---|---|---|---|---|---|---|---|---|
| | EVSYSC10CD | EVSYSC10SEL[2:0] | | | EVSYSC9CD | EVSYSC9SEL[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 31 – CVDCD** CVD Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 30:28 – CVDCSEL[2:0]** CVD Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 27 – ETHD** Ethernet Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 26:24 – ETHCLKSEL[2:0]** Ethernet Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 23 – TCC0CD** TCC0 Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 22:20 – TCC0CSEL[2:0]** TCC0 Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 19 – ACCD** Analog Comparator Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 18:16 – ACCSEL[2:0]** Analog Comparator Peripheral Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 15 – EVSYSC12CD** EVSYS Channel 12 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 14:12 – EVSYSC12SEL[2:0]** EVSYS Channel 12 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 11 – EVSYSC11CD** EVSYS Channel 11 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Clock is enabled |

**Bits 10:8 – EVSYSC11SEL[2:0]** EVSYS Channel 11 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 7 – EVSYSC10CD** EVSYS Channel 10 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 6:4 – EVSYSC10SEL[2:0]** EVSYS Channel 10 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 3 – EVSYSC9CD** EVSYS Channel 9 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 2:0 – EVSYSC9SEL[2:0]** EVSYS Channel 9 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Related Links**
Clock and Reset Unit (CRU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.11. Peripheral Clock Generator 4

**Name:** CFGPCLKGEN4
**Offset:** 0xB0
**Reset:** 0x00000000
**Property:** -

The CFGPCLKGEN4 dictates the peripheral clock selection described in the Clock and Reset Unit chapter. See *Clock and Reset Unit (CRU)* from Related Links.

There is no Flash location for this register because the purpose of this register is to provide an application-based peripheral clocking selection. This is best handled in the application software drivers.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | CAN1CD | CAN1CSEL[2:0] | | | CAN0CD | CAN0SEL[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 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | TC89CD | TC89CSEL[2:0] | | | TC67CD | TC67CSEL[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 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TC45CD | TC45CSEL[2:0] | | | TC23CD | TC23CSEL[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 |
|---|---|---|---|---|---|---|---|---|
| | TC1CD | TC1CSEL[2:0] | | | TC0CD | TC0CSEL[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 31 – CAN1CD** CAN1 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 30:28 – CAN1CSEL[2:0]** CAN1 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 27 – CAN0CD** CAN0 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 26:24 – CAN0SEL[2:0]** CAN0 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 23 – TC89CD** TC8 and TC9 Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 22:20 – TC89CSEL[2:0]** TC8 and TC9 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 19 – TC67CD** TC6 and TC7 Peripheral Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 18:16 – TC67CSEL[2:0]** TC6 and TC7 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 15 – TC45CD** TC4 and TC5 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 14:12 – TC45CSEL[2:0]** TC4 and TC5 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 11 – TC23CD** TC2 and TC3 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Clock is enabled |

**Bits 10:8 – TC23CSEL[2:0]** TC2 and TC3 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 7 – TC1CD** TC1 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 6:4 – TC1CSEL[2:0]** TC1 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 clock is selected |
| 7 | Low power clock is selected |

**Bit 3 – TC0CD** TC0 Clock Enable
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | Clock is disabled |
| 1 | Clock is enabled |

**Bits 2:0 – TC0CSEL[2:0]** TC0 Clock Selection
**Note:** This field is only writable when CFGCON0.PGLOCK is '0'.

| Value | Description |
|---|---|
| 0 | No clock is selected |
| 1-6 | REFO1-6 is clock selected |
| 7 | Low power clock is selected |

**Related Links**
Clock and Reset Unit (CRU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.12. User Unique ID

**Name:** USER_ID
**Offset:** 0xC0
**Reset:** 0x00000000
**Property:** -

The User ID is a 16-bit ID that can be programmed to differentiate products that use the same device. The User ID value may be read directly out of the USER_ID register or through the SWD interface.
There is no dedicated status bit to indicate when the User ID value is loaded into the USER_ID register and is ready to be read from SWD. It is assumed that a non-zero value for the User ID is used to indicate that the User ID is loaded.

The USER_ID register is reset on power-up, then is loaded with trusted data from FBCFG5 during the pre-boot period, and it is controlled.

Trusted data from Flash means that when there is no BCFG* fail status during Flash, configuration word reads. If accompanied by fail status BCFGFAIL (RCON[26]) or blank/erase indication, Reset values (described in the following register description) are retained and new values from FBCFG5/ FUSERID are not loaded.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | USER_ID[15:8] | | | | | | | |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

**Bits 15:0 – USER_ID[15:0]** User unique ID
**Note:** This field is only writable when CFGLOCK[1:0] is '00'.


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.13. System Key Register

**Name:** SYSKEY
**Offset:** 0xD0
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | SYSKEY[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 |
|---|---|---|---|---|---|---|---|---|
| | SYSKEY[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 |
|---|---|---|---|---|---|---|---|---|
| | SYSKEY[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 |
|---|---|---|---|---|---|---|---|---|
| | SYSKEY[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 – SYSKEY[31:0]** System Key
Keys are written to this register as part of a sequence to unlock system-critical registers. A successful key write to this register will set the system signal.


© 2025 Microchip Technology Inc. and its subsidiaries

### 22.9.14. Boot Configuration 0

**Name:** BCFG0
**Offset:** 0x200
**Reset:** 0x00
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BINFOVALID0 | | SIGN | CP | | | | |
| Access | R | | R | R | | | | |
| Reset | c | | c | c | | | | |

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | PCSCMODE | |
| Access | | | | | | | R | |
| Reset | | | | | | | c | |

**Bit 31 – BINFOVALID0** First 256-bit BCFG Information is Valid

**Notes:**

1. This bit is added to know if the information from Flash is valid or invalid. The BCFG area is critical to device boot-up.
2. Trusted FBCFG* data = (BINFOVALID = 0) and (BCFGFAIL = 0).
3. It is recommended for the application to program this bit to '0' for proper operation.

| Value | Description |
|---|---|
| 1 | FBCFG0 to FBCFG7 is not valid (Untrusted, flash values are ignored and safe values are used) |
| 0 | FBCFG0 to FBCFG7 is valid (Trusted and loaded from Flash) |

**Bit 29 – SIGN** Flash SIGN Bit

This bit is a read only bit. Reading this bit returns the value of the SIGN bit in the FSIGN0 fuse location (invisible to user) in the NVR memory.

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

**Bit 28 – CP** Code Protect

The CP bit is a read-only bit. Reading this bit returns the inverted value of the CP bit in the FCPN0 Flash location (~FCPN0.CP && ~FSIGN0.SIGN). To set Code Protect, the CP bit in the FCPN0 fuse location in the NVR memory must be set to '1'.

| Value | Description |
|---|---|
| 1 | Protection enabled |
| 0 | Protection disabled |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 1 – PCSCMODE** PCHE Single Cache Mode

**Note:** This bit must be changed only when there are no active accesses to Flash from CPU. If this bit is changed while the CPU is accessing Flash, a system hang may result.

| Value | Description |
|---|---|
| 1 | PCHE ICache Only. CPU Instructions (code, data) go to PCHE ICache only. |
| 0 | PCHE ICache and DCache. CPU opcodes go to PCHE ICache port and data goes to PCHE DCache port. |


© 2025 Microchip Technology Inc. and its subsidiaries

# 23. Peripheral Module Disable (PMD)

## 23.1. Overview

The Peripheral Module Disable (PMD) registers provide a method to disable a peripheral module by stopping all clock sources supplied to that module. When a peripheral is disabled using an appropriate PMD control bit, the peripheral is in a minimum power consumption state. The control and status registers associated with the peripheral are also disabled; therefore, writing to those registers does not have effect and read values are invalid.

## 23.2. Enabling Peripherals

The PMD register bits control the operation of individual peripherals on the device. When a peripheral's associated PMD bit is '0', the peripheral is enabled and operates as programmed. However, when the associated PMD bit is '1', the peripheral logic, memory map and SFR bits are removed from visibility and the peripheral is held in Reset. This disabled state provides for the lowest power state of the peripheral.

Before a peripheral may be configured or used, clear the corresponding PMD register bit to enable the peripheral.

There are some caveats to use PMD bits. The following must be observed:

1. Disabling a peripheral while its ON bit is '0' results in an undefined behavior of the external interface.
2. For bus initiators, the software must verify that the module is not busy after setting the ON bit to '0' before disabling it.
3. Setting the PMD bit when there is a pending interrupt results in undefined behavior. Therefore, all interrupt flags must be cleared before setting the associated PMD bit.

## 23.3. Controlling Configuration Changes

Because peripherals can be disabled during run time, some restrictions on disabling peripherals are needed to prevent accidental configuration changes. The PIC32CX-BZ6 devices include a Control register lock sequence feature to prevent alterations to the enabled or disabled peripherals.

### 23.3.1. Control Register Lock

Under normal operation, writing to the PMDx registers is not allowed. Attempted writes appear to execute normally, but the contents of the registers remain unchanged. To change these registers, they must be unlocked in the hardware. The CFGCON0.PMDLOCK Configuration bit controls the register lock. Setting CFGCON0.PMDLOCK prevents writes to the control registers, and clearing CFGCON0.PMDLOCK allows write operation.


© 2025 Microchip Technology Inc. and its subsidiaries

## 23.4. PMD Register Summary

See the *PMD* module in the *Product Memory Mapping Overview* from Related Links for the base address.

**Note:** All registers in this table have corresponding CLR, SET and INV registers at its virtual address, plus an offset of 0x4, 0x8 and 0xC, respectively. See *CLR, SET and INV Registers* from Related Links.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 ... 0xDF | Reserved | | | | | | | | | |
| 0xE0 | PMD1 | 7:0 | ADCMD | ACMD | DACMD | | | | | |
| | | 15:8 | | | | | | ADCSAR-SHRMD | CVDMD | ADCSARMD |
| | | 23:16 | | | | | | | | RTCC |
| | | 31:24 | | | QSPIMD | | | | | |
| 0xE4 ... 0xEF | Reserved | | | | | | | | | |
| 0xF0 | PMD2 | 7:0 | | | | | | | | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | REFO4MD | REFO3MD | REFO2MD | REFO1MD | | | REFO6MD | REFO5MD |
| 0xF4 ... 0xFF | Reserved | | | | | | | | | |
| 0x0100 | PMD3 | 7:0 | QEIMD | SER6MD | SER5MD | SER4MD | SER3MD | SER2MD | SER1MD | SER0MD |
| | | 15:8 | TC7MD | TC6MD | TC5MD | TC4MD | TC3MD | TC2MD | TC1MD | TC0MD |
| | | 23:16 | | TCC2MD | TCC1MD | TCC0MD | | | TC9MD | TC8MD |
| | | 31:24 | | | | | CAN1MD | CAN0MD | ETHMD | USBMD |

**Related Links**

Product Memory Mapping Overview

CLR, SET and INV Registers

## 23.5. Register Description

Some peripherals include module enable bits internally. The PMD bit is used for clock gating of the PBx_CLK and GCLK for all peripherals. If the peripheral also includes the internal enable bit, the PMD bit and internal enable Configuration bit must be configured by software for that peripheral.

The following table summarizes each peripheral's enable and disable controls. For more details on the internal enable/disable control, see *Peripheral Access Controller (PAC)* from Related Links.

**Table 23-1.** Module Enable/Disable Controls

| Module | PMD Control | Module Control | Enable/Disable Strategy |
|---|---|---|---|
| AC | Present | Present | Disable at PMD or Module |
| ADC | Present | Present | Disable at PMD or Module |
| CCL | NA | Present | Disable at Module |
| CVD | Present | Present | Disable at PMD or Module |
| CMCC | NA | Present | Disable at Module |
| DMAC | NA | Present | Disable at Module |
| DSU | NA | NA | Always Enabled (Dynamic ON/OFF) |
| EIC | NA | Present | Disable at Module |
| EVSYS | NA | NA | Always Enabled (Dynamic ON/OFF) |
| FREQM | NA | Present | Disable at Module |
| PAC | NA | NA | Always Enabled (Dynamic ON/OFF) |


© 2025 Microchip Technology Inc. and its subsidiaries

**Table 23-1.** Module Enable/Disable Controls (continued)

| Module | PMD Control | Module Control | Enable/Disable Strategy |
|---|---|---|---|
| RAMECC | NA | NA | Disabled by default |
| RTCC | Present | Present | Disable at PMD or Module |
| SERCOM | Present | Present based on VSEL | Disable at PMD or Module |
| TC | Present | Present | Disable at PMD or Module |
| TCC | Present | Present | Disable at PMD or Module |
| CANn | Present | Present based on VSEL | Disable at PMD or Module |
| USB | Present | Present based on VSEL | Disable at PMD or Module |
| ETH | Present | Present based on VSEL | Disable at PMD or Module |
| ADCSARHR | Present | Present | Disable at PMD or Module |
| ADCSAR | Present | Present | Disable at PMD or Module |
| DAC | Present | Present | Disable at PMD or Module |
| QEI | Present | Present | Disable at PMD or Module |

**Note:** For modules with both PMD control and Module control, Enable = PMDx=0 AND Module Enable=1, Disable =PMDx=1 OR Module Enable=0.

**Related Links**

Peripheral Access Controller (PAC)


© 2025 Microchip Technology Inc. and its subsidiaries

### 23.5.1. PMD1 - Peripheral Module Disable 1 Register

**Name:** PMD1
**Offset:** 0x00E0
**Reset:** 0x00000000
**Property:** -

**Note:** This register's bits are only writable when CFGCON0.PMDLOCK = 0.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | QSPIMD | | | | | |
| Access | | | R/W/L | | | | | |
| Reset | | | 0 | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | RTCC |
| Access | | | | | | | | R/W/L |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | ADCSAR-SHRMD | CVDMD | ADCSARMD |
| Access | | | | | | R/W/L | R/W/L | R/W/L |
| Reset | | | | | | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADCMD | ACMD | DACMD | | | | | |
| Access | R/W/L | R/W/L | R/W/L | | | | | |
| Reset | 0 | 0 | 0 | | | | | |

**Bit 29 – QSPIMD** SPI Module Disable
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the SPI module |
| 0 | Enables the SPI module |

**Bit 16 – RTCC** RTCC Module Disable bit
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disabled |
| 0 | Enabled |

**Bit 10 – ADCSAR-SHRMD** Shared ADC SAR Core Module Disable bit
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the shared ADC SAR core module |
| 0 | Enables the shared ADC SAR core module |

**Bit 9 – CVDMD** Shared CVD Module Disable Bit

| Value | Description |
|---|---|
| 1 | Disables the corresponding shared CVD module |
| 0 | Enables the corresponding shared CVD module |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bit 8 – ADCSARMD** Shared ADC SAR Core Module Disable Bit

| Value | Description |
|---|---|
| 1 | Disables the shared ADC SAR Core module |
| 0 | Enables the shared ADC SAR Core module |

**Bit 7 – ADCMD** ADC Controller Module Disable

| Value | Description |
|---|---|
| 1 | Disables the ADC Controller module |
| 0 | Enables the ADC Controller module |

**Bit 6 – ACMD** AC Module Disable

| Value | Description |
|---|---|
| 1 | Disables the AC module |
| 0 | Enables the AC module |

**Bit 5 – DACMD** DAC Module Disable

**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the DAC module |
| 0 | Enables the DAC module |


© 2025 Microchip Technology Inc. and its subsidiaries

### 23.5.2. PMD2 - Peripheral Module Disable 2 Register

**Name:** PMD2
**Offset:** 0x00F0
**Reset:** 0x00000000
**Property:** -

**Note:** This register bits are only writable when CFGCON0.PMDLOCK = 0.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | REFO4MD | REFO3MD | REFO2MD | REFO1MD | | | REFO6MD | REFO5MD |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | | | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

| 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 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

**Bit 31 – REFO4MD** Reference (Clock) Out 4 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 4 |
| 0 | Enables the Reference (clock) out 4 |

**Bit 30 – REFO3MD** Reference (Clock) Out 3 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 3 |
| 0 | Enables the Reference (clock) out 3 |

**Bit 29 – REFO2MD** Reference (Clock) Out 2 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 2 |
| 0 | Enables the Reference (clock) out 2 |

**Bit 28 – REFO1MD** Reference (Clock) Out 1 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 1 |
| 0 | Enables the Reference (clock) out 1 |

**Bit 25 – REFO6MD** Reference (Clock) Out 6 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 6 |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | Enables the Reference (clock) out 6 |

**Bit 24 – REFO5MD** Reference (Clock) Out 5 Disable

| Value | Description |
|---|---|
| 1 | Disables the Reference (clock) out 5 |
| 0 | Enables the Reference (clock) out 5 |


© 2025 Microchip Technology Inc. and its subsidiaries

### 23.5.3. PMD3 - Peripheral Module Disable 3 Register

**Name:** PMD3
**Offset:** 0x0100
**Reset:** 0x00000000
**Property:** -

**Note:** This register's bits are only writable when CFGCON0.PMDLOCK = 0.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | CAN1MD | CAN0MD | ETHMD | USBMD |
| Access | | | | | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | TCC2MD | TCC1MD | TCC0MD | | | TC9MD | TC8MD |
| Access | | R/W/L | R/W/L | R/W/L | | | R/W/L | R/W/L |
| Reset | | 0 | 0 | 0 | | | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | TC7MD | TC6MD | TC5MD | TC4MD | TC3MD | TC2MD | TC1MD | TC0MD |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | QEIMD | SER6MD | SER5MD | SER4MD | SER3MD | SER2MD | SER1MD | SER0MD |
| Access | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L | R/W/L |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 26, 27 – CANnMD** CANn Module Disable (n=0...1)
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the CANn module |
| 0 | Enables the CANn module |

**Bit 25 – ETHMD** Ethernet Module Disable
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the Ethernet module |
| 0 | Enables the Ethernet module |

**Bit 24 – USBMD** USB Module Disable
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the USB module |
| 0 | Enables the USB module |

**Bits 20, 21, 22 – TCCnMD** TCCn Module Disable (n=0...2)
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the TCCn module |
| 0 | Enables the TCCn module |


© 2025 Microchip Technology Inc. and its subsidiaries

**Bits 8, 9, 10, 11, 12, 13, 14, 15, 16, 17 – TCnMD** TCn Module Disable (n=0...9)
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the TCn module |
| 0 | Enables the TCn module |

**Bit 7 – QEIMD** QEI Module Disable
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the QEI module |
| 0 | Enables the QEI module |

**Bits 0, 1, 2, 3, 4, 5, 6 – SERnMD** SERCOM Module Disable (n=0...6)
**Note:** This bit is only writable when CFGCON0.PMDLOCK = 0

| Value | Description |
|---|---|
| 1 | Disables the SERCOMn module |
| 0 | Enables the SERCOMn module |


© 2025 Microchip Technology Inc. and its subsidiaries

# 24. Peripheral Access Controller (PAC)

## 24.1. Overview

The Peripheral Access Controller (PAC) provides an interface for the locking and unlocking of peripheral registers within the device. It reports all violations that might happen when accessing a peripheral:

- Write-protected access
- Illegal access
- Enable-protected access
- Access when clock synchronization or software reset is ongoing

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-existent address is detected.

**Note:** The modules attached to the PB-Bridge-PIC bridge and wireless subsystem, as well as RTCC and DSCON are excluded from PAC. The protection mechanism described in CFG protects critical system registers. See *System Configuration and Register Locking (CFG)* from Related Links.

**Related Links**

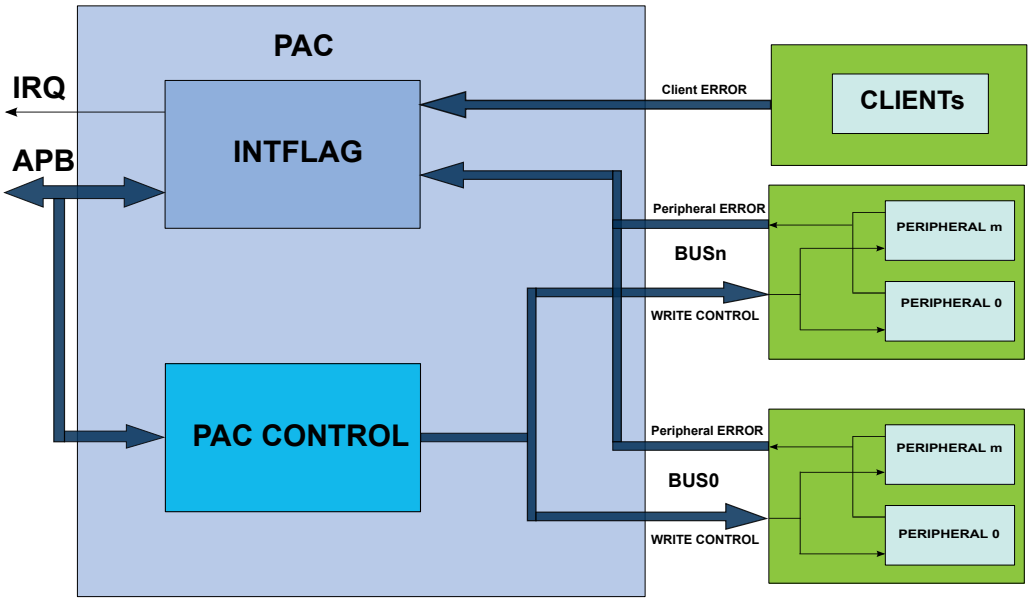
System Configuration and Register Locking (CFG)

## 24.2. Features

- Manages Write Protection Access and Reports Access Errors for the Peripheral Modules or Bridges
- Manages Security Attribution for the Peripheral Modules

## 24.3. Block Diagram

**Figure 24-1.** PAC Block Diagram




## 24.4. Product Dependencies

The following sections describe how the other parts of the system must be configured correctly to use this peripheral.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.4.1. IO Lines

Not applicable.

### 24.4.2. Power Management

The PAC can continue to operate in any sleep modes (Idle, Standby Sleep) 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.

### 24.4.3. Clocks

The PAC bus clocks (CLK_PAC_APB and CLK_PAC_AHB) can be enabled and disabled in the Main Clock module.

For details about default state of CLK_PAC_APB and CLK_PAC_AHB, see *System and Peripheral Bus Clock Generation (CLKGEN)* from Related links.

**Related Links**

System and Peripheral Bus Clock Generation (CLKGEN)

### 24.4.4. DMA

Not applicable.

### 24.4.5. Interrupts

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

**Table 24-1.** Interrupt Lines

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

### 24.4.6. Events

The events are connected to the Event System, which may need configuration. See *Event System (EVSYS)* from Related Links.

**Related Links**

Event System (EVSYS)

### 24.4.7. Debug Operation

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

### 24.4.8. Register Access Protection

All registers with write access can be write protected optionally by the 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 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.


© 2025 Microchip Technology Inc. and its subsidiaries

## 24.5. Functional Description

### 24.5.1. 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.

### 24.5.2. Basic Operation

#### 24.5.2.1. Initialization, Enabling and Resetting

The PAC is always enabled after reset.

Only a hardware reset will reset the PAC module.

#### 24.5.2.2. Operations

The PAC module allows the user to set, clear or lock the write protection status of all peripherals on all Peripheral Bridges, except the peripherals on PB-PIC-Bridge bus.

If a peripheral register violation occurs, the Peripheral Interrupt Flag n registers (INTFLAGn) are updated to inform the user of 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. See *Peripheral Access Errors* from Related Links.

The PAC module also reports the errors occurring at the client bus level when an access to a reserved area is detected. The AHB Subordinate Bus Interrupt Flag register (INTFLAGAHB) informs the user of the status of the violation in the corresponding client. See *AHB Subordinate Bus Errors* from Related Links.

**Related Links**

Peripheral Access Errors

AHB Subordinate Bus Errors

#### 24.5.2.3. Peripheral Access Errors

The following events 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 data sheet can be protected. If a peripheral is not write protected, write data accesses are performed as usual. If a peripheral is write protected and if a write access is attempted, data will not be written and the peripheral returns an access error. The corresponding interrupt flag bit in the INTFLAGn register is set.
- Illegal access – Access to an unimplemented register within the module
- Synchronized write error – For write-synchronized registers, an error is reported if the register is written while a synchronization is ongoing.

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

#### 24.5.2.4. 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 – An unique identifier corresponding to a peripheral
- WRCTRL.KEY – A key value that defines the operation to be done on the control access bit These operations can be:


© 2025 Microchip Technology Inc. and its subsidiaries

- Clear protection – Removes the write-access protection for the peripheral selected by WRCTRL.PERID. Write accesses are allowed for the registers in this peripheral.
- Set protection – Sets the write-access protection for the peripheral selected by WRCTRL.PERID. Write accesses are not allowed for the registers with the write-protection property in this peripheral.
- Set and lock protection bit – Sets 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.

#### 24.5.2.5. Write Access Protection Management Errors

Only word-wise writes to the WRCTRL register effectively changes the access protection. Other types of accesses have no effect and 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, the PAC returns an error and does so 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. However, in applications where a write-protected peripheral is used in several contexts, for example, interrupt, care must be taken so that either the interrupt cannot happen while the main application or other interrupt levels manipulate 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 (for example, key error, double protect error and so on) set the INTFLAGA.PAC flag.

#### 24.5.2.6. AHB Subordinate Bus Errors

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

#### 24.5.2.7. Generating Events

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

### 24.5.3. DMA Operation

Not applicable.

### 24.5.4. 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


© 2025 Microchip Technology Inc. and its subsidiaries

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 at the 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:** Interrupts must be globally enabled for interrupt requests to be generated.

### 24.5.5. Events

The PAC generates 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.

### 24.5.6. Sleep Mode Operation

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

### 24.5.7. Synchronization

Not applicable.


© 2025 Microchip Technology Inc. and its subsidiaries

## 24.6. Register Summary

See the *PAC* module in *Product Memory Mapping Overview* from Related Links for base address.

| 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 | PBBB | PBAB | PFLASH | CFLASH | SRAM4 | SRAM3 | SRAM2 | SRAM1 |
| | | 15:8 | | | PBDB | CRYPTO | RoT | QSPI | PBPICB | PBCB |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x14 | INTFLAGA | 7:0 | TC2 | TC1 | TC0 | SERCOM1 | SERCOM0 | EIC | FREQM | PAC |
| | | 15:8 | TCC2 | TCC1 | TCC0 | TC7 | TC6 | TC5 | TC4 | TC3 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x18 | INTFLAGB | 7:0 | | | ETH | RAMECC | EVSYS | DMAC | | DSU |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x1C | INTFLAGC | 7:0 | AC | CCL | | | | | SERCOM6 | QSPI |
| | | 15:8 | | | | | | | HMTX | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x20 | INTFLAGD | 7:0 | | | TC9 | TC8 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 |
| | | 15:8 | | | | | | | CAN1 | CAN0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x24 ... 0x33 | Reserved | | | | | | | | | |
| 0x34 | STATUSA | 7:0 | TC2 | TC1 | TC0 | SERCOM1 | SERCOM0 | EIC | FREQM | PAC |
| | | 15:8 | TCC2 | TCC1 | TCC0 | TC7 | TC6 | TC5 | TC4 | TC3 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x38 | STATUSB | 7:0 | | | ETH | RAMECC | EVSYS | DMAC | | DSU |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x3C | STATUSC | 7:0 | AC | CCL | | | | | SERCOM6 | QSPI |
| | | 15:8 | | | | | | | HMTX | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x40 | STATUSD | 7:0 | | | TC9 | TC8 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 |
| | | 15:8 | | | | | | | CAN1 | CAN0 |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |

**Related Links**

Product Memory Mapping Overview


© 2025 Microchip Technology Inc. and its subsidiaries

## 24.7. 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 PAC. Optional PAC write protection is denoted by the PAC Write-Protection property in each individual register description. See *Register Access Protection* from Related Links.

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.

**Related Links**

Register Access Protection


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.1. 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. Formula to calculate the peripheral identifier:
$PERID = 32^{*}\,\mathrm{BridgeNumber} + \mathrm{N}$

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. For example, PAC peripheral belongs to Peripheral Bridge A at the '0' bit position (see *INTFLAGA* from Related Links). Therefore, PERID = 32*0+0 = 0 for PAC peripheral.

**Table 24-2.** PERID Values

| Peripheral Bridge Name | BridgeNumber | PERID Values |
|---|---|---|
| A | 0 | 0+N |
| B | 1 | 32+N |
| C | 2 | 64+N |
| D | 3 | 96+N |
| E | 4 | 128+N |


© 2025 Microchip Technology Inc. and its subsidiaries

**Related Links**

INTFLAGA

Peripheral Interrupt Flag Status - Bridge A


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.2. Event Control

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

| 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 is 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.3. 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 can 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 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 clears 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.4. 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 can also be reflected in the Interrupt Enable Clear 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 is 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 sets 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.5. Bridge Interrupt Flag Status

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

These flags are cleared by writing a '1' to the corresponding bit.

These flags are set when an access error is detected by the corresponding AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '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 |
|---|---|---|---|---|---|---|---|---|
| | | | PBDB | CRYPTO | RoT | QSPI | PBPICB | PBCB |
| Access | | | RW | RW | RW | RW | RW | RW |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PBBB | PBAB | PFLASH | CFLASH | SRAM4 | SRAM3 | SRAM2 | SRAM1 |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 13 – PBDB** Interrupt Flag for PBDB (PB-Bridge-D)
This flag is set when an access error is detected by the PBDB AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PBDB interrupt flag.

**Bit 12 – CRYPTO** Interrupt Flag for Crypto
This flag is set when an access error is detected by the Crypto AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the crypto interrupt flag.

**Bit 11 – RoT** Interrupt Flag for RoT ROM
This flag is set when an access error is detected by the RoT ROM Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the RoT interrupt flag.

**Bit 10 – QSPI** Interrupt Flag for QSPI (Reserved for future use)
This flag is set when an access error is detected by the QSPI AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '1' to this bit clears the QSPI interrupt flag.

**Bit 9 – PBPICB** Interrupt Flag for PBPICB (PB-PIC-Bridge) (Reserved for future use)
This flag is set when an access error is detected by the PBPICB AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PBPICB interrupt flag.

**Bit 8 – PBCB** Interrupt Flag for PBCB (PB-Bridge-C)
This flag is set when an access error is detected by the PBCB Bridge AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PBCB interrupt flag.

**Bit 7 – PBBB** Interrupt Flag for PBBB (PB-Bridge-B)
This flag is set when an access error is detected by the PBBB Bridge AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PBBB interrupt flag.

**Bit 6 – PBAB** Interrupt Flag for PBAB (PB-Bridge-A)
This flag is set when an access error is detected by the PBAB Bridge AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PBAB interrupt flag.

**Bit 5 – PFLASH** Interrupt Flag for PFLASH (Peripheral Flash)
This flag is set when an access error is detected by the PFLASH AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the PFLASH interrupt flag.

**Bit 4 – CFLASH** Interrupt Flag for CFLASH (CPU Flash)
This flag is set when an access error is detected by the CFLASH AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the CFLASH interrupt flag.

**Bit 3 – SRAM4** Interrupt Flag for SRAM4
This flag is set when an access error is detected by the SRAM4 AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the SRAM4 interrupt flag.

**Bit 2 – SRAM3** Interrupt Flag for SRAM3
This flag is set when an access error is detected by the SRAM3 AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the SRAM3 interrupt flag.

**Bit 1 – SRAM2** Interrupt Flag for SRAM2
This flag is set when an access error is detected by the SRAM2 AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '1' to this bit clears the SRAM2 interrupt flag.

**Bit 0 – SRAM1** Interrupt Flag for SRAM1

This flag is set when an access error is detected by the SRAM1 AHB Client, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' has no effect.
Writing a '1' to this bit clears the SRAM1 interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.6. Peripheral Interrupt Flag Status - Bridge A

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

These flags are set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGx bit, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.

Writing a '0' to these bits has no effect.

Writing a '1' to these bits clears the corresponding INTFLAGx 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 |
|---|---|---|---|---|---|---|---|---|
| | TCC2 | TCC1 | TCC0 | TC7 | TC6 | TC5 | TC4 | TC3 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | TC2 | TC1 | TC0 | SERCOM1 | SERCOM0 | EIC | FREQM | PAC |
| Access | RW | RW | RW | RW | RW | RW | RW | RW |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – TCC2** Interrupt Flag for TCC2
This bit is set when a Peripheral Access Error occurs while accessing the TCC2, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 14 – TCC1** Interrupt Flag for TCC1
This bit is set when a Peripheral Access Error occurs while accessing the TCC1, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 13 – TCC0** Interrupt Flag for TCC0
This bit is set when a Peripheral Access Error occurs while accessing the TCC0, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 12 – TC7** Interrupt Flag for TC7
This bit is set when a Peripheral Access Error occurs while accessing the TC7, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 11 – TC6** Interrupt Flag for TC6
This bit is set when a Peripheral Access Error occurs while accessing the TC6, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 10 – TC5** Interrupt Flag for TC5
This bit is set when a Peripheral Access Error occurs while accessing the TC5, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 9 – TC4** Interrupt Flag for TC4
This bit is set when a Peripheral Access Error occurs while accessing the TC4, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 8 – TC3** Interrupt Flag for TC3
This bit is set when a Peripheral Access Error occurs while accessing the TC3, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 7 – TC2** Interrupt Flag for TC2
This bit is set when a Peripheral Access Error occurs while accessing the TC2, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 6 – TC1** Interrupt Flag for TC1
This bit is set when a Peripheral Access Error occurs while accessing the TC1, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 5 – TC0** Interrupt Flag for TC0
This bit is set when a Peripheral Access Error occurs while accessing the TC0, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 4 – SERCOM1** Interrupt Flag for SERCOM1
This bit is set when a Peripheral Access Error occurs while accessing the SERCOM1, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 3 – SERCOM0** Interrupt Flag for SERCOM0
This bit is set when a Peripheral Access Error occurs while accessing the SERCOM0, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 2 – EIC** Interrupt Flag for EIC
This bit is set when a Peripheral Access Error occurs while accessing the EIC, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 1 – FREQM** Interrupt Flag for FREQM
This bit is set when a Peripheral Access Error occurs while accessing the FREQM, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 0 – PAC** Interrupt Flag for PAC
This bit is set when a Peripheral Access Error occurs while accessing the PAC, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.7. Peripheral Interrupt Flag Status – Bridge B

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

These flags are set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGx bit, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.

Writing a '0' to these bits has no effect.

Writing a '1' to these bits clears the corresponding INTFLAGx 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 |
|---|---|---|---|---|---|---|---|---|
| | | | ETH | RAMECC | EVSYS | DMAC | | DSU |
| Access | | | RW | RW | RW | RW | | RW |
| Reset | | | 0 | 0 | 0 | 0 | | 0 |

**Bit 5 – ETH** Interrupt flag for ETH
This flag is set when a Peripheral Access Error occurs while accessing the Ethernet, 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 clears the ETH interrupt flag.

**Bit 4 – RAMECC** Interrupt Flag for RAMECC
This flag is set when a Peripheral Access Error occurs while accessing the RAMECC, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the RAMECC interrupt flag.

**Bit 3 – EVSYS** Interrupt Flag for EVSYS
This flag is set when a Peripheral Access Error occurs while accessing the EVSYS, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the EVSYS interrupt flag.

**Bit 2 – DMAC** Interrupt Flag for DMAC
This flag is set when a Peripheral Access Error occurs while accessing the DMAC, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the DMAC interrupt flag.

**Bit 0 – DSU** Interrupt Flag for DSU

This flag is set when a Peripheral Access Error occurs while accessing the DSU, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the DSU interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.8. Peripheral Interrupt Flag Status - Bridge C

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

These flags are set when a Peripheral Access Error occurs while accessing the peripheral associated with the respective INTFLAGx bit, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.

Writing a '0' to these bits has no effect.

Writing a '1' to these bits clears the corresponding INTFLAGx 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | HMTX | |
| Access | | | | | | | RW | |
| Reset | | | | | | | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | AC | CCL | | | | | SERCOM6 | QSPI |
| Access | RW | RW | | | | | RW | RW |
| Reset | 0 | 0 | | | | | 0 | 0 |

**Bit 9 – HMTX** HMATRIX APB Protection Enable
This flag is set when a Peripheral Access Error occurs while accessing the HMATRIX, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the HMATRIX interrupt flag.

**Bit 7 – AC** Interrupt Flag for AC
This flag is set when a Peripheral Access Error occurs while the AC, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the AC interrupt flag.

**Bit 6 – CCL** Interrupt Flag for CCL
This flag is set when a Peripheral Access Error occurs while accessing the CCL, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the CCL interrupt flag.

**Bit 1 – SERCOM6** Interrupt Flag for SERCOM6
This flag is set when a Peripheral Access Error occurs while accessing the SERCOM6, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the SERCOM6 interrupt flag.

**Bit 0 – QSPI** Interrupt Flag for QSPI
This flag is set when a Peripheral Access Error occurs while accessing the QSPI, and an interrupt request is generated if INTENCLR/SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the QSPI interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.9. Peripheral Interrupt Flag Status - Bridge D

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

These flags are set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.

Writing a '0' to these bits has no effect.

Writing a '1' to these bits clears the corresponding INTFLAGx 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | CAN1 | CAN0 |
| Access | | | | | | | RW | RW |
| Reset | | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TC9 | TC8 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 |
| Access | | | RW | RW | RW | RW | RW | RW |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 9 – CAN1** Interrupt Flag for CAN1
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 8 – CAN0** Interrupt Flag for CAN0
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 5 – TC9** Interrupt Flag for TC9
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 4 – TC8** Interrupt Flag for TC8
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.


© 2025 Microchip Technology Inc. and its subsidiaries

Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 3 – SERCOM5** Interrupt Flag for SERCOM5
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 2 – SERCOM4** Interrupt Flag for SERCOM4
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 1 – SERCOM3** Interrupt Flag for SERCOM3
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.

**Bit 0 – SERCOM2** Interrupt Flag for SERCOM2
This flag is set when a Peripheral Access Error occurs while accessing the peripheral, and an interrupt request is generated if SET.ERR is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.10. Peripheral Write Protection Status A

**Name:** STATUSA
**Offset:** 0x34
**Reset:** 0x00000000
**Property:** PAC Write-Protection

Writing to this register has no effect.

Reading the STATUS register returns the 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 |
|---|---|---|---|---|---|---|---|---|
| | TCC2 | TCC1 | TCC0 | TC7 | TC6 | TC5 | TC4 | TC3 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | TC2 | TC1 | TC0 | SERCOM1 | SERCOM0 | EIC | FREQM | PAC |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 15 – TCC2** TCC2 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TCC2 peripheral is not write protected |
| 1 | TCC2 peripheral is write protected |

**Bit 14 – TCC1** TCC1 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TCC1 peripheral is not write protected |
| 1 | TCC1 peripheral is write protected |

**Bit 13 – TCC0** TCC0 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TCC0 peripheral is not write protected |
| 1 | TCC0 peripheral is write protected |

**Bit 12 – TC7** TC7 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC7 peripheral is not write protected |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | TC7 peripheral is write protected |

**Bit 11 – TC6** TC6 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC6 peripheral is not write protected |
| 1 | TC6 peripheral is write protected |

**Bit 10 – TC5** TC5 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC5 peripheral is not write protected |
| 1 | TC5 peripheral is write protected |

**Bit 9 – TC4** TC4 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC4 peripheral is not write protected |
| 1 | TC4 peripheral is write protected |

**Bit 8 – TC3** TC3 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC3 peripheral is not write protected |
| 1 | TC3 peripheral is write protected |

**Bit 7 – TC2** TC2 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC2 peripheral is not write protected |
| 1 | TC2 peripheral is write protected |

**Bit 6 – TC1** TC1 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC1 peripheral is not write protected |
| 1 | TC1 peripheral is write protected |

**Bit 5 – TC0** TC0 APB Protect Enable

| Value | Description |
|---|---|
| 0 | TC0 peripheral is not write protected |
| 1 | TC0 peripheral is write protected |

**Bit 4 – SERCOM1** SERCOM1 APB Protect Enable

| Value | Description |
|---|---|
| 0 | SERCOM1 peripheral is not write protected |
| 1 | SERCOM1 peripheral is write protected |

**Bit 3 – SERCOM0** SERCOM0 APB Protect Enable

| Value | Description |
|---|---|
| 0 | SERCOM0 peripheral is not write protected |
| 1 | SERCOM0 peripheral is write protected |

**Bit 2 – EIC** EIC APB Protect Enable


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | EIC peripheral is not write protected |
| 1 | EIC peripheral is write protected |

**Bit 1 – FREQM** FREQM APB Protect Enable

| Value | Description |
|---|---|
| 0 | FREQM peripheral is not write protected |
| 1 | FREQM peripheral is write protected |

**Bit 0 – PAC** PAC APB Protect Enable

| Value | Description |
|---|---|
| 0 | PAC peripheral is not write protected |
| 1 | PAC peripheral is write protected |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.11. Peripheral Write Protection Status - Bridge B

**Name:** STATUSB
**Offset:** 0x38
**Reset:** 0x00000000
**Property:** PAC Write-Protection

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 |
|---|---|---|---|---|---|---|---|---|
| | | | ETH | RAMECC | EVSYS | DMAC | | DSU |
| Access | | | RW | R | R | R | | R |
| Reset | | | 0 | 0 | 0 | 0 | | 0 |

**Bit 5 – ETH** Interrupt flag for ETH
This flag is set when a Peripheral Access Error occurs while accessing the Ethernet, 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 clears the ETH interrupt flag.

**Bit 4 – RAMECC** RAMECC APB Protect Enable

| Value | Description |
|---|---|
| 0 | RAMECC peripheral is not write protected |
| 1 | RAMECC peripheral is write protected |

**Bit 3 – EVSYS** EVSYS APB Protect Enable

| Value | Description |
|---|---|
| 0 | EVSYS peripheral is not write protected |
| 1 | EVSYS peripheral is write protected |

**Bit 2 – DMAC** DMAC APB Protect Enable


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | DMAC peripheral is not write protected |
| 1 | DMAC peripheral is write protected |

**Bit 0 – DSU** DSU APB Protect Enable

| Value | Description |
|---|---|
| 0 | DSU peripheral is not write protected |
| 1 | DSU peripheral is write protected |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.12. Peripheral Write Protection Status - Bridge C

**Name:** STATUSC
**Offset:** 0x3C
**Reset:** 0x00000000
**Property:** PAC Write-Protection

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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | HMTX | |
| Access | | | | | | | R | |
| Reset | | | | | | | 0 | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | AC | CCL | | | | | SERCOM6 | QSPI |
| Access | R | R | | | | | R | R |
| Reset | 0 | 0 | | | | | 0 | 0 |

**Bit 9 – HMTX** HMATRIX APB Protection Enable

| Value | Description |
|---|---|
| 0 | HMATRIX APB is not write protected |
| 1 | HMATRIX APB is write protected |

**Bit 7 – AC** AC APB Protection Enable

| Value | Description |
|---|---|
| 0 | AC peripheral is not write protected |
| 1 | AC peripheral is write protected |

**Bit 6 – CCL** CCL APB Protection Enable

| Value | Description |
|---|---|
| 0 | CCL peripheral is not write protected |
| 1 | CCL peripheral is write protected |

**Bit 1 – SERCOM6** SERCOM6 APB Protection Enable

| Value | Description |
|---|---|
| 0 | SERCOM6 peripheral is not write protected |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | SERCOM6 peripheral is write protected |

**Bit 0 – QSPI** QSPI APB Protection Enable

| Value | Description |
|---|---|
| 0 | QSPI peripheral is not write protected |
| 1 | QSPI peripheral is write protected |


© 2025 Microchip Technology Inc. and its subsidiaries

### 24.7.13. Peripheral Write Protection Status - Bridge D

**Name:** STATUSD
**Offset:** 0x40
**Reset:** 0x00000000
**Property:** PAC Write-Protection

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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | CAN1 | CAN0 |
| Access | | | | | | | R | R |
| Reset | | | | | | | 0 | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | TC9 | TC8 | SERCOM5 | SERCOM4 | SERCOM3 | SERCOM2 |
| Access | | | R | R | R | R | R | R |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bit 9 – CAN1** CAN1 APB Protection Enable

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

**Bit 8 – CAN0** CAN0 APB Protection Enable

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

**Bit 5 – TC9** TC9 APB Protection Enable

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

**Bit 4 – TC8** TC8 APB Protection Enable

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


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Peripheral is write protected |

**Bit 3 – SERCOM5** SERCOM5 APB Protection Enable

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

**Bit 2 – SERCOM4** SERCOM4 APB Protection Enable

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

**Bit 1 – SERCOM3** SERCOM3 APB Protection Enable

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

**Bit 0 – SERCOM2** SERCOM2 APB Protection Enable

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


© 2025 Microchip Technology Inc. and its subsidiaries

# 25. Real-Time Counter and Calendar (RTCC)

## 25.1. Overview

The Real-Time Counter (RTCC) is a 32-bit counter with a 10-bit programmable prescaler that typically runs continuously to keep track of time. The RTCC can wake up the device from sleep modes using the alarm/compare wake-up, periodic wake-up, overflow wake-up mechanisms or from the wake inputs.

The RTCC 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 30.5 µs, 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.

## 25.2. Features

- 32-Bit Counter with 10-Bit Prescaler
- Multiple Clock Sources
- 32-Bit or 16-Bit Counter Mode
- Two 32-Bit or Four 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
- Four General Purpose Registers
- One Backup Register with Retention Capability
- Tamper Detection
  - Timestamp on event or up to four inputs with debouncing
  - Active layer protection


© 2025 Microchip Technology Inc. and its subsidiaries

## 25.3. Block Diagram

**Figure 25-1.** RTCC Block Diagram (Mode 0 — 32-Bit Counter)

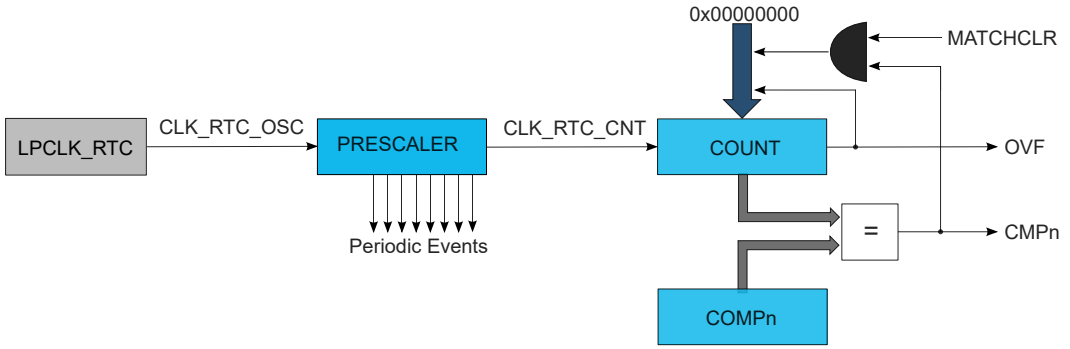



**Figure 25-2.** RTCC Block Diagram (Mode 1 — 16-Bit Counter)

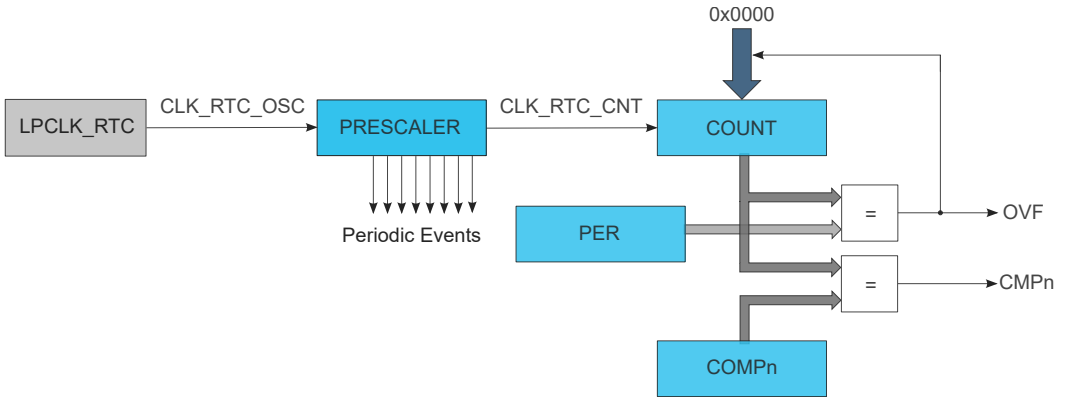
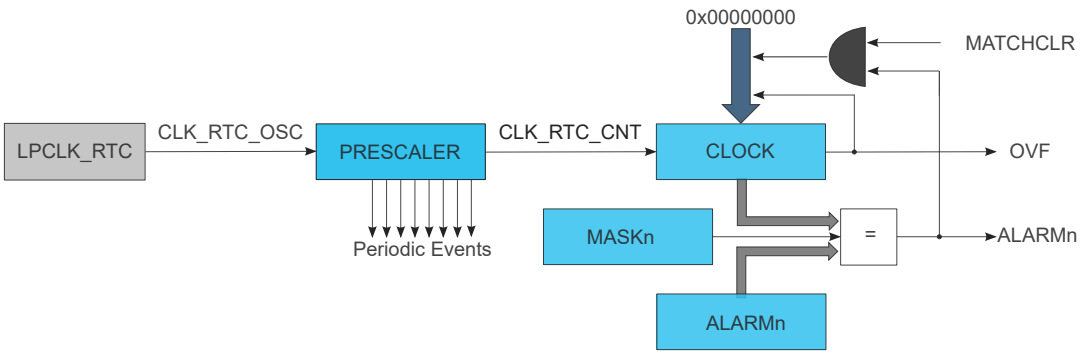


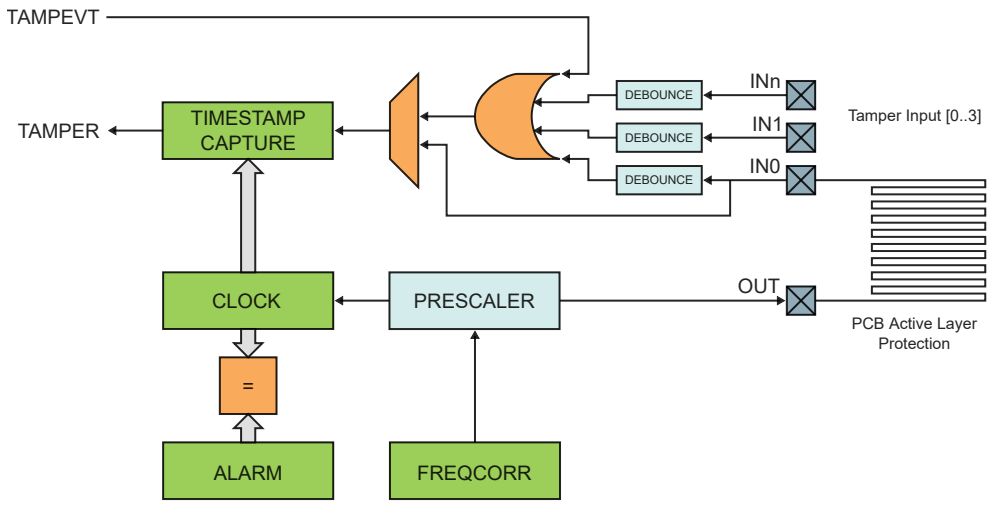

**Figure 25-3.** RTCC Block Diagram (Mode 2 — Clock/Calendar)





© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 25-4.** RTCC Block Diagram (Tamper Detection)




## 25.4. Signal Description

**Table 25-1.** Signal Description

| Signal | Description | Type |
|---|---|---|
| INn [n=0..3] | Tamper detection input | Digital input |
| OUT | Tamper detection output | Digital output |
| RTC_EVENT | RTC event output | Digital output |

## 25.5. Product Dependencies

The following sections describe how the other parts of the system must be configured correctly to use this peripheral.

### 25.5.1. I/O Lines

In order to use the I/O lines of this peripheral, the RTC must be enabled and no higher priority peripherals for the RTC pins can be enabled. See *I/O Ports and Peripheral Pin Select (PPS)* from Related Links.

**Related Links**

I/O Ports and Peripheral Pin Select (PPS)

### 25.5.2. Power Management

The RTC continues to operate in any sleep modes (Standby Sleep, Deep Sleep, Idle) 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 the sleep modes. See *Power Management Unit (PMU)* from Related Links for details on the different Sleep modes.

The RTCC 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).

**Related Links**

Power Management Unit (PMU)


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.5.3. Clocks

A 32 KHz or 1 KHz oscillator clock (CLK_RTC_OSC) is required to clock the RTC. The 32 KHz clock source can be FRC, POSC, SOSC or LPRC based on the mux selection controlled by the CFGCON4.VBKP_32KCSEL bit. The 1 KHz clock source is based on the mux selection controlled by the CFGCON4.VBKP_1KCSEL bit.

This oscillator clock is asynchronous to the bus clock (PB3_CLK). Due to this asynchronicity, writing to certain registers will require synchronization between the clock domains.

### 25.5.4. DMA

The DMA request lines (or line if only one request) are connected to the DMA Controller (DMAC). Using the RTC DMA requests requires the DMA Controller to be configured first. See *Direct Memory Access Controller (DMAC)* from Related Links.

**Related Links**

Direct Memory Access Controller (DMAC)

### 25.5.5. Interrupts

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

### 25.5.6. Events

The events are connected to the *Event System*. See *Event System (EVSYS)* from Related Links.

**Related Links**

Event System (EVSYS)

### 25.5.7. Debug Operation

When the CPU is halted in Debug mode, the RTC halts normal operation. The RTC can be forced to continue operation during debugging. See *DBGCTRL* from Related Links.

**Related Links**

DBGCTRL

Debug Control

### 25.5.8. Register Access Protection

All registers with write access are optionally write protected by the PAC, except the 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. See *Peripheral Access Controller (PAC)* from Related Links.

**Related Links**

Peripheral Access Controller (PAC)

## 25.6. Functional Description

### 25.6.1. 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


© 2025 Microchip Technology Inc. and its subsidiaries

- Mode 2 - CLOCK: RTC serves as clock/calendar with alarm functionality

## 25.6.2. Basic Operation

### 25.6.2.1. 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)
- BKUP registers Reset On Tamper bit in Control A register (CTRLA.BKTRST)
- GP registers Reset On Tamper Enable in Control A register (CTRLA.GPTRST)

The following registers are enable-protected:

- Control B register (CTRLB)
- Event Control register (EVCTRL)
- Tamper Control register (TAMPCTRL)

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, then, check whether the write synchronization has finished, then, change the desired bit field value. 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.

The RTC prescaler divides the source clock for the RTC counter.

**Note:** In the 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 derived by the following formula:

$$f_{\mathrm{CLK\_RTC\_CNT}} = \frac{f_{\mathrm{CLK\_RTC\_OSC}}}{2^{\mathrm{PRESCALER}}}$$

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.

### 25.6.2.2. 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.

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.

### 25.6.2.3. 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. See the *RTCC Block Diagram (Mode 0-32-Bit Counter)* figure in the *Block Diagram* from Related Links. When the RTC is enabled, the counter is incremented on every 0-to-1 transition of CLK_RTC_CNT. The counter increments until it reaches the top value of 0xFFFFFFFF, then wraps to 0x00000000. This sets the Overflow Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF).


© 2025 Microchip Technology Inc. and its subsidiaries

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 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) 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 COMPn 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.CMPn and INTFLAG.OVF will both be set simultaneously on a compare match with COMPn.

**Related Links**
Block Diagram

#### 25.6.2.4. 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. See *RTCC Block Diagram (Mode 1 – 16-Bit Counter)* figure in the *Block Diagram* from Related Links. 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, 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..). When a compare match occurs, the Compare n Interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.CMPn, n=0..) is set on the next 0-to-1 transition of CLK_RTC_CNT.

**Related Links**
Block Diagram

#### 25.6.2.5. 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. See the *RTCC Block Diagram (Mode 2 — Clock/Calendar)* figure in the *Block Diagram* from Related Links. When the RTC is enabled, the counter increments on every 0-to-1 transition of CLK_RTC_CNT. The selected clock source and RTC prescaler must be configured to provide a 1 Hz 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)
- Month as the numeric month of the year (1 = January, 2 = February and so on)
- 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.


© 2025 Microchip Technology Inc. and its subsidiaries

The RTC increments until it reaches the top value of 23:59:59 December 31 of year value 0x3F, then wraps to 00:00:00 January 1 of year value 0x00. This sets 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 registers (ALARMn, n=0–1). When an alarm match occurs, the Alarm n Interrupt flag in the Interrupt Flag Status and Clear registers (INTFLAG.ALARMn, n=0..1) is set on the next 0-to-1 transition of CLK_RTC_CNT. For example, for a 1 Hz clock counter, it means the Alarm n Interrupt flag is set with a delay of 1s after the alarm match occurs.

A valid alarm match depends on the setting of the Alarm Mask Selection bits in the Alarm n Mask register (MASKn.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 ALARMn occurs. This allows the RTC to generate periodic interrupts or events with longer periods than would be possible with the prescaler events only (see *Periodic Intervals* from Related Links).
**Note:** When CTRLA.MATCHCLR is '1', INTFLAG.ALARMn and INTFLAG.OVF will both be set simultaneously on an alarm match with ALARMn.

**Related Links**
Block Diagram
Periodic Intervals

### 25.6.3. DMA Operation

The RTC generates the following DMA request:

- Tamper (TAMPER): The request is set on capture of the timestamp. The request is cleared when the Timestamp register is read.

If the CPU accesses the registers which are source for DMA request set/clear condition, the DMA request can be lost or the DMA transfer can be corrupted, if enabled.

### 25.6.4. Interrupts

The RTC has the following interrupt sources:

- Overflow (OVF) – Indicates that the counter has reached its top value and wrapped to zero
- Tamper (TAMPER) – Indicates detection of valid signal on a tamper input pin or tamper event input
- Compare (CMPn) – Indicates a match between the counter value and the compare register
- Alarm (ALARMn) – Indicates a match between the clock value and the alarm register
- Period n (PERn) – The corresponding bit in the prescaler has toggled, see *Periodic Intervals* from Related Links

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.


© 2025 Microchip Technology Inc. and its subsidiaries

All interrupt requests from the peripheral are ORed together on system level to generate one combined interrupt request to the NVIC, see *Nested Vector Interrupt Controller (NVIC)* from Related Links. 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, see *Nested Vector Interrupt Controller (NVIC)* from Related Links.

**Related Links**

Nested Vector Interrupt Controller (NVIC)
Periodic Intervals

### 25.6.5. Events

The RTC can generate the following output events and can be used by the EVSYS module:

- Overflow (OVF) – Generated when the counter has reached its top value and wrapped to zero
- Tamper (TAMPER): Generated on detection of a valid signal on a tamper input pin or tamper event input
- Compare (CMPn) – Indicates a match between the counter value and the compare register
- Alarm (ALARMn) – Indicates a match between the clock value and the alarm register
- Period n (PERn) – The corresponding bit in the prescaler has toggled, see *Periodic Intervals* from Related Links
- Periodic Daily (PERD) – Generated when the COUNT/CLOCK has incremented at a fixed period of time
- RTC Event (RTC_EVENT) – Generates specific external signal on the RTC EVENT I/O pin

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. See *Event System (EVSYS)* from Related Links for more details on configuring the event system.

The RTC can take the following actions on an input event:

- Tamper (TAMPEVT) – Capture the RTC counter to the timestamp register. See *Tamper Detection* from Related Links.

Writing a one to an Event Input bit into the Event Control register (EVCTRL.xxxEI) enables the corresponding action on input event. Writing a zero to this bit disables the corresponding action on input event.

RTC Event (RTC_EVENT) – Other than the above events, which are mapped to the EVSYS module, the following events can generate a specific external signal on the RTC EVENT I/O pin:

- 32 KHz clock
- Alarm pulse
- 1-second clock

These event signals are configured using CFGCON4.RTCEVENTSEL[1:0] bits.

**Note:** The RTC_OUT and RTC_EVENT signals are multiplexed and any one of the signals can be out at a time in pin-limited variants. The selection between RTC_OUT and RTC_EVENT is configurable through the CFGCON4.RTCEVTYPE bit.

**Related Links**

Periodic Intervals
Tamper Detection
Event System (EVSYS)


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.6.6. Sleep Mode Operation

The RTC continues to operate in any Sleep modes (Standby Sleep, Deep Sleep) 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 is generated after the wake-up if the NVIC interrupt controller is configured accordingly. Otherwise, the CPU will wake up directly, without triggering any interrupt. In this case, the CPU continues executing right from the first instruction that followed the entry into sleep.

The periodic events can also wake up the CPU through the interrupt function of the Event System. In this case, the event must be enabled and connected to an event channel with its interrupt enabled. See *Event System (EVSYS)* from Related Links.

**Related Links**

Event System (EVSYS)

### 25.6.7. 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
- Count Read Synchronization bit in Control A register (CTRLA.COUNTSYNC)
- Clock Read Synchronization bit in Control A register (CTRLA.COUNTSYNC)

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 General Purpose n registers (GPn)

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'
- The Timestamp Value register (TIMESTAMP)

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.


© 2025 Microchip Technology Inc. and its subsidiaries

## 25.6.8. Additional Features

### 25.6.8.1. Periodic Intervals

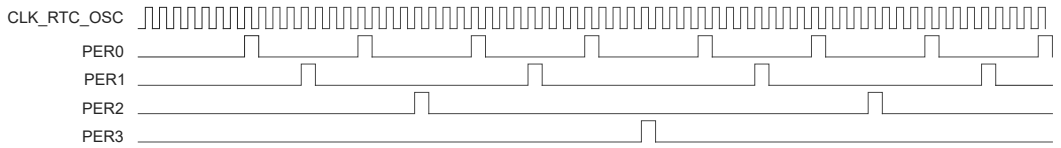
The RTC prescaler can generate interrupts and events at periodic intervals, allowing flexible system tick creation. Any of the upper eight bits of the prescaler (bits 2 to 9) can be the source of an interrupt/event. When one of the eight Periodic Event Output bits in the Event Control register (EVCTRL.PEREO[n=0..7]) 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:

$$f_{\mathrm{PERIODIC(n)}} = \frac{f_{\mathrm{CLK\_RTC\_OSC}}}{2^{n+3}}$$

$f_{CLK\_RTC\_OSC}$ is the frequency of the internal prescaler clock CLK_RTC_OSC, and n is the position of the EVCTRL.PEREOn bit. For example, PER0 generates an event every eight CLK_RTC_OSC cycles, PER1 every 16 cycles and so on. This is illustrated in the following figure.

Periodic events are independent of the prescaler setting used by the RTC counter, except if CTRLA.PRESCALER is '0'. Then, no periodic events are generated.

**Figure 25-5.** Example Periodic Events




**Note:** This example also applies to interrupts. Just replace EVCTRL.PEREOn with the PERn fields of INTENCLR, INTENSET and INTFLAG. For Modes 0 and 2, n = 0,..7. For Mode 1 n = 2...7.

### 25.6.8.2. 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 1 ppm steps. Digital correction is achieved by adding or skipping a single count in the prescaler once every 8192 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 128 of these periods. The resulting correction is as follows:

$$\text{Correction in ppm} = \frac{\text{FREQCORR.VALUE}}{8192 \cdot 128} \cdot 10^6 \text{ppm}$$

This results in a resolution of 0.95367 ppm.

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 can also be shortened or lengthened depending on the correction value.

### 25.6.8.3. Backup Registers

The RTC includes one Backup register (BKUP0). This register maintains its content in the Deep Sleep mode. It is used to store user-defined values.

Use General Purpose registers (GPn) if the stored user-defined data are more than what the Backup register can hold.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.6.8.4. General Purpose Registers

The RTC includes four General Purpose registers (GPn). These registers are reset only when the RTC is reset or when tamper detection occurs while CTRLA.GPTRST=1 and remain powered while the RTC is powered. They can be used to store user-defined values while other parts of the system are powered off.

The general purpose registers 2*n and 2*n+1 are enabled by writing a '1' to the General Purpose Enable bit n in the Control B register (CTRLB.GPnEN).

The GP registers share internal resources with the COMPARE/ALARM features. Each COMPARE/ALARM register has a separate read buffer and write buffer. When the general purpose feature is enabled, the even GP uses the read buffer while the odd GP uses the write buffer.

When the the COMPARE/ALARM register is written, the write buffer temporarily holds the COMPARE/ALARM value until the synchronization is complete (bit SYNCBUSY.COMPn going to '0'). After the write is completed, the write buffer can be used as an odd general purpose register without affecting the COMPARE/ALARM function.

If the COMPARE/ALARM function is not used, the read buffer can be used as an even general purpose register. In this case, writing the even GP will temporarily use the write buffer until the synchronization is complete (bit SYNCBUSY.GPn going to '0'). Thus, an even GP must be written before writing the odd GP. Changing or writing an even GP needs to temporarily save the value of the odd GP.

Before using an even GP, the associated COMPARE/ALARM feature must be disabled by writing a '1' to the General Purpose Enable bit in the Control B register (CTRLB.GPnEN). To re-enable the compare/alarm, CTRLB.GPnEN must be written to zero and the associated COMPn/ALARMn must be written with the correct value.

It is recommended to use the Backup register (BKUPn) first to store user-defined values, and use the GPn only when the user-defined values exceed the capacity of the provided BKUPn.

An example procedure to write the general purpose registers GP0 and GP1 is:

1. Wait for any ongoing write to COMP0 to complete (SYNCBUSY.COMP0=0). If the RTC is operating in Mode 1, wait for any ongoing write to COMP1 to complete as well (SYNCBUSY.COMP1=0).
2. Write CTRLB.GP0EN=1 if GP0 is needed.
3. Write GP0 if needed.
4. Wait for any ongoing write to GP0 to complete (SYNCBUSY.GP0=0).
   **Note:** GP1 will also show as busy when GP0 is busy.
5. Write GP1 if needed.

The following table provides the correspondence of General Purpose Registers and the COMPARE/ALARM read or write buffer in all RTC modes.

**Table 25-2.** General Purpose Registers Versus Compare/Alarm Registers: n in 0, 2, 4, 6...

| Register | Mode 0 | Mode 1 | Mode 2 | Write Before |
|---|---|---|---|---|
| GPn | COMPn/2 write buffer | (COMPn , COMPn+1) write buffer | ALARMn/2 write buffer | GPn+1 |
| GPn+1 | COMPn/2 read buffer | (COMPn , COMPn+1) read buffer | ALARMn/2 read buffer | — |

### 25.6.8.5. Tamper Detection

The RTC provides four tamper channels that can be used for tamper detection.

The action of each tamper channel is configured using the Input n Action bits in the Tamper Control register (TAMPCTRL.INnACT):


© 2025 Microchip Technology Inc. and its subsidiaries

- Off – Detection for tamper channel n is disabled.
- Wake – A transition on INn input (tamper channel n) matching TAMPCTRL.TAMPLVLn will be detected and the tamper interrupt flag (INTFLAG.TAMPER) will be set. The RTC value is not captured in the TIMESTAMP register.
- Capture – A transition on INn input (tamper channel n) matching TAMPCTRL.TAMPLVLn is detected and the tamper interrupt flag (INTFLAG.TAMPER) is set. The RTC value is captured in the TIMESTAMP register.
- Active Layer Protection – A mismatch of an internal RTC signal routed between INn and OUTn pins is detected, and the tamper interrupt flag (INTFLAG.TAMPER) is set. The RTC value is captured in the TIMESTAMP register.

To determine which tamper source caused a tamper event, the Tamper ID register (TAMPID) provides the detection status of each tamper channel. These bits remain active until cleared by software.

A single interrupt request (TAMPER) is available for all tamper channels.

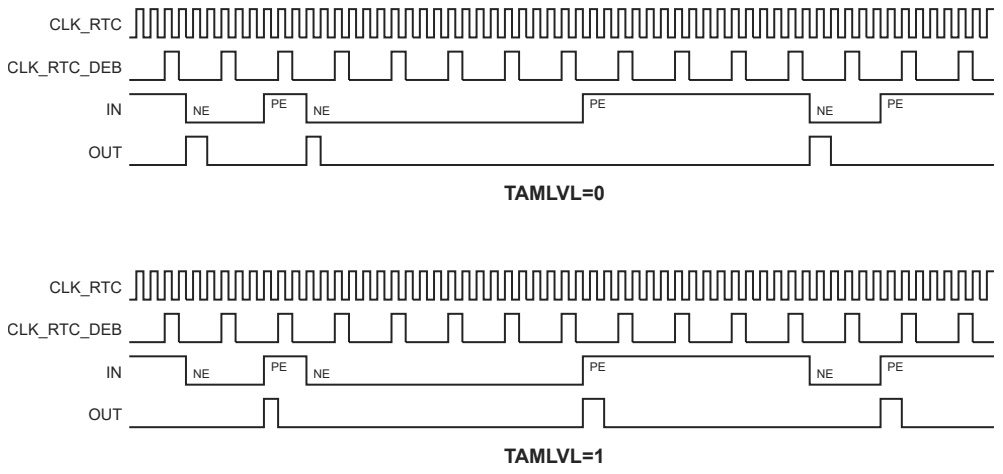
The RTC also supports an input event (TAMPEVT) for generating a tamper condition within the Event System. The tamper input event is enabled by the Tamper Input Event Enable bit in the Event Control register (EVCTRL.TAMPEVTEI).

Up to four polarity external inputs (INn) can be used for tamper detection. The polarity for each input is selected with the Tamper Level bits in the Tamper Control register (TAMPCTRL.TAMPLVLn).

Separate debouncers are embedded for each external input. The debouncer for each input is enabled/disabled with the Debounce Enable bits in the Tamper Control register (TAMPCTRL.DEBNCn). The debouncer configuration is fixed for all inputs as set by the Control B register (CTRLB). The debouncing period duration is configurable using the Debounce Frequency field in the Control B register (CTRLB.DEBF). The period is set for all debouncers. (In other words, the duration cannot be adjusted separately for each debouncer.)

When TAMPCTRL.DEBNCn = 0, INn is detected asynchronously. The following figure illustrates an example.

**Figure 25-6.** Edge Detection with Debouncer Disabled

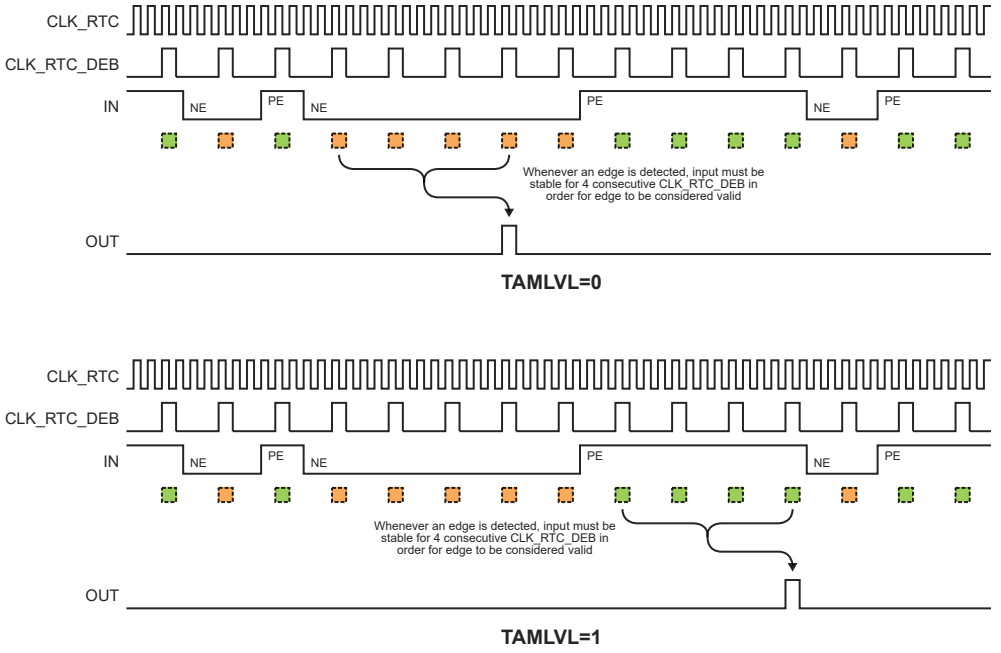



When TAMPCTRL.DEBNCn = 1, the detection time depends on whether the debouncer operates synchronously or asynchronously and whether majority detection is enabled or not. For more details, refer to Table 25-3. Synchronous versus asynchronous stability debouncing is configured by the Debounce Asynchronous Enable bit in the Control B register (CTRLB.DEBASYNC):


© 2025 Microchip Technology Inc. and its subsidiaries

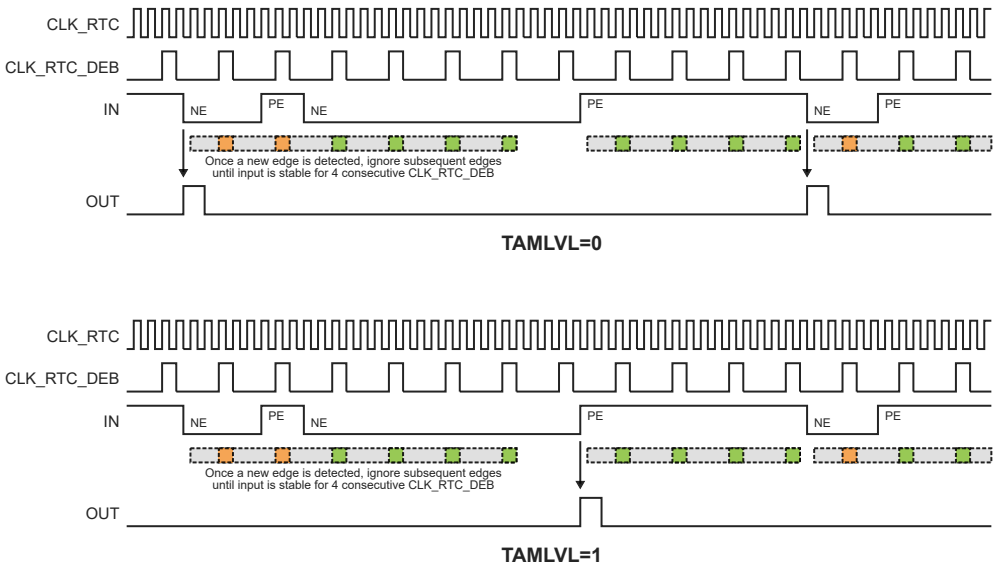
- Synchronous (CTRLB.DEBASYNC = 0): INn is synchronized in two CLK_RTC periods, then must remain stable for four CLK_RTC_DEB periods before a valid detection occurs. The following figure illustrates an example.

**Figure 25-7.** Edge Detection with Synchronous Stability Debouncing




- Asynchronous (CTRLB.DEBASYNC = 1): The first edge on INn is detected. Further detection is blanked until INn remains stable for four CLK_RTC_DEB periods. The following figure illustrates an example.

**Figure 25-8.** Edge Detection with Asynchronous Stability Debouncing




Majority debouncing is configured by the Debounce Majority Enable bit in the Control B register (CTRLB.DEBMAJ). INn must be valid for two out of three CLK_RTC_DEB periods. The following figure illustrates an example.


© 2025 Microchip Technology Inc. and its subsidiaries

**Figure 25-9.** Edge Detection with Majority Debouncing

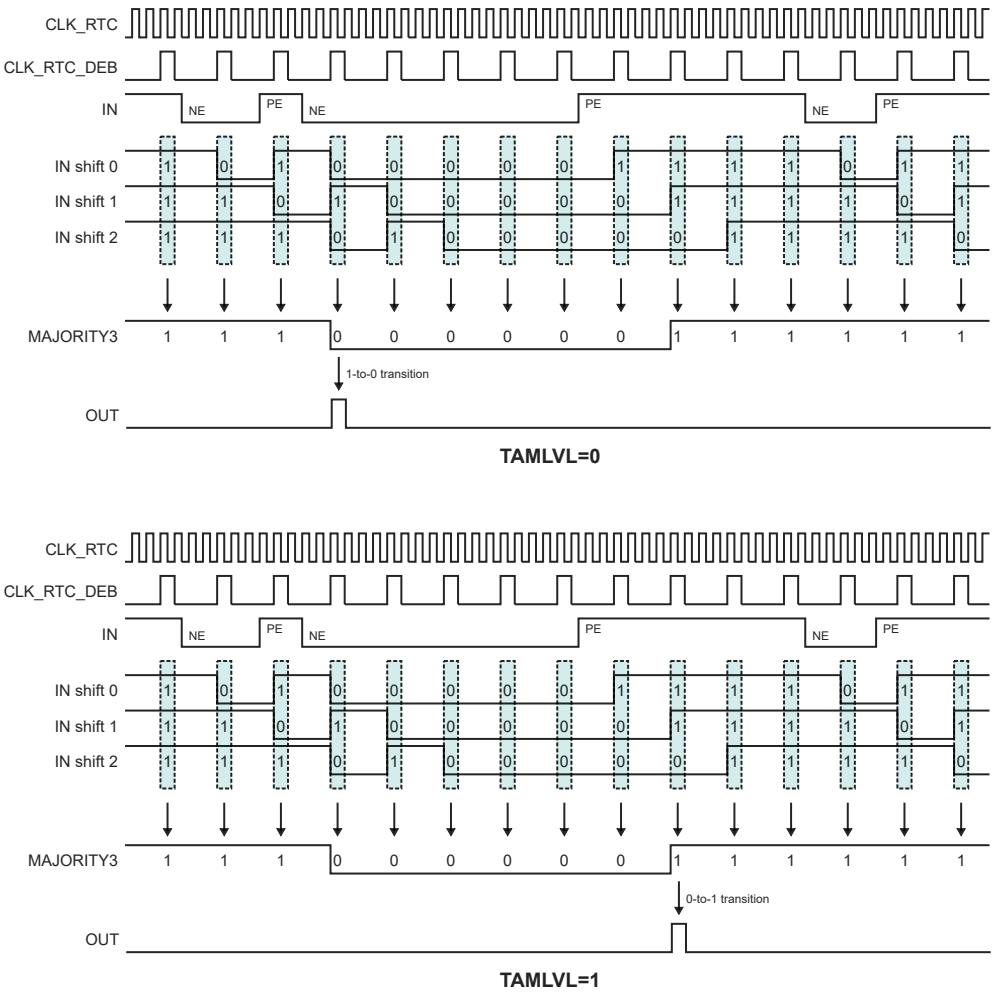



**Table 25-3.** Debouncer Configuration

| TAMPCTRL. DEBNCn | CTRLB. DEBMAJ | CTRLB. DEBASYNC | Description |
|---|---|---|---|
| 0 | X | X | Detect edge on INn with no debouncing. Every edge detected is immediately triggered. |
| 1 | 0 | 0 | Detect edge on INn with synchronous stability debouncing. Edge detected is only triggered when INn is stable for four consecutive CLK_RTC_DEB periods. |
| 1 | 0 | 1 | Detect edge on INn with asynchronous stability debouncing. First detected edge is triggered immediately. All subsequent detected edges are ignored until INn is stable for four consecutive CLK_RTC_DEB periods. |
| 1 | 1 | X | Detect edge on INn with majority debouncing. Pin INn is sampled for three consecutive CLK_RTC_DEB periods. Signal level is determined by majority-rule (LLL, LLH, LHL, HLL = 0 and LHH, HLH, HHL, HHH = 1). |

### 25.6.8.5.1. Timestamp

As part of tamper detection, the RTC can capture the counter value (COUNT/CLOCK) into the TIMESTAMP register. Three CLK_RTC periods are required to detect the tampering condition and capture the value. The TIMESTAMP value can be read once the Tamper flag in the Interrupt Flag


© 2025 Microchip Technology Inc. and its subsidiaries

register (INTFLAG.TAMPER) is set. If the DMA Enable bit in the Control B register (CTRLB.DMAEN) is '1', a DMA request is triggered by the timestamp. To determine which tamper source caused a capture, the Tamper ID register (TAMPID) provides the detection status of each tamper channel and the tamper input event. A DMA transfer can then read both TIMESTAMP and TAMPID in succession.

A new timestamp value cannot be captured until the Tamper flag is cleared, either by reading the timestamp or by writing a '1' to INTFLAG.TAMPER. If several tamper conditions occur in a short window before the flag is cleared, only the first timestamp may be logged. However, the detection of each tamper is still recorded in TAMPID.

The Tamper Input Event (TAMPEVT) is always performing a timestamp capture. To capture on the external inputs (INn), the corresponding Input Action field in the Tamper Control register (TAMPCTRL.INnACT) must be written to '1'. If an input is set for wake functionality, it does not capture the timestamp; however, the Tamper flag and TAMPID is still updated.
**Note:** The TIMESTAMP value must be read once, and INTFLAG.TAMPER must be cleared. The next value must be read only after the INTFLAG.TAMPER is set again.

#### 25.6.8.5.2. Active Layer Protection

The RTC provides a mean of detecting broken traces on the PCB, also known as Active layer Protection. In this mode, a generated internal RTC signal can be directly routed over critical components on the board using the RTC_OUT output pin to one RTC INn input pin. A tamper condition is detected if there is a mismatch on the generated RTC signal.

The Active Layer Protection mode and the generation of the RTC signal is enabled by setting the RTCOUT bit in the Control B register (CTRLB.RTCOUT).

**Note:** The Active Layer Protection works with one output pin (RTC_OUT) and multiple input pin INn. This is achieved by clearing the Separate Tamper Output bit CTRLB.SEPTO.

Enabling active layer protection requires the following steps:

- Enable the RTC prescaler output by writing a '1' to the RTC Out bit in the Control B register (CTRLB.RTCOUT). The I/O pins must also be configured to correctly route the signal to the external pins.
- Select the frequency of the output signal by configuring the RTC Active Layer Frequency field in the Control B register (CTRLB.ACTF).

  $$\text{CLK\_RTC\_OUT} = \frac{\text{CLK\_RTC}}{2^{\text{CTRLB.ACTF}+1}}$$

- Enable the tamper input n (INn) in Active Layer mode by writing '3' to the corresponding Input Action field in the Tamper Control register (TAMPCTRL.INnACT). When active layer protection is enabled and INn and OUTn pin are used, the value of INn is sampled on the falling edge of CLK_RTC and compared to the expected value of OUTn. Therefore up to one half of a CLK_RTC period is available for propagation delay through the trace.
- Enable Active Layer Protection by setting CTRLB.RTCOUT bit.


© 2025 Microchip Technology Inc. and its subsidiaries

## 25.7. Register Summary - Mode 0 - 32-Bit Counter

See the *RTCC* module in the *Product Memory Mapping Overview* from Related Links for the base address.

| 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 | GPTRST | BKTRST | | PRESCALER[3:0] | | | |
| 0x02 | CTRLB | 7:0 | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| | | 15:8 | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | TAMPEREO | | | | | CMPEOn[1:0] | |
| | | 23:16 | | | | | | | | TAMPEVEI |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | CMPn[3:0] | | | |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | CMPn[3:0] | | | |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | CMPn[3:0] | | | |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | COMP1 | COMP0 | | COUNT | FREQCORR | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | | | | |
| | | 23:16 | | | | | GP3 | GP2 | GP1 | GP0 |
| | | 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] | | | | | | | |
| 0x24 | COMP1 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |
| | | 23:16 | COMP[23:16] | | | | | | | |
| | | 31:24 | COMP[31:24] | | | | | | | |
| 0x28 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | GP0 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x44 | GP1 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x48 | GP2 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary - Mode 0 - 32-Bit Counter** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x4C | GP3 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x50 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | TAMPCTRL | 7:0 | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| | | 31:24 | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| 0x64 | TIMESTAMP | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| | | 23:16 | COUNT[23:16] | | | | | | | |
| | | 31:24 | COUNT[31:24] | | | | | | | |
| 0x68 | TAMPID | 7:0 | | | | | TAMPID3 | TAMPID2 | TAMPID1 | TAMPID0 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | TAMPEVT | | | | | | | |
| 0x6C ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | BKUP0 | 7:0 | BKUP[7:0] | | | | | | | |
| | | 15:8 | BKUP[15:8] | | | | | | | |
| | | 23:16 | BKUP[23:16] | | | | | | | |
| | | 31:24 | BKUP[31:24] | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 25.8. Register Description - Mode 0 - 32-Bit Counter

This Register Description section is valid if the RTC is in COUNT32 mode (CTRLA.MODE=0).


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.1. Control A in COUNT32 mode (CTRLA.MODE=0)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | GPTRST | BKTRST | | PRESCALER[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 | 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 prevents 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 |

**Bit 14 – GPTRST** GP Registers Reset On Tamper Enable
**Note:** Must disable this bit to avoid metastable reading of even GP registers.

Only GP registers enabled by the CTRLB.GPnEN bits are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

**Bit 13 – BKTRST** BKUP Registers Reset On Tamper Enable
All BKUPn registers are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | BKUPn registers will not reset when a tamper condition occurs. |
| 1 | BKUPn registers will reset when a tamper condition occurs. |

**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 |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 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 match |
| 1 | The counter is cleared on a Compare/Alarm match |

**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

**Note:** Access to registers/bits are disallowed until SYNCBUSY.ENABLE cleared by hardware.

Due to synchronization, there is a delay writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE reads back immediately, and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) is set. SYNCBUSY.ENABLE is 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 is disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write operation is discarded.
Due to synchronization, there is a delay writing CTRLA.SWRST until the Reset is complete.
CTRLA.SWRST is cleared when the Reset is complete.
**Notes:**

1. When the CTRLA.SWRST is written, the user must poll SYNCB.SWRST bit to determine when the reset operation is complete.
2. During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST is cleared by hardware.

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing |
| 1 | The Reset operation is ongoing |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.2. Control B in COUNT32 mode (CTRLA.MODE=0)

**Name:** CTRLB
**Offset:** 0x02
**Reset:** 0x0000
**Property:** Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| Access | R/W | R/W | R/W | R/W | | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

**Bits 14:12 – ACTF[2:0]** Active Layer Frequency
These bits define the prescaling factor for the RTC clock output (OUT) used during active layer protection in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_OUT = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_OUT = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_OUT = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_OUT = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_OUT = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_OUT = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_OUT = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_OUT = CLK_RTC / 256 |

**Bits 10:8 – DEBF[2:0]** Debounce Frequency
These bits define the prescaling factor for the input debouncers in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_DEB = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_DEB = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_DEB = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_DEB = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_DEB = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_DEB = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_DEB = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_DEB = CLK_RTC / 256 |

**Bit 7 – DMAEN** DMA Enable
The RTC can trigger a DMA request when the timestamp is ready in the TIMESTAMP register.

| Value | Description |
|---|---|
| 0 | Tamper DMA request is disabled. Reading TIMESTAMP has no effect on INTFLAG.TAMPER. |
| 1 | Tamper DMA request is enabled. Reading TIMESTAMP will clear INTFLAG.TAMPER. |

**Bit 6 – RTCOUT** RTC Output Enable

| Value | Description |
|---|---|
| 0 | The RTC active layer output is disabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The RTC active layer output is enabled. |

**Bit 5 – DEBASYNC** Debouncer Asynchronous Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers operate synchronously. |
| 1 | The tamper input debouncers operate asynchronously. |

**Bit 4 – DEBMAJ** Debouncer Majority Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers match three equal values. |
| 1 | The tamper input debouncers match majority two of three values. |

**Bit 1 – GP2EN** General Purpose 2 Enable

| Value | Description |
|---|---|
| 0 | COMP1 compare function enabled. GP2/GP3 disabled. |
| 1 | COMP1 compare function disabled. GP2/GP3 enabled. |

**Bit 0 – GP0EN** General Purpose 0 Enable

| Value | Description |
|---|---|
| 0 | COMP0 compare function enabled. GP0/GP1 disabled. |
| 1 | COMP0 compare function disabled. GP0/GP1 enabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.3. Event Control in COUNT32 mode (CTRLA.MODE=0)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** Enable-Protected

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | TAMPEVEI |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | TAMPEREO | | | | | CMPEOn[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 |
|---|---|---|---|---|---|---|---|---|
| | 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 16 – TAMPEVEI** Tamper Event Input Enable

| Value | Description |
|---|---|
| 0 | Tamper event input is disabled and incoming events will be ignored. |
| 1 | Tamper event input is enabled and incoming events will capture the COUNT value. |

**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 14 – TAMPEREO** Tamper Event Output Enable

| Value | Description |
|---|---|
| 0 | Tamper event output is disabled and will not be generated. |
| 1 | Tamper event output is enabled and will be generated for every tamper input. |

**Bits 9:8 – CMPEOn[1:0]** 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.4. Interrupt Enable Clear in COUNT32 mode (CTRLA.MODE=0)

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

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 | TAMPER | | | CMPn[3:0] | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 clears 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 14 – TAMPER** Tamper Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Tamper Interrupt Enable bit, which disables the Tamper interrupt.

| Value | Description |
|---|---|
| 0 | The Tamper interrupt is disabled. |
| 1 | The Tamper interrupt is enabled. |

**Bits 11:8 – CMPn[3:0]** Compare n Interrupt Enable [n = 3..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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 10:8 – CMPn[2:0]** Compare n Interrupt Enable [n = 2..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

**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 clears 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. |

**Bit 8 – CMPn** Compare 0 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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 clears 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.5. Interrupt Enable Set in COUNT32 mode (CTRLA.MODE=0)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** -

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 | TAMPER | | | CMPn[3:0] | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 clears 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 14 – TAMPER** Tamper Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Tamper Interrupt Enable bit, which disables the Tamper interrupt.

| Value | Description |
|---|---|
| 0 | The Tamper interrupt is disabled. |
| 1 | The Tamper interrupt is enabled. |

**Bits 11:8 – CMPn[3:0]** Compare n Interrupt Enable [n = 3..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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 10:8 – CMPn[2:0]** Compare n Interrupt Enable [n = 2..0]
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

**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 clears 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. |

**Bit 8 – CMPn** Compare 0 Interrupt Enable
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears 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 clears 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.6. 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 | TAMPER | | | CMPn[3:0] | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 is 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 14 – TAMPER** Tamper Event
This flag is set after a tamper condition occurs, and an interrupt request is generated if INTENCLR.TAMPER/INTENSET.TAMPER is '1'. Writing a '0' to this bit has no effect. Writing a '1' to this bit clears the Tamper interrupt flag.

**Bits 11:8 – CMPn[3:0]** Compare n [n = 3..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 is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bits 10:8 – CMPn[2:0]** Compare n [n = 2..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 is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bit 8 – 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 is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bit 8 – CMP0** Compare 0
This flag is cleared by writing a '1' to the flag.


© 2025 Microchip Technology Inc. and its subsidiaries

This flag is set on the next CLK_RTC_CNT cycle after a match with the compare condition, and an interrupt request is generated if INTENCLR/SET.COMP0 is '1'.
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 is generated if INTENCLR/SET.PERn is '1' .
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Periodic Interval n interrupt flag.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.7. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**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 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.8. 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | GP3 | GP2 | GP1 | GP0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| 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 | | COUNT | FREQCORR | ENABLE | SWRST |
| Access | | R | R | | R | R | R | R |
| Reset | | 0 | 0 | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – GPn** General Purpose n Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for GPn register is complete. |
| 1 | Write synchronization for GPn register is ongoing. |

**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 COMPx register is complete. |
| 1 | Write synchronization for COMPx 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. |


© 2025 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

**Note:** During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.9. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** 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, in other words., frequency is decreased. |
| 1 | The correction value is negative, in other words., frequency is 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.10. Counter Value in COUNT32 mode (CTRLA.MODE=0)

**Name:** COUNT
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** Write-Synchronized, Read-Synchronized

**Notes:**

- This register is read-synchronized when CTRLA.COUNTSYNC=1: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.
- 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 |
|---|---|---|---|---|---|---|---|---|
| | 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.11. Compare 0 Value in COUNT32 mode (CTRLA.MODE=0)

**Name:** COMP0
**Offset:** 0x20
**Reset:** 0x00000000
**Property:** Write-Synchronized

| 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 COMPn is continuously compared with the 32-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, and the counter value is cleared if CTRLA.MATCHCLR is one.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.12. Compare 1 Value in COUNT32 mode (CTRLA.MODE=1)

**Name:** COMP1
**Offset:** 0x24
**Reset:** 0x00000000
**Property:** Write-Synchronized

| 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 COMPn is continuously compared with the 32-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, and the counter value is cleared if CTRLA.MATCHCLR is one.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.13. General Purpose n

**Name:** GPn
**Offset:** 0x40 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 – GP[31:0]** General Purpose
These bits are for user-defined general purpose use, see *General Purpose Registers* from Related Links.

**Related Links**
General Purpose Registers


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.14. Tamper Control

**Name:** TAMPCTRL
**Offset:** 0x60
**Reset:** 0x00000000
**Property:** Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| Access | | | | | | | | |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| Access | | | | | | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – DEBNCn** Debounce Enable of Tamper Input INn [n=0..3]
**Note:** The Debounce feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | Debouncing is disabled for Tamper input INn |
| 1 | Debouncing is enabled for Tamper input INn |

**Bits 16, 17, 18, 19 – TAMLVLn** Tamper Level Select of Tamper Input INn [n=0..3]
**Note:** The Tamper Level feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | A falling edge condition will be detected on Tamper input INn. |
| 1 | A rising edge condition will be detected on Tamper input INn. |

**Bits 0:1, 2:3, 4:5, 6:7 – INnACT** Tamper Channel n Action [n=0..3]
These bits determine the action taken by Tamper Channel n.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Off (Disabled) |
| 0x1 | WAKE | Wake and set Tamper flag |
| 0x2 | CAPTURE | Capture timestamp and set Tamper flag |
| 0x3 | ACTL | Compare RTC signal routed between INn and OUT pins . When a mismatch occurs, capture timestamp and set Tamper flag |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.15. Timestamp

**Name:** TIMESTAMP
**Offset:** 0x64
**Reset:** 0x0
**Property:** -

**Note:** Once the value from TIMESTAMP register is read, INTFLAG.TAMPER bit must be cleared. The next value from this register should be read-only after the INTFLAG.TAMPER bit is set again.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 31:0 – COUNT[31:0]** Count Timestamp Value
The 32-bit value of COUNT is captured by the TIMESTAMP when a tamper condition occurs


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.16. Tamper ID

**Name:** TAMPID
**Offset:** 0x68
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TAMPEVT | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TAMPID3 | TAMPID2 | TAMPID1 | TAMPID0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 31 – TAMPEVT** Tamper Event Detected
Writing a '0' to this bit has no effect. Writing a '1' to this bit clears the tamper detection bit.

| Value | Description |
|---|---|
| 0 | A tamper input event has not been detected |
| 1 | A tamper input event has been detected |

**Bits 0, 1, 2, 3 – TAMPIDn** Tamper on Channel n Detected [n=0..3]
Writing a '0' to this bit has no effect. Writing a '1' to this bit clears the tamper detection bit.

| Value | Description |
|---|---|
| 0 | A tamper condition has not been detected on Channel n |
| 1 | A tamper condition has been detected on Channel n |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.8.17. Backup0

**Name:** BKUP0
**Offset:** 0x80
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 – BKUP[31:0]** Backup
These bits are user-defined for general purpose use in the Backup domain.


© 2025 Microchip Technology Inc. and its subsidiaries

## 25.9. Register Summary - Mode 1 - 16-Bit Counter

See the *RTCC* module in the *Product Memory Mapping Overview* from Related Links for the base address.

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CTRLA | 7:0 | | | | | MODE[1:0] | | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | GPTRST | BKTRST | | PRESCALER[3:0] | | | |
| 0x02 | CTRLB | 7:0 | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| | | 15:8 | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | TAMPEREO | | | CMPEOn[3:0] | | | |
| | | 23:16 | | | | | | | | TAMPEVEI |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | CMP3 | CMP2 | CMP1 | CMP0 |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | CMP3 | CMP2 | CMP1 | CMP0 |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | CMPn[5:0] | | | | | |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | COMP2 | COMP1 | COMP0 | PER | COUNT | FREQCORR | ENABLE | SWRST |
| | | 15:8 | COUNTSYNC | | | | | | | COMP3 |
| | | 23:16 | | | | | GP3 | GP2 | GP1 | GP0 |
| | | 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] | | | | | | | |
| 0x24 | COMP2 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |
| 0x26 | COMP3 | 7:0 | COMP[7:0] | | | | | | | |
| | | 15:8 | COMP[15:8] | | | | | | | |
| 0x28 ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | GP0 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x44 | GP1 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x48 | GP2 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary - Mode 1 - 16-Bit Counter** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x4C | GP3 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x50 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | TAMPCTRL | 7:0 | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| | | 31:24 | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| 0x64 | TIMESTAMP | 7:0 | COUNT[7:0] | | | | | | | |
| | | 15:8 | COUNT[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x68 | TAMPID | 7:0 | | | | | TAMPID3 | TAMPID2 | TAMPID1 | TAMPID0 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | TAMPEVT | | | | | | | |
| 0x6C ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | BKUP0 | 7:0 | BKUP[7:0] | | | | | | | |
| | | 15:8 | BKUP[15:8] | | | | | | | |
| | | 23:16 | BKUP[23:16] | | | | | | | |
| | | 31:24 | BKUP[31:24] | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 25.10. Register Description - Mode 1 - 16-Bit Counter

This Register Description section is valid if the RTC is in COUNT16 mode (CTRLA.MODE=1).


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.1. Control A in COUNT16 mode (CTRLA.MODE=1)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | GPTRST | BKTRST | | PRESCALER[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 | 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 prevents 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 |

**Bit 14 – GPTRST** GP Registers Reset On Tamper Enable
**Note:** Must disable this bit to avoid metastable reading of even GP registers.

Only GP registers enabled by the CTRLB.GPnEN bits are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | GPn registers will not reset when a tamper condition occurs. |
| 1 | GPn registers will reset when a tamper condition occurs. |

**Bit 13 – BKTRST** BKUP Registers Reset On Tamper Enable
All BKUPn registers are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | BKUPn registers will not reset when a tamper condition occurs. |
| 1 | BKUPn registers will reset when a tamper condition occurs. |

**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 |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 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 |

**Bit 1 – ENABLE** Enable
**Note:** Access to registers/bits are disallowed until SYNCBUSY.ENABLE cleared by hardware.

Due to synchronization, there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE reads back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) is set. SYNCBUSY.ENABLE is 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 is disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write operation is discarded.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the Reset is complete. CTRLA.SWRST is cleared when the Reset is complete.
**Note:** During a SWRST, access to registers/bits without SWRST is disallowed until SYNCBUSY.SWRST is cleared by hardware.

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing |
| 1 | The Reset operation is ongoing |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.2. Control B in COUNT16 mode (CTRLA.MODE=1)

**Name:** CTRLB
**Offset:** 0x02
**Reset:** 0x0000
**Property:** Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| Access | R/W | R/W | R/W | R/W | | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

**Bits 14:12 – ACTF[2:0]** Active Layer Frequency
These bits define the prescaling factor for the RTC clock output (OUT) used during active layer protection in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_OUT = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_OUT = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_OUT = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_OUT = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_OUT = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_OUT = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_OUT = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_OUT = CLK_RTC / 256 |

**Bits 10:8 – DEBF[2:0]** Debounce Frequency
These bits define the prescaling factor for the input debouncers in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_DEB = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_DEB = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_DEB = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_DEB = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_DEB = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_DEB = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_DEB = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_DEB = CLK_RTC / 256 |

**Bit 7 – DMAEN** DMA Enable
The RTC can trigger a DMA request when the timestamp is ready in the TIMESTAMP register.

| Value | Description |
|---|---|
| 0 | Tamper DMA request is disabled. Reading TIMESTAMP has no effect on INTFLAG.TAMPER. |
| 1 | Tamper DMA request is enabled. Reading TIMESTAMP will clear INTFLAG.TAMPER. |

**Bit 6 – RTCOUT** RTC Output Enable

| Value | Description |
|---|---|
| 0 | The RTC active layer output is disabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The RTC active layer output is enabled. |

**Bit 5 – DEBASYNC** Debouncer Asynchronous Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers operate synchronously. |
| 1 | The tamper input debouncers operate asynchronously. |

**Bit 4 – DEBMAJ** Debouncer Majority Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers match three equal values. |
| 1 | The tamper input debouncers match majority two of three values. |

**Bit 1 – GP2EN** General Purpose 2 Enable

| Value | Description |
|---|---|
| 0 | COMP1 compare function enabled. GP2/GP3 disabled. |
| 1 | COMP1 compare function disabled. GP2/GP3 enabled. |

**Bit 0 – GP0EN** General Purpose 0 Enable

| Value | Description |
|---|---|
| 0 | COMP0 compare function enabled. GP0/GP1 disabled. |
| 1 | COMP0 compare function disabled. GP0/GP1 enabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.3. Event Control in COUNT16 mode (CTRLA.MODE=1)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** Enable-Protected

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | TAMPEVEI |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | TAMPEREO | | | CMPEOn[3:0] | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 16 – TAMPEVEI** Tamper Event Input Enable

| Value | Description |
|---|---|
| 0 | Tamper event input is disabled, and incoming events will be ignored |
| 1 | Tamper event input is enabled, and incoming events will capture the COUNT value |

**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 14 – TAMPEREO** Tamper Event Output Enable

| Value | Description |
|---|---|
| 0 | Tamper event output is disabled, and will not be generated. |
| 1 | Tamper event output is enabled, and will be generated for every tamper input. |

**Bits 11:8 – CMPEOn[3:0]** Compare n Event Output Enable [n = 3..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. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.4. Interrupt Enable Clear in COUNT16 mode (CTRLA.MODE=1)

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

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

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVF | TAMPER | | | CMP3 | CMP2 | CMP1 | CMP0 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 14 – TAMPER** Tamper Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Tamper Interrupt Enable bit, which disables the Tamper interrupt.

| Value | Description |
|---|---|
| 0 | The Tamper interrupt is disabled. |
| 1 | The Tamper interrupt is enabled. |

**Bits 8, 9, 10, 11 – CMPn** Compare n Interrupt Enable [n = 3..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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.5. Interrupt Enable Set in COUNT16 mode (CTRLA.MODE=1)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** -

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 | TAMPER | | | CMP3 | CMP2 | CMP1 | CMP0 |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 14 – TAMPER** Tamper Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Tamper Interrupt Enable bit, which enables the Tamper interrupt.

| Value | Description |
|---|---|
| 0 | The Tamper interrupt is disabled. |
| 1 | The Tamper interrupt is enabled. |

**Bits 8, 9, 10, 11 – CMPn** Compare n Interrupt Enable [n = 3..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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.6. 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 | TAMPER | CMPn[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 |

| 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 is 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 14 – TAMPER** TAMPER
This flag is set after a tamper condition occurs, and an interrupt request is generated if INTENCLR.TAMPER/ INTENSET.TAMPER is '1' .
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Tamper interrupt flag.

**Bits 13:8 – CMPn[5:0]** Compare n [n = 5..0]
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 is generated if INTENCLR/SET.COMPn is '1' .
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bits 12:8 – CMPn[4:0]** Compare n [n = 4..0]
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 is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bits 8, 9, 10, 11 – CMPn** Compare n [n = 3..0]
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 is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n interrupt flag.

**Bits 10:8 – CMPn[2:0]** Compare n [n = 2..0]
This flag is cleared by writing a '1' to the flag.


© 2025 Microchip Technology Inc. and its subsidiaries

This flag is set on the next CLK_RTC_CNT cycle after an overflow condition occurs, and an interrupt request is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n 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 an overflow condition occurs, and an interrupt request is generated if INTENCLR/SET.COMPn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Compare n 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 an overflow condition occurs, and an interrupt request is generated if INTENCLR/SET.COMP0 is '1'.
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 is 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.7. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**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 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.8. Synchronization Busy in COUNT16 mode (CTRLA.MODE=1)

**Name:** SYNCBUSY
**Offset:** 0x10
**Reset:** 0x00000000

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | GP3 | GP2 | GP1 | GP0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | COUNTSYNC | | | | | | | COMP3 |
| Access | R | | | | | | | R |
| Reset | 0 | | | | | | | 0 |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COMP2 | COMP1 | COMP0 | PER | COUNT | FREQCORR | ENABLE | SWRST |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – GPn** General Purpose n Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for GPn register is complete. |
| 1 | Write synchronization for GPn register is ongoing. |

**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, 7, 8 – COMPn** Compare n Synchronization Busy Status [n = 3..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. |


© 2025 Microchip Technology Inc. and its subsidiaries

**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. |
| 1 | Write synchronization for CTRLA.ENABLE bit is ongoing. |

**Bit 0 – SWRST** Software Reset Synchronization Busy Status

**Note:** During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.9. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** 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, in other words., frequency is decreased. |
| 1 | The correction value is negative, in other words., frequency is 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.10. Counter Value in COUNT16 mode (CTRLA.MODE=1)

**Name:** COUNT
**Offset:** 0x18
**Reset:** 0x0000
**Property:** Write-Synchronized, Read-Synchronized

**Notes:**

- This register is read-synchronized when CTRLA.COUNTSYNC=1: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.
- This register is write-synchronized: SYNCBUSY.COUNT must be checked to ensure the COUNT register synchronization is complete.

| 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).


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.11. Counter Period in COUNT16 mode (CTRLA.MODE=1)

**Name:** PER
**Offset:** 0x1C
**Reset:** 0x0000
**Property:** 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).


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.12. Compare n Value in COUNT16 mode (CTRLA.MODE=1)

**Name:** COMP
**Offset:** 0x20 + n*0x02 [n=0..3]
**Reset:** 0x0000
**Property:** Write-Synchronized

| 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.13. Tamper Control

**Name:** TAMPCTRL
**Offset:** 0x60
**Reset:** 0x00000000
**Property:** Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| Access | | | | | | | | |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| Access | | | | | | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – DEBNCn** Debounce Enable of Tamper Input INn [n=0..3]
**Note:** The Debounce feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | Debouncing is disabled for Tamper input INn |
| 1 | Debouncing is enabled for Tamper input INn |

**Bits 16, 17, 18, 19 – TAMLVLn** Tamper Level Select of Tamper Input INn [n=0..3]
**Note:** The Tamper Level feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | A falling edge condition will be detected on Tamper input INn. |
| 1 | A rising edge condition will be detected on Tamper input INn. |

**Bits 0:1, 2:3, 4:5, 6:7 – INnACT** Tamper Channel n Action [n=0..3]
These bits determine the action taken by Tamper Channel n.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Off (Disabled) |
| 0x1 | WAKE | Wake and set Tamper flag |
| 0x2 | CAPTURE | Capture timestamp and set Tamper flag |
| 0x3 | ACTL | Compare RTC signal routed between INn and OUT pins . When a mismatch occurs, capture timestamp and set Tamper flag |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.14. Timestamp

**Name:** TIMESTAMP
**Offset:** 0x64
**Reset:** 0x0000
**Property:** -

**Note:** Once the value from TIMESTAMP register is read, INTFLAG.TAMPER bit must be cleared. The next value from this register should be read-only after the INTFLAG.TAMPER bit is set again.

| 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 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[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 |
|---|---|---|---|---|---|---|---|---|
| | COUNT[7:0] | | | | | | | |
| Access | R | R | R | R | R | R | R | R |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 15:0 – COUNT[15:0]** Count Timestamp Value
The 16-bit value of COUNT is captured by the TIMESTAMP when a tamper condition occurs.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.15. Tamper ID

**Name:** TAMPID
**Offset:** 0x68
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TAMPEVT | | | | | | | |
| Access | R/W | | | | | | | |
| Reset | 0 | | | | | | | |

| 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TAMPID3 | TAMPID2 | TAMPID1 | TAMPID0 |
| Access | | | | | R/W | R/W | R/W | R/W |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 31 – TAMPEVT** Tamper Event Detected
Writing a '0' to this bit has no effect. Writing a '1' to this bit clears the tamper detection bit.

| Value | Description |
|---|---|
| 0 | A tamper input event has not been detected |
| 1 | A tamper input event has been detected |

**Bits 0, 1, 2, 3 – TAMPIDn** Tamper on Channel n Detected [n=0..3]
Writing a '0' to this bit has no effect. Writing a '1' to this bit clears the tamper detection bit.

| Value | Description |
|---|---|
| 0 | A tamper condition has not been detected on Channel n |
| 1 | A tamper condition has been detected on Channel n |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.16. Backup0

**Name:** BKUP0
**Offset:** 0x80
**Reset:** 0x00000000

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 |
|---|---|---|---|---|---|---|---|---|
| | BKUP[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 – BKUP[31:0]** Backup
These bits are user-defined for general purpose use in the Backup domain.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.10.17. General Purpose n

**Name:** GPn
**Offset:** 0x40 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 – GP[31:0]** General Purpose
These bits are for user-defined general purpose use, see *General Purpose Registers* from Related Links.

**Related Links**
General Purpose Registers


© 2025 Microchip Technology Inc. and its subsidiaries

## 25.11. Register Summary - Mode 2 - Clock/Calendar

See the *RTCC* module in the *Product Memory Mapping Overview* from Related Links for the base address.

| 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 | GPTRST | BKTRST | | PRESCALER[3:0] | | | |
| 0x02 | CTRLB | 7:0 | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| | | 15:8 | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 0x04 | EVCTRL | 7:0 | PEREO7 | PEREO6 | PEREO5 | PEREO4 | PEREO3 | PEREO2 | PEREO1 | PEREO0 |
| | | 15:8 | OVFEO | TAMPEREO | | | | | ALARMEO1 | ALARMEO0 |
| | | 23:16 | | | | | | | | TAMPEVEI |
| | | 31:24 | | | | | | | | |
| 0x08 | INTENCLR | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | | | ALARM1 | ALARM0 |
| 0x0A | INTENSET | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | | | ALARM1 | ALARM0 |
| 0x0C | INTFLAG | 7:0 | PER7 | PER6 | PER5 | PER4 | PER3 | PER2 | PER1 | PER0 |
| | | 15:8 | OVF | TAMPER | | | ALARMn[3:0] | | | |
| 0x0E | DBGCTRL | 7:0 | | | | | | | | DBGRUN |
| 0x0F | Reserved | | | | | | | | | |
| 0x10 | SYNCBUSY | 7:0 | | ALARM1 | ALARM0 | | CLOCK | FREQCORR | ENABLE | SWRST |
| | | 15:8 | CLOCKSYNC | | | MASK1 | MASK0 | | | |
| | | 23:16 | | | | | GP3 | GP2 | GP1 | GP0 |
| | | 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 | ALARM0 | 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 | MASK0 | 7:0 | | | | | | SEL[2:0] | | |
| 0x25 ... 0x27 | Reserved | | | | | | | | | |
| 0x28 | ALARM1 | 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] | |
| 0x2C | MASK1 | 7:0 | | | | | | SEL[2:0] | | |
| 0x2D ... 0x3F | Reserved | | | | | | | | | |
| 0x40 | GP0 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x44 | GP1 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |


© 2025 Microchip Technology Inc. and its subsidiaries

**Register Summary - Mode 2 - Clock/Calendar** (continued)

| Offset | Name | Bit Pos. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0x48 | GP2 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x4C | GP3 | 7:0 | GP[7:0] | | | | | | | |
| | | 15:8 | GP[15:8] | | | | | | | |
| | | 23:16 | GP[23:16] | | | | | | | |
| | | 31:24 | GP[31:24] | | | | | | | |
| 0x50 ... 0x5F | Reserved | | | | | | | | | |
| 0x60 | TAMPCTRL | 7:0 | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| | | 31:24 | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| 0x64 | TIMESTAMP | 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] | |
| 0x68 | TAMPID | 7:0 | | | | | TAMPID3 | TAMPID2 | TAMPID1 | TAMPID0 |
| | | 15:8 | | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | TAMPEVT | | | | | | | |
| 0x6C ... 0x7F | Reserved | | | | | | | | | |
| 0x80 | BKUP0 | 7:0 | BKUP[7:0] | | | | | | | |
| | | 15:8 | BKUP[15:8] | | | | | | | |
| | | 23:16 | BKUP[23:16] | | | | | | | |
| | | 31:24 | BKUP[31:24] | | | | | | | |

**Related Links**

Product Memory Mapping Overview

## 25.12. Register Description - Mode 2 - Clock/Calendar

This Register Description section is valid if the RTC is in Clock/Calendar mode (CTRLA.MODE=2).


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.1. Control A in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** CTRLA
**Offset:** 0x00
**Reset:** 0x0000
**Property:** Enable-Protected, Write-Synchronized

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CLOCKSYNC | GPTRST | BKTRST | | PRESCALER[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 | 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 prevents 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 |

**Bit 14 – GPTRST** GP Registers Reset On Tamper Enable
**Note:** Must disable this bit to avoid metastable reading of even GP registers.

Only GP registers enabled by the CTRLB.GPnEN bits are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

**Bit 13 – BKTRST** BKUP Registers Reset On Tamper Enable
All BKUPn registers are affected. This bit can be written only when the peripheral is disabled.
This bit is not synchronized.

| Value | Description |
|---|---|
| 0 | BKUPn registers will not reset when a tamper condition occurs. |
| 1 | BKUPn registers will reset when a tamper condition occurs. |

**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 |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Name | Description |
|---|---|---|
| 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 match |
| 1 | The counter is cleared on a Compare/Alarm match |

**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

**Note:** Access to registers/bits are disallowed until SYNCBUSY.ENABLE cleared by hardware.

Due to synchronization there is delay from writing CTRLA.ENABLE until the peripheral is enabled/disabled. The value written to CTRLA.ENABLE reads back immediately and the Enable bit in the Synchronization Busy register (SYNCBUSY.ENABLE) is set. SYNCBUSY.ENABLE is 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 is disabled.
Writing a '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write operation is discarded.
Due to synchronization, there is a delay from writing CTRLA.SWRST until the Reset is complete. CTRLA.SWRST is cleared when the Reset is complete.
**Note:** During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST is cleared by hardware.


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 0 | There is no Reset operation ongoing |
| 1 | The Reset operation is ongoing |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.2. Control B in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** CTRLB
**Offset:** 0x2
**Reset:** 0x0000
**Property:** Enable-Protected

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | | ACTF[2:0] | | | | DEBF[2:0] | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | DMAEN | RTCOUT | DEBASYNC | DEBMAJ | | | GP2EN | GP0EN |
| Access | R/W | R/W | R/W | R/W | | | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | | | 0 | 0 |

**Bits 14:12 – ACTF[2:0]** Active Layer Frequency
These bits define the prescaling factor for the RTC clock output (OUT) used during active layer protection in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_OUT = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_OUT = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_OUT = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_OUT = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_OUT = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_OUT = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_OUT = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_OUT = CLK_RTC / 256 |

**Bits 10:8 – DEBF[2:0]** Debounce Frequency
These bits define the prescaling factor for the input debouncers in terms of the CLK_RTC.

| Value | Name | Description |
|---|---|---|
| 0x0 | DIV2 | CLK_RTC_DEB = CLK_RTC / 2 |
| 0x1 | DIV4 | CLK_RTC_DEB = CLK_RTC / 4 |
| 0x2 | DIV8 | CLK_RTC_DEB = CLK_RTC / 8 |
| 0x3 | DIV16 | CLK_RTC_DEB = CLK_RTC / 16 |
| 0x4 | DIV32 | CLK_RTC_DEB = CLK_RTC / 32 |
| 0x5 | DIV64 | CLK_RTC_DEB = CLK_RTC / 64 |
| 0x6 | DIV128 | CLK_RTC_DEB = CLK_RTC / 128 |
| 0x7 | DIV256 | CLK_RTC_DEB = CLK_RTC / 256 |

**Bit 7 – DMAEN** DMA Enable
The RTC can trigger a DMA request when the timestamp is ready in the TIMESTAMP register.

| Value | Description |
|---|---|
| 0 | Tamper DMA request is disabled. Reading TIMESTAMP has no effect on INTFLAG.TAMPER. |
| 1 | Tamper DMA request is enabled. Reading TIMESTAMP will clear INTFLAG.TAMPER. |

**Bit 6 – RTCOUT** RTC Out Enable

| Value | Description |
|---|---|
| 0 | The RTC active layer output is disabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | The RTC active layer output is enabled. |

**Bit 5 – DEBASYNC** Debouncer Asynchronous Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers operate synchronously. |
| 1 | The tamper input debouncers operate asynchronously. |

**Bit 4 – DEBMAJ** Debouncer Majority Enable

| Value | Description |
|---|---|
| 0 | The tamper input debouncers match three equal values. |
| 1 | The tamper input debouncers match majority two of three values. |

**Bit 1 – GP2EN** General Purpose 2 Enable

| Value | Description |
|---|---|
| 0 | COMP1 compare function enabled. GP2/GP3 disabled. |
| 1 | COMP1 compare function disabled. GP2/GP3 enabled. |

**Bit 0 – GP0EN** General Purpose 0 Enable

| Value | Description |
|---|---|
| 0 | COMP0 compare function enabled. GP0 disabled. |
| 1 | COMP0 compare function disabled. GP0 enabled. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.3. Event Control in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** EVCTRL
**Offset:** 0x04
**Reset:** 0x00000000
**Property:** Enable-Protected

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

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | TAMPEVEI |
| Access | | | | | | | | R/W |
| Reset | | | | | | | | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | OVFEO | TAMPEREO | | | | | ALARMEO1 | ALARMEO0 |
| Access | R/W | R/W | | | | | R/W | R/W |
| Reset | 0 | 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 16 – TAMPEVEI** Tamper Event Input Enable

| Value | Description |
|---|---|
| 0 | Tamper event input is disabled, and incoming events will be ignored. |
| 1 | Tamper event input is enabled, and all incoming events will capture the CLOCK value. |

**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 14 – TAMPEREO** Tamper Event Output Enable

| Value | Description |
|---|---|
| 0 | Tamper event output is disabled, and will not be generated |
| 1 | Tamper event output is enabled, and will be generated for every tamper input. |

**Bits 8, 9 – ALARMEOn** Alarm n Event Output Enable [n = 1..0]

| Value | Description |
|---|---|
| 0 | Alarm n event is disabled and will not be generated. |
| 1 | Alarm 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 1 | Periodic Interval n event is enabled and will be generated. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.4. Interrupt Enable Clear in Clock/Calendar mode (CTRLA.MODE=2)

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

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 | TAMPER | | | | | ALARM1 | ALARM0 |
| Access | R/W | R/W | | | | | R/W | R/W |
| Reset | 0 | 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. |

**Bit 14 – TAMPER** Tamper Interrupt Enable

**Bits 8, 9 – ALARMn** Alarm n Interrupt Enable [n = 1..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will clear the Alarm n Interrupt Enable bit, which disables the Alarm n interrupt.

| Value | Description |
|---|---|
| 0 | The Alarm n interrupt is disabled. |
| 1 | The Alarm 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.5. Interrupt Enable Set in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** INTENSET
**Offset:** 0x0A
**Reset:** 0x0000
**Property:** -

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 | TAMPER | | | | | ALARM1 | ALARM0 |
| Access | R/W | R/W | | | | | R/W | R/W |
| Reset | 0 | 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. |

**Bit 14 – TAMPER** Tamper Interrupt Enable
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Tamper Interrupt Enable bit, which enables the Tamper interrupt.

| Value | Description |
|---|---|
| 0 | The Tamper interrupt it disabled. |
| 1 | The Tamper interrupt is enabled. |

**Bits 8, 9 – ALARMn** Alarm n Interrupt Enable [n = 1..0]
Writing a '0' to this bit has no effect. Writing a '1' to this bit will set the Alarm n Interrupt Enable bit, which enables the Alarm n interrupt.

| Value | Description |
|---|---|
| 0 | The Alarm n interrupt is disabled. |
| 1 | The Alarm 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.6. 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 | TAMPER | | | ALARMn[3:0] | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | 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 is 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 14 – TAMPER** TAMPER
This flag is set after a tamper condition occurs, and an interrupt request is generated if INTENCLR.TAMPER/ INTENSET.TAMPER is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Tamper Interrupt flag.

**Bits 11:8 – ALARMn[3:0]** Alarm n [n = 3..0]
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 is generated if INTENCLR/SET.ALARMn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Alarm n interrupt flag.

**Bits 10:8 – ALARMn[2:0]** Alarm n [n = 2..0]
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 is generated if INTENCLR/SET.ALARMn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Alarm n interrupt flag.

**Bits 8, 9 – ALARMn** Alarm 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 an overflow condition occurs, and an interrupt request is generated if INTENCLR/SET.ALARMn is '1'.
Writing a '0' to this bit has no effect.
Writing a '1' to this bit clears the Alarm n interrupt flag.

**Bit 8 – ALARM0** Alarm 0
This flag is cleared by writing a '1' to the flag.


© 2025 Microchip Technology Inc. and its subsidiaries

This flag is set on the next CLK_RTC_CNT cycle after an overflow condition occurs, and an interrupt request is generated if INTENCLR/SET.ALARM0 is '1'.
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 is 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.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.7. Debug Control

**Name:** DBGCTRL
**Offset:** 0x0E
**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 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.8. 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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | GP3 | GP2 | GP1 | GP0 |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | CLOCKSYNC | | | MASK1 | MASK0 | | | |
| Access | R | | | R | R | | | |
| Reset | 0 | | | 0 | 0 | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | ALARM1 | ALARM0 | | CLOCK | FREQCORR | ENABLE | SWRST |
| Access | | R | R | | R | R | R | R |
| Reset | | 0 | 0 | | 0 | 0 | 0 | 0 |

**Bits 16, 17, 18, 19 – GPn** General Purpose n Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Write synchronization for GPn register is complete. |
| 1 | Write synchronization for GPn register is ongoing. |

**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. |

**Bits 11, 12 – MASKn** Mask n Synchronization Busy Status [n = 1..0]

| Value | Description |
|---|---|
| 0 | Write synchronization for MASKx register is complete. |
| 1 | Write synchronization for MASKx register is ongoing. |

**Bits 5, 6 – ALARMn** Alarm n Synchronization Busy Status [n = 1..0]

| Value | Description |
|---|---|
| 0 | Write synchronization for ALARMx register is complete. |
| 1 | Write synchronization for ALARMx register is ongoing. |

**Bit 3 – CLOCK** Clock Register Synchronization Busy Status

| Value | Description |
|---|---|
| 0 | Read/write synchronization for CLOCK register is complete. |


© 2025 Microchip Technology Inc. and its subsidiaries

| Value | Description |
|---|---|
| 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. |
| 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

**Note:** During a SWRST, access to registers/bits without SWRST are disallowed until SYNCBUSY.SWRST cleared by hardware.

| Value | Description |
|---|---|
| 0 | Write synchronization for CTRLA.SWRST bit is complete. |
| 1 | Write synchronization for CTRLA.SWRST bit is ongoing. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.9. Frequency Correction

**Name:** FREQCORR
**Offset:** 0x14
**Reset:** 0x00
**Property:** 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, in other words., frequency is decreased. |
| 1 | The correction value is negative, in other words., frequency is 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. |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.10. Clock Value in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** CLOCK
**Offset:** 0x18
**Reset:** 0x00000000
**Property:** Write-Synchronized, Read-Synchronized

**Notes:**

- This register is read-synchronized and write-synchronized: SYNCBUSY.CLOCK must be checked to ensure the CLOCK register synchronization is complete.
- 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


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.11. Alarm n Value in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** ALARM
**Offset:** 0x20 + n*0x08 [n=0..1]
**Reset:** 0x00000000
**Property:** Write-Synchronized

The 32-bit value of ALARMn is continuously compared with the 32-bit CLOCK value, based on the masking set by MASKn.SEL. When a match occurs, the Alarm n interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG.ALARMn) is set on the next counter cycle, and the counter is cleared if CTRLA.MATCHCLR is '1'.

| 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 MASKn.SEL is 6.

**Bits 25:22 – MONTH[3:0]** Month
The alarm month. Months are matched only if MASKn.SEL is greater than 4.

**Bits 21:17 – DAY[4:0]** Day
The alarm day. Days are matched only if MASKn.SEL is greater than 3.

**Bits 16:12 – HOUR[4:0]** Hour
The alarm hour. Hours are matched only if MASKn.SEL is greater than 2.

**Bits 11:6 – MINUTE[5:0]** Minute
The alarm minute. Minutes are matched only if MASKn.SEL is greater than 1.

**Bits 5:0 – SECOND[5:0]** Second
The alarm second. Seconds are matched only if MASKn.SEL is greater than 0.


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.12. Alarm n Mask in Clock/Calendar mode (CTRLA.MODE=2)

**Name:** MASK
**Offset:** 0x24 + n*0x08 [n=0..1]
**Reset:** 0x00
**Property:** Write-Synchronized

| 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 n 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 |


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.13. General Purpose n

**Name:** GPn
**Offset:** 0x40 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** -

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 |
|---|---|---|---|---|---|---|---|---|
| | GP[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 – GP[31:0]** General Purpose
These bits are for user-defined general purpose use, see *General Purpose Registers* from Related Links.

**Related Links**
General Purpose Registers


© 2025 Microchip Technology Inc. and its subsidiaries

### 25.12.14. Tamper Control

**Name:** TAMPCTRL
**Offset:** 0x60
**Reset:** 0x00000000
**Property:** Enable-Protected

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | | | | | DEBNC3 | DEBNC2 | DEBNC1 | DEBNC0 |
| Access | | | | | | | | |
| Reset | | | | | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | | | | TAMLVL3 | TAMLVL2 | TAMLVL1 | TAMLVL0 |
| Access | | | | | | | | |
| 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 |
|---|---|---|---|---|---|---|---|---|
| | IN3ACT[1:0] | | IN2ACT[1:0] | | IN1ACT[1:0] | | IN0ACT[1:0] | |
| Access | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 24, 25, 26, 27 – DEBNCn** Debounce Enable of Tamper Input INn [n=0..3]
**Note:** The Debounce feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | Debouncing is disabled for Tamper input INn |
| 1 | Debouncing is enabled for Tamper input INn |

**Bits 16, 17, 18, 19 – TAMLVLn** Tamper Level Select of Tamper Input INn [n=0..3]
**Note:** The Tamper Level feature does not apply to the Active Layer Protection mode (TAMPCTRL.INACT = ACTL).

| Value | Description |
|---|---|
| 0 | A falling edge condition will be detected on Tamper input INn. |
| 1 | A rising edge condition will be detected on Tamper input INn. |

**Bits 0:1, 2:3, 4:5, 6:7 – INnACT** Tamper Channel n Action [n=0..3]
These bits determine the action taken by Tamper Channel n.

| Value | Name | Description |
|---|---|---|
| 0x0 | OFF | Off (Disabled) |
| 0x1 | WAKE | Wake and set Tamper flag |
| 0x2 | CAPTURE | Capture timestamp and set Tamper flag |
| 0x3 | ACTL | Compare RTC signal routed between INn and OUT pins . When a mismatch occurs, capture timestamp and set Tamper flag |


© 2025 Microchip Technology Inc. and its subsidiaries