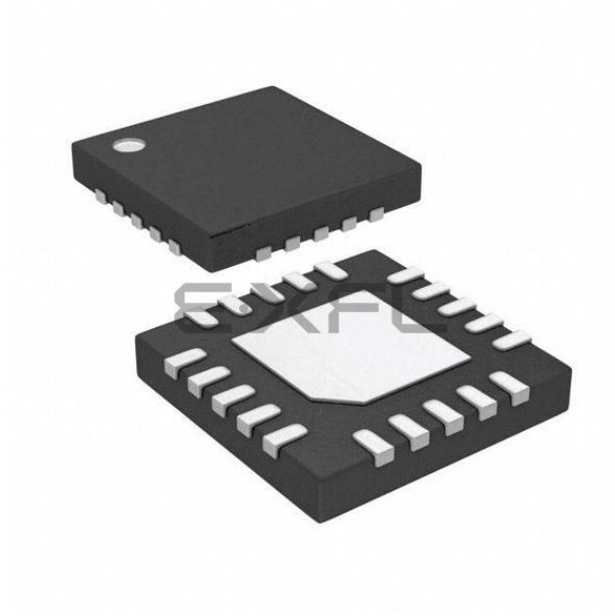


Welcome to E-XFL.COM

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	18
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)	2.3V ~ 5.5V
Data Converters	A/D 17x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-UFQFN Exposed Pad
Supplier Device Package	20-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18446t-i-gz

Address Range	Name of Region	Standard Device Information
8118h	FVRA1X	ADC FVR1 Output voltage for 1x setting (in mV)
8119h	FVRA2X	ADC FVR1 Output Voltage for 2x setting (in mV)
811Ah	FVRA4X(1)	ADC FVR1 Output Voltage for 4x setting (in mV)
811Bh	FVRC1X	Comparator FVR2 output voltage for 1x setting (in mV)
811Ch	FVRC2X	Comparator FVR2 output voltage for 2x setting (in mV)
811Dh	FVRC4X(1)	Comparator FVR2 output voltage for 4x setting (in mV)
811Eh-811Fh		Unassigned (2 Words)

Note:
1. Value not present on LF devices.

5.1 Microchip Unique identifier (MUI)

The PIC16(L)F184XX devices are individually encoded during final manufacturing with a Microchip Unique Identifier, or MUI. The MUI cannot be erased by a Bulk Erase command or any other user-accessible means. This feature allows for manufacturing traceability of Microchip Technology devices in applications where this is required. It may also be used by the application manufacturer for a number of functions that require unverified unique identification, such as:

- Tracking the device
- Unique serial number

The MUI consists of nine program words and one reserved program word. When taken together, these fields form a unique identifier. The MUI is stored in read-only locations, located between 8100h to 8109h in the DIA space. The above table lists the addresses of the identifier words.

Important: For applications that require verified unique identification, contact your Microchip Technology sales office to create a Serialized Quick Turn Programming option.

5.2 External Unique Identifier (EUI)

The EUI data is stored at locations 810Ah to 8111h in the program memory region. This region is an optional space for placing application specific information. The data is coded per customer requirements during manufacturing. The EUI cannot be erased by a Bulk Erase command.

Important: Data is stored in this address range on receiving a request from the customer. The customer may contact the local sales representative or Field Applications Engineer, and provide them the unique identifier information that is required to be stored in this region.

The CALLW instruction enables computed calls by combining PCLATH and W to form the destination address. A computed CALLW is accomplished by loading the W register with the desired address and executing CALLW. The PCL register is loaded with the value of W and PCH is loaded with PCLATH.

7.4.4 Branching

The branching instructions add an offset to the PC. This allows relocatable code and code that crosses page boundaries. There are two forms of branching, BRW and BRA. The PC will have incremented to fetch the next instruction in both cases. When using either branching instruction, a PCL memory boundary may be crossed.

If using BRW, load the W register with the desired unsigned address and execute BRW. The entire PC will be loaded with the address PC + 1 + W.

If using BRA, the entire PC will be loaded with PC + 1 + the signed value of the operand of the BRA instruction.

7.5 Stack

All devices have a 16-level by 15-bit wide hardware stack. The stack space is not part of either program or data space. The PC is PUSHed onto the stack when CALL or CALLW instructions are executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer if the STVREN configuration bit is programmed to '0'. This means that after the stack has been PUSHed sixteen times, the seventeenth PUSH overwrites the value that was stored from the first PUSH. The eighteenth PUSH overwrites the second PUSH (and so on). The STKOVF and STKUNF flag bits will be set on an Overflow/Underflow, regardless of whether the Reset is enabled.

Important: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, CALLW, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

7.5.1 Accessing the Stack

The stack is accessible through the TOSH, TOSL and STKPTR registers. STKPTR is the current value of the Stack Pointer. TOSH:TOSL register pair points to the TOP of the stack. Both registers are read/writable. TOS is split into TOSH and TOSL due to the 15-bit size of the PC. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL. STKPTR is five bits to allow detection of overflow and underflow.

Important: Care should be taken when modifying the STKPTR while interrupts are enabled.

During normal program operation, CALL, CALLW and interrupts will increment STKPTR while RETLW, RETURN, and RETFIE will decrement STKPTR. STKPTR can be monitored to obtain to value of stack

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

Core Registers

9.6.5 OSCEN

Name: OSCEN
Address: 0x891

Oscillator Manual Enable Register

Bit	7	6	5	4	3	2	1	0
	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN		
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

Bit 7 – EXTOEN External Oscillator Manual Request Enable bit

Value	Description
1	EXTOSC is explicitly enabled, operating as specified by CONFIG1[FEXTOSC]
0	EXTOSC is only enabled if requested by a peripheral

Bit 6 – HFOEN HFINTOSC Oscillator Manual Request Enable bit

Value	Description
1	HFINTOSC is explicitly enabled, operating as specified by OSCFRQ
0	HFINTOSC is only enabled if requested by a peripheral

Bit 5 – MFOEN MFINTOSC (500 kHz/31.25 kHz) Oscillator Manual Request Enable bit (Derived from HFINTOSC)

Value	Description
1	MFINTOSC is explicitly enabled
0	MFINTOSC is only enabled if requested by a peripheral

Bit 4 – LFOEN LFINTOSC (31 kHz) Oscillator Manual Request Enable bit

Value	Description
1	LFINTOSC is explicitly enabled
0	LFINTOSC is only enabled if requested by a peripheral

Bit 3 – SOSCEN Secondary Oscillator Manual Request Enable bit

Value	Description
1	Secondary Oscillator is explicitly enabled, operating as specified by SOSCPWR
0	Secondary Oscillator is only enabled if requested by a peripheral

Bit 2 – ADOEN ADC Oscillator Manual Request Enable bit

Value	Description
1	ADC oscillator is explicitly enabled
0	ADC oscillator is only enabled if requested by a peripheral

Related Links

CONFIG1

10.7.1 INTCON

Name: INTCON
Address: 0x00B

Interrupt Control Register

Bit	7	6	5	4	3	2	1	0
	GIE	PEIE						INTEDG
Access	R/W	R/W						R/W
Reset	0	0						1

Bit 7 – GIE Global Interrupt Enable bit

Value	Description
1	Enables all active interrupts
0	Disables all interrupts

Bit 6 – PEIE Peripheral Interrupt Enable bit

Value	Description
1	Enables all active peripheral interrupts
0	Disables all peripheral interrupts

Bit 0 – INTEDG External Interrupt Edge Select bit

Value	Description
1	Interrupt on rising edge of INT pin
0	Interrupt on falling edge of INT pin

Important: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

Value	Condition	Description
1	When NVMREG:NVMADR points to a Program Flash Memory location:	Initiates the operation indicated by table in *"WRERR Bit"* section.
0	When NVMREG:NVMADR points to a Program Flash Memory location:	NVM program/erase operation is complete and inactive.
1	When NVMREG:NVMADR points to a EEPROM location:	Initiates an erase/program cycle at the corresponding EEPROM location.
0	When NVMREG:NVMADR points to a EEPROM location:	NVM program/erase operation is complete and inactive.

Bit 0 – RD Read Control bit(7)

Value	Description
1	Initiates a read at address = NVMADR1, and loads data to NVMDAT Read takes one instruction cycle and the bit is cleared when the operation is complete. The bit can only be set (not cleared) in software.
0	NVM read operation is complete and inactive

Note:

1. Bit is undefined while WR = 1 (during the EEPROM write operation it may be '0' or '1').
2. Bit must be cleared by software; hardware will not clear this bit.
3. Bit may be written to '1' by the user in order to implement test sequences.
4. This bit can only be set by following the sequence described in the *"NVM Unlock Sequence"* section.
5. Operations are self-timed and the WR bit is cleared by hardware when complete.
6. Once a write operation is initiated, setting this bit to zero will have no effect.
7. Reading from EEPROM loads only NVMDATL.

Related Links

NVM Unlock Sequence

WRERR Bit

14. I/O Ports

14.1 PORT Availability

Table 14-1. PORT Availability Per Device

PORTs	PORT Description	PIC16(L)F18426	PIC16(L)F18446
PORTA	6-bit wide, bidirectional port.	•	•
PORTB	4-bit wide, bidirectional port.		•
PORTC	6/8-bit wide, bidirectional port.	•	•

14.2 I/O Ports Description

Each port has standard registers for its operation. These registers are:

- PORTx registers (reads the levels on the pins of the device)
- LATx registers (output latch)
- TRISx registers (data direction)
- ANSELx registers (analog select)
- WPUx registers (weak pull-up)
- INLVLx (input level control)
- SLRCONx registers (slew rate control)
- ODCONx registers (open-drain control)

Most port pins share functions with device peripherals, both analog and digital. In general, when a peripheral is enabled on a port pin, that pin cannot be used as a general purpose output; however, the pin can still be read.

The Data Latch (LATx registers) is useful for read-modify-write operations on the value that the I/O pins are driving.

A write operation to the LATx register has the same effect as a write to the corresponding PORTx register. A read of the LATx register reads of the values held in the I/O PORT latches, while a read of the PORTx register reads the actual I/O pin value.

Ports that support analog inputs have an associated ANSELx register. When an ANSELx bit is set, the digital input buffer associated with that bit is disabled.

Disabling the input buffer prevents analog signal levels on the pin between a logic high and low from causing excessive current in the logic input circuitry. A simplified model of a generic I/O port, without the interfaces to other peripherals, is shown in the following figure:

A pin can be configured to detect rising and falling edges simultaneously by setting both associated bits of the IOCxP and IOCxN registers, respectively.

17.3 Interrupt Flags

The bits located in the IOCxF registers are status flags that correspond to the interrupt-on-change pins of each port. If an expected edge is detected on an appropriately enabled pin, then the status flag for that pin will be set, and an interrupt will be generated if the IOCIE bit is set. The IOCIF bit of the PIR0 register reflects the status of all IOCxF bits.

Related Links

PIR0

17.3.1 Clearing Interrupt Flags

The individual status flags, (IOCxF register bits), can be cleared by resetting them to zero. If another edge is detected during this clearing operation, the associated status flag will be set at the end of the sequence, regardless of the value actually being written.

In order to ensure that no detected edge is lost while clearing flags, only AND operations masking out known changed bits should be performed. The following sequence is an example of what should be performed.

Clearing Interrupt Flags (PORTA Example)

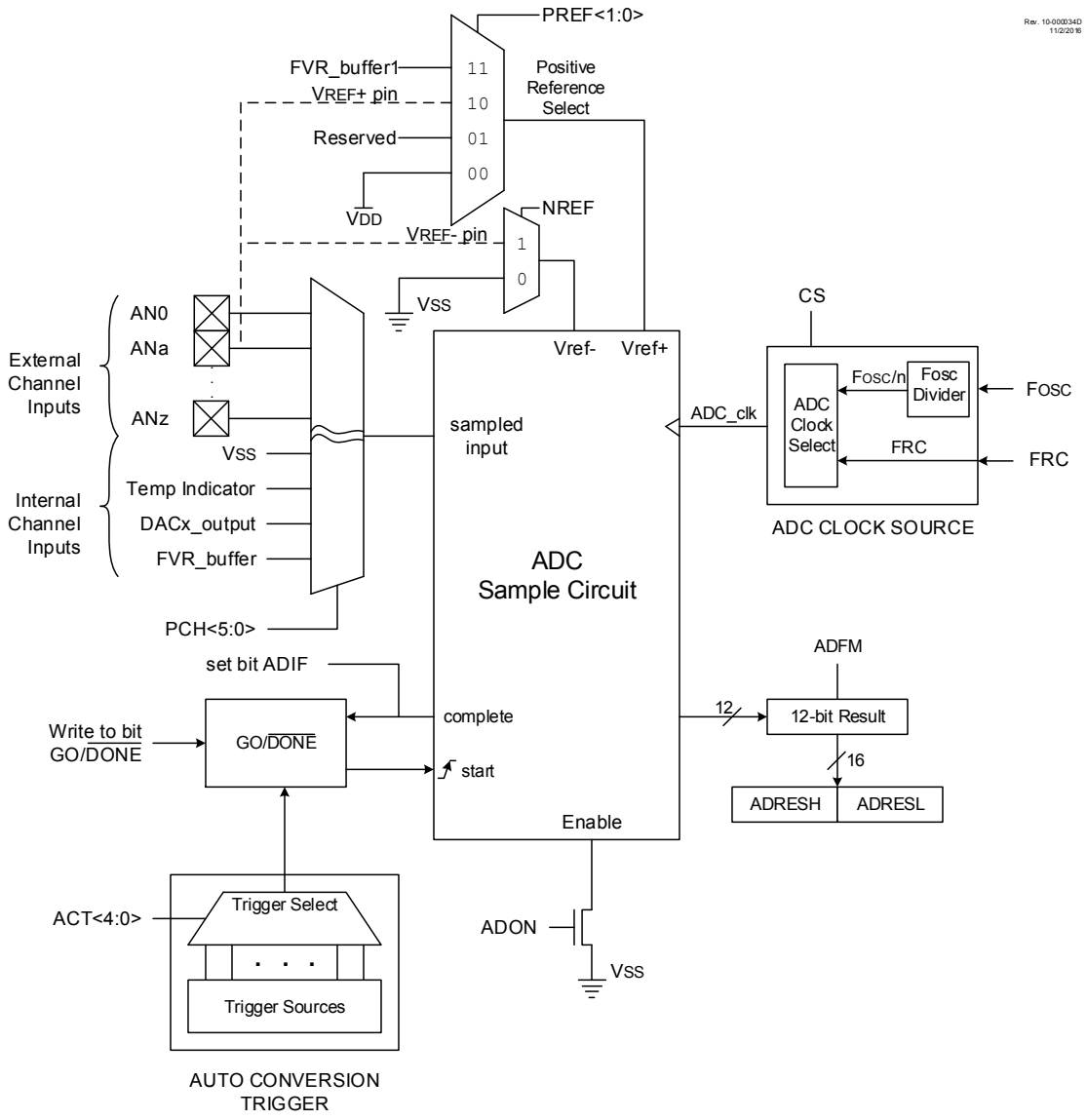
```
MOVLW    0xff
XORWF    IOCAF, W
ANDWF    IOCAF, F
```

17.4 Operation in Sleep

The interrupt-on-change interrupt sequence will wake the device from Sleep mode, if the IOCIE bit is set.

If an edge is detected while in Sleep mode, the IOCxF register will be updated prior to the first instruction executed out of Sleep.

Figure 20-1. ADC^2 Block Diagram



20.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port Configuration
- Channel Selection
- ADC Voltage Reference Selection
- ADC Conversion Clock Source
- Interrupt Control
- Result Formatting
- Conversion Trigger Selection
- ADC Acquisition Time

1. Configure Port:
   1.1. Disable pin output driver (Refer to the TRISx register)
   1.2. Configure pin as analog (Refer to the ANSELx register)
2. Configure the ADC module:
   2.1. Select ADC conversion clock
   2.2. Configure voltage reference
   2.3. Select ADC input channel (precharge+acquisition)
   2.4. Turn on ADC module
3. Configure ADC interrupt (optional):
   3.1. Clear ADC interrupt flag
   3.2. Enable ADC interrupt
   3.3. Enable peripheral interrupt (PIE bit)
   3.4. Enable global interrupt (GIE bit) (see Note 1 below)
4. If ADACQ = 0, software must wait the required acquisition time (see Note 2 below).
5. Start conversion by setting the GO bit.
6. Wait for ADC conversion to complete by one of the following:
   6.1. Polling the GO bit
   6.2. Polling the ADIF bit
   6.3. Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result.
8. Clear the ADC interrupt flag (required if interrupt is enabled).

Important:

1. The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.
2. Refer to the *"ADC Acquisition Requirements"* section.

ADC Conversion (assembly)

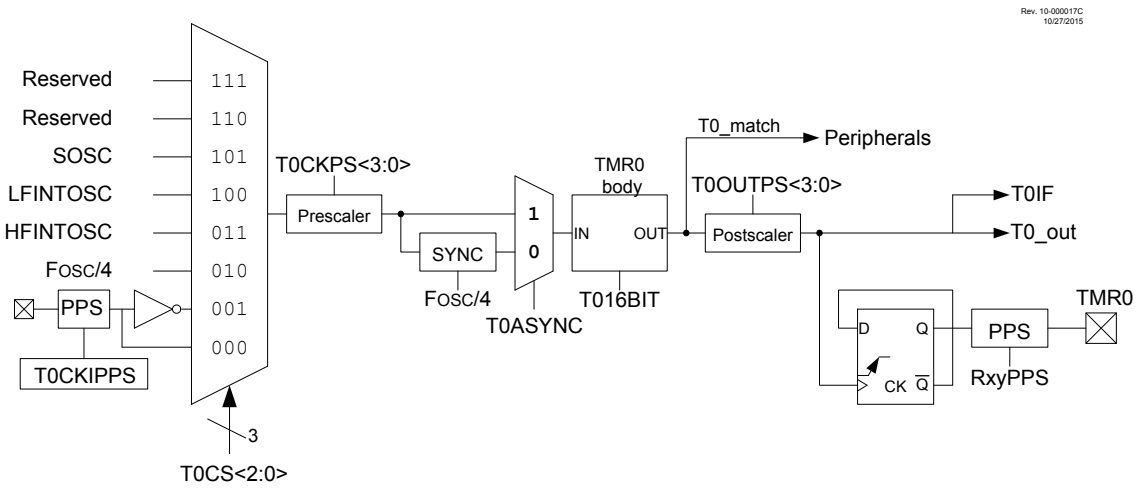
```
; This code block configures the ADC for polling, Vdd and Vss references,
; FRC oscillator, and AN0 input.
; Conversion start & polling for completion are included.
;
    BANKSEL ADCON1      ;
    movlw   B'11110000' ;Right justify, FRC oscillator
    movwf   ADCON1      ;Vdd and Vss Vref
    BANKSEL TRISA       ;
    bsf     TRISA,0     ;Set RA0 to input
    BANKSEL ANSEL       ;
    bsf     ANSEL,0     ;Set RA0 to analog
    BANKSEL ADCON0      ;
    movlw   B'00000001' ;Select channel AN0
    movwf   ADCON0      ;Turn ADC On
    call    SampleTime  ;Acquisiton delay
    bsf     ADCON0,ADGO ;Start conversion
    btfsc   ADCON0,ADGO ;Is conversion done?
    goto    $-1         ;No, test again
    BANKSEL ADRESH      ;
    movf    ADRESH,W    ;Read upper 2 bits
    movwf   RESULTHI    ;store in GPR space
    movf    ADRESL,W    ;Read lower 8 bits
    movwf   RESULTLO    ;Store in GPR space
```

25. Timer0 Module

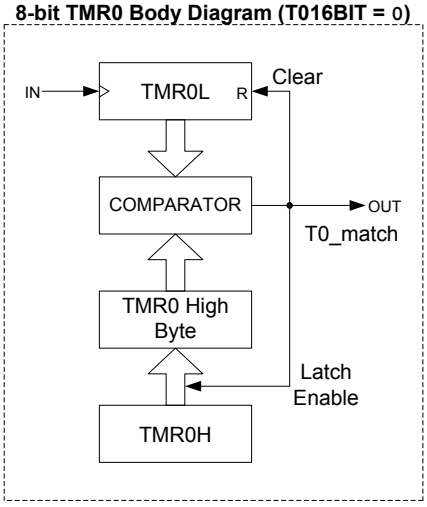
Timer0 module has the following features:

- 8-bit timer with programmable period
- 16-bit timer
- Selectable clock sources
- Synchronous and Asynchronous operation
- Programmable prescaler and postscaler
- Interrupt on match or overflow
- Output on I/O pin (via PPS) or to other peripherals
- Operation during Sleep

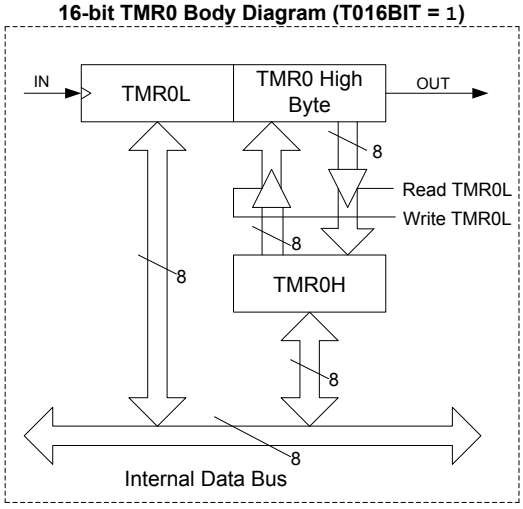
Figure 25-1. Block Diagram of Timer0



8-bit TMR0 Body Diagram (T016BIT = 0)



16-bit TMR0 Body Diagram (T016BIT = 1)



25.1 Timer0 Operation

Timer0 can operate as either an 8-bit or 16-bit timer. The mode is selected with the T016BIT bit.

25.1.1 8-bit Mode

In this mode Timer0 increments on the rising edge of the selected clock source. A prescaler on the clock input gives several prescale options (see prescaler control bits, T0CKPS).

In this mode as shown in the 8-bit TMR0 Body Diagram, a buffered version of TMR0H is maintained. This is compared with the value of TMR0L on each cycle of the selected clock source. When the two values match, the following events occur:

- TMR0L is reset
- The contents of TMR0H are copied to the TMR0H buffer for next comparison

25.1.2 16-Bit Mode

In this mode Timer0 increments on the rising edge of the selected clock source. A prescaler on the clock input gives several prescale options (see prescaler control bits, T0CKPS).

In this mode TMR0H:TMR0L form the 16-bit timer value. As shown in the 16-bit TMR0 Body Diagram, read and write of the TMR0H register are buffered. TMR0H register is updated with the contents of the high byte of Timer0 during a read of TMR0L register. Similarly, a write to the high byte of Timer0 takes place through the TMR0H buffer register. The high byte is updated with the contents of TMR0H register when a write occurs to TMR0L register. This allows all 16 bits of Timer0 to be read and written at the same time.

Timer 0 rolls over to 0x0000 on incrementing past 0xFFFF. This makes the timer free running. TMR0L/H registers cannot be reloaded in this mode once started.

25.2 Clock Selection

Timer0 has several options for clock source selections, option to operate synchronously/asynchronously and a programmable prescaler.

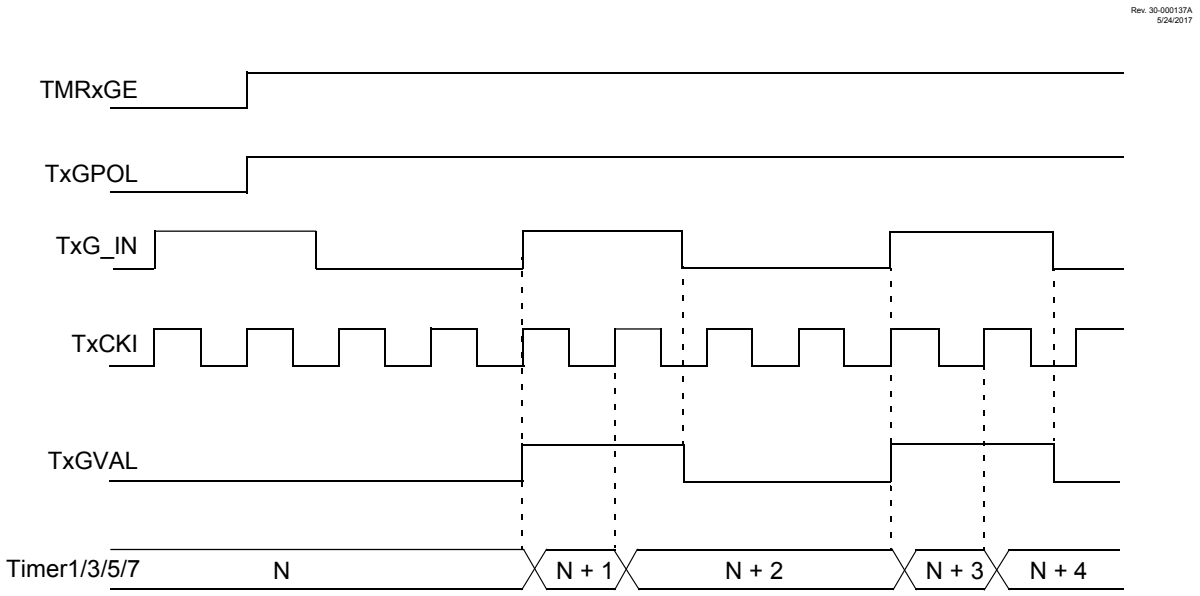
25.2.1 Clock Source Selection

The T0CS bits are used to select the clock source for Timer0. The possible clock sources are listed in the table below.

Table 25-1. Timer0 Clock Source Selections

T0CS	Clock Source
111	CLC1_out
110	SOSC
101	MFINTOSC(500 kHz)
100	LFINTOSC
011	HFINTOSC
010	$F_{OSC}/4$

Figure 26-4. Timer1 Gate Enable Mode



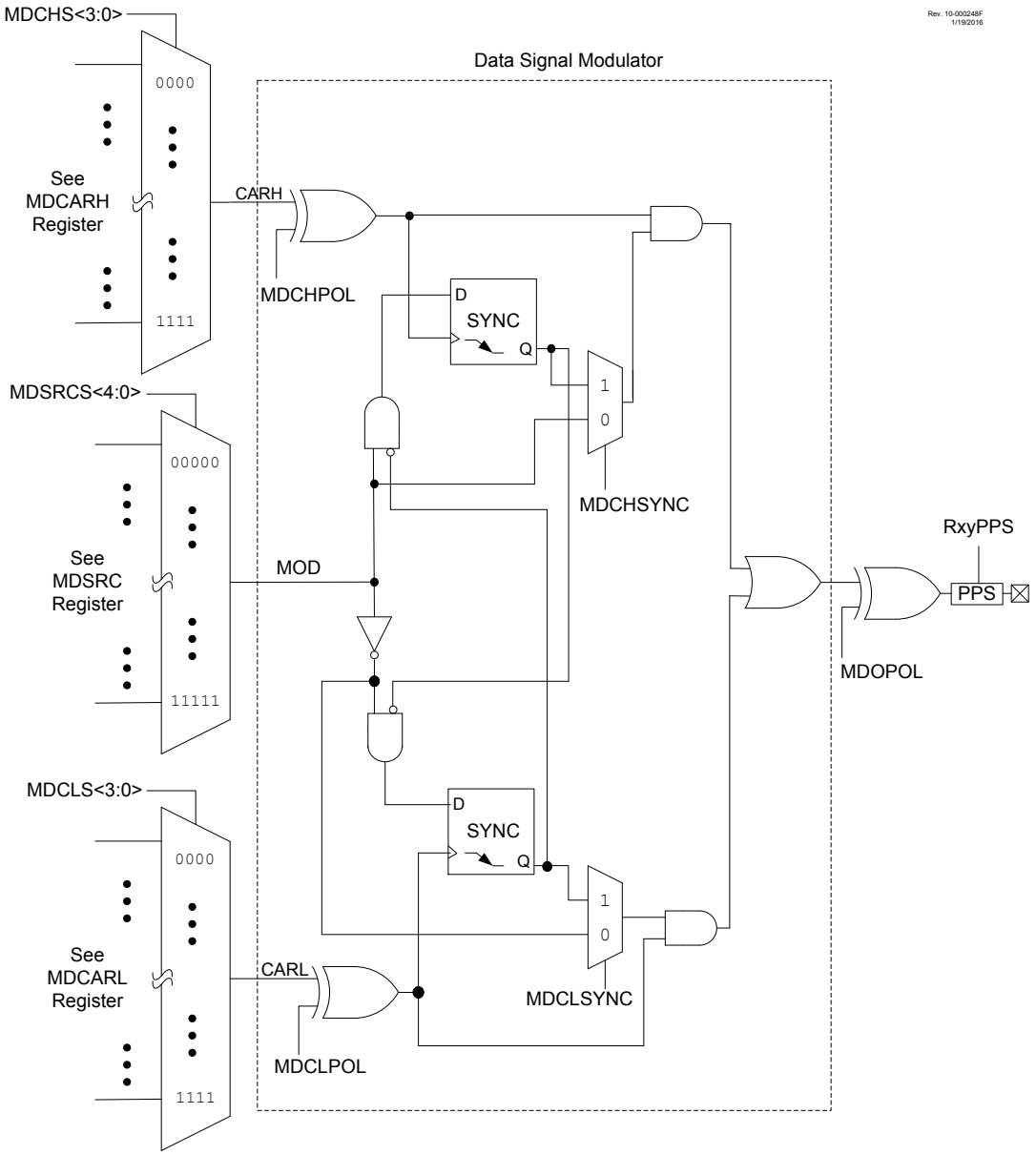
26.7.2 Timer1 Gate Source Selection

The gate source for Timer1 is selected using the GSS bits. The polarity selection for the gate source is controlled by the GPOL bit. The table below lists the gate source selections.

Table 26-4. Timer Gate Sources

GSS	Gate Source		
	Timer1	Timer3	Timer5
11111-11001	Reserved	Reserved	Reserved
10110	CLC4_out	CLC4_out	CLC4_out
10101	CLC3_out	CLC3_out	CLC3_out
10100	CLC2_out	CLC2_out	CLC2_out
10011	CLC1_out	CLC1_out	CLC1_out
10010	ZCD1_output	ZCD1_output	ZCD1_output
10001	C2OUT_sync	C2OUT_sync	C2OUT_sync
10000	C1OUT_sync	C1OUT_sync	C1OUT_sync
01111	NCO1_out	NCO1_out	NCO1_out
01110	PWM7_out	PWM7_out	PWM7_out
01101	PWM6_out	PWM6_out	PWM6_out
01100	CCP4_out	CCP4_out	CCP4_out
01011	CCP3_out	CCP3_out	CCP3_out

Figure 32-1. Simplified Block Diagram of the Data Signal Modulator



32.1 DSM Operation

The DSM module can be enabled by setting the EN bit in the MDCON0 register. Clearing the EN bit, disables the output of the module but retain the carrier and source signal selections. The module will resume operation when the EN bit is set again. The output of the DSM module can be rerouted to several pins using the RxyPPS register. When the EN bit is cleared the output pin is held low.

32.2 Modulator Signal Sources

The modulator signal can be supplied from the following sources selected with the SRCS bits:

33.8.4 CLCxSEL1

Name: CLCxSEL1
Address: 0x1E13,0x1E1D,0x1E27,0x1E31

Generic CLCx Data 1 Select Register

Bit	7	6	5	4	3	2	1	0
			D2S[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			x	x	x	x	x	x

Bits 5:0 – D2S[5:0]
CLCx Data2 Input Selection bits
Reset States: POR/BOR = xxxxxx
All Other Resets = uuuuuu

Value	Description
n	Refer to CLC Input Sources for input selections

TERM	Description
Slave	The device addressed by the master.
Multi-master	A bus with more than one device that can initiate data transfers.
Arbitration	Procedure to ensure that only one master at a time controls the bus. Winning arbitration ensures that the message is not corrupted.
Synchronization	Procedure to synchronize the clocks of two or more devices on the bus.
Idle	No master is controlling the bus, and both SDA and SCL lines are high.
Active	Any time one or more master devices are controlling the bus.
Addressed Slave	Slave device that has received a matching address and is actively being clocked by a master.
Matching Address	Address byte that is clocked into a slave that matches the value stored in SSPxADD.
Write Request	Slave receives a matching address with R/$\overline{W}$ bit clear, and is ready to clock in data.
Read Request	Master sends an address byte with the R/$\overline{W}$ bit set, indicating that it wishes to clock data out of the Slave. This data is the next and all following bytes until a Restart or Stop.
Clock Stretching	When a device on the bus hold SCL low to stall communication.
Bus Collision	Any time the SDA line is sampled low by the module while it is outputting and expected high state.

35.4.5 SDA and SCL Pins

Selection of any I^2C mode with the SSPEN bit set, forces the SCL and SDA pins to be open-drain. These pins should be set by the user to inputs by setting the appropriate TRIS bits.

Note:

1. Data is tied to output zero when an I^2C mode is enabled.
2. Any device pin can be selected for SDA and SCL functions with the PPS peripheral. These functions are bidirectional. The SDA input is selected with the SSPxDATPPS registers. The SCL input is selected with the SSPxCLKPPS registers. Outputs are selected with the RxyPPS registers. It is the user's responsibility to make the selections so that both the input and the output for each function is on the same pin.

35.4.6 SDA Hold Time

The hold time of the SDA pin is selected by the SDAHT bit. Hold time is the time SDA is held valid after the falling edge of SCL. Setting the SDAHT bit selects a longer 300 ns minimum hold time and may help on buses with large capacitance.

I^2C Bus Terms

35.4.7 Start Condition

The I^2C specification defines a Start condition as a transition of SDA from a high to a low state while SCL line is high. A Start condition is always generated by the master and signifies the transition of the bus from an Idle to an Active state. Figure 35-12 shows wave forms for Start and Stop conditions.

A bus collision can occur on a Start condition if the module samples the SDA line low before asserting it low. This does not conform to the I^2C Specification that states no bus collision can occur on a Start.

RETLW	Return literal to W			
Description:	The W register is loaded with the 8-bit literal ‘k’. The program counter is loaded from the top of the stack (the return address). This is a 2-cycle instruction.			
Words:	1			
Cycles:	2			

Example:

```
      CALL TABLE            ; W contains table
 ; offset value
 ; W now has
 ; table value
    :
 TABLE
 ADDWF PC             ; W = offset
 RETLW k1             ; Begin table
 RETLW k2             ;
 :
 :
 RETLW kn             ; End of table
```

Before Instruction
W = 07h

After Instruction

W = value of k8

RETURN	Return from Subroutine			
Syntax:	[*label*] RETURN			
Operands:	None			
Operation:	(TOS) → PC,			
Status Affected:	None			
Encoding:	0000	0000	0001	001s
Description:	Return from subroutine. The stack is POPped and the top of the stack (TOS) is loaded into the Program Counter. This is a 2-cycle instruction.			

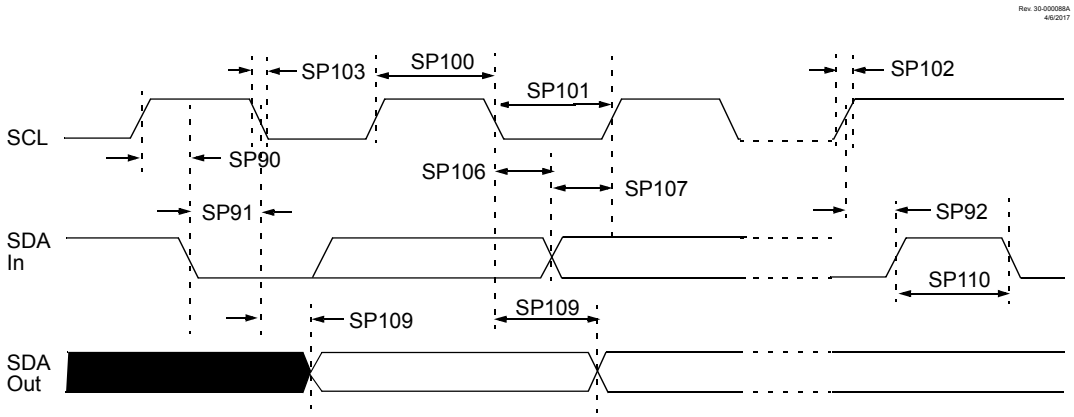
RLF	Rotate Left f through Carry
Syntax:	[*label*] RLF f, d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f<n>) → dest<n + 1>, (f<7>) → C,

Table 42-10. I/O and CLKOUT Timing Specifications

Standard Operating Conditions (unless otherwise stated)							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ. †**	**Max.**	**Units**	**Conditions**
IO1*	$T_{CLKOUTH}$	CLKOUT rising edge delay (rising edge F_{OSC} (Q1 cycle) to falling edge CLKOUT	—	—	70	ns	
IO2*	$T_{CLKOUTL}$	CLKOUT falling edge delay (rising edge F_{OSC} (Q3 cycle) to rising edge CLKOUT	—	—	72	ns	
IO3*	T_{IO_VALID}	Port output valid time (rising edge F_{OSC} (Q1 cycle) to port valid)	—	50	70	ns	
IO4*	T_{IO_SETUP}	Port input setup time (Setup time before rising edge F_{OSC} – Q2 cycle)	20	—	—	ns	
IO5*	T_{IO_HOLD}	Port input hold time (Hold time after rising edge F_{OSC} – Q2 cycle)	50	—	—	ns	
IO6*	T_{IOR_SLREN}	Port I/O rise time, slew rate enabled	—	25	—	ns	V_{DD}=3.0V
IO7*	T_{IOR_SLRDIS}	Port I/O rise time, slew rate disabled	—	5	—	ns	V_{DD}=3.0V
IO8*	T_{IOF_SLREN}	Port I/O fall time, slew rate enabled	—	25	—	ns	V_{DD}=3.0V
IO9*	T_{IOF_SLRDIS}	Port I/O fall time, slew rate disabled	—	5	—	ns	V_{DD}=3.0V
IO10*	T_{INT}	INT pin high or low time to trigger an interrupt	25	—	—	ns	
IO11*	T_{IOC}	Interrupt-on-Change minimum high or low time to trigger interrupt	25	—	—	ns	
* - These parameters are characterized but not tested.							

Standard Operating Conditions (unless otherwise stated)						
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
		if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line TR max. + $T_{SU:DAT}$ = 1000 + 250 = 1250 ns (according to the Standard mode I^2C bus specification), before the SCL line is released.				

Figure 42-22. I^2C Bus Data Timing



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

44. Packaging Information

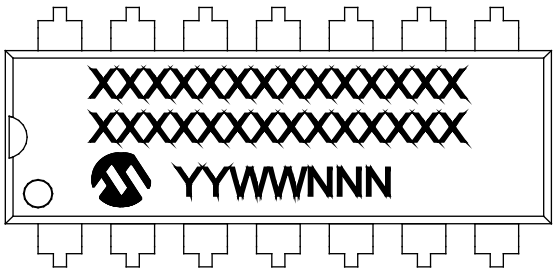
Package Marking Information

Rev. 30-009000A
5/17/2017

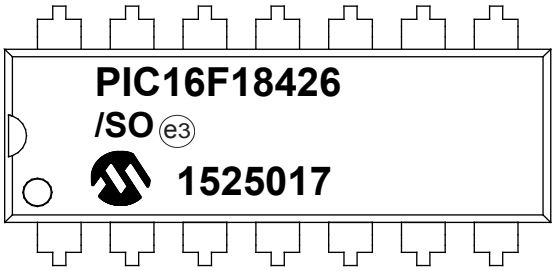
Legend:	XX...X	Customer-specific information or Microchip part number
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	b-free JEDEC ® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.
Note:	In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.	

Rev. 30-000014A
09/21/2017

14-Lead PDIP (300 mil)

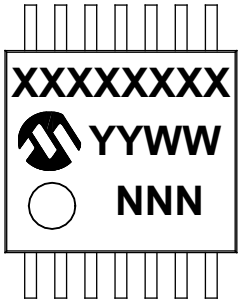


Example



Rev. 30-009014B
09/21/2017

14-Lead TSSOP (4.4 mm)



Example