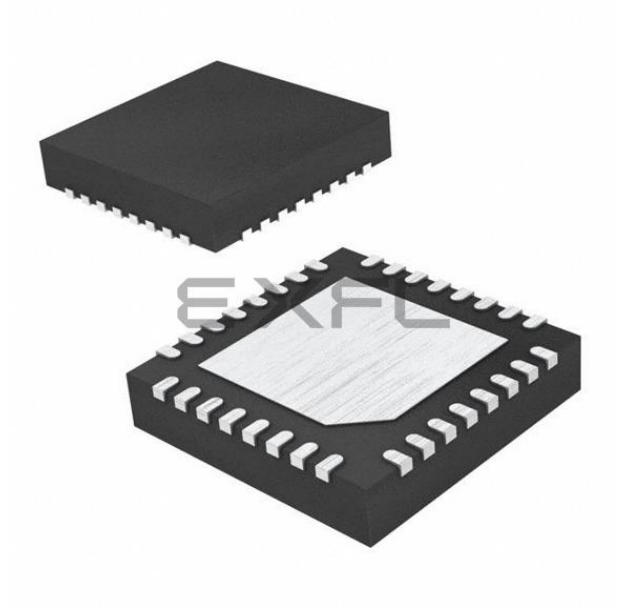


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, Power Control PWM, QEI, POR, PWM, WDT
Number of I/O	24
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 5x10b
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-S (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f2431-e-mm

4.0 POWER-MANAGED MODES

PIC18F2331/2431/4331/4431 devices offer a total of seven operating modes for more efficient power management. These modes provide a variety of options for selective power conservation in applications where resources may be limited (i.e., battery-powered devices).

There are three categories of power-managed modes:

- Run modes
- Idle modes
- Sleep mode

These categories define which portions of the device are clocked, and sometimes, what speed. The Run and Idle modes may use any of the three available clock sources (primary, secondary or internal oscillator block); the Sleep mode does not use a clock source.

The power-managed modes include several power-saving features offered on previous PIC® devices. One is the clock switching feature, offered in other PIC18 devices, allowing the controller to use the Timer1 oscillator in place of the primary oscillator. Also included is the Sleep mode, offered by all PIC devices, where all device clocks are stopped.

4.1 Selecting Power-Managed Modes

Selecting a power-managed mode requires two decisions: if the CPU is to be clocked or not and the selection of a clock source. The IDLEN bit (OSCCON<7>) controls CPU clocking, while the SCS<1:0> bits (OSCCON<1:0>) select the clock source. The individual modes, bit settings, clock sources and affected modules are summarized in Table 4-1.

4.1.1 CLOCK SOURCES

The SCS<1:0> bits allow the selection of one of three clock sources for power-managed modes. They are:

- the primary clock, as defined by the FOSC<3:0> Configuration bits
- the secondary clock (the Timer1 oscillator)
- the internal oscillator block (for RC modes)

4.1.2 ENTERING POWER-MANAGED MODES

Switching from one power-managed mode to another begins by loading the OSCCON register. The SCS<1:0> bits select the clock source and determine which Run or Idle mode is to be used. Changing these bits causes an immediate switch to the new clock source, assuming that it is running. The switch may also be subject to clock transition delays. These are discussed in **Section 4.1.3 "Clock Transitions and Status Indicators"** and subsequent sections.

Entry to the power-managed Idle or Sleep modes is triggered by the execution of a SLEEP instruction. The actual mode that results depends on the status of the IDLEN bit.

Depending on the current mode and the mode being switched to, a change to a power-managed mode does not always require setting all of these bits. Many transitions may be done by changing the oscillator select bits, or changing the IDLEN bit, prior to issuing a SLEEP instruction. If the IDLEN bit is already configured correctly, it may only be necessary to perform a SLEEP instruction to switch to the desired mode.

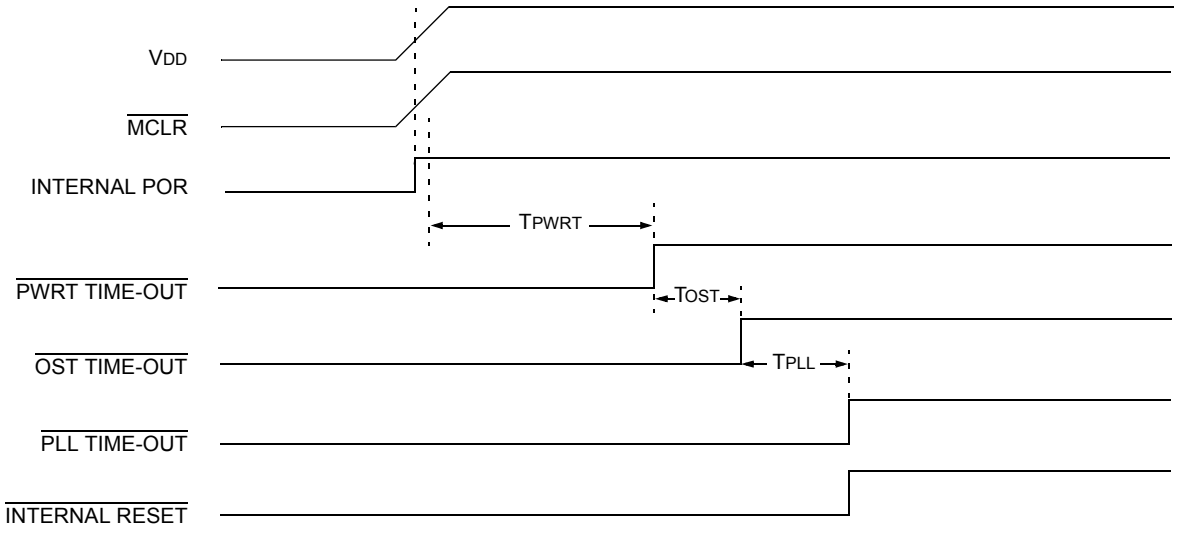
TABLE 4-1: POWER-MANAGED MODES

Mode	OSCCON Bits<7,1:0>		Module Clocking		Available Clock and Oscillator Source
	IDLEN(1)	SCS<1:0>	CPU	Peripherals	
Sleep	0	N/A	Off	Off	None – All clocks are disabled
PRI_RUN	N/A	00	Clocked	Clocked	Primary – LP, XT, HS, HSPLL, RC, EC and Internal Oscillator Block.(2) This is the normal, full-power execution mode.
SEC_RUN	N/A	01	Clocked	Clocked	Secondary – Timer1 Oscillator
RC_RUN	N/A	1x	Clocked	Clocked	Internal Oscillator Block(2)
PRI_IDLE	1	00	Off	Clocked	Primary – LP, XT, HS, HSPLL, RC, EC
SEC_IDLE	1	01	Off	Clocked	Secondary – Timer1 Oscillator
RC_IDLE	1	1x	Off	Clocked	Internal Oscillator Block(2)

Note 1: IDLEN reflects its value when the SLEEP instruction is executed.

2: Includes INTOSC and INTOSC postscaler, as well as the INTRC source.

FIGURE 5-7: TIME-OUT SEQUENCE ON POR w/PLL ENABLED (MCLR TIED TO VDD)



Note: TOST = 1024 clock cycles.
TPLL ≈ 2 ms max. First three stages of the PWRT timer.

TABLE 5-2: STATUS BITS, THEIR SIGNIFICANCE AND THE INITIALIZATION CONDITION FOR RCON REGISTER

Condition	Program Counter	RCON Register	RI	TO	PD	POR	BOR	STKFUL	STKUNF
Power-on Reset	0000h	0--1 1100	1	1	1	0	0	0	0
RESET Instruction	0000h	0--0 uuuu	0	u	u	u	u	u	u
Brown-out	0000h	0--1 11u-	1	1	1	u	0	u	u
MCLR Reset during power-managed Run modes	0000h	0--u 1uuu	u	1	u	u	u	u	u
MCLR Reset during power-managed Idle and Sleep modes	0000h	0--u 10uu	u	1	0	u	u	u	u
WDT Time-out during full power or power-managed Run modes	0000h	0--u 0uuu	u	0	u	u	u	u	u
MCLR Reset during full-power execution	0000h	0--u uuuu	u	u	u	u	u	u	u
Stack Full Reset (STVREN = 1)								1	u
Stack Underflow Reset (STVREN = 1)								u	1
Stack Underflow Error (not an actual Reset, STVREN = 0)	0000h	u--u uuuu	u	u	u	u	u	u	1
WDT time-out during power-managed Idle or Sleep modes	PC + 2	u--u 00uu	u	0	0	u	u	u	u
Interrupt exit from power-managed modes	PC + 2(1)	u--u u0uu	u	u	0	u	u	u	u

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0'.

Note 1: When the wake-up is due to an interrupt and the GIEH or GIEL bits are set, the PC is loaded with the interrupt vector (0x000008h or 0x000018h).

NOTES:

13.0 TIMER1 MODULE

The Timer1 timer/counter module has the following features:

- 16-bit timer/counter (two 8-bit registers; TMR1H and TMR1L)
- Readable and writable (both registers)
- Internal or external clock select
- Interrupt-on-overflow from FFFFh to 0000h
- Reset from CCP module Special Event Trigger
- Status of system clock operation

Figure 13-1 is a simplified block diagram of the Timer1 module.

Register 13-1 details the Timer1 Control register. This register controls the operating mode of the Timer1 module and contains the Timer1 Oscillator Enable bit (T1OSCEN). Timer1 can be enabled or disabled by setting or clearing control bit, TMR1ON (T1CON<0>).

The Timer1 oscillator can be used as a secondary clock source in power-managed modes. When the T1RUN bit is set, the Timer1 oscillator provides the system clock. If the Fail-Safe Clock Monitor is enabled and the Timer1 oscillator fails while providing the system clock, polling the T1RUN bit will indicate whether the clock is being provided by the Timer1 oscillator or another source.

Timer1 can also be used to provide Real-Time Clock (RTC) functionality to applications with only a minimal addition of external components and code overhead.

REGISTER 13-1: T1CON: TIMER1 CONTROL REGISTER

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
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 **RD16:** 16-Bit Read/Write Mode Enable bit
1 = Enables register read/write of TImer1 in one 16-bit operation
0 = Enables register read/write of Timer1 in two 8-bit operations

bit 6 **T1RUN:** Timer1 System Clock Status bit
1 = Device clock is derived from Timer1 oscillator
0 = Device clock is derived from another source

bit 5-4 **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

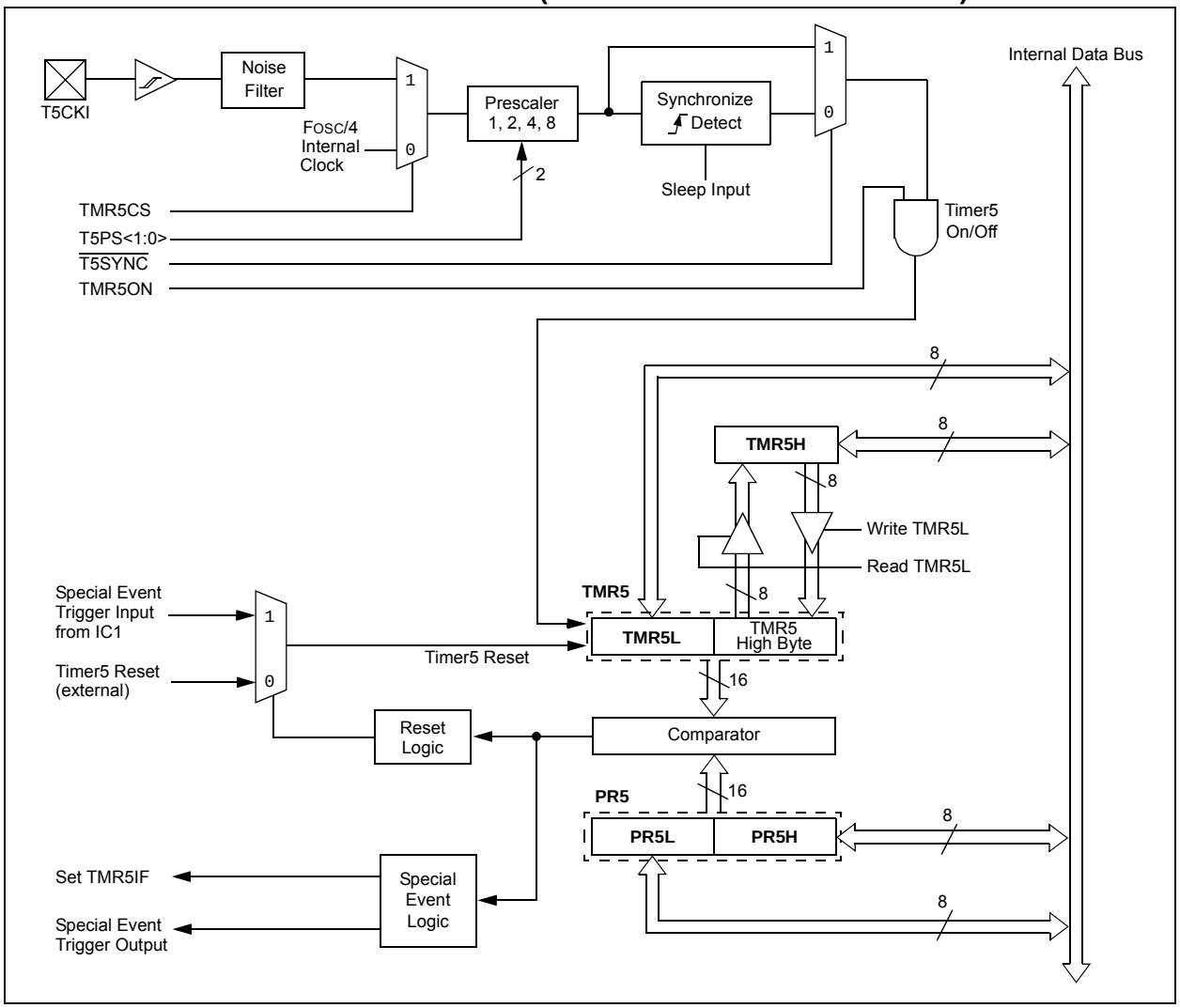
bit 3 **T1OSCEN:** Timer1 Oscillator Enable bit
1 = Timer1 oscillator is enabled
0 = Timer1 oscillator is shut off
The oscillator inverter and feedback resistor are turned off to eliminate power drain.

bit 2 **$\overline{\text{T1SYNC}}$:** Timer1 External Clock Input Synchronization Select bit
When TMR1CS = 1 (External Clock):
1 = Do not synchronize external clock input
0 = Synchronize external clock input
When TMR1CS = 0 (Internal Clock):
This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.

bit 1 **TMR1CS:** Timer1 Clock Source Select bit
1 = External clock from pin RC0/T1OSO/T1CKI (on the rising edge)
0 = Internal clock (FOSC/4)

bit 0 **TMR1ON:** Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

FIGURE 15-1: TIMER5 BLOCK DIAGRAM (16-BIT READ/WRITE MODE SHOWN)



15.1 Timer5 Operation

Timer5 combines two 8-bit registers to function as a 16-bit timer. The TMR5L register is the actual low byte of the timer; it can be read and written to directly. The high byte is contained in an unmapped register; it is read and written to through TMR5H, which serves as a buffer. Each register increments from 00h to FFh.

A second register pair, PR5H and PR5L, serves as the Period register; it sets the maximum count for the TMR5 register pair. When TMR5 reaches the value of PR5, the timer rolls over to 00h and sets the TMR5IF interrupt flag. A simplified block diagram of the Timer5 module is shown in Figure 2-1.

Note:	The Timer5 may be used as a general purpose timer and as the time base resource to the Motion Feedback Module (Input Capture or Quadrature Encoder Interface).

Timer5 supports three configurations:

- 16-Bit Synchronous Timer
- 16-Bit Synchronous Counter
- 16-Bit Asynchronous Counter

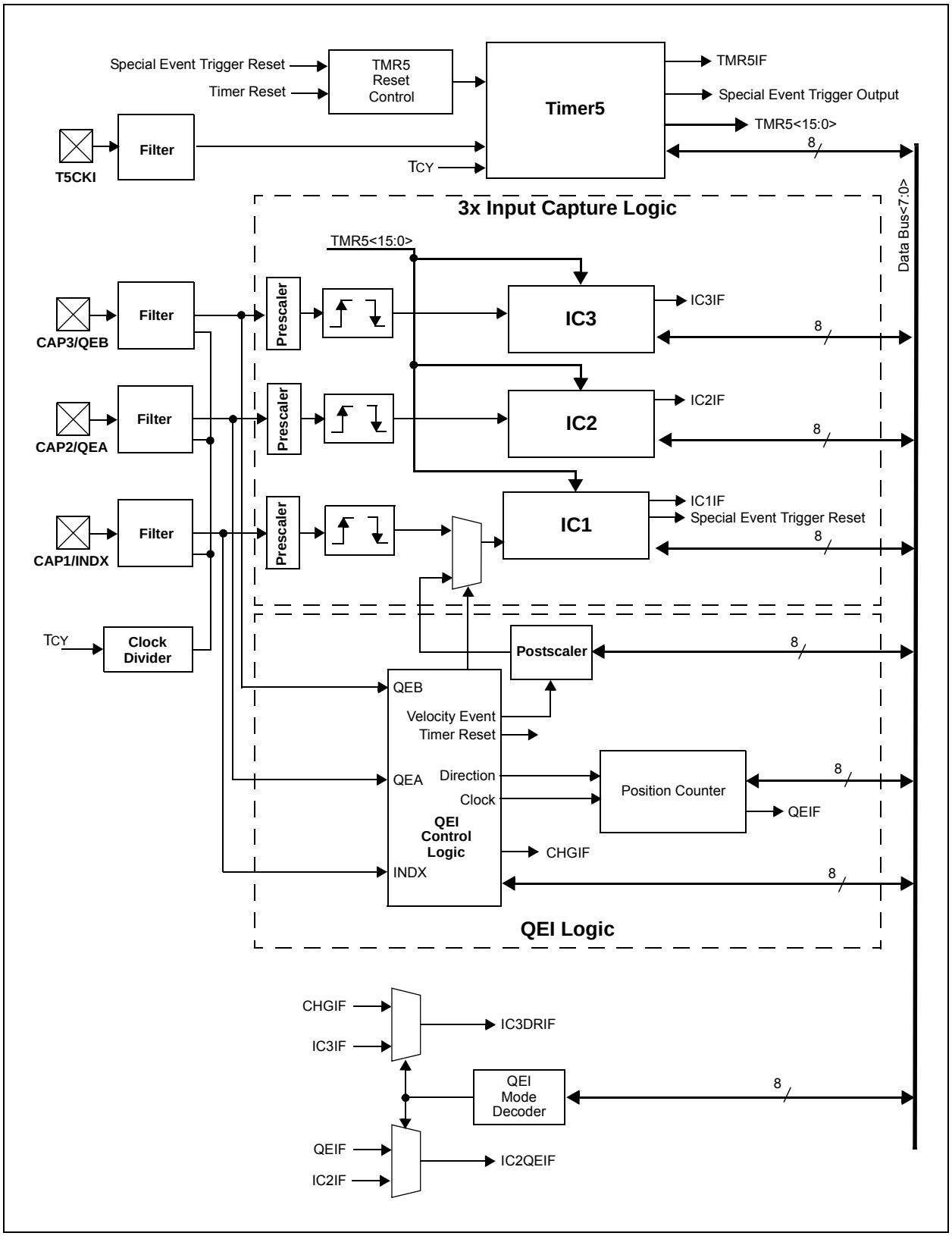
In Synchronous Timer configuration, the timer is clocked by the internal device clock. The optional Timer5 prescaler divides the input by 2, 4, 8 or not at all (1:1). The TMR5 register pair increments on Q1. Clearing TMR5CS (= 0) selects the internal device clock as the timer sampling clock.

TABLE 16-2: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE AND TIMER1

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	54
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	57
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	57
IPR1	—	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	57
TRISC	PORTC Data Direction Register								57
TMR1L	Timer1 Register Low Byte								55
TMR1H	Timer1 Register High Byte								55
T1CON	RD16	T1RUN	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	55
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								56
CCPR1H	Capture/Compare/PWM Register 1 High Byte								56
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	56
CCPR2L	Capture/Compare/PWM Register 2 Low Byte								56
CCPR2H	Capture/Compare/PWM Register 2 High Byte								56
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	56
PIR2	OSCFIF	—	—	EEIF	—	LVDIF	—	CCP2IF	57
PIE2	OSCFIE	—	—	EEIE	—	LVDIE	—	CCP2IE	57
IPR2	OSCFIP	—	—	EEIP	—	LVDIP	—	CCP2IP	57

Legend: — = unimplemented, read as '0'. Shaded cells are not used by Capture, Compare and Timer1.

FIGURE 17-1: MOTION FEEDBACK MODULE BLOCK DIAGRAM



REGISTER 18-4: PWMCON1: PWM CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
SEVOPS3	SEVOPS2	SEVOPS1	SEVOPS0	SEVTDIR	—	UDIS	OSYNC
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 7-4 **SEVOPS<3:0>:** PWM Special Event Trigger Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
.
.
.
1111 = 1:16 Postscale

bit 3 **SEVTDIR:** Special Event Trigger Time Base Direction bit
1 = A Special Event Trigger will occur when the PWM time base is counting downwards
0 = A Special Event Trigger will occur when the PWM time base is counting upwards

bit 2 **Unimplemented:** Read as '0'

bit 1 **UDIS:** PWM Update Disable bit
1 = Updates from Duty Cycle and Period Buffer registers are disabled
0 = Updates from Duty Cycle and Period Buffer registers are enabled

bit 0 **OSYNC:** PWM Output Override Synchronization bit
1 = Output overrides via the OVDCON register are synchronized to the PWM time base
0 = Output overrides via the OVDCON register are asynchronous

18.3.1 FREE-RUNNING MODE

In the Free-Running mode, the PWM Time Base registers (PTMRL and PTMRH) will begin counting upwards until the value in the PWM Time Base Period register, PTPER (PTPERL and PTPERH), is matched. The PTMR registers will be reset on the following input clock edge and the time base will continue counting upwards as long as the PTEN bit remains set.

18.3.2 SINGLE-SHOT MODE

In the Single-Shot mode, the PWM time base will begin counting upwards when the PTEN bit is set. When the value in the PTMR register matches the PTPER register, the PTMR register will be reset on the following input clock edge and the PTEN bit will be cleared by the hardware to halt the time base.

18.3.3 CONTINUOUS UP/DOWN COUNT MODES

In Continuous Up/Down Count modes, the PWM time base counts upwards until the value in the PTPER register matches with the PTMR register. On the following input clock edge, the timer counts downwards. The PTDIR bit in the PTCON1 register is read-only and indicates the counting direction. The PTDIR bit is set when the timer counts downwards.

Note: Since the PWM compare outputs are driven to the active state when the PWM time base is counting downwards and matches the duty cycle value, the PWM outputs are held inactive during the first half of the first period of the Continuous Up/Down Count mode until PTMR begins to count down from the PTPER value.

18.3.4 PWM TIME BASE PRESCALER

The input clock to PTMR (FOSC/4) has prescaler options of 1:1, 1:4, 1:16 or 1:64. These are selected by control bits, PTCKPS<1:0>, in the PTCON0 register. The prescaler counter is cleared when any of the following occurs:

- Write to the PTMR register
- Write to the PTCON (PTCON0 or PTCON1) register
- Any device Reset

Note: The PTMR register is not cleared when PTCONx is written.

REGISTER 19-2: SSPCON: SYNCHRONOUS SERIAL PORT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV(1)	SSPEN(2)	CKP	SSPM3(3)	SSPM2(3)	SSPM1(3)	SSPM0(3)
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 **WCOL:** Write Collision Detect bit

1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)

0 = No collision

bit 6 **SSPOV**: Receive Overflow Indicator bit(1)

In SPI mode:

1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in Slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.

0 = No overflow

In I^2C™ mode:

1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode. SSPOV must be cleared in software in either mode.

0 = No overflow

bit 5 **SSPEN**: Synchronous Serial Port Enable bit(2)

In SPI mode:

1 = Enables serial port and configures SCK, SDO and SDI as serial port pins

0 = Disables serial port and configures these pins as I/O port pins

In I^2C mode:

1 = Enables the serial port and configures the SDA and SCL pins as serial port pins

0 = Disables serial port and configures these pins as I/O port pins

In both modes, when enabled, these pins must be properly configured as input or output.

bit 4 **CKP**: Clock Polarity Select bit

In SPI mode:

1 = Idle state for clock is a high level

0 = Idle state for clock is a low level

In I^2C mode:

SCK release control.

1 = Enables clock

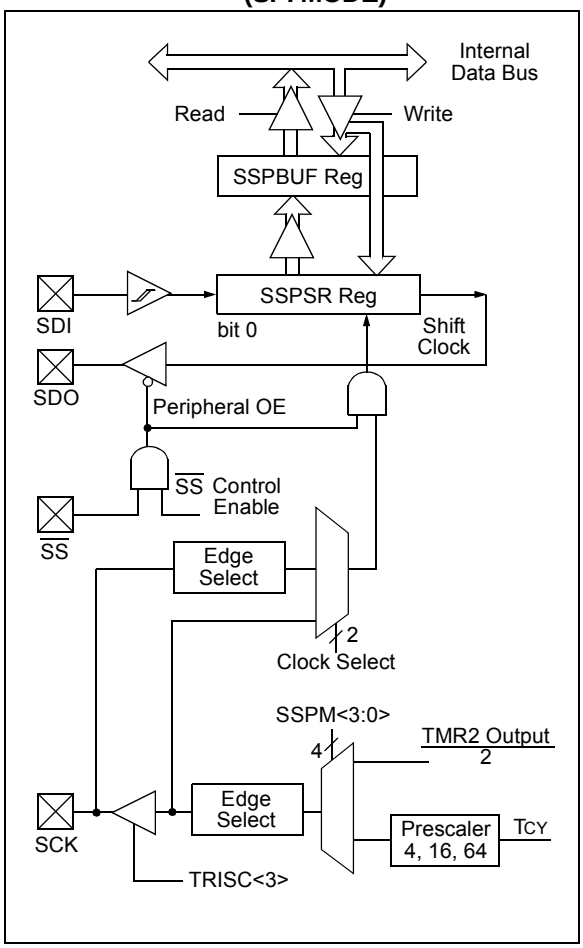
0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

Note 1: In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.

2: When enabled, these pins must be properly configured as inputs or outputs.

3: Bit combinations not specifically listed here are either reserved or implemented in I^2C™ mode only.

FIGURE 19-1: SSP BLOCK DIAGRAM (SPI MODE)



To enable the serial port, SSP Enable bit, SSPEN (SSPCON<5>), must be set. To reset or reconfigure SPI mode, clear bit SSPEN, reinitialize the SSPCON register and then set bit SSPEN. This configures the SDI, SDO, SCK and $\overline{SS}$ pins as serial port pins. For the pins to behave as the serial port function, they must have their data direction bits (in the TRISC register) appropriately programmed. That is:

- Serial Data Out (SDO) – RC7/RX/DT/SDO or RD1/SDO
- SDI must have TRISC<4> or TRISD<2> set
- SDO must have TRISC<7> or TRISD<1> cleared
- SCK (Master mode) must have TRISC<5> or TRISD<3> cleared
- SCK (Slave mode) must have TRISC<5> or TRISD<3> set
- $\overline{SS}$ must have TRISA<6> set

Note 1: When the SPI is in Slave mode, with the $\overline{SS}$ pin control enabled, (SSPCON<3:0> = 0100), the SPI module will reset if the $\overline{SS}$ pin is set to VDD.

2: If the SPI is used in Slave mode with CKE = 1, then the $\overline{SS}$ pin control must be enabled.

3: When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPCON<3:0> = 0100), the state of the $\overline{SS}$ pin can affect the state read back from the TRISC<6> bit. The peripheral OE signal from the SSP module into PORTC controls the state that is read back from the TRISC<6> bit (see **Section 11.3 "PORTC, TRISC and LATC Registers"** for information on PORTC). If Read-Modify-Write instructions, such as BSF, are performed on the TRISC register while the $\overline{SS}$ pin is high, this will cause the TRISC<6> bit to be set, thus disabling the SDO output.

FIGURE 19-2: SPI MODE TIMING, MASTER MODE

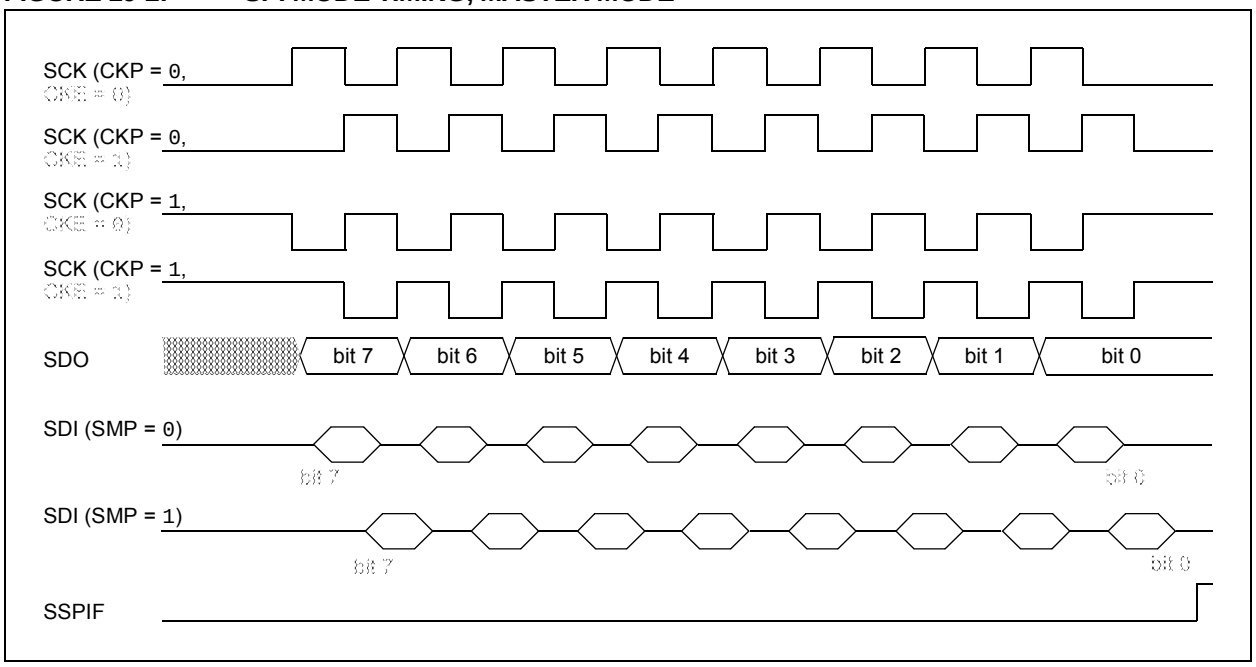
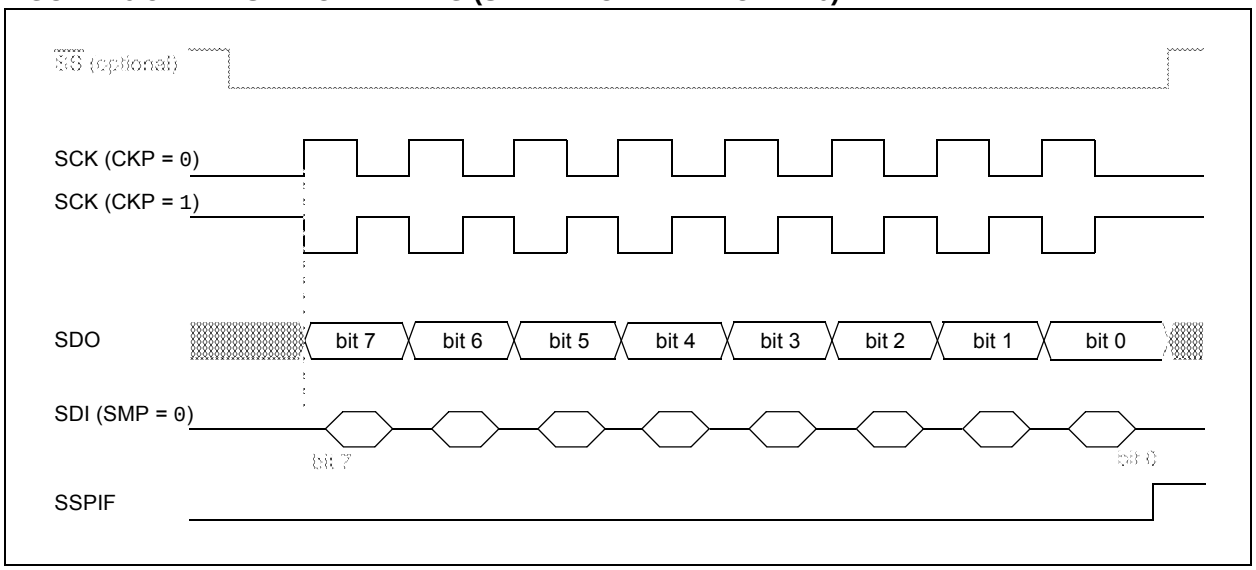


FIGURE 19-3: SPI MODE TIMING (SLAVE MODE WITH CKE = 0)



20.5.2 EUSART SYNCHRONOUS SLAVE RECEPTION

The operation of the Synchronous Master and Slave modes is identical, except in the case of Sleep, or any Idle mode and bit SREN, which is a “don’t care” in Slave mode.

If receive is enabled by setting the CREN bit prior to entering Sleep or any Idle mode, then a word may be received while in this Low-Power mode. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE enable bit is set, the interrupt generated will wake the chip from Low-Power mode. If the global interrupt is enabled, the program will branch to the interrupt vector.

To set up a Synchronous Slave Reception:

1. Enable the synchronous master serial port by setting bits, SYNC and SPEN, and clearing bit, CSRC.
2. If interrupts are desired, set enable bit, RCIE.
3. If 9-bit reception is desired, set bit, RX9.
4. To enable reception, set enable bit, CREN.
5. Flag bit, RCIF, will be set when reception is complete. An interrupt will be generated if enable bit, RCIE, was set.
6. Read the RCSTA register to get the 9th bit (if enabled) and determine if any error occurred during reception.
7. Read the 8-bit received data by reading the RCREG register.
8. If any error occurred, clear the error by clearing bit, CREN.
9. If using interrupts, ensure that the GIE and PEIE bits in the INTCON register (INTCON<7:6>) are set.

TABLE 20-10: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

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	54
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	57
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	57
IPR1	—	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	57
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	56
RCREG	EUSART Receive Register								56
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	56
BAUDCON	—	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	56
SPBRGH	EUSART Baud Rate Generator Register High Byte								56
SPBRG	EUSART Baud Rate Generator Register Low Byte								56

Legend: — = unimplemented, read as ‘0’. Shaded cells are not used for synchronous slave reception.

22.4 LVD Start-up Time

The internal reference voltage of the LVD module, specified in electrical specification Parameter D420, may be used by other internal circuitry, such as the Programmable Brown-out Reset. If the LVD, or other circuits using the voltage reference, are disabled to lower the device's current consumption, the reference voltage circuit will require time to become stable before a low-voltage condition can be reliably detected. This start-up time, TIRVST, is an interval that is independent of device clock speed. It is specified in electrical specification Parameter 36.

The LVD interrupt flag is not enabled until TIRVST has expired and a stable reference voltage is reached. For this reason, brief excursions beyond the set point may not be detected during this interval (refer to Figure 22-2).

FIGURE 22-2: LOW-VOLTAGE DETECT WAVEFORMS

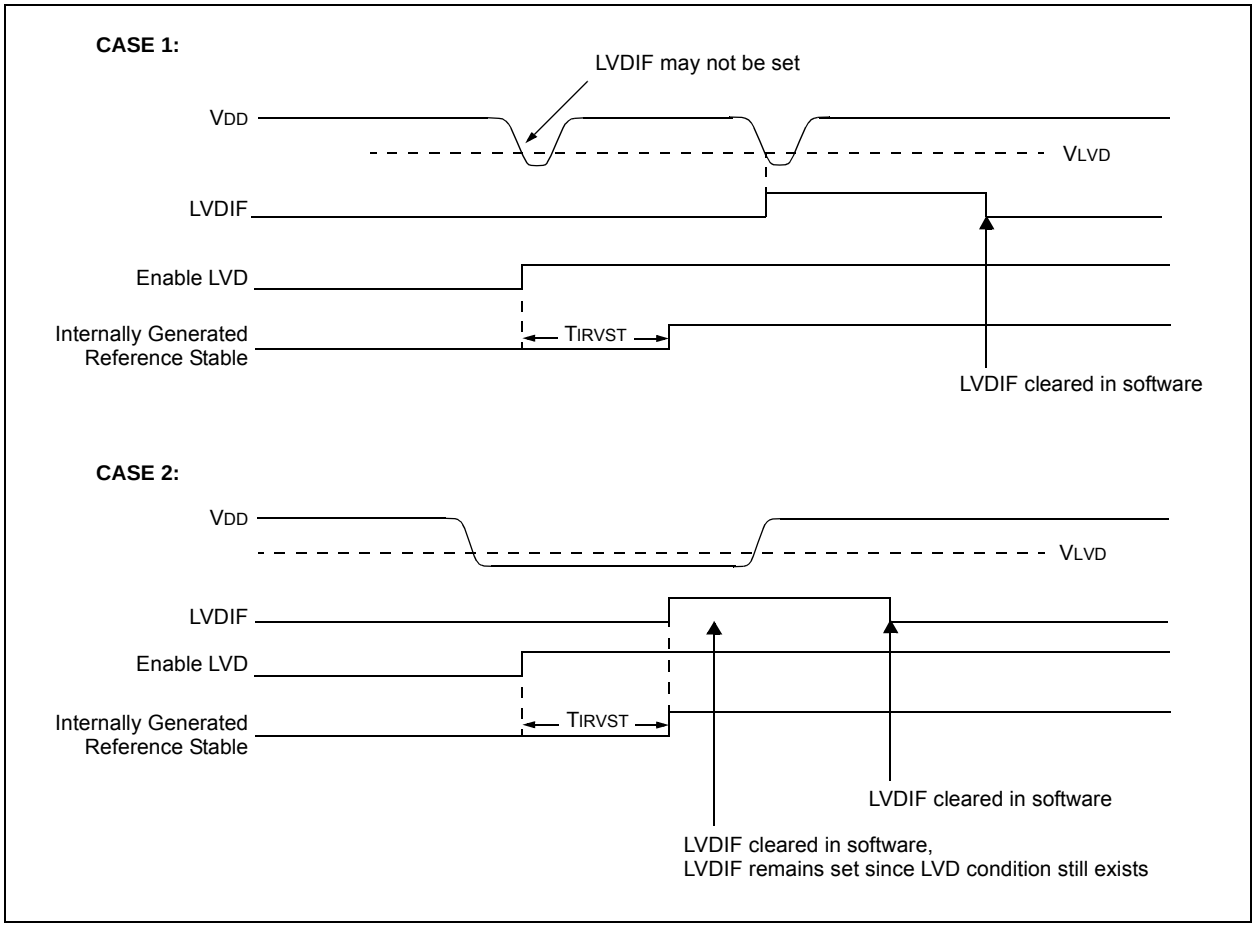
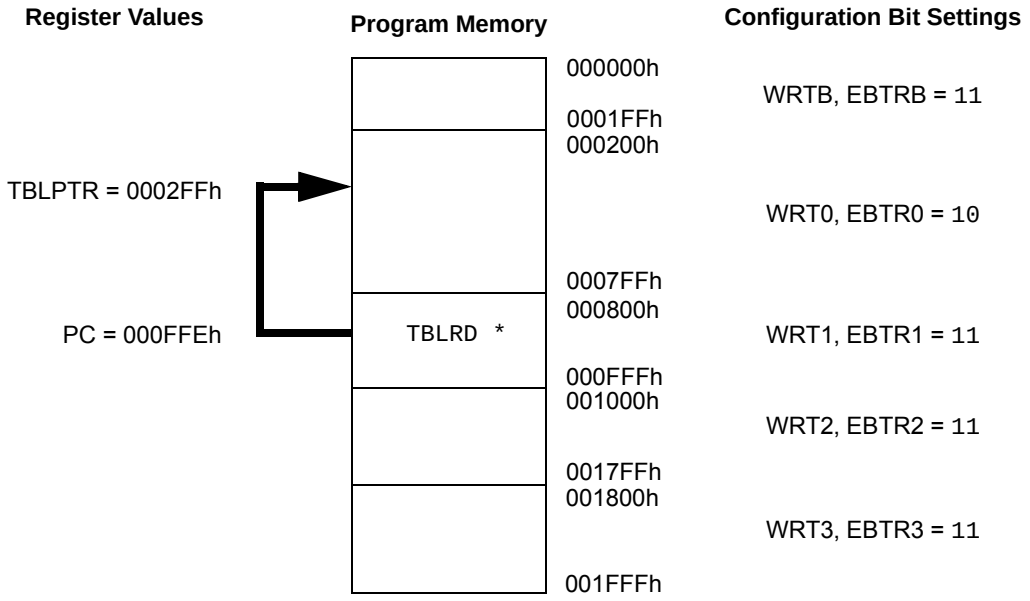
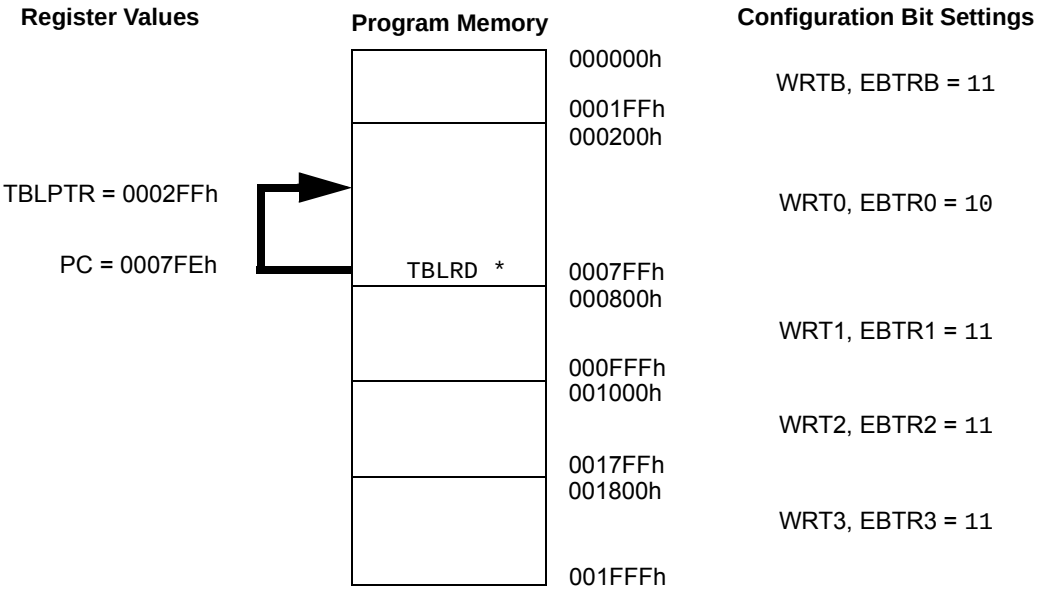


FIGURE 23-7: EXTERNAL BLOCK TABLE READ (EBTRn) DISALLOWED



Results: All table reads from external blocks to Blockn are disabled whenever EBTRn = 0.
The TABLAT register returns a value of '0'.

FIGURE 23-8: EXTERNAL BLOCK TABLE READ (EBTRn) ALLOWED



Results: Table reads permitted within Blockn, even when EBTRBn = 0.
The TABLAT register returns the value of the data at the location TBLPTR.

CLRF **Clear f**

Syntax: [*label*] CLRF f [,a]

Operands: $0 \le f \le 255$
$a \in [0,1]$

Operation: 000h → f,
1 → Z

Status Affected: Z

Encoding:

0110	101a	ffff	ffff

Description: Clears the contents of the specified register. If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' = 1, then the bank will be selected as per the BSR value.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: CLRF FLAG_REG

Before Instruction
FLAG_REG = 0x5A

After Instruction
FLAG_REG = 0x00

CLRWDT **Clear Watchdog Timer**

Syntax: [*label*] CLRWDT

Operands: None

Operation: 000h → WDT,
000h → WDT postscaler,
1 → $\overline{TO}$,
1 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Encoding:

0000	0000	0000	0100

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the postscaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	No operation

Example: CLRWDT

Before Instruction
WDT Counter = ?

After Instruction
WDT Counter = 0x00
WDT Postscaler = 0
$\overline{TO}$ = 1
$\overline{PD}$ = 1

RLNCF **Rotate Left f (No Carry)**

Syntax: [*label*] RLNCF f [,d [,a]]

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

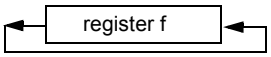
Operation: (f<n>) → dest<n + 1>,
(f<7>) → dest<0>

Status Affected: N, Z

Encoding:

0100	01da	ffff	ffff

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’. If ‘a’ is ‘0’, the Access Bank will be selected, overriding the BSR value. If ‘a’ is ‘1’, then the bank will be selected as per the BSR value.



Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register ‘f’	Process Data	Write to destination

Example: RLNCF REG

Before Instruction
REG = 1010 1011
After Instruction
REG = 0101 0111

RRCF **Rotate Right f through Carry**

Syntax: [*label*] RRCF f [,d [,a]]

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

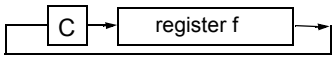
Operation: (f<n>) → dest<n – 1>,
(f<0>) → C,
(C) → dest<7>

Status Affected: C, N, Z

Encoding:

0011	00da	ffff	ffff

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’. If ‘a’ is ‘0’, the Access Bank will be selected, overriding the BSR value. If ‘a’ is ‘1’, then the bank will be selected as per the BSR value.



Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register ‘f’	Process Data	Write to destination

Example: RRCF REG, W

Before Instruction
REG = 1110 0110
C = 0
After Instruction
REG = 1110 0110
W = 0111 0011
C = 0

TBLWT Table Write

Syntax: [*label*] TBLWT (*; *+; *-; +*)

Operands: None

Operation: if TBLWT*,
(TABLAT) → Holding Register,
TBLPTR – No Change;
if TBLWT*+,
(TABLAT) → Holding Register,
(TBLPTR) + 1 → TBLPTR;
if TBLWT*-,
(TABLAT) → Holding Register,
(TBLPTR) – 1 → TBLPTR;
if TBLWT+*,
(TBLPTR) + 1 → TBLPTR,
(TABLAT) → Holding Register

Status Affected: None

Encoding:

0000	0000	0000	11nn nn = 0 * =1 *+ =2 *- =3 +*

Description: This instruction uses the 3 LSBs of TBLPTR to determine which of the 8 holding registers the TABLAT is written to. The holding registers are used to program the contents of Program Memory (P.M.). (Refer to **Section 8.0 "Flash Program Memory"** for additional details on programming Flash memory.)
The TBLPTR (a 21-bit pointer) points to each byte in the program memory. TBLPTR has a 2-Mbyte address range. The LSb of the TBLPTR selects which byte of the program memory location to access.

TBLPTR[0] = 0: Least Significant Byte of Program Memory Word

TBLPTR[0] = 1: Most Significant Byte of Program Memory Word

The TBLWT instruction can modify the value of TBLPTR as follows:

- no change
- post-increment
- post-decrement
- pre-increment

TBLWT Table Write (Continued)

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
No operation	No operation (Read TABLAT)	No operation	No operation (Write to Holding Register)

Example 1: TBLWT *+;

Before Instruction
TABLAT = 0x55
TBLPTR = 0x00A356
HOLDING REGISTER (0x00A356) = 0xFF

After Instructions (table write completion)
TABLAT = 0x55
TBLPTR = 0x00A357
HOLDING REGISTER (0x00A356) = 0x55

Example 2: TBLWT +*;

Before Instruction
TABLAT = 0x34
TBLPTR = 0x01389A
HOLDING REGISTER (0x01389A) = 0xFF
HOLDING REGISTER (0x01389B) = 0xFF

After Instruction (table write completion)
TABLAT = 0x34
TBLPTR = 0x01389B
HOLDING REGISTER (0x01389A) = 0xFF
HOLDING REGISTER (0x01389B) = 0x34

XORWF **Exclusive OR W with f**

Syntax: [*label*] XORWF f [,d [,a]]

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (W) .XOR. (f) → dest

Status Affected: N, Z

Encoding:

0001	10da	ffff	ffff

Description: 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'. If 'a' is '0', the Access Bank will be selected, overriding the BSR value. If 'a' is '1', then the bank will be selected as per the BSR value.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: XORWF REG

Before Instruction
REG = 0xAF
W = 0xB5

After Instruction
REG = 0x1A
W = 0xB5

26.2 DC Characteristics: Power-Down and Supply Current PIC18F2331/2431/4331/4431 (Industrial, Extended) PIC18LF2331/2431/4331/4431 (Industrial) (Continued)

PIC18LF2331/2431/4331/4431 (Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
PIC18F2331/2431/4331/4431 (Industrial, Extended)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD)(2,3)						
	PIC18LF2X31/4X31	5.1	9	µA	-10°C	VDD = 2.0V	FOSC = 32 kHz(4) (**SEC_RUN** mode, Timer1 as clock)
		5.8	9	µA	+25°C		
		7.9	11	µA	+70°C		
	PIC18LF2X31/4X31	7.9	12	µA	-10°C	VDD = 3.0V	
		8.9	12	µA	+25°C		
		10.5	14	µA	+70°C		
	All devices	12.5	20	µA	-10°C	VDD = 5.0V	
		16.3	20	µA	+25°C		
		18.9	25	µA	+70°C		
		150	850	µA	+125°C		
	PIC18LF2X31/4X31	9.2	15	µA	-10°C	VDD = 2.0V	FOSC = 32 kHz(4) (**SEC_IDLE** mode, Timer1 as clock)
		9.6	15	µA	+25°C		
		12.7	18	µA	+70°C		
	PIC18LF2X31/4X31	22.0	30	µA	-10°C	VDD = 3.0V	
		21.0	30	µA	+25°C		
		20.0	35	µA	+70°C		
	All devices	30	80	µA	-10°C	VDD = 5.0V	
		45	80	µA	+25°C		
		45	85	µA	+70°C		
		250	850	µA	+125°C		

Legend: Shading of rows is to assist in readability of the table.

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS, and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.
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 enabled/disabled as specified.

3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be estimated by the formula: Ir = VDD/2REXT (mA) with REXT in kΩ.

4: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

TABLE 26-20: A/D CONVERTER CHARACTERISTICS

PIC18LF2331/2431/4331/4431 (Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial				
PIC18F2331/2431/4331/4431 (Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Device Supply							
	AVDD	Analog VDD Supply	VDD – 0.3	—	VDD + 0.3	V	
	AVSS	Analog VSS Supply	VSS – 0.3	—	VSS + 0.3	V	
	IAD	Module Current (during conversion)	— —	500 250	— —	μA μA	VDD = 5V VDD = 2.5V
	IADO	Module Current Off	—	—	1.0	μA	
AC Timing Parameters							
A10	FTHR	Throughput Rate	— —	— —	200 75	ksps ksps	VDD = 5V, single channel VDD < 3V, single channel
A11	TAD	A/D Clock Period	385 1000	— —	20,000 20,000	ns ns	VDD = 5V VDD = 3V
A12	TRC	A/D Internal RC Oscillator Period	— — —	500 750 10000	1500 2250 20000	ns ns ns	PIC18F parts PIC18LF parts AVDD < 3.0V
A13	TCNV	Conversion Time(1)	12	12	12	TAD	
A14	TACQ	Acquisition Time(2)	2(2)	—	—	TAD	
A16	TTC	Conversion Start from External	1/4 TCY	—	—		
Reference Inputs							
A20	VREF	Reference Voltage for 10-Bit Resolution (VREF+ – VREF-)	1.5 1.8	— —	AVDD – AVSS AVDD – AVSS	V V	VDD ≥ 3V VDD < 3V
A21	VREFH	Reference Voltage High (AVDD or VREF+)	1.5V	—	AVDD	V	VDD ≥ 3V
A22	VREFL	Reference Voltage Low (AVSS or VREF-)	AVSS	—	VREFH – 1.5V	V	
A23	IREF	Reference Current	— —	150 μA 75 μA	— —		VDD = 5V VDD = 2.5V
Analog Input Characteristics							
A26	VAIN	Input Voltage(3)	AVSS – 0.3	—	AVDD + 0.3	V	
A30	ZAIN	Recommended Impedance of Analog Voltage Source	—	—	2.5	kΩ	
A31	ZCHIN	Analog Channel Input Impedance	—	—	10.0	kΩ	VDD = 3.0V
DC Performance							
A41	NR	Resolution	10 bits			—	
A42	EIL	Integral Nonlinearity	—	—	<±1	LSb	VDD ≥ 3.0V VREFH ≥ 3.0V
A43	EIL	Differential Nonlinearity	—	—	<±1	LSb	VDD ≥ 3.0V VREFH ≥ 3.0V
A45	EOFF	Offset Error	—	±0.5	<±1.5	LSb	VDD ≥ 3.0V VREFH ≥ 3.0V
A46	EGA	Gain Error	—	±0.5	<±1.5	LSb	VDD ≥ 3.0V VREFH ≥ 3.0V
A47	—	Monotonicity(4)	guaranteed			—	VDD ≥ 3.0V VREFH ≥ 3.0V

Note 1: Conversion time does not include acquisition time. See **Section 21.0 "10-Bit High-Speed Analog-to-Digital Converter (A/D) Module"** for a full discussion of acquisition time requirements.

2: In Sequential modes, TACQ should be 12 TAD or greater.

3: For VDD < 2.7V and temperature below 0°C, VAIN should be limited to range < VDD/2.

4: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.