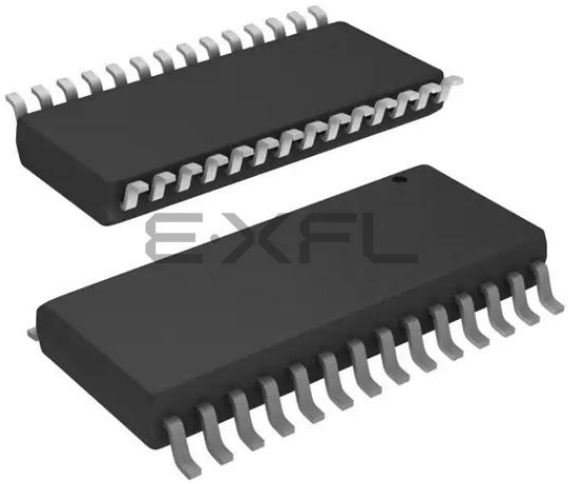


Welcome to E-XFL.COM

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	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 24x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18456t-i-so

Value	Description
1	Boot Block NOT write-protected
0	Boot Block write-protected

Bit 7 – $\overline{\text{WRTAPP}}$ Application Block Write Protection bit(1)

Value	Description
1	Application Block NOT write-protected
0	Application Block write-protected

Bit 4 – $\overline{\text{SAFEN}}$ SAF Enable bit(1)

Value	Description
1	SAF disabled
0	SAF enabled

Bit 3 – $\overline{\text{BBEN}}$ Boot Block Enable bit(1)

Value	Description
1	Boot Block disabled
0	Boot Block enabled

Bits 2:0 – BBSIZE[2:0] Boot Block Size Selection bits
BBSIZE is used only when $\overline{\text{BBEN}}$ = 0

BBSIZE bits can only be written while $\overline{\text{BBEN}}$ = 1; after $\overline{\text{BBEN}}$ = 0, BBSIZ is write-protected.

Table 4-1. Boot Block Size Bits

$\overline{\text{BBEN}}$	BBSIZE	Actual Boot Block Size User Program Memory Size (words)		Last Boot Block Memory Access
		PIC16(L)F18455	PIC16(L)F18456	
1	xxx	0	0	—
0	111	512	512	01FFh
0	110	1024	1024	03FFh
0	101	2048	2048	07FFh
0	100	4096	4096	0FFFh
0	011-000	—	8192	1FFFh

Note: The maximum boot block size is half the user program memory size. All selections higher than the maximum are set to half size. For example, all BBSIZE = 000 – 100 produce a boot block size of 4 kW on a 8 kW device.

Note:

1. Bits are implemented as sticky bits. Once protection is enabled, it can only be reset through a Bulk Erase.

Offset	Name	Bit Pos.								
0x02	PCL	7:0	PCL[7:0]							
0x03	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x04	FSR0	7:0	FSRL[7:0]							
0x05		15:8	FSRH[7:0]							
0x06	FSR1	7:0	FSRL[7:0]							
0x07		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 ... 0x0FEC	Reserved									
0x0FED	STKPTR	7:0				STKPTR[4:0]				
0x0FEE	TOS	7:0	TOSL[7:0]							
0x0FEF		15:8	TOSH[7:0]							

7.8 Register Definitions: Memory and Status

7.8.9 PCLATH

Name: PCLATH
Address: 0x0A + n*0x80 [n=0..63]

Program Counter Latches.

Write Buffer for the upper 7 bits of the Program Counter

Bit	7	6	5	4	3	2	1	0
		PCLATH[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bits 6:0 – PCLATH[6:0] High PC Latch register
Holding register for Program Counter bits <6:0>

Related Links

7.3.2 Core Registers

7.8.11 TOS

Name: TOS
Address: 0x1FEE

Top-of-Stack Registers
Contents of the stack pointed to by the STKPTR register. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL.

Bit	15	14	13	12	11	10	9	8
	TOSH[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

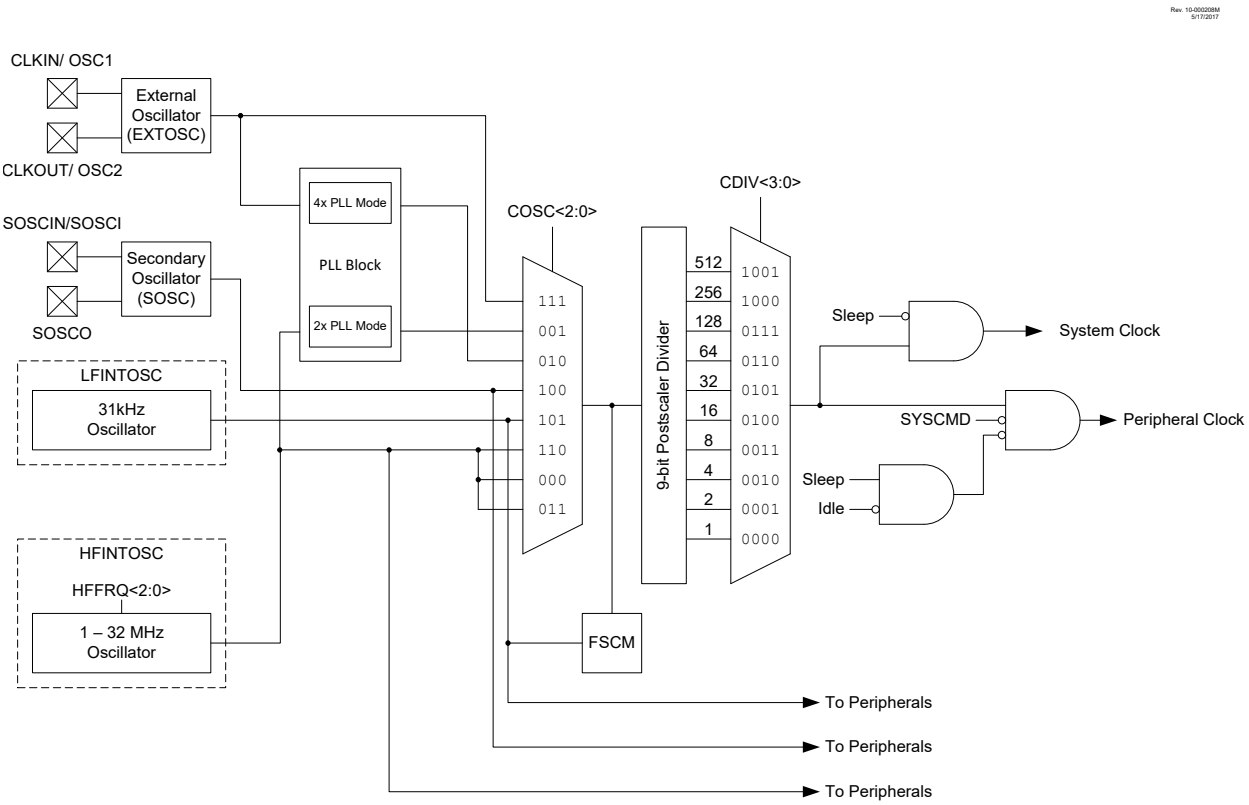
Bit	7	6	5	4	3	2	1	0
	TOSL[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 15:8 – TOSH[7:0] High Byte of the TOS Register
Bits <15:8> of the TOS

Bits 7:0 – TOSL[7:0] Low Byte TOS Register
Bits <7:0> of the TOS

Related Links
7.3.2 Core Registers

Figure 9-1. Simplified PIC® MCU Clock Source Block Diagram



Related Links

4.7.1 CONFIG1

9.2 Clock Source Types

Clock sources can be classified as external or internal.

External clock sources rely on external circuitry for the clock source to function. Examples are: oscillator modules (ECH, ECM, ECL mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes).

There is also a secondary oscillator block which is optimized for a 32.768 kHz external clock source, which can be used as an alternate clock source.

There are two internal oscillator blocks:

- HFINTOSC
- LFINTOSC

The HFINTOSC can produce clock frequencies from 1-32 MHz, and is responsible for generating the two MFINTOSC frequencies (500 kHz and 32 kHz) that can be used by some peripherals. The LFINTOSC generates a 31 kHz clock frequency.

There is a 4x PLL that can be used by the external oscillator. Additionally, there is a PLL that can be used by the HFINTOSC at certain frequencies.

Related Links

9.2.1.4 4x PLL

CDIV/NDIV	Clock Divider
0001	2
0000	1

Note:

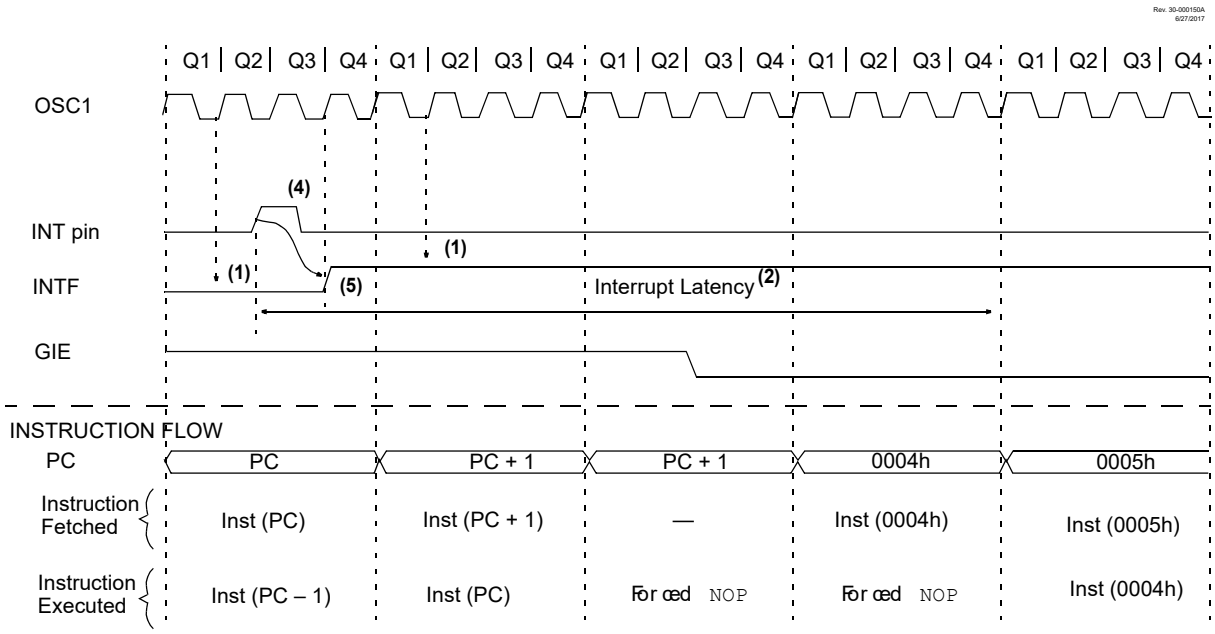
1. The POR value is the value present when user code execution begins.
2. The Reset value (n) is the same as the OSCCON1[NOSC/NDIV] bits.
3. EXTOSC configured by the CONFIG1[FEXTOSC] bits.
4. HFINTOSC frequency is configured with the FRQ bits of the OSCFRQ register

Related Links

4.7.1 CONFIG1
42.4.3 PLL Specifications

1. An interrupt may occur at any time during the interrupt window.
2. Since an interrupt may occur any time during the interrupt window, the actual latency can vary.

Figure 10-3. INT Pin Interrupt Timing



Note:

1. INTF flag is sampled here (every Q1).
2. Asynchronous interrupt latency = 3-5 T_{CY}. Synchronous latency = 3-4 T_{CY}, where T_{CY} = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.
3. For minimum width of INT pulse, refer to AC specifications in the *"Electrical Specifications"* section.
4. INTF may be set any time during the Q4-Q1 cycles.

10.3 Interrupts During Sleep

Interrupts can be used to wake from Sleep. To wake from Sleep, the peripheral must be able to operate without the system clock. The interrupt source must have the appropriate Interrupt Enable bit(s) set prior to entering Sleep.

On waking from Sleep, if the GIE bit is also set, the processor will branch to the interrupt vector. Otherwise, the processor will continue executing instructions after the `SLEEP` instruction. The instruction directly after the `SLEEP` instruction will always be executed before branching to the ISR.

Related Links

11. Power-Saving Operation Modes

10.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. Refer to Figure 10-3. This interrupt is enabled by setting the INTE bit of the PIE0 register. The INTEDG bit of the INTCON register determines on which edge the interrupt will occur. When the INTEDG bit is set, the rising edge will cause the interrupt. When the INTEDG bit is clear, the falling edge will cause the interrupt. The INTF

14.7.12 ANSELB

Name: ANSELB
Address: 0x1F43
Reset: 0x00

Analog Select Register

Bit	7	6	5	4	3	2	1	0
	ANSELB7	ANSELB6	ANSELB5	ANSELB4	ANSELB3	ANSELB2	ANSELB1	ANSELB0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bits 0, 1, 2, 3, 4, 5, 6, 7 – ANSELBn Analog Select on Pins RB<7:0>

Value	Description
1	Digital Input buffers are disabled
0	ST and TTL input buffers are enabled

16. (PMD) Peripheral Module Disable

This module provides the ability to selectively enable or disable a peripheral. Disabling a peripheral places it in its lowest possible power state. The user can disable unused modules to reduce the overall power consumption.

The PIC16(L)F18455/56 devices address this requirement by allowing peripheral modules to be selectively enabled or disabled. Disabling a peripheral places it in the lowest possible power mode.

Important: All modules are ON by default following any system Reset.

16.1 Disabling a Module

A peripheral can be disabled by setting the corresponding peripheral disable bit in the PMDx register. Disabling a module has the following effects:

- The module is held in Reset and does not function.
- All the SFRs pertaining to that peripheral become “unimplemented”
  - Writing is disabled
  - Reading returns 0x00
- Module outputs are disabled

Related Links

15.9.3 PPSLOCK

16.2 Enabling a Module

Clearing the corresponding module disable bit in the PMDx register, re-enables the module and the SFRs will reflect the Power-on Reset values.

Important: There should be no reads/writes to the module SFRs for at least two instruction cycles after it has been re-enabled.

16.3 System Clock Disable

Setting SYSCMD disables the system clock (F_{OSC}) distribution network to the peripherals. Not all peripherals make use of SYSCLK, so not all peripherals are affected. Refer to the specific peripheral description to see if it will be affected by this bit.

Related Links

16.5.1 PMD0

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{R_C}}\right) = V_{APPLIED}\left(1 - \frac{1}{\left(2^{n+1}\right) - 1}\right) \quad \text{;combining [1] and [2]}$$

Note: Where n = number of bits of the ADC.

Solving for T_C:

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ \ln(1/8191)$$

$$T_C = -28pF(1k\Omega + 7k\Omega + 10k\Omega)\ \ln(0.0001221)$$

$$T_C = 4.54us$$

Therefore:

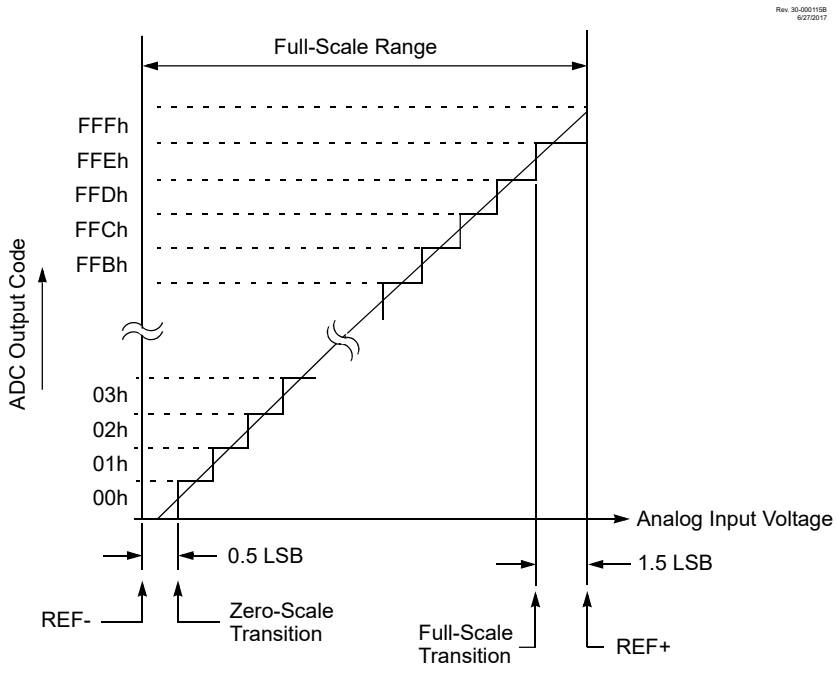
$$T_{ACQ} = 2us + 4.54us + \left[\left(50°C - 25°C\right)\left(0.05us/°C\right)\right]$$

$$T_{ACQ} = 7.79us$$

Note:

1. The reference voltage (V_{REF}) has no effect on the equation, since it cancels itself out.
2. The charge holding capacitor (C_{HOLD}) is not discharged after each conversion.
3. The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

Figure 20-5. ADC Transfer Function



Related Links

42.4.4 I/O and CLKOUT Timing Specifications

31.15.3 CWGxCLK

Name: CWGxCLK
Address: 0x60C,0x616,0x68C

CWGx Clock Input Selection Register

Bit	7	6	5	4	3	2	1	0
								CS
Access								R/W
Reset								0

Bit 0 – CS Clock Source
CWG Clock Source Selection Select bits

Value	Description
1	HFINTOSC (remains operating during Sleep)
0	F_{OSC}

mechanism to control the flow of data. When this detection is used on the SDA line, it is called arbitration. Arbitration ensures that there is only one master device communicating at any single time.

35.3.1 Register Definitions: I²C Mode

The MSSPx module has seven registers for I²C operation.

These are:

- MSSP Status register (SSPxSTAT)
- MSSP Control register 1 (SSPxCON1)
- MSSP Control register 2 (SSPxCON2)
- MSSP Control register 3 (SSPxCON3)
- Serial Receive/Transmit Buffer register (SSPxBUF)
- MSSP Address register (SSPxADD)
- I²C Slave Address Mask register (SSPxMSK)
- MSSP Shift register (SSPSR) – not directly accessible

SSPxCON1, SSPxCON2, SSPxCON3 and SSPxSTAT are the Control and STATUS registers in I²C mode operation. The SSPxCON1, SSPxCON2, and SSPxCON3 registers are readable and writable. The lower six bits of the SSPxSTAT are read-only. The upper two bits of the SSPxSTAT are read/write. SSPSR is the Shift register used for shifting data in or out. SSPxBUF is the buffer register to which data bytes are written to or read from. SSPxADD contains the slave device address when the MSSP is configured in I²C Slave mode. When the MSSP is configured in Master mode, the lower seven bits of SSPxADD act as the Baud Rate Generator reload value.

SSPxMSK holds the slave address mask value when the module is configured for 7-Bit Address Masking mode. While it is a separate register, it shares the same SFR address as SSPxADD; it is only accessible when the SSPM<3:0> bits are specifically set to permit access. In receive operations, SSPSR and SSPxBUF together, create a double-buffered receiver. When SSPSR receives a complete byte, it is transferred to SSPxBUF and the SSPxIF interrupt is set. During transmission, the SSPxBUF is not double-buffered. A write to SSPxBUF will write to both SSPxBUF and SSPSR.

35.4 I²C Mode Operation

All MSSP I²C communication is byte oriented and shifted out MSb first. Six SFR registers and two interrupt flags interface the module with the PIC® microcontroller and user software. Two pins, SDA and SCL, are exercised by the module to communicate with other external I²C devices.

35.4.1 Clock Stretching

When a slave device has not completed processing data, it can delay the transfer of more data through the process of clock stretching. An addressed slave device may hold the SCL clock line low after receiving or sending a bit, indicating that it is not yet ready to continue. The master that is communicating with the slave will attempt to raise the SCL line in order to transfer the next bit, but will detect that the clock line has not yet been released. Because the SCL connection is open-drain, the slave has the ability to hold that line low until it is ready to continue communicating.

Clock stretching allows receivers that cannot keep up with a transmitter to control the flow of incoming data.

35.9.1 SSPxSTAT

Name: SSPxSTAT
Address: 0x18F,0x199

MSSP Status Register

Bit	7	6	5	4	3	2	1	0
	SMP	CKE	$\overline{D/A}$	P	S	$\overline{R/W}$	UA	BF
Access	R/W	R/W	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0

Bit 7 – SMP Slew Rate Control bit

Value	Mode	Description
1	SPI Master	Input data is sampled at the end of data output time
0	SPI Master	Input data is sampled at the middle of data output time
0	SPI Slave	Keep this bit cleared in SPI Slave mode
1	I^2C	Slew rate control is disabled for Standard Speed mode (100 kHz and 1 MHz)
0	I^2C	Slew rate control is enabled for High-Speed mode (400 kHz)

Bit 6 – CKE

SPI: Clock select bit(4) I^2C: SMBus Select bit

Value	Mode	Description
1	SPI	Transmit occurs on the transition from active to Idle clock state
0	SPI	Transmit occurs on the transition from Idle to active clock state
1	I^2C	Enables SMBus-specific inputs
0	I^2C	Disables SMBus-specific inputs

Bit 5 – D/$\overline{A}$
Data/$\overline{Address}$ bit

Value	Mode	Description
x	SPI or I^2C Master	Reserved
1	I^2C Slave	Indicates that the last byte received or transmitted was data
0	I^2C Slave	Indicates that the last byte received or transmitted was address

Bit 4 – P
Stop bit(1)

Value	Mode	Description
x	SPI	Reserved
1	I^2C	Stop bit was detected last
0	I^2C	Stop bit was not detected last

Bit 3 – S
Start bit(1)

36.6.3 BAUDxCON

Name: BAUDxCON
Address: 0x11F,0xA1F

Baud Rate Control Register

Bit	7	6	5	4	3	2	1	0
	ABDOVF	RCIDL		SCKP	BRG16		WUE	ABDEN
Access	RO	RO		RW	RW		RW	RW
Reset	0	0		0	0		0	0

Bit 7 – ABDOVF Auto-Baud Detect Overflow bit

Value	Condition	Description
1	SYNC= 0	Auto-baud timer overflowed
0	SYNC= 0	Auto-baud timer did not overflow
X	SYNC= 1	Don't care

Bit 6 – RCIDL Receive Idle Flag bit

Value	Condition	Description
1	SYNC= 0	Receiver is Idle
0	SYNC= 0	Start bit has been received and the receiver is receiving
X	SYNC= 1	Don't care

Bit 4 – SCKP Synchronous Clock Polarity Select bit

Value	Condition	Description
1	SYNC= 0	Idle state for transmit (TX) is a low level (transmit data inverted)
0	SYNC= 0	Idle state for transmit (TX) is a high level (transmit data is non-inverted)
1	SYNC= 1	Data is clocked on rising edge of the clock
0	SYNC= 1	Data is clocked on falling edge of the clock

Bit 3 – BRG16 16-bit Baud Rate Generator Select bit

Value	Description
1	16-bit Baud Rate Generator is used
0	8-bit Baud Rate Generator is used

Bit 1 – WUE Wake-up Enable bit

Value	Condition	Description
1	SYNC= 0	Receiver is waiting for a falling edge. Upon falling edge no character will be received and flag RCxIF will be set. WUE will automatically clear after RCxIF is set.
0	SYNC= 0	Receiver is operating normally
X	SYNC= 1	Don't care

Bit 0 – ABDEN Auto-Baud Detect Enable bit

37.3.8 SMTxCPR

Name: SMTxCPR
Address: 0x48F,0x50F

SMT Captured Period Register

Bit	23	22	21	20	19	18	17	16
	CPRU[7:0]							
Access	RO	RO	RO	RO	RO	RO	RO	RO
Reset	x	x	x	x	x	x	x	x

Bit	15	14	13	12	11	10	9	8
	CPRH[7:0]							
Access	RO	RO	RO	RO	RO	RO	RO	RO
Reset	x	x	x	x	x	x	x	x

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

Bits 23:16 – CPRU[7:0] Upper byte of SMT capture period register
Reset States: POR/BOR = xxxxxxxx
All Other Resets = uuuuuuuu

Bits 15:8 – CPRH[7:0] High byte of SMT capture period register
Reset States: POR/BOR = xxxxxxxx
All Other Resets = uuuuuuuu

Bits 7:0 – CPRL[7:0] Lower byte of SMT capture period register
Reset States: POR/BOR = xxxxxxxx
All Other Resets = uuuuuuuu

Address	Name	Bit Pos.								
0x050C	SMT2TMR	7:0	TMRL[7:0]							
		15:8	TMRH[7:0]							
		23:16	TMRU[7:0]							
0x050F	SMT2CPR	7:0	CPRL[7:0]							
		15:8	CPRH[7:0]							
		23:16	CPRU[7:0]							
0x0512	SMT2CPW	7:0	CPWL[7:0]							
		15:8	CPWH[7:0]							
		23:16	CPWU[7:0]							
0x0515	SMT2PR	7:0	PRL[7:0]							
		15:8	PRH[7:0]							
		23:16	PRU[7:0]							
0x0518	SMT2CON0	7:0	EN		STP	WPOL	SPOL	CPOL	PS[1:0]	
0x0519	SMT2CON1	7:0	GO	REPEAT			MODE[3:0]			
0x051A	SMT2STAT	7:0	CPRUP	CPWUP		RST		TS	WS	AS
0x051B	SMT2CLK	7:0						CSEL[2:0]		
0x051C	SMT2SIG	7:0				SSEL[4:0]				
0x051D	SMT2WIN	7:0				WSEL[4:0]				
0x051E ... 0x057F	Reserved									
0x0580	INDF0	7:0	INDF0[7:0]							
0x0581	INDF1	7:0	INDF1[7:0]							
0x0582	PCL	7:0	PCL[7:0]							
0x0583	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x0584	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0586	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x0588	BSR	7:0			BSR[5:0]					
0x0589	WREG	7:0	WREG[7:0]							
0x058A	PCLATH	7:0		PCLATH[6:0]						
0x058B	INTCON	7:0	GIE	PEIE						INTEDG
0x058C	NCO1ACC	7:0	ACCL[7:0]							
		15:8	ACCH[7:0]							
		23:16					ACCU[3:0]			
0x058F	NCO1INC	7:0	INCL[7:0]							
		15:8	INCH[7:0]							
		23:16					INCU[3:0]			
0x0592	NCO1CON	7:0	EN		OUT	POL				PFM
0x0593	NCO1CLK	7:0	PWS[2:0]				CKS[3:0]			
0x0594 ... 0x059B	Reserved									
0x059C	TMR0L	7:0	TMR0L[7:0]							
0x059D	TMR0H	7:0	TMR0H[7:0]							

Address	Name	Bit Pos.								
0x1586	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1588	BSR	7:0			BSR[5:0]					
0x1589	WREG	7:0	WREG[7:0]							
0x158A	PCLATH	7:0		PCLATH[6:0]						
0x158B	INTCON	7:0	GIE	PEIE						INTEDG
0x158C ... 0x15FF	Reserved									
0x1600	INDF0	7:0	INDF0[7:0]							
0x1601	INDF1	7:0	INDF1[7:0]							
0x1602	PCL	7:0	PCL[7:0]							
0x1603	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x1604	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1606	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1608	BSR	7:0			BSR[5:0]					
0x1609	WREG	7:0	WREG[7:0]							
0x160A	PCLATH	7:0		PCLATH[6:0]						
0x160B	INTCON	7:0	GIE	PEIE						INTEDG
0x160C ... 0x167F	Reserved									
0x1680	INDF0	7:0	INDF0[7:0]							
0x1681	INDF1	7:0	INDF1[7:0]							
0x1682	PCL	7:0	PCL[7:0]							
0x1683	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x1684	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1686	FSR1	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1688	BSR	7:0			BSR[5:0]					
0x1689	WREG	7:0	WREG[7:0]							
0x168A	PCLATH	7:0		PCLATH[6:0]						
0x168B	INTCON	7:0	GIE	PEIE						INTEDG
0x168C ... 0x16FF	Reserved									
0x1700	INDF0	7:0	INDF0[7:0]							
0x1701	INDF1	7:0	INDF1[7:0]							
0x1702	PCL	7:0	PCL[7:0]							
0x1703	STATUS	7:0				$\overline{TO}$	$\overline{PD}$	Z	DC	C
0x1704	FSR0	7:0	FSRL[7:0]							
		15:8	FSRH[7:0]							
0x1706	FSR1	7:0	FSRL[7:0]							

ASRF	Arithmetic Right Shift
Status Affected:	C, Z
Description:	The contents of register ‘f’ are shifted one bit to the right through the Carry flag. The MSb remains unchanged. If ‘d’ is ‘0’, the result is placed in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’. **Register f → C**

BCF	Bit Clear f
Syntax:	[*label*] BCF f, b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	0 → f<b>
Status Affected:	None
Description:	Bit ‘b’ in register ‘f’ is cleared.

BRA	Relative Branch
Syntax:	[*label*] BRA label [*label*] BRA $+k
Operands:	-256 ≤ label - PC + ≤ 255 -256 ≤ k ≤ 255
Operation:	(PC) + 1 + k → PC
Status Affected:	None
Description:	Add the signed 9-bit literal ‘k’ to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a 2-cycle instruction. This branch has a limited range.

BRW	Relative Branch with W
Syntax:	[*label*] BRW
Operands:	None
Operation:	(PC) + (W) → PC

SUBWF	Subtract W from f
	C = 1, W ≤ f DC = 0, W[3:0] > f[3:0] DC = 1, W[3:0] ≤ f[3:0]

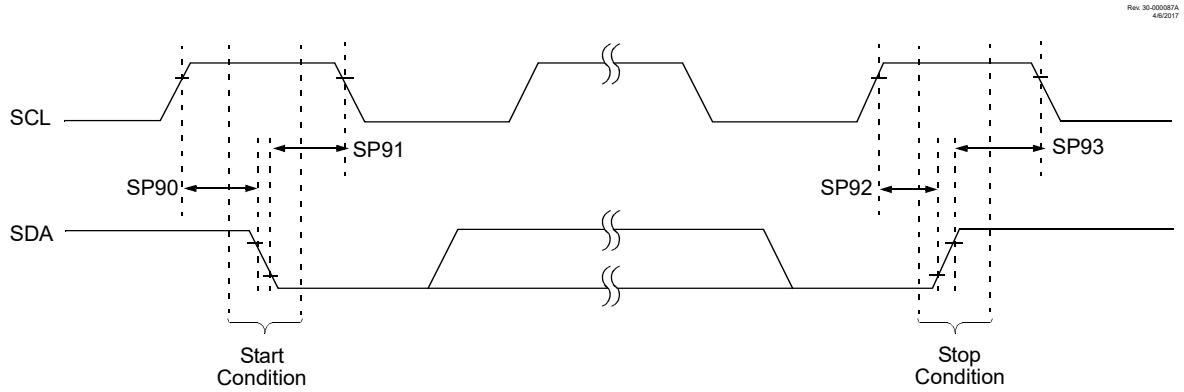
SUBFWB	Subtract W from f with Borrow
Syntax:	[*label*] SUBFWB f {,d}
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) – (f) – ($\overline{B}$) → dest
Status Affected:	C, DC, Z
Description:	Subtract W and the BORROW flag (CARRY) from register ‘f’ (2’s complement method). If ‘d’ is ‘0’, the result is stored in W. If ‘d’ is ‘1’, the result is stored back in register ‘f’.

SWAPF	Swap Nibbles in f
Syntax:	[*label*] SWAPF f, d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f<3:0>) → dest<7:4>, (f<7:4>) → dest<3:0>
Status Affected:	None
Description:	The upper and lower nibbles of register ‘f’ are exchanged. If ‘d’ is ‘0’, the result is placed in W. If ‘d’ is ‘1’, the result is placed in register ‘f’ (default).

TRIS	Load TRIS Register with W
Syntax:	[*label*] TRIS f
Operands:	5 ≤ f ≤ 7
Operation:	(W) → TRIS register ‘f’
Status Affected:	None
Description:	Move data from W register to TRIS register. When ‘f’ = 5, TRISA is loaded. When ‘f’ = 6, TRISB is loaded. When ‘f’ = 7, TRISC is loaded.

Standard Operating Conditions (unless otherwise stated)								
Param. No.	Sym.	Characteristic		Min.	Typ. †	Max.	Units	Conditions
			400 kHz mode	600	—	—		
* - These parameters are characterized but not tested.								

Figure 42-21. I²C Bus Start/Stop Bits Timing



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

42.4.20 I²C Bus Data Requirements

Table 42-26.

Standard Operating Conditions (unless otherwise stated)							
Param. No.	Sym.	Characteristic		Min.	Max.	Units	Conditions
SP100*	T_{HIGH}	Clock high time	100 kHz mode	4.0	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			SSP module	$1.5T_{CY}$	—		
SP101*	T_{LOW}	Clock low time	100 kHz mode	4.7	—	µs	Device must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz