



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	40MHz
Connectivity	CANbus, I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	96KB (48K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	3.25K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf4685-i-ml

Email: info@E-XFL.COM

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

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC18F2682
- PIC18F2685
- PIC18F4682
- PIC18F4685

This family of devices offers the advantages of all PIC18 microcontrollers – namely, high computational performance at an economical price – with the addition of high-endurance, Enhanced Flash program memory. In addition to these features, the PIC18F2682/2685/4682/4685 family introduces design enhancements that make these microcontrollers a logical choice for many high-performance, power sensitive applications.

1.1 New Core Features

1.1.1 nanoWatt TECHNOLOGY

All of the devices in the PIC18F2682/2685/4682/4685 family incorporate a range of features that can significantly reduce power consumption during operation. Key items include:

- Alternate Run Modes: By clocking the controller from the Timer1 source or the internal oscillator block, power consumption during code execution can be reduced by as much as 90%.
- Multiple Idle Modes: The controller can also run with its CPU core disabled but the peripherals still active. In these states, power consumption can be reduced even further, to as little as 4% of normal operation requirements.
- On-the-Fly Mode Switching: The powermanaged modes are invoked by user code during operation, allowing the user to incorporate power-saving ideas into their application's software design.
- Lower Consumption in Key Modules: The power requirements for both Timer1 and the Watchdog Timer have been reduced by up to 80%, with typical values of 1.1 and 2.1 μA, respectively.
- Extended Instruction Set: In addition to the standard 75 instructions of the PIC18 instruction set, PIC18F2682/2685/4682/4685 devices also provide an optional extension to the core CPU functionality. The added features include eight additional instructions that augment indirect and indexed addressing operations and the implementation of Indexed Literal Offset Addressing mode for many of the standard PIC18 instructions.

1.1.2 MULTIPLE OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC18F2682/2685/4682/4685 family offer ten different oscillator options, allowing users a wide range of choices in developing application hardware. These options include:

- Four Crystal modes, using crystals or ceramic resonators
- Two External Clock modes, offering the option of using two pins (oscillator input and a divide-by-4 clock output) or one pin (oscillator input, with the second pin reassigned as general I/O)
- Two External RC Oscillator modes with the same pin options as the External Clock modes
- An internal oscillator block which provides an 8 MHz clock (±2% accuracy) and an INTRC source (approximately 31 kHz, stable over temperature and VDD), as well as a range of 6 user selectable clock frequencies, between 125 kHz to 4 MHz, for a total of 8 clock frequencies. This option frees the two oscillator pins for use as additional general purpose I/O.
- A Phase Lock Loop (PLL) frequency multiplier, available to both the High-Speed Crystal and Internal Oscillator modes, which allows clock speeds of up to 40 MHz. Used with the internal oscillator, the PLL gives users a complete selection of clock speeds, from 31 kHz to 32 MHz – all without using an external crystal or clock circuit.

Besides its availability as a clock source, the internal oscillator block provides a stable reference source that gives the family additional features for robust operation:

- Fail-Safe Clock Monitor: This option constantly monitors the main clock source against a reference signal provided by the internal oscillator. If a clock failure occurs, the controller is switched to the internal oscillator block, allowing for continued low-speed operation or a safe application shutdown.
- Two-Speed Start-up: This option allows the internal oscillator to serve as the clock source from Power-on Reset, or wake-up from Sleep mode, until the primary clock source is available.

Pin Name	Pin Number PDIP, SOIC	Pin Type	Buffer Type	Description
				PORTA is a bidirectional I/O port.
RA0/AN0	2			
RA0		I/O	TTL	Digital I/O.
AN0		I	Analog	Analog input 0.
RA1/AN1	3			
RA1		I/O	TTL	Digital I/O.
AN1		I	Analog	Analog input 1.
RA2/AN2/VREF-	4			
RA2		I/O	TTL	Digital I/O.
AN2		I	Analog	Analog input 2.
VREF-		I	Analog	A/D reference voltage (low) input.
RA3/AN3/VREF+	5			
RA3		I/O	TTL	Digital I/O.
AN3		I	Analog	Analog input 3.
VREF+		I	Analog	A/D reference voltage (high) input.
RA4/T0CKI	6			
RA4		I/O	TTL	Digital I/O.
TOCKI		I	ST	Timer0 external clock input.
RA5/AN4/SS/HLVDIN	7			
RA5		I/O	TTL	Digital I/O.
AN4		I	Analog	Analog input 4.
SS		I	TTL	SPI slave select input.
HLVDIN		I	Analog	High/Low-Voltage Detect input.
RA6				See the OSC2/CLKO/RA6 pin.
RA7				See the OSC1/CLKI/RA7 pin.
Legend: TTL = TTL cor	npatible in	put		CMOS = CMOS compatible input or output
ST = Schmitt	Trigger in	out with	n CMOS le	evels I = Input
O = Output				P = Power

TABLE 1-2:	PIC18F2682/2685 PINOUT I/O DESCRIPTIONS ((CONTINUED)

ST = Schmitt O = Output = Schmitt Trigger input with CMOS levels

R/W-0	R/W-0 ⁽¹⁾	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0			
INTSRC	PLLEN ⁽¹⁾	—	TUN4	TUN3	TUN2	TUN1	TUN0			
bit 7	•					·	bit 0			
Legend:										
R = Readable	e bit	W = Writable	bit	U = Unimplei	mented bit, read	d as '0'				
-n = Value at	POR	'1' = Bit is set		'0' = Bit is cle	eared	x = Bit is unkr	nown			
bit 7	INTSRC: Inte	rnal Oscillator I	_ow-Frequen	cy Source Sele	ect bit					
	1 = 31.25 kHz	z device clock c	lerived from	8 MHz INTOSC	Source (divide	-by-256 enable	d)			
	0 = 31 kHz de	evice clock deri	ved directly f	rom INTRC inte	ernal oscillator					
bit 6	PLLEN: Freq	uency Multiplie	r PLL for INT	OSC Enable bi	it ⁽¹⁾					
	1 = PLL enab	led for INTOSC	C (4 MHz and	8 MHz only)						
	0 = PLL disat	bled								
bit 5	Unimplemen	ted: Read as ')							
bit 4-0	TUN4:TUN0: Frequency Tuning bits									
	01111 = Max	imum frequenc	y							
	•	•								
	•	•								
	00001 00000 - Contor frequency, Oscillator module is running at the calibrated frequency									
	11111	ter nequency.				a nequency.				
	•	•								
	•	•								
	10000 = Mini	mum frequency	/							

REGISTER 2-1: OSCTUNE: OSCILLATOR TUNING REGISTER

Note 1: Available only in certain oscillator configurations; otherwise, this bit is unavailable and reads as '0'. See text for details.

2.6.5.1 Compensating with the EUSART

An adjustment may be required when the EUSART begins to generate framing errors or receives data with errors while in Asynchronous mode. Framing errors indicate that the device clock frequency is too high. To adjust for this, decrement the value in OSCTUNE to reduce the clock frequency. On the other hand, errors in data may suggest that the clock speed is too low. To compensate, increment OSCTUNE to increase the clock frequency.

2.6.5.2 Compensating with the Timers

This technique compares device clock speed to some reference clock. Two timers may be used; one timer is clocked by the peripheral clock, while the other is clocked by a fixed reference source, such as the Timer1 oscillator.

Both timers are cleared, but the timer clocked by the reference generates interrupts. When an interrupt occurs, the internally clocked timer is read and both timers are cleared. If the internally clocked timer value is greater than expected, then the internal oscillator block is running too fast. To adjust for this, decrement the OSCTUNE register.

2.6.5.3 Compensating with the CCP1 Module in Capture Mode

The CCP1 module can use free running Timer1 (or Timer3), clocked by the internal oscillator block and an external event with a known period (i.e., AC power frequency). The time of the first event is captured in the CCPRxH:CCPRxL registers and is recorded for use later. When the second event causes a capture, the time of the first event is subtracted from the time of the second event. Since the period of the external event is known, the time difference between events can be calculated.

If the measured time is much greater than the calculated time, the internal oscillator block is running too fast. To compensate, decrement the OSCTUNE register. If the measured time is much less than the calculated time, the internal oscillator block is running too slow. To compensate, increment the OSCTUNE register.

^{© 2009} Microchip Technology Inc.

2.7.1 OSCILLATOR CONTROL REGISTER

The OSCCON register (Register 2-2) controls several aspects of the device clock's operation, both in full power operation and in power-managed modes.

The System Clock Select bits, SCS1:SCS0, select the clock source. The available clock sources are the primary clock (defined by the FOSC3:FOSC0 Configuration bits), the secondary clock (Timer1 oscillator) and the internal oscillator block. The clock source changes immediately after one or more of the bits is written to, following a brief clock transition interval. The SCS bits are cleared on all forms of Reset.

The Internal Oscillator Frequency Select bits, IRCF2:IRCF0, select the frequency output of the internal oscillator block to drive the device clock. The choices are the INTRC source, the INTOSC source (8 MHz) or one of the frequencies derived from the INTOSC postscaler (31 kHz to 4 MHz). If the internal oscillator block is supplying the device clock, changing the states of these bits will have an immediate change on the internal oscillator's output. On device Resets, the default output frequency of the internal oscillator block is set at 1 MHz.

When an output frequency of 31 kHz is selected (IRCF2:IRCF0 = 000), users may choose which internal oscillator acts as the source. This is done with the INTSRC bit in the OSCTUNE register (OSCTUNE<7>). Setting this bit selects INTOSC as a 31.25 kHz clock source by enabling the divide-by-256 output of the INTOSC postscaler. Clearing INTSRC selects INTRC (nominally 31 kHz) as the clock source.

This option allows users to select the tunable and more precise INTOSC as a clock source, while maintaining power savings with a very low clock speed. Regardless of the setting of INTSRC, INTRC always remains the clock source for features such as the Watchdog Timer and the Fail-Safe Clock Monitor.

The OSTS, IOFS and T1RUN bits indicate which clock source is currently providing the device clock. The OSTS bit indicates that the Oscillator Start-up Timer has timed out and the primary clock is providing the device clock in primary clock modes. The IOFS bit indicates when the internal oscillator block has stabilized and is providing the device clock in RC Clock modes. The T1RUN bit (T1CON<6>) indicates when the Timer1 oscillator is providing the device clock in secondary clock modes. In power-managed modes, only one of these three bits will be set at any time. If none of these bits are set, the INTRC is providing the clock or the internal oscillator block has just started and is not yet stable. The IDLEN bit determines if the device goes into Sleep mode or one of the Idle modes when the SLEEP instruction is executed.

The use of the flag and control bits in the OSCCON register is discussed in more detail in **Section 3.0** "Power-Managed Modes".

- Note 1: The Timer1 oscillator must be enabled to select the secondary clock source. The Timer1 oscillator is enabled by setting the T1OSCEN bit in the Timer1 Control register (T1CON<3>). If the Timer1 oscillator is not enabled, then any attempt to select a secondary clock source when executing a SLEEP instruction will be ignored.
 - 2: It is recommended that the Timer1 oscillator be operating and stable before executing the SLEEP instruction, or a very long delay may occur while the Timer1 oscillator starts.

2.7.2 OSCILLATOR TRANSITIONS

PIC18F2682/2685/4682/4685 devices contain circuitry to prevent clock "glitches" when switching between clock sources. A short pause in the device clock occurs during the clock switch. The length of this pause is the sum of two cycles of the old clock source and three to four cycles of the new clock source. This formula assumes that the new clock source is stable.

Clock transitions are discussed in greater detail in **Section 3.1.2 "Entering Power-Managed Modes"**.

4.5 Device Reset Timers

PIC18F2682/2685/4682/4685 devices incorporate three separate on-chip timers that help regulate the Power-on Reset process. Their main function is to ensure that the device clock is stable before code is executed. These timers are:

- Power-up Timer (PWRT)
- Oscillator Start-up Timer (OST)
- PLL Lock Time-out

4.5.1 POWER-UP TIMER (PWRT)

The Power-up Timer (PWRT) of PIC18F2682/2685/ 4682/4685 devices is an 11-bit counter which uses the INTRC source as the clock input. This yields an approximate time interval of 2048 x 32 μ s = 65.6 ms. While the PWRT is counting, the device is held in Reset.

The power-up time delay depends on the INTRC clock and will vary from chip-to-chip due to temperature and process variation. See DC parameter 33 for details.

The PWRT is enabled by clearing the PWRTEN Configuration bit.

4.5.2 OSCILLATOR START-UP TIMER (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over (parameter 33). This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP, HS and HSPLL modes and only on Power-on Reset or on exit from most power-managed modes.

4.5.3 PLL LOCK TIME-OUT

With the PLL enabled in its PLL mode, the time-out sequence following a Power-on Reset is slightly different from other oscillator modes. A separate timer is used to provide a fixed time-out that is sufficient for the PLL to lock to the main oscillator frequency. This PLL lock time-out (TPLL) is typically 2 ms and follows the oscillator start-up time-out.

4.5.4 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows:

- 1. After the POR pulse has cleared, PWRT time-out is invoked (if enabled).
- 2. Then, the OST is activated.

The total time-out will vary based on oscillator configuration and the status of the PWRT. Figure 4-3, Figure 4-4, Figure 4-5, Figure 4-6 and Figure 4-7 all depict time-out sequences on power-up, with the Power-up Timer enabled and the device operating in HS Oscillator mode. Figures 4-3 through 4-6 also apply to devices operating in XT or LP modes. For devices in RC mode and with the PWRT disabled, on the other hand, there will be no time-out at all.

Since the time-outs occur from the POR pulse, if MCLR is kept low long enough, all time-outs will expire. Bringing MCLR high will begin execution immediately (Figure 4-5). This is useful for testing purposes or to synchronize more than one PIC18FXXXX device operating in parallel.

Oscillator	Power-up ⁽²⁾ an	Exit From		
Configuration	PWRTEN = 0	PWRTEN = 1	Power-Managed Mode	
HSPLL	66 ms ⁽¹⁾ + 1024 Tosc + 2 ms ⁽²⁾	1024 Tosc + 2 ms ⁽²⁾	1024 Tosc + 2 ms ⁽²⁾	
HS, XT, LP	66 ms ⁽¹⁾ + 1024 Tosc	1024 Tosc	1024 Tosc	
EC, ECIO	66 ms ⁽¹⁾		—	
RC, RCIO	66 ms ⁽¹⁾	—	—	
INTIO1, INTIO2	66 ms ⁽¹⁾	_	—	

TABLE 4-2: TIME-OUT IN VARIOUS SITUATIONS

Note 1: 66 ms (65.5 ms) is the nominal Power-up Timer (PWRT) delay.

2: 2 ms is the nominal time required for the PLL to lock.

5.3 Data Memory Organization

Note: The operation of some aspects of data memory are changed when the PIC18 extended instruction set is enabled. See Section 5.6 "Data Memory and the Extended Instruction Set" for more information.

The data memory in PIC18 devices is implemented as static RAM. Each register in the data memory has a 12-bit address, allowing up to 4096 bytes of data memory. The memory space is divided into as many as 16 banks that contain 256 bytes each; PIC18F2682/2685/4682/4685 devices implement all 16 banks. Figure 5-5 shows the data memory organization for the PIC18F2682/2685/4682/4685 devices.

The data memory contains Special Function Registers (SFRs) and General Purpose Registers (GPRs). The SFRs are used for control and status of the controller and peripheral functions, while GPRs are used for data storage and scratchpad operations in the user's application. Any read of an unimplemented location will read as '0's.

The instruction set and architecture allow operations across all banks. The entire data memory may be accessed by Direct, Indirect or Indexed Addressing modes. Addressing modes are discussed later in this subsection.

To ensure that commonly used registers (SFRs and select GPRs) can be accessed in a single cycle, PIC18 devices implement an Access Bank. This is a 256-byte memory space that provides fast access to SFRs and the lower portion of GPR Bank 0 without using the BSR. **Section 5.3.2 "Access Bank"** provides a detailed description of the Access RAM.

5.3.1 BANK SELECT REGISTER (BSR)

Large areas of data memory require an efficient addressing scheme to make rapid access to any address possible. Ideally, this means that an entire address does not need to be provided for each read or write operation. For PIC18 devices, this is accomplished with a RAM banking scheme. This divides the memory space into 16 contiguous banks of 256 bytes. Depending on the instruction, each location can be addressed directly by its full 12-bit address, or an 8-bit low-order address and a 4-bit Bank Pointer.

Most instructions in the PIC18 instruction set make use of the Bank Pointer, known as the Bank Select Register (BSR). This SFR holds the 4 Most Significant bits of a location's address; the instruction itself includes the 8 Least Significant bits. Only the four lower bits of the BSR are implemented (BSR3:BSR0). The upper four bits are unused; they will always read '0' and cannot be written to. The BSR can be loaded directly by using the MOVLB instruction.

The value of the BSR indicates the bank in data memory. The 8 bits in the instruction show the location in the bank and can be thought of as an offset from the bank's lower boundary. The relationship between the BSR's value and the bank division in data memory is shown in Figure 5-6.

Since up to 16 registers may share the same low-order address, the user must always be careful to ensure that the proper bank is selected before performing a data read or write. For example, writing what should be program data to an 8-bit address of F9h while the BSR is 0Fh, will end up resetting the program counter.

While any bank can be selected, only those banks that are actually implemented can be read or written to. Writes to unimplemented banks are ignored, while reads from unimplemented banks will return '0's. Even so, the STATUS register will still be affected as if the operation was successful. The data memory map in Figure 5-5 indicates which banks are implemented.

In the core PIC18 instruction set, only the MOVFF instruction fully specifies the 12-bit address of the source and target registers. This instruction ignores the BSR completely when it executes. All other instructions include only the low-order address as an operand and must use either the BSR or the Access Bank to locate their target registers.

17.3.2 OPERATION

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPCON1<5:0> and SSPSTAT<7:6>). These control bits allow the following to be specified:

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

The MSSP module consists of a transmit/receive shift register (SSPSR) and a buffer register (SSPBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPBUF holds the data that was written to the SSPSR until the received data is ready. Once the 8 bits of data have been received, that byte is moved to the SSPBUF register. Then, the Buffer Full detect bit, BF (SSPSTAT<0>) and the interrupt flag bit, SSPIF, are set. This double-buffering of the received data (SSPBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPBUF register during transmission/reception of data will be ignored and the write collision detect bit, WCOL (SSPCON1<7>), will be set. User software must clear the WCOL bit so that it can be determined if the following write(s) to the SSPBUF register completed successfully.

When the application software is expecting to receive valid data, the SSPBUF should be read before the next byte of data to transfer is written to the SSPBUF. The Buffer Full bit, BF (SSPSTAT<0>), indicates when SSPBUF has been loaded with the received data (transmission is complete). When the SSPBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the MSSP interrupt is used to determine when the transmission/reception has completed. The SSPBUF must be read and/or written. 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. Example 17-1 shows the loading of the SSPBUF (SSPSR) for data transmission.

The SSPSR is not directly readable or writable and can only be accessed by addressing the SSPBUF register. Additionally, the MSSP Status register (SSPSTAT) indicates the various status conditions.

EXAMPLE 17-1: LOADING THE SSPBUF (SSPSR) REGISTER

LOOP	BTFSS	SSPSTAT, BF	;Has data been received (transmit complete)?
	BRA	LOOP	;No
	MOVF	SSPBUF, W	;WREG reg = contents of SSPBUF
	MOVWF	RXDATA	;Save in user RAM, if data is meaningful
	MOVF MOVWF	TXDATA, W SSPBUF	;W reg = contents of TXDATA ;New data to xmit



17.4.6 MASTER MODE

Master mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON1 and by setting the SSPEN bit. In Master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

Master mode of operation is supported by interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit is set or the bus is Idle, with both the S and P bits clear.

In Firmware Controlled Master mode, user code conducts all I^2C bus operations based on Start and Stop bit conditions.

Once Master mode is enabled, the user has six options.

- 1. Assert a Start condition on SDA and SCL.
- 2. Assert a Repeated Start condition on SDA and SCL.
- 3. Write to the SSPBUF register initiating transmission of data/address.
- 4. Configure the I²C port to receive data.
- 5. Generate an Acknowledge condition at the end of a received byte of data.
- 6. Generate a Stop condition on SDA and SCL.

Note: The MSSP module, when configured in I²C Master mode, does not allow queueing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPBUF register to initiate transmission before the Start condition is complete. In this case, the SSPBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

The following events will cause the MSSP Interrupt Flag bit, SSPIF, to be set (MSSP interrupt, if enabled):

- Start Condition
- Stop Condition
- Data Transfer Byte Transmitted/Received
- Acknowledge Transmit
- Repeated Start





TABLE 18-7: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	51
PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	54
PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	54
IPR1	PSPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	54
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	53
TXREG	EUSART T	ransmit Reg	ister						53
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	53
BAUDCON	ABDOVF	RCIDL	_	SCKP	BRG16	—	WUE	ABDEN	53
SPBRGH	EUSART E	Baud Rate G	enerator Re	gister High	Byte				53
SPBRG	EUSART E	Baud Rate G	enerator Re	gister Low	Byte				53

Legend: — = unimplemented, read as '0'. Shaded cells are not used for synchronous master transmission.

Note 1: Reserved in PIC18F2682/2685 devices; always maintain these bits clear.



22.5 Applications

In many applications, the ability to detect a drop below, or rise above a particular threshold is desirable. For example, the HLVD module could be periodically enabled to detect Universal Serial Bus (USB) attach or detach. This assumes the device is powered by a lower voltage source than the USB when detached. An attach would indicate a high-voltage detect from, for example, 3.3V to 5V (the voltage on USB) and vice versa for a detach. This feature could save a design a few extra components and an attach signal (input pin).

For general battery applications, Figure 22-4 shows a possible voltage curve. Over time, the device voltage decreases. When the device voltage reaches voltage VA, the HLVD logic generates an interrupt at time TA. The interrupt could cause the execution of an ISR, which would allow the application to perform "house-keeping tasks" and perform a controlled shutdown before the device voltage exits the valid operating range at TB. The HLVD, thus, would give the application a time window, represented by the difference between TA and TB, to safely exit.



NOTES:

$\label{eq:register23-11:} \textbf{TXBnDLC: TRANSMIT BUFFER n DATA LENGTH CODE $REGISTERS [0 \le n \le 2]$}$

U-0	R/W-x	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x					
_	TXRTR	_	_	DLC3	DLC2	DLC1	DLCO					
bit 7							bit 0					
Legend:												
R = Readab	le bit	W = Writable I	bit	U = Unimplei	mented bit, rea	d as '0'						
-n = Value a	t POR	'1' = Bit is set		'0' = Bit is cle	ared	x = Bit is unkr	nown					
bit 7	Unimplemer	nted: Read as ')'									
bit 6	TXRTR: Trar	nsmit Remote Fi	rame Transm	nission Request	t bit							
	1 = Transmit	1 = Transmitted message will have TXRTR bit set										
	0 = Transmit	ted message wi	ll have TXRT	R bit cleared								
bit 5-4	Unimplemer	nted: Read as '0)'									
bit 3-0	DLC3:DLC0: Data Length Code bits											
	1111 = Rese	1111 = Reserved										
	1110 = Rese	1110 = Reserved										
	1101 = Rese	1101 = Reserved										
	1100 = Rese	1100 = Reserved										
	1011 = Rese	1011 = Reserved										
	1010 = Rese	erved										
	1001 = Rese	1001 = Reserved										
	1000 = Data length = 8 bytes											
	0111 = Data	length = 7 bytes	5									
	0110 = Data	0110 = Data length = 6 bytes										
	0101 - Dala	0101 = Data length = 5 bytes										
	0011 = Data	length = 3 byte	5									
	0011 = Data	length = 2 byte	s									
	0001 = Data	length = 1 byte	5									
	0000 = Data	length = 0 bytes	S									

REGISTER 23-12: TXERRCNT: TRANSMIT ERROR COUNT REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read	as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **TEC7:TEC0:** Transmit Error Counter bits This register contains a value which is derived from the rate at which errors occur. When the error count overflows, the bus-off state occurs. When the bus has 128 occurrences of 11 consecutive recessive bits, the counter value is cleared.

23.3.4 LISTEN ONLY MODE

Listen Only mode provides a means for the PIC18F2682/2685/4682/4685 devices to receive all messages, including messages with errors. This mode can be used for bus monitor applications or for detecting the baud rate in 'hot plugging' situations. For Auto-Baud Detection, it is necessary that there are at least two other nodes which are communicating with each other. The baud rate can be detected empirically by testing different values until valid messages are received. The Listen Only mode is a silent mode, meaning no messages will be transmitted while in this state, including error flags or Acknowledge signals. The filters and masks can be used to allow only particular messages to be loaded into the receive registers or the filter masks can be set to all zeros to allow a message with any identifier to pass. The error counters are reset and deactivated in this state. The Listen Only mode is activated by setting the mode request bits in the CANCON register.

23.3.5 LOOPBACK MODE

This mode will allow internal transmission of messages from the transmit buffers to the receive buffers without actually transmitting messages on the CAN bus. This mode can be used in system development and testing. In this mode, the ACK bit is ignored and the device will allow incoming messages from itself, just as if they were coming from another node. The Loopback mode is a silent mode, meaning no messages will be transmitted while in this state, including error flags or Acknowledge signals. The TXCAN pin will revert to port I/O while the device is in this mode. The filters and masks can be used to allow only particular messages to be loaded into the receive registers. The masks can be set to all zeros to provide a mode that accepts all messages. The Loopback mode is activated by setting the mode request bits in the CANCON register.

23.3.6 ERROR RECOGNITION MODE

The module can be set to ignore all errors and receive any message. In functional Mode 0, the Error Recognition mode is activated by setting the RXM<1:0> bits in the RXBnCON registers to '11'. In this mode, the data which is in the message assembly buffer until the error time, is copied in the receive buffer and can be read via the CPU interface.

23.4 ECAN Module Functional Modes

In addition to CAN modes of operation, the ECAN module offers a total of 3 functional modes. Each of these modes are identified as Mode 0, Mode 1 and Mode 2.

23.4.1 MODE 0 – LEGACY MODE

Mode 0 is designed to be fully compatible with CAN modules used in PIC18CXX8 and PIC18FXX8 devices. This is the default mode of operation on all Reset conditions. As a result, module code written for the PIC18XX8 CAN module may be used on the ECAN module without any code changes.

The following is the list of resources available in Mode 0:

- Three transmit buffers: TXB0, TXB1 and TXB2
- Two receive buffers: RXB0 and RXB1
- Two acceptance masks, one for each receive buffer: RXM0, RXM1
- Six acceptance filters, 2 for RXB0 and 4 for RXB1: RXF0, RXF1, RXF2, RXF3, RXF4, RXF5

23.4.2 MODE 1 – ENHANCED LEGACY MODE

Mode 1 is similar to Mode 0, with the exception that more resources are available in Mode 1. There are 16 acceptance filter registers and two acceptance mask registers. Acceptance Filter 15 can be used as either an acceptance filter or an acceptance mask register. In addition to three transmit and two receive buffers, there are six more message buffers. One or more of these additional buffers can be programmed as transmit or receive buffers. These additional buffers can also be programmed to automatically handle RTR messages.

Fourteen of sixteen acceptance filter registers can be dynamically associated to any receive buffer and acceptance mask register. One can use this capability to associate more than one filter to any one buffer.

When a receive buffer is programmed to use standard identifier messages, part of the full acceptance filter register can be used as a data byte filter. The length of the data byte filter is programmable from 0 to 18 bits. This functionality simplifies implementation of high-level protocols, such as the DeviceNet[™] protocol.

The following is the list of resources available in Mode 1:

- Three transmit buffers: TXB0, TXB1 and TXB2
- Two receive buffers: RXB0 and RXB1
- Six buffers programmable as TX or RX: B0-B5
- Automatic RTR handling on B0-B5
- Sixteen dynamically assigned acceptance filters: RXF0-RXF15
- Two dedicated acceptance mask registers; RXF15 programmable as third mask: RXM0-RXM1, RXF15
- Programmable data filter on standard identifier messages: SDFLC

23.6.3 TRANSMIT PRIORITY

Transmit priority is a prioritization within the PIC18F2682/2685/4682/4685 devices of the pending transmittable messages. This is independent from and not related to any prioritization implicit in the message arbitration scheme built into the CAN protocol. Prior to sending the SOF, the priority of all buffers that are queued for transmission is compared. The transmit buffer with the highest priority will be sent first. If two buffers have the same priority setting, the buffer with the highest buffer number will be sent first. There are four levels of transmit priority. If TXP bits for a particular message buffer are set to '11', that buffer has the highest possible priority. If TXP bits for a particular message buffer are set to '00', that buffer has the lowest possible priority.



FIGURE 23-2: TRANSMIT BUFFERS

RLNCF	Rotate Left f (No Carry)	RRCF	Rotate Right f throu	igh Carry	
Syntax:	RLNCF f {,d {,a}}	Syntax:	RRCF f {,d {,a}}		
Operands:	$0 \le f \le 255$ $d \in [0,1]$ $a \in [0,1]$	Operands:	$0 \le f \le 255$ d $\in [0,1]$ a $\in [0,1]$		
Operation:	$(f < n >) \rightarrow dest < n + 1 >,$ $(f < 7 >) \rightarrow dest < 0 >$	Operation:	$(f < n >) \rightarrow dest < n - 1 >,$ $(f < 0 >) \rightarrow C,$ $(C) \rightarrow dest < 7 >$		
Status Affected:	N, Z	Status Affected:			
Encoding:	0100 01da ffff ffff	Encoding:			
Description:	The contents of register 'f' are rotated one bit to the left. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f' (default). If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the	Description:	The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in W If 'd' is '1', the result is placed back in register 'f' (default). If 'a' is '0', the Access Bank is selected If 'a' is '1', the BSR is used to select th GPR bank (default). If 'a' is '0' and the extended instruction set is enabled, this instruction operate in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode" for details.		
Words: Cycles: Q Cycle Activity: Q1 Decode	If a is '1', the BSR is used to select the GPR bank (default). If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode" for details. I teral Offset Mode" for details. Q2 Q3 Q4 Read Process Write to details	Words: Cycles: Q Cycle Activity:			
	register t Data destination	Decode	Q2 Q3 Read Process	Q4 Write to	
Example:	RLNCF REG, 1, 0	200000	register 'f' Data	destination	
Before Instruc REG After Instructio REG	ction = 1010 1011 on = 0101 0111	Example: Before Instruct REG C After Instructio REG W	RRCF REG, 0, tion = 1110 0110 = 0 on = 1110 0110 = 0111 0011	0	

TBL	BLRD Table Read							
Synta	ax:	TBLRD (*; *	+; *	-; +*)				
Oper	ands:	None						
Oper	ation:	if TBLRD *, (Prog Mem (TBLPTR)) \rightarrow TABLAT, TBLPTR – No Change; if TBLRD *+, (Prog Mem (TBLPTR)) \rightarrow TABLAT, (TBLPTR) + 1 \rightarrow TBLPTR; if TBLRD *-, (Prog Mem (TBLPTR)) \rightarrow TABLAT, (TBLPTR) – 1 \rightarrow TBLPTR; if TBLRD +*, (TBLPTR) + 1 \rightarrow TBLPTR, (Prog Mem (TBLPTR)) \rightarrow TABLAT						
Statu	s Affected:	None						
Enco	ding:	0000	0	000	00	00	10nn nn=0 * =1 *+ =2 *- =3 +*	
Desc	ription:	This instruction is used to read the contents of Program Memory (P.M.). To address the program memory, a pointer, called Table Pointer (TBLPTR), is used.						
		ach byte in the program memory. TBLP has a 2-Mbyte address range. TBLPTR[0] = 0: Least Significant Byte of Program Memory Word TBLPTR[0] = 1: Most Significant Byte of					. TBLPTR	
							nt Byte of ory Word It Byte of	
		Program Memory Word The TBLRD instruction can modify the value of TBLPTR as follows:						
		no change						
		post-increment						
		 post-decrement pre-increment 						
Word	ls.	1						
Cycle		2						
	vole Activity	<u> </u>						
QU	O1	. 02		0	3		Q4	
	Decode	No		No)		 No	
		operation		opera	tion	ор	eration	

No operation (Write

TÀBLAT)

TBLRD Table Read (Continued)

Example 1:	TBLRD	*+	;	
Before Instruction	on			
TABLAT			=	55h
TBLPTR	0040506		=	00A356h
MEMORY(00A356h)		=	34N
After Instruction				0.41-
			=	34N 00A357b
			-	00433711
Example 2:	TBLRD	+*	;	
Before Instruction	on			
TABLAT			=	0AAh
TBLPTR			=	01A357h
	01A35/N)		=	12n 34b
	01A33011)		-	3411
			_	0.45
			_	0411 014358b
			_	0170300

No

operation

No operation

(Read Program Memory) No

operation

XORWF	Exclusive	OR W with	n f			
Syntax:	XORWF	f {,d {,a}}				
Operands:	$0 \le f \le 255$ $d \in [0,1]$ $a \in [0,1]$					
Operation:	(W) .XOR.	(f) \rightarrow dest				
Status Affected:	N, Z					
Encoding:	0001	10da ff	ff ffff			
Description:	Exclusive C register 'f'. I in W. If 'd' is in the regis If 'a' is '0', t If 'a' is '1', t GPR bank If 'a' is '0' a set is enabl in Indexed mode wher Section 25 Bit-Oriente Literal Offe	Exclusive OR the contents of W with register 'f'. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in the register 'f' (default). If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default). If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See Section 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode" for details.				
Words:	1					
Cycles:	1					
Q Cycle Activity:						
Q1	Q2	Q3	Q4			
Decode	Read register 'f'	Process Data	Write to destination			
Example: XORWF REG, 1, 0						
Before Instruc REG W After Instructio REG W	tion = AFh = B5h on = 1Ah = B5h					



TABLE 27-17: EXAMPLE SPI SLAVE MODE REQUIREMENTS (CKE = 1)

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS} \downarrow$ to SCK \downarrow or SCK \uparrow Input		Тсү		ns	
71	TscH	SCK Input High Time	Continuous	1.25 Tcy + 30		ns	
71A			Single Byte	40		ns	(Note 1)
72	TscL	SCK Input Low Time	Continuous	1.25 Tcy + 30		ns	
72A			Single Byte	40		ns	(Note 1)
73A	Тв2в	Last Clock Edge of Byte 1 to the flrst	1.5 Tcy + 40		ns	(Note 2)	
74	TscH2DIL, TscL2DIL	Hold Time of SDI Data Input to SCI	100		ns		
75	TDOR	SDO Data Output Rise Time	PIC18FXXXX	—	25	ns	
			PIC18LFXXXX	—	45	ns	VDD = 2.0V
76	TDOF	SDO Data Output Fall Time		—	25	ns	
77	TssH2doZ	SS↑ to SDO Output High-Impedance		10	50	ns	
80	TscH2doV, TscL2doV	DV, SDO Data Output Valid after SCK	PIC18FXXXX	—	50	ns	
			PIC18LFXXXX	—	100	ns	VDD = 2.0V
82	TssL2doV	SDO Data Output Valid after $\overline{SS} \downarrow$ Edge	PIC18FXXXX	_	50	ns	
			PIC18LFXXXX	—	100	ns	VDD = 2.0V
83	TscH2ssH, TscL2ssH	SS ↑ after SCK Edge		1.5 Tcy + 40		ns	

Note 1: Requires the use of parameter 73A.

2: Only if parameter 71A and 72A are used.

FIGURE 27-18: MASTER SSP I²C[™] BUS START/STOP BITS TIMING WAVEFORMS



|--|

Param. No.	Symbol	Charac	teristic	Min	Max	Units	Conditions
90	TSU:STA	Start condition	100 kHz mode	2(Tosc)(BRG + 1)		ns	Only relevant for
		Setup Time	400 kHz mode	2(Tosc)(BRG + 1)			Repeated Start condition
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)			
91	THD:STA	Start Condition	100 kHz mode	2(Tosc)(BRG + 1)	—	ns	After this period, the first
		Hold Time	400 kHz mode	2(Tosc)(BRG + 1)	_		clock pulse is generated
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)	—		
92	Tsu:sto	Stop Condition	100 kHz mode	2(Tosc)(BRG + 1)	—	ns	
		Setup Time	400 kHz mode	2(Tosc)(BRG + 1)	_		
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)	—		
93	THD:STO	Stop Condition	100 kHz mode	2(Tosc)(BRG + 1)	—	ns	
		Hold Time	400 kHz mode	2(Tosc)(BRG + 1)	_		
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)			

Note 1: Maximum pin capacitance = 10 pF for all I^2C pins.

FIGURE 27-19: MASTER SSP I²C[™] BUS DATA TIMING

