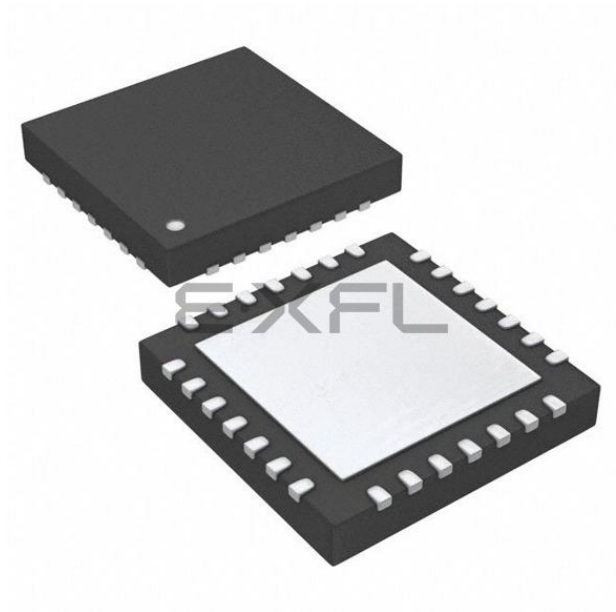


Welcome to E-XFL.COM

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	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	25
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 24x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15356-i-ml

4.1.1 READING PROGRAM MEMORY AS DATA

There are three methods of accessing constants in program memory. The first method is to use tables of RETLW instructions. The second method is to set an FSR to point to the program memory. The third method is to use the NVMREG interface to access the program memory. For an example of NVMREG interface use, reference Section 13.3, NVMREG Access.

4.1.1.1 RETLW Instruction

The RETLW instruction can be used to provide access to tables of constants. The recommended way to create such a table is shown in Example 4-1.

EXAMPLE 4-1: RETLW INSTRUCTION

```
constants
    BRW                 ;Add Index in W to
                        ;program counter to
                        ;select data
    RETLW DATA0         ;Index0 data
    RETLW DATA1         ;Index1 data
    RETLW DATA2
    RETLW DATA3


my_function
    ;… LOTS OF CODE…
    MOVLW       DATA_INDEX
    call constants
    ;… THE CONSTANT IS IN W
```

The BRW instruction makes this type of table very simple to implement.

4.1.1.2 Indirect Read with FSR

The program memory can be accessed as data by setting bit 7 of an FSRxH register and reading the matching INDFx register. The MOVIW instruction will place the lower eight bits of the addressed word in the W register. Writes to the program memory cannot be performed via the INDF registers. Instructions that read the program memory via the FSR require one extra instruction cycle to complete. Example 4-2 demonstrates reading the program memory via an FSR.

TABLE 4-11: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 6											
CPU CORE REGISTERS; see Table 4-3 for specifics											
30Ch	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	uuuu uuuu
30Dh	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	uuuu uuuu
30Eh	CCP1CON	EN	—	OUT	FMT	MODE<3:0>				0-00 0000	0-00 0000
30Fh	CCP1CAP	—	—	—	—	—	CTS<2:0>			---- -000	---- -000
310h	CCPR2L	Capture/Compare/PWM Register 2 (LSB)								xxxx xxxx	uuuu uuuu
311h	CCPR2H	Capture/Compare/PWM Register 2 (MSB)								xxxx xxxx	uuuu uuuu
312h	CCP2CON	EN	—	OUT	FMT	MODE<3:0>				0-00 0000	0-00 0000
313h	CCP2CAP	—	—	—	—	—	CTS<2:0>			---- -000	---- -000
314h	PWM3DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
315h	PWM3DCH	DC<9:0>								xxxx xxxx	uuuu uuuu
316h	PWM3CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
317h	—	Unimplemented								—	—
318h	PWM4DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
319h	PWM4DCH	DC<9:0>								xxxx xxxx	uuuu uuuu
31Ah	PWM4CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
31Bh	—	Unimplemented								—	—
31Ch	PWM5DCL	DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
31Dh	PWM5DCH	DC<9:0>								xxxx xxxx	uuuu uuuu
31Eh	PWM5CON	EN	—	OUT	POL	—	—	—	—	0-00 ----	0-00 ----
31Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

8.12 Memory Execution Violation

A Memory Execution Violation Reset occurs if executing an instruction being fetched from outside the valid execution area. The different valid execution areas are defined as follows:

- Flash Memory: Table 4-1 shows the addresses available on the PIC16(L)F15356/75/76/85/86 devices based on user Flash size. Execution outside this region generates a memory execution violation.
- Storage Area Flash (SAF): If Storage Area Flash (SAF) is enabled (**Section 4.2.3 “Storage Area Flash”**), the SAF area (Table 4-2) is not a valid execution area.

Prefetched instructions that are not executed do not cause memory execution violations. For example, a `GOTO` instruction in the last memory location will prefetch from an invalid location; this is not an error. If an instruction from an invalid location tries to execute, the memory violation is generated immediately, and any concurrent interrupt requests are ignored. When a memory execution violation is generated, the device is reset and flag $\overline{\text{MEMV}}$ is cleared in PCON1 (Register 8-3) to signal the cause. The flag needs to be set in code after a memory execution violation.

10.1 Operation

Interrupts are disabled upon any device Reset. They are enabled by setting the following bits:

- GIE bit of the INTCON register
- Interrupt Enable bit(s) of the PIEx[y] registers for the specific interrupt event(s)
- PEIE bit of the INTCON register (if the Interrupt Enable bit of the interrupt event is contained in the PIEx registers)

The PIR1, PIR2, PIR3, PIR4, PIR5, PIR6, and PIR7 registers record individual interrupts via interrupt flag bits. Interrupt flag bits will be set, regardless of the status of the GIE, PEIE and individual interrupt enable bits.

The following events happen when an interrupt event occurs while the GIE bit is set:

- Current prefetched instruction is flushed
- GIE bit is cleared
- Current Program Counter (PC) is pushed onto the stack
- Critical registers are automatically saved to the shadow registers (See **"Section 10.5 "Automatic Context Saving"**)
- PC is loaded with the interrupt vector 0004h

The firmware within the Interrupt Service Routine (ISR) should determine the source of the interrupt by polling the interrupt flag bits. The interrupt flag bits must be cleared before exiting the ISR to avoid repeated interrupts. Because the GIE bit is cleared, any interrupt that occurs while executing the ISR will be recorded through its interrupt flag, but will not cause the processor to redirect to the interrupt vector.

The `RETFIE` instruction exits the ISR by popping the previous address from the stack, restoring the saved context from the shadow registers and setting the GIE bit.

For additional information on a specific interrupts operation, refer to its peripheral chapter.

Note 1:	Individual interrupt flag bits are set, regardless of the state of any other enable bits.
2:	All interrupts will be ignored while the GIE bit is cleared. Any interrupt occurring while the GIE bit is clear will be serviced when the GIE bit is set again.

10.2 Interrupt Latency

Interrupt latency is defined as the time from when the interrupt event occurs to the time code execution at the interrupt vector begins. The interrupt is sampled during Q1 of the instruction cycle. The actual interrupt latency then depends on the instruction that is executing at the time the interrupt is detected. See Figure 10-2 and Figure 10-3 for more details.

TABLE 11-1: SUMMARY OF REGISTERS ASSOCIATED WITH POWER-DOWN MODE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	146
PIE0	—	—	TMR0IE	IOCIE	—	—	—	INTE	147
PIE1	OSFIE	CSWIE	—	—	—	—	—	ADIE	148
PIE2	—	ZCDIE	—	—	—	—	C2IE	C1IE	149
PIE3	RC2IE	TX2IE	RC1IE	TX1IE	BCL2IE	SSP2IE	BCL1IE	SSP1IE	150
PIE4	—	—	—	—	—	—	TMR2IE	TMR1IE	151
PIR0	—	—	TMR0IF	IOCIF	—	—	—	INTF	155
PIR1	OSFIF	CSWIF	—	—	—	—	—	ADIF	156
PIR2	—	ZCDIF	—	—	—	—	C2IF	C1IF	157
PIR3	RC2IF	TX2IF	RC1IF	TX1IF	BCL2IF	SSP2IF	BCL1IF	SSP1IF	158
PIR4	—	—	—	—	—	—	TMR2IF	TMR1IF	159
IOCAP	IOCAP7	IOCAP6	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	255
IOCAN	IOCAN7	IOCAN6	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	255
IOCAF	IOCAF7	IOCAF6	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	256
IOCBP	IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0	257
IOCBN	IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0	257
IOCBF	IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0	258
IOCCP	IOCCP7	IOCCP6	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	259
IOCCN	IOCCN7	IOCCN6	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	259
IOCCF	IOCCF7	IOCCF6	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	259
IOCEP	—	—	—	—	IOCEP3	IOCEP2(1)	IOCEP1(1)	IOCEP0(1)	260
IOCEN	—	—	—	—	IOCEN3	IOCEN2(1)	IOCEN1(1)	IOCEN0(1)	260
IOCEF	—	—	—	—	IOCEF3	IOCEF2(1)	IOCEF1(1)	IOCEF0(1)	261
STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	54
VREGCON	—	—	—	—	—	—	VREGPM	—	168
CPUDOZE	IDLEN	DOZEN	ROI	DOE	—	DOZE<2:0>			169
WDTCON0	—	—	WDTPS<4:0>					SWDTEN	175

Legend: — = unimplemented location, read as '0'. Shaded cells are not used in Power-Down mode.

Note 1: Present only in PIC16(L)F15375/76/85/86.

EXAMPLE 13-5: DEVICE ID ACCESS

```
; This write routine assumes the following:
; 1.  A full row of data are loaded, starting at the address in DATA_ADDR
; 2.  Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
; stored in little endian format
; 3.  A valid starting address (the least significant bits = 00000) is loaded in ADDRH:ADDRL
; 4.  ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)
; 5.  NVM interrupts are not taken into account

          BANKSEL             NVMADRH
          MOVF                ADDRH,W
          MOVWF               NVMADRH                 ; Load initial address
          MOVF                ADDRL,W
          MOVWF               NVMADRL
          MOVLW               LOW DATA_ADDR           ; Load initial data address
          MOVWF               FSR0L
          MOVLW               HIGH DATA_ADDR
          MOVWF               FSR0H
          BCF                 NVMCON1,NVMREGS         ; Set PFM as write location
          BSF                 NVMCON1,WREN            ; Enable writes
          BSF                 NVMCON1,LWLO            ; Load only write latches

LOOP
          MOVIW               FSR0++
          MOVWF               NVMDATL                 ; Load first data byte
          MOVIW               FSR0++
          MOVWF               NVMDATH                 ; Load second data byte

          CALL                UNLOCK_SEQ              ; If not, go load latch
          INCF                NVMADRL,F               ; Increment address
          MOVF                NVMADRL,W
          XORLW               0x1F                    ; Check if lower bits of address are 00000
          ANDLW               0x1F                    ; and if on last of 32 addresses
          BTFSC               STATUS,Z                ; Last of 32 words?
          GOTO                START_WRITE             ; If so, go write latches into memory

          GOTO                LOOP

START_WRITE
          BCF                 NVMCON1,LWLO            ; Latch writes complete, now write memory
          CALL                UNLOCK_SEQ              ; Perform required unlock sequence
          BCF                 NVMCON1,LWLO            ; Disable writes

UNLOCK_SEQ
          MOVLW               55h
          BCF                 INTCON,GIE              ; Disable interrupts
          MOVWF               NVMCON2                 ; Begin unlock sequence
          MOVLW               AAh
          MOVWF               NVMCON2
          BSF                 NVMCON1,WR
          BSF                 INTCON,GIE              ; Unlock sequence complete, re-enable interrupts
          return
```

REGISTER 13-6: NVMCON2: NONVOLATILE MEMORY CONTROL 2 REGISTER

W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0
NVMCON2<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
S = Bit can only be set	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-0 **NVMCON2<7:0>:** Flash Memory Unlock Pattern bits

To unlock writes, a 55h must be written first followed by an AAh before setting the WR bit of the NVMCON1 register. The value written to this register is used to unlock the writes.

TABLE 13-5: SUMMARY OF REGISTERS ASSOCIATED WITH NONVOLATILE MEMORY (NVM)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	146
PIE7	—	—	NVMIE	NCO1IE	—	—	—	CWG1IE	154
PIR7	—	—	NVMIF	NCO1IF	—	—	—	CWG1IF	162
NVMCON1	—	NVMREGS	LWLO	FREE	WRERR	WREN	WR	RD	195
NVMCON2	NVMCON2<7:0>								196
NVMADRL	NVMADR<7:0>								194
NVMADRH	—(1)	NVMADR<14:8>							194
NVMDATL	NVMDAT<7:0>								194
NVMDATH	—	—	NVMDAT<13:8>						194

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by NVM.

Note 1: Unimplemented, read as '1'.

TABLE 15-5: PPS OUTPUT SIGNAL ROUTING OPTIONS (PIC16(L)F15356)

Output Signal Name	RxyPPS Register Value	Remappable to Pins of PORTx		
		PIC16(L)F15356		
		PORTA	PORTB	PORTC
CLKR	0x1B		•	•
NCO1OUT	0x1A	•		•
TMR0	0x19		•	•
SDO2/SDA2	0x18		•	•
SCK2/SCL2	0x17		•	•
SDO1/SDA1	0x16		•	•
SCK1/SCL1	0x15		•	•
C2OUT	0x14	•		•
C1OUT	0x13	•		•
DT2	0x12		•	•
TX2/CK2	0x11		•	•
DT1	0x10		•	•
TX1/CK1	0x0F		•	•
PWM6OUT	0x0E	•		•
PWM5OUT	0x0D	•		•
PWM4OUT	0x0C		•	•
PWM3OUT	0x0B		•	•
CCP2	0x0A		•	•
CCP1	0x09		•	•
CWG1D	0x08		•	•
CWG1C	0x07		•	•
CWG1B	0x06		•	•
CWG1A	0x05		•	•
CLC4OUT	0x04		•	•
CLC3OUT	0x03		•	•
CLC2OUT	0x02	•		•
CLC1OUT	0x01	•		•

26.6.2 TIMER GATE SOURCE SELECTION

One of the several different external or internal signal sources may be chosen to gate the timer and allow the timer to increment. The gate input signal source can be selected based on the T1GATE register setting. See the T1GATE register (Register 26-4) description for a complete list of the available gate sources. The polarity for each available source is also selectable. Polarity selection is controlled by the GPOL bit of the T1GCON register.

26.6.2.1 T1G Pin Gate Operation

The T1G pin is one source for the timer gate control. It can be used to supply an external source to the time gate circuitry.

26.6.2.2 Timer0 Overflow Gate Operation

When Timer0 overflows, or a period register match condition occurs (in 8-bit mode), a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

26.6.2.3 Comparator C1 Gate Operation

The output resulting from a Comparator 1 operation can be selected as a source for the timer gate control. The Comparator 1 output can be synchronized to the timer clock or left asynchronous. For more information see **Section 23.4.1 "Comparator Output Synchronization"**.

26.6.2.4 Comparator C2 Gate Operation

The output resulting from a Comparator 2 operation can be selected as a source for the timer gate control. The Comparator 2 output can be synchronized to the timer clock or left asynchronous. For more information see **Section 23.4.1 "Comparator Output Synchronization"**.

26.6.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a timer gate signal, as opposed to the duration of a single level pulse.

The timer gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 26-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the GTM bit of the T1GCON register. When the GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation.

26.6.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single-pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the GSPM bit in the T1GCON register. Next, the GGO/$\overline{\text{DONE}}$ bit in the T1GCON register must be set. The timer will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the GGO/$\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment the timer until the GGO/$\overline{\text{DONE}}$ bit is once again set in software. See Figure 26-5 for timing details.

If the Single-Pulse Gate mode is disabled by clearing the GSPM bit in the T1GCON register, the GGO/$\overline{\text{DONE}}$ bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the timer gate source to be measured. See Figure 26-6 for timing details.

26.6.5 TIMER1 GATE VALUE STATUS

When Timer1 Gate Value Status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the GVAL bit in the T1GCON register. The GVAL bit is valid even when the timer gate is not enabled (GE bit is cleared).

26.6.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 Gate Event Interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of GVAL occurs, the TMR1GIF flag bit in the PIR5 register will be set. If the TMR1GIE bit in the PIE5 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the timer gate is not enabled (TMR1GE bit is cleared).

27.5.8 LEVEL RESET, EDGE-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

In Level -Triggered One-Shot mode the timer count is reset on the external signal level and starts counting on the rising/falling edge of the transition from Reset level to the active level while the ON bit is set. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = `01110`)
- High Reset level (MODE<4:0> = `01111`)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control a new external signal edge is required after the ON bit is set to start the counter.

When Level-Triggered Reset One-Shot mode is used in conjunction with the CCP PWM operation the PWM drive goes active with the external signal edge that starts the timer. The PWM drive goes inactive when the timer count equals the CCPRx pulse width count. The PWM drive does not go active when the timer count clears at the PRx period count match.

27.5.9 EDGE-TRIGGERED MONOSTABLE MODES

The Edge-Triggered Monostable modes start the timer on an edge from the external Reset signal input, after the ON bit is set, and stop incrementing the timer when the timer matches the PRx period value. The following edges will start the timer:

- Rising edge (MODE<4:0> = 10001)
- Falling edge (MODE<4:0> = 10010)
- Rising or Falling edge (MODE<4:0> = 10011)

When an Edge-Triggered Monostable mode is used in conjunction with the CCP PWM operation the PWM drive goes active with the external Reset signal edge that starts the timer, but will not go active when the timer matches the PRx value. While the timer is incrementing, additional edges on the external Reset signal will not affect the CCP PWM.

REGISTER 30-3: CWG1DBR: CWG1 RISING DEAD-BAND COUNTER REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	DBR<5:0>					
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **DBR<5:0>:** Rising Event Dead-Band Value for Counter bits

REGISTER 30-4: CWG1DBF: CWG1 FALLING DEAD-BAND COUNTER REGISTER

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	DBF<5:0>					
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	q = Value depends on condition

bit 7-6 **Unimplemented:** Read as ‘0’

bit 5-0 **DBF<5:0>:** Falling Event Dead-Band Value for Counter bits

REGISTER 31-2: CLCxPOL: SIGNAL POLARITY CONTROL REGISTER

R/W-0/0	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LCxPOL	—	—	—	LCxG4POL	LCxG3POL	LCxG2POL	LCxG1POL
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7 **LCxPOL:** CLCxOUT Output Polarity Control bit
1 = The output of the logic cell is inverted
0 = The output of the logic cell is not inverted

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **LCxG4POL:** Gate 3 Output Polarity Control bit
1 = The output of gate 3 is inverted when applied to the logic cell
0 = The output of gate 3 is not inverted

bit 2 **LCxG3POL:** Gate 2 Output Polarity Control bit
1 = The output of gate 2 is inverted when applied to the logic cell
0 = The output of gate 2 is not inverted

bit 1 **LCxG2POL:** Gate 1 Output Polarity Control bit
1 = The output of gate 1 is inverted when applied to the logic cell
0 = The output of gate 1 is not inverted

bit 0 **LCxG1POL:** Gate 0 Output Polarity Control bit
1 = The output of gate 0 is inverted when applied to the logic cell
0 = The output of gate 0 is not inverted

32.2.3 SPI MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK line. The master determines when the slave (Processor 2, Figure 32-5) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPxBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPxSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPxBUF register as if a normal received byte (interrupts and Status bits appropriately set).

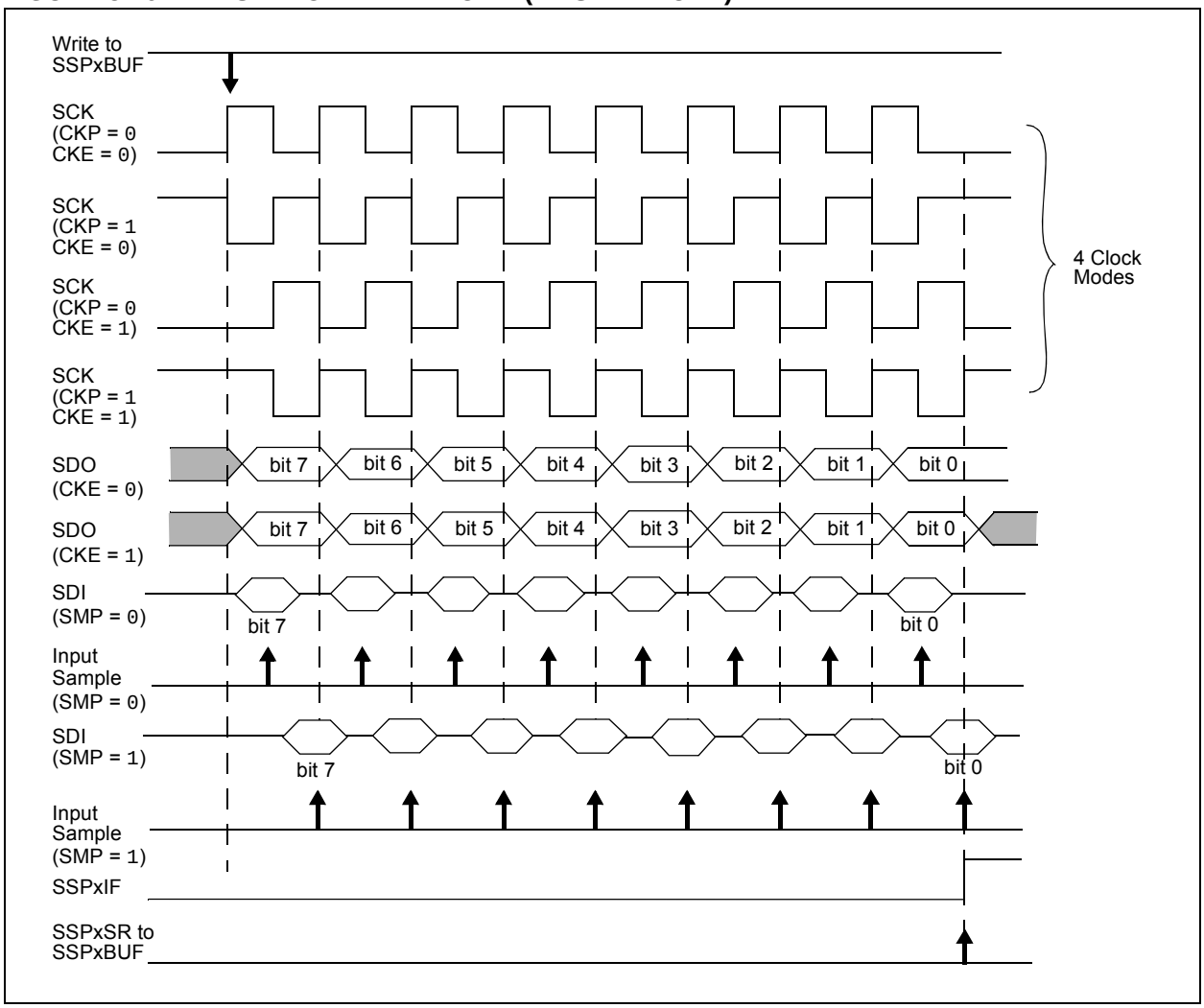
The clock polarity is selected by appropriately programming the CKP bit of the SSPxCON1 register and the CKE bit of the SSPxSTAT register. This then, would give waveforms for SPI communication as shown in Figure 32-6, Figure 32-8, Figure 32-9 and Figure 32-10, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 * TCY)
- FOSC/64 (or 16 * TCY)
- Timer2 output/2
- FOSC/(4 * (SSPxADD + 1))

Figure 32-6 shows the waveforms for Master mode.

When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPxBUF is loaded with the received data is shown.

FIGURE 32-6: SPI MODE WAVEFORM (MASTER MODE)



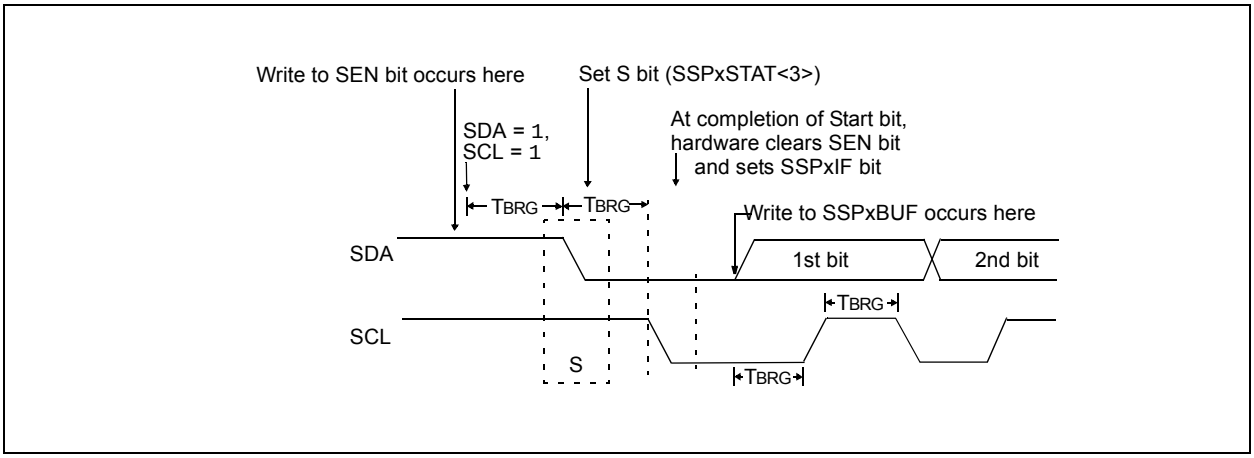
32.6.4 I²C MASTER MODE START CONDITION TIMING

To initiate a Start condition (Figure 32-26), the user sets the Start Enable bit, SEN bit of the SSPxCON2 register. If the SDA and SCL pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and starts its count. If SCL and SDA are both sampled high when the Baud Rate Generator times out (TBRG), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the Start condition and causes the S bit of the SSPxSTAT1 register to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and resumes its count. When the Baud Rate Generator times out (TBRG), the SEN bit of the SSPxCON2 register will be automatically cleared by hardware; the Baud Rate Generator is suspended, leaving the SDA line held low and the Start condition is complete.

Note 1: If at the beginning of the Start condition, the SDA and SCL pins are already sampled low, or if during the Start condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLIF, is set, the Start condition is aborted and the I²C module is reset into its Idle state.

2: The Philips I²C specification states that a bus collision cannot occur on a Start.

FIGURE 32-26: FIRST START BIT TIMING



32.6.6 I²C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address or the other half of a 10-bit address is accomplished by simply writing a value to the SSPxBUF register. This action will set the Buffer Full flag bit, BF, and allow the Baud Rate Generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDA pin after the falling edge of SCL is asserted. SCL is held low for one Baud Rate Generator rollover count (TBRG). Data should be valid before SCL is released high. When the SCL pin is released high, it is held that way for TBRG. The data on the SDA pin must remain stable for that duration and some hold time after the next falling edge of SCL. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDA. This allows the slave device being addressed to respond with an $\overline{ACK}$ bit during the ninth bit time if an address match occurred, or if data was received properly. The status of $\overline{ACK}$ is written into the ACKSTAT bit on the rising edge of the ninth clock. If the master receives an Acknowledge, the Acknowledge Status bit, ACKSTAT, is cleared. If not, the bit is set. After the ninth clock, the SSPxIF bit is set and the master clock (Baud Rate Generator) is suspended until the next data byte is loaded into the SSPxBUF, leaving SCL low and SDA unchanged (Figure 32-28).

After the write to the SSPxBUF, each bit of the address will be shifted out on the falling edge of SCL until all seven address bits and the R/$\overline{W}$ bit are completed. On the falling edge of the eighth clock, the master will release the SDA pin, allowing the slave to respond with an Acknowledge. On the falling edge of the ninth clock, the master will sample the SDA pin to see if the address was recognized by a slave. The status of the $\overline{ACK}$ bit is loaded into the ACKSTAT Status bit of the SSPxCON2 register. Following the falling edge of the ninth clock transmission of the address, the SSPxIF is set, the BF flag is cleared and the Baud Rate Generator is turned off until another write to the SSPxBUF takes place, holding SCL low and allowing SDA to float.

32.6.6.1 BF Status Flag

In Transmit mode, the BF bit of the SSPxSTAT register is set when the CPU writes to SSPxBUF and is cleared when all eight bits are shifted out.

32.6.6.2 WCOL Status Flag

If the user writes the SSPxBUF when a transmit is already in progress (i.e., SSPxSR is still shifting out a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

WCOL must be cleared by software before the next transmission.

32.6.6.3 ACKSTAT Status Flag

In Transmit mode, the ACKSTAT bit of the SSPxCON2 register is cleared when the slave has sent an Acknowledge ($\overline{ACK}$ = 0) and is set when the slave does not Acknowledge ($\overline{ACK}$ = 1). A slave sends an Acknowledge when it has recognized its address (including a general call), or when the slave has properly received its data.

32.6.6.4 Typical transmit sequence:

1. The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
2. SSPxIF is set by hardware on completion of the Start.
3. SSPxIF is cleared by software.
4. The MSSP module will wait the required start time before any other operation takes place.
5. The user loads the SSPxBUF with the slave address to transmit.
6. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
7. The MSSP module shifts in the $\overline{ACK}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
8. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
9. The user loads the SSPxBUF with eight bits of data.
10. Data is shifted out the SDA pin until all eight bits are transmitted.
11. The MSSP module shifts in the $\overline{ACK}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
12. Steps 8-11 are repeated for all transmitted data bytes.
13. The user generates a Stop or Restart condition by setting the PEN or RSEN bits of the SSPxCON2 register. Interrupt is generated once the Stop/Restart condition is complete.

REGISTER 32-4: SSPxCON3: SSPx CONTROL REGISTER 3

R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ACKTIM(3)	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **ACKTIM:** Acknowledge Time Status bit (I^2C mode only)(3)
1 = Indicates the I^2C bus is in an Acknowledge sequence, set on 8th falling edge of SCL clock
0 = Not an Acknowledge sequence, cleared on 9TH rising edge of SCL clock

bit 6 **PCIE**: Stop Condition Interrupt Enable bit (I^2C mode only)
1 = Enable interrupt on detection of Stop condition
0 = Stop detection interrupts are disabled(2)

bit 5 **SCIE**: Start Condition Interrupt Enable bit (I^2C mode only)
1 = Enable interrupt on detection of Start or Restart conditions
0 = Start detection interrupts are disabled(2)

bit 4 **BOEN:** Buffer Overwrite Enable bit
In SPI Slave mode:(1)
1 = SSPxBUF updates every time that a new data byte is shifted in ignoring the BF bit
0 = If new byte is received with BF bit of the SSPxSTAT register already set, SSPOV bit of the SSPxCON1 register is set, and the buffer is not updated
In I^2C Master mode and SPI Master mode:
This bit is ignored.
In I^2C Slave mode:
1 = SSPxBUF is updated and $\overline{ACK}$ is generated for a received address/data byte, ignoring the state of the SSPOV bit only if the BF bit = 0.
0 = SSPxBUF is only updated when SSPOV is clear

bit 3 **SDAHT:** SDA Hold Time Selection bit (I^2C mode only)
1 = Minimum of 300 ns hold time on SDA after the falling edge of SCL
0 = Minimum of 100 ns hold time on SDA after the falling edge of SCL

bit 2 **SBCDE:** Slave Mode Bus Collision Detect Enable bit (I^2C Slave mode only)
If, on the rising edge of SCL, SDA is sampled low when the module is outputting a high state, the BCL1IF bit of the PIR3 register is set, and bus goes idle
1 = Enable slave bus collision interrupts
0 = Slave bus collision interrupts are disabled

bit 1 **AHEN:** Address Hold Enable bit (I^2C Slave mode only)
1 = Following the eighth falling edge of SCL for a matching received address byte; CKP bit of the SSPxCON1 register will be cleared and the SCL will be held low.
0 = Address holding is disabled

bit 0 **DHEN:** Data Hold Enable bit (I^2C Slave mode only)
1 = Following the eighth falling edge of SCL for a received data byte; slave hardware clears the CKP bit of the SSPxCON1 register and SCL is held low.
0 = Data holding is disabled

Note 1: For daisy-chained SPI operation; allows the user to ignore all but the last received byte. SSPOV is still set when a new byte is received and BF = 1, but hardware continues to write the most recent byte to SSPxBUF.
2: This bit has no effect in Slave modes that Start and Stop condition detection is explicitly listed as enabled.
3: The ACKTIM Status bit is only active when the AHEN bit or DHEN bit is set.

FIGURE 35-2: PICkit™ PROGRAMMER STYLE CONNECTOR INTERFACE

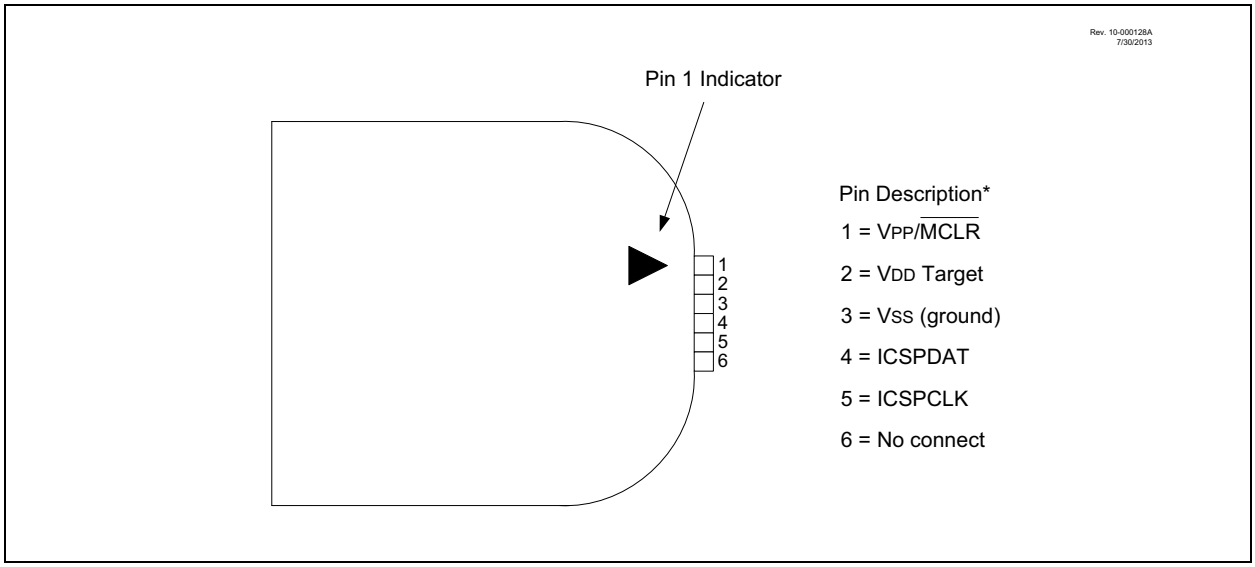


FIGURE 35-3: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING

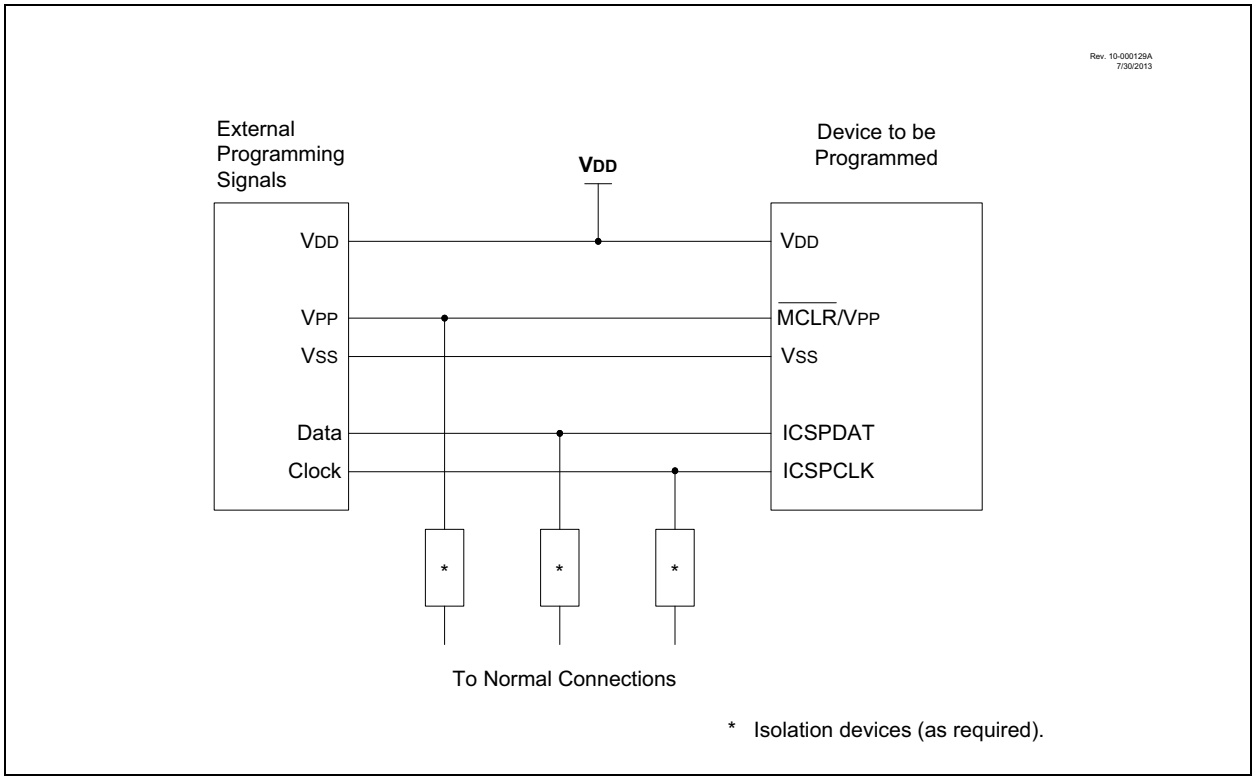
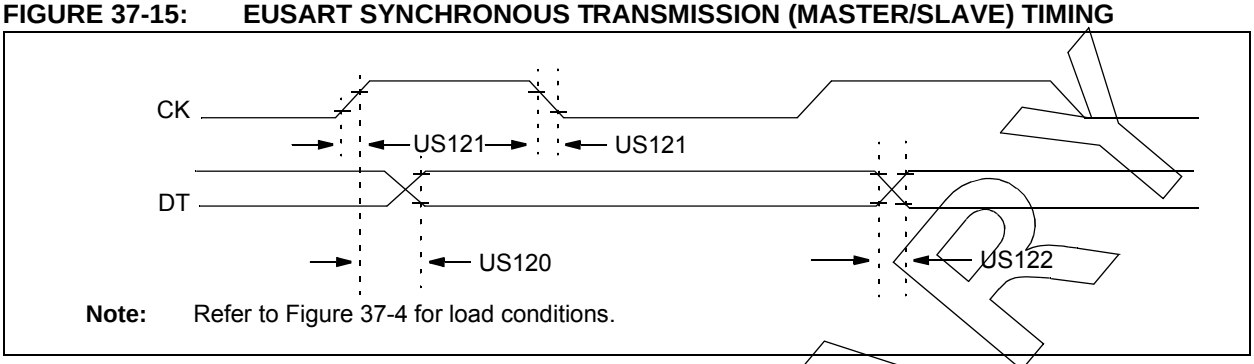


FIGURE 37-15: EUSART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

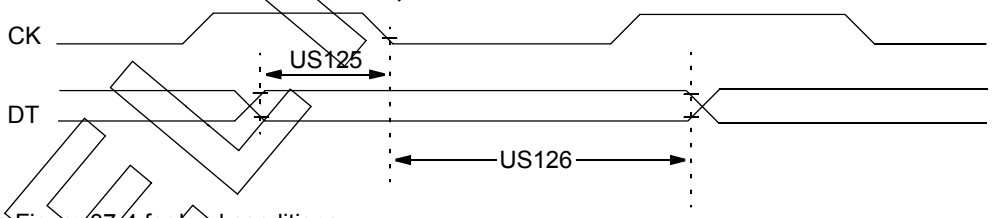


Note: Refer to Figure 37-4 for load conditions.

TABLE 37-21: EUSART SYNCHRONOUS TRANSMISSION CHARACTERISTICS

Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US120	TCKH2DTV	SYNC XMIT (Master and Slave) Clock high to data-out valid	—	80	ns	3.0V ≤ VDD ≤ 5.5V
			—	100	ns	1.8V ≤ VDD ≤ 5.5V
US121	TCKRF	Clock out rise time and fall time (Master mode)	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V
US122	TDTRF	Data-out rise time and fall time	—	45	ns	3.0V ≤ VDD ≤ 5.5V
			—	50	ns	1.8V ≤ VDD ≤ 5.5V

FIGURE 37-16: EUSART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING



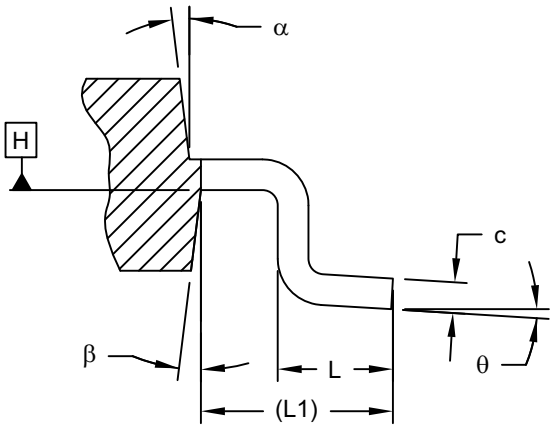
Note: Refer to Figure 37-4 for load conditions.

TABLE 37-22: EUSART SYNCHRONOUS RECEIVE REQUIREMENTS

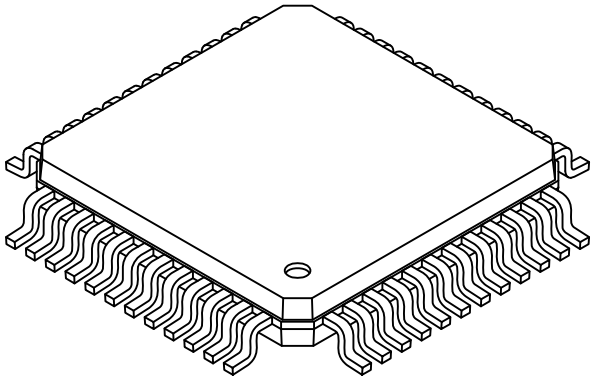
Standard Operating Conditions (unless otherwise stated)						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
US125	TDTV2CKL	SYNC RCV (Master and Slave) Data-setup before CK ↓ (DT hold time)	10	—	ns	
US126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	—	ns	

48-Lead Thin Quad Flatpack (PT) - 7x7x1.0 mm Body [TQFP]

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



SECTION A-A



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Leads	N		48	
Lead Pitch	e		0.50 BSC	
Overall Height	A	-	-	1.20
Standoff	A1	0.05	-	0.15
Molded Package Thickness	A2	0.95	1.00	1.05
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E		9.00 BSC	
Overall Length	D		9.00 BSC	
Molded Package Width	E1		7.00 BSC	
Molded Package Length	D1		7.00 BSC	
Lead Thickness	c	0.09	-	0.16
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M

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

   REF: Reference Dimension, usually without tolerance, for information purposes only.
5. Datums A-B and D to be determined at center line between leads where leads exit plastic body at datum plane H

Microchip Technology Drawing C04-300-PT Rev A Sheet 2 of 2