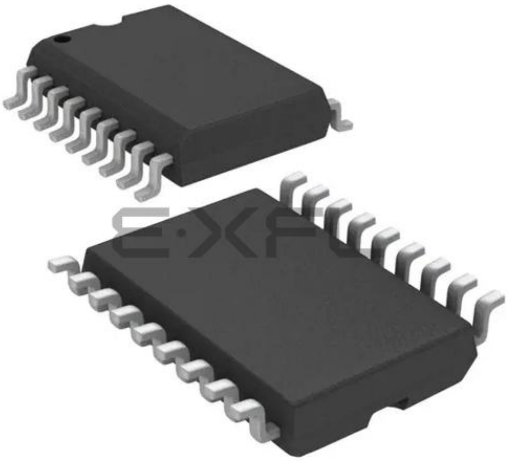


Welcome to E-XFL.COM

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	20MHz
Connectivity	UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	224 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f628a-i-so

NOTES:

FIGURE 4-2: DATA MEMORY MAP OF THE PIC16F627A AND PIC16F628A

Bank 0	File Address	Bank 1	File Address	Bank 2	File Address	Bank 3	File Address
Indirect addr.(1)	00h	Indirect addr.(1)	80h	Indirect addr.(1)	100h	Indirect addr.(1)	180h
TMR0	01h	OPTION	81h	TMR0	101h	OPTION	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch		10Ch		18Ch
	0Dh		8Dh		10Dh		18Dh
TMR1L	0Eh	PCON	8Eh		10Eh		18Eh
TMR1H	0Fh		8Fh		10Fh		18Fh
T1CON	10h		90h				
TMR2	11h		91h				
T2CON	12h	PR2	92h				
	13h		93h				
	14h		94h				
CCPR1L	15h		95h				
CCPR1H	16h		96h				
CCP1CON	17h		97h				
RCSTA	18h	TXSTA	98h				
TXREG	19h	SPBRG	99h				
RCREG	1Ah	EEDATA	9Ah				
	1Bh	EEADR	9Bh				
	1Ch	EECON1	9Ch				
	1Dh	EECON2(1)	9Dh				
	1Eh		9Eh				
CMCON	1Fh	VRCON	9Fh		11Fh		
General Purpose Register 80 Bytes	20h–6Fh	General Purpose Register 80 Bytes	A0h–EFh	General Purpose Register 48 Bytes	120h–14Fh		
					150h–16Fh		1EFh
16 Bytes	70h–7Fh	accesses 70h-7Fh	F0h–FFh	accesses 70h-7Fh	170h–17Fh	accesses 70h-7Fh	1F0h–1FFh

■ Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

4.2.2.2 OPTION Register

The Option register is a readable and writable register, which contains various control bits to configure the TMR0/WDT prescaler, the external RB0/INT interrupt, TMR0 and the weak pull-ups on PORTB.

> **Note:** To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT (PSA = 1). See **Section 6.3.1 "Switching Prescaler Assignment"**.

REGISTER 4-2: OPTION_REG – OPTION REGISTER (ADDRESS: 81h, 181h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

bit 7 **$\overline{\text{RBPU}}$**: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5 **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI/CMP2 pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4 **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI/CMP2 pin
0 = Increment on low-to-high transition on RA4/T0CKI/CMP2 pin

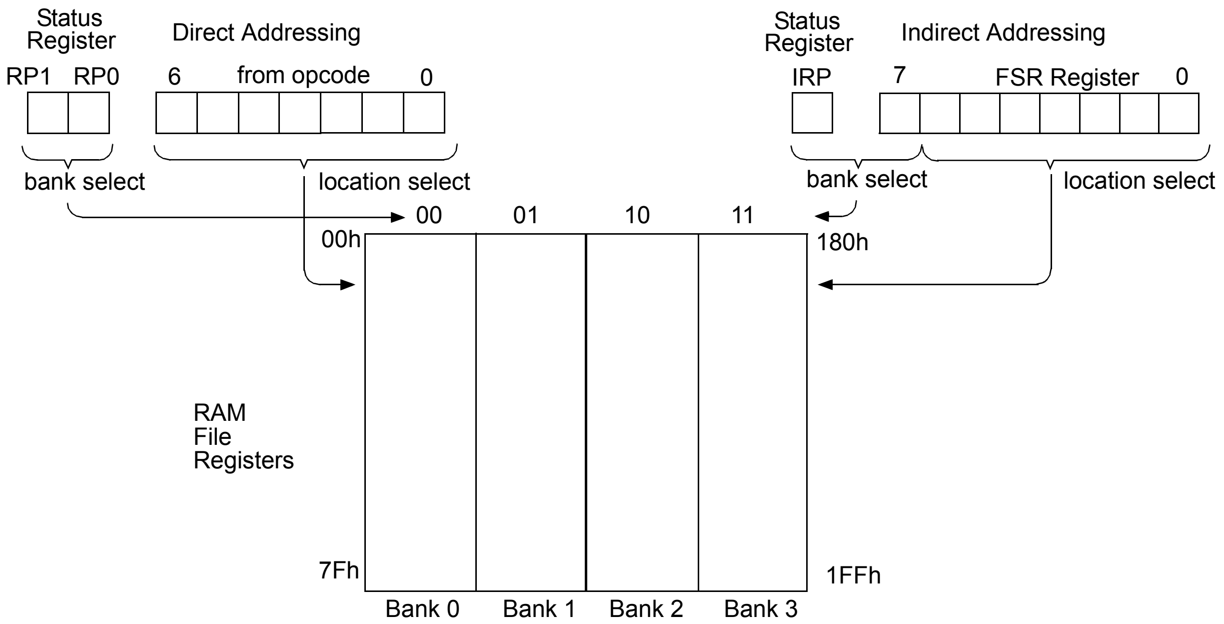
bit 3 **PSA**: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>**: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

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

FIGURE 4-5: DIRECT/INDIRECT ADDRESSING PIC16F627A/628A/648A



Note: For memory map detail see Figure 4-3, Figure 4-2 and Figure 4-1.

FIGURE 5-4: BLOCK DIAGRAM OF RA4/T0CKI/CMP2 PIN

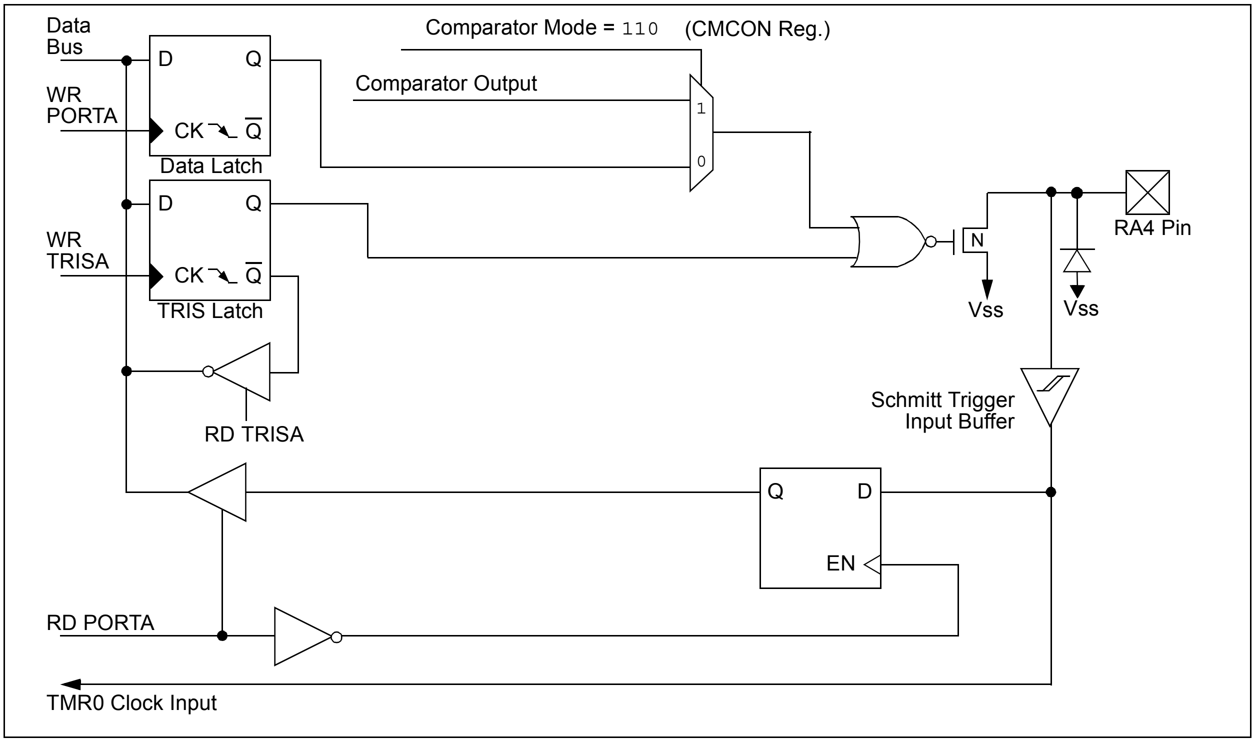


FIGURE 5-5: BLOCK DIAGRAM OF THE RA5/MCLR/VPP PIN

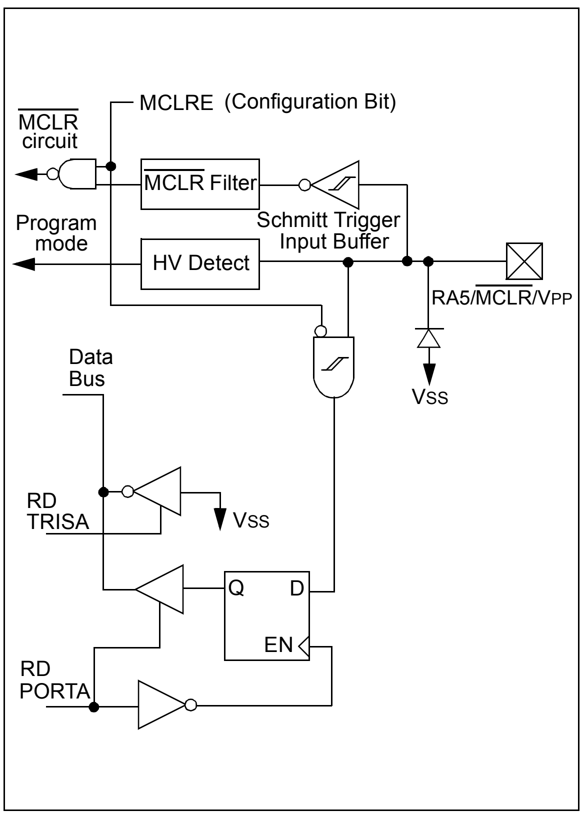
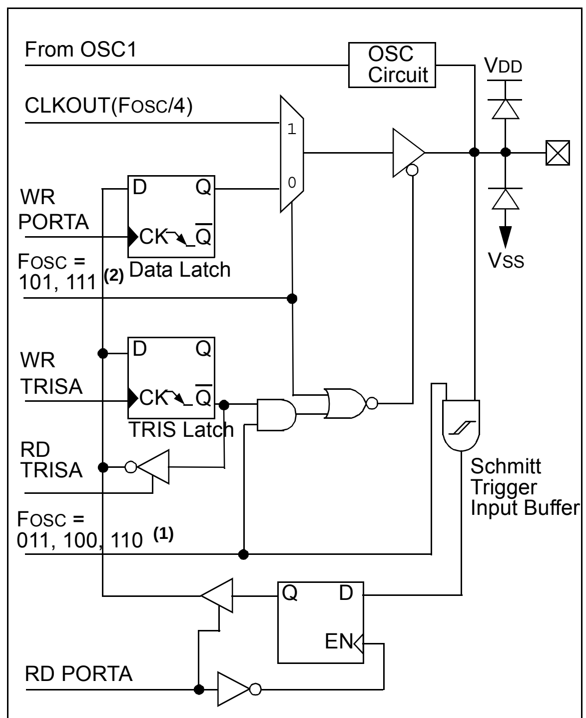
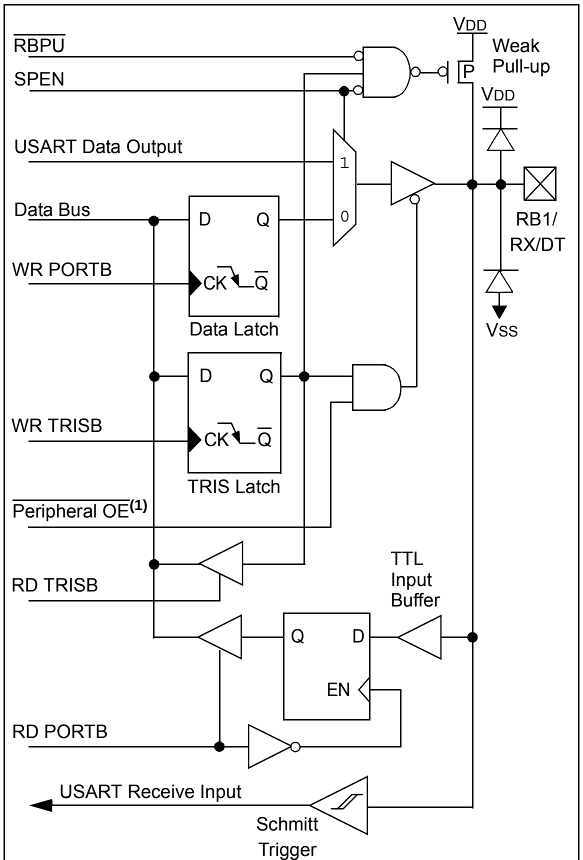


FIGURE 5-6: BLOCK DIAGRAM OF RA6/OSC2/CLKOUT PIN



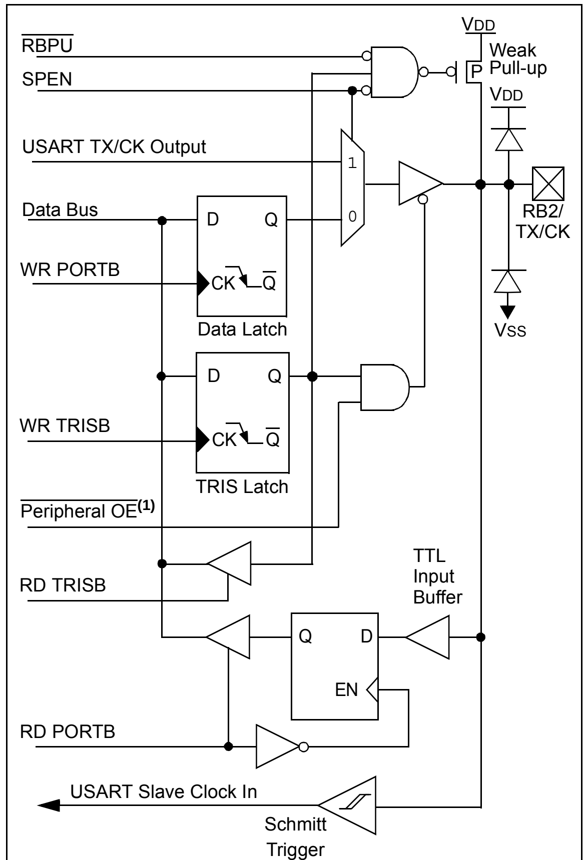
Note 1: INTOSC with RA6 = I/O or RC with RA6 = I/O.
2: INTOSC with RA6 = CLKOUT or RC with RA6 = CLKOUT.

FIGURE 5-9: BLOCK DIAGRAM OF RB1/RX/DT PIN



Note 1: Peripheral OE (output enable) is only active if peripheral select is active.

FIGURE 5-10: BLOCK DIAGRAM OF RB2/TX/CK PIN



Note 1: Peripheral OE (output enable) is only active if peripheral select is active.

6.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Read/write capabilities
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 6-1 is a simplified block diagram of the Timer0 module. Additional information is available in the *“PIC® Mid-Range MCU Family Reference Manual”* (DS33023).

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In Timer mode, the TMR0 register value will increment every instruction cycle (without prescaler). If the TMR0 register is written to, the increment is inhibited for the following two cycles. The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting the T0CS bit. In this mode the TMR0 register value will increment either on every rising or falling edge of pin RA4/T0CKI/CMP2. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION<4>). Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 6.2 “Using Timer0 with External Clock”**.

The prescaler is shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale value of 1:2, 1:4,..., 1:256 are selectable. **Section 6.3 “Timer0 Prescaler”** details the operation of the prescaler.

6.1 Timer0 Interrupt

Timer0 interrupt is generated when the TMR0 register timer/counter overflows from FFh to 00h. This overflow sets the T0IF bit. The interrupt can be masked by clearing the T0IE bit (INTCON<5>). The T0IF bit (INTCON<2>) must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The Timer0 interrupt cannot wake the processor from Sleep since the timer is shut off during Sleep.

6.2 Using Timer0 with External Clock

When an external clock input is used for Timer0, it must meet certain requirements. The external clock requirement is due to internal phase clock (TOSC) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.

6.2.1 EXTERNAL CLOCK SYNCHRONIZATION

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-1). Therefore, it is necessary for T0CKI to be high for at least 2TOSC (and a small RC delay of 20 ns) and low for at least 2TOSC (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

When a prescaler is used, the external clock input is divided by the asynchronous ripple-counter type prescaler so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple-counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4TOSC (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device. See Table 17-8.

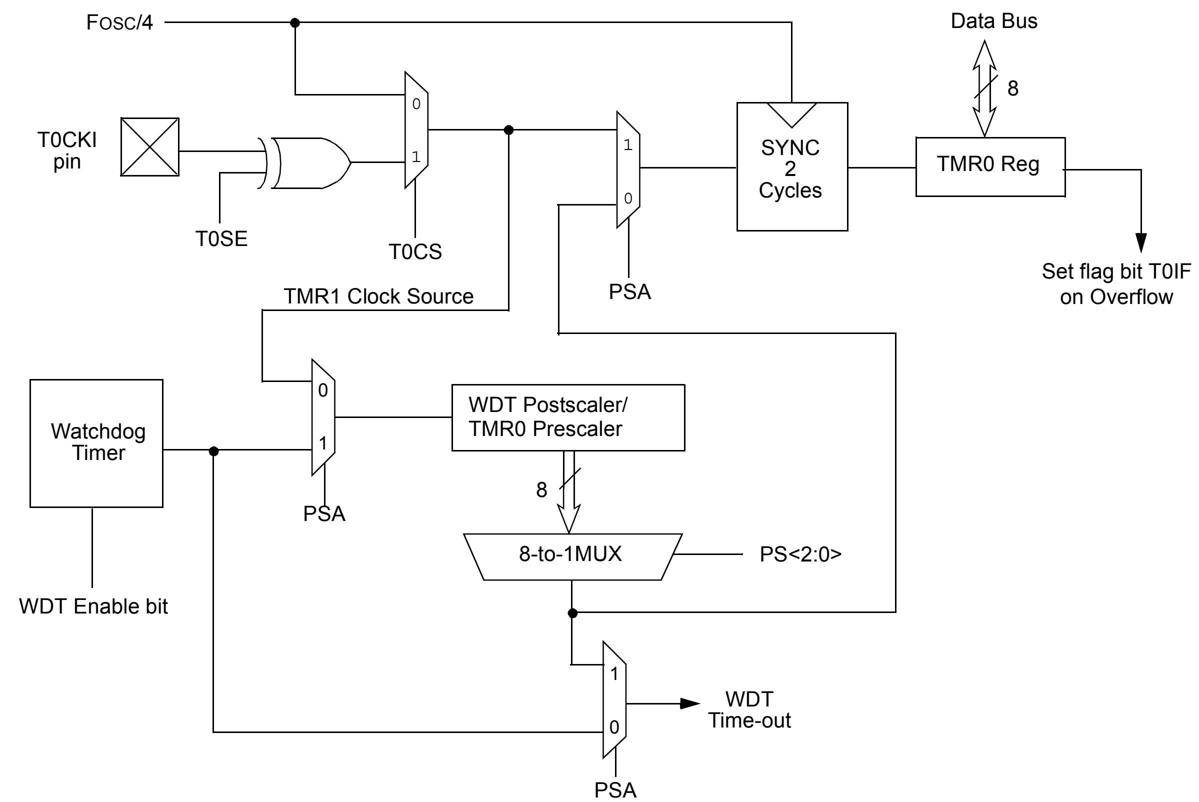
6.3 Timer0 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer. A prescaler assignment for the Timer0 module means that there is no postscaler for the Watchdog Timer, and vice-versa.

The PSA and PS<2:0> bits (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF 1, MOVWF 1, BSF 1, x....etc.`) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

FIGURE 6-1: BLOCK DIAGRAM OF THE TIMER0/WDT



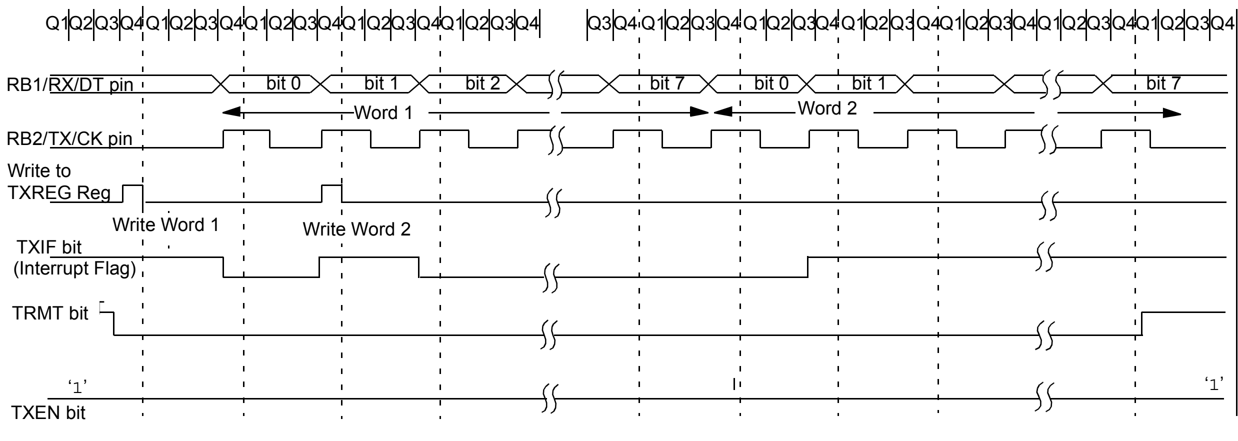
Note: T0SE, T0CS, PSA, PS<2:0> are bits in the Option Register.

TABLE 12-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other Resets
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	0000 -000	0000 -000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	0000 -000	0000 -000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

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

FIGURE 12-8: SYNCHRONOUS TRANSMISSION



Note: Sync Master Mode; SPBRG = 0. Continuous transmission of two 8-bit words.

FIGURE 12-9: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

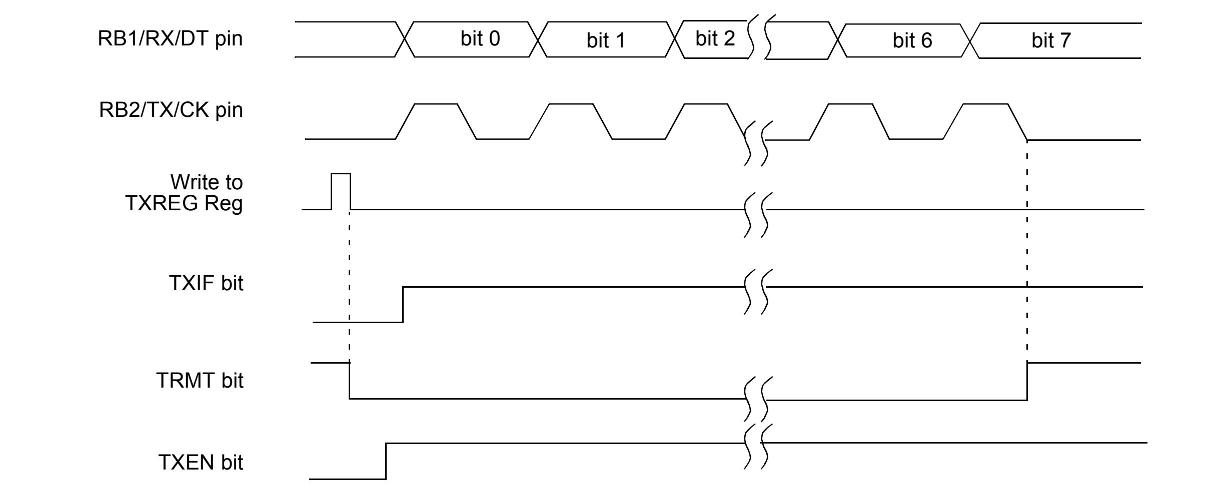
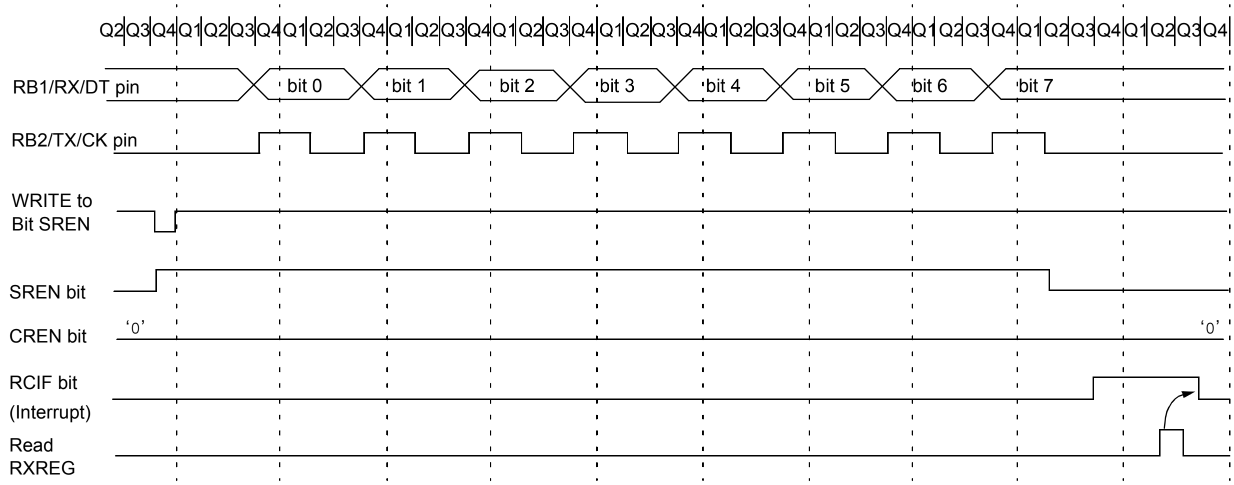


FIGURE 12-10: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)



Note: Timing diagram demonstrates Sync Master mode with bit SREN = `1` and bit BRG = `0`.

12.5 USART Synchronous Slave Mode

Synchronous Slave mode differs from the Master mode in the fact that the shift clock is supplied externally at the RB2/TX/CK pin (instead of being supplied internally in Master mode). This allows the device to transfer or receive data while in Sleep mode. Slave mode is entered by clearing bit CSRC (TXSTA<7>).

12.5.1 USART SYNCHRONOUS SLAVE TRANSMIT

The operation of the Synchronous Master and Slave modes are identical except in the case of the Sleep mode.

If two words are written to the TXREG and then the `SLEEP` instruction is executed, the following will occur:

a) The first word will immediately transfer to the TSR register and transmit.

b) The second word will remain in TXREG register.

c) Flag bit TXIF will not be set.

d) When the first word has been shifted out of TSR, the TXREG register will transfer the second word to the TSR and flag bit TXIF will now be set.

e) If enable bit TXIE is set, the interrupt will wake the chip from Sleep and if the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

Follow these steps when setting up a Synchronous Slave Transmission:

1. TRISB<1> and TRISB<2> should both be set to '`1`' to configure the RB1/RX/DT and RB2/TX/CK pins as inputs. Output drive, when required, is controlled by the peripheral circuitry.
2. Enable the synchronous slave serial port by setting bits SYNC and SPEN and clearing bit CSRC.
3. Clear bits CREN and SREN.
4. If interrupts are desired, then set enable bit TXIE.
5. If 9-bit transmission is desired, then set bit TX9.
6. Enable the transmission by setting enable bit TXEN.
7. If 9-bit transmission is selected, the ninth bit should be loaded in bit TX9D.
8. Start transmission by loading data to the TXREG register.

13.1 EEADR

The PIC16F648A EEADR register addresses 256 bytes of data EEPROM. All eight bits in the register (EEADR<7:0>) are required.

The PIC16F627A/628A EEADR register addresses only the first 128 bytes of data EEPROM so only seven of the eight bits in the register (EEADR<6:0>) are required. The upper bit is address decoded. This means that this bit should always be '0' to ensure that the address is in the 128 byte memory space.

13.2 EECON1 and EECON2 Registers

EECON1 is the control register with four low order bits physically implemented. The upper-four bits are non-existent and read as '0's.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a $\overline{MCLR}$ Reset or a WDT Time-out Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit and rewrite the location. The data and address will be unchanged in the EEDATA and EEADR registers.

Interrupt flag bit EEIF in the PIR1 register is set when write is complete. This bit must be cleared in software.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the data EEPROM write sequence.

REGISTER 13-3: EECON1 – EEPROM CONTROL REGISTER 1 (ADDRESS: 9Ch)

U-0	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	—	WRERR	WREN	WR	RD
bit 7							bit 0

bit 7-4 **Unimplemented**: Read as '0'

bit 3 **WRERR**: EEPROM Error Flag bit
1 = A write operation is prematurely terminated (any $\overline{MCLR}$ Reset, any WDT Reset during normal operation or BOR Reset)
0 = The write operation completed

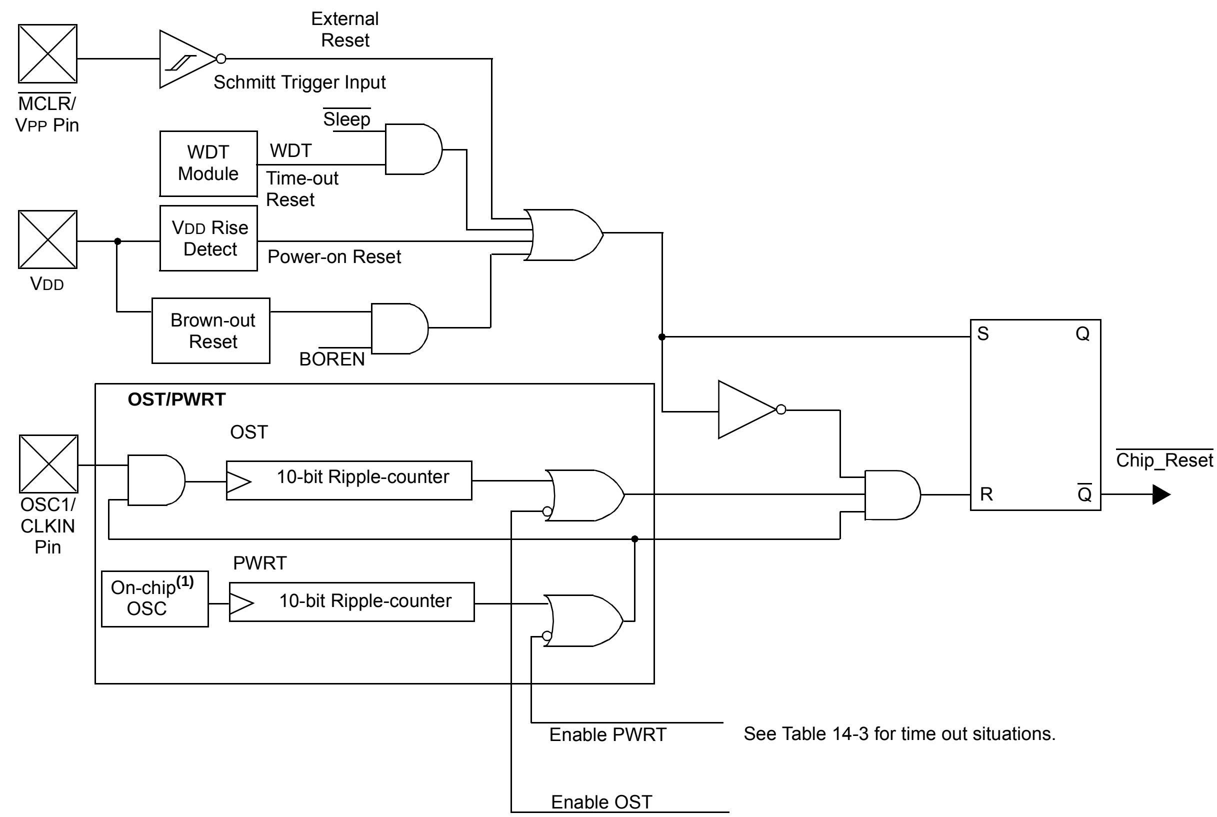
bit 2 **WREN**: EEPROM Write Enable bit
1 = Allows write cycles
0 = Inhibits write to the data EEPROM

bit 1 **WR**: Write Control bit
1 = initiates a write cycle. (The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.
0 = Write cycle to the data EEPROM is complete

bit 0 **RD**: Read Control bit
1 = Initiates an EEPROM read (read takes one cycle. RD is cleared in hardware. The RD bit can only be set (not cleared) in software).
0 = Does not initiate an EEPROM read

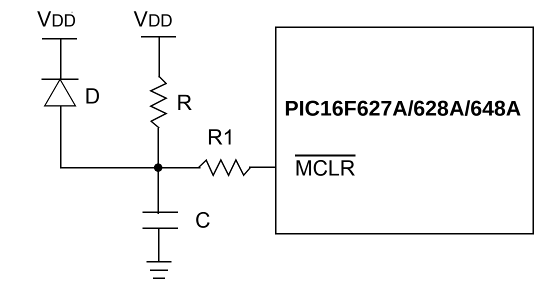
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

FIGURE 14-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is a separate oscillator from the INTOSC/RC oscillator.

FIGURE 14-11: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)

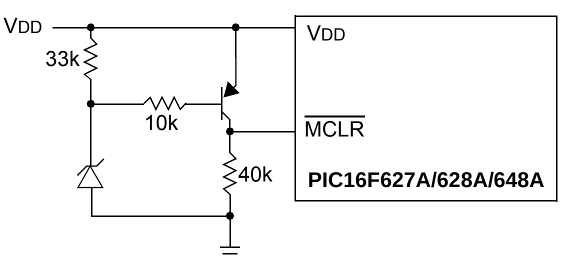


Note 1: External Power-on Reset circuit is required only if VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.

2: R < 40 kΩ is recommended to make sure that voltage drop across R does not violate the device's electrical specification.

3: R1 = 100Ω to 1 kΩ will limit any current flowing into MCLR from external capacitor C in the event of MCLR/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

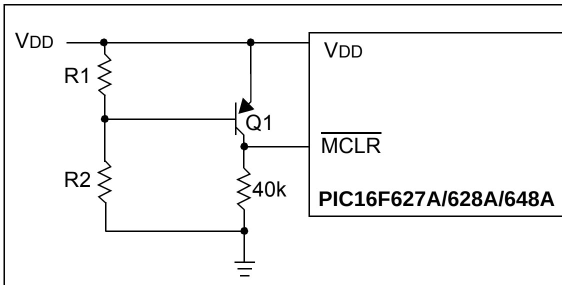
FIGURE 14-12: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1



Note 1: This circuit will activate Reset when VDD goes below (Vz + 0.7V) where Vz = Zener voltage.

2: Internal Brown-out Reset circuitry should be disabled when using this circuit.

FIGURE 14-13: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2



Note 1: This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

$$V_{DD} \times \frac{R1}{R1 + R2} = 0.7\ V$$

2: Internal Brown-out Reset should be disabled when using this circuit.

3: Resistors should be adjusted for the characteristics of the transistor.

14.5 Interrupts

The PIC16F627A/628A/648A has 10 sources of interrupt:

- External Interrupt RB0/INT
- TMR0 Overflow Interrupt
- PORTB Change Interrupts (pins RB<7:4>)
- Comparator Interrupt
- USART Interrupt TX
- USART Interrupt RX
- CCP Interrupt
- TMR1 Overflow Interrupt
- TMR2 Match Interrupt
- Data EEPROM Interrupt

The Interrupt Control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

A Global Interrupt Enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in INTCON register. GIE is cleared on Reset.

The “return-from-interrupt” instruction, `RETFIE`, exits interrupt routine as well as sets the GIE bit, which re-enables RB0/INT interrupts.

The INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

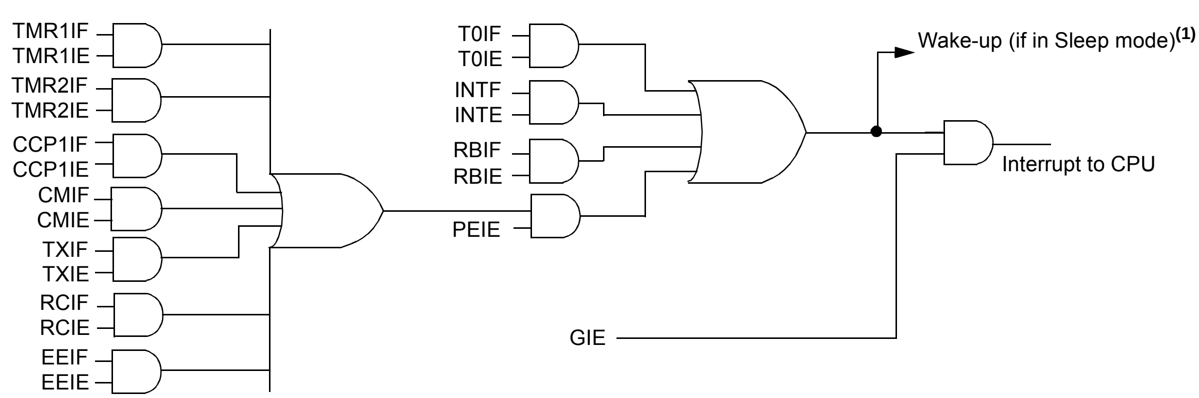
The peripheral interrupt flag is contained in the special register PIR1. The corresponding interrupt enable bit is contained in special registers PIE1.

When an interrupt is responded to, the GIE is cleared to disable any further interrupt, the return address is pushed into the stack and the PC is loaded with 0004h. Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid RB0/INT recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs (Figure 14-15). The latency is the same for one or two-cycle instructions. Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

Note 1:	Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.
2:	When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The CPU will execute a `NOP` in the cycle immediately following the instruction which clears the GIE bit. The interrupts which were ignored are still pending to be serviced when the GIE bit is set again.

FIGURE 14-14: INTERRUPT LOGIC



Note 1: Some peripherals depend upon the system clock for operation. Since the system clock is suspended during Sleep, only those peripherals which do not depend upon the system clock will wake the part from Sleep. See **Section 14.8.1 “Wake-up from Sleep”**.

TABLE 14-8: SUMMARY OF INTERRUPT REGISTERS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR Reset	Value on all other Resets(1)
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	`0000 000x`	`0000 000u`
0Ch	PIR1	EEIF	CMIF	RCIF	TXIF	—	CCP1IF	TMR2IF	TMR1IF	`0000 -000`	`0000 -000`
8Ch	PIE1	EEIE	CMIE	RCIE	TXIE	—	CCP1IE	TMR2IE	TMR1IE	`0000 -000`	`0000 -000`

Note 1: Other (non Power-up) Resets include $\overline{\text{MCLR}}$ Reset, Brown-out Reset and Watchdog Timer Reset during normal operation.

14.6 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt (e.g., W register and Status register). This must be implemented in software.

Example 14-1 stores and restores the Status and W registers. The user register, W_TEMP, must be defined in a common memory location (i.e., W_TEMP is defined at 0x70 in Bank 0 and is therefore, accessible at 0xF0, 0x170 and 0x1F0). The Example 14-1:

- Stores the W register
- Stores the Status register
- Executes the ISR code
- Restores the Status (and bank select bit register)
- Restores the W register

EXAMPLE 14-1: SAVING THE STATUS AND W REGISTERS IN RAM

```
MOVWF   W_TEMP       ;copy W to temp register,
                     ;could be in any bank
SWAPF   STATUS,W     ;swap status to be saved
                     ;into W
BCF     STATUS,RP0   ;change to bank 0
                     ;regardless of current
                     ;bank
MOVWF   STATUS_TEMP  ;save status to bank 0
                     ;register
    :
    :(ISR)
    :

SWAPF   STATUS_TEMP,W;swap STATUS_TEMP
register
                     ;into W, sets bank to
original
                     ;state
MOVWF   STATUS       ;move W into STATUS
                     ;register
SWAPF   W_TEMP,F     ;swap W_TEMP
SWAPF   W_TEMP,W     ;swap W_TEMP into W
```

14.7 Watchdog Timer (WDT)

The Watchdog Timer is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the CLKIN pin. That means that the WDT will run, even if the clock on the OSC1 and OSC2 pins of the device has been stopped, for example, by execution of a `SLEEP` instruction. During normal operation, a WDT time out generates a device Reset. If the device is in Sleep mode, a WDT time out causes the device to wake-up and continue with normal operation. The WDT can be permanently disabled by programming the configuration bit WDTE as clear (**Section 14.1 "Configuration Bits"**).

14.7.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (see DC Specifications, Table 17-7). If longer time-out periods are desired, a postscaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION register. Thus, time-out periods up to 2.3 seconds can be realized.

The `CLRWDT` and `SLEEP` instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device Reset.

The $\overline{\text{TO}}$ bit in the Status register will be cleared upon a Watchdog Timer time out.

14.7.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst case conditions (VDD = Min., Temperature = Max., max. WDT prescaler) it may take several seconds before a WDT time out occurs.

XORLW	Exclusive OR Literal with W
Syntax:	[*label*] XORLW k
Operands:	$0 \leq k \leq 255$
Operation:	(W) .XOR. k $\rightarrow$ (W)
Status Affected:	Z
Encoding:	`11` `1010` `kkkk` `kkkk`
Description:	The contents of the W register are XOR'ed with the eight-bit literal 'k'. The result is placed in the W register.
Words:	1
Cycles:	1
Example:	`XORLW 0xAF` Before Instruction W = 0xB5 After Instruction W = 0x1A

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	(W) .XOR. (f) $\rightarrow$ (dest)
Status Affected:	Z
Encoding:	`00` `0110` `dfff` `ffff`
Description:	Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.
Words:	1
Cycles:	1
Example	`XORWF REG1, 1` Before Instruction REG1 = 0xAF W = 0xB5 After Instruction REG1 = 0x1A W = 0xB5

17.4 DC Characteristics: PIC16F627A/628A/648A (Industrial, Extended) PIC16LF627A/628A/648A (Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for industrial and -40°C ≤ T_A ≤ +125°C for extended Operating voltage V_{DD} range as described in DC specification Table 17-2 and Table 17-3				
Param. No.	**Sym**	**Characteristic/Device**	**Min**	**Typ†**	**Max**	**Unit**	**Conditions**
	V_{IL}	**Input Low Voltage**					
		I/O ports					
D030		with TTL buffer	V_{SS}	—	0.8	V	V_{DD} = 4.5V to 5.5V
			V_{SS}	—	0.15 V_{DD}	V	otherwise
D031		with Schmitt Trigger input(4)	V_{SS}	—	0.2 V_{DD}	V	
D032		$\overline{MCLR}$, RA4/T0CKI,OSC1 (in RC mode)	V_{SS}	—	0.2 V_{DD}	V	**(Note1)**
D033		OSC1 (in HS)	V_{SS}	—	0.3 V_{DD}	V	
		OSC1 (in LP and XT)	V_{SS}	—	0.6	V	
	V_{IH}	**Input High Voltage**					
		I/O ports					
D040		with TTL buffer	2.0V	—	V_{DD}	V	V_{DD} = 4.5V to 5.5V
			.25 V_{DD} + 0.8V	—	V_{DD}	V	otherwise
D041		with Schmitt Trigger input(4)	0.8 V_{DD}	—	V_{DD}	V	
D042		$\overline{MCLR}$ RA4/T0CKI	0.8 V_{DD}	—	V_{DD}	V	
D043		OSC1 (XT and LP)	1.3	—	V_{DD}	V	
D043A		OSC1 (in RC mode)	0.9 V_{DD}	—	V_{DD}	V	**(Note1)**
D043B		OSC1 (in HS mode)	0.7 V_{DD}		V_{DD}	V	
D070	I_{PURB}	**PORTB weak pull-up current**	50	200	400	μA	V_{DD} = 5.0V, V_{PIN} = V_{SS}
	I_{IL}	**Input Leakage Current(2), (3)**					
		I/O ports (Except PORTA)	—	—	±1.0	μA	V_{SS} ≤ V_{PIN} ≤ V_{DD}, pin at high-impedance
D060		PORTA(4)	—	—	±0.5	μA	V_{SS} ≤ V_{PIN} ≤ V_{DD}, pin at high-impedance
D061		RA4/T0CKI	—	—	±1.0	μA	V_{SS} ≤ V_{PIN} ≤ V_{DD}
D063		OSC1, $\overline{MCLR}$	—	—	±5.0	μA	V_{SS} ≤ V_{PIN} ≤ V_{DD}, XT, HS and LP oscillator configuration
	V_{OL}	**Output Low Voltage**					
D080		I/O ports(4)	—	—	0.6	V	I_{OL} = 8.5 mA, V_{DD} = 4.5 V, -40° to +85°C
			—	—	0.6	V	I_{OL} = 7.0 mA, V_{DD} = 4.5 V, +85° to +125°C
	V_{OH}	**Output High Voltage(3)**					
D090		I/O ports (Except RA4(4))	V_{DD} – 0.7	—	—	V	I_{OH} = -3.0 mA, V_{DD} = 4.5 V, -40° to +85°C
			V_{DD} – 0.7	—	—	V	I_{OH} = -2.5 mA, V_{DD} = 4.5 V, +85° to +125°C
D150	V_{OD}	**Open-Drain High Voltage**	—	—	8.5*	V	RA4 pin PIC16F627A/628A/648A, PIC16LF627A/628A/648A
		Capacitive Loading Specs on Output Pins					
D100*	C_{OSC2}	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock used to drive OSC1.
D101*	C_{IO}	All I/O pins/OSC2 (in RC mode)	—	—	50	pF	

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. It is not recommended that the PIC16F627A/628A/648A be driven with external clock in RC mode.

2: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

3: Negative current is defined as coming out of the pin.

4: Includes OSC1 and OSC2 when configured as I/O pins, CLKIN or CLKOUT.

TABLE 17-5: PRECISION INTERNAL OSCILLATOR PARAMETERS

Parameter No.	Sym	Characteristic	Min	Typ	Max	Units	Conditions
F10	FIOSC	Oscillator Center frequency	—	4	—	MHz	
F13	ΔIOSC	Oscillator Accuracy	3.96	4	4.04	MHz	VDD = 3.5 V, 25°C
			3.92	4	4.08	MHz	2.0V ≤ VDD ≤ 5.5V 0°C ≤ TA ≤ +85°C
			3.80	4	4.20	MHz	2.0V ≤ VDD ≤ 5.5V -40°C ≤ TA ≤ +85°C (IND) -40°C ≤ TA ≤ +125°C (EXT)
F14*	TIOSCST	Oscillator Wake-up from Sleep start-up time	—	6	8	μs	VDD = 2.0V, -40°C to +85°C
			—	4	6	μs	VDD = 3.0V, -40°C to +85°C
			—	3	5	μs	VDD = 5.0V, -40°C to +85°C

Legend: TBD = To Be Determined.
* Characterized but not tested.

FIGURE 17-5: CLKOUT AND I/O TIMING

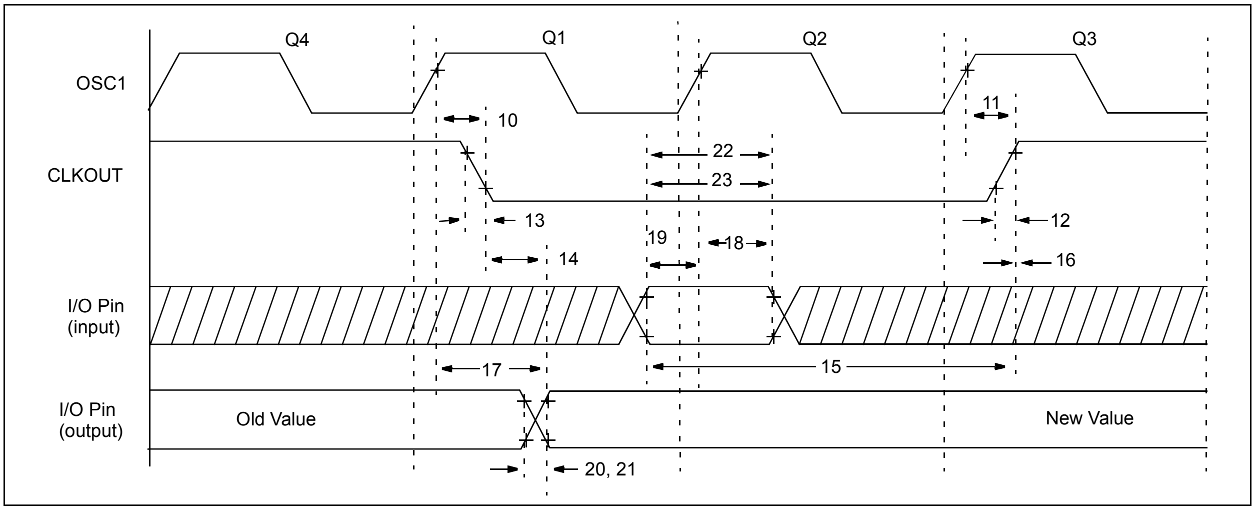


FIGURE 18-18: SUPPLY CURRENT (IDD vs. VDD, FOSC = 4 MHz (XT OSCILLATOR MODE)

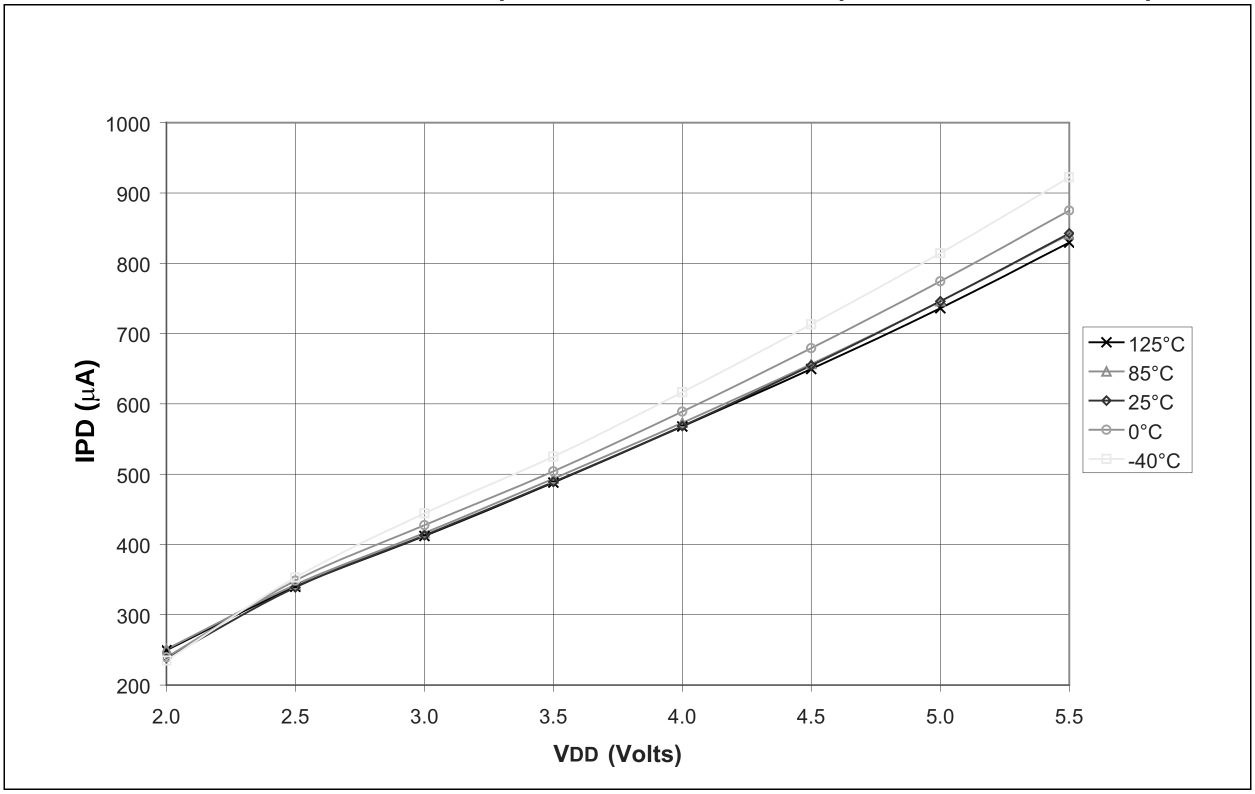
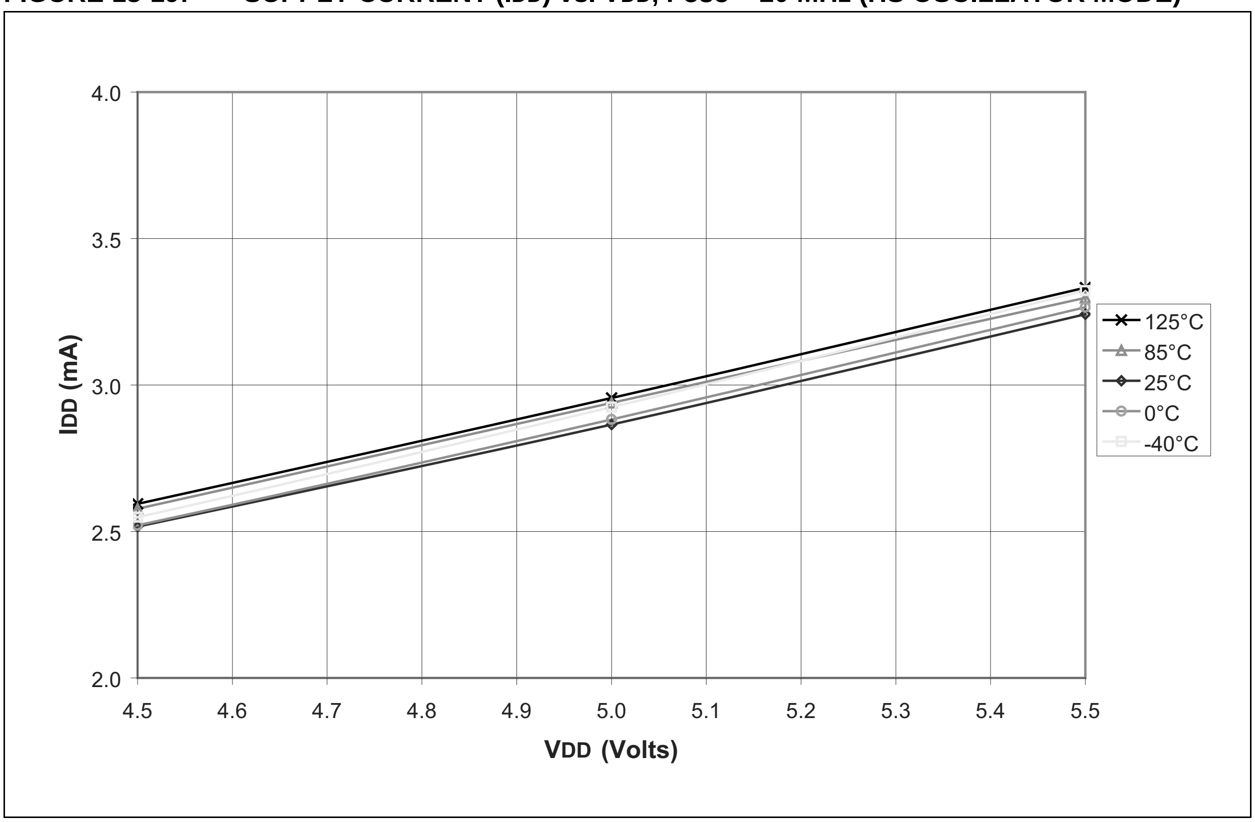


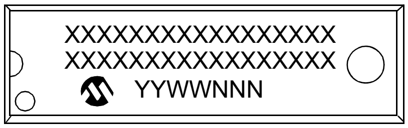
FIGURE 18-19: SUPPLY CURRENT (IDD) vs. VDD, FOSC = 20 MHz (HS OSCILLATOR MODE)



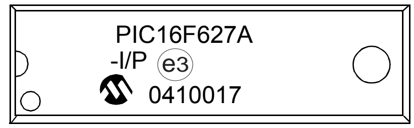
19.0 PACKAGING INFORMATION

19.1 Package Marking Information

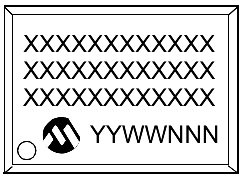
18-Lead PDIP



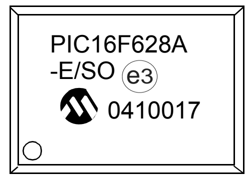
Example



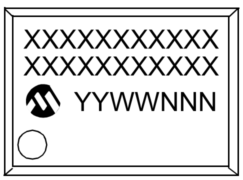
18-Lead SOIC (.300")



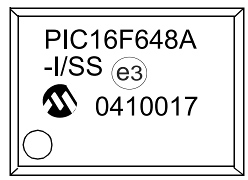
Example



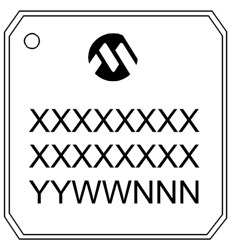
20-Lead SSOP



Example



28-Lead QFN



Example

Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	e3	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 event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

