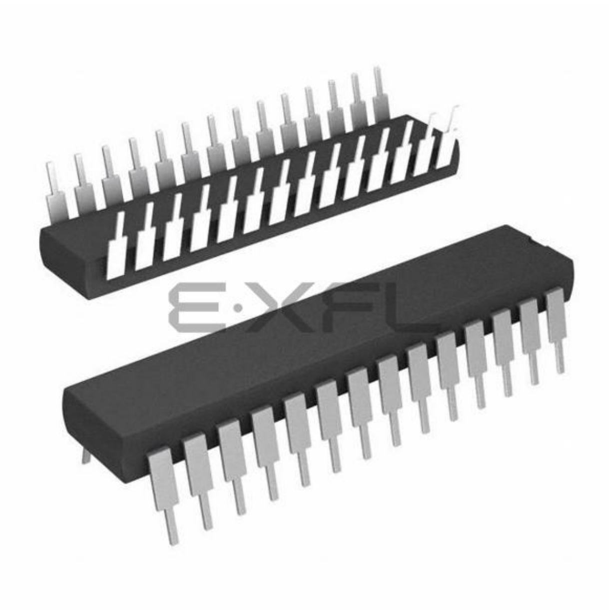
Microchip Technology - PIC16LF18455-E/SP Datasheet



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	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	26
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.300", 7.62mm)
Supplier Device Package	28-SPDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18455-e-sp

7.1 Program Memory Organization
7.2.3 Storage Area Flash
7.2.5 Memory Violation

8.12 Determining the Cause of a Reset

Upon any Reset, multiple bits in the STATUS and PCON0 registers are updated to indicate the cause of the Reset. The following tables show the Reset conditions of these registers.

Table 8-3. Reset Status Bits and Their Significance

STOVF	STKUNF	$\overline{\text{RWDT}}$	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	$\overline{\text{MEMV}}$	Condition
0	0	1	1	1	0	x	1	1	1	Power-on Reset
0	0	1	1	1	0	x	0	x	u	Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$
0	0	1	1	1	0	x	x	0	u	Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$
0	0	u	1	1	u	0	1	1	u	Brown-out Reset
u	u	0	u	u	u	u	0	u	u	WWDT Reset
u	u	u	u	u	u	u	0	0	u	WWDT Wake-up from Sleep
u	u	u	u	u	u	u	1	0	u	Interrupt Wake-up from Sleep
u	u	u	0	u	u	u	u	u	1	$\overline{\text{MCLR}}$ Reset during normal operation
u	u	u	0	u	u	u	1	0	u	$\overline{\text{MCLR}}$ Reset during Sleep
u	u	u	u	0	u	u	u	u	u	RESET Instruction Executed
1	u	u	u	u	u	u	u	u	u	Stack Overflow Reset (STVREN = 1)
u	1	u	u	u	u	u	u	u	u	Stack Underflow Reset (STVREN = 1)
u	u	u	u	u	u	u	u	u	0	Memory violation Reset

Table 8-4. Reset Condition for Special Registers

Condition	Program Counter	STATUS Register	PCON0 Register	PCON1 Register
Power-on Reset	0	---1 1000	0011 110x	---- --1-
Brown-out Reset	0	---1 1000	0011 11u0	---- --u-
$\overline{\text{MCLR}}$ Reset during normal operation	0	-uuu uuuu	uuuu 0uuu	---- --1-
$\overline{\text{MCLR}}$ Reset during Sleep	0	---1 0uuu	uuuu 0uuu	---- --u-
WWDT Time-out Reset	0	---0 uuuu	uuu0 uuuu	---- --u-
WWDT Wake-up from Sleep	PC + 1	---0 0uuu	uuuu uuuu	---- --u-

12.8.4 WDTPSL

Name: WDTPSL
Address: 0x80E

WWDT Prescale Select Low Register (Read-Only)

Bit	7	6	5	4	3	2	1	0
	PSCNTL[7:0]							
Access	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0

Bits 7:0 – PSCNTL[7:0] Prescale Select Low Byte bits[1]

Note:
1. The 18-bit WDT prescale value, PSCNT<17:0> includes the WDTPSL, WDTPSH and the lower bits of the WDTTMR registers. PSCNT<17:0> is intended for debug operations and should be read during normal operation.

13.6.3 NVMCON1

Name: NVMCON1
Address: 0x81E

Nonvolatile Memory Control 1 Register

Bit	7	6	5	4	3	2	1	0
		NVMREGS	LWLO	FREE	WRERR	WREN	WR	RD
Access		R/W	R/W	R/S/HC	R/W/HS	R/W	R/S/HC	R/S/HC
Reset		0	0	0	x	0	0	0

Bit 6 – NVMREGS NVM Region Selection bit

Value	Description
1	Access EEPROM, DIA, DCI, Configuration, User ID and Device ID Registers
0	Access Program Flash Memory

Bit 5 – LWLO Load Write Latches Only bit

Value	Condition	Description
1	When FREE = 0	The next WR command updates the write latch for this word within the row; no memory operation is initiated.
0	When FREE = 0	The next WR command writes data or erases
–	Otherwise:	This bit is ignored.

Bit 4 – FREE Program Flash Memory Erase Enable bit

Value	Description
1	Performs an erase operation with the next WR command; the 32-word pseudo-row containing the indicated address is erased (to all 1s) to prepare for writing.
0	The next WR command writes without erasing.

Bit 3 – WRERR
Write-Reset Error Flag bit(1,2,3)
Reset States: POR/BOR = x
All Other Resets = q

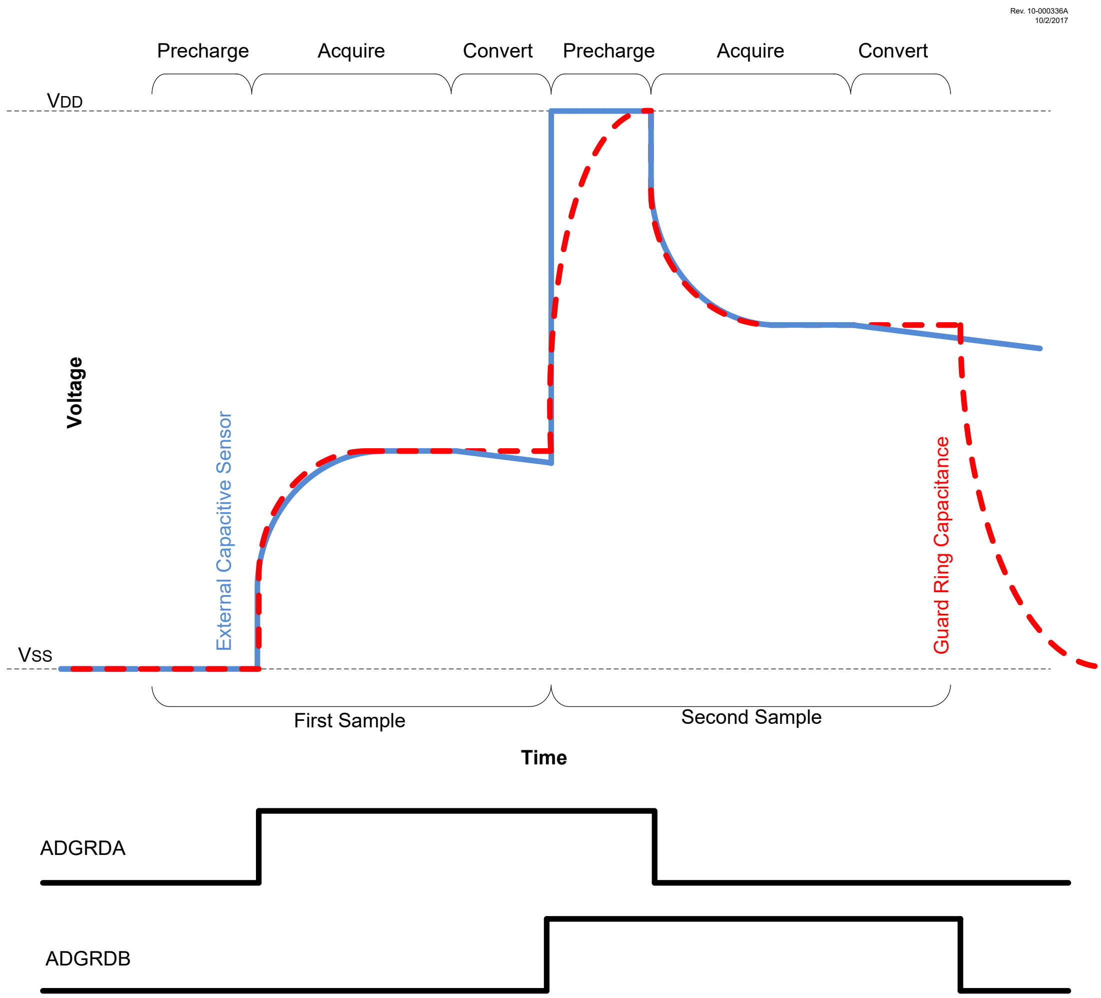
Value	Description
1	A write operation was interrupted by a Reset, interrupted unlock sequence, or WR was written to one while NVMADR points to a write-protected address.
0	All write operations have completed normally.

Bit 2 – WREN Program/Erase Enable bit

Value	Description
1	Allows program/erase cycles
0	Inhibits programming/erasing of program Flash

Bit 1 – WR Write Control bit(4,5,6)

Figure 20-8. Differential CVD with Guard Ring Output Waveform



Related Links

20.8.2 ADCON1

15. (PPS) Peripheral Pin Select Module

20.5.5 Additional Sample and Hold Capacitance

Additional capacitance can be added in parallel with the internal sample and hold capacitor (C_{HOLD}) by using the ADCAP register. This register selects a digitally programmable capacitance, which is added to the ADC conversion bus, increasing the effective internal capacitance of the sample and hold capacitor in the ADC module. This is used to improve the match between internal and external capacitance for a better sensing performance. The additional capacitance does not affect analog performance of the ADC because it is not connected during conversion.

Related Links

20.6 Computation Operation

20.8.11 ADCAP

Mode	MODE<4:0>		Output Operation	Operation	Timer Control		
	<4:3>	<2:0>			Start	Reset	Stop
		100	Edge Triggered Start and Hardware Reset (Note 1)	Rising edge start and Rising edge Reset (Figure 27-9)	ON = 1 and TMRx_ers ↑	TMRx_ers ↑	
		101		Falling edge start and Falling edge Reset	ON = 1 and TMRx_ers ↓	TMRx_ers ↓	
		110		Rising edge start and Low level Reset (Figure 27-10)	ON = 1 and TMRx_ers ↑	TMRx_ers = 0	
		111		Falling edge start and High level Reset	ON = 1 and TMRx_ers ↓	TMRx_ers = 1	
Mono-stable	10	000	Reserved				
		001	Edge Triggered Start (Note 1)	Rising edge start (Figure 27-11)	ON = 1 and TMRx_ers ↑	—	ON = 0 or Next clock after TMRx = PRx (Note 3)
		010		Falling edge start	ON = 1 and TMRx_ers ↓	—	
		011		Any edge start	ON = 1 and TMRx_ers ↕	—	
Reserved		100	Reserved				
Reserved		101	Reserved				
One-shot		110	Level Triggered Start and Hardware Reset	High level start and Low level Reset (Figure 27-12)	ON = 1 and TMRx_ers = 1	TMRx_ers = 0	ON = 0 or Held in Reset (Note 2)
		111		Low level start & High level Reset	ON = 1 and TMRx_ers = 0	TMRx_ers = 1	

30. (PWM) Pulse-Width Modulation

The PWM module generates a Pulse-Width Modulated signal determined by the duty cycle, period, and resolution that are configured by the following registers:

- TxPR
- TxCON
- PWMxDC
- PWMxCON

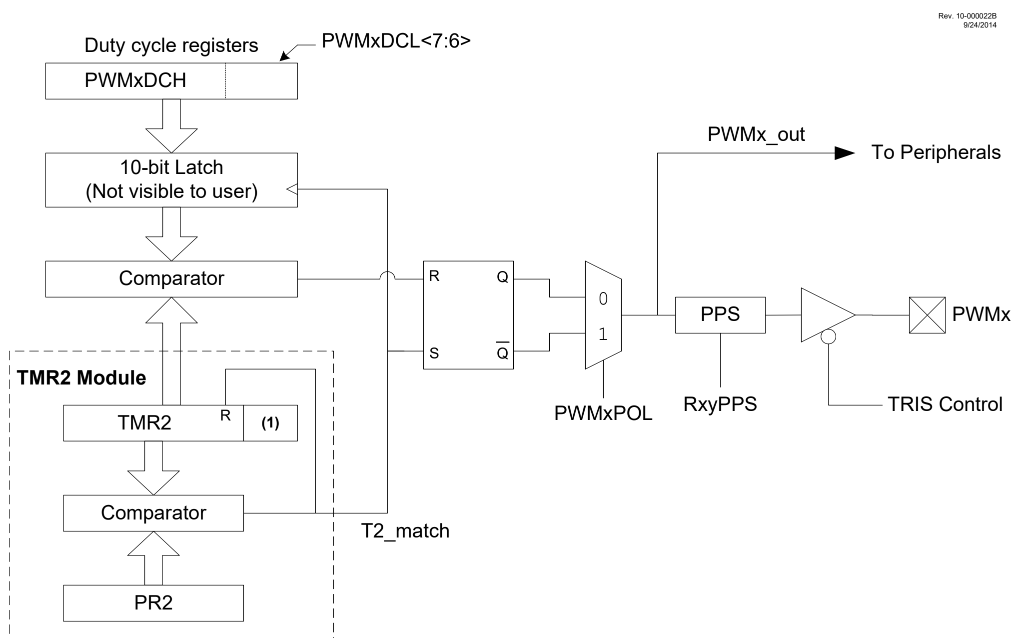
Important: The corresponding TRIS bit must be cleared to enable the PWM output on the PWMx pin.

Each PWM module can select the timer source that controls the module. Note that the PWM mode operation is described with respect to TMR2 in the following sections.

Figure 30-1 shows a simplified block diagram of PWM operation.

Figure 30-2 shows a typical waveform of the PWM signal.

Figure 30-1. Simplified PWM Block Diagram



Note:

1. 8-bit timer is concatenated with two bits generated by Fosc or two bits of the internal prescaler to create 10-bit time base.

ISM	Data Source
0001	CCP1_out
0000	Pin selected by CWGxINPPS

31.6 Output Control

31.6.1 CWG Outputs

Each CWG output can be routed to a Peripheral Pin Select (PPS) output via the RxyPPS register.

Related Links

15. (PPS) Peripheral Pin Select Module

31.6.2 Polarity Control

The polarity of each CWG output can be selected independently. When the output polarity bit is set, the corresponding output is active-high. Clearing the output polarity bit configures the corresponding output as active-low. However, polarity does not affect the override levels. Output polarity is selected with the POLy bits. Auto-shutdown and steering options are unaffected by polarity.

31.7 Dead-Band Control

The dead-band control provides non-overlapping PWM signals to prevent shoot-through current in PWM switches. Dead-band operation is employed for Half-Bridge and Full-Bridge modes. The CWG contains two 6-bit dead-band counters. One is used for the rising edge of the input source control in Half-Bridge mode or for reverse dead-band Full-Bridge mode. The other is used for the falling edge of the input source control in Half-Bridge mode or for forward dead band in Full-Bridge mode.

Dead band is timed by counting CWG clock periods from zero up to the value in the rising or falling dead-band counter registers.

31.7.1 Dead-Band Functionality in Half-Bridge mode

In Half-Bridge mode, the dead-band counters dictate the delay between the falling edge of the normal output and the rising edge of the inverted output. This can be seen in Figure 31-1.

31.7.2 Dead-Band Functionality in Full-Bridge mode

In Full-Bridge mode, the dead-band counters are used when undergoing a direction change. The MODE<0> bit can be set or cleared while the CWG is running, allowing for changes from Forward to Reverse mode. The CWGxA and CWGxC signals will change immediately upon the first rising input edge following a direction change, but the modulated signals (CWGxB or CWGxD, depending on the direction of the change) will experience a delay dictated by the dead-band counters.

31.8 Rising Edge and Reverse Dead Band

In Half-Bridge mode, the rising edge dead band delays the turn-on of the CWGxA output after the rising edge of the CWG data input. In Full-Bridge mode, the reverse dead-band delay is only inserted when changing directions from Forward mode to Reverse mode, and only the modulated output CWGxB is affected.

The 31.15.8 CWGxDBR register determines the duration of the dead-band interval on the rising edge of the input source signal. This duration is from 0 to 64 periods of the CWG clock.

32.11 Register Summary - DSM

Address	Name	Bit Pos.								
0x0897	MD1CON0	7:0	EN		OUT	OPOL				BIT
0x0898	MD1CON1	7:0			CHPOL	CHSYNC			CLPOL	CLSYNC
0x0899	MD1SRC	7:0				SRCS[4:0]				
0x089A	MD1CARL	7:0					CLS[3:0]			
0x089B	MD1CARH	7:0					CHS[3:0]			

32.12 Register Definitions: Modulation Control

Long bit name prefixes for the Modulation Control peripherals are shown in the table below. Refer to the *"Long Bit Names Section"* for more information.

Table 32-4. Modulation Control Long Bit Name Prefixes

Peripheral	Bit Name Prefix
MD	MD

Related Links

1.4.2.2 Long Bit Names

The SSPSR is not directly readable or writable and can only be accessed by addressing the SSPxBUF register. Additionally, the SSPxSTAT register indicates the various Status conditions.

35.2.1 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 35-3) 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 SSPSR 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 and the CKE bit. This then, would give waveforms for SPI communication as shown in Figure 35-4, Figure 35-6, Figure 35-7 and Figure 35-8, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- $F_{OSC}/4$ (or T_{CY})
- $F_{OSC}/16$ (or 4 * T_{CY})
- $F_{OSC}/64$ (or 16 * T_{CY})
- Timer2 output/2
- $F_{OSC}/(4 * (SSPxADD + 1))$

Figure 35-4 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.

Important: In Master mode the clock signal output to the SCK pin is also the clock signal input to the peripheral. The pin selected for output with the RxyPPS register must also be selected as the peripheral input with the SSPxCLKPPS register. The pin that is selected using the SSPxCLKPPS register should also be made a digital I/O. This is done by clearing the corresponding ANSEL bit.

is held low until SSPxADD is updated to receive a high byte again. When SSPxADD is updated the UA bit is cleared. This ensures the module is ready to receive the high address byte on the next communication.

A high and low address match as a write request is required at the start of all 10-bit addressing communication. A transmission can be initiated by issuing a Restart once the slave is addressed, and clocking in the high address with the R/$\overline{W}$ bit set. The slave hardware will then acknowledge the read request and prepare to clock out data. This is only valid for a slave after it has received a complete high and low address byte match.

35.5.2 Slave Reception

When the R/$\overline{W}$ bit of a matching received address byte is clear, the R/W bit is cleared. The received address is loaded into the SSPxBUF register and acknowledged.

When the overflow condition exists for a received address, then not Acknowledge is given. An overflow condition is defined as either bit BF is set, or bit SSPOV is set. The BOEN bit modifies this operation. For more information see SSPxCON3.

An MSSP interrupt is generated for each transferred data byte. Flag bit, SSPxIF, must be cleared by software.

When the SEN bit is set, SCL will be held low (clock stretch) following each received byte. The clock must be released by setting the CKP bit, except sometimes in 10-bit mode. See 35.5.6.2 10-bit Addressing Mode for more detail.

35.5.2.1 7-bit Addressing Reception

This section describes a standard sequence of events for the MSSP module configured as an I²C slave in 7-bit Addressing mode. Figure 35-14 and Figure 35-15 is used as a visual reference for this description.

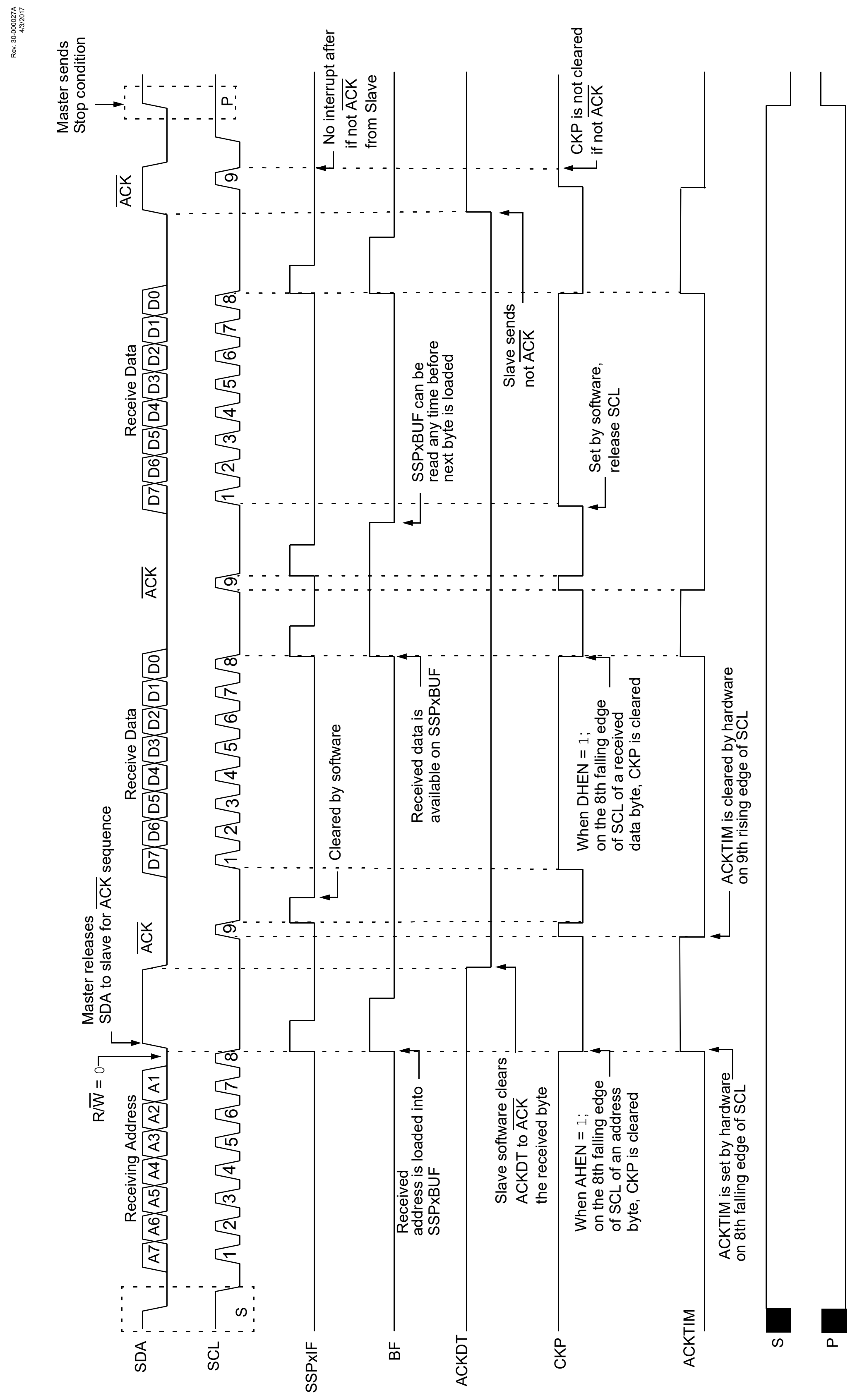
This is a step by step process of what typically must be done to accomplish I²C communication.

1. Start bit detected.
2. S bit is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Matching address with R/W bit clear is received.
4. The slave pulls SDA low sending an $\overline{ACK}$ to the master, and sets SSPxIF bit.
5. Software clears the SSPxIF bit.
6. Software reads received address from SSPxBUF clearing the BF flag.
7. If SEN = `1`; Slave software sets CKP bit to release the SCL line.
8. The master clocks out a data byte.
9. Slave drives SDA low sending an $\overline{ACK}$ to the master, and sets SSPxIF bit.
10. Software clears SSPxIF.
11. Software reads the received byte from SSPxBUF clearing BF.
12. Steps 8-12 are repeated for all received bytes from the master.
13. Master sends Stop condition, setting P bit, and the bus goes idle.

35.5.2.2 7-bit Reception with AHEN and DHEN

Slave device reception with AHEN and DHEN set operate the same as without these options with extra interrupts and clock stretching added after the eighth falling edge of SCL. These additional interrupts allow the slave software to decide whether it wants to $\overline{ACK}$ the receive address or data byte, rather than the hardware. This functionality adds support for PMBus™ that was not present on previous versions of this module.

Figure 35-17. I²C Slave, 7-bit Address, Reception (SEN = 1, AHEN = 1, DHEN = 1)



4. Write ‘55h’ to TXxREG to load the Sync character into the transmit FIFO buffer.
5. After the Break has been sent, the SENDB bit is reset by hardware and the Sync character is then transmitted.

When the TXxREG becomes empty, as indicated by the TXxIF, the next data byte can be written to TXxREG.

36.2.5 Receiving a Break Character

The EUSART module can receive a Break character in two ways.

The first method to detect a Break character uses the FERR bit of the RCxSTA register and the received data as indicated by RCxREG. The Baud Rate Generator is assumed to have been initialized to the expected baud rate.

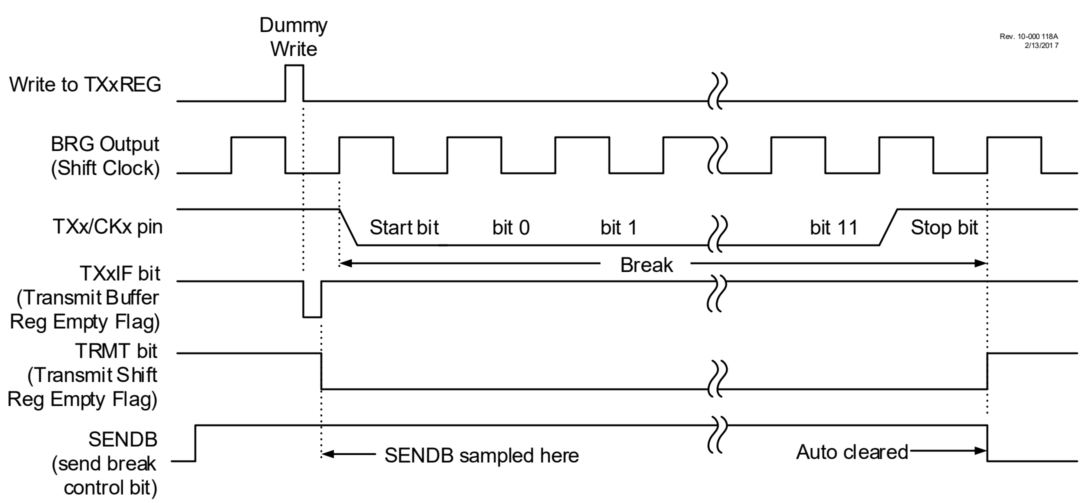
A Break character has been received when all three of the following conditions are true:

- RCxIF bit is set
- FERR bit is set
- RCxREG = 00h

The second method uses the Auto-Wake-up feature described in 36.2.3 Auto-Wake-up on Break. By enabling this feature, the EUSART will sample the next two transitions on RX/DT, cause an RCxIF interrupt, and receive the next data byte followed by another interrupt.

Note that following a Break character, the user will typically want to enable the Auto-Baud Detect feature. For both methods, the user can set the ABDEN bit of the BAUDxCON register before placing the EUSART in Sleep mode.

Figure 36-9. Send Break Character Sequence



36.3 EUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

38. Register Summary

Address	Name	Bit Pos.								
0x00	INDF0	7:0	INDF0[7:0]							
0x01	INDF1	7:0	INDF1[7:0]							
0x02	PCL	7:0	PCL[7:0]							
0x03	STATUS	7:0				$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
0x04	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x06	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x08	BSR	7:0			BSR[5:0]					
0x09	WREG	7:0	WREG[7:0]							
0x0A	PCLATH	7:0		PCLATH[6:0]						
0x0B	INTCON	7:0	GIE	PEIE						INTEDG
0x0C	PORTA	7:0	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
0x0D	PORTB	7:0	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
0x0E	PORTC	7:0	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
0x0F	Reserved									
0x10	PORTE	7:0					RE3			
0x11	Reserved									
0x12	TRISA	7:0	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
0x13	TRISB	7:0	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
0x14	TRISC	7:0	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
0x15 ... 0x17	Reserved									
0x18	LATA	7:0	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0
0x19	LATB	7:0	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0
0x1A	LATC	7:0	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
0x1B ... 0x7F	Reserved									
0x80	INDF0	7:0	INDF0[7:0]							
0x81	INDF1	7:0	INDF1[7:0]							
0x82	PCL	7:0	PCL[7:0]							
0x83	STATUS	7:0				$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
0x84	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x86	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x88	BSR	7:0			BSR[5:0]					
0x89	WREG	7:0	WREG[7:0]							
0x8A	PCLATH	7:0		PCLATH[6:0]						
0x8B	INTCON	7:0	GIE	PEIE						INTEDG
0x8C	ADLTH	7:0	LTHL[7:0]							

Address	Name	Bit Pos.								
0x059E	T0CON0	7:0	T0EN		T0OUT	T016BIT	T0OUTPS[3:0]			
0x059F	T0CON1	7:0	T0CS[2:0]			T0ASYNC	T0CKPS[3:0]			
0x05A0 ... 0x05FF	Reserved									
0x0600	INDF0	7:0	INDF0[7:0]							
0x0601	INDF1	7:0	INDF1[7:0]							
0x0602	PCL	7:0	PCL[7:0]							
0x0603	STATUS	7:0				$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
0x0604	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0606	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0608	BSR	7:0			BSR[5:0]					
0x0609	WREG	7:0	WREG[7:0]							
0x060A	PCLATH	7:0		PCLATH[6:0]						
0x060B	INTCON	7:0	GIE	PEIE						INTEDG
0x060C	CWG1CLK	7:0								CS
0x060D	CWG1ISM	7:0					ISM[3:0]			
0x060E	CWG1DBR	7:0			DBR[5:0]					
0x060F	CWG1DBF	7:0			DBF[5:0]					
0x0610	CWG1CON0	7:0	EN	LD				MODE[2:0]		
0x0611	CWG1CON1	7:0			IN		POLD	POLC	POLB	POLA
0x0612	CWG1AS0	7:0	SHUTDOWN	REN	LSBD[1:0]		LSAC[1:0]			
0x0613	CWG1AS1	7:0			AS5E	AS4E	AS3E	AS2E	AS1E	AS0E
0x0614	CWG1STR	7:0	OVRD	OVRC	OVRB	OVRA	STRD	STRC	STRB	STRA
0x0615	Reserved									
0x0616	CWG2CLK	7:0								CS
0x0617	CWG2ISM	7:0					ISM[3:0]			
0x0618	CWG2DBR	7:0			DBR[5:0]					
0x0619	CWG2DBF	7:0			DBF[5:0]					
0x061A	CWG2CON0	7:0	EN	LD				MODE[2:0]		
0x061B	CWG2CON1	7:0			IN		POLD	POLC	POLB	POLA
0x061C	CWG2AS0	7:0	SHUTDOWN	REN	LSBD[1:0]		LSAC[1:0]			
0x061D	CWG2AS1	7:0			AS5E	AS4E	AS3E	AS2E	AS1E	AS0E
0x061E	CWG2STR	7:0	OVRD	OVRC	OVRB	OVRA	STRD	STRC	STRB	STRA
0x061F ... 0x067F	Reserved									
0x0680	INDF0	7:0	INDF0[7:0]							
0x0681	INDF1	7:0	INDF1[7:0]							
0x0682	PCL	7:0	PCL[7:0]							
0x0683	STATUS	7:0				$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
0x0684	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0686	FSR1	7:0	FSRL[7:0]							

41.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

41.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

41.9 PICkit 3 In-Circuit Debugger/Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

41.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at Vddmin and Vddmax for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

41.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping

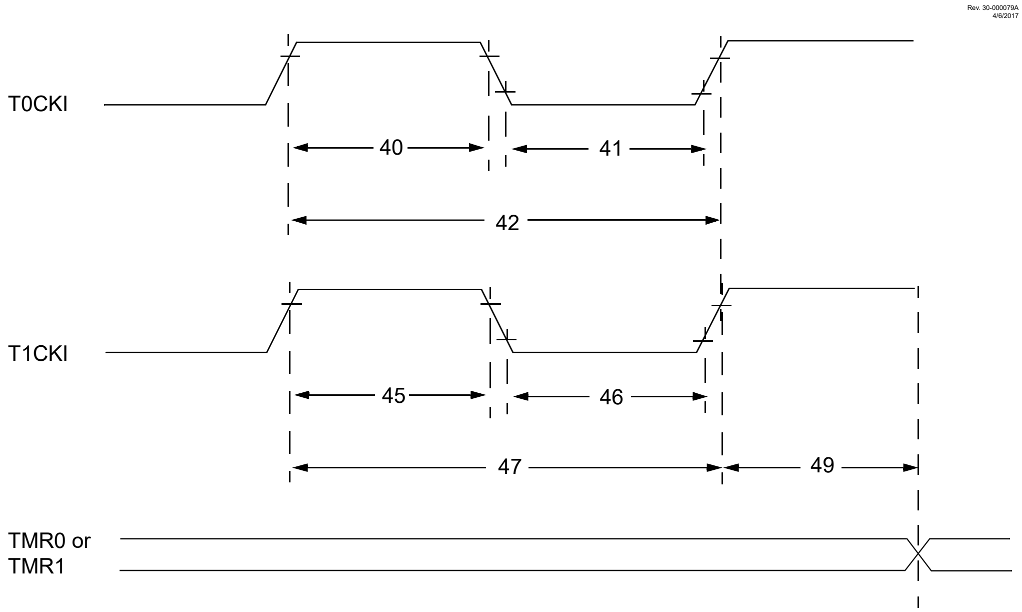
Standard Operating Conditions (unless otherwise stated)							
V_{DD} = 3.0V, T_A = 25°C, T_{AD} = 1μs							
Param No.	Sym.	Characteristic	Min.	Typ. †	Max.	Units	Conditions
† Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested. **Note:** 1. Total Absolute Error is the sum of the offset, gain and integral non-linearity (INL) errors. 2. The ADC conversion result never decreases with an increase in the input and has no missing codes.							

42.4.8 Analog-to-Digital Converter (ADC) Conversion Timing Specifications

Table 42-14.

Standard Operating Conditions (unless otherwise stated)							
Param No.	Sym.	Characteristic	Min.	Typ. †	Max.	Units	Conditions
AD20	T_{AD}	ADC Clock Period	1	—	9	μs	Using F_{OSC} as the ADC clock source ADCS = 1
AD21			—	2	—	μs	Using F_{RC} as the ADC clock source ADCS = 0
AD22	T_{CNV}	Conversion Time(1)	—	$13T_{AD}+3T_{CY}$	—	—	Using F_{OSC} as the ADC clock source ADCS = 1
			—	$16T_{AD}+2T_{CY}$	—	—	Using F_{RC} as the ADC clock source ADCS = 0
AD23	T_{ACQ}	Acquisition Time	—	2	—	μs	
AD24	T_{HCD}	Sample and Hold Capacitor Disconnect Time	—	$2T_{AD}+1T_{CY}$	—	—	Using F_{OSC} as the ADC clock source ADCS = 1
			—	$3T_{AD}+2T_{CY}$	—	—	Using F_{RC} as the ADC clock source ADCS = 0
* - These parameters are characterized but not tested. † Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested. **Note:** 1. Does not apply for the ADCRC oscillator.							

Figure 42-12. Timer0 and Timing1 External Clock Timings

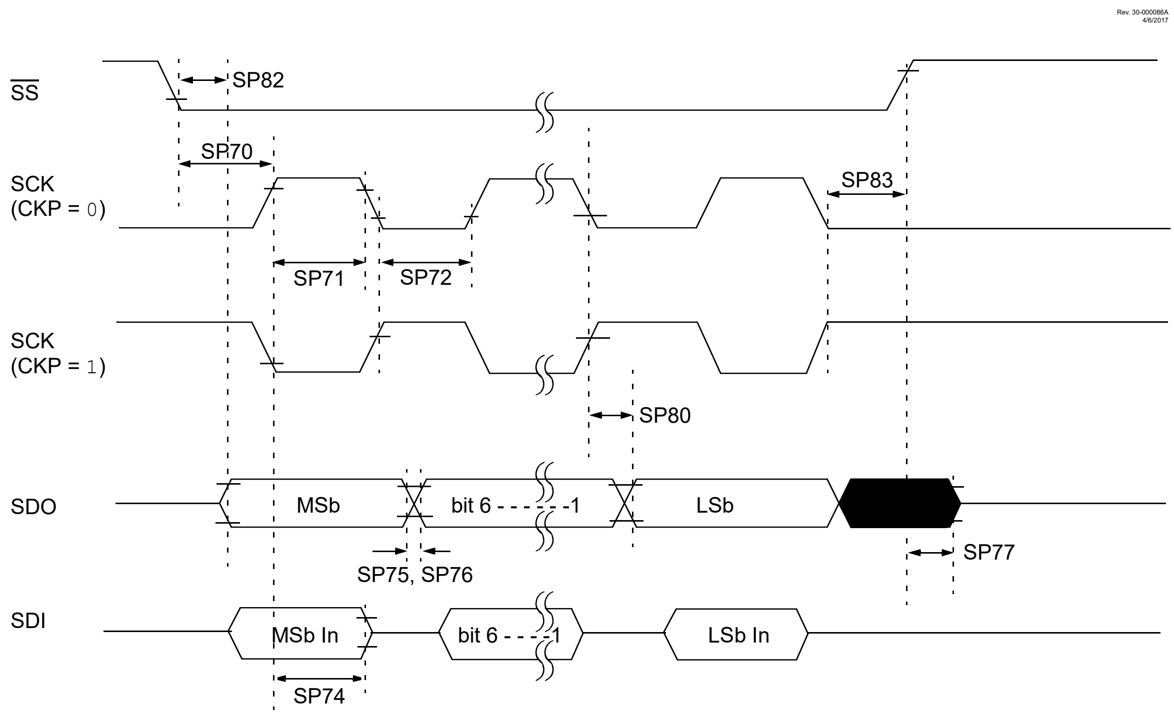


42.4.14 Capture/Compare/PWM Requirements (CCP)

Table 42-20.

Standard Operating Conditions (unless otherwise stated) Operating Temperature: $-40°C \le T_A \le +125°C$								
Param No.	**Sym.**	**Characteristic**		**Min.**	**Typ. †**	**Max.**	**Units**	**Conditions**
CC01*	$T_{CC}L$	CCPx Input Low Time	No Prescaler	$0.5T_{CY}$+20	—	—	ns	
			With Prescaler	20	—	—	ns	
CC02*	$T_{CC}H$	CCPx Input High Time	No Prescaler	$0.5T_{CY}$+20	—	—	ns	
			With Prescaler	20	—	—	ns	
CC03*	$T_{CC}P$	CCPx Input Period		$(3T_{CY}+40)/N$	—	—	ns	N = Prescale value
* - These parameters are characterized but not tested. † Data in “Typ” column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.								

Figure 42-20. SPI Slave Mode Timing (CKE = 1)



Note: Refer to Figure 42-4 for load conditions.

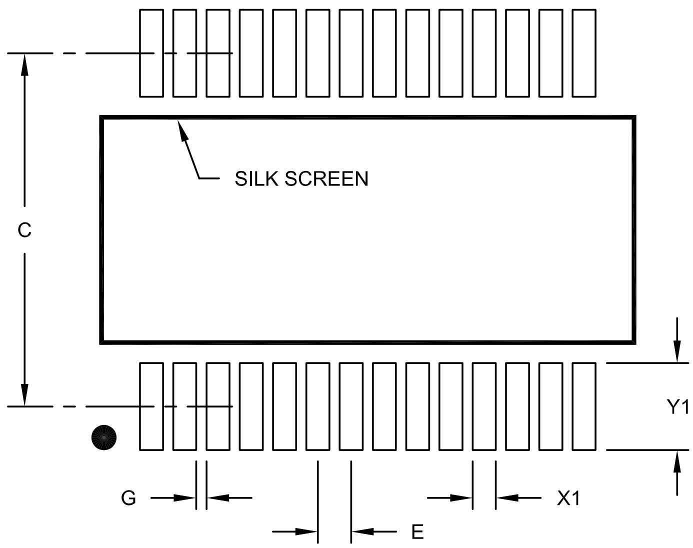
42.4.19 I²C Bus Start/Stop Bits Requirements

Table 42-25.

Standard Operating Conditions (unless otherwise stated)								
Param. No.	**Sym.**	**Characteristic**		**Min.**	**Typ. †**	**Max.**	**Units**	**Conditions**
SP90*	$T_{SU:STA}$	Start condition Setup time	100 kHz mode	4700	—	—	ns	Only relevant for Repeated Start Setup time 400 kHz mode 600 condition
			400 kHz mode	600	—	—		
SP91*	$T_{HD:STA}$	Start condition Hold time	100 kHz mode	4000	—	—	ns	After this period, the first clock Hold time 400 kHz mode 600 — — pulse is generated
			400 kHz mode	600	—	—		
SP92*	$T_{SU:STO}$	Stop condition Setup time	100 kHz mode	4700	—	—	ns	
			400 kHz mode	600	—	—		
SP93*	$T_{HD:STO}$	Stop condition Hold time	100 kHz mode	4000	—	—	ns	

28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X28)	X1			0.45
Contact Pad Length (X28)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2073A