Section 9. Prefetch Module for Devices with L1 CPU Cache

HIGHLIGHTS

This section of the manual contains the following major topics:

Note: This family reference manual section is meant to serve as a complement to the PIC32MZ W1device data sheet.

Please consult the note at the beginning of the **"Prefetch Module"** chapter in the current device data sheet to determine if this document supports the device you are using.

Device data sheets and family reference manual sections are available for download from the Microchip Worldwide website at: http://www.microchip.com.

9.1 INTRODUCTION

This section describes the features and operation of the Prefetch module for PIC32MZ W1 devices with L1 CPU Cache. The Prefetch module features increase system performance for most applications.

9.1.1 Prefetch Module Features

The Prefetch module includes the following features:

- 12- x 32-byte Fully Associative Lines
- Four Lines for CPU Instructions
- Four Lines for CPU Data
- Four Lines for Peripheral Data
- 32-byte Cache Lines (256 bits) Parallel Memory Fetch
- Configurable Predictive Prefetch
- Error Detection and Correction

9.2 PREFETCH MODULE OVERVIEW

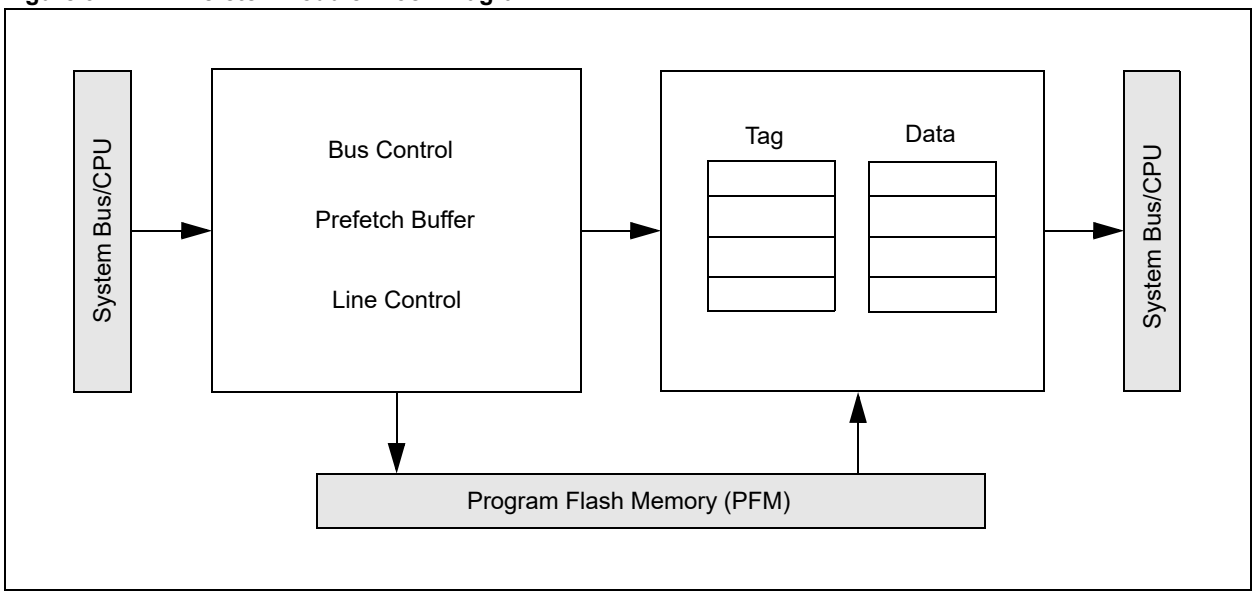
The Prefetch module is a performance-enhancing module included in PIC32MZ W1 devices with L1 CPU caches. When running at high-clock rates, Wait states must be inserted into Program Flash Memory (PFM) read transactions to meet the access time of the PFM. Wait states can be hidden to the core by prefetching and storing 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. Normally, the lines hold a copy of what is currently in memory to make instructions or data available to the CPU without Wait states.

Data located in the PFM may be requested by the CPU or by a peripheral. If the requested data is not currently stored in a 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.

Figure 9-1 shows a block diagram of the Prefetch module. Logically, the Prefetch module fits between the System Bus module and the PFM.

Figure 9-1: Prefetch Module Block Diagram



9.2.1 Prefetch Module Line Organization

The Prefetch module consists of two arrays, data and tag, each of which holds 12 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.

Figure 9-2 and Figure 9-3 illustrate the organization of a line.

Figure 9-2: Tag Line

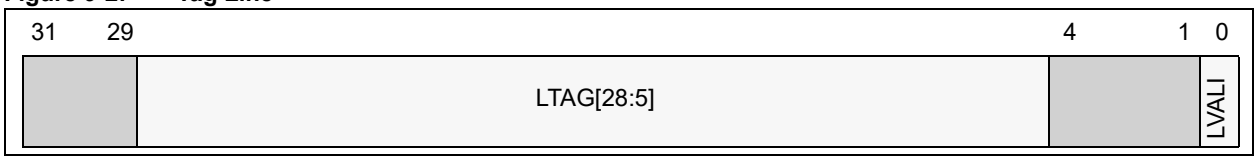


Figure 9-3: Data Line

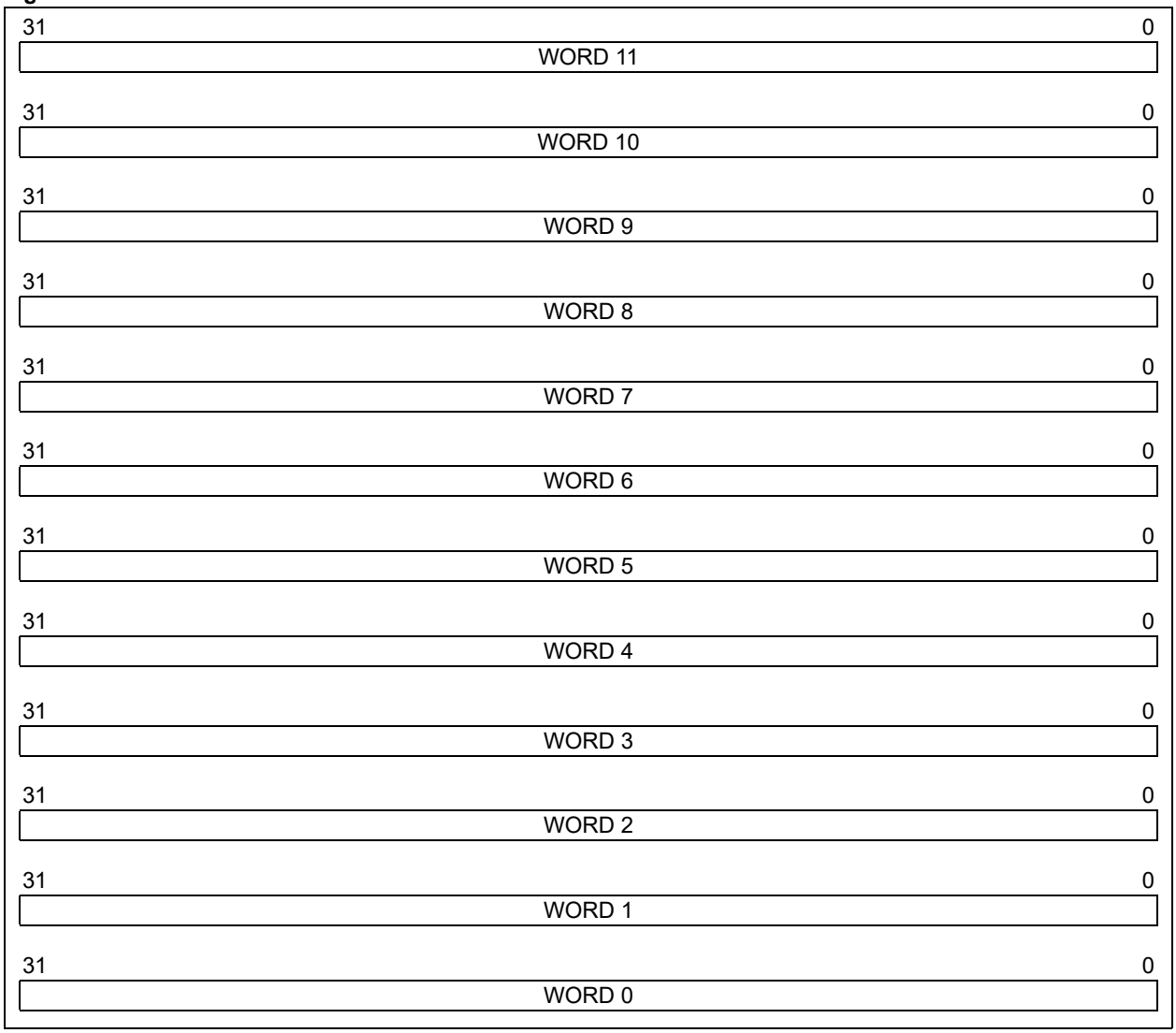


Figure 9-4: Prefetch Module Arrays

Line #	Tag Array(1)		Data Array(1)							
0	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
1	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
2	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
3	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
4	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
5	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
6	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
7	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
8	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
9	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
10	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0
11	TAG	Valid	WORD 7	WORD 6	WORD 5	WORD 4	WORD 3	WORD 2	WORD 1	WORD 0

Note 1: These arrays cannot be read or written by the user application.

9.3 CONTROL REGISTERS

The Prefetch module for PIC32MZ W1 devices with L1 CPU cache contains the following Special Function Registers (SFRs):

- **PRECON: Prefetch Module Control Register(1)**(1)

  This register manages the configuration of the Prefetch module and controls Wait states. Applicable only for PIC32MZ W1 devices.

- **PRESTAT: Prefetch Module Status Register**

  This register contains status information for error correction and detection.

- **PREHIT: Prefetch Module Hit Statistics Register(1)**(2)

  This register contains Status information for cache hit count statistics.

- **PREMIS: Prefetch Module Miss Statistics Register (1)** (2)

  This register contains Status information for cache miss count statistics.

Table 9-1 provides a brief summary of the related Prefetch module registers. Corresponding registers appear after the summary, followed by a detailed description of each bit.

Table 9-1: Prefetch Module SFR Summary

Name		Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 18/2	Bit 17/1	Bit 16/0
PRECON (1)(3)	31:16	—	—	—	—	—	PER-CHEEN	DCHEEN	ICHEEN	—	PERCHE-INV	DCHE-INV	ICHE-INV	—	PER-CHECOH	DCHECOH	ICHECOH
	15:0	—	—	—	CHEPERFEN	—	—	—	PFMAWSEN	PFMSE-CEN	—	PREFEN[1:0]		PFMWS[3:0](2)			
PRESTAT(1)	31:16	—	—	—	—	PFMDED	PFMSEC	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	PFMSECCNT[7:0]							
PREHIT(2)	31:16	CHEHIT[31:16]															
	15:00	CHEHIT[15:0]															
PREMIS(2)	31:16	CHEMIS[31:16]															
	15:00	CHEMIS[15:0]															

Legend: — = unimplemented, read as '0'.

Note 1: These registers have associated Clear, Set and Invert registers at offsets of 0x4, 0x8 and 0xC bytes, respectively. The Clear, Set and Invert registers have the same name with CLR, SET or INV appended to the register name (e.g., PRECONCLR). Writing a '1' to any bit position in these registers will clear, set or invert valid bits in the associated register. It is recommended that reads from these registers be ignored.

2: The number of bits may vary from device to device. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine the width.

3: Based on the device type, the SFR definition is selected. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine the SFR definition.

Register 9-1: PRECON: Prefetch Module Control Register(1)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
	—	—	—	—	—	PERCHEEN	DCHEEN	ICHEEN
23:16	U-0	R/S/HC-0	R/S/HC-0	R/S/HC-0	U-0	R/W-0	R/W-0	R/W-0
	—	PERCHEINV	DCHEINV	ICHEINV	—	PERCHECOH	DCHECOH	ICHECOH
15:8	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0	R/W-1
	—	—	—	CHEPERFEN	—	—	—	PFMAWSEN
7:0	R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
	PFMSECEN	—	PREFEN[1:0]		PFMWS[3:0](2)			

Legend:	HC = Hardware Cleared	HS = Hardware Set	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
S = Settable bit	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-27 **Unimplemented:** Read as '0'

bit 26, **PERCHEEN:** Peripheral Data Cache Enable bit
1 = Caching enabled
0 = Caching disabled (and all lines invalidated)

bit 25 **DCHEEN:** Data Cache Enable bit
1 = Caching enabled
0 = Caching disabled (and all lines invalidated)

bit 24 **ICHEEN:** Instruction Cache Enable bit
1 = Caching enabled
0 = Caching disabled (and all lines invalidated)

bit 23 **Unimplemented:** Read as '0'

bit 22 **PERCHEINV:** Manual Invalidate Control for Peripheral Data Cache
1 = Force invalidate cache/invalidate busy
0 = Cache invalidation follows PERCHECOH/invalid complete

Note 1: PFB is included with the iCache invalidate.

2: Hardware auto clears this bit when the cache invalidate completes. Bits may clear at different times.

bit 21 **DCHEINV:** Manual Invalidate Control for Data Cache
1 = Force invalidate cache/invalidate busy
0 = Cache invalidation follows DCHECOH/invalid complete

Note 1: PFB is included with the iCache invalidate.

2: Hardware auto clears this bit when the cache invalidate completes. Bits may clear at different times.

Note 1: This register is not available on all devices. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine availability.

2: For the Wait states to SYSCLK relationship, refer to the **"Electrical Specification"** chapter in the specific device data sheet.

Register 9-1: PRECON: Prefetch Module Control Register(1) (Continued)

bit 20 **ICHEINV**: Manual Invalidate Control for Instruction Cache
1 = Force invalidate cache/invalidate busy
0 = Cache invalidation follows ICHECOH/invalid complete

Note 1: PFB is included with the CPU instruction Cache invalidate.

2: Hardware auto clears this bit when the cache invalidate completes. Bits may clear at different times.

bit 19 **Unimplemented:** Read as '0'

bit 18 **PERCHECOH:** Auto Cache Coherency Control for Peripheral Data Cache
1 = Auto invalidate cache on a programming event
0 = No auto invalidated cache on a programming event

Note: PRECHEOH must be stable before initiation of programming to ensure correct invalidation of data.

bit 17 **DCHECOH:** Auto Cache Coherency Control for Data Cache
1 = Auto invalidate cache on a programming event
0 = No auto invalidated cache on a programming event

Note: DCHECOH must be stable before initiation of programming to ensure correct invalidation of data.

bit 16 **ICHECOH:** Auto Cache Coherency Control for Instruction Cache
1 = Auto invalidate cache on a programming event
0 = No auto invalidated cache on a programming event

Note: ICHECOH must be stable before initiation of programming to ensure correct invalidation of data.

bit 15-13 **Unimplemented:** Read as '0'

bit 12 **CHEPERFEN**: Cache Performance Counters Enable
1 = Performance counters is enabled
0 = Performance counters is disabled

Note: Performance counters are reset on 0 to 1 transition of this bit.

bit 11-9 **Unimplemented:** Read as '0'

bit 8 **PFMAWSEN:** Address Wait State Enable
Total Flash wait states are PFMAWSEN + PFMWS.
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
1 = Generate an interrupt when PFMSEC is set
0 = Do not generate an interrupt when PFMSEC is set

bit 6 **Unimplemented:** Read as '0'

bit 5-4 **PREFEN[1:0]**: Instruction Predictive Prefetch Enable
01 = Instruction predictive prefetch enabled for cacheable regions only
00 = Instruction predictive prefetch disabled
Other values are unavailable.

Note 1: This register is not available on all devices. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine availability.

2: For the Wait states to SYSCLK relationship, refer to the **"Electrical Specification"** chapter in the specific device data sheet.

Register 9-1: PRECON: Prefetch Module Control Register(1) (Continued)

bit 3-0 **PFMWS[3:0]:** PFM Access Time Defined in Terms of SYSCLK Wait States bits(2)

Total Flash Wait states are PFMAWSEN + PFMWS.

1111 = Fifteen Wait states
1110 = Fourteen Wait states
...

0001 = One Wait state
0000 = Zero Wait state

Note: This is not the Wait state seen by the CPU.

Note 1: This register is not available on all devices. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine availability.

2: For the Wait states to SYSCLK relationship, refer to the **"Electrical Specification"** chapter in the specific device data sheet.

Register 9-2: PRESTAT: Prefetch Module Status Register

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	HS, R/C-0	HS, R/W-0	U-0	U-0
	—	—	—	—	PFMDED	PFMSEC	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0	HS, HC, R/W-0
	PFMSECCNT[7:0]							

Legend:	HS = Set by hardware	HC = Cleared by hardware	C = Clearable bit
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

bit 31-28 **Unimplemented:** Write '0'; ignore read

bit 27 **PFMDED:** Flash Double-bit Error Detected (DED) Status bit
This bit is set in hardware and can only be cleared (i.e., set to '0') in software.
1 = A DED error has occurred
0 = A DED error has not occurred

bit 26 **PFMSEC:** Flash Single-bit Error Corrected (SEC) Status bit
1 = A SEC error occurred when PFMSECCNT[7:0] was equal to zero
0 = A SEC error has not occurred

bit 25-8 **Unimplemented:** Write '0'; ignore read

bit 7-0 **PFMSECCNT[7:0]:** Flash SEC Count bits
Decrements by 1 its count value 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.

Register 9-3: PREHIT: Prefetch Module Hit Statistics Register(1)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEHIT[31:24]							
23:16	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEHIT[23:16]							
15:8	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEHIT[15:8]							
7:0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEHIT[7:0]							

Legend:	HC = Hardware Cleared	HS = Hardware Set	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
S = Settable bit	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-0 **CHEHIT[31:0]**: Instruction Cache Hit Count bits
When CHECON.CHEPERF = 1, CHEHIT increments once per iCache or PFB hit.

Note: CHEHIT is reset on 0 to 1 transition of CHECON.CHEPERF.

Note 1: This register is not available on all devices. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine availability.

Register 9-4: PREMIS: Prefetch Module Miss Statistics Register (1)

Bit Range	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEMIS[31:24]							
23:16	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEMIS[23:16]							
15:8	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEMIS[15:8]							
7:0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0	R/H/C-0
	CHEMIS[7:0]							

Legend:	HC = Hardware Cleared	HS = Hardware Set	
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
S = Settable bit	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 31-0 **CHEMIS[31:0]**: Instruction Cache Miss Count bits
When CHECON.CHEPERF = 1, CHEMIS increments once per iCache or PFB miss.

Note: CHEMIS is reset on 0 to 1 transition of CHECON.CHEPERF.

Note 1: This register is not available on all devices. Refer to the **"Prefetch Module"** chapter in the specific device data sheet to determine availability.

9.4 PREFETCH MODULE OPERATION

The Prefetch module is designed to complement an L1 CPU cache rather than replace it. A single 256-bit (32-byte) line holds instructions or data from the PFM. The Prefetch module uses the Wait states value from the PFMWS[3:0] bits (PRECON[3:0]) and Address wait state PFMAWSEN[7] bit (PRECON[7]) to determine how long it must wait for a Flash access when it reads instructions or data from the PFM. If the instructions or data already reside in a Prefetch module line, the Prefetch module returns the instruction or data in zero Wait states. For CPU instructions, if prefetch is enabled and the code is 100% linear, the Prefetch module will provide instructions back to the CPU with Wait states only on the first instruction of the Prefetch module line.

The line allocated to CPU data and filled by CPU data can be read by a CPU instruction read or a non-CPU peripheral data read. A non-CPU peripheral could be DMA or any other peripheral that has read access to the PFM.

9.5 PREFETCH MODULE CONFIGURATIONS

The PRECON register controls the general configurations available for accelerating instruction and data accesses to the Flash memory system. The Prefetch module implements the following general options:

- The PFMWS[3:0] bits (PRECON[3:0]) control the number of system clock cycles required to access the PFM.
- The PREFEN[1:0] bits (PRECON[5:4]) control Instruction Predictive Prefetched.
- The PFMSECEN bit (PRECON[7]) controls whether the Prefetch module generates an interrupt event on a specific count of single bit errors corrected by the Flash Error Correction Code (ECC).
- The PFMAWSEN bit (PRECON[8]) controls the number of the Wait state.
- The CHEPERFEN bit (PRECON[12]) controls the statistics gathering of the CPU instruction Cache.
- The ICHECOH bit (PRECON[16]) controls the Auto invalidate for the CPU Instruction Cache.
- The DCHECOH bit (PRECON[17]) controls the Auto invalidate for the CPU Data Cache.
- The PERCHECOH bit (PRECON[18]) controls the Auto invalidate for the Peripheral Data Cache.
- The ICHEINV bit (PRECON[20]) controls the Manual invalidate for the CPU Instruction Cache.
- The DCHEINV bit (PRECON[21]) controls the Manual invalidate for the CPU Data Cache.
- The PERCHEINV bit (PRECON[22]) controls the Manual invalidate for the Peripheral Data Cache.
- The ICHEEN bit (PRECON[24]) controls the CPU instruction Cache Enable.
- The DCHEEN bit (PRECON[25]) controls the CPU Data Cache Enable.
- The PRECHEEN bit (PRECON[26]) controls the Peripheral Data Cache Enable.

9.6 PREFETCH MODULE BYPASS BEHAVIOR

When the processor accesses an address with uncacheable cache attributes, the cache is bypassed. In Bypass mode, the module accesses the PFM for every instruction, incurring, and PFMAWSEN bit (PRECON[8]) defines the address setup time, and PFMWS[3:0] bits (PRECON[3:0]) defines a Flash access time. The total Flash wait states are, therefore, the sum of PFMAWSEN and PFMWS.

Bypass mode is also forced for a cache if its associated DCHEFN bit (PRECON[25]) and ICHEFN bit (PRECON[24]) is zero. To allow caching for Instruction and/or Data caches based on the cache attributes, the ICHEFN and/or DCHEFN bit must be set to 1. The peripheral cache does not use the CPU's cache attribute. It only uses the PERCHEFN bit.

9.7 PREFETCH MODULE CACHE BEHAVIOR

Cache lines are aligned to the Flash memory's native read data width. If the CPU-requested address is not aligned, the module aligns the address by dropping the lower address. Data returned on the system is, however, from the actual address requested.

The default cache behavior, while accessing cacheable memory regions, is to load multiple instructions into a line on a cache miss. The Least Recently Used (LRU) algorithm is used to select the line that receives the new set of instructions. On a miss, the cache drives the miss address to the Flash combinatorially when the PFMAWSEN bit (PRECON[8]) is set to 0 or via an address pipeline stage if the PFMAWSEN bit (PRECON[8]) set to 1. Then, the Flash system fills the cache line and returns data to the CPU after the number of wait states selected by the PFMWS[3:0] bits (PRECON[3:0]) is met. On a hit, the cache returns data in zero wait states.

The inclusion of the prefetch buffer (PFB) alters the instruction cache performance. For more information on the enabled prefetch buffer, refer to **Section 9.8 "Prefetch Module Predictive Prefetch Behavior"**.

9.8 PREFETCH MODULE PREDICTIVE PREFETCH BEHAVIOR

When configured for predictive prefetch, the module starts predicting based on the first CPU instruction fetch that hits a cache line. When the first line is placed in the cache, the module increments the address to the next aligned address and starts Flash access. After the PFMAWSEN and PFMWS cycle count, it stores the instructions in the prefetch buffer (PFB). If a miss of the instruction cache and a hit of the PFB buffer occurs, it returns the instruction from the PFB.

The performance of predictive prefetch is dependent on the total Flash wait states of PFMAWSEN and PFMWS. For 100% sequential code, the PFB provides zero wait state execution by utilizing a double buffer ping-pong approach. One buffer stores the instructions from an aligned address and the other buffer fetches instructions from Flash for the next higher aligned address. While one buffer is returning instructions to the CPU, the other buffer gathers the next set.

Predictive prefetching is supported with the instruction cache, and not with the data cache.

9.9 COHERENCY SUPPORT

The Prefetch module supports two methods for coherency control:

- Auto Invalidate
- Manual Invalidate

The user can enable the auto invalidate option for an individual cache or all the caches by setting the ICHECOH bit (PRECON[16]), DCHECOH bit (PRECON[17]) and PERCHECOH bit (PRECON[18]) to 1 and disable for an individual cache or all the caches by setting the ICHECOH bit (PRECON[16]), DCHECOH bit (PRECON[17]) and PERCHECOH bit (PRECON[18]) to 0. The user must take responsibility for invalidating the prefetch cache for all the conditions. Microchip recommends using the auto invalidate option.

The user can change the value of the ICHECOH bit (PRECON[16]), DCHECOH bit (PRECON[17]) and PERCHECOH bit (PRECON[18]) bit any time, but the sampling takes effect only at the beginning of a programming operation. Microchip recommends changing the value before initiating the programming event. Changing the value during a programming event will not cause an invalid for that event. The user needs to invalidate the cache area for the addresses being programmed and has previously read via a cacheable address. Then, the user needs to invalidate the cache area to maintain coherency.

The manual invalidate option is an alternate method to invalidate the cache. If using the manual invalidate option to invalidate an individual cache or all the caches due to the programming event, stop fetching all instruction/data from the desired Flash, set the manual invalidate option, wait for it to clear and, then, start the programming sequence. When using the manual invalidate option to invalidate the individual cache or all the caches, other than programming, it can be set at any time, but only takes effect after completing the pending transactions. The cache

controller stalls new requests until the invalidation succeeds. The Prefetch module auto clears the ICHEINV bit (PRECON[20]), DCHEINV bit (PRECON[21]) and PERCHEINV bit (PRECON[22]) when the invalidation completes. Clear can occur at different times as per cache.

9.10 EFFECTS OF RESET

9.10.1 On Reset

Upon a device Reset, the following occurs:

- All lines are invalidated
- All tag bits are cleared

9.10.2 After Reset

The module operates as per the values in the PRECON register (Register 9-1).

9.11 ERROR CONDITIONS

The Prefetch module handles and reports information about two error types: ECC Double-bit Error Detected (DED) and ECC Single-bit Error Corrected (SEC). The ECC Error detection logic is enabled and disabled using the configuration bits, FECCCON[1:0] (CFGCON0[29:28]). Refer to the **"Special Features"** chapter in the specific device data sheet for information on the CFGCON0 Configuration register.

The ECC logic increases the read access delay from the PFM. Depending on the frequency of the system clock, the wait states may be different between ECC enabled and ECC disabled. Please see the specific device data sheet for Flash access timing specifications for a particular device.

Note:	ECC errors are captured for predictive prefetch reads of the PFM. However, those errors are not reported until, and unless, that data is used by the system.

9.11.1 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 (PRESTAT[27]) is set. The exception handling code can then check this bit to determine whether the exception was caused by a PFM ECC DED event. This bit must be cleared in the software by the exception handler.

Note:	CPU instructions or data prefetched from the PFM will always be loaded into the Prefetch module, even if a DED error is generated. The Prefetch module line containing the DED data will be tagged as valid until the line is replaced.

9.11.2 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 (PRECON[26]). 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 (PRESTAT[7:0]) to decrement by one. If PFMSECCNT[7:0] is zero and a PFM ECC SEC event occurs, the PFMSEC bit (PRESTAT[26]) is set and an interrupt is generated. Therefore, it is recommended that the PFMSECCNT[7:0] bits 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, it is recommended that PFMSECCNT[7:0] is set to four ('00000100'). The Prefetch module does not reload the PFMSECCNT[7:0] bits when it reaches zero. The software must write the desired count each time it services the PFMSEC interrupt.

The software can generate an ECC SEC interrupt by setting the PFMSECEN bit and, then, setting the PFMSEC bit. If the PFMSEC bit is already set when PFMSECEN is set, the Prefetch module will also generate an ECC SEC interrupt. The ECC SEC interrupt persists as long as the PFMSECEN and PFMSEC bits remain set.

9.12 OPERATION IN POWER-SAVING MODES

9.12.1 Sleep Mode

When the device enters Sleep mode, the Prefetch module is disabled and placed into a low-power state where no clocking occurs in the module.

9.12.2 Idle Mode

When the device enters Idle mode, the Prefetch module and its clock source remain functional and the CPU stops executing code. Any outstanding prefetch completes before the Prefetch module stops its clock through automatic clock gating.

9.12.3 Debug Mode

The behavior of the Prefetch module is unaltered in Debug mode.

9.13 REVISION HISTORY

Revision A (September 2020)

This is the initial released version of the document.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods being used in attempts to breach the code protection features of the Microchip devices. We believe that these methods require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Attempts to breach these code protection features, most likely, cannot be accomplished without violating Microchip's intellectual property rights.
- Microchip is willing to work with any customer who is concerned about the integrity of its code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is "unbreakable." Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

ISBN: 978-1-5224-6595-9

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4485-5910
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-72400

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820