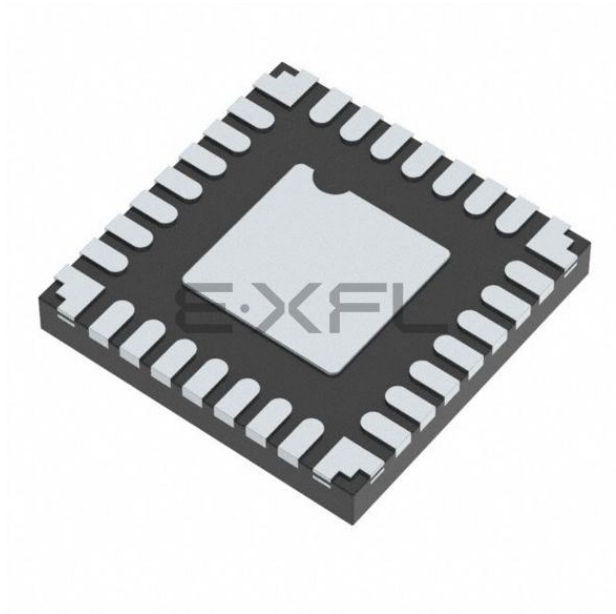
Microchip Technology - PIC32MM0256GPM028T-I/M6 Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	MIPS32® microAptiv™
Core Size	32-Bit Single-Core
Speed	25MHz
Connectivity	IrDA, LINbus, SPI, UART/USART, USB, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, HLVD, I²S, POR, PWM, WDT
Number of I/O	21
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 12x10/12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-UFQFN Exposed Pad
Supplier Device Package	28-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic32mm0256gpm028t-i-m6

3.0 CPU

Note: This data sheet summarizes the features of the PIC32MM0256GPM064 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 50. "CPU for Devices with MIPS32® microAptiv™ and M-Class Cores"** (DS60001192) in the *"PIC32 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com/PIC32). MIPS32® microAptiv™ UC microprocessor core resources are available at: www.imgtec.com. The information in this data sheet supersedes the information in the FRM.

The MIPS32® microAptiv™ UC microprocessor core is the heart of the PIC32MM0256GPM064 family devices. The CPU fetches instructions, decodes each instruction, fetches source operands, executes each instruction and writes the results of the instruction execution to the proper destinations.

3.1 Features

The PIC32MM0256GPM064 family processor core key features include:

- 5-Stage Pipeline
- 32-Bit Address and Data Paths
- MIPS32 Enhanced Architecture:
  - Multiply-add and multiply-subtract instructions.
  - Targeted multiply instruction.
  - Zero and one detect instructions.
  - `WAIT` instruction.
  - Conditional move instructions.
  - Vectored interrupts.
  - Atomic interrupt enable/disable.
  - One GPR shadow set to minimize latency of interrupts.
  - Bit field manipulation instructions.
- microMIPS™ Instruction Set:
  - microMIPS allows improving the code size density over MIPS32, while maintaining MIPS32 performance.
  - microMIPS supports all MIPS32 instructions (except for branch-likely instructions) with new optimized 32-bit encoding. Frequent MIPS32 instructions are available as 16-bit instructions.
  - Added seventeen new and thirty-five MIPS32® corresponding, commonly used instructions in 16-bit opcode format.
  - Stack Pointer implicit in instruction.
  - MIPS32 assembly and ABI compatible.
- Memory Management Unit with Simple Fixed Mapping Translation (FMT) Mechanism
- Multiply/Divide Unit (MDU):
  - Configurable using high-performance multiplier array.
  - Maximum issue rate of one 32x16 multiply per clock.
  - Maximum issue rate of one 32x32 multiply every other clock.
  - Early-in iterative divide. Minimum 11 and maximum 33 clock latency (dividend (rs) sign extension dependent).
- Power Control:
  - No minimum frequency: 0 MHz.
  - Power-Down mode (triggered by `WAIT` instruction).
- EJTAG Debug/Profiling:
  - CPU control with start, stop and single stepping.
  - Software breakpoints via the `SDBBP` instruction.
  - Simple hardware breakpoints on virtual addresses, 4 instruction and 2 data breakpoints.
  - PC and/or load/store address sampling for profiling.
  - Performance counters.
  - Supports Fast Debug Channel (FDC).

A block diagram of the PIC32MM0256GPM064 family processor core is shown in Figure 3-1.

3.2 Architecture Overview

The MIPS32® microAptiv™ UC microprocessor core in the PIC32MM0256GPM064 family devices contains several logic blocks, working together in parallel, providing an efficient high-performance computing engine. The following blocks are included with the core:

- Execution Unit
- General Purpose Register (GPR)
- Multiply/Divide Unit (MDU)
- System Control Coprocessor (CP0)
- Memory Management Unit (MMU)
- Power Management
- microMIPS Instructions Decoder
- Enhanced JTAG (EJTAG) Controller

3.2.1 EXECUTION UNIT

The processor core execution unit implements a load/store architecture with single-cycle ALU operations (logical, shift, add, subtract) and an autonomous Multiply/Divide Unit (MDU). The core contains thirty-two 32-bit General Purpose Registers (GPRs) used for integer operations and address calculation. One additional register file shadow set (containing thirty-two registers) is added to minimize context switching overhead during interrupt/exception processing. The register file consists of two read ports and one write port, and is fully bypassed to minimize operation latency in the pipeline.

The execution unit includes:

- 32-bit adder used for calculating the data address
- Address unit for calculating the next instruction address
- Logic for branch determination and branch target address calculation
- Load aligner
- Bypass multiplexers used to avoid Stalls when executing instruction streams where data producing instructions are followed closely by consumers for their results
- Leading zero/one detect unit for implementing the `CLZ` and `CLO` instructions
- Arithmetic Logic Unit (ALU) for performing arithmetic and bitwise logical operations
- Shifter and store aligner

3.2.2 MULTIPLY/DIVIDE UNIT (MDU)

The microAptiv UC core includes a Multiply/Divide Unit (MDU) that contains a separate pipeline for multiply and divide operations. This pipeline operates in parallel with the Integer Unit (IU) pipeline and does not stall when the IU pipeline stalls. This allows the long-running MDU operations to be partially masked by system Stalls and/or other Integer Unit instructions.

The high-performance MDU consists of a 32x16 booth recoded multiplier, Result/Accumulation registers (HI and LO), a divide state machine, and the necessary multiplexers and control logic. The first number shown ('32' of 32x16) represents the rs operand. The second number ('16' of 32x16) represents the rt operand. The microAptiv UC core only checks the value of the rt operand to determine how many times the operation must pass through the multiplier. The 16x16 and 32x16 operations pass through the multiplier once. A 32x32 operation passes through the multiplier twice.

The MDU supports execution of one 16x16 or 32x16 multiply operation every clock cycle; 32x32 multiply operations can be issued every other clock cycle. Appropriate interlocks are implemented to stall the issuance of back-to-back, 32x32 multiply operations. The multiply operand size is automatically determined by logic built into the MDU. Divide operations are implemented with a simple 1-bit-per-clock iterative algorithm. An early-in detection checks the sign extension of the dividend (rs) operand. If rs is 8 bits wide, 23 iterations are skipped. For a 16-bit wide rs, 15 iterations are skipped, and for a 24-bit wide rs, 7 iterations are skipped. Any attempt to issue a subsequent MDU instruction while a divide is still active causes an IU pipeline Stall until the divide operation has completed.

Table 3-1 lists the repeat rate (peak issue rate of cycles until the operation can be re-issued), and latency (number of cycles until a result is available) for the microAptiv UC core multiply and divide instructions. The approximate latency and repeat rates are listed in terms of pipeline clocks.

TABLE 3-1: MULTIPLY/DIVIDE UNIT LATENCIES AND REPEAT RATES

Opcode	Operand Size (mul *rt*) (div *rs*)	Latency	Repeat Rate
`MULT/MULTU`, `MADD/MADDU`, `MSUB/MSUBU`	16 bits	1	1
	32 bits	2	2
`MUL` (GPR destination)	16 bits	2	1
	32 bits	3	2
`DIV/DIVU`	8 bits	12	11
	16 bits	19	18
	24 bits	26	25
	32 bits	33	32

3.3 Power Management

The processor core offers a number of power management features, including low-power design, active power management and Power-Down modes of operation. The core is a static design that supports slowing or halting of the clocks, which reduces system power consumption during Idle periods.

The mechanism for invoking Power-Down mode is implemented through execution of the WAIT instruction, used to initiate Sleep or Idle. The majority of the power consumed by the processor core is in the clock tree and clocking registers. The PIC32MM family makes extensive use of local gated clocks to reduce this dynamic power consumption.

3.4 EJTAG Debug Support

The microAptiv UC core has an Enhanced JTAG (EJTAG) interface for use in the software debug. In addition to the standard mode of operation, the microAptiv UC core provides a Debug mode that is entered after a debug exception (derived from a hardware breakpoint, single-step exception, etc.) is taken and continues until a Debug Exception Return (DERET) instruction is executed. During this time, the processor executes the debug exception handler routine.

The EJTAG interface operates through the Test Access Port (TAP), a serial communication port used for transferring test data in and out of the microAptiv UC core. In addition to the standard JTAG instructions, special instructions defined in the EJTAG specification specify which registers are selected and how they are used.

3.5 MIPS32® microAptiv™ UC Core Configuration

Register 3-1 through Register 3-4 show the default configuration of the microAptiv UC core, which is included on PIC32MM0256GPM064 family devices.

REGISTER 3-4: CONFIG5: CONFIGURATION REGISTER 5; CP0 REGISTER 16, SELECT 5

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	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
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	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-1
	—	—	—	—	—	—	—	NF

Legend:			
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 31-1 **Unimplemented:** Read as '0'

bit 0 **NF:** Nested Fault bit

1 = Nested Fault feature is implemented

5.0 FLASH PROGRAM MEMORY

Note: This data sheet summarizes the features of the PIC32MM0256GPM064 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 5. "Flash Programming"** (DS60001121) in the *"PIC32 Family Reference Manual"*, which is available from the Microchip web site (www.microchip.com/PIC32). The information in this data sheet supersedes the information in the FRM.

PIC32MM0256GPM064 family devices contain an internal Flash program memory for executing user code. The program and Boot Flash can be write-protected. The erase page size is 512 32-bit words. The program row size is 64 32-bit words. The memory can be programmed by rows or by two 32-bit words, called double-words.

Note: Double-words must be 64-bit aligned.

The devices implement a 6-bit Error Correcting Code (ECC). The memory control block contains a logic to write and read ECC bits to and from the Flash memory. The Flash is programmed at the same time as the corresponding ECC bits. The ECC provides improved resistance to Flash errors. The ECC single-bit error generates an interrupt and can be transparently corrected. The ECC double-bit error results in a bus error exception.

There are three methods by which the user can program this memory:

- Run-Time Self-Programming (RTSP)
- EJTAG Programming
- In-Circuit Serial Programming™ (ICSP™)

RTSP is performed by software executing from either Flash or RAM memory. Information about RTSP techniques is described in **Section 5. "Flash Programming"** (DS60001121) in the *"PIC32 Family Reference Manual"*. EJTAG programming is performed using the JTAG port of the device. ICSP programming requires fewer connections than for EJTAG programming. The EJTAG and ICSP methods are described in the *"PIC32 Flash Programming Specification"* (DS60001145), which is available for download from the Microchip web site.

5.1 Flash Controller Registers Write Protection

The NVMPWP and NVMBWP registers, and the WR bit in the NVMCON register are protected (locked) from an accidental write. Each time a special unlock sequence is required to modify the content of these registers or bits. To unlock, the following steps should be done:

1. Disable interrupts prior to the unlock sequence.
2. Execute the system unlock sequence by writing the key values of 0xAA996655 and 0x556699AA to the NVMKEY register.
3. Write the new value to the required bits.
4. Re-enable interrupts.
5. Relock the system.

Refer to Example 5-1.

EXAMPLE 5-1:

```
// unlock sequence
NVMKEY = AA996655;
NVMKEY = 556699AA;


// relock
NVMKEY = 0;
```

REGISTER 6-3: RNMICON: NON-MASKABLE INTERRUPT (NMI) CONTROL REGISTER(2)

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	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	WDTR
23:16	R/W-0	U-0	U-0	U-0	R/W-0	U-0	R/W-0	R/W-0
	SWNMI	—	—	—	GNMI	—	CF	WDTS
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	NMICNT<15:8>							
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	NMICNT<7:0>							

Legend:			
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 31-25 **Unimplemented:** Read as '0'

bit 24 **WDTR:** Watchdog Timer Time-out Flag bit
1 = A Run mode WDT time-out has occurred and caused an NMI
0 = WDT time-out has not occurred
Setting this bit will cause a WDT NMI event and NMICNT will begin counting.

bit 23 **SWNMI:** Software NMI Trigger bit
1 = An NMI has been generated
0 = An NMI has not been generated

bit 22-20 **Unimplemented:** Read as '0'

bit 19 **GNMI:** Software General NMI Trigger bit
1 = A general NMI has been generated
0 = A general NMI has not been generated

bit 18 **Unimplemented:** Read as '0'

bit 17 **CF:** Clock Fail Detect bit
1 = FSCM has detected clock failure and caused an NMI
0 = FSCM has not detected clock failure
Setting this bit will cause a CF NMI event, but will not cause a clock switch to the FRC.

bit 16 **WDTS:** Watchdog Timer Time-out in Sleep Mode Flag bit
1 = WDT time-out has occurred during Sleep mode and caused a wake-up from Sleep
0 = WDT time-out has not occurred during Sleep mode
Setting this bit will cause a WDT NMI.

bit 15-0 **NMICNT<15:0>:** NMI Reset Counter Value bits
These bits specify the reload value used by the NMI Reset counter.
0xFFFF-0x0001 = Number of SYSCLK cycles before a device Reset occurs(1)
0x0000 = No delay between NMI assertion and device Reset event

Note 1: If a Watchdog Timer NMI event (when not in Sleep or Idle mode) is cleared before this counter reaches '0', no device Reset is asserted. This NMI Reset counter is only applicable to the Watchdog Timer NMI event.

2: The system unlock sequence must be performed before the RNMICON register can be written. Refer to **Section 26.4 "System Registers Write Protection"** for details.

REGISTER 7-7: IPCx: INTERRUPT PRIORITY CONTROL REGISTER x

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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP3<2:0>			IS3<1:0>	
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP2<2:0>			IS2<1:0>	
15:8	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP1<2:0>			IS1<1:0>	
7:0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	IP0<2:0>			IS0<1:0>	

Legend:			
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 31-29 **Unimplemented:** Read as '0'

bit 28-26 **IP3<2:0>:** Interrupt Priority 3 bits
- 111 = Interrupt priority is 7
- •
- •
- •
- 010 = Interrupt priority is 2
- 001 = Interrupt priority is 1
- 000 = Interrupt is disabled

bit 25-24 **IS3<1:0>:** Interrupt Subpriority 3 bits
- 11 = Interrupt subpriority is 3
- 10 = Interrupt subpriority is 2
- 01 = Interrupt subpriority is 1
- 00 = Interrupt subpriority is 0

bit 23-21 **Unimplemented:** Read as '0'

bit 20-18 **IP2<2:0>:** Interrupt Priority 2 bits
- 111 = Interrupt priority is 7
- •
- •
- •
- 010 = Interrupt priority is 2
- 001 = Interrupt priority is 1
- 000 = Interrupt is disabled

bit 17-16 **IS2<1:0>:** Interrupt Subpriority 2 bits
- 11 = Interrupt subpriority is 3
- 10 = Interrupt subpriority is 2
- 01 = Interrupt subpriority is 1
- 00 = Interrupt subpriority is 0

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

Note: This register represents a generic definition of the IPCx register. Refer to Table 7-3 for the exact bit definitions.

NOTES:

REGISTER 8-7: DCHxCON: DMA CHANNEL x CONTROL 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	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	CHBUSY	—	—	—	—	—	—	CHCHNS(1)
7:0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R-0	R/W-0	R/W-0
	CHEN(2)	CHAED	CHCHN	CHAEN	—	CHEDET	CHPRI<1:0>	

Legend:			
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 31-16 **Unimplemented:** Read as '0'

bit 15 **CHBUSY:** Channel Busy bit
- 1 = Channel is active or has been enabled
- 0 = Channel is inactive or has been disabled

bit 14-9 **Unimplemented:** Read as '0'

bit 8 **CHCHNS:** Chain Channel Selection bit(1)
- 1 = Chain to channel lower in natural priority (CH1 will be enabled by CH2 transfer complete)
- 0 = Chain to channel higher in natural priority (CH1 will be enabled by CH0 transfer complete)

bit 7 **CHEN:** Channel Enable bit(2)
- 1 = Channel is enabled
- 0 = Channel is disabled

bit 6 **CHAED:** Channel Allow Events if Disabled bit
- 1 = Channel start/abort events will be registered, even if the channel is disabled
- 0 = Channel start/abort events will be ignored if the channel is disabled

bit **CHCHN:** Channel Chain Enable bit
- 1 = Allows channel to be chained
- 0 = Does not allow channel to be chained

bit 4 **CHAEN:** Channel Automatic Enable bit
- 1 = Channel is continuously enabled and not automatically disabled after a block transfer is complete
- 0 = Channel is disabled on a block transfer complete

bit 3 **Unimplemented:** Read as '0'

bit 2 **CHEDET:** Channel Event Detected bit
- 1 = An event has been detected
- 0 = No events have been detected

bit 1-0 **CHPRI<1:0>:** Channel Priority bits
- 11 = Channel has Priority 3 (highest)
- 10 = Channel has Priority 2
- 01 = Channel has Priority 1
- 00 = Channel has Priority 0

Note 1: The chain selection bit takes effect when chaining is enabled (CHCHN = 1).

2: When the channel is suspended by clearing this bit, the user application should poll the CHBUSY bit (if available on the device variant) to see when the channel is suspended, as it may take some clock cycles to complete a current transaction before the channel is suspended.

REGISTER 18-9: U1EIE: USB ERROR INTERRUPT ENABLE 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	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
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	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BTSEE	BMXEE	DMAEE	BTOEE	DFN8EE	CRC16EE	CRC5EE(1) / EOFEE(2)	PIDEE

Legend:			
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 31-8 **Unimplemented:** Read as '0'

bit 7 **BTSEE:** Bit Stuff Error Interrupt Enable bit
- 1 = BTSEF interrupt is enabled
- 0 = BTSEF interrupt is disabled

bit 6 **BMXEE:** Bus Matrix Error Interrupt Enable bit
- 1 = BMXEF interrupt is enabled
- 0 = BMXEF interrupt is disabled

bit 5 **DMAEE:** DMA Error Interrupt Enable bit
- 1 = DMAEF interrupt is enabled
- 0 = DMAEF interrupt is disabled

bit 4 **BTOEE:** Bus Turnaround Time-out Error Interrupt Enable bit
- 1 = BTOEF interrupt is enabled
- 0 = BTOEF interrupt is disabled

bit 3 **DFN8EE:** Data Field Size Error Interrupt Enable bit
- 1 = DFN8EF interrupt is enabled
- 0 = DFN8EF interrupt is disabled

bit 2 **CRC16EE:** CRC16 Failure Interrupt Enable bit
- 1 = CRC16EF interrupt is enabled
- 0 = CRC16EF interrupt is disabled

bit 1 **CRC5EE:** CRC5 Host Error Interrupt Enable bit(1)
- 1 = CRC5EF interrupt is enabled
- 0 = CRC5EF interrupt is disabled

EOFEE: EOF Error Interrupt Enable bit(2)
- 1 = EOF interrupt is enabled
- 0 = EOF interrupt is disabled

bit 0 **PIDEE:** PID Check Failure Interrupt Enable bit
- 1 = PIDEF interrupt is enabled
- 0 = PIDEF interrupt is disabled

Note 1: Device mode.
2: Host mode.

REGISTER 18-21: U1EP0-U1EP15: USB ENDPOINT CONTROL 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	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
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	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	LSPD	RETRYDIS	—	EPCONDIS	EPRXEN	EPTXEN	EPSTALL	EPHSHK

Legend:			
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 31-8 **Unimplemented:** Read as '0'

bit 7 **LSPD:** Low-Speed Direct Connection Enable bit (Host mode and U1EP0 only)
1 = Direct connection to a low-speed device is enabled
0 = Direct connection to a low-speed device is disabled; hub required with PRE_PID

bit 6 **RETRYDIS:** Retry Disable bit (Host mode and U1EP0 only)
1 = Retry NACK'd transactions are disabled
0 = Retry NACK'd transactions are enabled; retry done in hardware

bit 5 **Unimplemented:** Read as '0'

bit 4 **EPCONDIS:** Bidirectional Endpoint Control bit
If EPTXEN = 1 and EPRXEN = 1:
1 = Disables Endpoint n from control transfers; only TX and RX transfers are allowed
0 = Enables Endpoint n for control (SETUP) transfers; TX and RX transfers are also allowed
Otherwise, this bit is ignored.

bit 3 **EPRXEN:** Endpoint Receive Enable bit
1 = Endpoint n receive is enabled
0 = Endpoint n receive is disabled

bit 2 **EPTXEN:** Endpoint Transmit Enable bit
1 = Endpoint n transmit is enabled
0 = Endpoint n transmit is disabled

bit 1 **EPSTALL:** Endpoint Stall Status bit
1 = Endpoint n was stalled
0 = Endpoint n was not stalled

bit 0 **EPHSHK:** Endpoint Handshake Enable bit
1 = Endpoint handshake is enabled
0 = Endpoint handshake is disabled (typically used for isochronous endpoints)

REGISTER 19-5: RTCDATE/ALMDATE: RTCC DATE/ALARM REGISTERS

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	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	MTHTEN	MTHONE<3:0>			
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	DAYTEN<1:0>		DAYONE<3:0>			
7:0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	—	WDAY<2:0>		

Legend:			
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 31-21 **Unimplemented:** Read as '0'

bit 20 **MTHTEN:** Binary Coded Decimal Value of Months 10-Digit bit
Contains a value from 0 to 1.

bit 19-16 **MTHONE<3:0>:** Binary Coded Decimal Value of Months 1-Digit bits
Contains a value from 0 to 9.

bit 15-14 **Unimplemented:** Read as '0'

bit 13-12 **DAYTEN<1:0>:** Binary Coded Decimal Value of Days 10-Digit bits
Contains a value from 0 to 3.

bit 11-8 **DAYONE<3:0>:** Binary Coded Decimal Value of Days 1-Digit bits
Contains a value from 0 to 9.

bit 7-3 **Unimplemented:** Read as '0'

bit 2-0 **WDAY<2:0>:** Binary Coded Decimal Value of Weekdays Digit bits
Contains a value from 0 to 6.

REGISTER 20-7: AD1CHIT: ADC COMPARE HIT 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	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	—	—	CHH<19:16>(1,2,3)			
15:8	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	—	—	CHH<13:8>					
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	CHH<7:0>							

Legend:			
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 31-20 **Unimplemented:** Read as '0'

bit 19-0 **CHH<21:16>:** ADC Compare Hit bits(1,2,3)

If CM<1:0> = 11:

1 = ADC Result Buffer n has been written with data or a match has occurred

0 = ADC Result Buffer n has not been written with data

For All Other Values of CM<1:0>:

1 = A match has occurred on ADC Result Channel n

0 = No match has occurred on ADC Result Channel n

Note 1: The CHH<19:12> bits are not implemented in 28-pin devices

2: The CHH<19:15> bits are not implemented in 36-pin and 40-pin devices

3: The CHH<17:14> bits are not implemented in 48-pin devices

REGISTER 21-3: CLCxGLS: CLCx GATE LOGIC INPUT SELECT REGISTER (CONTINUED)

bit 20 **G3D3N:** Gate 3 Data Source 3 Negated Enable bit
1 = The Data Source 3 inverted signal is enabled for Gate 3
0 = The Data Source 3 inverted signal is disabled for Gate 3

bit 19 **G3D2T:** Gate 3 Data Source 2 True Enable bit
1 = The Data Source 2 signal is enabled for Gate 3
0 = The Data Source 2 signal is disabled for Gate 3

bit 18 **G3D2N:** Gate 3 Data Source 2 Negated Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 3
0 = The Data Source 2 inverted signal is disabled for Gate 3

bit 17 **G3D1T:** Gate 3 Data Source 1 True Enable bit
1 = The Data Source 1 signal is enabled for Gate 3
0 = The Data Source 1 signal is disabled for Gate 3

bit 16 **G3D1N:** Gate 3 Data Source 1 Negated Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 3
0 = The Data Source 1 inverted signal is disabled for Gate 3

bit 15 **G2D4T:** Gate 2 Data Source 4 True Enable bit
1 = The Data Source 4 signal is enabled for Gate 2
0 = The Data Source 4 signal is disabled for Gate 2

bit 14 **G2D4N:** Gate 2 Data Source 4 Negated Enable bit
1 = The Data Source 4 inverted signal is enabled for Gate 2
0 = The Data Source 4 inverted signal is disabled for Gate 2

bit 13 **G2D3T:** Gate 2 Data Source 3 True Enable bit
1 = The Data Source 3 signal is enabled for Gate 2
0 = The Data Source 3 signal is disabled for Gate 2

bit 12 **G2D3N:** Gate 2 Data Source 3 Negated Enable bit
1 = The Data Source 3 inverted signal is enabled for Gate 2
0 = The Data Source 3 inverted signal is disabled for Gate 2

bit 11 **G2D2T:** Gate 2 Data Source 2 True Enable bit
1 = The Data Source 2 signal is enabled for Gate 2
0 = The Data Source 2 signal is disabled for Gate 2

bit 10 **G2D2N:** Gate 2 Data Source 2 Negated Enable bit
1 = The Data Source 2 inverted signal is enabled for Gate 2
0 = The Data Source 2 inverted signal is disabled for Gate 2

bit 9 **G2D1T:** Gate 2 Data Source 1 True Enable bit
1 = The Data Source 1 signal is enabled for Gate 2
0 = The Data Source 1 signal is disabled for Gate 2

bit 8 **G2D1N:** Gate 2 Data Source 1 Negated Enable bit
1 = The Data Source 1 inverted signal is enabled for Gate 2
0 = The Data Source 1 inverted signal is disabled for Gate 2

bit 7 **G1D4T:** Gate 1 Data Source 4 True Enable bit
1 = The Data Source 4 signal is enabled for Gate 1
0 = The Data Source 4 signal is disabled for Gate 1

bit 6 **G1D4N:** Gate 1 Data Source 4 Negated Enable bit
1 = The Data Source 4 inverted signal is enabled for Gate 1
0 = The Data Source 4 inverted signal is disabled for Gate 1

bit 5 **G1D3T:** Gate 1 Data Source 3 True Enable bit
1 = The Data Source 3 signal is enabled for Gate 1
0 = The Data Source 3 signal is disabled for Gate 1

25.0 POWER-SAVING FEATURES

Note:	This data sheet summarizes the features of the PIC32MM0256GPM064 family of devices. It is not intended to be a comprehensive reference source. To complement the information in this data sheet, refer to **Section 10. "Power-Saving Modes"** (DS60001130) in the *"PIC32 Family "Reference Manual"*, which is available from the Microchip web site (www.microchip.com/PIC32). The information in this data sheet supersedes the information in the FRM.

This section describes the power-saving features for the PIC32MM0256GPM064 family devices. These devices offer various methods and modes that allow the application to balance power consumption with device performance. In all of the methods and modes described in this section, power saving is controlled by software. The peripherals and CPU can be halted or disabled to reduce power consumption.

25.1 Sleep Mode

In Sleep mode, the CPU and most peripherals are halted and the associated clocks are disabled. Some peripherals can continue to operate in Sleep mode and can be used to wake the device from Sleep. See the individual peripheral module sections for descriptions of behavior in Sleep. The device enters Sleep mode when the SLPEN bit (OSCCON<4>) is set and a `WAIT` instruction is executed.

Sleep mode includes the following characteristics:

- There can be a Wake-up Delay based on the Oscillator Selection
- The Fail-Safe Clock Monitor (FSCM) does not Operate During Sleep mode
- The BOR Circuit remains Operative during Sleep mode
- The WDT, if Enabled, is not automatically Cleared prior to Entering Sleep mode
- Some Peripherals can Continue to Operate at Limited Functionality in Sleep mode; these Peripherals include I/O Pins that Detect a Change in the Input Signal, WDT, ADC, UART and Peripherals that use an External Clock Input or the Internal LPRC Oscillator (e.g., RTCC and Timer1)
- I/O Pins Continue to Sink or Source Current in the Same Manner as they do when the Device is not in Sleep

The processor will exit, or "wake-up", from Sleep on one of the following events:

- On any interrupt from an enabled source that is operating in Sleep. The interrupt priority must be greater than the current CPU priority.
- On any form of device Reset.
- On a WDT time-out.

If the interrupt priority is lower than or equal to the current priority, the CPU will remain halted, but the Peripheral Bus Clock (PBCLK) will start running and the device will enter into Idle mode. To set or clear the SLPEN bit, an unlock sequence must be executed. Refer to **Section 26.4 "System Registers Write Protection"** for details.

25.2 Standby Sleep Mode

Standby Sleep mode places the voltage regulator in Standby mode. This mode draws less power than Sleep mode but has a longer wake-up time. Standby Sleep mode is entered by setting the VREGS bit (PWRCON<0>) prior to entering Sleep by executing a `WAIT` instruction. All peripherals that can operate in Sleep mode can operate in Standby Sleep mode.

25.3 Retention Sleep Mode

Retention Sleep uses a separate voltage regulator to provide the lowest power Sleep mode. This mode has a longer wake-up time than Sleep or Standby Sleep. This mode is entered by clearing the RETVR Configuration bit (FPOR<2>) and setting the RETEN bit (PWRCON<1>), prior to entering Sleep mode, and executing a `WAIT` instruction.

Only select peripherals, such as Timer1, WDT, RTCC and REFO, can operate in Retention Sleep mode.

Note:	In Retention mode, the maximum peripheral output frequency to an I/O pin must be less than 33 kHz.

Note:	When $\overline{\text{MCLR}}$ is used to wake the device from Retention Sleep, a POR Reset will occur.

25.4 Idle Mode

In Idle mode, the CPU is halted; however, all clocks are still enabled. This allows peripherals to continue to operate. Peripherals can be individually configured to halt when entering Idle by setting their respective SIDL bit. Latency, when exiting Idle mode, is very low due to the CPU oscillator source remaining active.

The device enters Idle mode when the SLPEN bit (OSCCON<4>) is clear and a `WAIT` instruction is executed.

The processor will wake or exit from Idle mode on the following events:

- On any interrupt event for which the interrupt source is enabled. The priority of the interrupt event must be greater than the current priority of the CPU. If the priority of the interrupt event is lower than or equal to the current priority of the CPU, the CPU will remain halted and the device will remain in Idle mode.
- On any form of device Reset.
- On a WDT time-out interrupt.

To set or clear the SLPEN bit, an unlock sequence must be executed. Refer to **Section 26.4 "System Registers Write Protection"** for details.

TABLE 25-2: PERIPHERAL MODULE DISABLE REGISTERS MAP

Virtual Address (BF80_#)	Register Name(1)	Bit Range	Bits																All Resets
			31/15	30/14	29/13	28/12	27/11	26/10	25/9	24/8	23/7	22/6	21/5	20/4	19/3	18/2	17/1	16/0	
35B0	PMDCON	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FFFF
		15:0	—	—	—	—	PMDLOCK	—	—	—	—	—	—	—	—	—	—	—	F7FF
35C0	PMD1	31:16	—	—	—	—	—	—	—	—	—	—	—	HLVDMD	—	—	—	—	FFEF
		15:0	—	—	—	VREFMD	—	—	—	—	—	—	—	—	—	—	—	ADCMD	EFFE
35D0	PMD2	31:16	—	—	—	—	CLC4MD	CLC3MD	CLC2MD	CLC1MD	—	—	—	—	—	—	—	—	F0FF
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	CMP3MD	CMP2MD	CMP1MD	FFF8
35E0	PMD3	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CCP9MD	FFFE
		15:0	CCP8MD	CCP7MD	CCP6MD	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	—	—	—	—	—	—	—	—	00FF
35F0	PMD4	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FFFF
		15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	T3MD	T2MD	T1MD	FFF8
3600	PMD5	31:16	—	—	—	—	—	—	—	USBMD	—	—	—	—	—	I2C3MD	I2C2MD	I2C1MD	FEF8
		15:0	—	—	—	—	—	SPI3MD	SPI2MD	SPI1MD	—	—	—	—	—	U3MD	U2MD	U1MD	F8F8
3610	PMD6	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FEFF
		15:0	—	—	—	—	—	—	—	REFOMD	—	—	—	—	—	—	—	RTCCMD	FEFE
3620	PMD7	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FFFF
		15:0	—	—	—	—	—	—	—	—	—	—	—	DMAMD	—	—	—	—	FFEF

Legend: — = unimplemented, read as '0'. Reset values are shown in hexadecimal.

Note 1: All registers in this table have corresponding CLR, SET and INV registers at their virtual addresses, plus offsets of 0x4, 0x8 and 0xC, respectively.

REGISTER 26-1: FDEVOPT/AFDEVOPT: DEVICE OPTIONS CONFIGURATION 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	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	USERID<15:8>							
23:16	R/P	R/P	R/P	R/P	R/P	R/P	R/P	R/P
	USERID<7:0>							
15:8	R/P	R/P	r-1	r-1	r-1	r-1	r-1	r-1
	FVBUSIO	FUSBIDIO	—	—	—	—	—	—
7:0	r-1	r-1	r-1	R/P	R/P	r-1	r-1	r-1
	—	—	—	ALTI2C	SOSCHP	—	—	—

Legend:	r = Reserved bit	P = Programmable 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 31-16 **USERID<15:0>:** User ID bits (2 bytes which can programmed to any value)

bit 15 **FVBUSIO:** USB VBUS_ON Selection bit
1 = VBUSON pin is controlled by the USB module
0 = VBUSON pin is controlled by the port function

bit 14 **FUSBIDIO:** USB USBID Selection bit
1 = USBID pin is controlled by the USB module
0 = USBID pin is controlled by the port function

bit 13-5 **Reserved:** Program as '1'

bit 4 **ALTI2C:** Alternate I2C1 Location Select bit
1 = SDA1 and SCL1 are on pins, RB8 and RB9
0 = SDA1 and SCL1 are moved to alternate I^2C locations, RB5 and RC9

bit 3 **SOSCHP:** Secondary Oscillator (SOSC) High-Power Enable bit
1 = SOSC operates in normal power mode
0 = SOSC operates in High-Power mode

bit 2-0 **Reserved:** Program as '1'

REGISTER 26-5: FOSCSEL/AFOSCSEL: OSCILLATOR SELECTION CONFIGURATION REGISTER (CONTINUED)

bit 2-0 **FNOSC<2:0>:** Oscillator Selection bits
110 and 111 = Reserved (selects Fast RC (FRC) Oscillator with Divide-by-N)
101 = Low-Power RC Oscillator (LPRC)
100 = Secondary Oscillator (SOSC)
011 = Reserved
010 = Primary Oscillator (XT, HS, EC)
001 = Primary or FRC Oscillator with PLL
000 = Fast RC Oscillator (FRC) with Divide-by-N

REGISTER 26-6: FSEC/AFSEC: CODE-PROTECT CONFIGURATION 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	R/P	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	CP	—	—	—	—	—	—	—
23:16	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
15:8	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—
7:0	r-1	r-1	r-1	r-1	r-1	r-1	r-1	r-1
	—	—	—	—	—	—	—	—

Legend:	r = Reserved bit	P = Programmable 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 **CP:** Code Protection Enable bit
1 = Code protection is disabled
0 = Code protection is enabled

bit 30-0 **Reserved:** Program as ‘1’

TABLE 29-9: DC CHARACTERISTICS: I/O PIN INPUT INJECTION CURRENT SPECIFICATIONS

DC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C				
Param No.	**Symbol**	**Characteristics**	**Min.**	**Typical(1)**	**Max.**	**Units**	**Conditions**
DI60a	IICL	**Input Low Injection Current**	0	—	-5(2,5)	mA	This parameter applies to all pins. Maximum IICH current for this exception is 0 mA.
DI60b	IICH	**Input High Injection Current**	0	—	+5(3,4,5)	mA	This parameter applies to all pins, with the exception of all 5V tolerant pins and SOSCI. Maximum IICH current for these exceptions is 0 mA.
DI60c	∑IICT	**Total Input Injection Current (sum of all I/O and control pins)**	-20(6)	—	+20(6)	mA	Absolute instantaneous sum of all ± input injection currents from all I/O pins: (| IICL + | IICH |) ≤ ∑IICT

Note 1: Data in the "Typical" column is at 3.3V, +25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

2: VIL Source < (VSS – 0.3). Characterized but not tested.

3: VIH Source > (VDD + 0.3) for non-5V tolerant pins only.

4: Digital 5V tolerant pins do not have an internal high side diode to VDD, and therefore, cannot tolerate any "positive" input injection current.

5: Injection currents > | 0 | can affect the ADC results by approximately 4 to 6 counts (i.e., VIH Source > (VDD + 0.3) or VIL Source < (VSS – 0.3)).

6: Any number and/or combination of I/O pins, not excluded under IICL or IICH conditions, are permitted provided the "absolute instantaneous" sum of the input injection currents from all pins do not exceed the specified limit. If **Note 2**, IICL = (((VSS – 0.3) – VIL Source)/RS). If **Note 3**, IICH = (((IICH Source – (VDD + 0.3))/RS). RS = Resistance between input source voltage and device pin. If (VSS – 0.3) ≤ VSOURCE ≤ (VDD + 0.3), Injection Current = 0.

TABLE 29-23: RESET AND BROWN-OUT RESET REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 2.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
SY10	TMCL	$\overline{\text{MCLR}}$ Pulse Width (Low)	2	—	—	μs	
SY13	TIOZ	I/O High-Impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	—	1	—	μs	Device running or in Idle
SY25	TBOR	Brown-out Reset Pulse Width	1	—	—	μs	VDD ≤ VBOR
SY45	TRST	Internal State Reset Time	—	25	—	μs	
SY71	TPM	Program Memory Wake-up Time	—	22	—	μs	Sleep wake-up with VREGS = 0
			—	3.8	—	μs	Sleep wake-up with VREGS = 1
SY72	TLVR	Low-Voltage Regulator Wake-up Time	—	163	—	μs	Sleep wake-up with VREGS = 0
			—	23	—	μs	Sleep wake-up with VREGS = 1

Note 1: Parameters are for design guidance and are not tested.