



Welcome to [E-XFL.COM](https://www.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](#)"

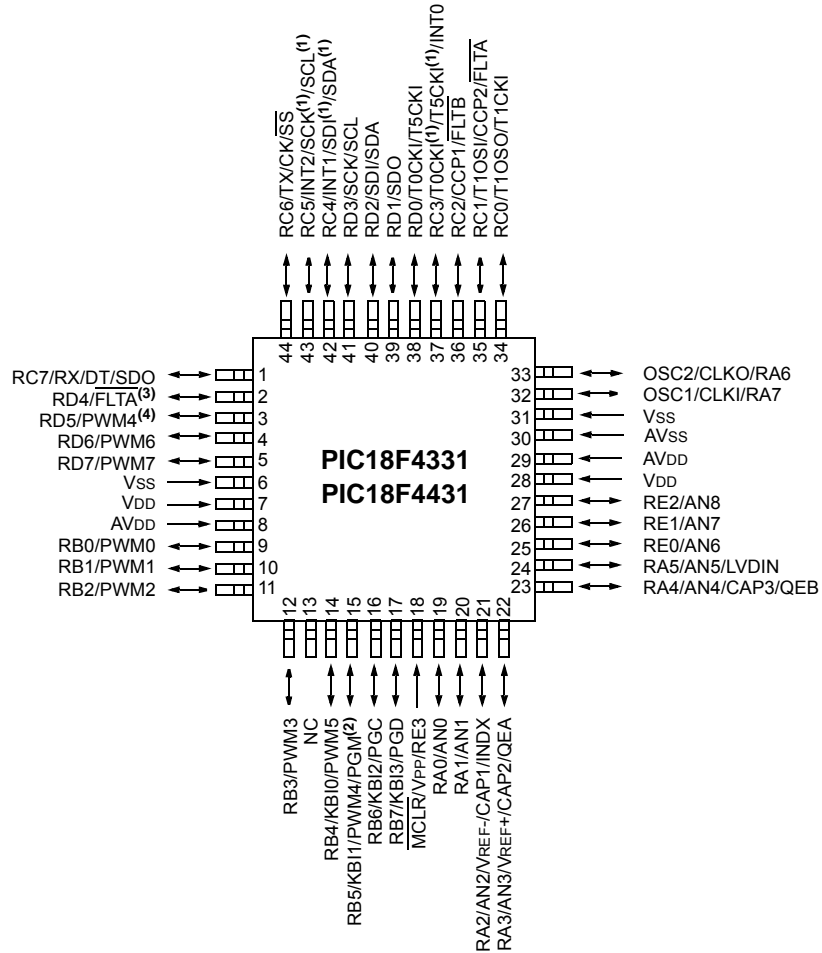
Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, Power Control PWM, QEI, POR, PWM, WDT
Number of I/O	36
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)	2V ~ 5.5V
Data Converters	A/D 9x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf4431-i-pt

PIC18F2331/2431/4331/4431

Pin Diagrams (Continued)

44-Pin QFN⁽²⁾



Note 1: RC3 is the alternate pin for T0CKI/T5CKI; RC4 is the alternate pin for SDI/SDA; RC5 is the alternate pin for SCK/SCL.

Note 2: For the QFN package, it is recommended that the bottom pad be connected to Vss.

Note 3: RD4 is the alternate pin for FLTA.

Note 4: RD5 is the alternate pin for PWM4.

PIC18F2331/2431/4331/4431

Table of Contents

1.0	Device Overview	11
2.0	Guidelines for Getting Started with PIC18F Microcontrollers	25
3.0	Oscillator Configurations	29
4.0	Power-Managed Modes	39
5.0	Reset	47
6.0	Memory Organization	61
7.0	Data EEPROM Memory	79
8.0	Flash Program Memory	85
9.0	8 x 8 Hardware Multiplier	95
10.0	Interrupts	97
11.0	I/O Ports	113
12.0	Timer0 Module	127
13.0	Timer1 Module	131
14.0	Timer2 Module	136
15.0	Timer5 Module	139
16.0	Capture/Compare/PWM (CCP) Modules	145
17.0	Motion Feedback Module	151
18.0	Power Control PWM Module	173
19.0	Synchronous Serial Port (SSP) Module	205
20.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART)	217
21.0	10-Bit High-Speed Analog-to-Digital Converter (A/D) Module	239
22.0	Low-Voltage Detect (LVD)	257
23.0	Special Features of the CPU	263
24.0	Instruction Set Summary	283
25.0	Development Support	325
26.0	Electrical Characteristics	329
27.0	Packaging Information	363
	Appendix A: Revision History	375
	Appendix B: Device Differences	375
	Appendix C: Conversion Considerations	376
	Appendix D: Migration from Baseline to Enhanced Devices	376
	Appendix E: Migration From Mid-Range to Enhanced Devices	377
	Appendix F: Migration From High-End to Enhanced Devices	377
	INDEX	379
	The Microchip Web Site	389
	Customer Change Notification Service	389
	Customer Support	389
	Reader Response	390
	Product Identification System	391

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@microchip.com or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at www.microchip.com to receive the most current information on all of our products.

2.5 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 3.0 “Oscillator Configurations”** for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

Layout suggestions are shown in Figure 2-4. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins and other signals in close proximity to the oscillator are benign (i.e., free of high frequencies, short rise and fall times, and other similar noise).

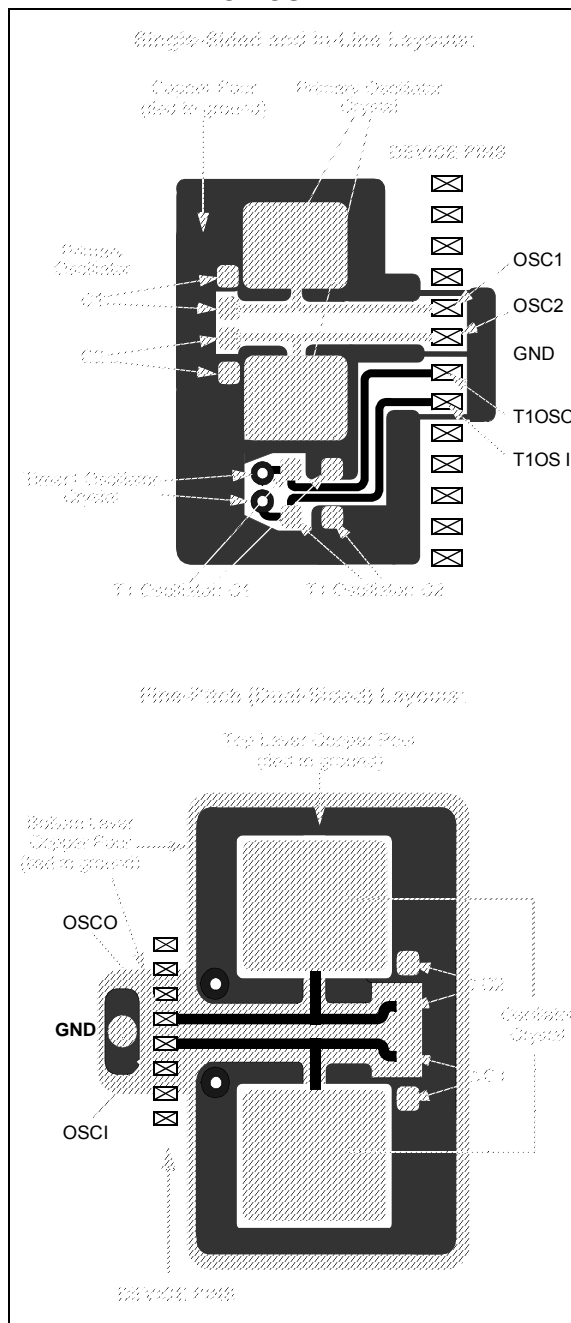
For additional information and design guidance on oscillator circuits, please refer to these Microchip Application Notes, available at the corporate web site (www.microchip.com):

- AN826, “Crystal Oscillator Basics and Crystal Selection for rPIC™ and PICmicro® Devices”
- AN849, “Basic PICmicro® Oscillator Design”
- AN943, “Practical PICmicro® Oscillator Analysis and Design”
- AN949, “Making Your Oscillator Work”

2.6 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 kΩ to 10 kΩ resistor to V_{SS} on unused pins and drive the output to logic low.

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



PIC18F2331/2431/4331/4431

REGISTER 3-2: OSCCON: OSCILLATOR CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R ⁽¹⁾	R-0	R/W-0	R/W-0
IDLEN	IRCF2	IRCF1	IRCF0	OSTS	IOFS	SCS1	SCS0
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 **IDLEN:** Idle Enable bit

1 = Idle mode enabled; CPU core is not clocked in power-managed modes

0 = Run mode enabled; CPU core is clocked in power-managed modes

bit 6-4 **IRCF<2:0>:** Internal Oscillator Frequency Select bits

111 = 8 MHz (8 MHz source drives clock directly)

110 = 4 MHz (default)

101 = 2 MHz

100 = 1 MHz

011 = 500 kHz

010 = 250 kHz

001 = 125 kHz

000 = 31 kHz (INTRC source drives clock directly)⁽²⁾

bit 3 **OSTS:** Oscillator Start-up Timer Time-out Status bit⁽¹⁾

1 = Oscillator Start-up Timer time-out has expired; primary oscillator is running

0 = Oscillator Start-up Timer time-out is running; primary oscillator is not ready

bit 2 **IOFS:** INTOSC Frequency Stable bit

1 = INTOSC frequency is stable

0 = INTOSC frequency is not stable

bit 1-0 **SCS<1:0>:** System Clock Select bits

1x = Internal oscillator block

01 = Secondary (Timer1) oscillator

00 = Primary oscillator

Note 1: Depends on the state of the IESO bit in Configuration Register 1H.

2: Default output frequency of INTOSC on Reset.

FIGURE 4-1: TRANSITION TIMING FOR ENTRY TO SEC_RUN MODE

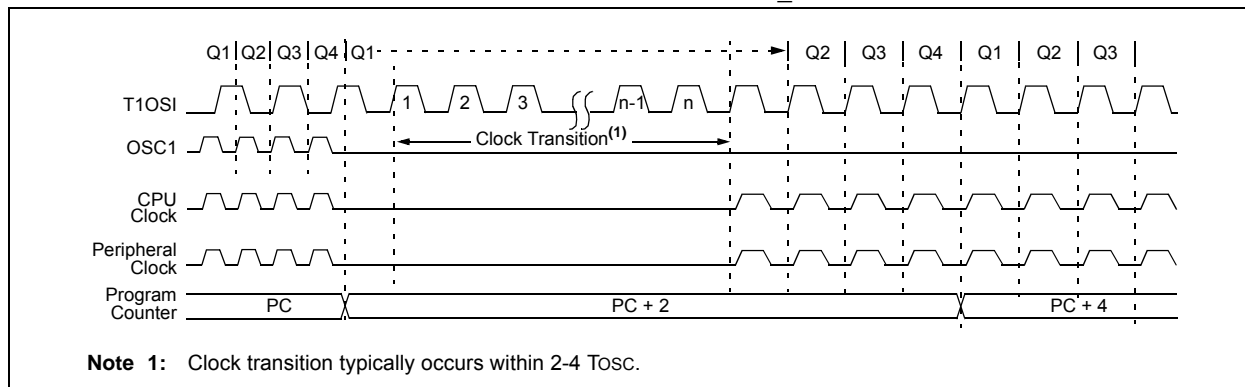
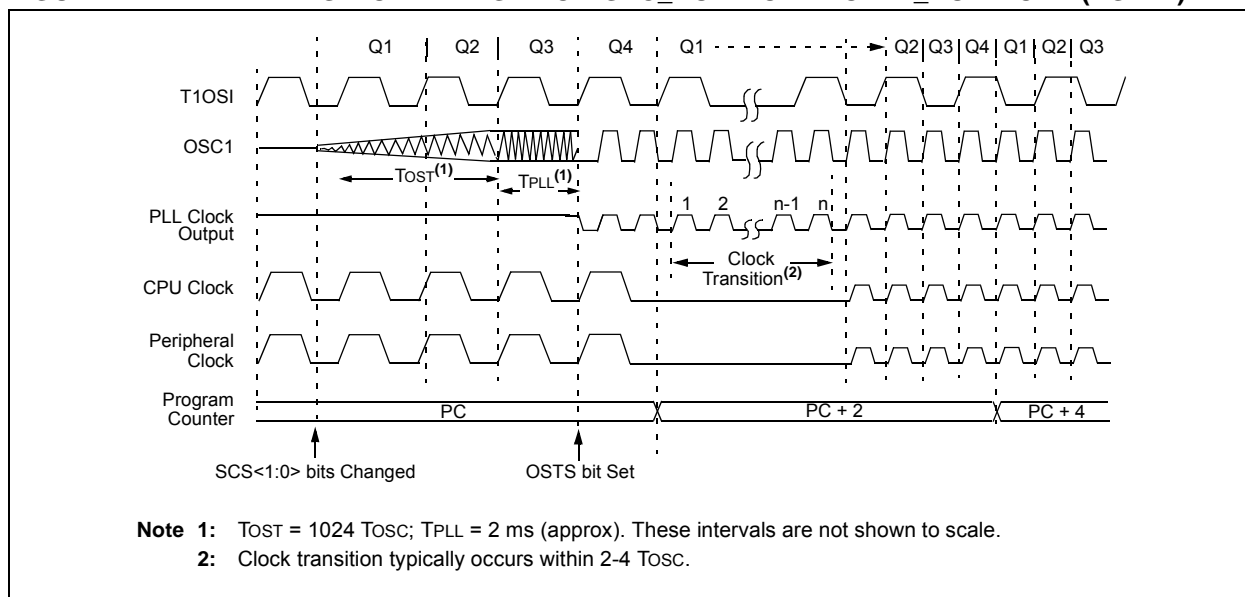


FIGURE 4-2: TRANSITION TIMING FROM SEC_RUN MODE TO PRI_RUN MODE (HSPLL)



4.2.3 RC_RUN MODE

In RC_RUN mode, the CPU and peripherals are clocked from the internal oscillator block using the INTOSC multiplexer. In this mode, the primary clock is shut down. When using the INTRC source, this mode provides the best power conservation of all the Run modes, while still executing code. It works well for user applications which are not highly timing-sensitive or do not require high-speed clocks at all times.

If the primary clock source is the internal oscillator block (either INTRC or INTOSC), there are no distinguishable differences between PRI_RUN and RC_RUN modes during execution. However, a clock switch delay will occur during entry to and exit from RC_RUN mode. Therefore, if the primary clock source is the internal oscillator block, the use of RC_RUN mode is not recommended.

This mode is entered by setting the SCS1 bit to '1'. Although it is ignored, it is recommended that the SCS0 bit also be cleared; this is to maintain software compatibility with future devices. When the clock source is switched to the INTOSC multiplexer (see Figure 4-3), the primary oscillator is shut down and the OSTS bit is cleared. The IRCF bits may be modified at any time to immediately change the clock speed.

Note: Caution should be used when modifying a single IRCF bit. If V_{DD} is less than 3V, it is possible to select a higher clock speed than is supported by the low V_{DD}. Improper device operation may result if the V_{DD}/F_{OSC} specifications are violated.

6.1.1 PROGRAM COUNTER

The Program Counter (PC) specifies the address of the instruction to fetch for execution. The PC is 21 bits wide and contained in three 8-bit registers. The low byte, known as the PCL register, is both readable and writable. The high byte (PCH register) contains the PC<15:8> bits and is not directly readable or writable.

Updates to the PCH register are performed through the PCLATH register. The upper byte is the PCU register and contains the bits, PC<20:16>. This register is also not directly readable or writable. Updates to the PCU register are performed through the PCLATU register.

The contents of PCLATH and PCLATU are transferred to the program counter by any operation that writes to the PCL. Similarly, the upper two bytes of the program counter are transferred to PCLATH and PCLATU by an operation that reads PCL. This is useful for computed offsets to the PC (see **Section 6.1.4.1 “Computed GOTO”**).

The PC addresses bytes in the program memory. To prevent the PC from becoming misaligned with word instructions, the Least Significant bit of the PCL is fixed to a value of '0'. The PC increments by two to address sequential instructions in the program memory.

The CALL, RCALL, GOTO and program branch instructions write to the program counter directly. For these instructions, the contents of PCLATH and PCLATU are not transferred to the program counter.

6.1.2 RETURN ADDRESS STACK

The return address stack allows any combination of up to 31 program calls and interrupts to occur. The PC (Program Counter) is pushed onto the stack when a CALL or RCALL instruction is executed, or an interrupt is Acknowledged. The PC value is pulled off the stack on a RETURN, RETLW or a RETFIE instruction. PCLATU and PCLATH are not affected by any of the RETURN or CALL instructions.

The stack operates as a 31-word by 21-bit RAM and a 5-bit Stack Pointer, with the Stack Pointer initialized to 00000b after all Resets. There is no RAM associated with Stack Pointer, 00000b. This is only a Reset value. During a CALL type instruction, causing a push onto the stack, the Stack Pointer is first incremented and the RAM location pointed to by the Stack Pointer is written with the contents of the PC (already pointing to the instruction following the CALL). During a RETURN type instruction, causing a pop from the stack, the contents of the RAM location pointed to by the STKPTR are transferred to the PC and then the Stack Pointer is decremented.

The stack space is not part of either program or data space. The Stack Pointer is readable and writable, and the address on the top of the stack is readable and writable through the Top-of-Stack (TOS) Special Function Registers. Data can also be pushed to, or popped from, the stack using the Top-of-Stack SFRs. Status bits indicate if the stack is full, has overflowed or underflowed.

6.1.2.1 Top-of-Stack Access

The top of the stack is readable and writable. Three register locations, TOSU, TOSH and TOSL, hold the contents of the stack location pointed to by the STKPTR register (Figure 6-3). This allows users to implement a software stack if necessary. After a CALL, RCALL or interrupt, the software can read the pushed value by reading the TOSU, TOSH and TOSL registers. These values can be placed on a user-defined software stack. At return time, the software can replace the TOSU, TOSH and TOSL and do a return.

The user must disable the global interrupt enable bits while accessing the stack to prevent inadvertent stack corruption.

6.1.2.2 Return Stack Pointer (STKPTR)

The STKPTR register (Register 6-1) contains the Stack Pointer value, the STKFUL (Stack Full) status bit and the STKUNF (Stack Underflow) status bits. The value of the Stack Pointer can be 0 through 31. The Stack Pointer increments before values are pushed onto the stack and decrements after values are popped off the stack. At Reset, the Stack Pointer value will be zero. The user may read and write the Stack Pointer value. This feature can be used by a Real-Time Operating System (RTOS) for return stack maintenance.

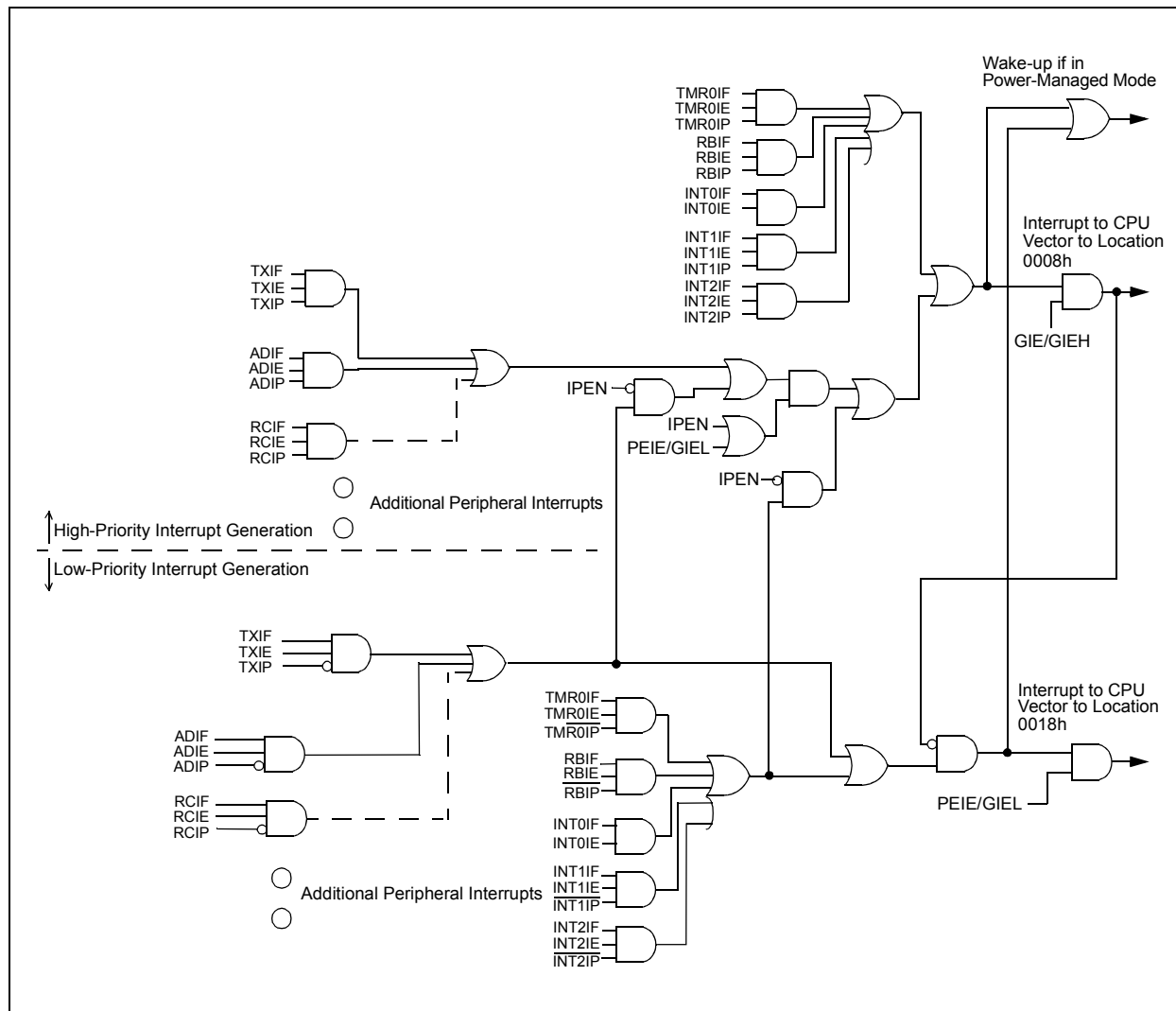
After the PC is pushed onto the stack 31 times (without popping any values off the stack), the STKFUL bit is set. The STKFUL bit is cleared by software or by a POR.

The action that takes place when the stack becomes full depends on the state of the STVREN (Stack Overflow Reset Enable) Configuration bit. (Refer to **Section 23.1 “Configuration Bits”** for a description of the device Configuration bits.) If STVREN is set (default), the 31st push will push the (PC + 2) value onto the stack, set the STKFUL bit and reset the device. The STKFUL bit will remain set and the Stack Pointer will be set to zero.

If STVREN is cleared, the STKFUL bit will be set on the 31st push and the Stack Pointer will increment to 31. Any additional pushes will not overwrite the 31st push and STKPTR will remain at 31.

PIC18F2331/2431/4331/4431

FIGURE 10-1: INTERRUPT LOGIC



PIC18F2331/2431/4331/4431

FIGURE 12-1: TIMER0 BLOCK DIAGRAM (8-BIT MODE)

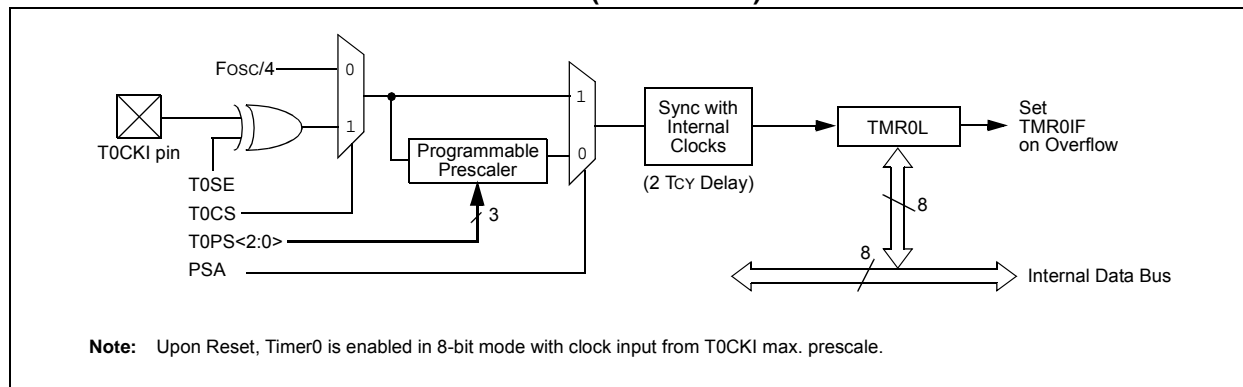
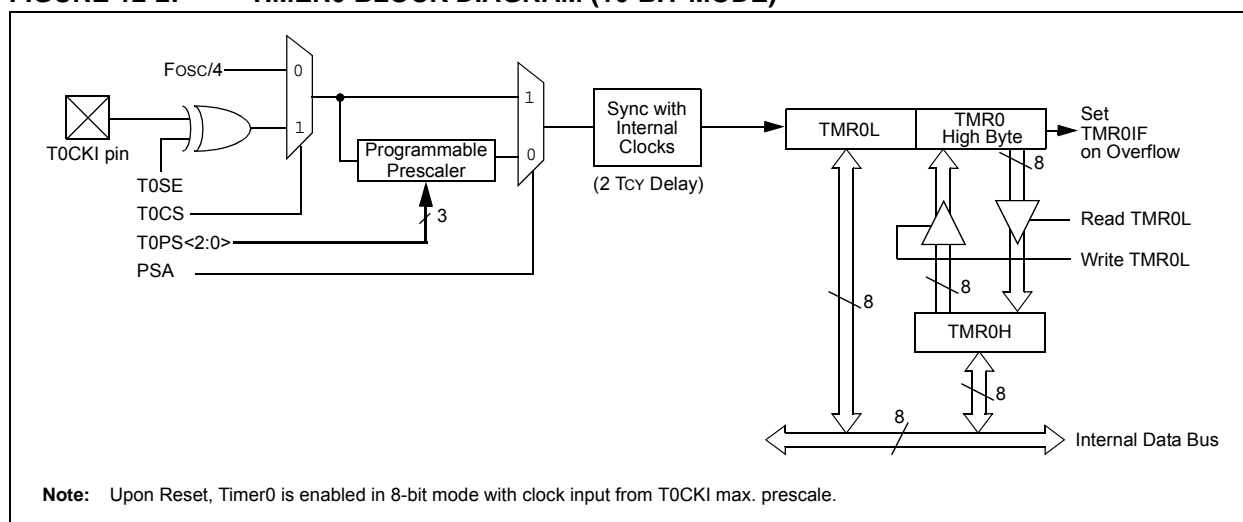
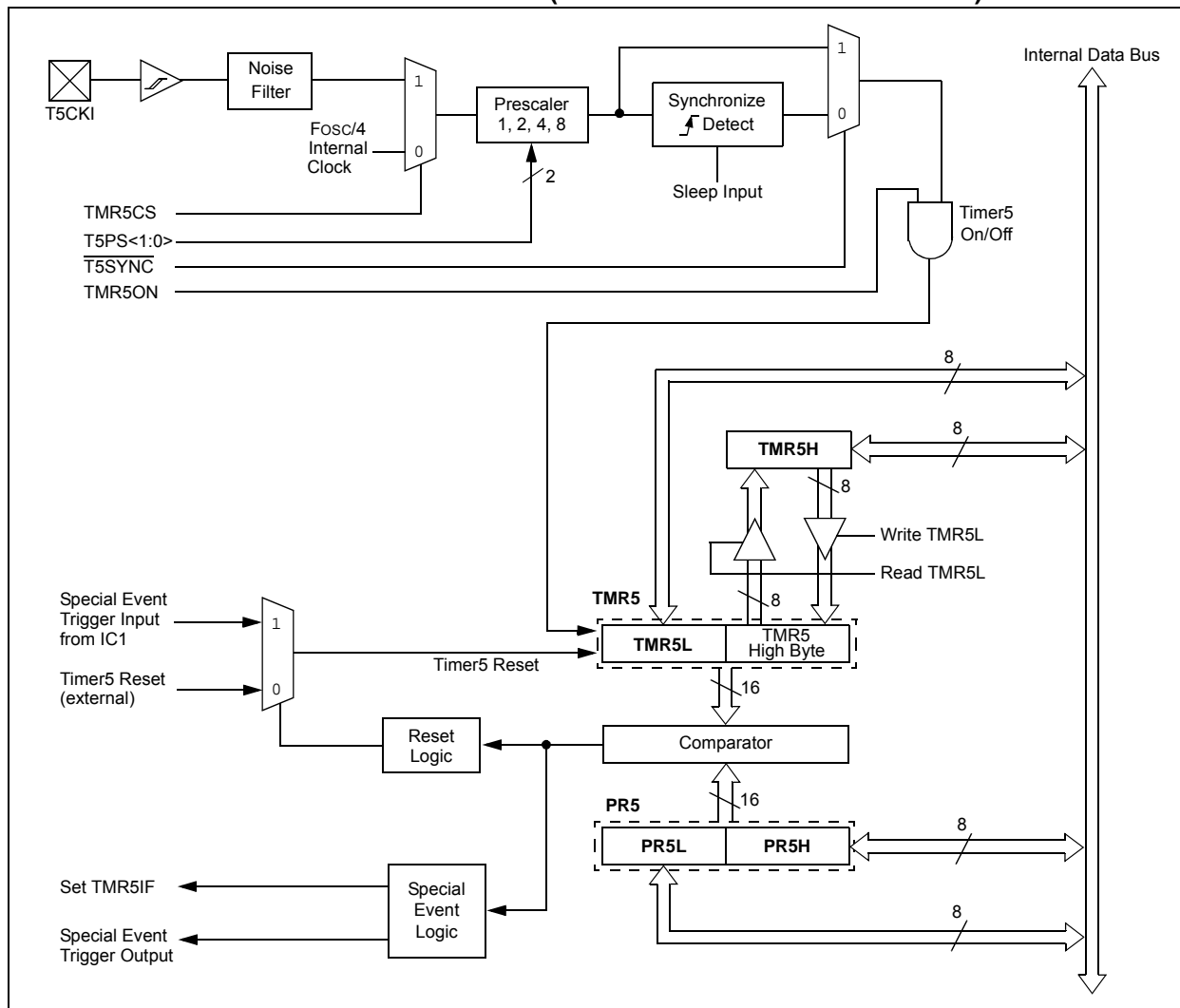


FIGURE 12-2: TIMER0 BLOCK DIAGRAM (16-BIT MODE)



PIC18F2331/2431/4331/4431

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.

PIC18F2331/2431/4331/4431

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 ($F_{osc}/4$) 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.

PIC18F2331/2431/4331/4431

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 I²C™ 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 I²C 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 I²C 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 I²C™ mode only.

PIC18F2331/2431/4331/4431

TABLE 20-5: REGISTERS ASSOCIATED WITH ASYNCHRONOUS 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	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
TXREG	EUSART Transmit 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 asynchronous transmission.

PIC18F2331/2431/4331/4431

REGISTER 21-5: ADCHS: A/D CHANNEL SELECT 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
GDSEL1	GDSEL0	GBSEL1	GBSEL0	GCSEL1	GCSEL0	GASEL1	GASEL0
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 7-6 **GDSEL<1:0>**: Group D Select bits

S/H-2 positive input.

00 = AN3

01 = AN7⁽¹⁾

1x = Reserved

bit 5-4 **GBSEL<1:0>**: Group B Select bits

S/H-2 positive input.

00 = AN1

01 = AN5⁽¹⁾

1x = Reserved

bit 3-2 **GCSEL<1:0>**: Group C Select bits

S/H-1 positive input.

00 = AN2

01 = AN6⁽¹⁾

1x = Reserved

bit 1-0 **GASEL<1:0>**: Group A Select bits

S/H-1 positive input.

00 = AN0

01 = AN4

10 = AN8⁽¹⁾

11 = Reserved

Note 1: AN5 through AN8 are available only in PIC18F4331/4431 devices.

PIC18F2331/2431/4331/4431

22.0 LOW-VOLTAGE DETECT (LVD)

PIC18F2331/2431/4331/4431 devices have a Low-Voltage Detect module (LVD), a programmable circuit that enables the user to specify a device voltage trip point. If the device experiences an excursion below the trip point, an interrupt flag is set. If the interrupt is enabled, the program execution will branch to the interrupt vector address and the software can then respond to the interrupt.

The Low-Voltage Detect Control register (Register 22-1) completely controls the operation of the LVD module. This allows the circuitry to be “turned off” by the user under software control, which minimizes the current consumption for the device.

The block diagram for the LVD module is shown in Figure 22-1.

The module is enabled by setting the LVDEN bit, but the circuitry requires some time to stabilize each time that it is enabled. The IRVST bit is a read-only bit used to indicate when the circuit is stable. The module can only generate an interrupt after the circuit is stable and the IRVST bit is set. The module monitors for drops in VDD below a predetermined set point.

REGISTER 22-1: LVDCON: LOW-VOLTAGE DETECT CONTROL REGISTER

U-0	U-0	R-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
—	—	IRVST	LVDEN	LVDL3 ⁽¹⁾	LVDL2 ⁽¹⁾	LVDL1 ⁽¹⁾	LVDL0 ⁽¹⁾
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 7-6 **Unimplemented:** Read as ‘0’

bit 5 **IRVST:** Internal Reference Voltage Stable Flag bit

1 = Indicates that the Low-Voltage Detect logic will generate the interrupt flag at the specified voltage range

0 = Indicates that the Low-Voltage Detect logic will not generate the interrupt flag at the specified voltage range and the LVD interrupt should not be enabled

bit 4 **LVDEN:** Low-Voltage Detect Power Enable bit

1 = Enables LVD, powers up LVD circuit

0 = Disables LVD, powers down LVD circuit

bit 3-0 **LVDL<3:0>:** Low-Voltage Detection Limit bits⁽¹⁾

1111 = External analog input is used (input comes from the LVDIN pin)

1110 = Maximum setting

•

•

•

0010 = Minimum setting

0001 = Reserved

0000 = Reserved

Note 1: LVDL<3:0> bit modes, which result in a trip point below the valid operating voltage of the device, are not tested.

PIC18F2331/2431/4331/4431

TABLE 24-2: PIC18FXXXX INSTRUCTION SET (CONTINUED)

Mnemonic, Operands	Description	Cycles	16-Bit Instruction Word				Status Affected	Notes
			MSb		LSb			
LITERAL OPERATIONS								
ADDLW k	Add Literal and WREG	1	0000	1111	kkkk	kkkk	C, DC, Z, OV, N	
ANDLW k	AND Literal with WREG	1	0000	1011	kkkk	kkkk	Z, N	
IORLW k	Inclusive OR Literal with WREG	1	0000	1001	kkkk	kkkk	Z, N	
LFSR f, k	Load Literal (12-bit) 2nd word to FSRx 1st word	2	1110	1110	00ff	kkkk	None	
			1111	0000	kkkk	kkkk		
MOVLB k	Move Literal to BSR<3:0>	1	0000	0001	0000	kkkk	None	
MOVLW k	Move Literal to WREG	1	0000	1110	kkkk	kkkk	None	
MULLW k	Multiply Literal with WREG	1	0000	1101	kkkk	kkkk	None	
RETLW k	Return with Literal in WREG	2	0000	1100	kkkk	kkkk	None	
SUBLW k	Subtract WREG from Literal	1	0000	1000	kkkk	kkkk	C, DC, Z, OV, N	
XORLW k	Exclusive OR Literal with WREG	1	0000	1010	kkkk	kkkk	Z, N	
DATA MEMORY ↔ PROGRAM MEMORY OPERATIONS								
TBLRD*	Table Read	2	0000	0000	0000	1000	None	
TBLRD*+	Table Read with Post-Increment		0000	0000	0000	1001	None	
TBLRD*~	Table Read with Post-Decrement		0000	0000	0000	1010	None	
TBLRD*~	Table Read with Pre-Increment		0000	0000	0000	1011	None	
TBLWT*	Table Write	2 (5)	0000	0000	0000	1100	None	
TBLWT*+	Table Write with Post-Increment		0000	0000	0000	1101	None	
TBLWT*~	Table Write with Post-Decrement		0000	0000	0000	1110	None	
TBLWT*~	Table Write with Pre-Increment		0000	0000	0000	1111	None	

- Note 1:** When a PORT register is modified as a function of itself (e.g., `MOVF PORTB, 1, 0`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as an input and is driven low by an external device, the data will be written back with a '0'.
- 2:** If this instruction is executed on the TMR0 register (and, where applicable, $d = 1$), the prescaler will be cleared if assigned.
- 3:** If the Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a `NOP`.
- 4:** Some instructions are 2-word instructions. The second word of these instructions will be executed as a `NOP` unless the first word of the instruction retrieves the information embedded in these 16 bits. This ensures that all program memory locations have a valid instruction.
- 5:** If the table write starts the write cycle to internal memory, the write will continue until terminated.

PIC18F2331/2431/4331/4431

FIGURE 26-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

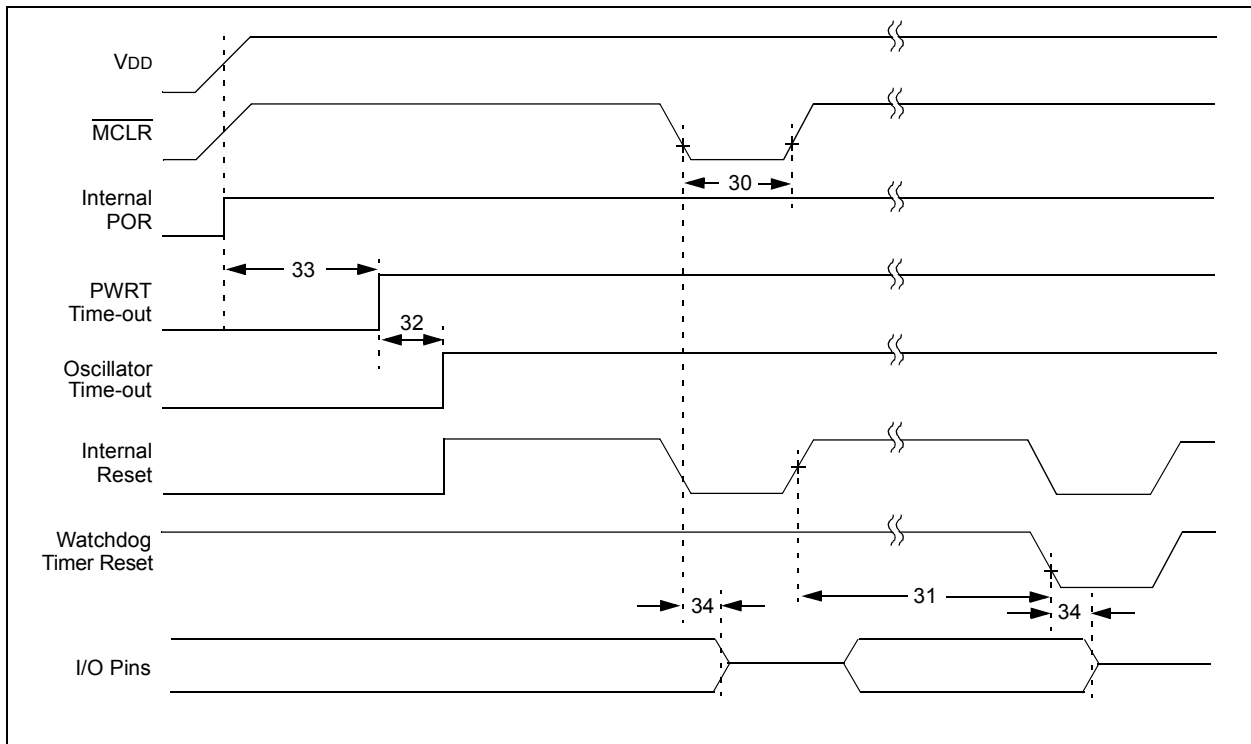
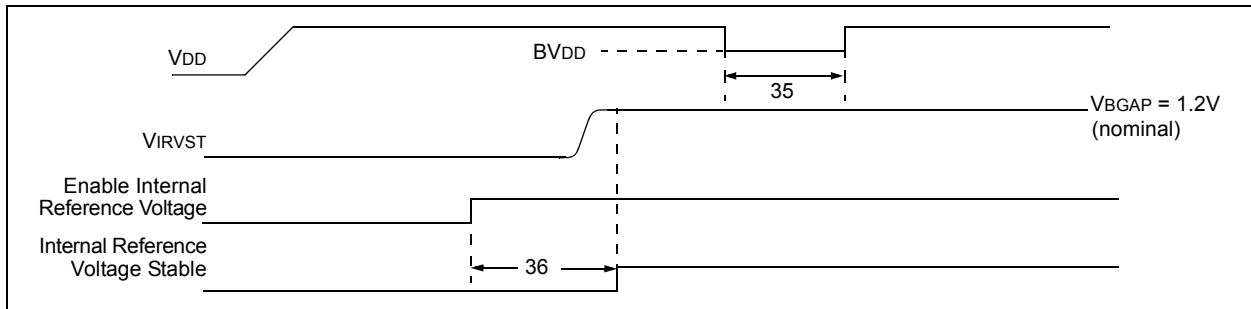


FIGURE 26-8: BROWN-OUT RESET TIMING



PIC18F2331/2431/4331/4431

NOTES:

APPENDIX E: MIGRATION FROM MID-RANGE TO ENHANCED DEVICES

A detailed discussion of the differences between the mid-range MCU devices (i.e., PIC16CXXX) and the enhanced devices (i.e., PIC18FXXX) is provided in AN716, "Migrating Designs from PIC16C74A/74B to PIC18F442." The changes discussed, while device-specific, are generally applicable to all mid-range to enhanced device migrations.

This Application Note is available on Microchip's web site: www.Microchip.com.

APPENDIX F: MIGRATION FROM HIGH-END TO ENHANCED DEVICES

A detailed discussion of the migration pathway and differences between the high-end MCU devices (i.e., PIC17CXXX) and the enhanced devices (i.e., PIC18FXXX) is provided in AN726, "PIC17CXXX to PIC18FXXX Migration."

This Application Note is available on Microchip's web site: www.Microchip.com.

PIC18F2331/2431/4331/4431

CCP2	145
CCPR2H Register	145
CCPR2L Register	145
Compare Mode. See Compare.	
Timer Resources	145
CKE Bit	206
CKP Bit	207
Clock Sources	34
Effects of Power-Managed Modes	37
Selection Using OSCCON Register	34
Clocking Scheme/Instruction Cycle	65
CLRF	299
CLRWDT	299
Code Examples	
Changing Between Capture Prescalers	146
Computed GOTO Using an Offset Value	64
Data EEPROM Read	81
Data EEPROM Refresh Routine	82
Data EEPROM Write	81
Erasing a Flash Program Memory Row	90
Fast Register Stack	64
How to Clear RAM (Bank 1) Using	
Indirect Addressing	75
Implementing a Real-Time Clock Using a	
Timer1 Interrupt Service	135
Initializing PORTA	113
Initializing PORTB	116
Initializing PORTC	119
Initializing PORTD	122
Initializing PORTE	124
Reading a Flash Program Memory Word	89
Saving STATUS, WREG and BSR	
Registers in RAM	112
Writing to Flash Program Memory	93–94
16 x 16 Signed Multiply Routine	96
16 x 16 Unsigned Multiply Routine	96
8 x 8 Signed Multiply Routine	95
8 x 8 Unsigned Multiply Routine	95
Code Protection	263, 279
Associated Registers	279
Data EEPROM	282
Program Memory	280
COMF	300
Compare (CCP Module)	147
Associated Registers	148
CCP Pin Configuration	147
CCPR1 Register	147
CCPR2 Register	147
Software Interrupt Mode	147
Special Event Trigger	147
Timer1 Mode Selection	147
Configuration Bits	263
Configuration Register Protection	282
Conversion Considerations	376
CPFSEQ	300
CPFSGT	301
CPFSLT	301
Crystal Oscillator/Ceramic Resonators	29
Customer Change Notification Service	387
Customer Notification Service	387
Customer Support	387

D

D/A Bit	206
Data Addressing Modes	75
Direct	75
Indirect	75
Inherent and Literal	75
Data EEPROM Memory	79
Associated Registers	83
EEADR Register	79
EECON1 and EECON2 Registers	79
Operation During Code-Protect	82
Protection Against Spurious Write	81
Reading	81
Using	82
Write Verify	81
Writing	81
Data Memory	67
Access Bank	68
Bank Select Register (BSR)	68
General Purpose Register (GPR) File	68
Map for PIC18F2331/2431/4331/4431	67
Special Function Registers (SFRs)	69
DAW	302
DC Characteristics	339
Power-Down and Supply Current	332
Supply Voltage	331
DCFSNZ	303
DECF	302
DECFSZ	303
Development Support	325
Device Differences	375
Device Overview	11
Features (table)	13
New Core Features	11
Other Special Features	12
Device Reset Timers	
Oscillator Start-up Timer (OST)	50
PLL Lock Time-out	50
Power-up Timer (PWRT)	50
Time-out Sequence	50
Direct Addressing	76

E

Electrical Characteristics	329
Enhanced Universal Synchronous Asynchronous	
Receiver Transmitter (EUSART)	217
Equations	
A/D Acquisition Time	249
Conversion Time for Multi-Channel Modes	254
Minimum A/D Holding Capacitor Charging Time	249
PWM Period for Free-Running Mode	185
PWM Period for Up/Down Count Mode	185
PWM Resolution	185
16 x 16 Signed Multiplication Algorithm	96
16 x 16 Unsigned Multiplication Algorithm	96
Errata	9