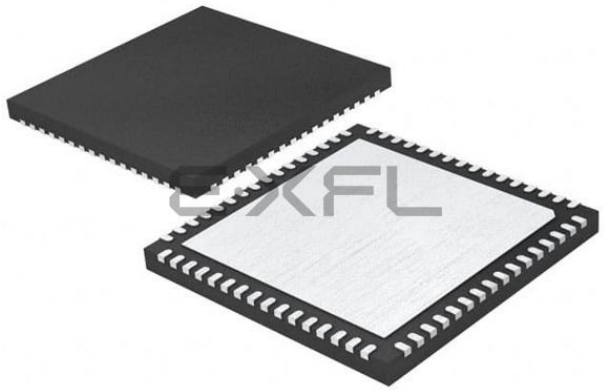
Microchip Technology - DSPIC33FJ32GS406T-50I/MR Datasheet



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	dsPIC
Core Size	16-Bit
Speed	50 MIPs
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, QEI, POR, PWM, WDT
Number of I/O	58
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic33fj32gs406t-50i-mr

TABLE 4-60: PMD REGISTER MAP FOR dsPIC33FJ32GS610 DEVICES

File Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADCMD	0000
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—	I2C2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—	—	—	0000
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—	—	—	0000
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—	—	PWM9MD	0000

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

TABLE 4-61: PMD REGISTER MAP FOR dsPIC33FJ64GS608 DEVICES

File Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	C1MD	ADCMD	0000
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—	I2C2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—	—	—	0000
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—	—	—	0000
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—	—	—	0000

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

TABLE 4-62: PMD REGISTER MAP FOR dsPIC33FJ32GS608 DEVICES

File Name	SFR Addr	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	All Resets
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEI1MD	PWMMD	—	I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	—	—	ADCMD	0000
PMD2	0772	—	—	—	—	IC4MD	IC3MD	IC2MD	IC1MD	—	—	—	—	OC4MD	OC3MD	OC2MD	OC1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	QEI2MD	—	—	—	I2C2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	—	—	—	—	REFOMD	—	—	—	0000
PMD6	077A	PWM8MD	PWM7MD	PWM6MD	PWM5MD	PWM4MD	PWM3MD	PWM2MD	PWM1MD	—	—	—	—	—	—	—	—	0000
PMD7	077C	—	—	—	—	CMP4MD	CMP3MD	CMP2MD	CMP1MD	—	—	—	—	—	—	—	—	0000

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

4.6.2 DATA ACCESS FROM PROGRAM MEMORY USING TABLE INSTRUCTIONS

The TBLRDL and TBLWTL instructions offer a direct method of reading or writing the lower word of any address within the program space without going through data space. The TBLRDH and TBLWTH instructions are the only method to read or write the upper 8 bits of a program space word as data.

The PC is incremented by two for each successive 24-bit program word. This allows program memory addresses to directly map to data space addresses. Program memory can thus be regarded as two 16-bit-wide word address spaces, residing side by side, each with the same address range. TBLRDL and TBLWTL access the space that contains the least significant data word. TBLRDH and TBLWTH access the space that contains the upper data byte.

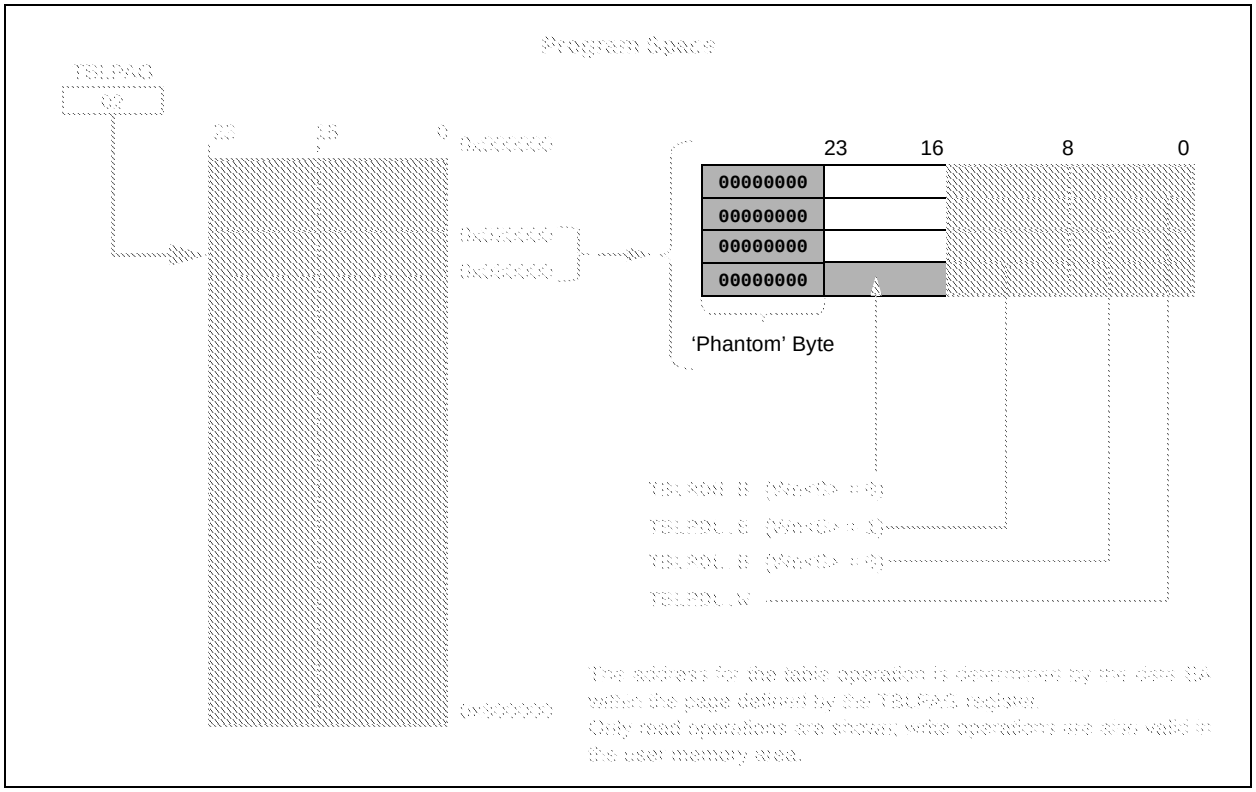
Two table instructions are provided to move byte or word-sized (16-bit) data to and from program space. Both function as either byte or word operations.

- TBLRDL (Table Read Low):
  - In Word mode, this instruction maps the lower word of the program space location (P<15:0>) to a data address (D<15:0>).
  - In Byte mode, either the upper or lower byte of the lower program word is mapped to the lower byte of a data address. The upper byte is selected when Byte Select is '1'; the lower byte is selected when it is '0'.
- TBLRDH (Table Read High):
  - In Word mode, this instruction maps the entire upper word of a program address (P<23:16>) to a data address. Note that D<15:8>, the 'phantom byte', will always be '0'.
  - In Byte mode, this instruction maps the upper or lower byte of the program word to D<7:0> of the data address, in the TBLRDL instruction. The data is always '0' when the upper 'phantom' byte is selected (Byte Select = 1).

Similarly, two table instructions, TBLWTH and TBLWTL, are used to write individual bytes or words to a program space address. The details of their operation are explained in **Section 5.0 "Flash Program Memory"**.

For all table operations, the area of program memory space to be accessed is determined by the Table Page register (TBLPAG). TBLPAG covers the entire program memory space of the device, including user and configuration spaces. When TBLPAG<7> = 0, the table page is located in the user memory space. When TBLPAG<7> = 1, the page is located in configuration space.

FIGURE 4-10: ACCESSING PROGRAM MEMORY WITH TABLE INSTRUCTIONS



5.4.1 PROGRAMMING ALGORITHM FOR FLASH PROGRAM MEMORY

One row of program Flash memory can be programmed at a time. To achieve this, it is necessary to erase the 8-row erase page that contains the desired row. The general process is:

1. Read eight rows of program memory (512 instructions) and store in data RAM.
2. Update the program data in RAM with the desired new data.
3. Erase the block (see Example 5-1):
   a) Set the NVMOPx bits (NVMCON<3:0>) to '0010' to configure for block erase. Set the ERASE (NVMCON<6>) and WREN (NVMCON<14>) bits.
   b) Write the starting address of the page to be erased into the TBLPAG and W registers.
   c) Write 0x55 to NVMKEY.
   d) Write 0xAA to NVMKEY.
   e) Set the WR bit (NVMCON<15>). The erase cycle begins and the CPU stalls for the duration of the erase cycle. When the erase is done, the WR bit is cleared automatically.
4. Write the first 64 instructions from data RAM into the program memory buffers (see Example 5-2).
5. Write the program block to Flash memory:
   a) Set the NVMOPx bits to '0001' to configure for row programming. Clear the ERASE bit and set the WREN bit.
   b) Write 0x55 to NVMKEY.
   c) Write 0xAA to NVMKEY.
   d) Set the WR bit. The programming cycle begins and the CPU stalls for the duration of the write cycle. When the write to Flash memory is done, the WR bit is cleared automatically.
6. Repeat Steps 4 and 5, using the next available 64 instructions from the block in data RAM by incrementing the value in TBLPAG, until all 512 instructions are written back to Flash memory.

For protection against accidental operations, the write initiate sequence for NVMKEY must be used to allow any erase or program operation to proceed. After the programming command has been executed, the user application must wait for the programming time until programming is complete. The two instructions following the start of the programming sequence should be NOPs, as shown in Example 5-3.

EXAMPLE 5-1: ERASING A PROGRAM MEMORY PAGE

```
; Set up NVMCON for block erase operation
       MOV     #0x4042, W0                 ;
       MOV     W0, NVMCON                  ; Initialize NVMCON
; Init pointer to row to be ERASED
       MOV     #tblpage(PROG_ADDR), W0     ;
       MOV     W0, TBLPAG                  ; Initialize PM Page Boundary SFR
       MOV     #tbloffset(PROG_ADDR), W0   ; Initialize in-page EA[15:0] pointer
       TBLWTL  W0, [W0]                    ; Set base address of erase block
       DISI    #5                          ; Block all interrupts with priority <7
                                           ; for next 5 instructions
       MOV     #0x55, W0
       MOV     W0, NVMKEY                  ; Write the 55 key
       MOV     #0xAA, W1                   ;
       MOV     W1, NVMKEY                  ; Write the AA key
       BSET    NVMCON, #WR                 ; Start the erase sequence
       NOP                                 ; Insert two NOPs after the erase
       NOP                                 ; command is asserted
```

6.4 External Reset (EXTR)

The External Reset is generated by driving the MCLR pin low. The MCLR pin is a Schmitt Trigger input with an additional glitch filter. Reset pulses that are longer than the minimum pulse width will generate a Reset. Refer to **Section 27.0 "Electrical Characteristics"** for minimum pulse width specifications. The External Reset (MCLR) pin (EXTR) bit in the Reset Control (RCON) register is set to indicate the MCLR Reset.

6.4.1 EXTERNAL SUPERVISORY CIRCUIT

Many systems have external supervisory circuits that generate Reset signals to reset multiple devices in the system. This external Reset signal can be directly connected to the MCLR pin to reset the device when the rest of system is reset.

6.4.2 INTERNAL SUPERVISORY CIRCUIT

When using the internal power supervisory circuit to reset the device, the External Reset pin (MCLR) should be tied directly or resistively to VDD. In this case, the MCLR pin will not be used to generate a Reset. The External Reset pin (MCLR) does not have an internal pull-up and must not be left unconnected.

6.5 Software RESET Instruction (SWR)

Whenever the RESET instruction is executed, the device will assert SYSRST, placing the device in a special Reset state. This Reset state will not re-initialize the clock. The clock source in effect prior to the RESET instruction will remain. SYSRST is released at the next instruction cycle and the Reset vector fetch will commence.

The Software Reset (SWR) flag (instruction) in the Reset Control (RCON<6>) register is set to indicate the Software Reset.

6.6 Watchdog Timer Time-out Reset (WDTO)

Whenever a Watchdog Timer Time-out Reset occurs, the device will asynchronously assert SYSRST. The clock source will remain unchanged. A WDT time-out during Sleep or Idle mode will wake-up the processor, but will not reset the processor.

The Watchdog Timer Time-out (WDTO) flag in the Reset Control (RCON<4>) register is set to indicate the Watchdog Timer Reset. Refer to **Section 24.4 "Watchdog Timer (WDT)"** for more information on the Watchdog Timer Reset.

6.7 Trap Conflict Reset

If a lower priority hard trap occurs while a higher priority trap is being processed, a hard Trap Conflict Reset occurs. The hard traps include exceptions of Priority Level 13 through Level 15, inclusive. The address error (Level 13) and oscillator error (Level 14) traps fall into this category.

The Trap Reset (TRAPR) flag in the Reset Control (RCON<15>) register is set to indicate the Trap Conflict Reset. Refer to **Section 7.0 "Interrupt Controller"** for more information on Trap Conflict Resets.

6.8 Illegal Condition Device Reset

An illegal condition device Reset occurs due to the following sources:

- Illegal Opcode Reset
- Uninitialized W Register Reset
- Security Reset

The Illegal Opcode or Uninitialized W Access Reset (IOPUWR) flag in the Reset Control (RCON<14>) register is set to indicate the illegal condition device Reset.

6.8.1 ILLEGAL OPCODE RESET

A device Reset is generated if the device attempts to execute an illegal opcode value that is fetched from program memory.

The Illegal Opcode Reset function can prevent the device from executing program memory sections that are used to store constant data. To take advantage of the Illegal Opcode Reset, use only the lower 16 bits of each program memory section to store the data values. The upper 8 bits should be programmed with 3Fh, which is an illegal opcode value.

6.8.2 UNINITIALIZED W REGISTER RESET

Any attempt to use the Uninitialized W register as an Address Pointer will reset the device. The W register array (with the exception of W15) is cleared during all Resets and is considered uninitialized until written to.

6.8.3 SECURITY RESET

If a Program Flow Change (PFC) or Vector Flow Change (VFC) targets a restricted location in a protected segment (Boot and Secure Segment), that operation will cause a Security Reset.

The PFC occurs when the Program Counter is reloaded as a result of a call, jump, computed jump, return, return from subroutine or other form of branch instruction.

The VFC occurs when the Program Counter is reloaded with an interrupt or trap vector.

Refer to **Section 24.8 "Code Protection and CodeGuard™ Security"** for more information on Security Reset.

REGISTER 7-11: IFS6: INTERRUPT FLAG STATUS REGISTER 6

R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
ADCP1IF	ADCP0IF	—	—	—	—	AC4IF	AC3IF
bit 15							bit 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
AC2IF	PWM9IF	PWM8IF	PWM7IF	PWM6IF	PWM5IF	PWM4IF	PWM3IF
bit 7							bit 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 15 **ADCP1IF:** ADC Pair 1 Conversion Done Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 14 **ADCP0IF:** ADC Pair 0 Conversion Done Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 13-10 **Unimplemented:** Read as '0'

bit 9 **AC4IF:** Analog Comparator 4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 8 **AC3IF:** Analog Comparator 3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 7 **AC2IF:** Analog Comparator 2 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 6 **PWM9IF:** PWM9 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 5 **PWM8IF:** PWM8 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 4 **PWM7IF:** PWM7 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 3 **PWM6IF:** PWM6 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 2 **PWM5IF:** PWM5 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 1 **PWM4IF:** PWM4 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

bit 0 **PWM3IF:** PWM3 Interrupt Flag Status bit
1 = Interrupt request has occurred
0 = Interrupt request has not occurred

REGISTER 7-21: IPC0: INTERRUPT PRIORITY CONTROL REGISTER 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP2	T1IP1	T1IP0	—	OC1IP2	OC1IP1	OC1IP0
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC1IP2	IC1IP1	IC1IP0	—	INT0IP2	INT0IP1	INT0IP0
bit 7							bit 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 15 **Unimplemented:** Read as '0'

bit 14-12 **T1IP<2:0>:** Timer1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 10-8 **OC1IP<2:0>:** Output Compare Channel 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 6-4 **IC1IP<2:0>:** Input Capture Channel 1 Interrupt Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

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

bit 2-0 **INT0IP<2:0>:** External Interrupt 0 Priority bits
111 = Interrupt is Priority 7 (highest priority interrupt)
•
•
•
001 = Interrupt is Priority 1
000 = Interrupt source is disabled

REGISTER 8-6: DMAxCNT: DMA CHANNEL x TRANSFER COUNT REGISTER(1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CNT<9:8>(2)	
bit 15							

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CNT<7:0>(2)							
bit 7							bit 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 15-10 **Unimplemented:** Read as '0'

bit 9-0 **CNT<9:0>:** DMA Transfer Count Register bits(2)

Note 1: If the channel is enabled (i.e., active), writes to this register may result in unpredictable behavior of the DMA channel and should be avoided.

2: Number of DMA transfers = CNT<9:0> + 1.

NOTES:

REGISTER 16-11: PWMCONx: PWM CONTROL x REGISTER (CONTINUED)

bit 7-6 **DTC<1:0>:** Dead-Time Control bits
11 = Dead-Time Compensation mode
10 = Dead-time function is disabled
01 = Negative dead time is actively applied for Complementary Output mode
00 = Positive dead time is actively applied for all output modes

bit 5 **DTCP:** Dead-Time Compensation Polarity bit(4)
1 = If DTCMPx = 0, PWMxL is shortened and PWMxH is lengthened;
If DTCMPx = 1, PWMxH is shortened and PWMxL is lengthened
0 = If DTCMPx = 0, PWMxH is shortened and PWMLx is lengthened;
If DTCMPx = 1, PWMxL is shortened and PWMxH is lengthened

bit 4 **Unimplemented:** Read as '0'

bit 3 **MTBS:** Master Time Base Select bit
1 = PWM generator uses the secondary master time base for synchronization and the clock source for the PWM generation logic (if secondary time base is available)
0 = PWM generator uses the primary master time base for synchronization and the clock source for the PWM generation logic

bit 2 **CAM:** Center-Aligned Mode Enable bit(2,3,5)
1 = Center-Aligned mode is enabled
0 = Edge-Aligned mode is enabled

bit 1 **XPRES:** External PWM Reset Control bit(6)
1 = Current-limit source resets the time base for this PWM generator if it is in Independent Time Base mode
0 = External pins do not affect PWM time base

bit 0 **IUE:** Immediate Update Enable bit
1 = Updates to the active MDC/PDCx/SDCx registers are immediate
0 = Updates to the active PDCx registers are synchronized to the PWM time base

Note 1: Software must clear the interrupt status here and in the corresponding IFSx bit in the interrupt controller.
2: The Independent Time Base mode (ITB = 1) must be enabled to use Center-Aligned mode. If ITB = 0, the CAM bit is ignored.
3: These bits should not be changed after the PWM is enabled by setting PTEN (PTCON<15>) = 1.
4: For DTCP to be effective, DTC<1:0> must be set to '11'; otherwise, DTCP is ignored.
5: Center-Aligned mode ignores the Least Significant 3 bits of the Duty Cycle, Phase and Dead-Time registers. The highest Center-Aligned mode resolution available is 8.32 ns with the clock prescaler set to the fastest clock.
6: Configure CLMOD (FCLCONX<8>) = 0 and ITB (PWMCONx<9>) = 1 to operate in External Period Reset mode.

REGISTER 20-1: UxMODE: UARTx MODE REGISTER (CONTINUED)

bit 4 **URXINV:** Receive Polarity Inversion bit
1 = UxRX Idle state is '0'
0 = UxRX Idle state is '1'

bit 3 **BRGH:** High Baud Rate Enable bit
1 = BRG generates 4 clocks per bit period (4x baud clock, High-Speed mode)
0 = BRG generates 16 clocks per bit period (16x baud clock, Standard mode)

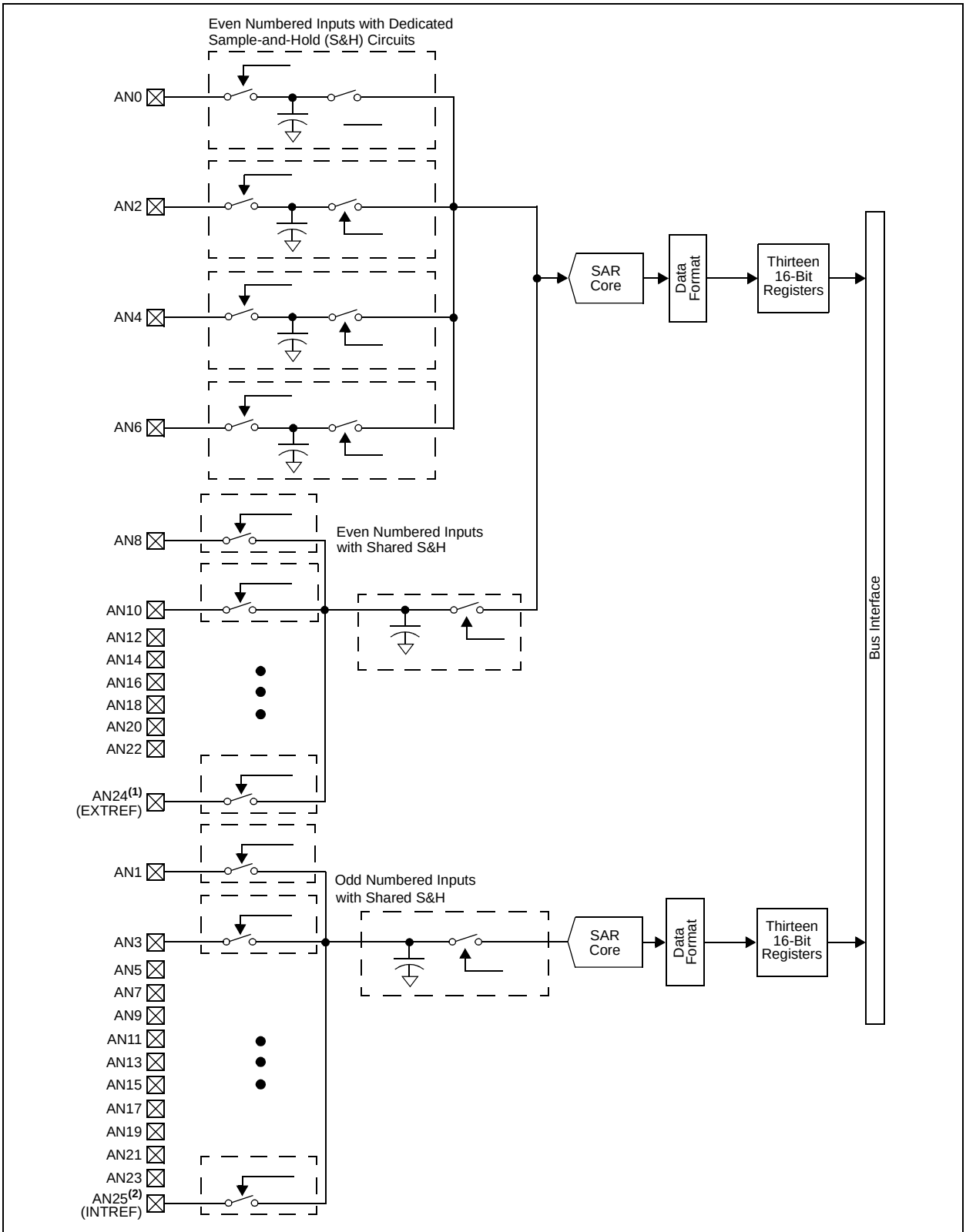
bit 2-1 **PDSEL<1:0>:** Parity and Data Selection bits
11 = 9-bit data, no parity
10 = 8-bit data, odd parity
01 = 8-bit data, even parity
00 = 8-bit data, no parity

bit 0 **STSEL:** Stop Bit Selection bit
1 = Two Stop bits
0 = One Stop bit

Note 1: Refer to **"UART"** (DS70188) in the *"dsPIC33/PIC24 Family Reference Manual"* for information on enabling the UART module for receive or transmit operation. That section of the manual is available on the Microchip web site: www.microchip.com.

2: This feature is only available for the 16x BRG mode (BRGH = 0).

FIGURE 22-4: ADC BLOCK DIAGRAM FOR dsPIC33FJ32GS610 AND dsPIC33FJ64GS610 DEVICES WITH TWO SARs



Note 1: AN24 (EXTREF) is an internal analog input. To measure the voltage at AN24 (EXTREF), an analog comparator must be enabled and EXTREF must be selected as the comparator reference.

2: AN25 (INTREF) is an internal analog input and is not available on a pin.

REGISTER 22-10: ADCPC4: ADC CONVERT PAIR CONTROL REGISTER 4 (CONTINUED)

bit 12-8 **TRGSRC9<4:0>:** Trigger 9 Source Selection bits

Selects trigger source for conversion of analog channels AN19 and AN18.
11111 = Timer2 period match
11110 = PWM Generator 8 current-limit ADC trigger
11101 = PWM Generator 7 current-limit ADC trigger
11100 = PWM Generator 6 current-limit ADC trigger
11011 = PWM Generator 5 current-limit ADC trigger
11010 = PWM Generator 4 current-limit ADC trigger
11001 = PWM Generator 3 current-limit ADC trigger
11000 = PWM Generator 2 current-limit ADC trigger
10111 = PWM Generator 1 current-limit ADC trigger
10110 = PWM Generator 9 secondary trigger selected
10101 = PWM Generator 8 secondary trigger selected
10100 = PWM Generator 7 secondary trigger selected
10011 = PWM Generator 6 secondary trigger selected
10010 = PWM Generator 5 secondary trigger selected
10001 = PWM Generator 4 secondary trigger selected
10000 = PWM Generator 3 secondary trigger selected
01111 = PWM Generator 2 secondary trigger selected
01110 = PWM Generator 1 secondary trigger selected
01101 = PWM secondary Special Event Trigger selected
01100 = Timer1 period match
01011 = PWM Generator 8 primary trigger selected
01010 = PWM Generator 7 primary trigger selected
01001 = PWM Generator 6 primary trigger selected
01000 = PWM Generator 5 primary trigger selected
00111 = PWM Generator 4 primary trigger selected
00110 = PWM Generator 3 primary trigger selected
00101 = PWM Generator 2 primary trigger selected
00100 = PWM Generator 1 primary trigger selected
00011 = PWM Special Event Trigger selected
00010 = Global software trigger selected
00001 = Individual software trigger selected
00000 = No conversion is enabled

bit 7 **IRQEN8:** Interrupt Request Enable 8 bit

1 = Enables IRQ generation when requested conversion of Channels AN17 and AN16 is completed
0 = IRQ is not generated

bit 6 **PEND8:** Pending Conversion Status 8 bit

1 = Conversion of Channels AN17 and AN16 is pending; set when selected trigger is asserted
0 = Conversion is complete

bit 5 **SWTRG8:** Software Trigger 8 bit

1 = Starts conversion of AN17 and AN16 (if selected by TRGSRC bits)**(1)**
This bit is automatically cleared by hardware when the PEND8 bit is set.
0 = Conversion has not started

Note 1: The trigger source must be set as an individual software trigger prior to setting this bit to '1'. If other conversions are in progress, the conversion is performed when the conversion resources are available.

TABLE 25-1: SYMBOLS USED IN OPCODE DESCRIPTIONS (CONTINUED)

Field	Description
Wm*Wm	Multiplicand and Multiplier Working register pair for Square instructions ∈ {W4 * W4,W5 * W5,W6 * W6,W7 * W7}
Wm*Wn	Multiplicand and Multiplier Working register pair for DSP instructions ∈ {W4 * W5,W4 * W6,W4 * W7,W5 * W6,W5 * W7,W6 * W7}
Wn	One of 16 Working registers ∈ {W0..W15}
Wnd	One of 16 Destination Working registers ∈ {W0...W15}
Wns	One of 16 Source Working registers ∈ {W0...W15}
WREG	W0 (Working register used in file register instructions)
Ws	Source W register ∈ { Ws, [Ws], [Ws++], [Ws--], [++Ws], [--Ws] }
Wso	Source W register ∈ { Wns, [Wns], [Wns++], [Wns--], [++Wns], [--Wns], [Wns+Wb] }
Wx	X Data Space Prefetch Address register for DSP instructions ∈ {[W8] + = 6, [W8] + = 4, [W8] + = 2, [W8], [W8] - = 6, [W8] - = 4, [W8] - = 2, [W9] + = 6, [W9] + = 4, [W9] + = 2, [W9], [W9] - = 6, [W9] - = 4, [W9] - = 2, [W9 + W12], none}
Wxd	X Data Space Prefetch Destination register for DSP instructions ∈ {W4...W7}
Wy	Y Data Space Prefetch Address register for DSP instructions ∈ {[W10] + = 6, [W10] + = 4, [W10] + = 2, [W10], [W10] - = 6, [W10] - = 4, [W10] - = 2, [W11] + = 6, [W11] + = 4, [W11] + = 2, [W11], [W11] - = 6, [W11] - = 4, [W11] - = 2, [W11 + W12], none}
Wyd	Y Data Space Prefetch Destination register for DSP instructions ∈ {W4...W7}

FIGURE 27-2: EXTERNAL CLOCK TIMING

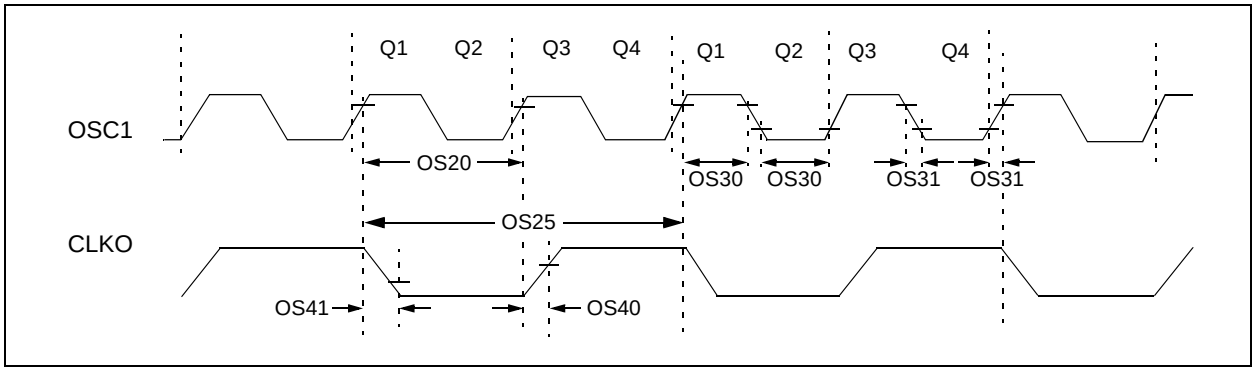


TABLE 27-16: EXTERNAL CLOCK TIMING REQUIREMENTS

AC CHARACTERISTICS			Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Symb**	**Characteristic**	**Min**	**Typ(1)**	**Max**	**Units**	**Conditions**
OS10	FIN	External CLKI Frequency (external clocks allowed only in EC and ECPLL modes)	DC	—	40	MHz	EC
		Oscillator Crystal Frequency	3.5 — 10	— — —	10 33 40	MHz kHz MHz	XT SOSC HS
OS20	TOSC	TOSC = 1/FOSC	12.5	—	DC	ns	
OS25	TCY	Instruction Cycle Time(2)	25	—	DC	ns	
OS30	TosL, TosH	External Clock in (OSC1) High or Low Time	0.375 x TOSC	—	0.625 x TOSC	ns	EC
OS31	TosR, TosF	External Clock in (OSC1) Rise or Fall Time	—	—	20	ns	EC
OS40	TckR	CLKO Rise Time(3)	—	5.2	—	ns	
OS41	TckF	CLKO Fall Time(3)	—	5.2	—	ns	
OS41	GM	External Oscillator Transconductance	14	16	18	mA/V	VDD = 3.3V, TA = +25ºC

Note 1: Data in “Typ” column is at 3.3V, +25°C unless otherwise stated.

2: Instruction cycle period (TCY) equals two times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type, under standard operating conditions, with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “Min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “Max.” cycle time limit is “DC” (no clock) for all devices.

3: Measurements are taken in EC mode. The CLKO signal is measured on the OSC2 pin.

TABLE 27-19: AC CHARACTERISTICS: INTERNAL FRC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for Extended					
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**	
Internal FRC Accuracy @ FRC Frequency = 7.37 MHz(1)							
F20a	FRC	-1	—	+1	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F20b	FRC	-2	—	+2	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

Note 1: Frequency calibrated at +25°C and 3.3V. The TUN<5:0> bits can be used to compensate for temperature drift.

TABLE 27-20: AC CHARACTERISTICS: INTERNAL LPRC ACCURACY

AC CHARACTERISTICS		Standard Operating Conditions: 3.0V to 3.6V (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for Industrial -40°C ≤ TA ≤ +125°C for Extended				
Param No.	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
LPRC @ 32.768 kHz(1)						
F21a	LPRC	-40	—	+40	%	-40°C ≤ TA ≤ +85°C
F21b	LPRC	-50	—	+50	%	-40°C ≤ TA ≤ +125°C

Note 1: Change of LPRC frequency as VDD changes.

FIGURE 27-21: I2Cx BUS START/STOP BITS TIMING CHARACTERISTICS (SLAVE MODE)

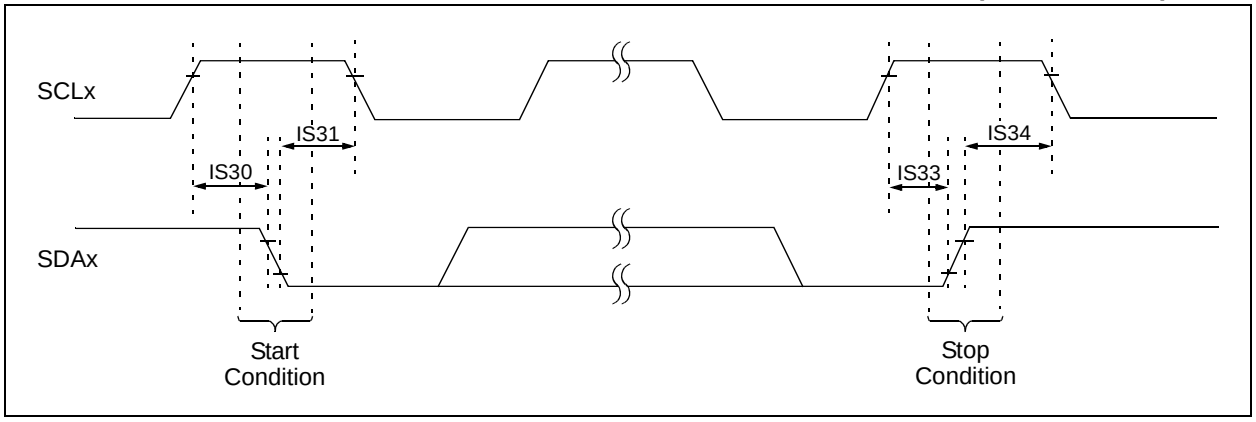


FIGURE 27-22: I2Cx BUS DATA TIMING CHARACTERISTICS (SLAVE MODE)

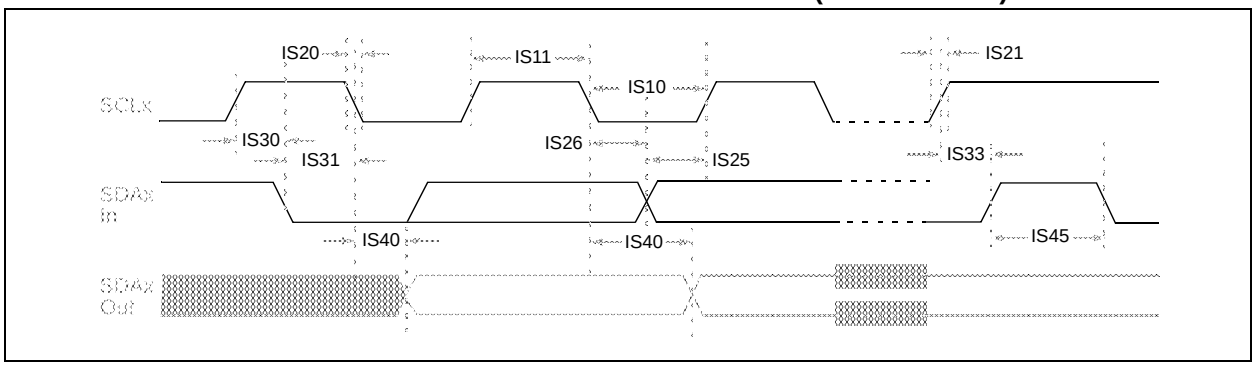


FIGURE 29-4: VOL – 4x DRIVER PINS

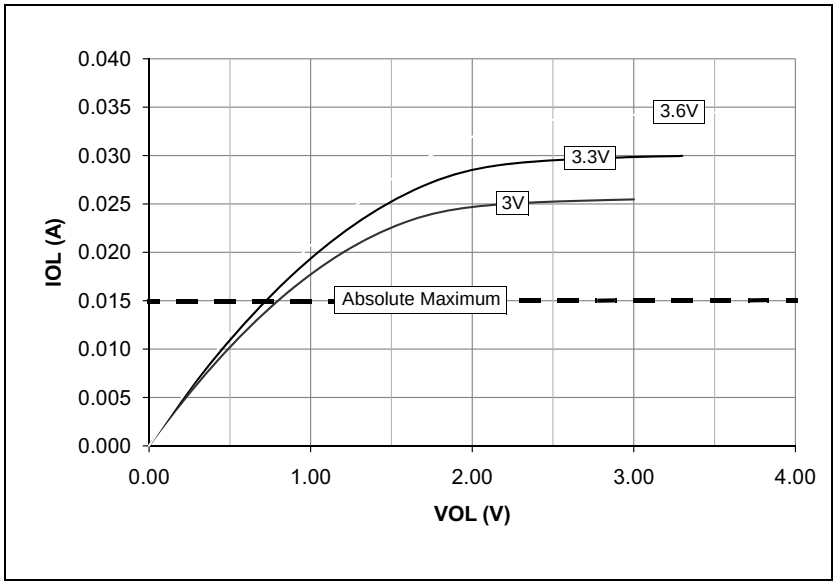


FIGURE 29-5: VOL – 8x DRIVER PINS

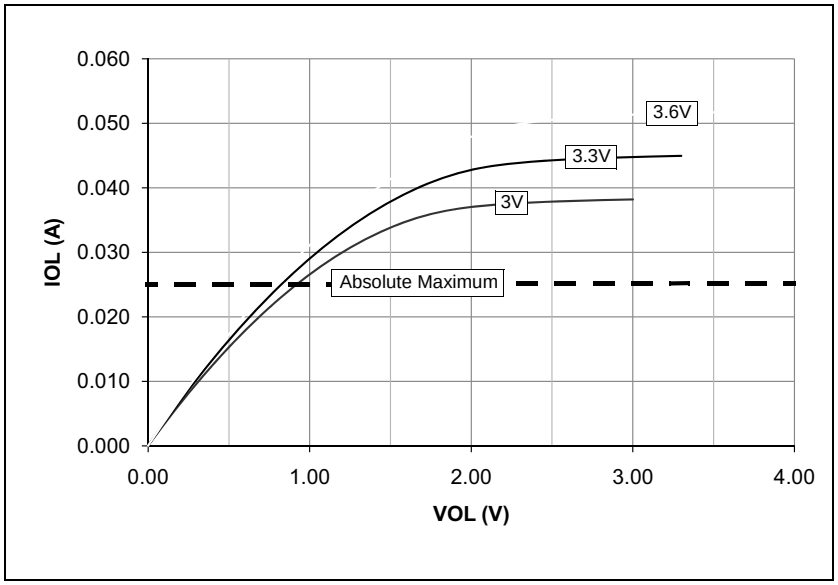
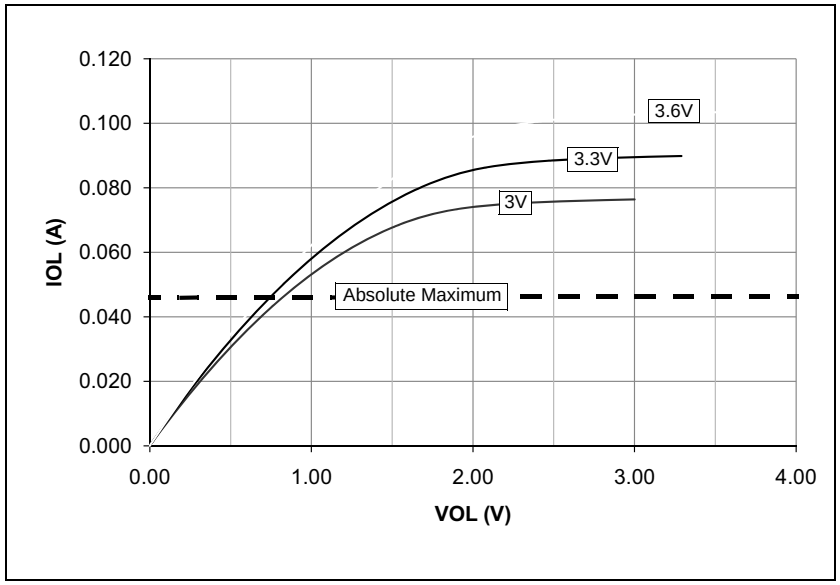
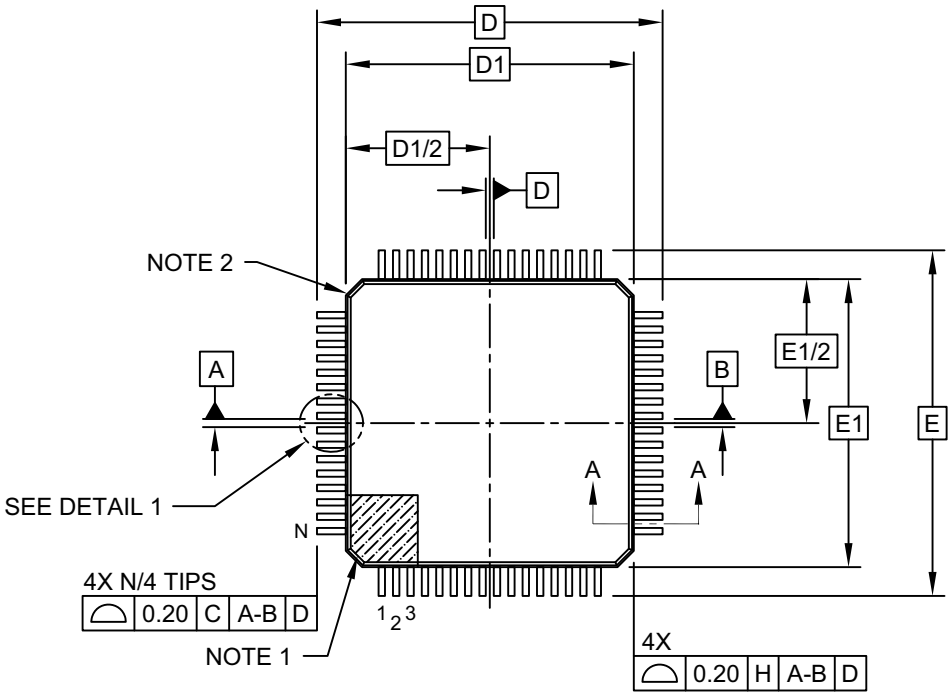


FIGURE 29-6: VOL – 16x DRIVER PINS

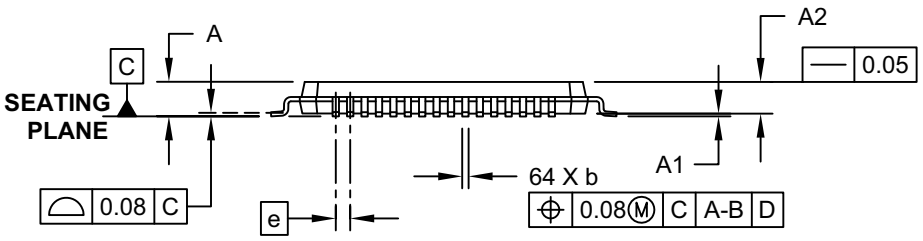


64-Lead Plastic Thin Quad Flatpack (PT)-10x10x1 mm Body, 2.00 mm Footprint [TQFP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW



SIDE VIEW

Microchip Technology Drawing C04-085C Sheet 1 of 2

TABLE B-3: MAJOR SECTION UPDATES (CONTINUED)

Section Name	Update Description
Section 22.0 "High-Speed, 10-Bit Analog-to-Digital Converter (ADC)"	Updated the ADC Block Diagram for dsPIC33FJ32GS406 and dsPIC33FJ64GS406 Devices with one SAR (see Table 22-1). Added Note 2 to ADCPC6: ADC Convert Pair Control Register 6 (see Register 22-12).
Section 23.0 "High-Speed Analog Comparator"	Added Note 1 to the High-Speed Analog Comparator Module block diagram (see Figure 23-1).
Section 24.0 "Special Features"	Updated **Section 24.1 "Configuration Bits"**. Added the RTSP Effect column to the dsPIC33F Configuration Bits Description (see Table 24-2). Added Note 3 to the Connections for the On-chip Voltage Regulator (see Figure 24-1).
Section 27.0 "Electrical Characteristics"	Updated the Absolute Maximum Ratings. Updated the Operating MIPS vs. Voltage and added Note 1 (see Table 27-1). Updated Note 4 and removed parameter DC18 from the DC Temperature and Voltage Specifications (see Table 27-4). Updated Note 2, Typical and Maximum values for parameters DC20-DC24, and the Conditions for parameters DC25-DC28 in the Operating Current DC Characteristics (see Table 27-5). Updated Note 2 in the Idle Current DC Characteristics (see Table 27-6). Updated Note 2 in the Power-down Current DC Characteristics (see Table 27-7). Added Note 2 to the Doze Current DC Characteristics (see Table 27-8). Added parameters DI60a, DI60b, and DI60c to the I/O Pin Input Specifications (see Table 27-9). Updated all I/O Pin Output Specifications (see Table 27-10). Updated parameter BO10 and added Note 2 and Note 3 to the BOR Electrical Characteristics (see Table 27-11). Added Note 1 to the Internal Voltage Regulator Specifications (see Table 27-13). Updated the OS25 parameter in the External Clock Timing diagram (see Figure 27-2). Added the Secondary Oscillator (SOSC) to parameter OS10, added parameter OS42 (GM), and added Note 2 to the External Clock Timing Requirements (see Table 27-16). Updated Note 2 in the Internal FRC Accuracy AC Characteristics (see Table 27-19). Updated parameters DO31 and DO32 in the I/O Timing Requirements (see Table 27-21).

NOTES: