



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	15
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 ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-UFQFN Exposed Pad
Supplier Device Package	28-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1847-i-mv

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

3.0 MEMORY ORGANIZATION

There are three types of memory in PIC16(L)F1847: Data Memory, Program Memory and Data EEPROM Memory⁽¹⁾.

- Program Memory
- Data Memory
 - Core Registers
 - Special Function Registers
 - General Purpose RAM
 - Common RAM
 - Device Memory Maps
 - Special Function Registers Summary
- Data EEPROM memory⁽¹⁾

Note 1: The Data EEPROM Memory and the method to access Flash memory through the EECON registers is described in Section 11.0 "Data EEPROM and Flash Program Memory Control". The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing

3.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing a $32K \times 14$ program memory space. Table 3-1 shows the memory sizes implemented for the PIC16(L)F1847 family. Accessing a location above these boundaries will cause a wrap-around within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see Figure 3-1).

TABLE 3-1:DEVICE SIZES AND ADDRESSES

Device	Program Memory Space (Words)	Last Program Memory Address		
PIC16(L)F1847	8,192	1FFFh		

5.2 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 (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.

Internal clock sources are contained internally within the oscillator module. The internal oscillator block has two internal oscillators and a dedicated Phase-Locked Loop (HFPLL) that are used to generate three internal system clock sources: the 16 MHz High-Frequency Internal Oscillator (HFINTOSC), 500 kHz (MFINTOSC) and the 31 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bits in the OSCCON register. See Section 5.3 "Clock Switching" for additional information.

5.2.1 EXTERNAL CLOCK SOURCES

An external clock source can be used as the device system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in the Configuration Words to select an external clock source that will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to:
 - Timer1 Oscillator during run-time, or
 - An external clock source determined by the value of the FOSC bits.

See **Section 5.3 "Clock Switching**" for more information.

5.2.1.1 EC Mode

The External Clock (EC) mode allows an externally generated logic level signal to be the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. Figure 5-2 shows the pin connections for EC mode.

EC mode has three power modes to select from through Configuration Word 1:

- High power, 4-32 MHz (FOSC = 111)
- Medium power, 0.5-4 MHz (FOSC = 110)
- Low power, 0-0.5 MHz (FOSC = 101)

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep. Because the PIC[®] MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.



: EXTERNAL CLOCK (EC) MODE OPERATION



5.2.1.2 LP, XT, HS Modes

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 5-3). The three modes select a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

LP Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP mode current consumption is the least of the three modes. This mode is designed to drive only 32.768 kHz tuning-fork type crystals (watch crystals).

XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.

HS Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HS mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

Figure 5-3 and Figure 5-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

5.2.2 INTERNAL CLOCK SOURCES

The device may be configured to use the internal oscillator block as the system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in Configuration Words to select the INTOSC clock source, which will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to the internal oscillator during run-time. See Section 5.3 "Clock Switching"for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT is available for general purpose I/O or CLKOUT.

The function of the <u>OSC2/CLKOUT</u> pin is determined by the state of the <u>CLKOUTEN</u> bit in Configuration Words.

The internal oscillator block has two independent oscillators and a dedicated Phase-Locked Loop, HFPLL that can produce one of three internal system clock sources.

- The HFINTOSC (High-Frequency Internal Oscillator) is factory calibrated and operates at 16 MHz. The HFINTOSC source is generated from the 500 kHz MFINTOSC source and the dedicated Phase-Locked Loop, HFPLL. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
- The MFINTOSC (Medium-Frequency Internal Oscillator) is factory calibrated and operates at 500 kHz. The frequency of the MFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 5-3).
- 3. The **LFINTOSC** (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

5.2.2.1 HFINTOSC

The High-Frequency Internal Oscillator (HFINTOSC) is a factory calibrated 16 MHz internal clock source. The frequency of the HFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

The output of the HFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of nine frequencies derived from the HFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See Section 5.2.2.7 "Internal Oscillator Clock Switch Timing" for more information.

The HFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'.

The High-Frequency Internal Oscillator Ready bit (HFIOFR) of the OSCSTAT register indicates when the HFINTOSC is running and can be utilized.

The High-Frequency Internal Oscillator Status Locked bit (HFIOFL) of the OSCSTAT register indicates when the HFINTOSC is running within 2% of its final value.

The High-Frequency Internal Oscillator Status Stable bit (HFIOFS) of the OSCSTAT register indicates when the HFINTOSC is running within 0.5% of its final value.

5.2.2.2 MFINTOSC

The Medium-Frequency Internal Oscillator (MFINTOSC) is a factory calibrated 500 kHz internal clock source. The frequency of the MFINTOSC can be altered via software using the OSCTUNE register (Register 5-3).

The output of the MFINTOSC connects to a postscaler and multiplexer (see Figure 5-1). One of nine frequencies derived from the MFINTOSC can be selected via software using the IRCF<3:0> bits of the OSCCON register. See Section 5.2.2.7 "Internal Oscillator Clock Switch Timing" for more information.

The MFINTOSC is enabled by:

- Configure the IRCF<3:0> bits of the OSCCON register for the desired HF frequency, and
- FOSC<2:0> = 100, or
- Set the System Clock Source (SCS) bits of the OSCCON register to '1x'

The Medium-Frequency Internal Oscillator Ready bit (MFIOFR) of the OSCSTAT register indicates when the MFINTOSC is running and can be utilized.

8.0 INTERRUPTS

The interrupt feature allows certain events to preempt normal program flow. Firmware is used to determine the source of the interrupt and act accordingly. Some interrupts can be configured to wake the MCU from Sleep mode.

This chapter contains the following information for Interrupts:

- Operation
- Interrupt Latency
- Interrupts During Sleep
- INT Pin
- · Automatic Context Saving

Many peripherals produce Interrupts. Refer to the corresponding chapters for details.

A block diagram of the interrupt logic is shown in Figure 8-1.

FIGURE 8-1: INTERRUPT LOGIC



8.3 Interrupts During Sleep

Some 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. Refer to the Section 9.0 "Power-Down Mode (Sleep)" for more details.

8.4 INT Pin

The INT pin can be used to generate an asynchronous edge-triggered interrupt. This interrupt is enabled by setting the INTE bit of the INTCON register. The INTEDG bit of the OPTION_REG 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 bit of the INTCON register will be set when a valid edge appears on the INT pin. If the GIE and INTE bits are also set, the processor will redirect program execution to the interrupt vector.

8.5 Automatic Context Saving

Upon entering an interrupt, the return PC address is saved on the stack. Additionally, the following registers are automatically saved in the Shadow registers:

- W register
- STATUS register (except for TO and PD)
- BSR register
- FSR registers
- PCLATH register

Upon exiting the Interrupt Service Routine, these registers are automatically restored. Any modifications to these registers during the ISR will be lost. If modifications to any of these registers are desired, the corresponding shadow register should be modified and the value will be restored when exiting the ISR. The shadow registers are available in Bank 31 and are readable and writable. Depending on the user's application, other registers may also need to be saved.

PIC16(L)F1847









21.7 **Timer1** Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- TMR1ON bit of the T1CON register
- TMR1IE bit of the PIE1 register
- · PEIE bit of the INTCON register
- · GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

The TMR1H:TMR1L register pair and the Note: TMR1IF bit should be cleared before enabling interrupts.

21.8 **Timer1 Operation During Sleep**

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- · PEIE bit of the INTCON register must be set
- T1SYNC bit of the T1CON register must be set
- TMR1CS bits of the T1CON register must be configured
- T1OSCEN bit of the T1CON register must be configured

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine.

Timer1 oscillator will continue to operate in Sleep regardless of the $\overline{\text{T1SYNC}}$ bit setting.

21.9 ECCP/CCP Capture/Compare Time Base

The CCP modules use the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPR1H:CCPR1L register pair on a configured event.

In Compare mode, an event is triggered when the value CCPR1H:CCPR1L register pair matches the value in the TMR1H:TMR1L register pair. This event can be a Special Event Trigger.

For more information, see Section 24.0 "Capture/Compare/PWM Modules".

21.10 ECCP/CCP Special Event Trigger

When any of the CCP's are configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt.

In this mode of operation, the CCPR1H:CCPR1L register pair becomes the period register for Timer1.

Timer1 should be synchronized and Fosc/4 should be selected as the clock source in order to utilize the Special Event Trigger. Asynchronous operation of Timer1 can cause a Special Event Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with a Special Event Trigger from the CCP, the write will take precedence.

For more information, see Section 16.2.5 "Special Event Trigger".



FIGURE 21-2: TIMER1 INCREMENTING EDGE

23.1 DSM Operation

The DSM module can be enabled by setting the MDEN bit in the MDCON register. Clearing the MDEN bit in the MDCON register, disables the DSM module by automatically switching the carrier high and carrier low signals to the Vss signal source. The modulator signal source is also switched to the MDBIT in the MDCON register. This not only assures that the DSM module is inactive, but that it is also consuming the least amount of current.

The values used to select the carrier high, carrier low, and modulator sources held by the Modulation Source, Modulation High Carrier, and Modulation Low Carrier control registers are not affected when the MDEN bit is cleared and the DSM module is disabled. The values inside these registers remain unchanged while the DSM is inactive. The sources for the carrier high, carrier low and modulator signals will once again be selected when the MDEN bit is set and the DSM module is again enabled and active.

The modulated output signal can be disabled without shutting down the DSM module. The DSM module will remain active and continue to mix signals, but the output value will not be sent to the MDOUT pin. During the time that the output is disabled, the MDOUT pin will remain low. The modulated output can be disabled by clearing the MDOE bit in the MDCON register.

23.2 Modulator Signal Sources

The modulator signal can be supplied from the following sources:

- CCP1 Signal
- CCP2 Signal
- CCP3 Signal
- CCP4 Signal
- MSSP1 SDO1 Signal (SPI Mode Only)
- MSSP2 SDO2 Signal (SPI Mode Only)
- Comparator C1 Signal
- · Comparator C2 Signal
- EUSART TX Signal
- External Signal on MDMIN1 pin
- MDBIT bit in the MDCON register

The modulator signal is selected by configuring the MDMS <3:0> bits in the MDSRC register.

23.3 Carrier Signal Sources

The carrier high signal and carrier low signal can be supplied from the following sources:

- CCP1 Signal
- CCP2 Signal
- CCP3 Signal
- CCP4 Signal
- Reference Clock Module Signal
- External Signal on MDCIN1 pin
- External Signal on MDCIN2 pin
- Vss

The carrier high signal is selected by configuring the MDCH <3:0> bits in the MDCARH register. The carrier low signal is selected by configuring the MDCL <3:0> bits in the MDCARL register.

23.4 Carrier Synchronization

During the time when the DSM switches between carrier high and carrier low signal sources, the carrier data in the modulated output signal can become truncated. To prevent this, the carrier signal can be synchronized to the modulator signal. When synchronization is enabled, the carrier pulse that is being mixed at the time of the transition is allowed to transition low before the DSM switches over to the next carrier source.

Synchronization is enabled separately for the carrier high and carrier low signal sources. Synchronization for the carrier high signal can be enabled by setting the MDCHSYNC bit in the MDCARH register. Synchronization for the carrier low signal can be enabled by setting the MDCLSYNC bit in the MDCARL register.

Figure 23-1 through Figure 23-5 show timing diagrams of using various synchronization methods.

25.2.2 SPI MODE OPERATION

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPxCON1<5:0> and SSPxSTAT<7:6>). These control bits allow the following to be specified:

- · Master mode (SCKx is the clock output)
- Slave mode (SCKx is the clock input)
- Clock Polarity (Idle state of SCKx)
- Data Input Sample Phase (middle or end of data output time)
- Clock Edge (output data on rising/falling edge of SCKx)
- Clock Rate (Master mode only)
- · Slave Select mode (Slave mode only)

To enable the serial port, SSPx Enable bit, SSPEN of the SSPxCON1 register, must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPxCONx registers and then set the SSPEN bit. This configures the SDIx, SDOx, SCKx and SSx pins as serial port pins. For the pins to behave as the serial port function, some must have their data direction bits (in the TRIS register) appropriately programmed as follows:

- · SDIx must have corresponding TRIS bit set
- · SDOx must have corresponding TRIS bit cleared
- SCKx (Master mode) must have corresponding TRIS bit cleared
- SCKx (Slave mode) must have corresponding TRIS bit set
- SSx must have corresponding TRIS bit set

Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value. The MSSPx consists of a transmit/receive shift register (SSPxSR) and a buffer register (SSPxBUF). The SSPxSR shifts the data in and out of the device, MSb first. The SSPxBUF holds the data that was written to the SSPxSR until the received data is ready. Once the eight bits of data have been received, that byte is moved to the SSPxBUF register. Then, the Buffer Full Detect bit, BF of the SSPxSTAT register, and the interrupt flag bit, SSPxIF, are set. This double-buffering of the received data (SSPxBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPxBUF register during transmission/reception of data will be ignored and the write collision detect bit WCOL of the SSPxCON1 register, will be set. User software must clear the WCOL bit to allow the following write(s) to the SSPxBUF register to complete successfully.

When the application software is expecting to receive valid data, the SSPxBUF should be read before the next byte of data to transfer is written to the SSPxBUF. The Buffer Full bit, BF of the SSPxSTAT register, indicates when SSPxBUF has been loaded with the received data (transmission is complete). When the SSPxBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the MSSPx interrupt is used to determine when the transmission/reception has completed. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur.

The SSPxSR 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.



FIGURE 25-5: SPI MASTER/SLAVE CONNECTION

25.6.7 I²C MASTER MODE RECEPTION

Master mode reception (Figure 25-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSPxCON2 register.

Note:	The MSSPx module must be in an Idle
	state before the RCEN bit is set or the
	RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCLx pin changes (high-to-low/low-to-high) and data is shifted into the SSPxSR. After the falling edge of the 8th clock, the receive enable flag is automatically cleared, the contents of the SSPxSR are loaded into the SSPxBUF, the BF flag bit is set, the SSPxIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCLx low. The MSSPx is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPxCON2 register.

25.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPxBUF from SSPxSR. It is cleared when the SSPxBUF register is read.

25.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when eight bits are received into the SSPxSR and the BF flag bit is already set from a previous reception.

25.6.7.3 WCOL Status Flag

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

25.6.7.4 Typical Receive 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. User writes SSPxBUF with the slave address to transmit and the R/W bit set.
- 5. Address is shifted out the SDAx pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
- The MSSPx module shifts in the ACK bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
- The MSSPx module generates an interrupt at the end of the 9th clock cycle by setting the SSPxIF bit.
- User sets the RCEN bit of the SSPxCON2 register and the Master clocks in a byte from the slave.
- 9. After the 8th falling edge of SCLx, SSPxIF and BF are set.
- 10. Master clears SSPxIF and reads the received byte from SSPxBUF, clears BF.
- 11. Master sets ACK value sent to slave in ACKDT bit of the SSPxCON2 register and initiates the ACK by setting the ACKEN bit.
- 12. Masters ACK is clocked out to the Slave and SSPxIF is set.
- 13. User clears SSPxIF.
- 14. Steps 8-13 are repeated for each received byte from the slave.
- 15. Master sends a not ACK or Stop to end communication.

26.2 **Clock Accuracy with Asynchronous Operation**

Г

The factory calibrates the internal oscillator block output (INTOSC). However, the INTOSC frequency may drift as VDD or temperature changes, and this directly affects the asynchronous baud rate. Two methods may be used to adjust the baud rate clock, but both require a reference clock source of some kind.

The first (preferred) method uses the OSCTUNE register to adjust the INTOSC output. Adjusting the value in the OSCTUNE register allows for fine resolution changes to the system clock source. See Section 5.2.2 "Internal Clock Sources" for more information.

The other method adjusts the value in the Baud Rate Generator. This can be done automatically with the Auto-Baud Detect feature (see Section 26.3.1 "Auto-Baud Detect"). There may not be fine enough resolution when adjusting the Baud Rate Generator to compensate for a gradual change in the peripheral clock frequency.

REGISTER 26-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN ⁽¹⁾	SYNC	SENDB	BRGH	TRMT	TX9D
bit 7		u					bit 0
Legend:							
R = Readable I	bit	W = Writable bi	t	U = Unimplem	ented bit, read as	ʻ0'	
u = Bit is uncha	anged	x = Bit is unkno	wn	-n/n = Value at	t POR and BOR/V	alue at all other	Resets
'1' = Bit is set		'0' = Bit is clear	ed				
bit 7	CSRC: Clock S	Source Select bit					
	Asynchronous	mode:					
	Don't care						
	Synchronous n	node:					
	1 = Master m	ode (clock gener	ated internally	from BRG)			
hit C	0 = Slave mo		ternal source)				
DIL O	1 = Selects 9						
	0 = Selects 8	-bit transmission					
bit 5	TXEN: Transm	it Enable bit ⁽¹⁾					
	1 = Transmit e	enabled					
	0 = Transmit o	disabled					
bit 4	SYNC: EUSAF	RT Mode Select b	it				
	1 = Synchrone	ous mode					
	0 = Asynchroi	nous mode					
bit 3	SENDB: Send	Break Character	bit				
	Asynchronous	mode:	,				
	1 = Send Syn	c Break on next t	ransmission (c	leared by hardwa	are upon completion	on)	
	Synchronous n	ak transmission o node:	Shipleted				
	Don't care	<u>node</u> .					
bit 2	BRGH: Hiah B	aud Rate Select I	oit				
	Asynchronous	mode:					
	1 = High spee	ed					
	0 = Low spee	d					
	Synchronous n	node:					
	Unused in this	mode					
bit 1	TRMT: Transm	it Shift Register S	tatus bit				
	1 = TSR emp	ty					
	0 = TSR full						
bit 0	TX9D: Ninth bi	t of Transmit Data	a				
	Can be addres	s/data bit or a par	ity bit.				
Note 1: SR	REN/CREN overrid	les TXEN in Sync	mode.				









Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	_	_	—	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	122
ANSELC	_	_	—	—	—	—	—	_	_
CPSCON0	CPSON	CPSRM	—	—	CPSRNG<1:0>		CPSOUT	TOXCS	323
CPSCON1	_	_	—	—		CPSCH<3:0>			324
INTCON	GIE	PEIE	TMR0IE	INTE	IOCE	TMR0IF	INTF	IOCF	88
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			175
T1CON	TMR1C	:S<1:0>	T1CKP	'S<1:0>	T1OSCEN	1OSCEN TISYNC — TMR1ON		191	
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	120
TRISC	—	-	—	_	—	—	—		-

TABLE 27-3: SUMMARY OF REGISTERS ASSOCIATED WITH CAPACITIVE SENSING

Legend: — = Unimplemented locations, read as '0'. Shaded cells are not used by the CPS module.

29.0 INSTRUCTION SET SUMMARY

Each PIC16 instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- · Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 29-3 lists the instructions recognized by the MPASM $^{\rm TM}$ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of four oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

29.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

TABLE 29-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
n	FSR or INDF number. (0-1)
mm	Pre-post increment-decrement mode selection

TABLE 29-2: ABBREVIATION DESCRIPTIONS

Field	Description					
PC	Program Counter					
TO	Time-out bit					
С	Carry bit					
DC	Digit carry bit					
Z	Zero bit					
PD	Power-down bit					

30.2 Standard Operating Conditions

The standard operating co	nditions for any device are defined as:	
Operating Voltage:	$V\text{DDMIN} \leq V\text{DD} \leq V\text{DDMAX}$	
Operating Temperature:	$TA_MIN \le TA \le TA_MAX$	
VDD — Operating Supply	^v Voltage ⁽¹⁾	
PIC16LF1847		
VDDMIN (Fo	osc ≤ 16 MHz)	
VDDMIN (Fo	$osc \leq 20 \text{ MHz}$)	+2.5V
VDDMAX		+3.6V
PIC16F1847		
VDDMIN (Fo	osc ≤ 16 MHz)	+2.3V
VDDMIN (Fo	osc ≤ 20 MHz)	
VDDMAX		+5.5V
TA — Operating Ambient	Temperature Range	
Industrial Temperatu	ure	
TA_MIN		40°C
Та_мах		+85°C
Extended Temperate	ure	
TA_MIN		40°C
Та_мах		+125°C

Note 1: See Parameter D001, DS Characteristics: Supply Voltage.

PIC16LF1847		Standard Operating Conditions (unless otherwise stated)							
PIC16F1847				_					
Param.	Device	Min	Tunt	Мах	Units		Conditions		
No.	Characteristics		турт	wax.		Vdd	Note		
D019		_	3.50	3.70	mA	3.0	Fosc = 32 MHz		
		_	4.20	4.30	mA	5.0	HFINTOSC (Note 3)		
D020		—	3.20	3.50	mA	3.0	Fosc = 32 MHz		
		_	3.70	3.90	mA	3.6	HS Oscillator (Note 4)		
D020		_	3.30	3.60	mA	3.0	Fosc = 32 MHz		
			3.70	4.10	mA	5.0	HS Oscillator (Note 4)		
D021		—	252	350	μA	1.8	Fosc = 4 MHz		
		_	480	580	μA	3.0	EXTRC (Note 5)		
D021			302	425	μA	1.8	Fosc = 4 MHz		
			440	680	μA	3.0	EXTRC (Note 5)		
			511	780	μA	5.0			
*	These parameters are	charact	terized hi	it not tes	sted				

TABLE 30-2: SUPPLY CURRENT (IDD)^(1,2) (CONTINUED)

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: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.

- 2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.
- **3:** 8 MHz internal oscillator with 4x PLL enabled.
- 4: 8 MHz crystal oscillator with 4x PLL enabled.
- 5: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in k Ω .

Standard Operating Conditions (unless otherwise stated)								
Param. No.	Symbol	Characteri	Characteristic		Typ†	Max.	Units	Conditions
CS01*	ISRC	Current Source	High	—	-8	—	μA	(Note 1)
			Medium	_	-1.5	_	μA	(Note 1)
			Low	—	-0.3	_	μA	(Note 1)
CS02*	Isnk	Current Sink	High	_	7.5	_	μA	(Note 1)
			Medium	_	1.5	_	μA	(Note 1)
			Low	_	0.25	_	μA	(Note 1)
CS03*	VСтн	Cap Threshold		_	0.8	_	V	
CS04*	VCTL	Cap Threshold		-	0.4	_	V	
CS05*	VCHYST	CAP HYSTERESIS	High	—	525	_	mV	
		(VCTH - VCTL)	Medium	_	375	_	mV	
			Low	_	300	_	mV	

TABLE 30-23: CAP SENSE OSCILLATOR SPECIFICATIONS

* 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: See Figure 31-62: Cap Sense Current Sink/Source Characteristics Fixed Voltage Reference (CPSRM = 0), High Current Range (CPSRNG = 11),

Figure 31-63: Cap Sense Current Sink/Source Characteristics Fixed Voltage Reference (CPSRM = 0), Medium Current Range (CPSRNG = 10) and

Figure 31-64: Cap Sense Current Sink/Source Characteristics Fixed Voltage Reference (CPSRM = 0), Low Current Range (CPSRNG = 01)







FIGURE 31-40: IPD, COMPARATOR, NORMAL-POWER MODE, CxSP = 1, PIC16F1847 ONLY



PIC16(L)F1847







28-Lead Plastic Quad Flat, No Lead Package (ML) - 6x6 mm Body [QFN] With 0.55 mm Terminal Length

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



	Units	MILLIMETERS				
Dimensior	Limits	MIN	NOM	MAX		
Number of Pins	N		28			
Pitch	е		0.65 BSC			
Overall Height	A	0.80	0.90	1.00		
Standoff	A1	0.00	0.02	0.05		
Terminal Thickness	A3	0.20 REF				
Overall Width	E	6.00 BSC				
Exposed Pad Width	E2	3.65	3.70	4.20		
Overall Length	D	6.00 BSC				
Exposed Pad Length	D2	3.65	3.70	4.20		
Terminal Width	b	0.23	0.30	0.35		
Terminal Length	L	0.50	0.55	0.70		
Terminal-to-Exposed Pad	K	0.20	-	-		

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.

2. Package is saw singulated

3. 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.

Microchip Technology Drawing C04-105C Sheet 2 of 2