



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	20 MIPS
Connectivity	CANbus, I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, Motor Control PWM, QEI, POR, PWM, WDT
Number of I/O	30
Program Memory Size	48KB (16K x 24)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 9x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/dspic30f4011-20i-pt

dsPIC30F4011/4012

FIGURE 3-6: dsPIC30F4011/4012 DATA SPACE MEMORY MAP

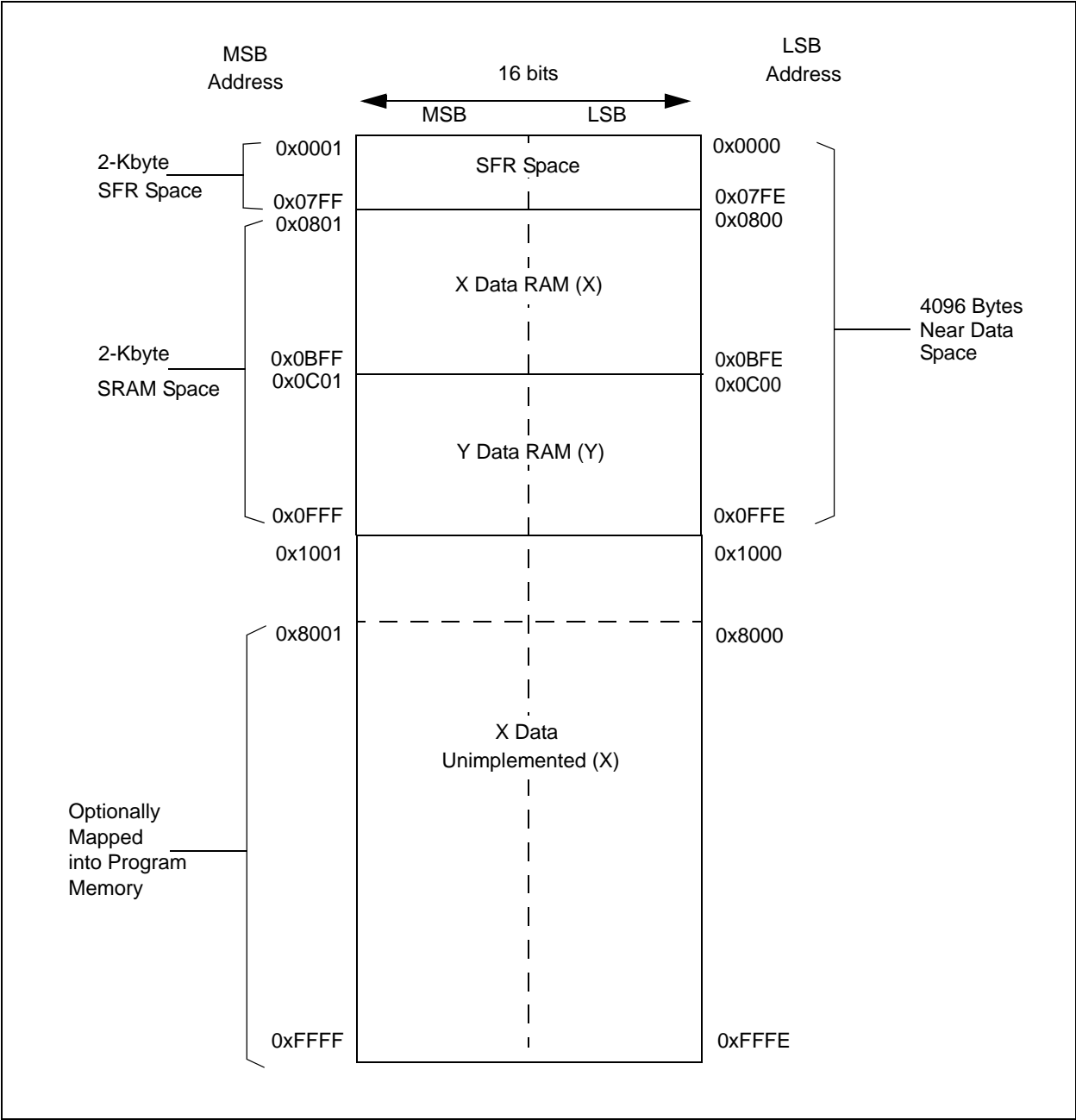
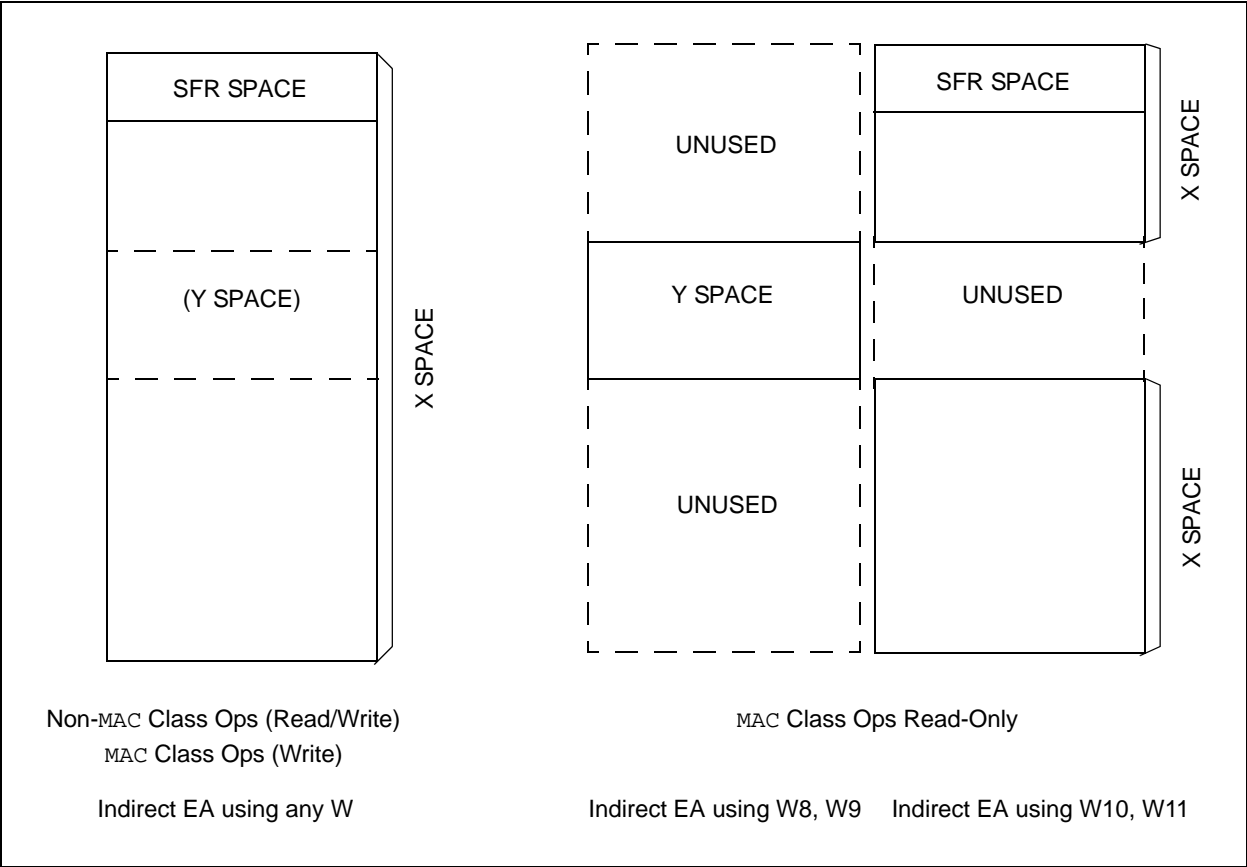


FIGURE 3-7: DATA SPACE FOR MCU AND DSP (MAC CLASS) INSTRUCTIONS EXAMPLE



4.2.3 MODULO ADDRESSING APPLICABILITY

Modulo Addressing can be applied to the Effective Address (EA) calculation associated with any W register. It is important to realize that the address boundaries check for addresses less than or greater than the upper (for incrementing buffers) and lower (for decrementing buffers) boundary addresses (not just equal to). Address changes may, therefore, jump beyond boundaries and still be adjusted correctly.

Note: The modulo corrected Effective Address is written back to the register only when Pre-Modify or Post-Modify Addressing mode is used to compute the Effective Address. When an address offset (e.g., [W7+W2]) is used, Modulo Addressing correction is performed, but the contents of the register remain unchanged.

4.3 Bit-Reversed Addressing

Bit-Reversed Addressing is intended to simplify data reordering for radix-2 FFT algorithms. It is supported by the X AGU for data writes only.

The modifier, which may be a constant value or register contents, is regarded as having its bit order reversed. The address source and destination are kept in normal order. Thus, the only operand requiring reversal is the modifier.

4.3.1 BIT-REVERSED ADDRESSING IMPLEMENTATION

Bit-Reversed Addressing is enabled when:

1. BWM (W register selection) in the MODCON register is any value other than 15 (the stack can not be accessed using Bit-Reversed Addressing) and
2. The BREN bit is set in the XBREV register and
3. The addressing mode used is Register Indirect with Pre-Increment or Post-Increment.

If the length of a bit-reversed buffer is $M = 2^N$ bytes, then the last 'N' bits of the data buffer start address must be zeros.

XB<14:0> is the Bit-Reversed Addressing modifier, or 'pivot point', which is typically a constant. In the case of an FFT computation, its value is equal to half of the FFT data buffer size.

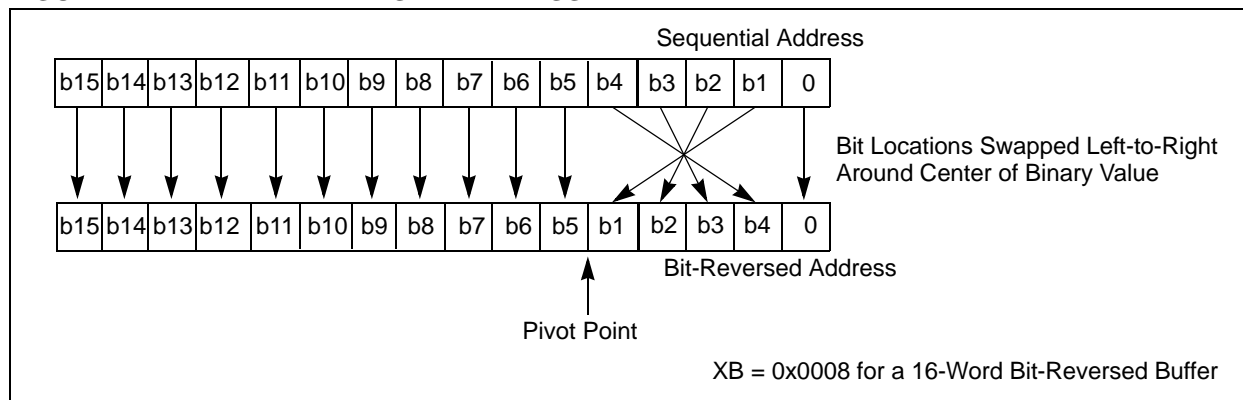
Note: All bit-reversed EA calculations assume word-sized data (LSb of every EA is always clear). The XB value is scaled accordingly to generate compatible (byte) addresses.

When enabled, Bit-Reversed Addressing is only executed for Register Indirect with Pre-Increment or Post-Increment Addressing mode and word-sized data writes. It will not function for any other addressing mode or for byte-sized data, and normal addresses will be generated instead. When Bit-Reversed Addressing is active, the W Address Pointer will always be added to the address modifier (XB) and the offset associated with the Register Indirect Addressing mode will be ignored. In addition, as word-sized data is a requirement, the LSb of the EA is ignored (and always clear).

Note: Modulo Addressing and Bit-Reversed Addressing should not be enabled together. In the event that the user attempts to do this, Bit-Reversed Addressing will assume priority when active for the X WAGU, and X WAGU Modulo Addressing will be disabled. However, Modulo Addressing will continue to function in the X RAGU.

If Bit-Reversed Addressing has already been enabled by setting the BREN bit (XBREV<15>), then a write to the XBREV register should not be immediately followed by an indirect read operation using the W register that has been designated as the Bit-Reversed Pointer.

FIGURE 4-2: BIT-REVERSED ADDRESS EXAMPLE



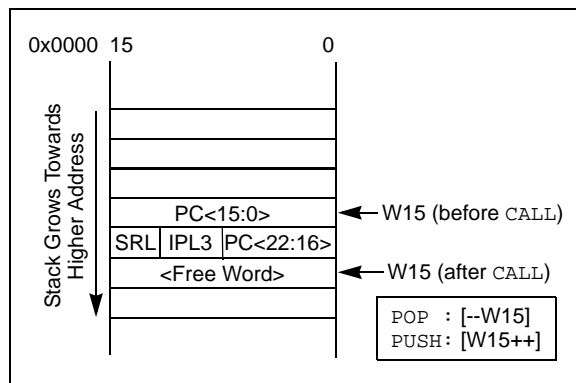
5.4 Interrupt Sequence

All interrupt event flags are sampled in the beginning of each instruction cycle by the IFSx registers. A pending interrupt request (IRQ) is indicated by the flag bit being equal to a '1' in an IFSx register. The IRQ will cause an interrupt to occur if the corresponding bit in the interrupt enable (IECx) register is set. For the remainder of the instruction cycle, the priorities of all pending interrupt requests are evaluated.

If there is a pending IRQ with a priority level greater than the current processor priority level in the IPL bits, the processor will be interrupted.

The processor then stacks the current program counter and the low byte of the processor STATUS register (SRL), as shown in Figure 5-2. The low byte of the STATUS register contains the processor priority level at the time prior to the beginning of the interrupt cycle. The processor then loads the priority level for this interrupt into the STATUS register. This action will disable all lower priority interrupts until the completion of the Interrupt Service Routine (ISR).

FIGURE 5-2: INTERRUPT STACK FRAME



Note 1: The user can always lower the priority level by writing a new value into SR. The Interrupt Service Routine must clear the interrupt flag bits in the IFSx register before lowering the processor interrupt priority in order to avoid recursive interrupts.

2: The IPL3 bit (CORCON<3>) is always clear when interrupts are being processed. It is set only during execution of traps.

The RETFIE (return from interrupt) instruction will unstack the program counter and STATUS registers to return the processor to its state prior to the interrupt sequence.

5.5 Alternate Interrupt Vector Table

In program memory, the Interrupt Vector Table (IVT) is followed by the Alternate Interrupt Vector Table (AIVT), as shown in Figure 5-1. Access to the Alternate Interrupt Vector Table is provided by the ALTIVT bit in the INTCON2 register. If the ALTIVT bit is set, all interrupt and exception processes will use the alternate vectors instead of the default vectors. The alternate vectors are organized in the same manner as the default vectors. The AIVT supports emulation and debugging efforts by providing a means to switch between an application and a support environment, without requiring the interrupt vectors to be reprogrammed. This feature also enables switching between applications for evaluation of different software algorithms at run time.

If the AIVT is not required, the program memory allocated to the AIVT may be used for other purposes. AIVT is not a protected section and may be freely programmed by the user.

5.6 Fast Context Saving

A context saving option is available using shadow registers. Shadow registers are provided for the DC, N, OV, Z and C bits in SR, and the registers W0 through W3. The shadows are only one-level deep. The shadow registers are accessible using the PUSH.S and POP.S instructions only.

When the processor vectors to an interrupt, the PUSH.S instruction can be used to store the current value of the aforementioned registers into their respective shadow registers.

If an ISR of a certain priority uses the PUSH.S and POP.S instructions for fast context saving, then a higher priority ISR should not include the same instructions. Users must save the key registers in software during a lower priority interrupt if the higher priority ISR uses fast context saving.

5.7 External Interrupt Requests

The interrupt controller supports three external interrupt request signals, INT0-INT2. These inputs are edge sensitive; they require a low-to-high, or a high-to-low transition, to generate an interrupt request. The INTCON2 register has three bits, INT0EP-INT2EP, that select the polarity of the edge detection circuitry.

5.8 Wake-up from Sleep and Idle

The interrupt controller may be used to wake-up the processor from either Sleep or Idle modes if Sleep or Idle modes are active when the interrupt is generated.

If an enabled interrupt request of sufficient priority is received by the interrupt controller, then the standard interrupt request is presented to the processor. At the same time, the processor will wake-up from Sleep or Idle and begin execution of the ISR needed to process the interrupt request.

dsPIC30F4011/4012

7.3.2 WRITING A BLOCK OF DATA EEPROM

To write a block of data EEPROM, write to all sixteen latches first, then set the NVMCON register and program the block.

EXAMPLE 7-5: DATA EEPROM BLOCK WRITE

```
MOV      #LOW_ADDR_WORD,W0 ; Init pointer
MOV      #HIGH_ADDR_WORD,W1
MOV      W1,TBLPAG
MOV      #data1,W2          ; Get 1st data
TBLWTL   W2,[W0]++          ; write data
MOV      #data2,W2          ; Get 2nd data
TBLWTL   W2,[W0]++          ; write data
MOV      #data3,W2          ; Get 3rd data
TBLWTL   W2,[W0]++          ; write data
MOV      #data4,W2          ; Get 4th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data5,W2          ; Get 5th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data6,W2          ; Get 6th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data7,W2          ; Get 7th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data8,W2          ; Get 8th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data9,W2          ; Get 9th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data10,W2         ; Get 10th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data11,W2         ; Get 11th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data12,W2         ; Get 12th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data13,W2         ; Get 13th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data14,W2         ; Get 14th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data15,W2         ; Get 15th data
TBLWTL   W2,[W0]++          ; write data
MOV      #data16,W2         ; Get 16th data
TBLWTL   W2,[W0]++          ; write data. The NVMADR captures last table access address.
MOV      #0x400A,W0         ; Select data EEPROM for multi word op
MOV      W0,NVMCON           ; Operate Key to allow program operation
DISI     #5                  ; Block all interrupts with priority < 7
                                ; for next 5 instructions

MOV      #0x55,W0
MOV      W0,NVMKEY           ; Write the 0x55 key
MOV      #0xAA,W1
MOV      W1,NVMKEY           ; Write the 0xAA key
BSET     NVMCON,#WR          ; Start write cycle
NOP
NOP
```

TABLE 9-1: TIMER1 REGISTER MAP⁽¹⁾

SFR Name	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	Reset State
TMR1	0100	Timer1 Register																uuuu uuuu uuuu uuuu
PR1	0102	Period Register 1																1111 1111 1111 1111
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000 0000 0000 0000

Legend: u = uninitialized bit; — = unimplemented bit, read as '0'
Note 1: Refer to the “dsPIC30F Family Reference Manual” (DS70046) for descriptions of register bit fields.

15.7.1 DEAD-TIME GENERATORS

Each complementary output pair for the PWM module has a 6-bit down counter that is used to produce the dead-time insertion. As shown in Figure 15-4, each dead-time unit has a rising and falling edge detector connected to the duty cycle comparison output.

15.7.2 DEAD-TIME RANGES

The amount of dead time provided by the dead-time unit is selected by specifying the input clock prescaler value and a 6-bit unsigned value.

Four input clock prescaler selections have been provided to allow a suitable range of dead time based on the device operating frequency. The dead-time clock prescaler values are selected using the

DTAPS<1:0> control bits in the DTCON1 SFR. One of four clock prescaler options (T_{CY} , $2 T_{CY}$, $4 T_{CY}$ or $8 T_{CY}$) may be selected.

After the prescaler value is selected, the dead time is adjusted by loading 6-bit unsigned values into the DTCON1 SFR.

The dead-time unit prescaler is cleared on the following events:

- On a load of the down timer due to a duty cycle comparison edge event.
- On a write to the DTCON1 register.
- On any device Reset.

Note: The user should not modify the DTCON1 register value while the PWM module is operating ($PTEN = 1$). Unexpected results may occur.

FIGURE 15-4: DEAD-TIME TIMING DIAGRAM

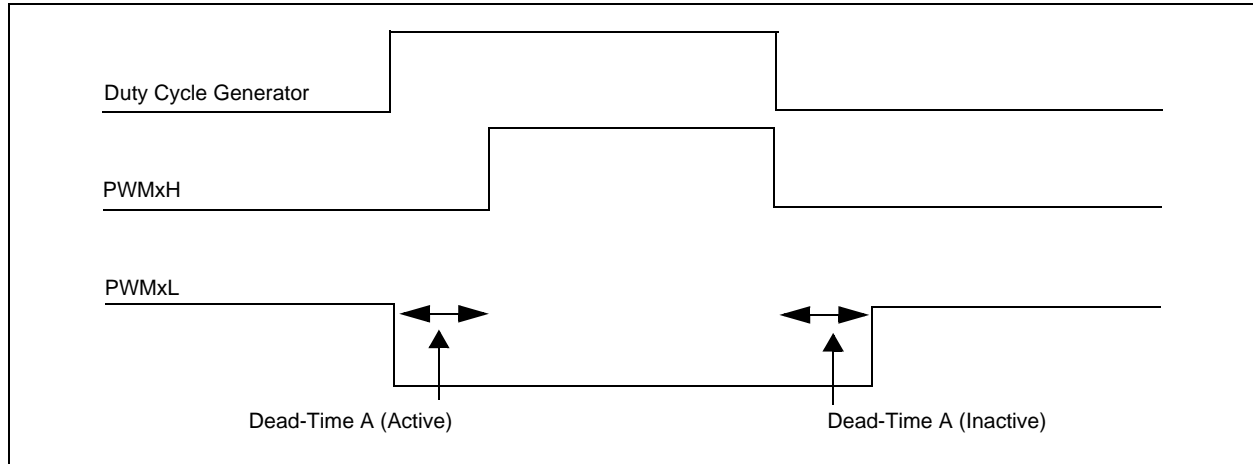


TABLE 15-1: 6-OUTPUT PWM REGISTER MAP⁽¹⁾

SFR Name	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	Reset State
PTCON	01C0	PTEN	—	PTSIDL	—	—	—	—	—	PTOPS<3:0>				PTCKPS<1:0>		PTMOD<1:0>		0000 0000 0000 0000
PTMR	01C2	PTDIR	PWM Timer Count Value															0000 0000 0000 0000
PTPER	01C4	—	PWM Time Base Period Register															0111 1111 1111 1111
SEVTCMP	01C6	SEVTDIR	PWM Special Event Compare Register															0000 0000 0000 0000
PWMCON1	01C8	—	—	—	—	—	PTMOD3	PTMOD2	PTMOD1	—	PEN3H	PEN2H	PEN1H	—	PEN3L	PEN2L	PEN1L	0000 0000 1111 1111
PWMCON2	01CA	—	—	—	—	SEVOPS<3:0>				—	—	—	—	—	IUE	OSYNC	UDIS	0000 0000 0000 0000
DTCON1	01CC	—	—	—	—	—	—	—	—	DTAPS<1:0>		Dead-Time A Value						0000 0000 0000 0000
FLTACON	01D0	—	—	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L	FLTAM	—	—	—	—	FAEN3	FAEN2	FAEN1	0000 0000 0000 0000
OVDCON	01D4	—	—	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	—	—	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	1111 1111 0000 0000
PDC1	01D6	PWM Duty Cycle 1 Register																0000 0000 0000 0000
PDC2	01D8	PWM Duty Cycle 2 Register																0000 0000 0000 0000
PDC3	01DA	PWM Duty Cycle 3 Register																0000 0000 0000 0000

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

Note 1: Refer to the “dsPIC30F Family Reference Manual” (DS70046) for descriptions of register bit fields.

TABLE 16-1: SPI1 REGISTER MAP⁽¹⁾

SFR Name	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	Reset State
SPI1STAT	0220	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI1CON	0222	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI1BUF	0224	Transmit and Receive Buffer																0000 0000 0000 0000

Legend: u = uninitialized bit; — = unimplemented bit, read as '0'
Note 1: Refer to the “dsPIC30F Family Reference Manual” (DS70046) for descriptions of register bit fields.

17.0 I²C™ MODULE

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the *dsPIC30F Family Reference Manual* (DS70046). For more information on the device instruction set and programming, refer to the *“16-bit MCU and DSC Reference Manual”* (DS70157).

The Inter-Integrated Circuit (I²C™) module provides complete hardware support for both Slave and Multi-Master modes of the I²C serial communication standard with a 16-bit interface.

This module offers the following key features:

- I²C Interface supporting both Master and Slave Operation
- I²C Slave mode supports 7-bit and 10-bit addressing
- I²C Master mode supports 7-bit and 10-bit addressing
- I²C Port allows Bidirectional Transfers between Master and Slaves
- Serial Clock Synchronization for I²C Port can be used as a Handshake Mechanism to Suspend and Resume Serial Transfer (SCLREL control)
- I²C supports Multi-Master Operation; detects bus collision and will arbitrate accordingly

17.1 Operating Function Description

The hardware fully implements all the master and slave functions of the I²C Standard and Fast mode specifications, as well as 7 and 10-bit addressing.

Thus, the I²C module can operate either as a slave or a master on an I²C bus.

17.1.1 VARIOUS I²C MODES

The following types of I²C operation are supported:

- I²C slave operation with 7-bit addressing
- I²C slave operation with 10-bit addressing
- I²C master operation with 7-bit or 10-bit addressing

See the I²C programmer's model in Figure 17-1.

17.1.2 PIN CONFIGURATION IN I²C MODE

I²C has a 2-pin interface. The SCL pin is clock and the SDA pin is data.

17.1.3 I²C REGISTERS

I2CCON and I2CSTAT are control and status registers, respectively. The I2CCON register is readable and writable. The lower 6 bits of I2CSTAT are read-only. The remaining bits of the I2CSTAT are read/write.

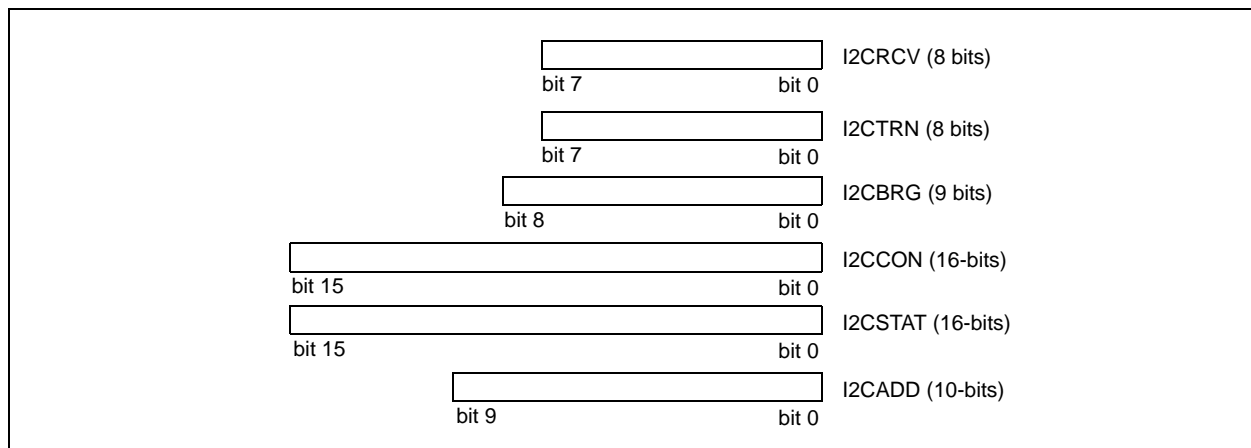
I2CRSR is the shift register used for shifting data, whereas I2CRCV is the buffer register to which data bytes are written or from which data bytes are read. I2CRCV is the receive buffer, as shown in Figure 17-1. I2CTRN is the transmit register to which bytes are written during a transmit operation, as shown in Figure 17-2.

The I2CADD register holds the slave address. A status bit, ADD10, indicates 10-bit Address mode. The I2CBRG acts as the Baud Rate Generator reload value.

In receive operations, I2CRSR and I2CRCV together form a double-buffered receiver. When I2CRSR receives a complete byte, it is transferred to I2CRCV and an interrupt pulse is generated. During transmission, the I2CTRN is not double-buffered.

Note: Following a Restart condition in 10-bit mode, the user only needs to match the first 7-bit address.

FIGURE 17-1: PROGRAMMER'S MODEL



17.12.2 I²C MASTER RECEPTION

Master mode reception is enabled by programming the receive enable (RCEN) bit (I2CCON<3>). The I²C module must be Idle before the RCEN bit is set, otherwise the RCEN bit will be disregarded. The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin toggles and data is shifted in to the I2CRSR on the rising edge of each clock.

17.12.3 BAUD RATE GENERATOR (BRG)

In I²C Master mode, the reload value for the BRG is located in the I2CBRG register. When the BRG is loaded with this value, the BRG counts down to '0' and stops until another reload has taken place. If clock arbitration is taking place, for instance, the BRG is reloaded when the SCL pin is sampled high.

As per the I²C standard, F_{SCK} may be 100 kHz or 400 kHz. However, the user can specify any baud rate up to 1 MHz. I2CBRG values of '0' or '1' are illegal.

EQUATION 17-1: I2CBRG VALUE

$$I2CBRG = \left(\frac{FCY}{F_{SCL}} - \frac{FCY}{1,111,111} \right) - 1$$

17.12.4 CLOCK ARBITRATION

Clock arbitration occurs when the master deasserts the SCL pin (SCL allowed to float high) during any receive, transmit or Restart/Stop condition. When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of I2CBRG and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device.

17.12.5 MULTI-MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-master operation support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high, while another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the MI2CIF pulse and reset the master portion of the I²C port to its Idle state.

If a transmit was in progress when the bus collision occurred, the transmission is halted, the TBF flag is cleared, the SDA and SCL lines are deasserted and a value can now be written to I2CTRN. When the user services the I²C master event Interrupt Service Routine and if the I²C bus is free (i.e., the P bit is set), the user can resume communication by asserting a Start condition.

If a Start, Restart, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the I2CCON register are cleared to '0'. When the user services the bus collision Interrupt Service Routine, and if the I²C bus is free, the user can resume communication by asserting a Start condition.

The Master will continue to monitor the SDA and SCL pins, and if a Stop condition occurs, the MI2CIF bit will be set.

A write to the I2CTRN will start the transmission of data at the first data bit, regardless of where the transmitter left off when bus collision occurred.

In a Multi-Master environment, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I²C bus can be taken when the P bit is set in the I2CSTAT register, or the bus is Idle and the S and P bits are cleared.

17.13 I²C Module Operation During CPU Sleep and Idle Modes

17.13.1 I²C OPERATION DURING CPU SLEEP MODE

When the device enters Sleep mode, all clock sources to the module are shut down and stay at logic '0'. If Sleep occurs in the middle of a transmission, and the state machine is partially into a transmission as the clocks stop, then the transmission is aborted. Similarly, if Sleep occurs in the middle of a reception, then the reception is aborted.

17.13.2 I²C OPERATION DURING CPU IDLE MODE

For the I²C, the I2CSIDL bit selects if the module will stop on Idle or continue on Idle. If I2CSIDL = 0, the module will continue operation on assertion of the Idle mode. If I2CSIDL = 1, the module will stop on Idle.

18.5.2 FRAMING ERROR (FERR)

The FERR bit (UxSTA<2>) is set if a '0' is detected instead of a Stop bit. If two Stop bits are selected, both Stop bits must be '1', otherwise FERR will be set. The read-only FERR bit is buffered along with the received data; it is cleared on any Reset.

18.5.3 PARITY ERROR (PERR)

The PERR bit (UxSTA<3>) is set if the parity of the received word is incorrect. This error bit is applicable only if a Parity mode (odd or even) is selected. The read-only PERR bit is buffered along with the received data bytes; it is cleared on any Reset.

18.5.4 IDLE STATUS

When the receiver is active (i.e., between the initial detection of the Start bit and the completion of the Stop bit), the RIDLE bit (UxSTA<4>) is '0'. Between the completion of the Stop bit and detection of the next Start bit, the RIDLE bit is '1', indicating that the UART is Idle.

18.5.5 RECEIVE BREAK

The receiver will count and expect a certain number of bit times based on the values programmed in the PDSEL<1:0> (UxMODE<2:1>) and STSEL (UxMODE<0>) bits.

If the Break is longer than 13 bit times, the reception is considered complete after the number of bit times specified by PDSEL and STSEL. The URXDA bit is set, FERR is set, zeros are loaded into the receive FIFO, interrupts are generated, if appropriate and the RIDLE bit is set.

When the module receives a long Break signal and the receiver has detected the Start bit, the data bits and the invalid Stop bit (which sets the FERR), the receiver must wait for a valid Stop bit before looking for the next Start bit. It cannot assume that the Break condition on the line is the next Start bit.

Break is regarded as a character containing all '0's, with the FERR bit set. The Break character is loaded into the buffer. No further reception can occur until a Stop bit is received. Note that RIDLE goes high when the Stop bit has not been received yet.

18.6 Address Detect Mode

Setting the ADDEN bit (UxSTA<5>) enables this special mode in which a 9th bit (URX8) value of '1' identifies the received word as an address, rather than data. This mode is only applicable for 9-bit data communication. The URXISELx control bit does not have any impact on interrupt generation in this mode, since an interrupt (if enabled) will be generated every time the received word has the 9th bit set.

18.7 Loopback Mode

Setting the LPBACK bit enables this special mode in which the UxTX pin is internally connected to the UxRX pin. When configured for the Loopback mode, the UxRX pin is disconnected from the internal UART receive logic. However, the UxTX pin still functions as in a normal operation.

To select this mode:

- Configure UART for desired mode of operation.
- Set LPBACK = 1 to enable Loopback mode.
- Enable transmission as defined in **Section 18.3 "Transmitting Data"**.

18.8 Baud Rate Generator

The UART has a 16-bit Baud Rate Generator to allow maximum flexibility in baud rate generation. The Baud Rate Generator register (UxBRG) is readable and writable. The baud rate is computed as follows:

BRG = 16-bit value held in UxBRG register
(0 through 65535)

FCY = Instruction Clock Rate (1/Tcy)

The baud rate is given by Equation 18-1.

EQUATION 18-1: BAUD RATE

$$\text{Baud Rate} = \text{FCY} / (16 * (\text{BRG} + 1))$$

Therefore, maximum baud rate possible is:

$\text{FCY} / 16$ (if BRG = 0),

and the minimum baud rate possible is:

$\text{FCY} / (16 * 65536)$.

With a full, 16-bit Baud Rate Generator, at 30 MIPS operation, the minimum baud rate achievable is 28.5 bps.

20.7 A/D Conversion Speeds

The dsPIC30F 10-bit ADC specifications permit a maximum 1 Msps sampling rate. Table 20-1 summarizes the conversion speeds for the dsPIC30F 10-bit ADC and the required operating conditions.

TABLE 20-1: 10-BIT A/D CONVERSION RATE PARAMETERS

dsPIC30F 10-bit A/D Converter Conversion Rates						
A/D Speed	TAD Minimum	Sampling Time Min.	Rs Max.	VDD	Temperature	A/D Channels Configuration
Up to 1 Msps ⁽¹⁾	83.33 ns	12 TAD	500Ω	4.5V to 5.5V	-40°C to +85°C	
Up to 750 ksp/s ⁽¹⁾	95.24 ns	2 TAD	500Ω	4.5V to 5.5V	-40°C to +85°C	
Up to 600 ksp/s ⁽¹⁾	138.89 ns	12 TAD	500Ω	3.0V to 5.5V	-40°C to +125°C	
Up to 500 ksp/s	153.85 ns	1 TAD	5.0 kΩ	4.5V to 5.5V	-40°C to +125°C	
Up to 300 ksp/s	256.41 ns	1 TAD	5.0 kΩ	3.0V to 5.5V	-40°C to +125°C	

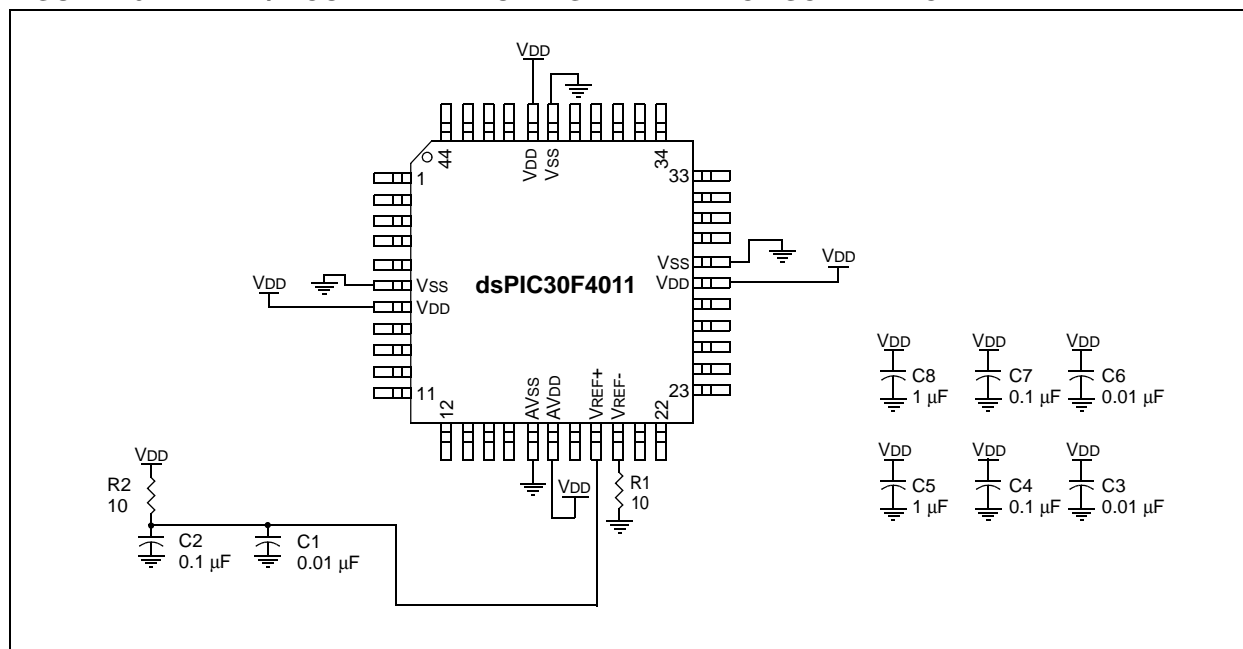
Note 1: External VREF- and VREF+ pins must be used for correct operation. See Figure 20-2 for recommended circuit.

dsPIC30F4011/4012

The configuration guidelines give the required setup values for the conversion speeds above 500 ksps, since they require external VREF pins usage and there are some differences in the configuration procedure. Configuration details that are not critical to the conversion speed have been omitted.

Figure 20-2 illustrates the recommended circuit for the conversion rates above 500 ksps.

FIGURE 20-2: A/D CONVERTER VOLTAGE REFERENCE SCHEMATIC



20.7.1 1 Msps CONFIGURATION GUIDELINE

The configuration for 1 Msps operation is dependent on whether a single input pin is to be sampled or whether multiple pins are to be sampled.

20.7.1.1 Single Analog Input

For conversions at 1 Msps for a single analog input, at least two sample and hold channels must be enabled. The analog input multiplexer must be configured so that the same input pin is connected to both sample and hold channels. The A/D converts the value held on one S&H channel while the second S&H channel acquires a new input sample.

20.7.1.2 Multiple Analog Inputs

The ADC can also be used to sample multiple analog inputs using multiple sample and hold channels. In this case, the total 1 Msps conversion rate is divided among the different input signals. For example, four inputs can be sampled at a rate of 250 ksps for each signal, or two inputs could be sampled at a rate of 500 ksps for each signal. Sequential sampling must be used in this configuration to allow adequate sampling time on each input.

21.7 In-Circuit Debugger

When MPLAB® ICD 2 is selected as a debugger, the in-circuit debugging functionality is enabled. This function allows simple debugging functions when used with MPLAB IDE. When the device has this feature enabled, some of the resources are not available for general use. These resources include the first 80 bytes of data RAM and two I/O pins.

one of four pairs of debug I/O pins may be selected by the user using configuration options in MPLAB IDE. These pin pairs are named EMUD/EMUC, EMUD1/EMUC1, EMUD2/EMUC2 and EMUD3/EMUC3.

In each case, the selected EMUD pin is the emulation/debug data line and the EMUC pin is the emulation/debug clock line. These pins interface to the MPLAB ICD 2 module available from Microchip. The selected pair of debug I/O pins is used by MPLAB ICD 2 to send commands and receive responses, as well as to send and receive data. To use the in-circuit debugger function of the device, the design must implement ICSP connections to MCLR, VDD, VSS, PGC, PGD and the selected EMUDx/EMUCx pin pair.

This gives rise to two possibilities:

1. If EMUD/EMUC is selected as the debug I/O pin pair, then only a 5-pin interface is required as the EMUD and EMUC pin functions are multiplexed with the PGD and PGC pin functions in all dsPIC30F devices.
2. If EMUD1/EMUC1, EMUD2/EMUC2 or EMUD3/EMUC3 is selected as the debug I/O pin pair, then a 7-pin interface is required as the EMUDx/EMUCx pin functions (x = 1, 2 or 3) are not multiplexed with the PGD and PGC pin functions.

23.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

23.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

23.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

23.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/librarian features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

23.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

dsPIC30F4011/4012

FIGURE 24-8: QEI MODULE EXTERNAL CLOCK TIMING CHARACTERISTICS

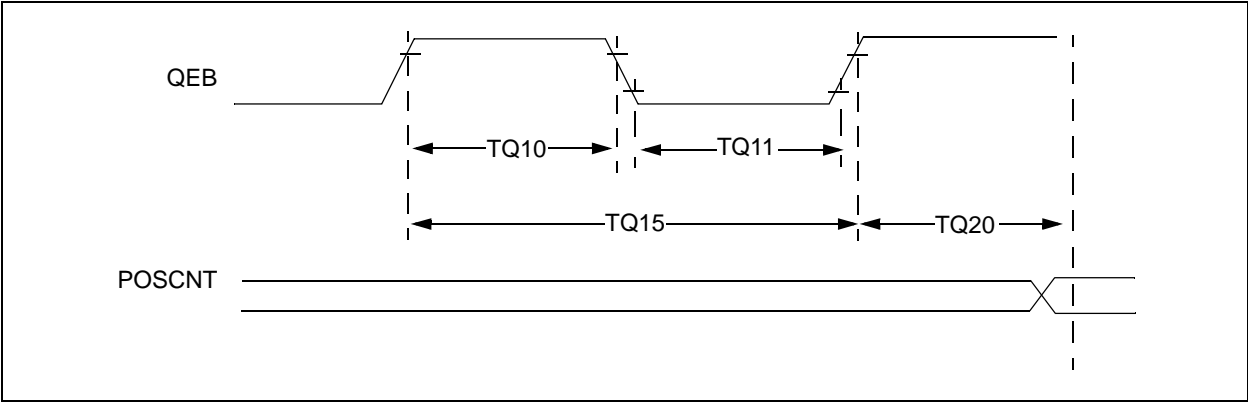


TABLE 24-25: QEI MODULE EXTERNAL CLOCK TIMING REQUIREMENTS

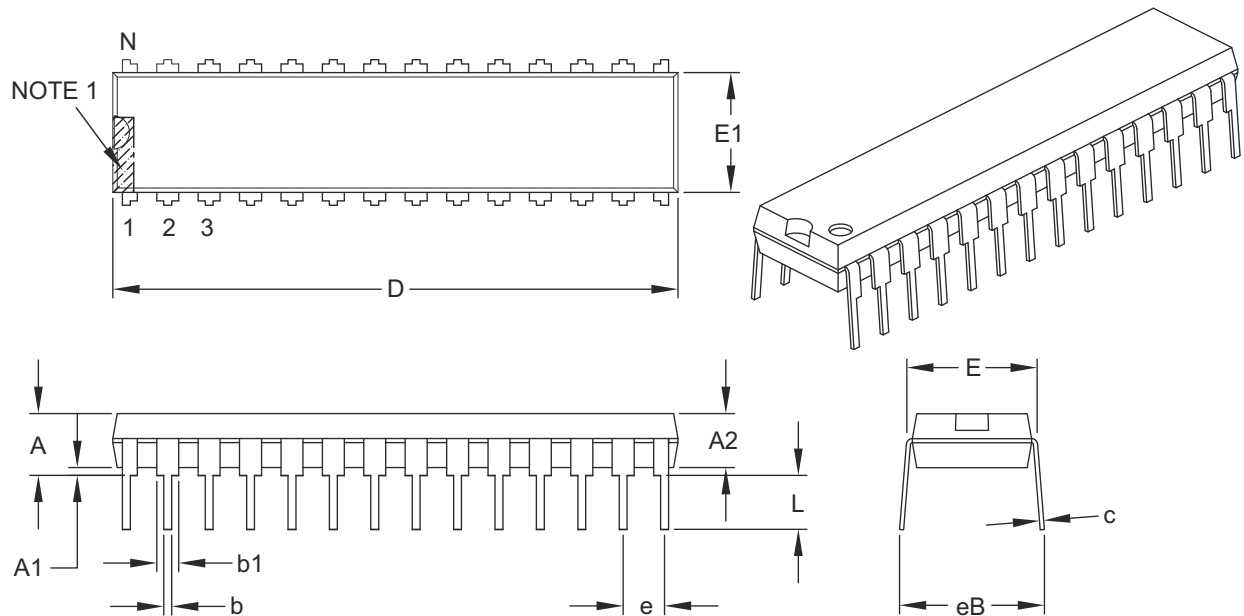
AC CHARACTERISTICS				Standard Operating Conditions: 2.5V to 5.5V (unless otherwise stated) Operating temperature -40°C ≤TA ≤+85°C for Industrial -40°C ≤TA ≤+125°C for Extended				
Param No.	Symbol	Characteristic ⁽¹⁾		Min	Typ	Max	Units	Conditions
TQ10	TtQH	TxCK High Time	Synchronous, with prescaler	TcY + 20	—	—	ns	Must also meet parameter TQ15
TQ11	TtQL	TxCK Low Time	Synchronous, with prescaler	TcY + 20	—	—	ns	Must also meet parameter TQ15
TQ15	TtQP	TxCK Input Period	Synchronous, with prescaler	2 * TcY + 40	—	—	ns	
TQ20	TCKEXTMRL	Delay from External TxCK Clock Edge to Timer Increment		0.5 TcY	—	1.5 TcY	ns	

Note 1: These parameters are characterized but not tested in manufacturing.

25.2 Package Details

28-Lead Skinny Plastic Dual In-Line (SP) – 300 mil Body [SPDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	–	–	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

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://support.microchip.com>
Web Address:
www.microchip.com

Atlanta

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

Boston

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

Chicago

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

Cleveland

Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas

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

Detroit

Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo

Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles

Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara

Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto

Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing

Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hong Kong SAR

Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen

Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen

Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai

Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama

Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu

Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul

Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang

Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila

Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore

Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu

Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung

Tel: 886-7-213-7830
Fax: 886-7-330-9305

Taiwan - Taipei

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok

Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels

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

Denmark - Copenhagen

Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris

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

Germany - Munich

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

Italy - Milan

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

Netherlands - Drunen

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

Spain - Madrid

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

UK - Wokingham

Tel: 44-118-921-5869
Fax: 44-118-921-5820

08/04/10