



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	25MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	4KB (2K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VQFN Exposed Pad
Supplier Device Package	28-QFN (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f2221-e-ml

Email: info@E-XFL.COM

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

6.0 MEMORY ORGANIZATION

There are three types of memory in PIC18 Enhanced microcontroller devices:

- Program Memory
- Data RAM
- Data EEPROM

As Harvard architecture devices, the data and program memories use separate busses; this allows for concurrent access of the two memory spaces. The data EEPROM, for practical purposes, can be regarded as a peripheral device, since it is addressed and accessed through a set of control registers.

Additional detailed information on the operation of the Flash program memory is provided in Section 7.0 "Flash Program Memory". Data EEPROM is discussed separately in Section 8.0 "Data EEPROM Memory".

6.1 Program Memory Organization

PIC18 microcontrollers implement a 21-bit program counter, which is capable of addressing a 2-Mbyte program memory space. Accessing a location between the upper boundary of the physically implemented memory and the 2-Mbyte address will return all '0's (a NOP instruction).

The PIC18F2221 and PIC18F4221 each have 4 Kbytes of Flash memory and can store up to 2048 single-word instructions. The PIC18F2321 and PIC18F4321 each have 8 Kbytes of Flash memory and can store up to 4096 single-word instructions.

PIC18 devices have two interrupt vectors. The Reset vector address is at 0000h and the interrupt vector addresses are at 0008h and 0018h.

The program memory maps for PIC18F2221/4221 and PIC18F2321/4321 devices are shown in Figure 6-1.



FIGURE 6-1: PROGRAM MEMORY MAP AND STACK FOR PIC18F2221/2321/4221/4321 FAMILY

6.4.3.1 FSR Registers and the INDF Operand

At the core of indirect addressing are three sets of registers: FSR0, FSR1 and FSR2. Each represents a pair of 8-bit registers, FSRnH and FSRnL. The four upper bits of the FSRnH register are not used so each FSR pair holds a 12-bit value. This represents a value that can address the entire range of the data memory in a linear fashion. The FSR register pairs, then, serve as pointers to data memory locations.

Indirect addressing is accomplished with a set of Indirect File Operands, INDF0 through INDF2. These can be thought of as "virtual" registers: they are mapped in the SFR space but are not physically implemented. Reading or writing to a particular INDF register actually accesses its corresponding FSR register pair. A read from INDF1, for example, reads the data at the address indicated by FSR1H:FSR1L. Instructions that use the INDF registers as operands actually use the contents of their corresponding FSR as a pointer to the instruction's target. The INDF operand is just a convenient way of using the pointer.

Because indirect addressing uses a full 12-bit address, data RAM banking is not necessary. Thus, the current contents of the BSR and the Access RAM bit have no effect on determining the target address.

6.4.3.2 FSR Registers and POSTINC, POSTDEC, PREINC and PLUSW

In addition to the INDF operand, each FSR register pair also has four additional indirect operands. Like INDF, these are "virtual" registers that cannot be indirectly read or written to. Accessing these registers actually accesses the associated FSR register pair, but also performs a specific action on its stored value. They are:

- POSTDEC: accesses the FSR value, then automatically decrements it by 1 afterwards
- POSTINC: accesses the FSR value, then automatically increments it by 1 afterwards
- PREINC: increments the FSR value by 1, then uses it in the operation
- PLUSW: adds the signed value of the W register (range of -127 to 128) to that of the FSR and uses the new value in the operation.

In this context, accessing an INDF register uses the value in the FSR registers without changing them. Similarly, accessing a PLUSW register gives the FSR value offset by that in the W register; neither value is actually changed in the operation. Accessing the other virtual registers changes the value of the FSR registers.

Operations on the FSRs with POSTDEC, POSTINC and PREINC affect the entire register pair; that is, rollovers of the FSRnL register from FFh to 00h carry over to the FSRnH register. On the other hand, results of these operations do not change the value of any flags in the STATUS register (e.g., Z, N, OV, etc.).



FIGURE 6-7: INDIRECT ADDRESSING

PIC18F2221/2321/4221/4321 FAMILY

R 11-1:	I RISE RE	GISTER (4	40/44-PIN	DEVICES O	NLY)								
	R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1					
	IBF	OBF	IBOV	PSPMODE		TRISE2	TRISE1	TRISE0					
	bit 7							bit 0					
bit 7	IBF: Input	Buffer Full S	Status bit										
	1 = A word 0 = No wor	l has been r rd has been	eceived and received	d waiting to be	e read by the	e CPU							
bit 6	OBF: Output Buffer Full Status bit												
	 1 = The output buffer still holds a previously written word 0 = The output buffer has been read 												
bit 5	IBOV: Inpu	ut Buffer Ove	erflow Dete	ct bit (in Micro	processor n	node)							
	1 = A write 0 = No ove	 1 = A write occurred when a previously input word has not been read (must be cleared in software) 0 = No overflow occurred 											
bit 4	PSPMODE	: Parallel S	lave Port M	ode Select bit									
	1 = Paralle	el Slave Port	tmode										
	0 = Genera	al Purpose I	/O mode										
bit 3	Unimplem	ented: Rea	id as '0'										
bit 2	TRISE2: R	E2 Direction	n Control bi	t									
	1 = Input 0 = Output	:											
bit 1	TRISE1: R	E1 Direction	n Control bi	t									
	1 = Input												
	0 = Output												
bit 0	TRISE0: R	E0 Direction	n Control bi	t									
	1 = Input												
	0 = Output	:											
	Legend:												

REGISTER 11-1

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

12.3 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module. The prescaler is not directly readable or writable; its value is set by the PSA and T0PS<2:0> bits (T0CON<3:0>) which determine the prescaler assignment and prescale ratio.

Clearing the PSA bit assigns the prescaler to the Timer0 module. When it is assigned, prescale values from 1:2 through 1:256 in power-of-2 increments are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., CLRF TMR0, MOVWF TMR0, BSF TMR0, etc.) clear the prescaler count.

Note:	Writing to TMR0 when the prescaler is
	assigned to Timer0 will clear the prescaler
	count but will not change the prescaler
	assignment.

12.3.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control and can be changed "on-the-fly" during program execution.

12.4 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h in 8-bit mode, or from FFFFh to 0000h in 16-bit mode. This overflow sets the TMR0IF flag bit. The interrupt can be masked by clearing the TMR0IE bit (INTCON<5>). Before reenabling the interrupt, the TMR0IF bit must be cleared in software by the Interrupt Service Routine.

Since Timer0 is shut down in Sleep mode, the TMR0 interrupt cannot awaken the processor from Sleep.

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page			
TMR0L	Timer0 Reg	imer0 Register Low Byte										
TMR0H	Timer0 Reg	ister High By	/te						56			
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	55			
TOCON	TMR0ON	TMR0ON T08BIT T0CS T0SE PSA T0PS2 T0PS1 T0PS0							56			
TRISA	RA7 ⁽¹⁾	RA6 ⁽¹⁾	RA5	RA4	RA3	RA2	RA1	RA0	58			

 TABLE 12-1:
 REGISTERS ASSOCIATED WITH TIMER0

Legend: Shaded cells are not used by Timer0.

Note 1: PORTA<7:6> and their direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

14.2 Timer2 Interrupt

Timer2 can also generate an optional device interrupt. The Timer2 output signal (TMR2 to PR2 match) provides the input for the 4-bit output counter/postscaler. This counter generates the TMR2 match interrupt flag which is latched in TMR2IF (PIR1<1>). The interrupt is enabled by setting the TMR2 Match Interrupt Enable bit, TMR2IE (PIE1<1>).

A range of 16 postscale options (from 1:1 through 1:16 inclusive) can be selected with the postscaler control bits, T2OUTPS<3:0> (T2CON<6:3>).

14.3 Timer2 Output

The unscaled output of TMR2 is available primarily to the CCP modules, where it is used as a time base for operations in PWM mode.

Timer2 can be optionally used as the shift clock source for the MSSP module operating in SPI mode. Additional information is provided in Section 18.0 "Master Synchronous Serial Port (MSSP) Module".



FIGURE 14-1: TIMER2 BLOCK DIAGRAM

TABLE 14-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

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	55
PIR1	PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	58
PIE1	PSPIE ⁽¹⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	58
IPR1	PSPIP ⁽¹⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	58
TMR2	Timer2 Reg	gister							56
T2CON	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0	56
PR2	Timer2 Per	iod Register							56

Legend: — = unimplemented, read as '0'. Shaded cells are not used by the Timer2 module.

Note 1: These bits are unimplemented on 28-pin devices and read as '0'.

15.2 Timer3 16-Bit Read/Write Mode

Timer3 can be configured for 16-bit reads and writes (see Figure 15-2). When the RD16 control bit (T3CON<7>) is set, the address for TMR3H is mapped to a buffer register for the high byte of Timer3. A read from TMR3L will load the contents of the high byte of Timer3 into the Timer3 High Byte Buffer register. This provides the user with the ability to accurately read all 16 bits of Timer1 without having to determine whether a read of the high byte, followed by a read of the low byte, has become invalid due to a rollover between reads.

A write to the high byte of Timer3 must also take place through the TMR3H Buffer register. The Timer3 high byte is updated with the contents of TMR3H when a write occurs to TMR3L. This allows a user to write all 16 bits to both the high and low bytes of Timer3 at once.

The high byte of Timer3 is not directly readable or writable in this mode. All reads and writes must take place through the Timer3 High Byte Buffer register.

Writes to TMR3H do not clear the Timer3 prescaler. The prescaler is only cleared on writes to TMR3L.

15.3 Using the Timer1 Oscillator as the Timer3 Clock Source

The Timer1 internal oscillator may be used as the clock source for Timer3. The Timer1 oscillator is enabled by setting the T1OSCEN (T1CON<3>) bit. To use it as the Timer3 clock source, the TMR3CS bit must also be set. As previously noted, this also configures Timer3 to increment on every rising edge of the oscillator source.

The Timer1 oscillator is described in Section 13.0 "Timer1 Module".

15.4 Timer3 Interrupt

The TMR3 register pair (TMR3H:TMR3L) increments from 0000h to FFFFh and overflows to 0000h. The Timer3 interrupt, if enabled, is generated on overflow and is latched in interrupt flag bit, TMR3IF (PIR2<1>). This interrupt can be enabled or disabled by setting or clearing the Timer3 Interrupt Enable bit, TMR3IE (PIE2<1>).

15.5 Resetting Timer3 Using the CCP Special Event Trigger

If either of the CCP modules is configured to use Timer3 and to generate a Special Event Trigger in Compare mode (CCP1M<3:0> or CCP2M<3:0> = 1011), this signal will reset Timer3. It will also start an A/D conversion if the A/D module is enabled (see **Section 16.3.4** "**Special Event Trigger**" for more information).

The module must be configured as either a timer or synchronous counter to take advantage of this feature. When used this way, the CCPR2H:CCPR2L register pair effectively becomes a period register for Timer3.

If Timer3 is running in Asynchronous Counter mode, the Reset operation may not work.

In the event that a write to Timer3 coincides with a Special Event Trigger from a CCP module, the write will take precedence.

Note: The Special Event Triggers from the CCP2 module will not set the TMR3IF interrupt flag bit (PIR2<1>).

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	55
PIR2	OSCFIF	CMIF	_	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF	58
PIE2	OSCFIE	CMIE		EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE	58
IPR2	OSCFIP	CMIP	_	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP	58
TMR3L	Timer3 Reg	gister Low By	yte						57
TMR3H	Timer3 Reg	gister High B	yte						57
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T10SCEN	T1SYNC	TMR1CS	TMR10N	56
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC	TMR3CS	TMR3ON	57

TABLE 15-1: REGISTERS ASSOCIATED WITH TIMER3 AS A TIMER/COUNTER

Legend: — = unimplemented, read as '0'. Shaded cells are not used by the Timer3 module.

PIC18F2221/2321/4221/4321 FAMILY

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	55			
RCON	IPEN	SBOREN ⁽¹⁾	_	RI	TO	PD	POR	BOR	54			
PIR1	PSPIF ⁽²⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	58			
PIE1	PSPIE ⁽²⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	58			
IPR1	PSPIP ⁽²⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	58			
PIR2	OSCFIF	CMIF	_	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF	58			
PIE2	OSCFIE	CMIE	_	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE	58			
IPR2	OSCFIP	CMIP		EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP	58			
TRISB	PORTB Da	ta Direction R	egister						58			
TRISC	PORTC Da	PORTC Data Direction Register										
TRISD ⁽²⁾	PORTD Data Direction Register											
TMR1L	Timer1 Reg	jister Low Byt	е						56			
TMR1H	Timer1 Reg	jister High By	te						56			
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR10N	56			
TMR2	Timer2 Reg	jister							56			
T2CON		T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0	56			
PR2	Timer2 Peri	iod Register							56			
TMR3L	Timer3 Reg	jister Low Byt	е						57			
TMR3H	Timer3 Reg	jister High By	te						57			
T3CON	RD16	T3CCP2	T3CKPS1	T3CKPS0	T3CCP1	T3SYNC	TMR3CS	TMR3ON	57			
CCPR1L	Capture/Co	mpare/PWM	Register 1 Lo	w Byte					57			
CCPR1H	Capture/Co	Capture/Compare/PWM Register 1 High Byte										
CCP1CON	P1M1 ⁽²⁾	P1M0 ⁽²⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	57			
ECCP1AS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1 ⁽²⁾	PSSBD0 ⁽²⁾	57			
ECCP1DEL	PRSEN	PDC6 ⁽²⁾	PDC5 ⁽²⁾	PDC4 ⁽²⁾	PDC3 ⁽²⁾	PDC2 ⁽²⁾	PDC1 ⁽²⁾	PDC0 ⁽²⁾	57			

TABLE 17-3: REGISTERS ASSOCIATED WITH ECCP1 MODULE AND TIMER1 TO TIMER3

Legend: — = unimplemented, read as '0'. Shaded cells are not used during ECCP operation.

Note 1: The SBOREN bit is only available when the BOREN<1:0> Configuration bits = 01; otherwise, it is disabled and reads as '0'. See Section 5.4 "Brown-out Reset (BOR)".

2: These registers and/or bits are unimplemented on 28-pin devices; always maintain these bits clear.

PIC18F2221/2321/4221/4321 FAMILY

FIGURE 18-5:	SPI N	IODE W	/AVEFO	RM (SL	AVE MC	DE WIT	H CKE :	= 0)			
- SS Opäend	• • •										
940R (C242P = 0											
- CBASS # 30) - SACK	: 4		· · ·	· · ·		· ·		· : : ······			3 3
(ONA + 1 OKE + 0)	: : :										
Write to SSPSUF		•	2 2 2 2	: : : :	e Garana E E	•	2 2 2	· 	(*
9870		K 158 V	 X	X 58.6		X68.3	X 68.0			<u>(3</u> 3-6)	
809 (9849 = 0)	· · · · ·			dijijijijiji							* 1
inguja Samagoiae	- - - 		, , , , , , , , , , , , , , , , , , , ,	. 49.						<i>a</i> .	
(68892 # 0) 6582999 100800896 20820896	: : :		- 5 5 5 5	e • • • •	- 5 5 6 6	: :	· 5 5 5 5	, , ; ;			
SSPSR 5 SSPBLF	, , 6	• • •)) //////////////////////////////////	: : 	t t 5	• • •) ; ,	: : · · · · · · · · · · · · · · · · · ·		2002 (20 2020 (20 2020 (20	n yayoker Ta

FIGURE 18-6: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)



18.4.5 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I²C bus is such that the first byte after the Start condition usually determines which device will be the slave addressed by the master. The exception is the general call address which can address all devices. When this address is used, all devices should, in theory, respond with an Acknowledge.

The general call address is one of eight addresses reserved for specific purposes by the I^2C protocol. It consists of all '0's with R/W = 0.

The general call address is recognized when the General Call Enable bit, GCEN, is enabled (SSPCON2<7> is set). Following a Start bit detect, 8 bits are shifted into the SSPSR and the address is compared against the SSPADD. It is also compared to the general call address and fixed in hardware.

If the general call address matches, the SSPSR is transferred to the SSPBUF, the BF flag bit is set (eighth bit) and on the falling edge of the ninth bit (ACK bit), the SSPIF interrupt flag bit is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the SSPBUF. The value can be used to determine if the address was device specific or a general call address.

In 10-bit mode, the SSPADD is required to be updated for the second half of the address to match and the UA bit (SSPSTAT<1>) is set. If the general call address is sampled when the GCEN bit is set, while the slave is configured in 10-Bit Addressing mode, then the second half of the address is not necessary, the UA bit will not be set and the slave will begin receiving data after the Acknowledge (Figure 18-17).





18.4.14 SLEEP OPERATION

While in Sleep mode, the I^2C module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

18.4.15 EFFECTS OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

18.4.16 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I²C bus may be taken when the P bit (SSPSTAT<4>) is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the MSSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed in hardware with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- · Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

18.4.17 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF and reset the I^2C port to its Idle state (Figure 18-27).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I^2C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPSTAT register, or the bus is Idle and the S and P bits are cleared.

FIGURE 18-27: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



18.4.17.2 Bus Collision During a Repeated Start Condition

During a Repeated Start condition, a bus collision occurs if:

- a) A low level is sampled on SDA when SCL goes from low level to high level.
- b) SCL goes low before SDA is asserted low, indicating that another master is attempting to transmit a data '1'.

When the user deasserts SDA and the pin is allowed to float high, the BRG is loaded with SSPADD<6:0> and counts down to 0. The SCL pin is then deasserted and when sampled high, the SDA pin is sampled.

If SDA is low, a bus collision has occurred (i.e., another master is attempting to transmit a data '0', Figure 18-31). If SDA is sampled high, the BRG is reloaded and begins counting. If SDA goes from high-to-low before the BRG times out, no bus collision occurs because no two masters can assert SDA at exactly the same time.

If SCL goes from high-to-low before the BRG times out and SDA has not already been asserted, a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition, see Figure 18-32.

If, at the end of the BRG time-out, both SCL and SDA are still high, the SDA pin is driven low and the BRG is reloaded and begins counting. At the end of the count, regardless of the status of the SCL pin, the SCL pin is driven low and the Repeated Start condition is complete.



FIGURE 18-31: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

FIGURE 18-32: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



					SYNC	= 0, BRGH	I = 0, BRO	616 = 0				
BAUD	Fosc	= 40.000) MHz	Fosc	= 20.000) MHz	Fosc = 10.000 MHz			Fosc = 8.000 MHz		
(K)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)
0.3	_						_			_		
1.2	—	—	—	1.221	1.73	255	1.202	0.16	129	1.201	-0.16	103
2.4	2.441	1.73	255	2.404	0.16	129	2.404	0.16	64	2.403	-0.16	51
9.6	9.615	0.16	64	9.766	1.73	31	9.766	1.73	15	9.615	-0.16	12
19.2	19.531	1.73	31	19.531	1.73	15	19.531	1.73	7	—	_	_
57.6	56.818	-1.36	10	62.500	8.51	4	52.083	-9.58	2	—	_	_
115.2	125.000	8.51	4	104.167	-9.58	2	78.125	-32.18	1	—	_	_

TABLE 19-3: BAUD RATES FOR ASYNCHRONOUS MODES

			S	YNC = 0, E	BRGH = (), BRG16 =	0			
BAUD	Fos	c = 4.000	MHz	Fos	c = 2.000	MHz	Fosc = 1.000 MHz			
(K)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	
0.3	0.300	0.16	207	0.300	-0.16	103	0.300	-0.16	51	
1.2	1.202	0.16	51	1.201	-0.16	25	1.201	-0.16	12	
2.4	2.404	0.16	25	2.403	-0.16	12	—	_	_	
9.6	8.929	-6.99	6	—	_	_	—	_	_	
19.2	20.833	8.51	2	—	_	_	—	_	_	
57.6	62.500	8.51	0	—	_	_	—	_	_	
115.2	62.500	-45.75	0	_	—	_	_	—	—	

					SYNC	= 0, BRGH	i = 1, BRG	16 = 0					
BAUD	Fosc	= 40.000) MHz	Fosc = 20.000 MHz			Fosc = 10.000 MHz			Fos	Fosc = 8.000 MHz		
(K)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	ual % SPBRG ıte % value () Error (decimal		Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	
0.3	_	_	_							—	_	_	
1.2	—	—	—	—	—	—	—	—	—	—	—	—	
2.4	_	_	_	—	_	_	2.441	1.73	255	2.403	-0.16	207	
9.6	9.766	1.73	255	9.615	0.16	129	9.615	0.16	64	9.615	-0.16	51	
19.2	19.231	0.16	129	19.231	0.16	64	19.531	1.73	31	19.230	-0.16	25	
57.6	58.140	0.94	42	56.818	-1.36	21	56.818	-1.36	10	55.555	3.55	8	
115.2	113.636	-1.36	21	113.636	-1.36	10	125.000	8.51	4	—	—	_	

			S	YNC = 0, E	BRGH = 1	L, BRG16 =	i = 0				
BAUD	Fose	c = 4.000	MHz	Fos	c = 2.000	MHz	Fosc = 1.000 MI		MHz		
(K)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)	Actual Rate (K)	% Error	SPBRG value (decimal)		
0.3			_	_		_	0.300	-0.16	207		
1.2	1.202	0.16	207	1.201	-0.16	103	1.201	-0.16	51		
2.4	2.404	0.16	103	2.403	-0.16	51	2.403	-0.16	25		
9.6	9.615	0.16	25	9.615	-0.16	12	—	—	—		
19.2	19.231	0.16	12	—	_	_	—	_	_		
57.6	62.500	8.51	3	—	—	_	—	_	_		
115.2	125.000	8.51	1	—	_	—	—	_	_		

© 2009 Microchip Technology Inc.

21.0 COMPARATOR MODULE

The analog comparator module contains two comparators that can be configured in a variety of ways. The inputs can be selected from the analog inputs multiplexed with pins RA0 through RA5, as well as the on-chip voltage reference (see Section 22.0 "Comparator Voltage Reference Module"). The digital outputs (normal or inverted) are available at the pin level and can also be read through the control register.

The CMCON register (Register 21-1) selects the comparator input and output configuration. Block diagrams of the various comparator configurations are shown in Figure 21-1.

REGISTER 21-1: CMCON: COMPARATOR CONTROL REGISTER

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1
C2OUT	C10UT	C2INV	C1INV	CIS	CM2	CM1	CM0
bit 7							bit 0

bit 7 C2OUT: Comparator 2 Output bit When C2INV = 0: 1 = C2 VIN+ > C2 VIN-0 = C2 VIN + < C2 VIN -When C2INV = 1: 1 = C2 VIN + < C2 VIN -0 = C2 VIN+ > C2 VINbit 6 C1OUT: Comparator 1 Output bit When C1INV = 0: 1 = C1 VIN+ > C1 VIN-0 = C1 VIN + < C1 VIN -When C1INV = 1: 1 = C1 VIN + < C1 VIN -0 = C1 VIN + > C1 VIN bit 5 C2INV: Comparator 2 Output Inversion bit 1 = C2 output inverted 0 = C2 output not inverted bit 4 C1INV: Comparator 1 Output Inversion bit 1 = C1 output inverted 0 = C1 output not inverted bit 3 CIS: Comparator Input Switch bit When CM<2:0> = 110: 1 = C1 VIN- connects to RA3/AN3/VREF+ C2 VIN- connects to RA2/AN2/VREF-/CVREF 0 = C1 VIN- connects to RA0/AN0 C2 VIN- connects to RA1/AN1 bit 2-0 CM<2:0>: Comparator Mode bits Figure 21-1 shows the Comparator modes and the CM<2:0> bit settings. I agand.

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

File	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
300008h	CONFIG5L		_	—	—	_	_	CP1	CP0
300009h	CONFIG5H	CPD	CPB	—	—	—	—	—	—
30000Ah	CONFIG6L	_	—	—	—	—	—	WRT1	WRT0
30000Bh	CONFIG6H	WRTD	WRTB	WRTC	—	—	—	—	—
30000Ch	CONFIG7L	_	—	—	—	—	—	EBTR1	EBTR0
30000Dh	CONFIG7H	_	EBTRB	—	—	—	—	—	_

TABLE 24-3: SUMMARY OF CODE PROTECTION REGISTERS

Legend: Shaded cells are unimplemented.

24.5.1 PROGRAM MEMORY CODE PROTECTION

The program memory may be read to or written from any location using the table read and table write instructions. The device ID may be read with table reads. The Configuration registers may be read and written with the table read and table write instructions.

In normal execution mode, the CPn bits have no direct effect. CPn bits inhibit external reads and writes. A block of user memory may be protected from table writes if the WRTn Configuration bit is '0'. The EBTRn bits control table reads. For a block of user memory with the EBTRn bit set to '0', a table read instruction that executes from within that block is allowed to read. A table read instruction that executes from a location outside of that block is not allowed to read and will result in reading '0's. Figures 24-6 through 24-8 illustrate table write and table read protection.

Note: Code protection bits may only be written to a '0' from a '1' state. It is not possible to write a '1' to a bit in the '0' state. Code protection bits are only set to '1' by a full chip erase or block erase function. The full chip erase and block erase functions can only be initiated via ICSP operation or an external programmer.

FIGURE 24-6: TABLE WRITE (WRTn) DISALLOWED



PIC18F2221/2321/4221/4321 FAMILY

FIGURE 24-7: EXTERNAL BLOCK TABLE READ (EBTRn) DISALLOWED



FIGURE 24-8: EXTERNAL BLOCK TABLE READ (EBTRn) ALLOWED



25.1.1 STANDARD INSTRUCTION SET

ADD	DLW	ADD Liter	ral to W	1					
Synta	ax:	ADDLW	ADDLW k						
Oper	ands:	$0 \le k \le 255$							
Oper	ation:	(W) + k \rightarrow V	N						
Statu	is Affected:	N, OV, C, E	N, OV, C, DC, Z						
Enco	oding:	0000	1111	kkkk	kkkk				
Desc	cription:	The conten 8-bit literal W.	ts of W a 'k' and th	are added ie result is	to the placed in				
Word	ds:	1	1						
Cycle	es:	1	1						
QC	ycle Activity:								
	Q1	Q2	Q3	3	Q4				
	Decode	Read literal 'k'	Proce Data	ess W a	rite to W				
Exan	nple:	ADDLW 1	.5h						
Before Instruction		tion							
	- VV =	10h							
	After Instruction	on							
	W =	25h							

ADDWF	ADD W to f
Syntax:	ADDWF f {,d {,a}}
Operands:	$0 \le f \le 255$ $d \in [0, 1]$ $a \in [0, 1]$
Operation:	(W) + (f) \rightarrow dest
Status Affected:	N, OV, C, DC, Z
Encoding:	0010 01da ffff ffff
Description:	Add W to register 'f'. If 'd' is '0', the result is stored 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 '0', 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

QC	ycle Activity:			
	Q1	Q2	Q3	Q4
	Decode	Read register 'f'	Process Data	Write to destination
<u>Exan</u>	<u>nple:</u>	ADDWF	REG, 0, 0	
	Before Instruc	tion		
	W REG After Instructio	= 17h = 0C2h		

0D9h

0C2h

=

=

W

REG

Note: All PIC18 instructions may take an optional label argument preceding the instruction mnemonic for use in symbolic addressing. If a label is used, the instruction format then becomes: {label} instruction argument(s).

Param No.	Sym	Characteristic	Min	Тур†	Max	Units	Conditions
F10	Fosc	Oscillator Frequency Range	4	_	10	MHz	HS mode only
F11	Fsys	On-Chip VCO System Frequency	16	—	40	MHz	HS mode only
F12	t _{rc}	PLL Start-up Time (Lock Time)	—	—	2	ms	
F13	ΔCLK	CLKO Stability (Jitter)	-2	—	+2	%	

TABLE 27-7: PLL CLOCK TIMING SPECIFICATIONS (VDD = 4.2V TO 5.5V)

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

TABLE 27-8: AC CHARACTERISTICS: INTERNAL RC ACCURACY

Standard Operating Conditions (unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +85^{\circ}C$ for industrial $-40^{\circ}C \le TA \le +125^{\circ}C$ for extended									
Param No.	m Device Min Typ Max Units Conditions								
	INTOSC Accuracy @ Freq = 8 MHz	, 4 MHz,	2 MHz, ′	I MHz, 50	0 kHz, 2	250 kHz, 125 kHz, 31	kHz ⁽¹⁾		
	PIC18LF2221/2321/4221/4321	-2	+/-1	2	%	+25°C	VDD = 2.0-5.5V		
		-5		5	%	-10°C to +85°C	VDD = 2.0-5.5V		
		-10	+/-1	10	%	-40°C to +85°C	VDD = 2.0-5.5V		
	PIC18F2221/2321/4221/4321	-2	+/-1	2	%	+25°C	VDD = 4.2-5.5V		
		-5	_	5	%	-10°C to +85°C	VDD = 4.2-5.5V		
		-10	+/-1	10	%	-40°C to +85°C	VDD = 4.2-5.5V		
	INTRC Accuracy @ Freq = 31 kHz								
	PIC18LF2221/2321/4221/4321	26.562	_	35.938	kHz	-40°C to +85°C	VDD = 2.0-5.5V		
	PIC18F2221/2321/4221/4321	26.562		35.938	kHz	-40°C to +85°C	VDD = 4.2-5.5V		

Note 1: Frequency calibrated at 25°C. OSCTUNE register can be used to compensate for temperature drift.

Param. No.	Symbol	Charact	eristic	Min	Max	Units	Conditions
100	Thigh	Clock High Time	100 kHz mode	4.0	—	μS	
			400 kHz mode	0.6		μs	
			MSSP Module	1.5 Tcy	_		
101	TLOW	Clock Low Time	100 kHz mode	4.7	—	μs	
			400 kHz mode	1.3	_	μs	
			MSSP Module	1.5 Tcy	—		
102	TR	SDA and SCL Rise	100 kHz mode	—	1000	ns	
		Time	400 kHz mode	20 + 0.1 Св	300	ns	CB is specified to be from 10 to 400 pF
103	TF	SDA and SCL Fall	100 kHz mode		300	ns	
		Time	400 kHz mode	20 + 0.1 Св	300	ns	CB is specified to be from 10 to 400 pF
90	TSU:STA	Start Condition	100 kHz mode	4.7		μs	Only relevant for Repeated
		Setup Time	400 kHz mode	0.6		μs	Start condition
91	THD:STA	Start Condition	100 kHz mode	4.0	—	μs	After this period, the first
		Hold Time	400 kHz mode	0.6	—	μs	clock pulse is generated
106	THD:DAT	Data Input Hold	100 kHz mode	0	_	ns	
		Time	400 kHz mode	0	0.9	μS	
107	TSU:DAT	Data Input Setup	100 kHz mode	250	—	ns	(Note 2)
		Time	400 kHz mode	100	_	ns	
92	Tsu:sto	Stop Condition	100 kHz mode	4.7	—	μS	
		Setup Time	400 kHz mode	0.6	_	μS	
109	ΤΑΑ	Output Valid from	100 kHz mode		3500	ns	(Note 1)
		Clock	400 kHz mode	—	—	ns	
110	TBUF	Bus Free Time	100 kHz mode	4.7	_	μS	Time the bus must be free
			400 kHz mode	1.3	—	μS	before a new transmission can start
D102	Св	Bus Capacitive Load	ding	_	400	pF	

TABLE 27-19: I²C[™] BUS DATA REQUIREMENTS (SLAVE MODE)

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode I²C bus device can be used in a Standard mode I²C bus system, but the requirement Tsu:DAT ≥ 250 ns must then be met. This will automatically be the case 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. + Tsu:DAT = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification), before the SCL line is released.

28.0 PACKAGING INFORMATION

28.1 Package Marking Information

28-Lead SPDIP



28-Lead SOIC



28-Lead QFN



28-Lead SSOP





Example



Example



Example



Legend	: XXX Y YY WW NNN @3 *	Customer-specific information Year code (last digit of calendar year) Year code (last 2 digits of calendar year) Week code (week of January 1 is week '01') Alphanumeric traceability code Pb-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 even be carried characters	nt the full Microchip part number cannot be marked on one line, it will d over to the next line, thus limiting the number of available s for customer-specific information.

Example Frequencies/Resolutions	151
Operation Setup	
Period	150
TMR2 to PR2 Match	150, 155
PWM (ECCP Module)	
CCPR1H:CCPR1L Registers	
Duty Cycle	156
Effects of a Reset	
Enhanced PWM Auto-Shutdown	
Example Frequencies/Resolutions	
Full-Bridge Application Example	
Full-Bridge Mode	
Direction Change	
Half-Bridge Mode	
Half-Bridge Output Mode Applications	
Example	
Operation in Power-Managed Modes	
Operation with Fail-Safe Clock Monitor	
Output Configurations	
Output Relationships (Active-High)	
Output Relationships (Active-Low)	
Period	
Programmable Dead-Band Delay	
Setup for PWM Operation	
Start-up Considerations	
Q	
	151 156
	101. 100

R

ĸ	
RAM. See Data Memory.	
RC Oscillator	
RCIO Oscillator Mode	31
RC_IDLE Mode	
RC_RUN Mode	41
RCALL	309
RCON Register	
Bit Status During Initialization	54
Reader Response	400
Register File	67
Register File Summary	69–71
Registers	
ADCON0 (A/D Control 0)	233
ADCON1 (A/D Control 1)	234
ADCON2 (A/D Control 2)	235
BAUDCON (Baud Rate Control)	214
CCP1CON (Enhanced Capture/Compare/PWM	
Control 1)	153
CCPxCON (CCPx Control)	145
CMCON (Comparator Control)	243
CONFIG1H (Configuration 1 High)	260
CONFIG2H (Configuration 2 High)	262
CONFIG2L (Configuration 2 Low)	261
CONFIG3H (Configuration 3 High)	263
CONFIG4L (Configuration 4 Low)	264
CONFIG5H (Configuration 5 High)	265
CONFIG5L (Configuration 5 Low)	265
CONFIG6H (Configuration 6 High)	266
CONFIG6L (Configuration 6 Low)	266
CONFIG7H (Configuration 7 High)	267
CONFIG7L (Configuration 7 Low)	267
CVRCON (Comparator Voltage	
Reference Control)	249
DEVID1 (Device ID 1)	268
DEVID2 (Device ID 2)	268
ECCP1AS (ECCP Auto-Shutdown Control)	163

ECCP1DEL (PWM Dead-Band Delay) 162	2
EECON1 (Data EEPROM Control 1) 81, 90	С
HLVDCON (High/Low-Voltage Detect Control) 253	3
INTCON (Interrupt Control)	9
INTCON2 (Interrupt Control 2) 100	0
INTCON3 (Interrupt Control 3)	1
IPR1 (Peripheral Interrupt Priority 1)	6
IPR2 (Perinheral Interrupt Priority 2) 107	7
OSCCON (Oscillator Control)	7
OSCIUNE (Oscillator Tuning)	י כ
DIE1 (Device and Interrupt Enable 1)	ך ג
PIE1 (Peripheral Interrupt Enable 1) 104	+
PIE2 (Peripheral Interrupt Enable 2) 10	C
PIR1 (Peripheral Interrupt Request (Flag) 1) 102	2
PIR2 (Peripheral Interrupt Request (Flag) 2) 103	3
RCON (Reset Control) 48, 108	3
RCSTA (Receive Status and Control) 213	3
SSPADD(MSSP Address)	C
SSPCON1 (MSSP Control 1, I ² C Mode)	8
SSPCON1 (MSSP Control 1 SPI Mode) 169	9
SSPCON2 (MSSP Control 2, I ² C Mode) 179	à
SSPSTAT (MSSP Status 12C Mode) 17	7
SSESTAT (MSSE Status, TC Wode)	r D
	2 2
	2
STKPTR (Stack Pointer)	1
TOCON (Timero Control) 128	9
T1CON (Timer1 Control) 133	3
T2CON (Timer2 Control) 139	Э
T3CON (Timer3 Control) 147	1
TRISE (PORTE/PSP Control) 124	4
TXSTA (Transmit Status and Control)	2
WDTCON (Watchdog Timer Control) 27(0
RESET 300	ģ
Reset State of Peristers	1
Resets 17 250	т О
Prown out Pooot (POP)	9 D
Diowil-out Reset (DOR)	2
Oscillator Start-up Timer (OST)	9
Power-on Reset (POR)	9
Power-up Timer (PWRT)	9
RETFIE	C
RETLW	C
RETURN	1
Return Address Stack 60	C
Associated Registers 60	C
Return Stack Pointer (STKPTR)	1
Revision History	5
RICE 31	1
RINCE 31	2
RRCF 247	2
	2
NNNOT	J
S	
SCK 46	7
SDI 101	7
	7
	(A
SEC_IDLE MODE	4
SEC_RUN Mode 40	J

 Slave Select (SS)
 167

 SLEEP
 314

OSC1 and OSC2 Pin States 38

Serial Peripheral Interface. See SPI Mode.

Single-Supply ICSP Programming.

Sleep