FLASH-Based 8-Bit Microcontroller with EEPROM

FEATURES

- Only 41 single word instructions
- All instructions are single cycle except for program branches which are two-cycle
- 14-bit wide instructions
- All ROM area GOTO instruction
- All ROM area subroutine CALL instruction
- 8-bit wide data path
- 8-level deep hardware stack
- 2K x 14 bits on chip FLASH program memory
- 128 bytes on chip general purpose registers (SRAM)
- 256 bytes on chip EEPROM data memory
- Operating speed: DC-20 MHz clock input
  DC-100 ns instruction cycle
- Direct, indirect addressing modes for data accessing
- 8-bit real time clock/counter (Timer0) with 8-bit programmable prescaler
- 16-bit timer/counter (Timer1) with 3-bit programmable prescaler and optional external enable input (T1GB)
- 8-bit timer/counter (Timer2) with 2-bit programmable prescaler and 4-bit programmable postscaler
- Capture / Compare / PWM modules
- 12 I/O pins with independent direction control
- Soft-ware I/O pull-high control
- 2 analog comparator module
- 1 IR output channel with programmable frequency and duty cycle
- Internal Power-on Reset (POR)
- Built-in Low Voltage Detector (LVD) for Brown-out Reset (BOR)
- Power-up Reset Timer (PWRT) and Oscillator Start-up Timer(OST)
- On chip Watchdog Timer (WDT) with internal oscillator for reliable operation and soft-ware watch-dog enable/disable control
- 11 interrupt source: INT pin, Timer0 overflow, Port A input status change, Comparator1, Comparator2, Timer1 overflow, Timer2 match, Oscillator fail, Capture/Compare/PWM, EEPROM write finish, and IROUT counter match
- Wake-up from SLEEP by INT pin, Port A input change, or a peripheral interrupt
- Ultra low-power wake-up
- Power saving SLEEP mode
- Built-in 8MHz (HFIRC) / 31KHz (LFIRC) internal RC oscillator
- Programmable Code Protection
- Selectable oscillator options:
  - ERC: External Resistor/Capacitor Oscillator
  - HF: High Frequency Crystal/Resonator Oscillator
  - XT: Crystal/Resonator Oscillator
  - LF: Low Frequency Crystal Oscillator
  - IRC: Internal Resistor/Capacitor Oscillator
  - ERIC: External Resistor/Internal Capacitor Oscillator
  - EC: External Clock Input
- Two speed start-up mode
- Clock mode switching during operation for power saving
- Oscillator fail detect for critical applications
- Wide-operating voltage range: 3.3V to 5.5V (only for HV package : FM8PE32AF)
  2.3V to 3.3V (only for LV package : FM8PE32BF)

GENERAL DESCRIPTION

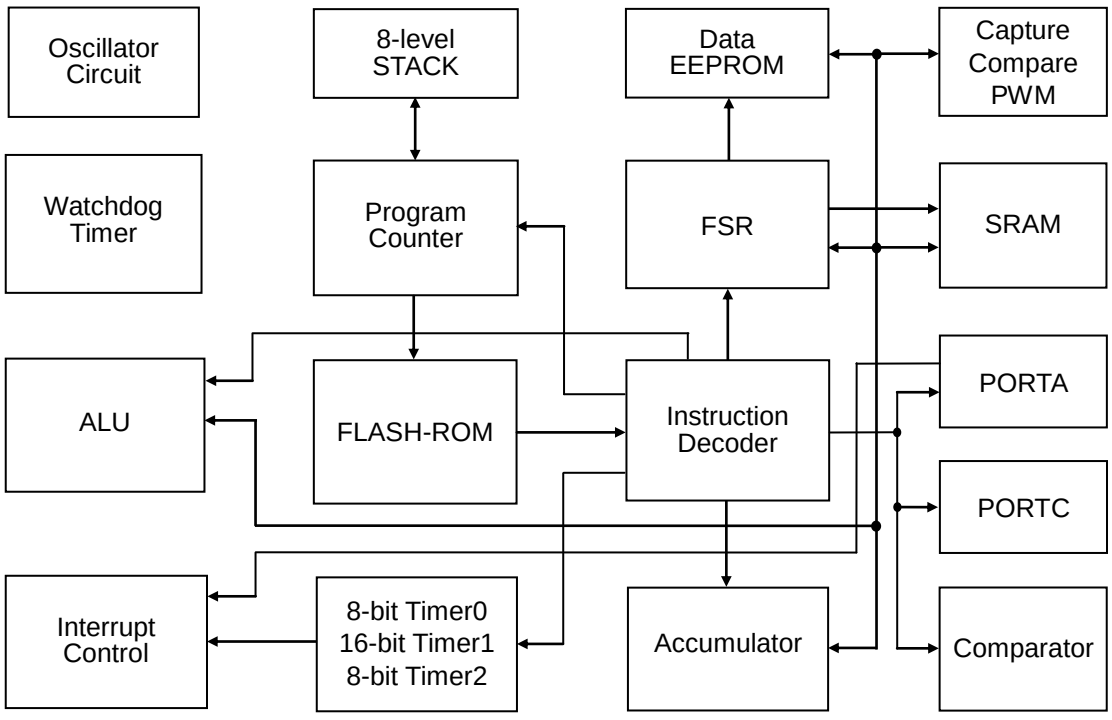
The FM8PE32F is a low-cost, high speed, high noise immunity, FLASH-based 8-bit CMOS microcontrollers. It employs a RISC architecture with only 41 instructions. All instructions are single cycle except for program branches which take two cycles. The easy to use and easy to remember instruction set reduces development time significantly.

The FM8PE32F consists of Power-on Reset (POR), Brown-out Reset (BOR), Power-up Reset Timer (PWRT), Oscillator Start-up Timer(OST), Watchdog Timer, data EEPROM, SRAM, tri-state I/O port, I/O pull-high control, Comparator, Power saving SLEEP mode, real time programmable clock/counter, Interrupt, Capture / Compare / PWM, Wake-up from SLEEP mode, and Code Protection. There are three oscillator configurations to choose from, including the power-saving LF (Low Power) oscillator and cost saving RC oscillator.

The FM8PE32F address 2K×14 of program memory.

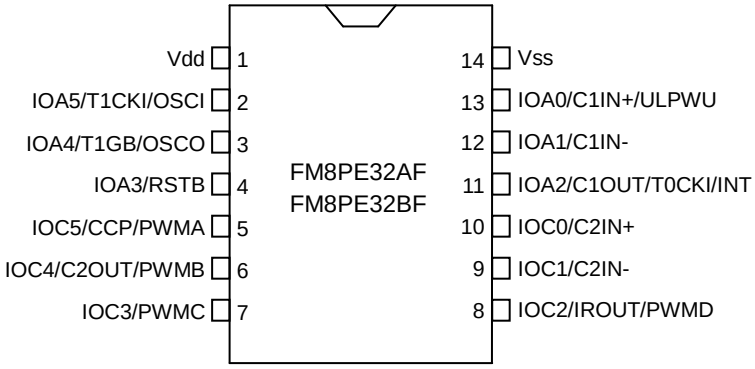
The FM8PE32F can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory.

BLOCK DIAGRAM

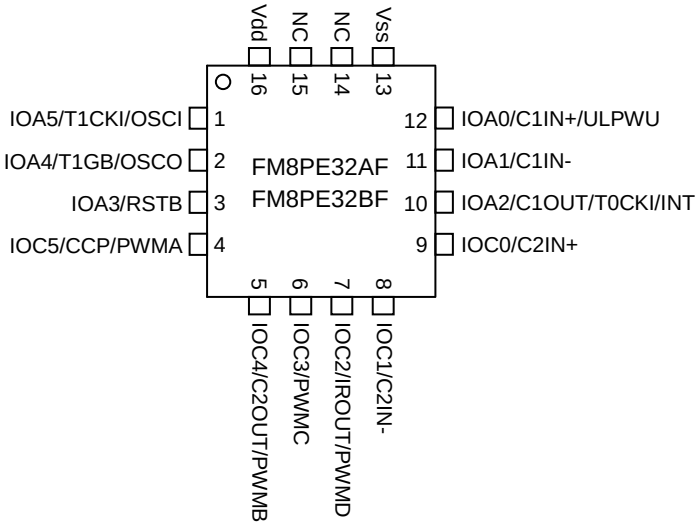


PIN CONNECTION

PDIP, SOP, TSSOP



QFN



PIN DESCRIPTIONS

Name	I/O	Description
IOA0/C1IN+/ULPWU	I/O	-Bi-direction I/O pin with S/W controlled pull-high and interrupt/wake-up function -The Vin+ input pin of the comparator 1 -Ultra low-power wake-up input
IOA1/C1IN-	I/O	-Bi-direction I/O pin with S/W controlled pull-high and interrupt/wake-up function -The Vin- input pin of the comparator 1
IOA2/C1OUT/T0CKI/INT	I/O	-Bi-direction I/O pin with S/W controlled pull-high and interrupt/wake-up function -The output of the comparator 1 -Timer0 clock input -External interrupt input
IOA3/RSTB	I	-Input pin only with interrupt function -System clear (RESET) input. Active low RESET to the device. Weak pull-high always on if configured as RSTB.
IOA4/T1GB/OSCO	I/O	-Bi-direction I/O pin with S/W controlled pull-high and interrupt/wake-up function -Timer1 gate -Oscillator crystal output (HF, XT, LF mode) Outputs with the instruction cycle rate (ERC, IRC mode)
IOA5/T1CKI/OSCI	I/O	-Bi-direction I/O pin with S/W controlled pull-high and interrupt/wake-up function -Oscillator crystal input (HF, XT, LF mode) -External clock source input (ERC, IRC, EC mode)
IOC0/C2IN+	I/O	-Bi-direction I/O pin -The Vin+ input pin of the comparator 2
IOC1/C2IN-	I/O	-Bi-direction I/O pin -The Vin- input pin of the comparator 2
IOC2/IROUT/PWMD	I/O	-Bi-direction I/O pin -IR output pin -PWM D output
IOC3/PWMC	I/O	-Bi-direction I/O pin -PWM C output
IOC4/C2OUT/PWMB	I/O	-Bi-direction I/O pin -The output of the comparator 2 -PWM B output
IOC5/CCP/PWMA	I/O	-Bi-direction I/O pin -Capture input / Compare output -PWM A output
Vdd	-	Positive supply
Vss	-	Ground

Legend: I=input, O=output, I/O=input/output

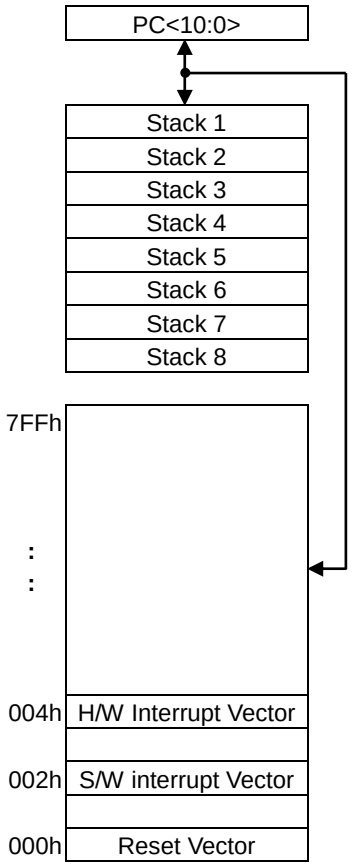
1.0 MEMORY ORGANIZATION

FM8PE32F memory is organized into program memory and data memory.

1.1 Program Memory Organization

The FM8PE32F have an 11-bit Program Counter capable of addressing a 2K×14 program memory space.
The RESET vector for the FM8PE32F is at 000h.
The interrupt vector is at 004h.
FM8PE32F supports all ROM area CALL/GOTO instructions without page.

FIGURE 1.1: Program Memory Map and STACK



FM8PE32F

1.2 Data Memory Organization

Data memory is composed of Special Function Registers and General Purpose Registers.
The General Purpose Registers are accessed either directly or indirectly through the FSR register.
The Special Function Registers are registers used by the CPU and peripheral functions to control the operation of the device.
In FM8PE32F, the data memory is partitioned into two banks. Switching between these banks requires the IRP0 bit in the FSR register (for indirect addressing) or the RP0 bit in the STATUS register (for direct addressing) to be configured for the desired bank.

TABLE 1.1: Registers File Map for FM8PE32F Series

RP0/IRP0 / Address	Description	
	Bank 0	**Bank 1**
00h	INDF	INDF
01h	TMR0	OPTION_REG
02h	PCL	PCL
03h	STATUS	STATUS
04h	FSR	FSR
05h	PORTA	IOSTA
06h		
07h	PORTC	IOSTC
08h		
09h		
0Ah	PCHBUF	PCHBUF
0Bh	INTCON	INTCON
0Ch	INTFLAG	INTEN
0Dh		
0Eh	TMR1L	PCON
0Fh	TMR1H	OSCCON
10h	T1CON	OSCTUNE
11h	TMR2	
12h	T2CON	PR2
13h	CCPRL	
14h	CCPRH	
15h	CCPCON	APHCON
16h	PWMCON	AWUCON
17h	CCPAS	
18h	WDTCON	
19h	CMPCON1	CMPCON3
1Ah	CMPCON2	EEDATA
1Bh	IRCYCLE	EEADR
1Ch	IRDUTY	EECON1
1Dh	IRCPR	EECON2
1Eh	IRCON	
1Fh		
20h | 3Fh	General Purpose Registers	General Purpose Registers
40h | 7Fh	General Purpose Registers	Memory back to address in Bank 0

TABLE 1.2: Operational Registers Map

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
Bank 0									
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
01h (r/w)	TMR0	8-bit real-time clock/counter 0							
02h (r/w)	PCL	Low order 8 bits of PC							
03h (r/w)	STATUS	0	0	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
04h (r/w)	FSR	IRP0	Indirect data memory address pointer						
05h (r/w)	PORTA	-	-	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
07h (r/w)	PORTC	-	-	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
0Ah (r/w)	PCHBUF	-	-	-	-	-	3 MSBs Buffer of PC		
0Bh (r/w)	INTCON	GIE	PEIE	T0IE	INTIE	PAIE	T0IF	INTIF	PAIF
0Ch (r/w)	INTFLAG	EEIF	IRIF	CCPIF	CMP2IF	CMP1IF	OSFIF	T2IF	T1IF
0Eh (r/w)	TMR1L	Low byte of 16-bit real-time clock/counter 1							
0Fh (r/w)	TMR1H	High byte of 16-bit real-time clock/counter 1							
10h (r/w)	T1CON	T1GINV	T1GE	T1PS1	T1PS0	T1OSC	T1SYNC	T1CS	T1ON
11h (r/w)	TMR2	TMR27	TMR26	TMR25	TMR24	TMR23	TMR22	TMR21	TMR20
12h (r/w)	T2CON	-	T2OPS3	T2OPS2	T2OPS1	T2OPS0	T2ON	T2PS1	T2PS0
13h (r/w)	CCPRL	Capture/Compare/PWM register low byte							
14h (r/w)	CCPRH	Capture/Compare/PWM register high byte							
15h (r/w)	CCPCON	PWMMOD1	PWMMOD0	DCB1	DCB0	CCPMOD3	CCPMOD2	CCPMOD1	CCPMOD0
16h (r/w)	PWMCON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
17h (r/w)	CCPAS	CCPASE	CCPAS2	CCPAS1	CCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
18h (r/w)	WDTCON	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTE
19h (r/w)	CMPCON1	C2OUT	C1OUT	C2INV	C1INV	CINS	CM2	CM1	CM0
1Ah (r/w)	CMPCON2	-	-	-	-	-	-	T1GSS	C2SYNC
1Bh (r/w)	IRCYCLE	IRC7	IRC6	IRC5	IRC4	IRC3	IRC2	IRC1	IRC0
1Ch (r/w)	IRDUTY	IRD7	IRD6	IRD5	IRD4	IRD3	IRD2	IRD1	IRD0
1Dh (r/w)	IRCPR	IRCPR7	IRCPR6	IRCPR5	IRCPR4	IRCPR3	IRCPR2	IRCPR1	IRCPR0
1Eh (r/w)	IRCON	IREN	IROEN	IRCEN	IRSC	-	-	IRPS1	IRPS0
Bank 1									
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							
01h (r/w)	OPTION_REG	/PHA	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
02h (r/w)	PCL	Low order 8 bits of PC							
03h (r/w)	STATUS	0	0	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
04h (r/w)	FSR	IRP0	Indirect data memory address pointer						
05h (r/w)	IOSTA	-	-	IOSTA5	IOSTA4	*	IOSTA2	IOSTA1	IOSTA0
07h (r/w)	IOSTC	-	-	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	IOSTC0
0Ah (r/w)	PCHBUF	-	-	-	-	-	3 MSBs Buffer of PC		
0Bh (r/w)	INTCON	GIE	PEIE	T0IE	INTIE	PAIE	T0IF	INTIF	PAIF
0Ch (r/w)	INTEN	EEIE	IRIE	CCPIE	CMP2IE	CMP1IE	OSFIE	T2IE	T1IE
0Eh (r/w)	PCON	-	-	ULPWUE	LVDTE	-	-	$\overline{POR}$	$\overline{BOD}$
0Fh (r/w)	OSCCON	-	IRCF2	IRCF1	IRCF0	OSTS	HTS	LTS	SCS
10h (r/w)	OSCTUNE	-	-	-	TUN4	TUN3	TUN2	TUN1	TUN0
12h (r/w)	PR2	PR27	PR26	PR25	PR24	PR23	PR22	PR21	PR20
15h (r/w)	APHCON	-	-	PHA5	PHA4	-	PHA2	PHA1	PHA0
16h (r/w)	AWUCON	-	-	WUA5	WUA4	WUA3	WUA2	WUA1	WUA0
19h (r/w)	CMPCON3	CVREN	-	CVRR	-	CVR3	CVR2	CVR1	CVR0
1Ah (r/w)	EEDATA	EEPROM data register							
1Bh (r/w)	EEADR	EEPROM address register							
1Ch (r/w)	EECON1	-	-	-	WRONLY	WRERR	WREN	WR	RD
1Dh (r/w)	EECON2	EEPROM control register 2 (not a physical register)							

Legend: - = unimplemented, read as '0', * = unimplemented, read as '1'

2.0 FUNCTIONAL DESCRIPTIONS

2.1 Operational Registers

2.1.1 INDF (Indirect Addressing Register)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
00h (r/w)	INDF	Uses contents of FSR to address data memory (not a physical register)							

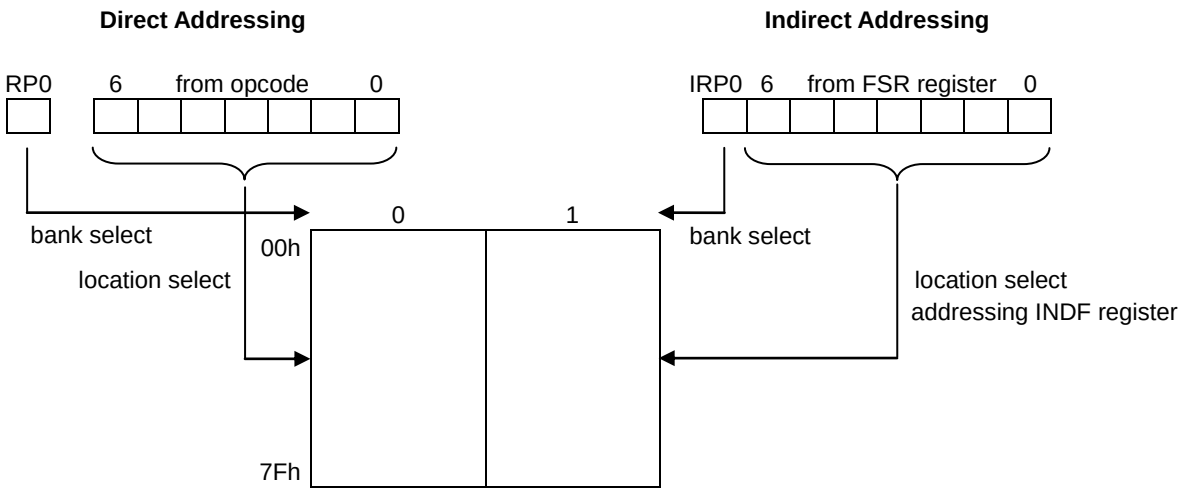
The INDF Register is not a physical register. Any instruction accessing the INDF register can actually access the register pointed by FSR Register. Reading the INDF register itself indirectly (FSR="0") will read 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected).
The bits 6-0 of FSR register are used to select up to 128 registers (address: 00h ~ 7Fh).
In FM8PE32F, the data memory is partitioned into two banks. Switching between these banks requires the IRP0 bit in the FSR register (for indirect addressing) or the RP0 bit in the STATUS register (for direct addressing) to be configured for the desired bank. The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers. All Special Function Registers and some of General Purpose Registers from other banks are mirrored in bank 0 for code reduction and quicker access.

RP0 (direct addressing) IRP0 (indirect addressing)	Accessed Bank
0	0
1	1

EXAMPLE 2.1: INDIRECT ADDRESSING

- Register file 38 contains the value 10h
- Register file 39 contains the value 0Ah
- Load the value 38 into the FSR Register
- A read of the INDF Register will return the value of 10h
- Increment the value of the FSR Register by one (@FSR=39h)
- A read of the INDR register now will return the value of 0Ah.

FIGURE 2.1: Direct/Indirect Addressing for FM8PE32F



2.1.2 TMR0 (Timer 0 Clock/Counter Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
01h (r/w)	TMR0	8-bit real-time clock/counter 0							

The Timer0 is an 8-bit timer/counter. The clock source of Timer0 can come from the instruction cycle clock or by an external clock source (T0CKI pin) defined by T0CS bit (OPTION_REG<5>). If T0CKI pin is selected, the Timer0 is increased by T0CKI signal rising/falling edge (selected by T0SE bit (OPTION_REG<4>)).
The prescaler is assigned to Timer0 by clearing the PSA bit (OPTION_REG<3>). In this case, the prescaler will be cleared when TMR0 register is written with a value.

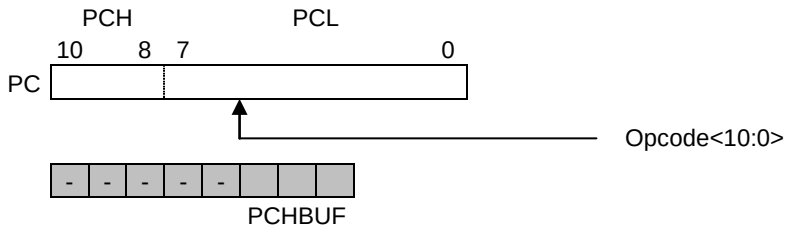
2.1.3 PCL (Low Bytes of Program Counter) & Stack

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
02h (r/w)	PCL	Low order 8 bits of PC							

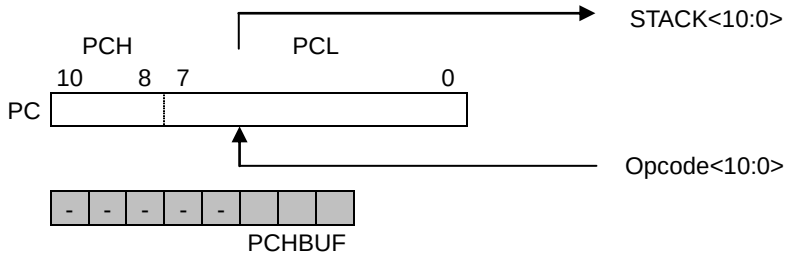
FM8PE32F devices have an 11-bit wide Program Counter (PC) and eight-level deep 11-bit hardware push/pop stack. The low byte of PC is called the PCL register. This register is readable and writable. The high byte of PC is called the PCH register. This register contains the PC<10:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCHBUF register. As a program instruction is executed, the Program Counter will contain the address of the next program instruction to be executed. The PC value is increased by one, every instruction cycle, unless an instruction changes the PC.
For a GOTO instruction, the PC<10:0> is provided by the GOTO instruction word. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For a CALL instruction, the PC<10:0> is provided by the CALL instruction word. The next PC will be loaded (PUSHed) onto the top of STACK. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For a RETIA, RETFIE, or RETURN instruction, the PC are updated (POPed) from the top of STACK. The PCL register is mapped to PC<7:0>, and the PCHBUF register is not updated.
For any instruction where the PCL is the destination, the PC<7:0> is provided by the instruction word or ALU result. However, the PC<10:8> will come from the PCHBUF<2:0> bits (PCHBUF → PCH).
PCHBUF register is never updated with the contents of PCH.

FIGURE 2.2: Loading of PC in Different Situations

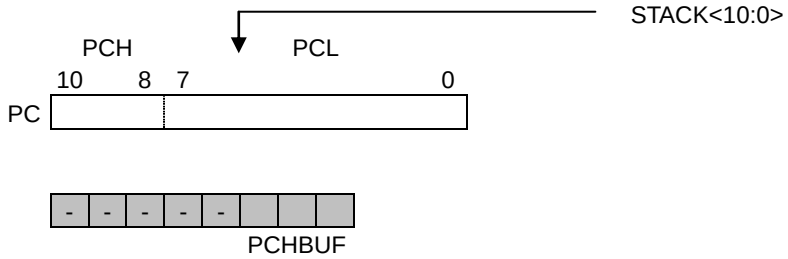
Situation 1: GOTO Instruction



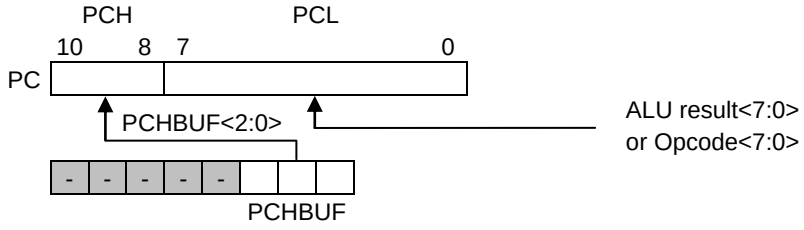
Situation 2: CALL Instruction



Situation 3: RETIA, RETFIE, or RETURN Instruction



Situation 4: Instruction with PCL as destination



Note: 1. PCHBUF is used only for instruction with PCL as destination.

2.1.4 STATUS (Status Register)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
03h (r/w)	STATUS	0	0	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C

This register contains the arithmetic status of the ALU, the RESET status.
If the STATUS Register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS Register as destination may be different than intended. For example, CLRR STATUS will clear the upper three bits and set the Z bit. This leaves the STATUS Register as 000u u1uu (where u = unchanged).

C : Carry/borrow bit.
ADDAR, ADDIA
= 1, a carry occurred.
= 0, a carry did not occur.
SUBAR, SUBIA
= 1, a borrow did not occur.
= 0, a borrow occurred.
Note : A subtraction is executed by adding the two's complement of the second operand. For rotate (RRR, RLR) instructions, this bit is loaded with either the high or low order bit of the source register.

DC : Half carry/half borrow bit.
ADDAR, ADDIA
= 1, a carry from the 4th low order bit of the result occurred.
= 0, a carry from the 4th low order bit of the result did not occur.
SUBAR, SUBIA
= 1, a borrow from the 4th low order bit of the result did not occur.
= 0, a borrow from the 4th low order bit of the result occurred.

Z : Zero bit.
= 1, the result of a logic operation is zero.
= 0, the result of a logic operation is not zero.

$\overline{PD}$: Power down flag bit.
= 1, after power-up or by the CLRWDT instruction.
= 0, by the SLEEP instruction.

$\overline{TO}$: Time overflow flag bit.
= 1, after power-up or by the CLRWDT or SLEEP instruction.
= 0, a watch-dog time overflow occurred.

RP0 : Register bank select bit (used for direct addressing).
= 1, Bank 1 is selected.
= 0, Bank 0 is selected.

Bit7:Bit6 : Not used. Read as "0"s.

2.1.5 FSR (Indirect Data Memory Address Pointer)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
04h (r/w)	FSR	IRP0	Indirect data memory address pointer						

Bit6:Bit0 : Select registers address in the indirect addressing mode. See 2.1.1 for detail description.

IRP0 : Register bank select bit (used for indirect addressing).
= 1, Bank 1 is selected.
= 0, Bank 0 is selected.

2.1.6 PORTA, PORTC (Port Data Registers) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
05h (r/w)	PORTA	-	-	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
07h (r/w)	PORTC	-	-	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0

Reading the port (PORTA, PORTC register) reads the status of the pins independent of the pin's input/output modes. Writing to these ports will write to the port data latch.
Both PORTA and PORTC are 6-bit port data Register. Only the low order 6 bits are used (PORTA<5:0>, PORTC<5:0>). Bits 7-6 are unimplemented and read as '0's. IOA3 is input only.

IOA5:IOA0 : PORTA I/O pin.
= 1, Port pin is high level.
= 0, Port pin is low level.

Bit7:Bit6 (PORTA) : Not used. Read as "0"s.

IOC5:IOC0 : PORTC I/O pin.
= 1, Port pin is high level.
= 0, Port pin is low level.

Bit7:Bit6 (PORTC) : Not used. Read as "0"s.

2.1.7 PCHBUF (High Byte Buffer of Program Counter)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Ah (r/w)	PCHBUF	-	-	-	-	-	3 MSBs Buffer of PC		

Bit2:Bit0 : See section 2.1.3 for detail description.

Bit7:Bit3 : Not used. Read as "0"s.

2.1.8 INTCON (Interrupt Control Register)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Bh (r/w)	INTCON	GIE	PEIE	T0IE	INTIE	PAIE	T0IF	INTIF	PAIF

PAIF : Port A input change interrupt flag. Set when Port A input changes or by software, reset by software.

INTIF : External INT pin interrupt flag. Set by rising/falling (selected by INTEDG bit (OPTION_REG<6>)) edge on INT pin or by software, reset by software.

T0IF : Timer0 overflow interrupt flag. Set when Timer0 overflows or by software, reset by software.

PAIE : Port A input change interrupt enable bit.
= 1, Enable the Port A input change interrupt.
= 0, Disable the Port A input change interrupt.

INTIE : External INT pin interrupt enable bit.
= 1, Enable the External INT pin interrupt.
= 0, Disable the External INT pin interrupt.

T0IE : Timer0 overflow interrupt enable bit.
= 1, Enable the Timer0 overflow interrupt.
= 0, Disable the Timer0 overflow interrupt.

PEIE : Peripheral Interrupt Enable bit.
= 1, Enable all un-masked peripheral interrupts.
= 0, Disable all peripheral interrupts.

GIE : Global interrupt enable bit.
= 1, Enable all un-masked interrupts. For wake-up from SLEEP mode through an interrupt event, the device will branch to the interrupt address (004h).
= 0, Disable all interrupts. For wake-up from SLEEP mode through an interrupt event, the device will continue execution at the instruction after the SLEEP instruction.
Note : When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts. The RETFIE instruction will exit the interrupt routine and set the GIE bit to re-enable interrupt.

2.1.9 INTFLAG (Peripheral Interrupt Status Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Ch (r/w)	INTFLAG	EEIF	IRIF	CCPIF	CMP2IF	CMP1IF	OSFIF	T2IF	T1IF

T1IF : Timer1 overflow interrupt flag. Set when Timer1 overflows or by software, reset by software.

T2IF : Timer2 match to PR2 interrupt flag. Set when TMR2 register matches to PR2 register or by software, reset by software.

OSFIF : Oscillator fail interrupt flag. Set when system oscillator failed, clock input has changed to IRC or by software, reset by software.

CMP1IF : Comparator 1 interrupt flag. Set when comparator 1 output has changed or by software, reset by software.

CMP2IF : Comparator 2 interrupt flag. Set when comparator 2 output has changed or by software, reset by software.

CCPIF : CCP interrupt flag.
Capture Mode : Set when a TMR1 capture occurred or by software, reset by software.
Compare Mode : Set when a TMR1 compare match occurred or by software, reset by software.
PWM Mode : Unused in this mode.

IRIF : IR counter match interrupt flag. Set when IROUT counter matches to IRCPR register or by software, reset by software.

EEIF : EEPROM Write Operation interrupt flag. Set when the write operation completed or by software, reset by software.

2.1.10 TMR1 (Timer 1 Clock/Counter Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Eh (r/w)	TMR1L	Low byte of 16-bit real-time clock/counter 1							
0Fh (r/w)	TMR1H	High byte of 16-bit real-time clock/counter 1							

The Timer1 is an 16-bit timer/counter. The clock source of Timer1 can come from the instruction cycle clock or by an external clock source (T1CKI pin) defined by T1CS bit (T1CON<1>). If T1CKI pin is selected, the Timer1 is increased by T1CKI signal rising edge.

2.1.11 T1CON (Timer 1 Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
10h (r/w)	T1CON	T1GINV	T1GE	T1PS1	T1PS0	T1OSC	T1SYNC	T1CS	T1ON

T1ON : Timer1 on/off bit.
= 1, Enables Timer1.
= 0, Stops Timer1.

T1CS : Timer1 clock source select bit.
= 1, External clock from T1CKI pin (on the rising edge).
= 0, Internal instruction clock cycle.

T1SYNC : Timer1 external clock input synchronization control bit.
If T1CS = 1:
= 1, Do not synchronize external clock input.
= 0, Synchronize external clock input.
If T1CS = 0:
This bit is ignored. Timer1 uses the internal clock.

T1OSC : LF oscillator enable control bit.
If IRC mode without OSCO output is active:
= 1, LF oscillator is enabled for Timer1 clock.
= 0, LF oscillator is off.
Else:
This bit is ignored.

T1PS1:T1PS0 : Timer1 prescaler rate select bits.

T1PS1:T1PS0	Timer1 Rate
0 0	1:1
0 1	1:2
1 0	1:4
1 1	1:8

T1GE: Timer1 gate enable bit.
If T1ON = 0:
This bit is ignored.
If T1ON = 1:
= 1, Timer1 is on if T1GB pin is not active.
= 0, Timer1 is on.

T1GINV : Timer1 gate inverter bit.
= 1, Timer1 gate is active high (Timer1 counts when T1GB is high).
= 0, Timer1 gate is active low (Timer1 counts when T1GB is low).

2.1.12 TMR2 (Timer 2 Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
11h (r/w)	TMR2	TMR27	TMR26	TMR25	TMR24	TMR23	TMR22	TMR21	TMR20

TMR27:TMR20 : Timer 2 register and increase until the value matches to PR2 register, and then reset to “0”.

2.1.13 T2CON (Timer 2 Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
12h (r/w)	T2CON	-	T2OPS3	T2OPS2	T2OPS1	T2OPS0	T2ON	T2PS1	T2PS0

T2PS1:T2PS0 : Timer2 prescaler select bits.
= 00 → Prescaler is 1
= 01 → Prescaler is 4
= 1X → Prescaler is 16

T2ON : Timer2 on/off bit.
= 1, Enables Timer2.
= 0, Stops Timer2.

T2OPS3:T2OPS0 : Timer2 output postscaler select bits.
= 0000 → 1:1 Postscaler
= 0001 → 1:2 Postscaler
= 0010 → 1:3 Postscaler
= 0011 → 1:4 Postscaler
= 0100 → 1:5 Postscaler
= 0101 → 1:6 Postscaler
= 0110 → 1:7 Postscaler
= 0111 → 1:8 Postscaler
= 1000 → 1:9 Postscaler
= 1001 → 1:10 Postscaler
= 1010 → 1:11 Postscaler
= 1011 → 1:12 Postscaler
= 1100 → 1:13 Postscaler
= 1101 → 1:14 Postscaler
= 1110 → 1:15 Postscaler
= 1111 → 1:16 Postscaler

Bit7 : Not used. Read as “0”.

2.1.14 CCPR (CCP Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
13h (r/w)	CCPRL	Capture/Compare/PWM register low byte							
14h (r/w)	CCPRH	Capture/Compare/PWM register high byte							

In Capture Mode, **CCPRH:CCPRL** captures the 16-bit value of TMR1 register when an event occurs on CCP pin.
In Compare Mode, the 16-bit **CCPRH:CCPRL** register value is constantly compared against the TMR1 register value.
In PWM Mode, the 8-bit **CCPRL** register is the eight MSBs of the PWM duty cycle, and the 8-bit **CCPRH** register is the synchronized buffer of the 8-bit **CCPRL** register **CCPRL** .

2.1.15 CCPCON (Capture/Compare/PWM Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
15h (r/w)	CCPCON	PWMMOD1	PWMMOD0	DCB1	DCB0	CCPMOD3	CCPMOD2	CCPMOD1	CCPMOD0

CCPMOD3:CCPMOD0 : CCP mode selection bits.
= 0000 → Capture/Compare/PWM off (disable CCP module)
= 0001 → Unused
= 0010 → Compare mode, toggle output on match (CCPIF is set)
= 0011 → Unused
= 0100 → Capture mode, every falling edge
= 0101 → Capture mode, every rising edge
= 0110 → Capture mode, every 4th rising edge
= 0111 → Capture mode, every 16th rising edge
= 1000 → Compare mode, set output on match (CCPIF is set)
= 1001 → Compare mode, clear output on match (CCPIF is set)
= 1010 → Compare mode, generate S/W interrupt on match (CCPIF is set, CCP pin is unaffected)
= 1011 → Compare mode, trigger special event on match (CCPIF is set, CCP resets TMR1)
= 1100 → PWM mode; PWMA, PWMC active high; PWMB, PWMD active high
= 1101 → PWM mode; PWMA, PWMC active high; PWMB, PWMD active low
= 1110 → PWM mode; PWMA, PWMC active low; PWMB, PWMD active high
= 1111 → PWM mode; PWMA, PWMC active low; PWMB, PWMD active low

DCB1:DCB0 :
Capture Mode : Unused.
Compare Mode : Unused.
PWM Mode : These bits are the two LSBs of the PWM duty cycle. The eight MSBs are found in CCPRL.

PWMMOD1:PWMMOD0 :
Capture Mode : Unused, CCP/PWMA assigned as Capture input; PWMB, PWMC, PWMD assigned as I/O pins.
Compare Mode : Unused, CCP/PWMA pin is assigned as Compare input; PWMB, PWMC, PWMD assigned as I/O pins.
PWM Mode : PWM output mode selection bits.
= 00 → Single output; PWMA modulated; PWMB, PWMC, PWMD assigned as I/O pins.
= 01 → Full-bridge output forward; PWMD modulated; PWMA active; PWMB, PWMC inactive.
= 10 → Half-bridge output; PWMA, PWMB modulated with dead-band delay control (see PWMCON register); PWMC, PWMD assigned as I/O pins.
= 11 → Full-bridge output reverse; PWMB modulated; PWMC active; PWMA, PWMD inactive.

2.1.16 PWMCON (PWM Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
16h (r/w)	PWMCON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0

PDC6:PDC0 : PWM dead-band delay count bits. These bits define the time interval between a PWM signal **should** transition active and the **actual** time it transitions active.
Dead Band Delay Time = (PDC6:PDC0) x 4 x Tosc, which Tosc = system oscillator time period

PRSEN : PWM restart enable bit.
= 1, Upon auto-shutdown, the CCPASE bit clears automatically once the shutdown event goes away; the PWM restarts automatically.
= 0, Upon auto-shutdown, the CCPASE bit must be cleared by Firmware to restart the PWM.

2.1.17 CCPAS (Capture/Compare/PWM Auto-Shutdown Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
17h (r/w)	CCPAS	CCPASE	CCPAS2	CCPAS1	CCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0

PSSBD1:PSSBD0 : Pins PWMB & PWMD shutdown state control bits.
= 00 → Drive to “0”.
= 01 → Drive to “1”.
= 1X → Tri-state.

PSSAC1:PSSAC0 : Pins PWMA & PWMC shutdown state control bits.
= 00 → Drive to “0”.
= 01 → Drive to “1”.
= 1X → Tri-state.

CCPAS2:CCPAS0 : CCP auto-shutdown source select bits.
= 000 → Auto-shutdown is disabled.
= 001 → Comparator 1 output change.
= 010 → Comparator 2 output change.
= 011 → Either comparator 1 or 2 output change.
= 100 → “Low level” on INT pin.
= 101 → “Low level” on INT pin or comparator 1 output change.
= 110 → “Low level” on INT pin or comparator 2 output change.
= 111 → “Low level” on INT pin or comparator 1 or 2 output change.

CCPASE : CCP auto-shutdown event status bit.
= 1, A shutdown event has occurred; CCP outputs are in shutdown state.
= 0, CCP outputs are operating.

2.1.18 WDTCON (Watch-dog Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
18h (r/w)	WDTCON	-	-	-	WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTE

WDTE : WDT (watch-dog timer) S/W enable bit.
= 1, Enable WDT.
= 0, Disable WDT.

WDTPS3:WDTPS0 : Watchdog timer period select bits.
= 0000 → 1:32 prescale rate
= 0001 → 1:64 prescale rate
= 0010 → 1:128 prescale rate
= 0011 → 1:256 prescale rate
= 0100 → 1:512 prescale rate (Reset value)
= 0101 → 1:1024 prescale rate
= 0110 → 1:2048 prescale rate
= 0111 → 1:4096 prescale rate
= 1000 → 1:8192 prescale rate
= 1001 → 1:16384 prescale rate
= 1010 → 1:32768 prescale rate
= 1011 → 1:65536 prescale rate
= 1100 → Reserved
= 1101 → Reserved
= 1110 → Reserved
= 1111 → Reserved

Bit7:Bit5 : Not used. Read as “0”s.

2.1.19 CMPCON1 (Comparator Control Register 1) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
19h (r/w)	CMPCON1	C2OUT	C1OUT	C2INV	C1INV	CINS	CM2	CM1	CM0

CM2:CM0 : Comparator mode selection bits. See FIGURE 2.9 for detail description.
= 000 → Comparator Reset (low power)
= 001 → Comparator with Output
= 010 → Comparator without Output
= 011 → Comparator with Output, with CV_{REF}
= 100 → Comparator without Output, with CV_{REF}
= 101 → Multiplexed Input with Output, with CV_{REF}
= 110 → Multiplexed Input without Output, with CV_{REF}
= 111 → Comparator Off (lowest power)

CINS : Comparator input switch bit.
When CM2:CM0 = 001 or 010:
= 1, VIN- connects to CIN+.
= 0, VIN- connects to CIN-.

C1INV : Comparator 1 output polarity inversion bit.
= 1, C1 output inverted.
= 0, C1 output not inverted.

C2INV : Comparator 2 output polarity inversion bit.
= 1, C2 output inverted.
= 0, C2 output not inverted.

C1OUT : Comparator 1 output bit.
If C1INV = 0:
= 1, VIN+ > VIN-.
= 0, VIN+ < VIN-.
If C1INV = 1:
= 1, VIN+ < VIN-.
= 0, VIN+ > VIN-.

C2OUT : Comparator 2 output bit.
If C2INV = 0:
= 1, VIN+ > VIN-.
= 0, VIN+ < VIN-.
If C2INV = 1:
= 1, VIN+ < VIN-.
= 0, VIN+ > VIN-.

2.1.20 CMPCON2 (Comparator Control Register 2) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ah (r/w)	CMPCON2	-	-	-	-	-	-	T1GSS	C2SYNC

C2SYNC : Comparator 2 output synchronization control bit. (Refer to Figure **2.26**)
= 1, Output is synchronized with falling edge of Timer 1 clock.
= 0, Output is asynchronous.

T1GSS : Timer 1 gate source select bit. (Refer to Section **2.8.5** "Timer 1 Gate")
= 1, Timer 1 gate source is T1GB pin (pin should be configured as digital input).
= 0, Timer 1 gate source is comparator 2 output.

Bit7:Bit2 : Not used. Read as "0"s.

2.1.21 IRCYCLE (IROUT Cycle Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Bh (r/w)	IRCYCLE	IRC7	IRC6	IRC5	IRC4	IRC3	IRC2	IRC1	IRC0

IRC7:IRC0 : IROUT (IR Carrier output) frequency = (IR clock source frequency) / (IRC7:IRC0).

2.1.22 IRDUTY (IROUT Duty Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ch (r/w)	IRDUTY	IRD7	IRD6	IRD5	IRD4	IRD3	IRD2	IRD1	IRD0

IRD7:IRD0 : IROUT (IR Carrier output) duty cycle = (IRD7:IRD0) / (IRC7:IRC0).
(IRD7:IRD0) must be less than (IRC7:IRC0).

2.1.23 IRCPR (IROUT Counter Pre-set Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Dh (r/w)	IRCPR	IRCPR7	IRCPR6	IRCPR5	IRCPR4	IRCPR3	IRCPR2	IRCPR1	IRCPR0

IRP7:IRP0 : IROUT counter pre-set bits. IROUT counter increase on every leading edge of internal IR pulse until the value of IR counter matches to IRCPR register, and then the IR counter will be reset to “0”, and increase again.

2.1.24 IRCON (IROUT Control Register) (Bank 0)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Eh (r/w)	IRCON	IREN	IROEN	IRCEN	IRSC	-	-	IRPS1	IRPS0

IREN : IOC2/IROUT pin select bit.
= 1, IROUT is selected and IR module is enabled.
= 0, IOC2 is selected and IR module is disabled.

IROEN :IROUT output enable bit.
= 1, IROUT is enabled.
= 0, IROUT is disabled.

IRCEN :IROUT counter enable bit.
= 1, IROUT counter is enabled and start to count.
= 0, IROUT counter is disabled and be reset to “0”.

IRSC : IROUT pin drive/sink current select bit.
= 1, Heavy.
= 0, Normal.

Bit3:Bit2 : Not used. Read as “0”s.

IRPS1:IRPS0 : IR module clock source prescaler bits.

IRPS1 : IRPS0	IR Module Clock Source Frequency
0, 0	Oscillator Frequency / 1
0, 1	Oscillator Frequency / 2
1, 0	Oscillator Frequency / 4
1, 1	Oscillator Frequency / 8

2.1.25 OPTION-REG Register (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
01h (r/w)	OPTION_REG	/PHA	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

PS2:PS0 : Timer0/WDT prescaler rate select bits.

PS2:PS0	Timer0 Rate	WDT Rate
0 0 0	1:2	1:1
0 0 1	1:4	1:2
0 1 0	1:8	1:4
0 1 1	1:16	1:8
1 0 0	1:32	1:16
1 0 1	1:64	1:32
1 1 0	1:128	1:64
1 1 1	1:256	1:128

PSA : Prescaler assign bit.
= 1, WDT (watch-dog timer).
= 0, TMR0 (Timer0).

T0SE : TMR0 source edge select bit.
= 1, Falling edge on T0CKI pin.
= 0, Rising edge on T0CKI pin.

T0CS : TMR0 clock source select bit.
= 1, External T0CKI pin.
= 0, internal instruction clock cycle.

INTEDG : Interrupt edge select bit.
= 1, interrupt on rising edge of INT pin.
= 0, interrupt on falling edge of INT pin.

/PHA : Port A pull-high enable bit.
= 1, Port A pull-high is disabled.
= 0, Port A pull-high is enabled by individual pull-high control bits (APHCON register).

2.1.26 IOSTA & IOSTC (Port I/O Control Registers) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
05h (r/w)	IOSTA	-	-	IOSTA5	IOSTA4	*	IOSTA2	IOSTA1	IOSTA0
07h (r/w)	IOSTC	-	-	IOSTC5	IOSTC4	IOSTC3	IOSTC2	IOSTC1	IOSTC0

IOSTA5:IOSTA0 : Port A I/O control bit.
= 1, PORTA pin configured as an input (tri-stated).
= 0, PORTA pin configured as an output.
Note: IOSTA3 always reads “1”.

Bit3 (IOSTA) : Not used. Read as “1”.

Bit7:Bit6 (IOSTA) : Not used. Read as “0”s.

IOSTC5:IOSTC0 : Port C I/O control bit.
= 1, PORTC pin configured as an input (tri-stated).

= 0, PORTC pin configured as an output.

Bit7:Bit6 (IOSTC) : Not used. Read as “0”s.

2.1.27 INTEN (Peripheral Interrupt Mask Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Ch (r/w)	INTEN	EEIE	IRIE	CCPIE	CMP2IE	CMP1IE	OSFIE	T2IE	T1IE

T1IE : Timer1 overflow interrupt enable bit.
= 1, Enable the Timer1 overflow interrupt.
= 0, Disable the Timer1 overflow interrupt.

T2IE : Timer2 match to PR2 interrupt enable bit.
= 1, Enable the Timer2 match to PR2 interrupt.
= 0, Disable the Timer2 match to PR2 interrupt.

OSFIE : Oscillator fail interrupt enable bit.
= 1, Enable the Oscillator fail interrupt.
= 0, Disable the Oscillator fail interrupt.

CMP1IE : Comparator 1 interrupt enable bit.
= 1, Enable the comparator 1 interrupt.
= 0, Disable the comparator 1 interrupt.

CMP2IE : Comparator 2 interrupt enable bit.
= 1, Enable the comparator 2 interrupt.
= 0, Disable the comparator 2 interrupt.

CCPIE : CCP interrupt enable bit.
= 1, Enable the CCP interrupt.
= 0, Disable the CCP interrupt.

IRIE : IROUT counter match interrupt enable bit.
= 1, Enable the IROUT counter match interrupt.
= 0, Disable the IROUT counter match interrupt.

EEIE : EEPROM Write Complete Interrupt Enable bit.
= 1, Enable the EEPROM Write Complete interrupt.
= 0, Disable the EEPROM Write Complete interrupt.

2.1.28 PCON (Power Control Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Eh (r/w)	PCON	-	-	ULPWUE	LVDTE	-	-	$\overline{\text{POR}}$	$\overline{\text{BOD}}$

$\overline{\text{BOD}}$: Brown-out detect bit.
= 1, No brown-out occurred.
= 0, A brown-out occurred. Must be set by software after a brown-out occurred.

$\overline{\text{POR}}$: Power-up reset detect bit.
= 1, No power-up reset occurred.
= 0, A power-up reset occurred. Must be set by software after a power-up reset occurred.

Bit3:Bit2 : Not used. Read as “0”s.

LVDTE : LVDT (low voltage detector) enable bit.
= 1, Enable LVDT.
= 0, Disable LVDT.

ULPWUE : Ultra lower-power wake-up (ULPWU) enable bit.
= 1, Enable ultra lower-power wake-up on IOA0.
= 0, Disable ultra lower-power wake-up on IOA0.

Bit7:Bit6 : Not used. Read as “0”s.

2.1.29 OSCCON (Oscillator Control Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
0Fh (r/w)	OSCCON	-	IRCF2	IRCF1	IRCF0	OSTS	HTS	LTS	SCS

SCS : System clock select bit.
= 1, IRC (HFIRC or LFIRC) is used for system clock.
= 0, Clock source defined by Fosc<2:0> of the CONFIG register.

LTS : LFIRC status bit (low frequency ~ 31KHz).
= 1, LFIRC is stable.
= 0, LFIRC is not stable.

HTS : HFIRC status bit (high frequency = 8MHz to 125KHz).
= 1, HFIRC is stable.
= 0, HFIRC is not stable.

OSTS : Oscillator start-up time-out status bit. Bit resets to ‘0’ with Two-Speed Start-up and LP, XT or HS selected as the Oscillator mode or Fail-Safe mode is enabled.
= 1, Device is running from the clock source defined by Fosc<2:0> of the CONFIG register.
= 0, Device is running from the IRC oscillator (HFIRC or LFIRC).

IRCF2:IRCF0 : IRC oscillator frequency select bits.
= 111 → 8MHz.
= 110 → 4MHz (default).
= 101 → 2MHz.
= 100 → 1MHz.
= 011 → 500KHz.
= 010 → 250KHz.
= 001 → 125KHz.
= 000 → 31KHz (LFIRC).

Bit7 : Not used. Read as “0”.

2.1.30 OSCTUNE (Oscillator Tunning Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
10h (r/w)	OSCTUNE	-	-	-	TUN4	TUN3	TUN2	TUN1	TUN0

TUN4:TUN0 : HFIRC frequency tunning bits.
= 01111 → Maximum frequency.
= 01110
•
•
•
= 00001
= 00000 → HFIRC is running at the calibrated frequency.
= 11111 →
•
•
•
= 10001 →
= 10000 → Minimum frequency.

Bit7:Bit5 : Not used. Read as "0"s.

2.1.31 PR2 (Timer2 Period Control Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
12h (r/w)	PR2	PR27	PR26	PR25	PR24	PR23	PR22	PR21	PR20

PR27:PR20 : Timer 2 period register.

2.1.32 APHCON (PortA Pull-high Control Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
15h (r/w)	APHCON	-	-	PHA5	PHA4	-	PHA2	PHA1	PHA0

PHA0 : = 1, Enable the internal pull-high of IOA0 pin.
= 0, Disable the internal pull-high of IOA0 pin.

PHA1 : = 1, Enable the internal pull-high of IOA1 pin.
= 0, Disable the internal pull-high of IOA1 pin.

PHA2 : = 1, Enable the internal pull-high of IOA2 pin.
= 0, Disable the internal pull-high of IOA2 pin.

Bit3 : Not used. Read as "0".

PHA4 : = 1, Enable the internal pull-high of IOA4 pin.
= 0, Disable the internal pull-high of IOA4 pin.

PHA5 : = 1, Enable the internal pull-high of IOA5 pin.
= 0, Disable the internal pull-high of IOA5 pin.

Bit7:Bit6 : Not used. Read as "0"s.

2.1.33 AWUCON (Port A Input Change Interrupt/Wake-up Control Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
16h (r/w)	AWUCON	-	-	WUA5	WUA4	WUA3	WUA2	WUA1	WUA0

WUA0 : = 1, Enable the input change interrupt/wake-up function of IOA0 pin.
= 0, Disable the input change interrupt/wake-up function of IOA0 pin.

WUA1 : = 1, Enable the input change interrupt/wake-up function of IOA1 pin.
= 0, Disable the input change interrupt/wake-up function of IOA1 pin.

WUA2 : = 1, Enable the input change interrupt/wake-up function of IOA2 pin.
= 0, Disable the input change interrupt/wake-up function of IOA2 pin.

WUA3 : = 1, Enable the input change interrupt/wake-up function of IOA3 pin.
= 0, Disable the input change interrupt/wake-up function of IOA3 pin.

WUA4 : = 1, Enable the input change interrupt/wake-up function of IOA4 pin.
= 0, Disable the input change interrupt/wake-up function of IOA4 pin.

WUA5 : = 1, Enable the input change interrupt/wake-up function of IOA5 pin.
= 0, Disable the input change interrupt/wake-up function of IOA5 pin.

Bit7:Bit6 : Not used. Read as “0”s.

2.1.34 CMPCON3 (Comparator Control Register 3) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
19h (r/w)	CMPCON2	CVREN	-	CVRR	-	CVR3	CVR2	CVR1	CVR0

CVR3:CVR0 : Comparator voltage reference (CV_{REF}) selection bits.
If CVRR = 1: CV_{REF} = (CVR3:CVR0) * Vdd / 24.
If CVRR = 0: CV_{REF} = Vdd / 4 + (CVR3:CVR0) * Vdd / 32.

Bit4 : Not used. Read as “0”.

CVRR : CV_{REF} range selection bit.
= 1, Low range.
= 0, High range.

Bit6 : Not used. Read as “0”.

CVREN : Comparator voltage reference (CV_{REF}) module enable bit.
= 1, Enable CV_{REF} module.
= 0, Disable CV_{REF} module.

2.1.35 EEDATA (EEPROM Data Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ah (r/w)	EEDATA	EED7	EED6	EED5	EED4	EED3	EED2	EED1	EED0

EED7:EED0 : Byte value to write to or read from Data EEPROM.

2.1.36 EEADR (EEPROM Address Register) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Bh (r/w)	EEADR	EEA7	EEA6	EEA5	EEA4	EEA3	EEA2	EEA1	EEA0

EEA7:EEA0 : Specifies one of 256 locations for EEPROM Read/Write Operation.

2.1.37 EECON1 (EEPROM Control Register 1) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Ch (r/w)	EECON1	-	-	-	WRONLY	WRERR	WREN	WR	RD

RD: EEPROM read control bit. This bit can only be set, not cleared in software.
= 1, Initiates an EEPROM read (cleared by hardware once read is complete).
= 0, Does not initiate an EEPROM read; or read data from EEPROM is finished.

WR : EEPROM write control bit. This bit can only be set, not cleared in software.
= 1, Initiates a write cycle (both erase and program cycle are included), cleared by hardware once write cycle is complete. ***This allows software changes the EEPROM data both from "1" to "0" and "0" to "1".***
= 0, Does not initiate an EEPROM write; or write cycle to the data EEPROM is complete.

WREN : EEPROM write or write-only enable bit.
= 1, Allows write or write-only cycles.
= 0, Inhibits write or write-only to the data EEPROM.

WRERR : EEPROM write error flag.
= 1, A write operation is prematurely terminated (any RSTB Reset, any WDT Reset during normal operation or Brown-out Reset).
= 0, The write operation completed.

WRONLY : EEPROM write-only control bit. This bit can only be set, not cleared in software.
= 1, Initiates a write-only cycle (only program cycle is included), cleared by hardware once write-only cycle is complete. ***This only allows software changes the EEPROM data from "0" to "1".***
= 0, Does not initiate an EEPROM write-only; or write-only cycle to the data EEPROM is complete.
Note: ***The WRONLY bit will be fixed to "0" by H/W if the configuration bit WRONLYE = 0.***

Bit7:Bit5 : Not used. Read as "0"s.

2.1.38 EECON2 (EEPROM Control Register 2) (Bank 1)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
1Dh (r/w)	EECON2	EEPROM control register 2 (not a physical register)							

The EECON2 register is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the Data EEPROM write sequence (write 55h to EECON2 register, then write AAh to EECON2, and then set WR bit).

2.1.39 ACC (Accumulator)

Address	Name	B7	B6	B5	B4	B3	B2	B1	B0
N/A (r/w)	ACC	Accumulator							

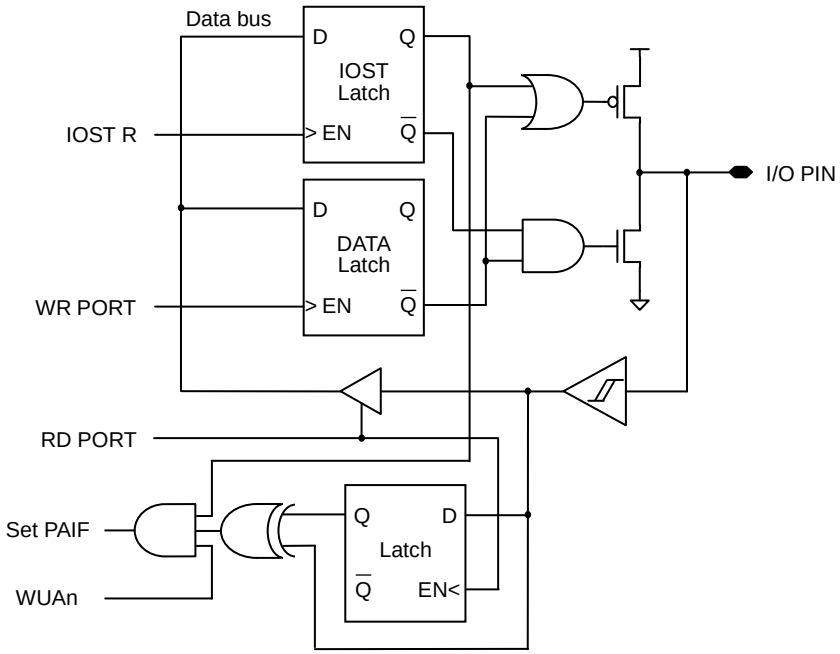
Accumulator is an internal data transfer, or instruction operand holding. It can not be addressed.

2.2 I/O Ports

Port A and Port C are bi-directional tri-state I/O ports. Both Port A and Port C are 6-pin I/O port. Please note that IOA3 is an input only pin.
All I/O pins have data direction control registers (IOSTA, IOSTC) which can configure these pins as output or input. The exceptions are IOA3 which is input only.
IOA<5:4> and IOA<2:0> have its corresponding pull-high control bits (APHCON register) to enable the weak internal pull-high. The weak pull-high is automatically turned off when the pin is configured as an output pin.
IOA<5:0> also provides the input change interrupt/wake-up function. Each pin has its corresponding input change interrupt/wake-up enable bits (AWUCON) to select the input change interrupt/wake-up source.
The IOA2 is also an external interrupt input signal.
The CONFIGURATION words can set several I/Os to alternate functions. When acting as alternate functions the pins will read as “0” during port read.

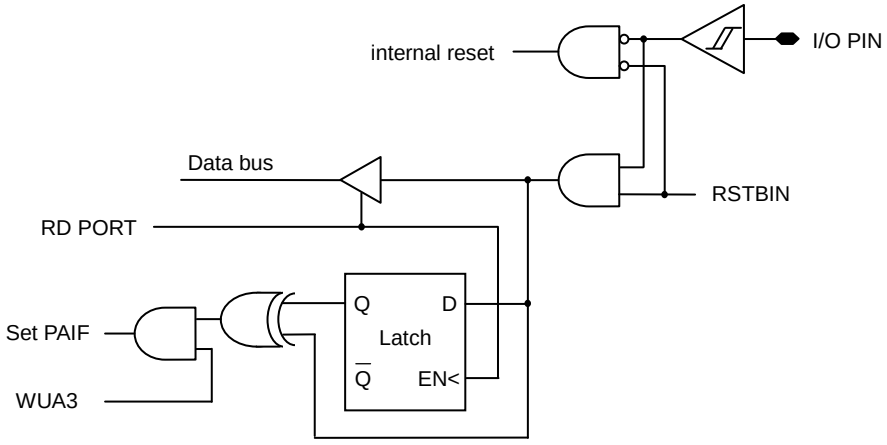
FIGURE 2.3: Block Diagram of I/O PINs

IOA5, IOA4, IOA2 ~ IOA0 :

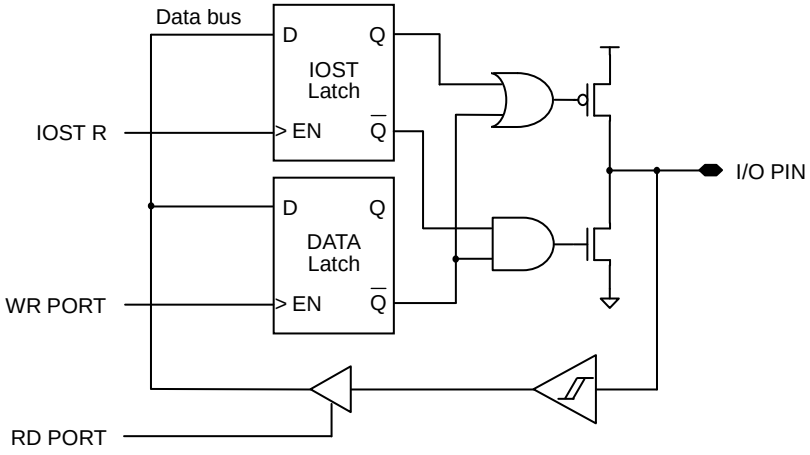


Pull-high / Analog input are not shown in the figure

IOA3 :



IOC5 ~ IOC0 :



2.3 Timer0/WDT & Prescler

2.3.1 Timer0

The Timer0 is a 8-bit timer/counter. The clock source of Timer0 can come from the internal clock or by an external clock source (T0CKI pin).

2.3.1.1 Using Timer0 with an Internal Clock : Timer mode

Timer mode is selected by clearing the T0CS bit (OPTION_REG<5>). In timer mode, the timer0 register (TMR0) will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles.

2.3.1.2 Using Timer0 with an External Clock : Counter mode

Counter mode is selected by setting the T0CS bit (OPTON<5>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The incrementing edge is determined by the source edge select bit T0SE (OPTION_REG<4>).
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.
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 T2 and T4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2 T_{OSC} and low for at least 2 Tosc.
When a prescaler is used, the external clock input is divided by the asynchronous prescaler. 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 divided by the prescaler value.

2.3.1.3 Timer0 Interrupt

An overflow (FFh → 00h) in the TMR0 register will set the flag bit T0IF (INTCON<2>). To enable the interrupt, T0IE bit (INTCON<5>), and GIE bit (INTCON<7>) should be set. The T0IF bit must be cleared by software before re-enabling this interrupt.

2.3.2 Watchdog Timer (WDT)

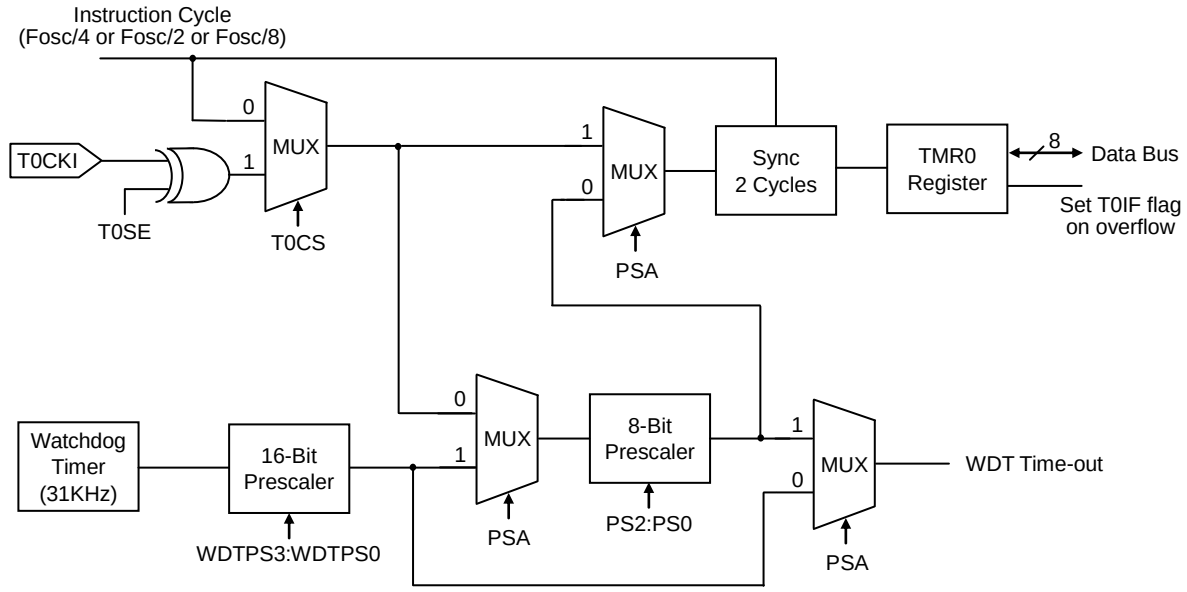
The Watchdog Timer (WDT) is a free running on-chip RC oscillator which does not require any external components. So the WDT will still run even if the clock on the OSCI and OSCO pins is turned off, such as in SLEEP mode. During normal operation, a WDT time-out will cause the device reset and the $\overline{TO}$ bit (STATUS<4>) will be cleared. 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 disabled by clearing the control bit WDTE (WDTCON<0>) to “0”.
The WDT has a nominal time-out period of 1 ms (1:32) to 2.1 second (1:65536) selected by WDTPS<3:0> bits (WDTCON<4:1>). If a longer time-out period is desired, the TMR0/WDT prescaler with a division ratio of up to 1:128 can be assigned to the WDT controlled by the OPTION_REG register. Thus, the WDT time-out period is from 1 ms to 268 seconds.
The CLRWDT instruction clears the WDT and the prescaler, if assigned to the WDT, and prevents it from timing out and generating a device reset.
The SLEEP instruction resets the WDT and the prescaler, if assigned to the WDT. This gives the maximum SLEEP time before a WDT Wake-up Reset.

2.3.3 Prescaler

An 8-bit counter (down counter) is available as a prescaler for the Timer0, or as a postscaler for the Watchdog Timer (WDT). Note that the prescaler may be used by either the Timer0 module or the WDT, but not both. Thus, a prescaler assignment for the Timer0 means that there is no prescaler for the WDT, and vice-versa.

The PSA bit (OPTION_REG<3>) determines prescaler assignment. The PS<2:0> bits (OPTION_REG<2:0>) determine prescaler ratio.
When the prescaler is assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler. When it is assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.
The prescaler is neither readable nor writable. On a RESET, the prescaler contains all '1's.
To avoid an unintended device reset, CLRWDT or CLRR TMR0 instructions must be executed when changing the prescaler assignment from Timer0 to the WDT, and vice-versa.

FIGURE 2.4: Block Diagram of the Timer0/WDT & Prescaler



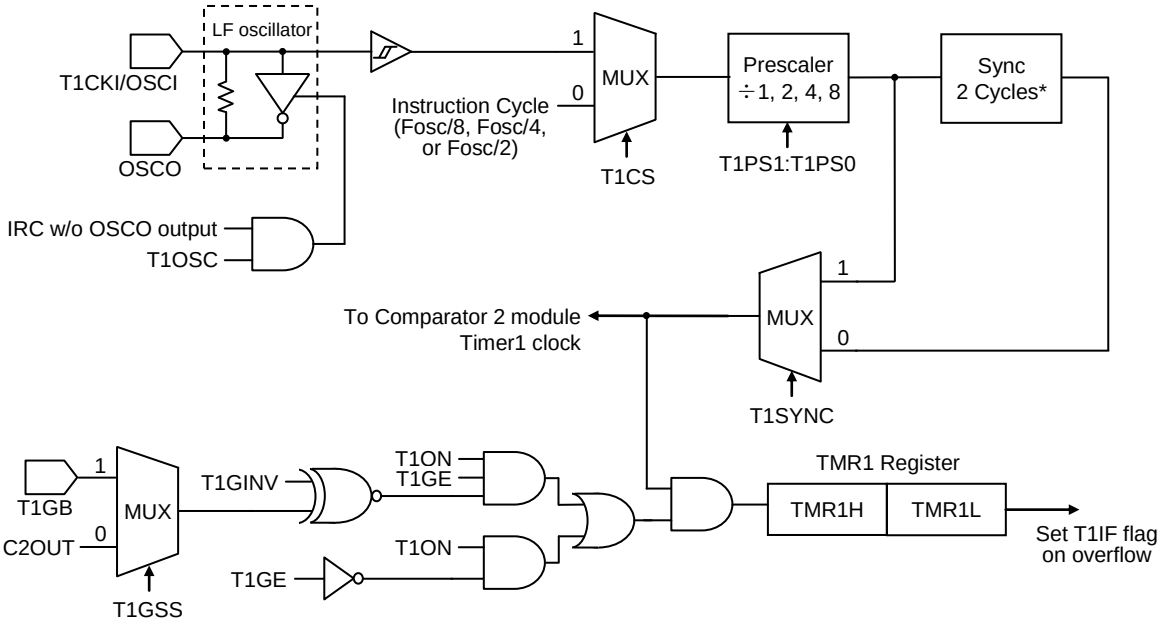
2.4 Timer1 with Gate Control

The Timer1 is a 16-bit timer/counter with the following features:
• 16-bit timer/counter (TMR1H:TMR1L)
• Readable and writable
• Internal or external clock selection
• Synchronous or asynchronous operation
• Interrupt on overflow from FFFFh to 0000h
• Wake-up upon overflow (external clock, Asynchronous mode only)
• Timer1 gate (count enable) via comparator 2 output or T1GB pin
• Optional LF oscillator
• Time base for the Capture/Compare function
• Special event trigger (with CCP)
The T1CON register is used to enable/disable Timer1 and select the various features of the Timer1 module.

2.4.1 Timer1 Modes of Operation

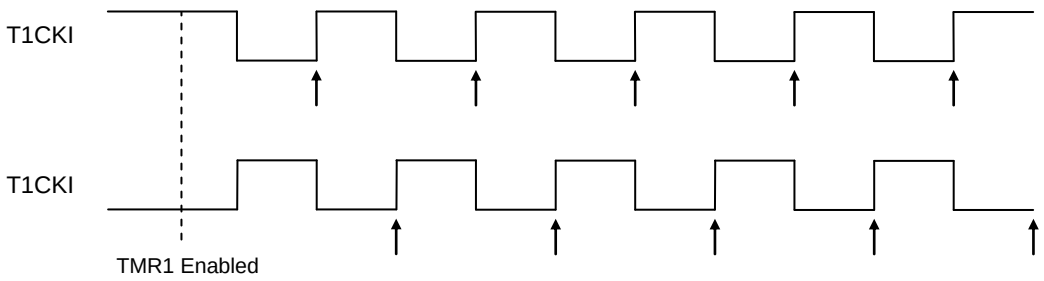
Timer1 can operate in one of three modes:
• 16-bit timer : T1SYNC, T1CS (T1CON<2:1>) = X, 0. Timer1 is incremented on every instruction cycle.
• 16-bit synchronous counter : T1SYNC, T1CS (T1CON<2:1>) = 0, 1. Timer1 is incremented on the rising edge of the external clock input T1CKI.
• 16-bit asynchronous counter : T1SYNC, T1CS (T1CON<2:1>) = 1, 1. Timer1 is incremented on the rising edge of the external clock input T1CKI.
In Counter and Timer modules, the counter/timer clock can be gated by the T1GB input.
If an external clock oscillator is needed (in IRC mode without OSCO output), Timer1 can use the LF oscillator as a clock source.
Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge.

FIGURE 2.5: Block Diagram of the Timer1



Note: Synchronization does not operate while in sleep.

FIGURE 2.6: Timer1 Incrementing Edge



Note 1: Arrows indicate counter increments.
2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

2.4.2 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4, or 8 divisions of the clock input. The T1PS1:T1PS0 bits (T1CON<5:4>) control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

2.4.3 Timer1 Operation in Asynchronous Counter Mode

If control bit T1SYNC (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer. (Section 5.4.1).
Note: The CMPCON1 registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read ‘0’.

2.4.3.1 Reading and Writing Timer1 in Asynchronous Counter Mode

Reading TMR1H or TMR1L, while the timer is running from an external asynchronous clock, will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.
For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the timer register.

2.4.4 Timer1 Oscillator

A crystal oscillator circuit is built-in between pins OSCI and OSCO. It is enabled by setting the T1OSC (T1CON<3>) bit. The Timer1 oscillator will continue to run during SLEEP. The oscillator requires a start-up and stabilization time before use. Thus, T1OSC should be set and a suitable delay observed prior to enabling Timer1.
The Timer1 oscillator is shared with the system LF oscillator. Thus, Timer1 can use this mode only when the system clock is derived from the internal RC oscillator (IRC mode) without OSCO output.
IOSTA5 and IOSTA4 bits are set when the Timer1 oscillator is enabled. IOA5 and IOA4 read as ‘0’ and IOSTA5 and IOSTA4 bits read as ‘1’.

2.4.5 Timer1 Gate

Timer1 gate source is S/W configurable to be the T1GB pin or the output of Comparator 2. This allows the device to directly time external events using T1GB or analog events using Comparator 2. T1GSS bit of the CMPCON2 register is used to select either T1GB or C2OUT as the Timer1 gate source. T1GE bit of the T1CON register must be

set to enable gate function.
Timer1 gate can be inverted using the T1GINV bit of the T1CON register, whether it originates from the T1GB pin or Comparator 2 output. This configures Timer1 to measure either the active-high or active-low time between events.

2.4.6 CCP Capture/Compare Time Base

The CCP module uses the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.
In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPRH:CCPRL register pair on a configured event.
In Compare mode, an event is triggered when the value CCPRH:CCPRL register pair matches the value in the TMR1H:TMR1L register pair. This event can be a special event trigger.

2.4.7 CCP Special Event Trigger

When the CCP is configured to trigger a special event, the trigger will clear the TMR1H:TMR1L register pair. This special event does not cause a Timer1 interrupt. The CCP module may still be configured to generate a CCP interrupt.
In this mode of operation, the CCPRH:CCPRL register pair effectively becomes the period register for Timer1.
Timer1 should be synchronized to the Fosc to utilize the special event trigger. Asynchronous operation of Timer1 can cause a special event trigger to be missed.
In the event that a write to TMR1H or TMR1L coincides with a special event trigger from the CCP, the write will take precedence.

2.4.8 Comparator 2 Synchronization

The same clock used to increment Timer1 can also be used to synchronize the comparator 2 output. This feature is enabled in the Comparator module.
When using the comparator 2 output for Timer1 gate, the comparator 2 output should be synchronized to Timer1. This ensures Timer1 does not miss an increment if the comparator changes.

2.4.9 Timer1 Interrupt

An overflow (FFFFh → 0000h) in the TMR1H:TMR1L register pair will set the flag bit T1IF (INTFLAG<0>). To enable the interrupt, T1IE bit (INTEN<0>), PEIE bit (INTCON<6>), and GIE bit (INTCON<7>) should be set.
The T1IF bit must be cleared by software before re-enabling this interrupt.

2.4.10 Timer1 Operation During SLEEP

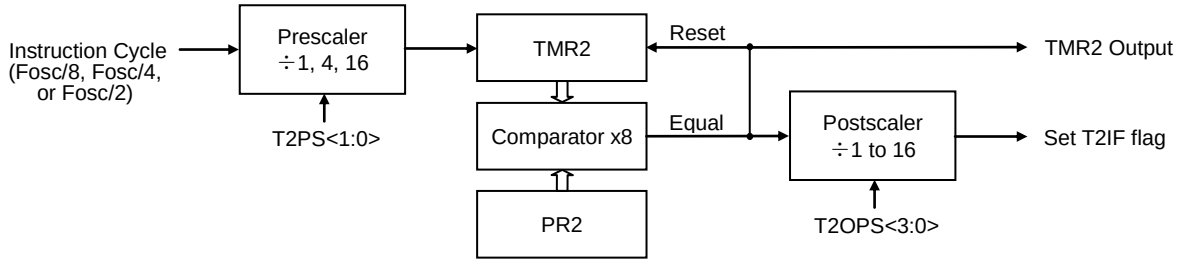
Timer1 can only operate during SLEEP when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter.
The device will wake-up on an overflow. If the GIE bit (INTCON<7>) is set, the device will wake-up and jump to the interrupt vector. If the GIE bit is cleared, the device will wake-up and continue execution at the instruction after the SLEEP instruction.

2.5 Timer2 Module

The Timer2 module is an eight-bit timer with the following features:
• 8-bit timer register (TMR2)
• 8-bit period register (PR2)
• Interrupt on TMR2 match to PR2
• Software programmable prescaler (1:1, 1:4, 1:16)
• Software programmable postscaler (1:1 to 1:16)

The clock input to the Timer2 module is the system instruction clock (Fosc/8, Fosc/4, or Fosc/2). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.
The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, TMR2 is reset to 00h on the next increment cycle and the Timer2 postscaler is incremented
The match output of the Timer2/PR2 comparator is fed into the Timer2 postscaler. The postscaler has postscale options of 1:1 to 1:16 inclusive. The output of the Timer2 postscaler is used to set the T2IF interrupt flag bit in the INTFLAG register.
The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.
Timer2 is turned on by setting the T2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the T2ON bit to a '0'.
The Timer2 prescaler is controlled by the T2PS bits in the T2CON register. The Timer2 postscaler is controlled by the T2OPS bits in the T2CON register. The prescaler and postscaler counters are cleared when:
• A write to TMR2 occurs.
• A write to T2CON occurs. Note: TMR2 is not cleared when T2CON is written.
• Any device Reset occurs.

FIGURE 2.7: Block Diagram of the Timer2



2.6 Capture/Compare/PWM Module (CCP) with Auto-Shutdown & Dead Band Delay Control

The Enhanced Capture/Compare/PWM module allows the user to time and control different events. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate a Pulse-Width Modulated signal of varying frequency and duty cycle.

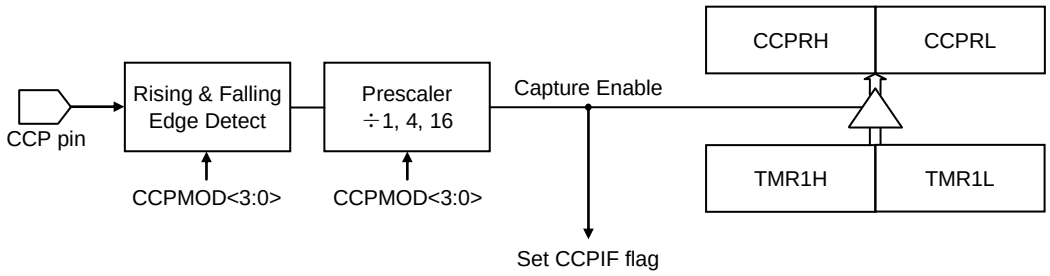
TABLE 2.1: CCP Mode – Timer Resources Required

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

2.6.1 Capture Mode

In Capture mode, CCPRH:CCPRL captures the 16-bit value of the TMR1 register when an event occurs on pin CCP. An event is defined as one of the following and is configured by the CCPMOD<3:0> bits of the CCPCON register:
• Every falling edge
• Every rising edge
• Every 4th rising edge
• Every 16th rising edge
When a capture is made, the Interrupt Request Flag bit CCPIF of the INTFLAG register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPRH:CCPRL register pair is read, the old captured value is overwritten by the new captured value.

FIGURE 2.8: Block Diagram of Capture Mode

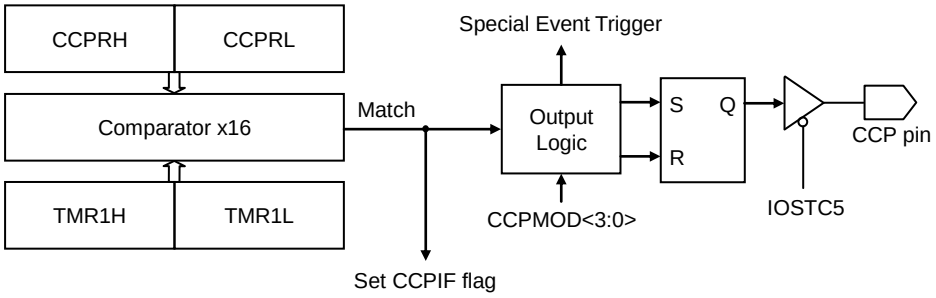


In Capture mode, the CCP pin should be configured as an input by setting the associated IOSTC5 control bit. If the CCP pin is configured as an output, a write to the port can cause a capture condition.
Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.
There are four prescaler settings specified by the CCPMOD<3:0> bits of the CCPCON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any reset will clear the prescaler counter.
Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCPCON register before changing the prescaler
When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPIE interrupt enable bit of the INTEN register clear to avoid false interrupts. Additionally, the user should clear the CCPIF interrupt flag bit of the INTFLAG register following any change in operating mode.

2.6.2 Compare Mode

In Compare mode, the 16-bit CCPRH:CCPRL register pair value is constantly compared against the TMR1 register pair value. When a match occurs, the CCP module may:
• Toggle the CCP output
• Set the CCP output
• Clear the CCP output
• Generate a special event trigger
• Generate a software interrupt
The action on the pin is based on the value of the CCPMOD<3:0> control bits of the CCPCON register.
All Compare modes can generate an interrupt.

FIGURE 2.9: Block Diagram of Compare Mode

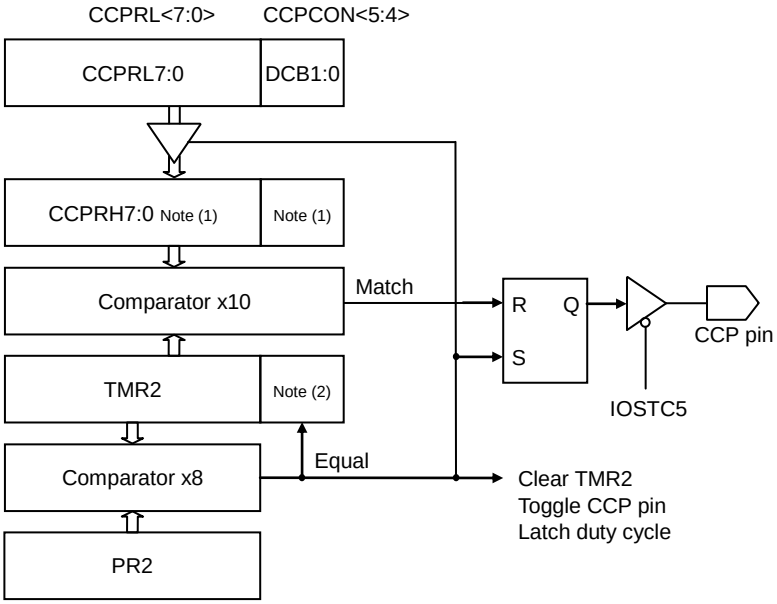


The user must configure the CCP pin as an output by clearing the associated IOSTC5 bit.
Clearing the CCPCON register will force the CCP compare output latch to the default low level. This is not the port I/O data latch.
In Compare mode, Timer1 must be running in either timer mode or synchronized counter mode. The compare operation may not work in asynchronous counter mode.
When generate software interrupt mode is chosen (CCPMOD<3:0> = 1010), the CCP module does not assert control of the CCP pin.
When special event trigger mode is chosen (CCPMOD<3:0> = 1011), the CCP module will resets Timer1, the CCP module does not assert control of the CCP pin. The special event trigger output of the CCP occurs immediately upon a match between the TMR1H:TMR1L register pair and the CCPR1H:CCPR1L register pair. The TMR1H:TMR1L register pair is not reset until the next rising edge of the Timer1 clock. This allows the CCPRH:CCPRL register pair to effectively provide a 16-bit programmable period register for Timer1.
The special event trigger from the CCP module does not set interrupt flag bit T1IF of the INTFLAG register.
Removing the match condition by changing the contents of the CCPRH and CCPRL register pair, between the clock edge that generates the special event trigger and the clock edge that generates the Timer1 reset, will preclude the reset from occurring.

2.6.3 PWM Mode

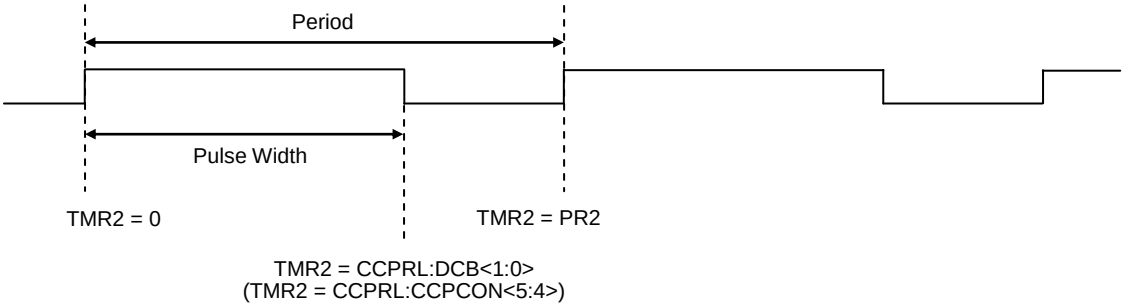
The PWM mode generates a Pulse-Width Modulated signal on the CCP pin. The duty cycle, period and resolution are determined by the PR2, T2CON, CCPRL, CCPCON registers.
In Pulse-Width Modulation (PWM) mode, the CCP module produces up to a 10-bit resolution PWM output on the CCP pin. Since the CCP pin is multiplexed with the PORT data latch, the IOST for that pin must be cleared to enable the CCP pin output driver.
Note: Clearing the CCPCON register will relinquish CCP control of the CCP pin.

FIGURE 2.10: Block Diagram of PWM Mode



Note 1: The CCPRH register and a 2-bit internal latch are used to double buffer the PWM duty cycle. The CCPRH is a read-only register in PWM mode.
2: The 8-bit TMR2 register is concatenated with either the 2-bit internal system clock (Fosc), or 2-bit of the prescaler, to create the 10-bit time base.

FIGURE 2.11: PWM Output



2.6.3.1 PWM Period

The PWM period is specified by the PR2 register of Timer2. The PWM period can be calculated using the formula below:

PWM Period =[PR2+ 1]•4•TOSC • (TMR2 Prescale Value)

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:
• TMR2 is cleared
• The CCP pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
• The PWM duty cycle is latched from CCPRL into CCPRH.
Note: The Timer2 postscaler is not used in the determination of the PWM frequency.

2.6.3.2 PWM Duty Cycle

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPRL register and DCB<1:0> bits of the CCPCON register. The CCPRL contains the eight MSBs and the DCB<1:0> bits of the CCPCON register contain the two LSBs. CCPRL and DCB<1:0> bits of the CCPCON register can be written to at any time. The duty cycle value is not latched into CCPRH until after the period completes (i.e., a match between PR2 and TMR2 registers occurs). While using the PWM, the CCPRH register is read-only.

Pulse Width =(CCPRL:CCPCON<5:4>) •TOSC • (TMR2 Prescale Value)

Duty Cycle Ratio =(CCPRL:CCPCON<5:4>)/4(PR2 + 1)

The CCPRH register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation. The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or 2 bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1. When the 10-bit time base matches the CCPRH and 2-bit latch, then the CCP pin is cleared. In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the CCP pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.
The following steps should be taken when configuring the CCP module for PWM operation:
1. Disable the PWM pin (CCP) output driver by setting the associated IOST bit.
2. Set the PWM period by loading the PR2 register.
3. Configure the CCP module for the PWM mode by loading the CCPCON register with the appropriate values.
4. Set the PWM duty cycle by loading the CCPRL register and DCB<1:0> bits of the CCPCON register.
5. Configure and start Timer2:
   • Clear the T2IF interrupt flag bit of the INTFLAG register.
   • Set the Timer2 prescale value by loading the T2PS<1:0> bits of the T2CON register.
   • Enable Timer2 by setting the T2ON bit of the T2CON register.
6. Enable PWM output after a new PWM cycle has started:
   • Wait until Timer2 overflows (T2IF bit of the INTFLAG register is set).
   • Enable the CCP pin output driver by clearing the associated IOST bit.

2.6.4 Enhanced PWM Mode

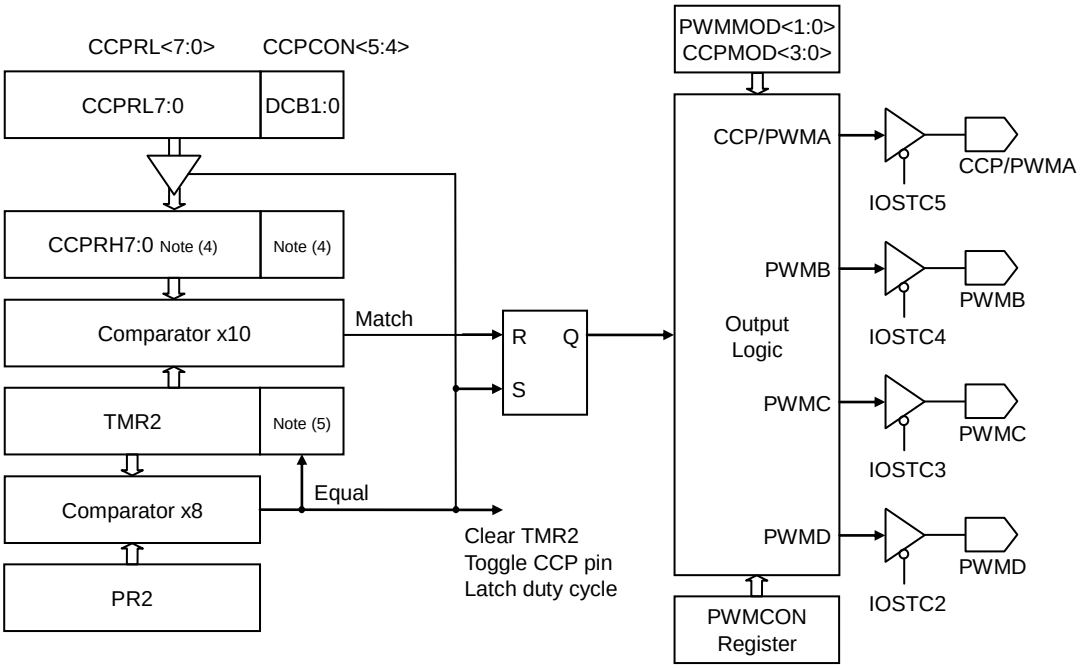
The Enhanced PWM Mode can generate a PWM signal on up to four different output pins with up to 10-bits of resolution. It can do this through four different PWM output modes:

- Single PWM
- Half-Bridge PWM
- Full-Bridge PWM, Forward mode
- Full-Bridge PWM, Reverse mode

To select an Enhanced PWM mode, the PWMMOD<1:0> bits of the CCPCON register must be set appropriately. The PWM outputs are multiplexed with I/O pins and are designated PWMA, PWMB, PWMC and PWMD. The polarity of the PWM pins is configurable and is selected by setting the CCPMOD bits in the CCPCON register appropriately.

To prevent the generation of an incomplete waveform when the PWM is first enabled, the CCP module waits until the start of a new PWM period before generating a PWM signal.

FIGURE 2.12: Block Diagram of Enhanced PWM Mode



Note 1: The IOST register value for each PWM output must be configured appropriately.
2: Clearing the CCPCON register will relinquish CCP control of all PWM output pins.
3: Any pin not used by an Enhanced PWM mode is available for alternate pin functions.
4: The CCPRH register and a 2-bit internal latch are used to double buffer the PWM duty cycle.
5: The 8-bit TMR2 register is concatenated with either the 2-bit internal system clock (Fosc), or 2-bit of the prescaler, to create the 10-bit time base.

TABLE 2.2: Pin Assignments for Various PWM Enhanced Modes

PWMMOD<1:0>	Enhanced PWM Mode	CCP/PWMA	PWMB	PWMC	PWMD
0, 0	Single	Yes	No	No	No
1, 0	Half-Bridge	Yes	Yes	No	No
0, 1	Full-Bridge, Forward	Yes	Yes	Yes	Yes
1, 1	Full-Bridge, Reverse	Yes	Yes	Yes	Yes

FIGURE 2.13: Example of Enhanced PWM Mode Output Relationships (Active-High State)

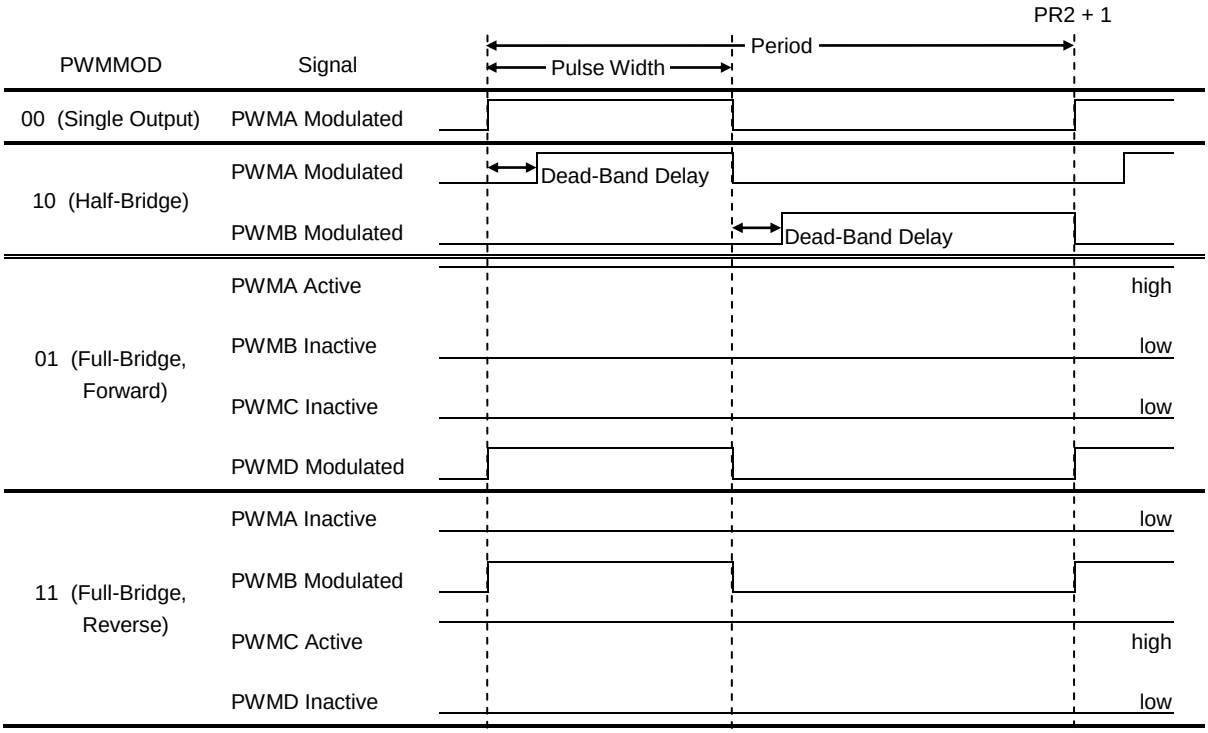
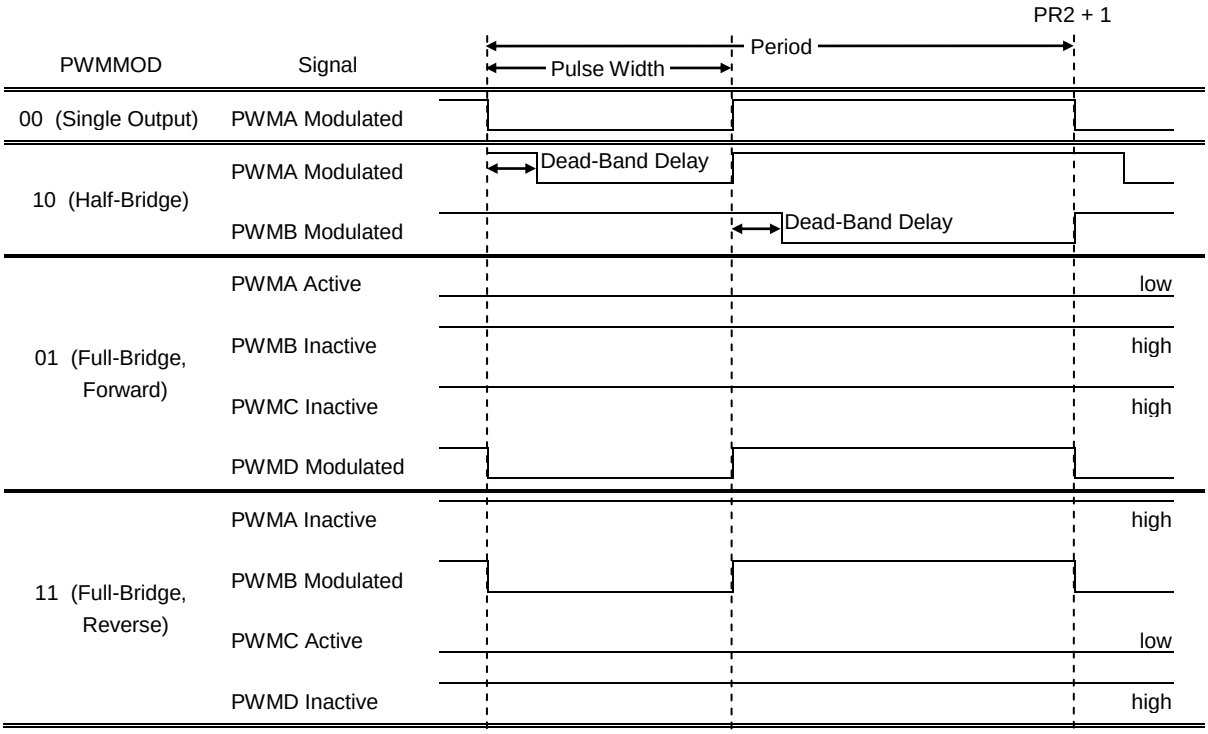


FIGURE 2.14: Example of Enhanced PWM Mode Output Relationships (Active-Low State)



Note 1: Period =[PR2+ 1]•4•TOSC • (TMR2 Prescale Value)

2: Pulse Width =(CCPRL:CCPCON<5:4>) •TOSC • (TMR2 Prescale Value)

3: Dead Band Delay Time = (PDC6:PDC0) x 4 x Tosc

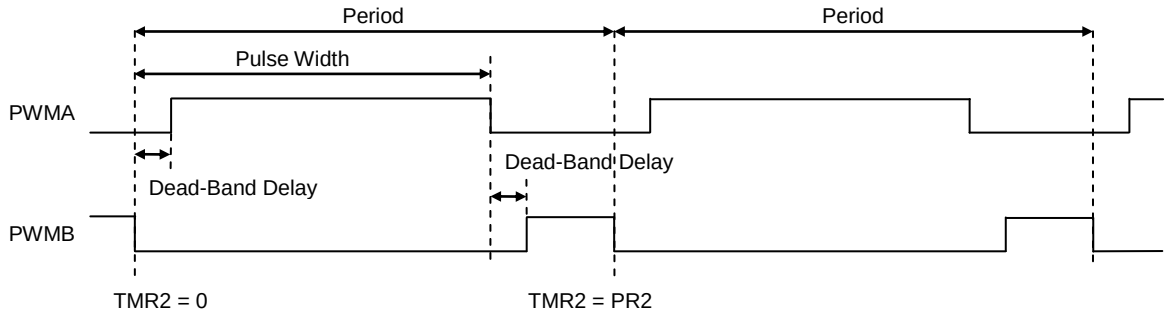
2.6.4.1 Half-Bridge Mode

In Half-Bridge mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the CCP/PWMA pin, while the complementary PWM output signal is output on the PWMB pin. This mode can be used for half-bridge applications, or for full-bridge applications, where four power switches are being modulated with two PWM signals.
In Half-Bridge mode, the programmable dead-band delay can be used to prevent shoot-through current in half-bridge power devices. The value of the PDC<6:0> bits of the PWMCON register sets the delay period in terms of instruction cycles (TCY or 4 TOSC) before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle.

Dead Band Delay Time = (PDC6:PDC0) x 4 x Tosc

Since the PWMA and PWMB outputs are multiplexed with the PORT data latches, the associated IOST bits must be cleared to configure PWMA and PWMB as outputs.

FIGURE 2.15: Example of Half-Bridge PWM Output (Active-High State)



2.6.4.2 Full-Bridge Mode

In Full-Bridge mode, all four pins are used as outputs.
In the Forward mode, pin CCP/PWMA is driven to its active state, pin PWMD is modulated, while PWMB and PWMC will be driven to their inactive state.
In the Reverse mode, PWMC is driven to its active state, pin PWMB is modulated, while PWMA and PWMD will be driven to their inactive state.
PWMA, PWMB, PWMC and PWMD outputs are multiplexed with the PORT data latches. The associated IOST bits must be cleared to configure the PWMA, PWMB, PWMC and PWMD pins as outputs.

FIGURE 2.16: Example of Full-Bridge PWM Output (Forward Mode, Active-High State)

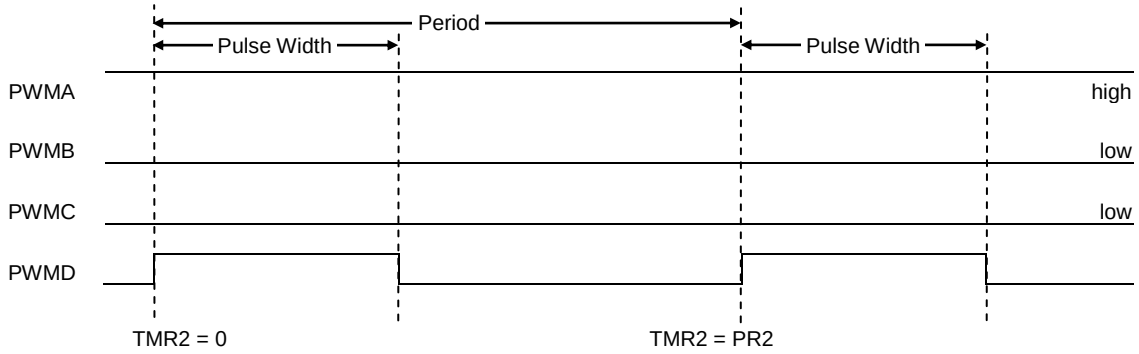
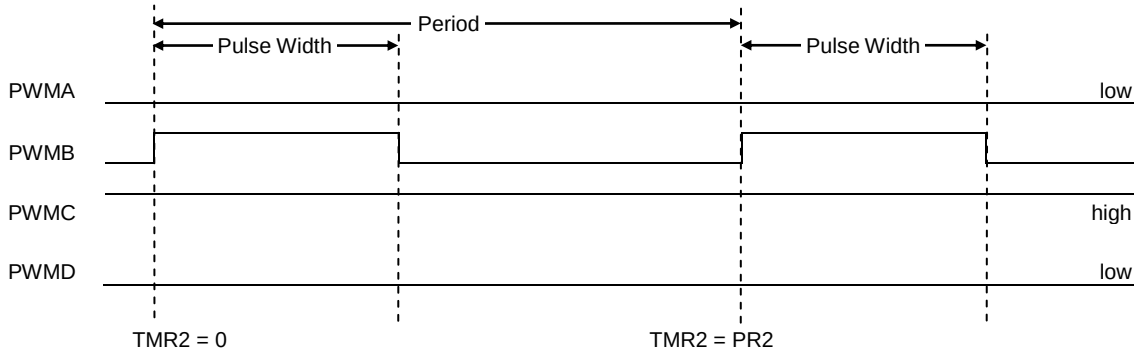
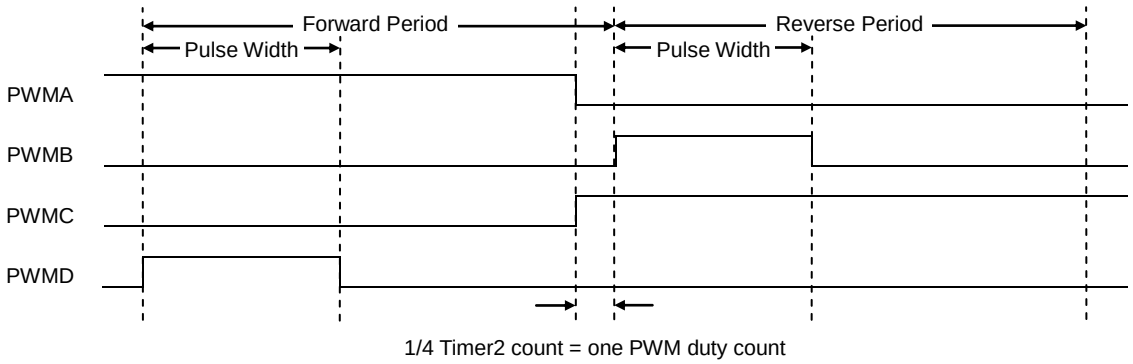


FIGURE 2.17: Example of Full-Bridge PWM Output (Reverse Mode, Active-High State)



In the Full-Bridge mode, the PWMMOD1 bit in the CCPCON register allows users to control the forward/reverse direction. When the application firmware changes this direction control bit, the module will change to the new direction on the next PWM cycle. The following sequence occurs four Timer2 cycles prior to the end of the current PWM period:
1. The modulated outputs (PWMB and PWMD) are placed in their inactive state.
2. The associated unmodulated outputs (PWMA and PWMC) are switched to drive in the opposite direction.
3. PWM modulation resumes at the beginning of the next period.
The Full-Bridge mode does not provide dead-band delay. As one output is modulated at a time, dead-band delay is generally not required. Only if both of the following conditions are true, the dead-band delay is required to prevent the shoot-through current:
1. The direction of the PWM output changes when the duty cycle of the output is at or near 100%.
2. The turn off time of the power switch, including the power device and driver circuit, is greater than the turn on time.
If changing PWM direction at high duty cycle is required for an application, two possible solutions for prevent the shoot-through current are:
1. Reduce PWM duty cycle for one PWM period before changing directions.
2. Use switch drivers that can drive the switches off faster than they can drive them on.

FIGURE 2.18: Example of PWM Direction Change (Forward → Reverse, Active-High State)



1/4 Timer2 count = one PWM duty count

Note 1: The direction bit PWMMOD1 of the CCPCON register is written any time during the PWM cycle.
2: When changing directions, the PWMA and PWMC signals switch before the end of the current PWM cycle. The modulated PWMB and PWMD signals are inactive at this time. The length of this time is four Timer2 counts.

2.6.4.3 Start-up Considerations

All of the I/O pins are in the input (high-impedance) state after reset. When any PWM mode is used, the external application circuits must use the proper external pull-up and/or pull-down resistors on the PWM output pins to keep the power switch devices in the OFF state until the microcontroller drives the I/O pins with the proper signal levels or activates the PWM output(s).
The CCPMOD<1:0> bits of the CCPCON register allow the user to choose whether the PWM output signals are active-high or active-low for each pair of PWM output pins (PWMA/PWMC and PWMB/PWMD). The PWM output polarities must be selected before the PWM pin output drivers are enabled. Changing the polarity configuration while the PWM pin output drivers are enabled is not recommended since it may result in damage to the application circuits.
The PWMA, PWMB, PWMC and PWMD output latches may not be in the proper states when the PWM module is initialized. Enabling the PWM pin output drivers at the same time as the Enhanced PWM modes may cause damage to the application circuit. The Enhanced PWM modes must be enabled in the proper Output mode and complete a full PWM cycle before enabling the PWM pin output drivers. The completion of a full PWM cycle is indicated by the T2IF bit of the INTFLAG register being set as the second PWM period begins.

2.6.4.4 PWM Auto-Shutdown Mode

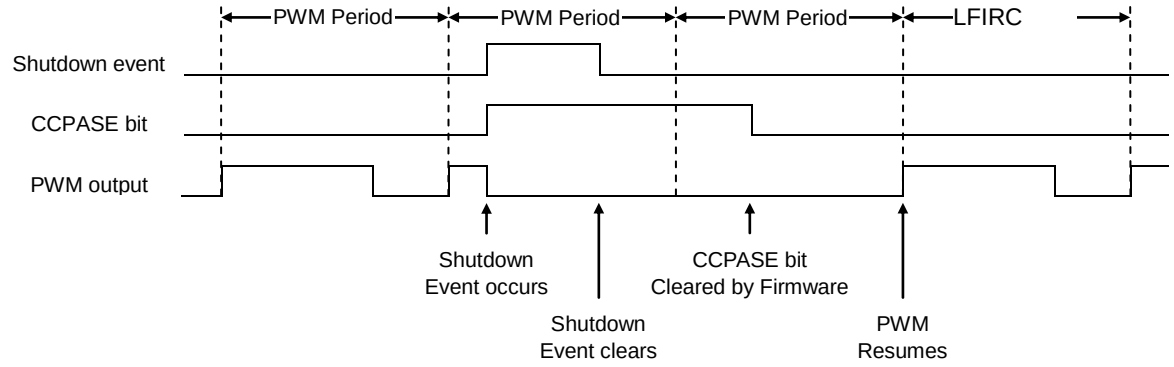
The PWM mode supports an Auto-Shutdown mode that will disable the PWM outputs when an external shutdown event occurs. Auto-Shutdown mode places the PWM output pins into a predetermined state. This mode is used to help prevent the PWM from damaging the application.
The auto-shutdown sources are selected using the CCPASx bits of the CCPAS register. A shutdown event may be generated by:
1. A logic ‘0’ on the INT pin
2. Comparator 1
3. Comparator 2
4. Setting the CCPASE bit in firmware

A shutdown condition is indicated by the CCPASE (Auto-Shutdown Event Status) bit of the CCPAS register. If the bit is a ‘0’, the PWM pins are operating normally. If the bit is a ‘1’, the PWM outputs are in the shutdown state.
When a shutdown event occurs, two things happen:
The CCPASE bit is set to ‘1’. The CCPASE will remain set until cleared in firmware or an auto-restart occurs.
The enabled PWM pins are asynchronously placed in their shutdown states. The PWM output pins are grouped into pairs [PWMA/PWMC] and [PWMB/PWMD]. The state of each pin pair is determined by the PSSAC and PSSBD bits of the CCPAS register. Each pin pair may be placed into one of three states:

1. Drive logic ‘1’
2. Drive logic ‘0’
3. Tri-state (high-impedance)

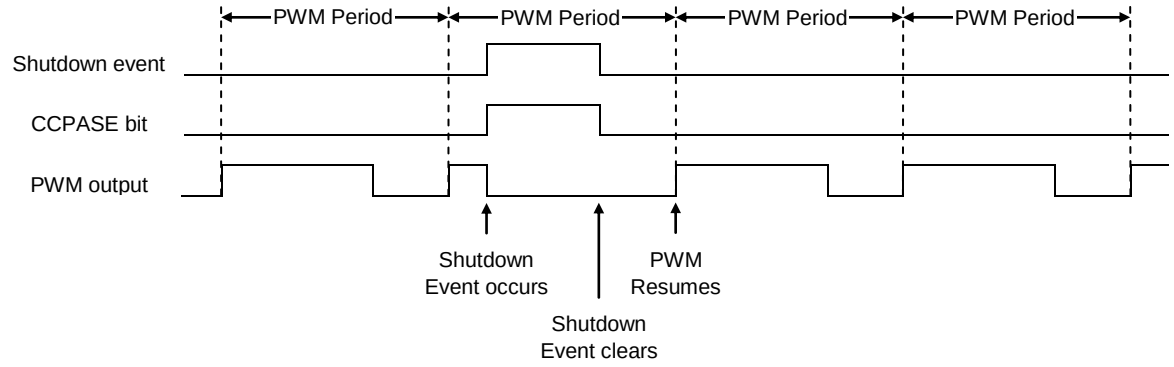
Note 1: The auto-shutdown condition is a level-based signal, not an edge-based signal. As long as the level is present, the auto-shutdown will persist.
2: Writing to the CCPASE bit is disabled while an auto-shutdown condition persists.
3: Once the auto-shutdown condition has been removed and the PWM restarted (either through firmware or auto-restart) the PWM signal will always restart at the beginning of the next PWM period.

FIGURE 2.19: PWM Auto-Shutdown with Firmware Restart (PRSEN = 0)



The Enhanced PWM can be configured to automatically restart the PWM signal once the auto-shutdown condition has been removed. Auto-restart is enabled by setting the PRSEN bit in the PWMCON register.
If auto-restart is enabled, the CCPASE bit will remain set as long as the auto-shutdown condition is active.
When the auto-shutdown condition is removed, the CCPASE bit will be cleared via hardware and normal operation will resume.

FIGURE 2.20: PWM Auto-Shutdown with Auto-Restart Enabled (PRSEN = 1)



2.7 IR Carrier Output (IROUT)

FM8PE32F is build-in an IR carrier output generator. The output is controlled by IREN (IRCON<7>), IROEN (IRCON<6>), IRCEN (IRCON<5>), IRSC (IRCON<4>), IRPS1:IRPS0 (IRCON<1:0>) bits and IRCYCLE, IRDUTY, IRCPR registers.

TABLE 2.3: IR Module Clock Source Prescaler Bits.

IRPS1 : IRPS0	IR Module Clock Source Frequency
0, 0	Oscillator Frequency / 1
0, 1	Oscillator Frequency / 2
1, 0	Oscillator Frequency / 4
1, 1	Oscillator Frequency / 8

The IROUT frequency and duty cycle are following the equations below:

IROUT frequency = (IR Module Clock Source Frequency) / IRCYCLE<7:0>
IROUT duty cycle = IRDUTY<7:0> / IRCYCLE<7:0>

For example, if oscillator frequency is equal to 455KHz, and the IRPS1:IRPS0 = (0, 0), IRCYCLE = 12, and IRDUTY = 6, then

IR Module Clock Source Frequency = 455 KHz / 1 = 455 KHz;
IROUT frequency = 455KHz / 12 = 38KHz, and
IROUT duty cycle = 6 / 12 = 50%

Note: 1. before enabling the IROUT (set IREN = “1”), set the IOC2 pin to be an output pin and output “high” for negative pulse or “low” for positive pulse is needed.
2. The value of IRDUTY<7:0> must be less than IRCYCLE<7:0>.

The IR module is also build-in an IROUT counter which increase on every leading edge of internal IR pulse until the value of IR counter matches to IRCPR register, and then the IR counter will be reset to “0”, and increase again. **The IR counter is also cleared when IRCEN (IRCON<5>) bit is cleared, and during any kind of reset as well.**

FIGURE 2.21: Block Diagram of The IROUT

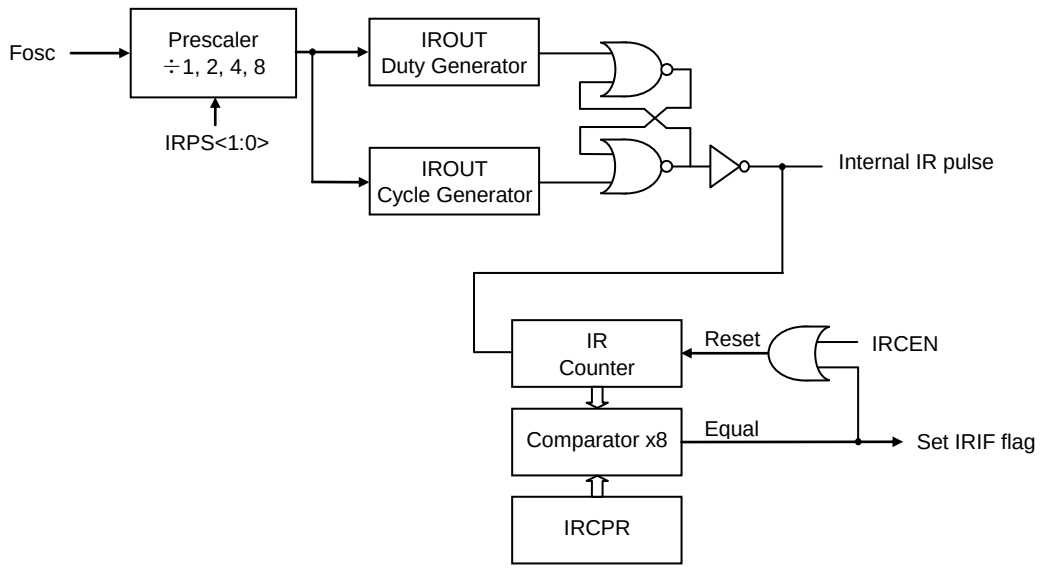


FIGURE 2.22: IROUT Waveform with Negative Pulse

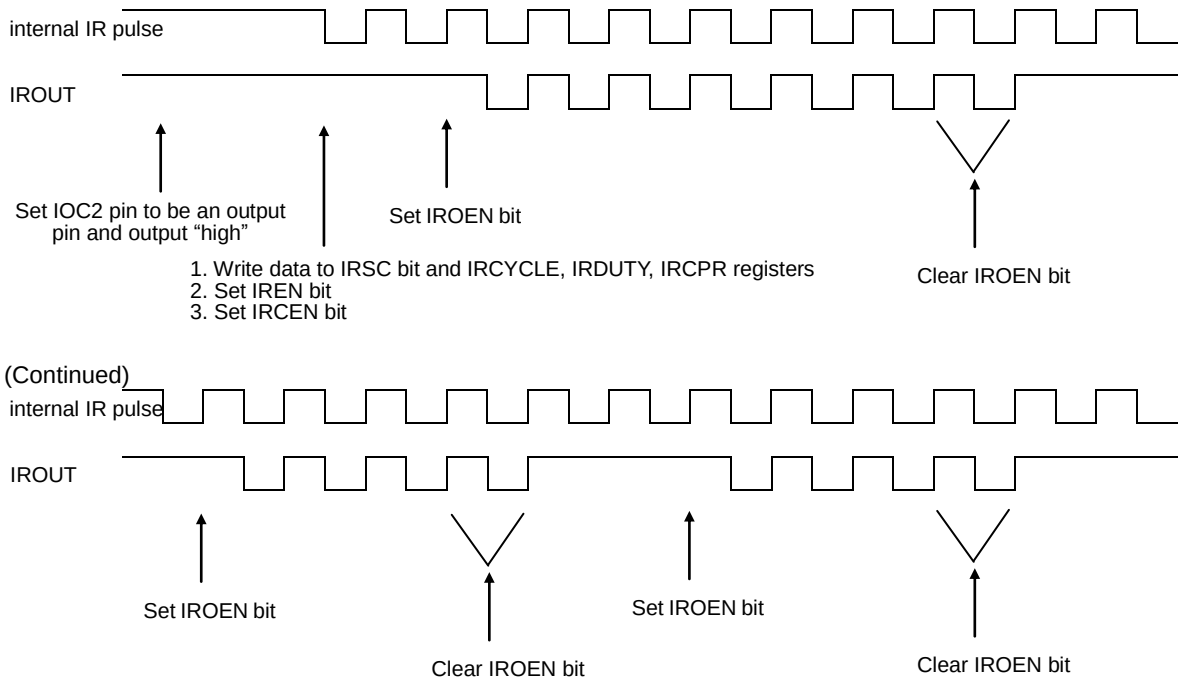
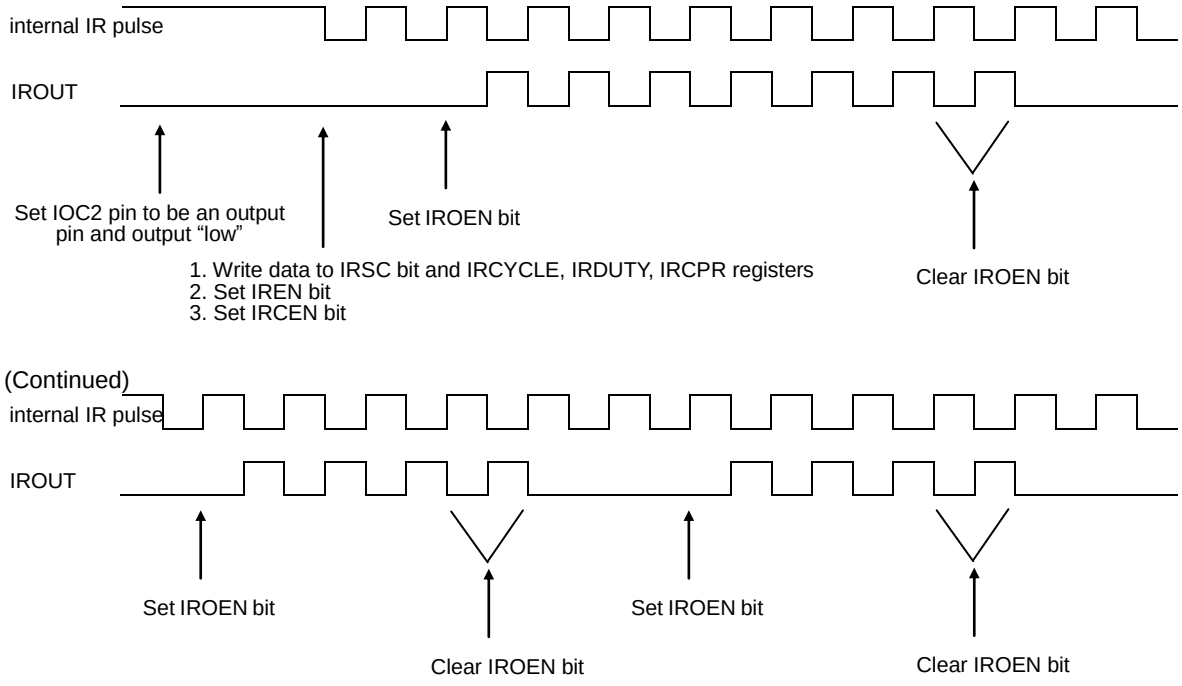


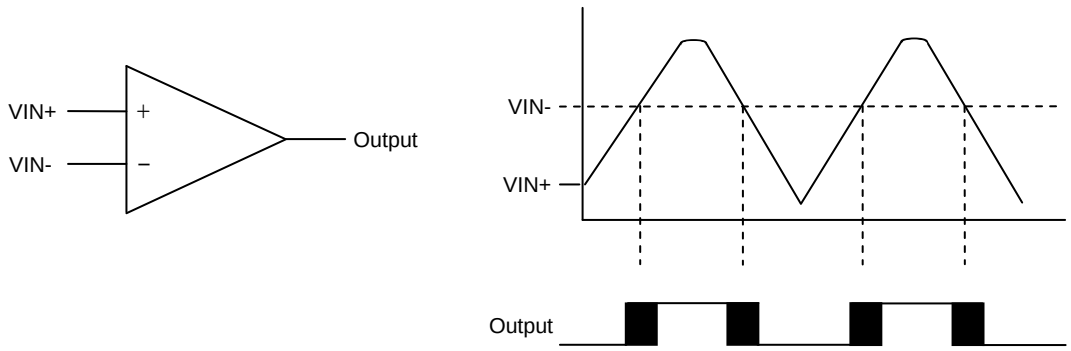
FIGURE 2.23: IROUT Waveform with Positive Pulse



2.8 Comparator Module

The FM8PE32F device has two comparators. The inputs to the comparator 1 (C1IN+, C1IN-) are multiplexed with the IOA0 and IOA1 pins, and the inputs to the comparator 2 (C2IN+, C2IN-) are multiplexed with the IOC0 and IOC1 pins. There is an on-chip comparator voltage reference (CVREF) that can also be applied to an input of these comparators. In addition, IOA2 can be configured as the comparator 1 output (C1OUT), and IOC4 can be configured as the comparator 2 output (C2OUT)
The compared results are stored in the C1OUT and C2OUT bits of CMPCON1 register. These bits are read-only. And/or the compared results may also be direct output to IOA2/C1OUT and IOC4/C2OUT pins in one of the eight possible modes. IOA2/C1OUT and IOC4/C2OUT pins must be defined as an output if implemented as the comparator digital outputs.
All pins configured as comparator analog inputs will read as a '0's.

FIGURE 2.24: Single Comparator



2.8.1 Comparator Output State

Each comparator output state can be read internally via the CxOUT (x=1~2) bit of the CMPCON1register. The comparator outputs are directed to CxOUT pins when CM<2:0> = 110. In this mode, the IOST bits for the associated CxOUT pins must be cleared to enable the output drivers.

2.8.2 Comparator Output Polarity

The polarity of the comparator output can be inverted by setting the CxINV (x=1~2) bit of the CMPCON1register. Clearing CxINV bit to "0" results in a non-inverted output. Setting CxINV bit to "1" results in a polarity inverted output.

TABLE 2.4: CxOUT Status/Output vs. Input Conditions & CxINV

Input Conditions	Comparator Output	CxINV	CxOUT
VIN+ < VIN-	0	0	0
VIN+ > VIN-	1	0	1
VIN+ < VIN-	0	1	1
VIN+ > VIN-	1	1	0

2.8.3 Comparator Input Switch

The inverting input (VIN-) of the comparators may be switched between two analog pins if CM<2:0> = 001 or 010. In these modes, both pins remain in analog mode regardless of which pin is selected as the input. The CINS bit of the CMPCON1register controls the comparator input switch.

FIGURE 2.25: Comparator 1 Output

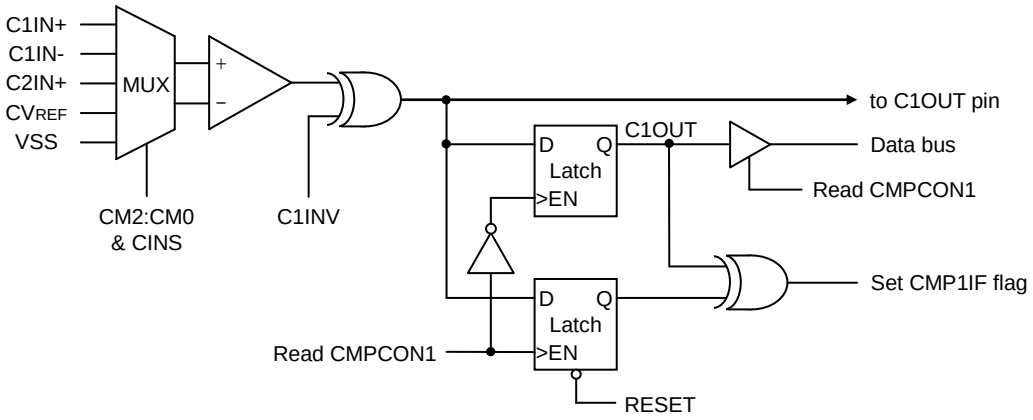
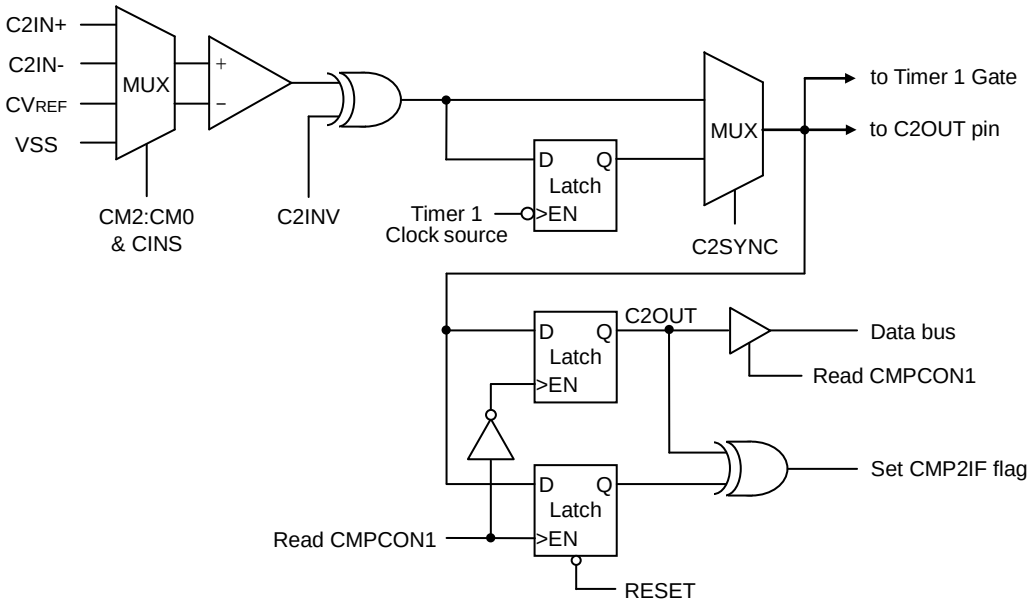


FIGURE 2.26: Comparator 2 Output

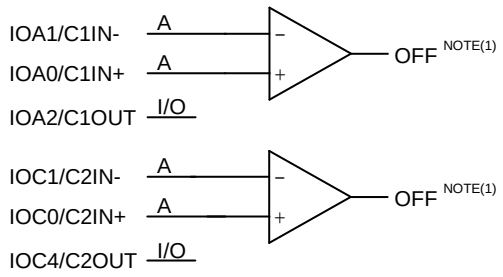


2.8.4 Comparator Operating Modes

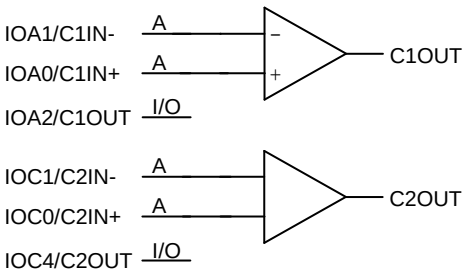
There are eight modes of operation for the comparator. The CM<2:0> bits of CMPCON1 register is used to select these modes.
The pins denoted as “A” will read as a “0” regardless the state of the I/O pin or the I/O control IOSTx bit. If pins are used as comparator analog inputs, the corresponding IOSTx bit must be set to “1”.
Pins denoted as “D” means pins are used as comparator digital output, and the corresponding IOSTx bit need to be cleared to “0”.
If the comparator mode is changed, the comparator output level may not be valid for a specified period of time, so comparator interrupts should be disabled during a comparator mode change.

FIGURE 2.27: Comparator I/O Operating Modes

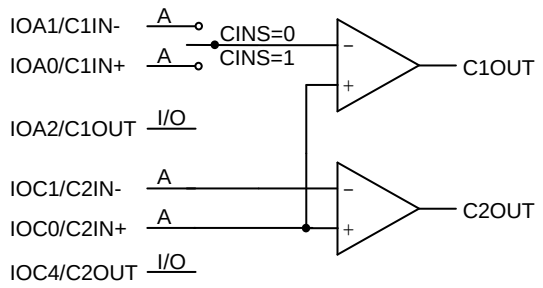
CM2:CM0 = 000
Comparator Reset (low power)



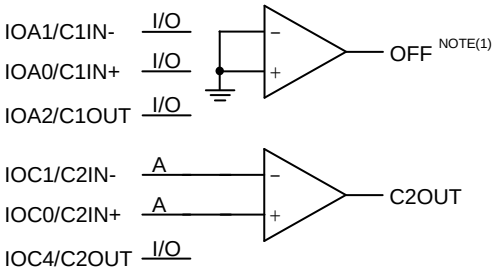
CM2:CM0 = 100
Two Independent Comparators



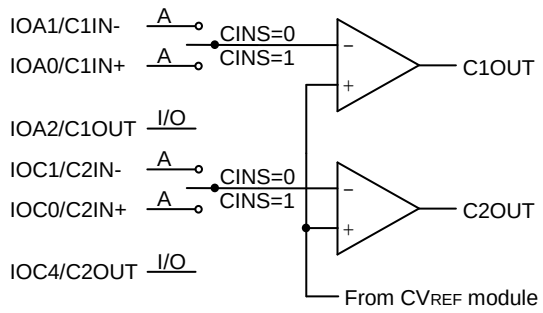
CM2:CM0 = 001
Three Inputs Multiplexed to Two Comparators



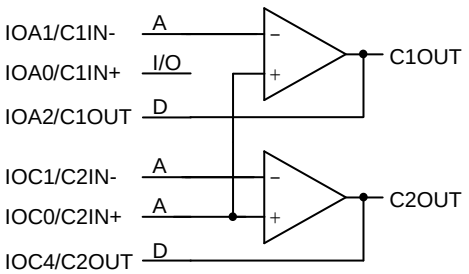
CM2:CM0 = 101
One Independent Comparators



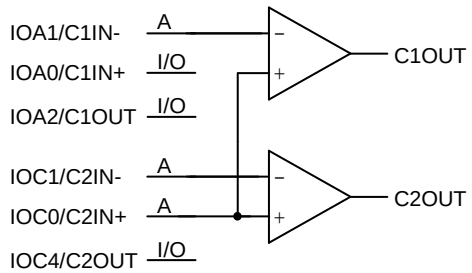
CM2:CM0 = 010
Four Inputs Multiplexed to Two Comparators



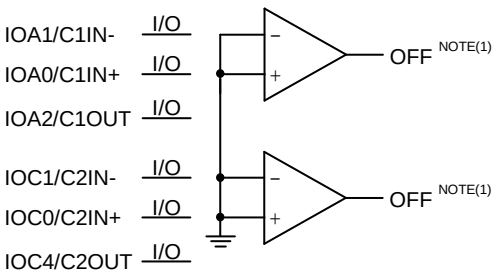
CM2:CM0 = 110
Four Inputs Multiplexed to Two Comparators



CM2:CM0 = 011
Two Common Reference Comparators



CM2:CM0 = 111
Comparator Off (lowest power)



Note(1) : Reads as “0”, unless CxINV = 1
Note(2) : A = Comparator analog input, pins always read ‘0’ D = Comparator digital output
CINS = Comparator input switch (CMPCON1<3>) I/O = Normal port I/O

2.8.5 Comparator 2 Gating Timer 1

This feature can be used to time the duration or interval of analog events. Clearing the T1GSS bit of CMPCON2 register will enable Timer 1 to increment based on the output of Comparator 2. This requires that Timer 1 is on and gating is enabled.

2.8.6 Synchronizing Comparator 2 Output to Timer 1

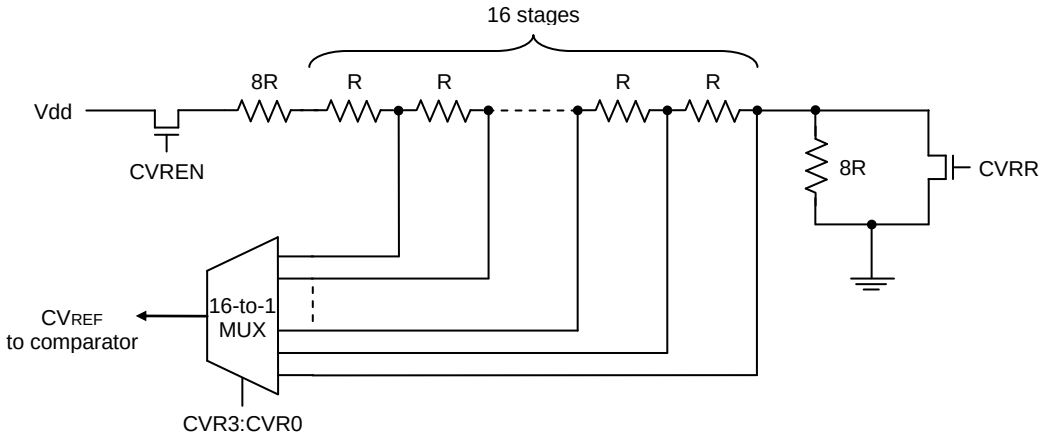
It is recommended to synchronize Comparator 2 with Timer 1 by setting the C2SYNC bit of CMPCON2 register when the comparator is used as the Timer 1 gating source. When enabled, the comparator output is latched on the falling edge of the Timer 1 clock source. If a prescaler is used with Timer 1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer 1 clock source and Timer 1 increments on the rising edge of its clock source. This ensures Timer 1 does not miss an increment if the comparator changes during an increment.

2.8.7 Comparator Voltage Reference (CV_{REF})

The comparator module also allows the selection of an internally generated voltage reference for one of the comparator inputs. The internal reference signal is used for four of the eight comparator modes. The CMPCON2 register controls the voltage reference module.
The voltage reference can output 32 distinct voltage levels, 16 in a high range and 16 in a low range. The following equations determine the output voltages:

If CVRR = 1 (low range): CV_{REF} = (CVR3:CVR0) * Vdd / 24.
If CVRR = 0 (high range): CV_{REF} = Vdd / 4 + (CVR3:CVR0) * Vdd / 32.

FIGURE 2.28: Comparator Voltage Reference



2.8.8 Comparator Status Change Interrupt

The comparator interrupt flag bits CMP1IF/CMP2IF bit of INTFLAG register is set whenever there is a change in the output value of the comparator 1/2.
Before the comparator 1/2 interrupt is enabled, reading CMPCON1 (any instruction accessed to CMPCON1, including read/write instructions) to get the status of the C1OUT/C2OUT bit is necessary.
The CMP1IF/CMP2IF bit must be reset in software by clearing it to '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.
The CMPxIE bit of INTEN register, and the PEIE and GIE bits of the INTCON register must be all be set to enable comparator interrupts. If any of these bits are cleared, the interrupt is not enabled, although the CMPxIF bit will still be set if an interrupt condition occurs.
User can clear the interrupt in the interrupt service routine in the following manner:
a) Any read or write of CMPCON1. This will end the mismatch condition.
b) Clear flag bit CMP1IF/CMP2IF.

A mismatch condition will continue to set flag bit CMP1IF/CMP2IF. Either the CMPCON1is read or the comparator 1/2 output returns to the previous state will end the mismatch condition and allow flag bit CMP1IF/CMP2IF to be cleared.

2.8.9 Comparator Operation During SLEEP

Both the comparator and voltage reference, if enabled before entering SLEEP mode, remain active during SLEEP. This results in higher SLEEP currents than shown in the power-down specifications. To minimize power consumption while in SLEEP mode, turn off the comparator by selecting comparator mode CM2:CM0 = 000 or 111, and voltage reference enable bit CVREN (CMPCON2<7>) = 0.
While the comparator is enabled during SLEEP, a change to the comparator output can wake-up the device. To enable the comparator to wake the device from sleep, the CMPxIE bit of the INTEN register and the PEIE bit of the INTCON register must be set. If the GIE bit (INTCON<7>) is set, the device will wake-up and jump to the interrupt vector. If the GIE bit is cleared, the device will wake-up and continue execution at the instruction after the SLEEP instruction.

2.9 Data EEPROM Memory

The EEPROM data memory is readable and writable during normal operation (full VDD range). This memory is indirectly addressed through the SFRs (Special Function Registers). There are four SFRs used to read and write this memory: EEDATA, EEADR, EECON1, and EECON2 (not a physically implemented register).
EEDATA holds the 8-bit data for read/write, and EEADR holds the 8-bit address of the EEPROM location being accessed. This device has 256 bytes of data EEPROM with an address range from 0h to FFh.
EECON1 is the control register with five low order bits physically implemented. The upper three bits are unimplemented and read as '0's.
Control bits RD, WR and WRONLY initiate read and write (with erase) and write only (without erase), respectively. These bits is set by software and cleared by hardware at completion of the read or write operation.
To clear the WREN bit by software prevents the accidental, premature termination of a write operation. To set the WREN bit will allow a write operation.
The WRERR bit is set when a write operation is interrupted by a RSTB Reset, or a WDT Time-out Reset during normal operation. In these situations, following RESET, the user can check the WRERR bit, clear it, and rewrite the location. The data and address will be cleared, therefore, the EEDATA and EEADR registers will need to be re-initialized.
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.
The EEPROM data memory allows byte read and byte write. Set “WR” bit to “1” automatically erases the location and writes the new data. Set “WRONLY” bit to “1” writes the new data only without any erasing. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature as well as from chip to chip.
Interrupt flag bit EEIF (INTFLAG<7>) is set when write is complete. This bit must be cleared in software.

2.9.1 Reading the EEPROM Data Memory

To read a data memory location, the user must write the address to the EEADR register and then set control bit RD (EECON1<0>). The data is available in the EEDATA register. Therefore, it can be read in the next instruction. EEDATA holds this value until another read, or until it is written to by the user (during a write operation).

EXAMPLE 2.2: Data EEPROM Read

```
BSR     STATUS, RP0     ;Bank 1
MOVIA   CONFIG_ADDR     ;
MOVAR   EEADR           ;Address to read
BSR     EECON1, RD      ;EE Read
MOVR    EEDATA, ACC     ;Move data to ACC
```

2.9.2 Writing to the EEPROM Data Memory

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific sequence to initiate the write for each byte.

EXAMPLE 2.3: Data EEPROM Write

```
BSR     STATUS, RP0     ;Bank 1
BSR     EECON1, WREN    ;Enable write
BCR     INTCON, GIE     ;Disable INTs
MOVIA   55h             ;Unlock write
MOVAR   EECON2          ;
MOVIA   AAh             ;
MOVAR   EECON2          ;
BSR     EECON1, WR      ;Start the write
BSR     INTCON, GIE     ;Enable INTs
```

Required Sequence

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR or WRONLY bit) for each byte. A cycle count is executed during the required sequence. Any number that is not equal to the required cycles to execute the required sequence will prevent the data from being written into the EEPROM.
Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.
After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.
At the completion of the write cycle, the WR or WRONLY bit is cleared in hardware and the EEIF (INTFLAG<7>) is set. The user can either enable this interrupt or poll this bit. The EEIF flag must be cleared by software.

2.9.3 Write Verify

Depending on the application, good programming practice may dictate that the value written to the Data EEPROM should be verified to the desired value to be written.

EXAMPLE 2.4: Write Verify

```
MOVR    EEDATA, ACC     ;EEDATA & EEADR not changed from previous write
BSR     STATUS, RP0     ;Bank 1
BSR     EECON1, RD      ;Read the value written
XORAR   EEDATA, ACC
BTRSS   STATUS, Z       ;Is data the same
GOTO    WRITE_ERR       ;No, handle error
  :                     ;Yes, continue
```

2.10 Interrupts

The FM8PE32F series has up to eleven sources of interrupt:

1. Port A input status change interrupt
2. External interrupt INT pin.
3. TMR0 overflow interrupt.
4. TMR1 overflow interrupt.
5. TMR2 match interrupt.
6. Comparator 1 status change interrupt.
7. Comparator 2 status change interrupt.
8. Oscillator fail interrupt.
9. CCP interrupt.
10. EEPROM data write interrupt.
11. IROUT interrupt.

INTCON<2:0> and INTFLAG is the interrupt flag register that records the interrupt requests in the relative flags.
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 enabled/disabled through their corresponding enable bits in INTCON<6:3> and INTEN register regardless of the status of the GIE bit.
When an interrupt event occur with the GIE bit and its corresponding interrupt enable bit are all set, the GIE bit will be cleared by hardware to disable any further interrupts, and the next instruction will be fetched from address 004h.
The interrupt flag bits must be cleared by software before re-enabling GIE bit to avoid recursive interrupts.
The RETFIE instruction exits the interrupt routine and set the GIE bit to re-enable interrupt.
The flag bits set by interrupt event regardless of the status of its mask bit or the GIE bit. Reading the INTFLAG register will be the logic AND of INTFLAG and INTEN.

2.10.1 Port A Input Status Change Interrupt

An input change on IOA<5:0> set flag bit PAIF (INTCON<0>).
The PAIE bit (INTCON<3>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.
Before the port A input status change interrupt is enabled, reading PORTA (any instruction accessed to PORTA, including read/write instructions) is necessary. Any pin which corresponding WUAn bit (AWUCON<5:0>) is cleared to “0” or configured as output will be excluded from this function.
User can clear the interrupt in the interrupt service routine in the following manner:
1. Any read or write of PORTA. This will end the mismatch condition.
2. Clear flag bit PAIF.
A mismatch condition will continue to set flag bit PAIF. Reading PORTA will end the mismatch condition and allow flag bit PAIF to be cleared.

2.10.2 External INT Interrupt

External interrupt on INT pin is rising or falling edge triggered selected by INTEDG (OPTION_REG<6>).
When a valid edge appears on the INT pin the flag bit INTIF (INTCON<1>) is set.
The INTIE bit (INTCON<4>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.
The INT pin interrupt can wake-up the system from SLEEP condition, if bit INTIE was set before going to SLEEP. If GIE bit was set, the program will execute interrupt service routine after wake-up; or if GIE bit was cleared, the program will execute next PC after wake-up.

2.10.3 Timer0 Overflow Interrupt

An overflow (FFh → 00h) in the TMR0 register will set the flag bit T0IF (INTCON<2>).
The T0IE bit (INTCON<5>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.4 Timer1 Overflow Interrupt

An overflow (FFFFh → 0000h) in the TMR1H:TMR1L register pair will set the flag bit T1IF (INTFLAG<0>).
The T1IE bit (INTEN<0>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.5 Timer2 Match Interrupt

A match condition (TMR2 = PR2) in the TMR2 register will set the flag bits T2IF (INTFLAG<1>).
The T2IE bit (INTEN<1>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.6 Comparator 1 Status Change Interrupt

The comparator 1 interrupt flag bit CMP1IF (INTFLAG<3>) is set whenever there is a change in the output value of the comparator 1.
The CMP1IE bit (INTEN<3>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.7 Comparator 2 Status Change Interrupt

The comparator 2 interrupt flag bit CMP2IF (INTFLAG<4>) is set whenever there is a change in the output value of the comparator 2.
The CMP2IE bit (INTEN<4>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.8 Oscillator Fail Interrupt

The Oscillator fail interrupt flag bit OSFIF (INTFLAG<2>) is set whenever the external clock fails.
The OSFIE bit (INTEN<2>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.9 CCP Interrupt

In Capture mode, the CCP interrupt flag bit CCPIF (INTFLAG<5>) is set whenever a TMR1 capture occurred.
In Compare mode, the CCP interrupt flag bit CCPIF is set whenever a TMR1 compare match occurred.
In PWM mode, the CCP interrupt flag bit CCPIF is unused.
The CCPIE bit (INTEN<5>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.10 EEPROM Data Write Interrupt

The EEPROM data write interrupt flag bit EEIF (INTFLAG<7>) is set when EEPROM write is complete.
The EEIE bit (INTEN<7>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.10.11 IROUT Interrupt

The IROUT interrupt flag bit IRIF (INTFLAG<5>) is set whenever the value of IR counter matches to IRCPR register.
The IRIE bit (INTEN<5>), PEIE bit (INTCON<6>) and GIE bit (INTCON<7>) must be set to enable the interrupt. If any of these bits are cleared, the interrupt is not enabled.

2.11 Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.
When SLEEP instruction is executed, the $\overline{PD}$ bit (STATUS<3>) is cleared, the $\overline{TO}$ bit is set, the watchdog timer will be cleared and keeps running, and the oscillator driver is turned off.
All I/O pins maintain the status they had before the SLEEP instruction was executed.

2.11.1 Wake-up from SLEEP Mode

The device can wake-up from SLEEP mode through one of the following events:
1. RSTB reset.
2. WDT time-out wake-up (if enabled).
3. Interrupt from RA2/INT pin, PORTA change, or a peripheral interrupt.

External RSTB reset will cause a device reset. WDT time-out and interrupt events are considered a continuation of program execution.
The $\overline{PD}$ and $\overline{TO}$ bits can be used to determine the cause of device reset. The $\overline{PD}$ bit is set on power-up and is cleared when SLEEP instruction is executed. The $\overline{TO}$ bit is cleared if a WDT time-out occurred.
For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set. Wake-up is regardless of the GIE bit. If GIE bit is cleared, the device will continue execution at the instruction after the SLEEP instruction. If the GIE bit is set, the device will branch to the interrupt address (004h).
In HF, XT or LF oscillation mode, the system wake-up delay time is 1024 oscillator cycles time. And in ERC, IRC, ERIC or EC oscillation mode, the system wake-up delay time is 24 oscillator cycles time.

2.12 Reset

FM8PE32F devices may be RESET in one of the following ways:
1. Power-on Reset (POR)
2. Brown-out Reset (BOR)
3. RSTB Pin Reset
4. WDT time-out Reset during normal operation

Some registers are not affected in any RESET condition. Their status is unknown on Power-on Reset and unchanged in any other RESET. Most other registers are reset to a “reset state” on Power-on Reset, RSTB or WDT Reset.
A Power-on RESET pulse is generated on-chip when Vdd rise is detected. To use this feature, the user merely ties the RSTB pin to Vdd.
On-chip Low Voltage Detector (LVD) places the device into reset when Vdd is below a fixed voltage. This ensures that the device does not continue program execution outside the valid operation Vdd range. Brown-out RESET is typically used in AC line or heavy loads switched applications.
A RSTB or WDT time-out during normal operation also results in a device RESET.
The $\overline{TO}$ and $\overline{PD}$ bits (STATUS<4:3>) are set or cleared depending on the different reset conditions.

2.12.1 Power-up Reset Timer (PWRT)

The Power-up Reset Timer provides a nominal 0/18/36/72ms (selected by PWRT<1:0> bits of configuration word) delay after Power-on Reset (POR) or Brown-out Reset (BOR). The device is kept in reset state as long as the PWRT is active. A configuration bit, PWRTE can disable or enable the Power-up Reset Timer.
The PWDT delay will vary from device to device due to Vdd, temperature, and process variation.

2.12.2 Oscillator Start-up Timer (OST)

The OST timer provides a 1024 oscillator cycle delay (from OSCI input) after the PWRT delay is over in HF, XT or LF oscillation mode and only on Power-up Reset, Brown-out Reset, or wake-up from SLEEP. This delay ensures that the X’tal oscillator or resonator has started and stabilized. The device is kept in reset state as long as the OST is active.

This counter only starts incrementing after the amplitude of the OSCI signal reaches the oscillator input thresholds.
Note: In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected.

TABLE 2.5: PWRT & OST Period

Oscillator Mode	Power-on Reset (POR) Brown-out Reset (BOR)			
	PWRT = 1, 1	PWRT = 1, 0	PWRT = 0, 1	PWRT = 0, 0
HF, XT, LF	1024*Tosc	18 ms + 1024*Tosc	36 ms + 1024*Tosc	72 ms + 1024*Tosc
ERC, IRC, ERIC, EC	24*Tosc	18 ms + 24*Tosc	36 ms + 24*Tosc	72 ms + 24*Tosc

Oscillator Mode	RSTB Reset WDT Time-out Reset (during Normal Operation)	WDT Time-out Wake-up (during SLEEP)	Wake-up from SLEEP
HF, XT, LF	-	1024*Tosc	1024*Tosc
ERC, IRC, ERIC, EC	-	24*Tosc	24*Tosc

2.12.3 Reset Sequence

When Power-on Reset (POR), or Brown-out Reset (BOR) is detected, the reset sequence is as follows:
1. The reset latch is set and the PWRT & OST are cleared.
2. When the internal POR, BOR pulse is finished, then the PWRT begins counting.
3. After the PWRT time-out, the OST is activated.
4. And after the OST delay is over, the reset latch will be cleared and thus end the on-chip reset signal.

In HF, XT or LF oscillation mode, the totally system reset delay time is 0/18/36/72ms plus 1024 oscillator cycle time after POR or BOR.
And in ERC, IRC, ERIC or EC oscillation mode, the totally system reset delay time is 0/18/36/72ms plus 24 oscillator cycle time after POR or BOR.

FIGURE 2.29: Simplified Block Diagram of on-chip Reset Circuit

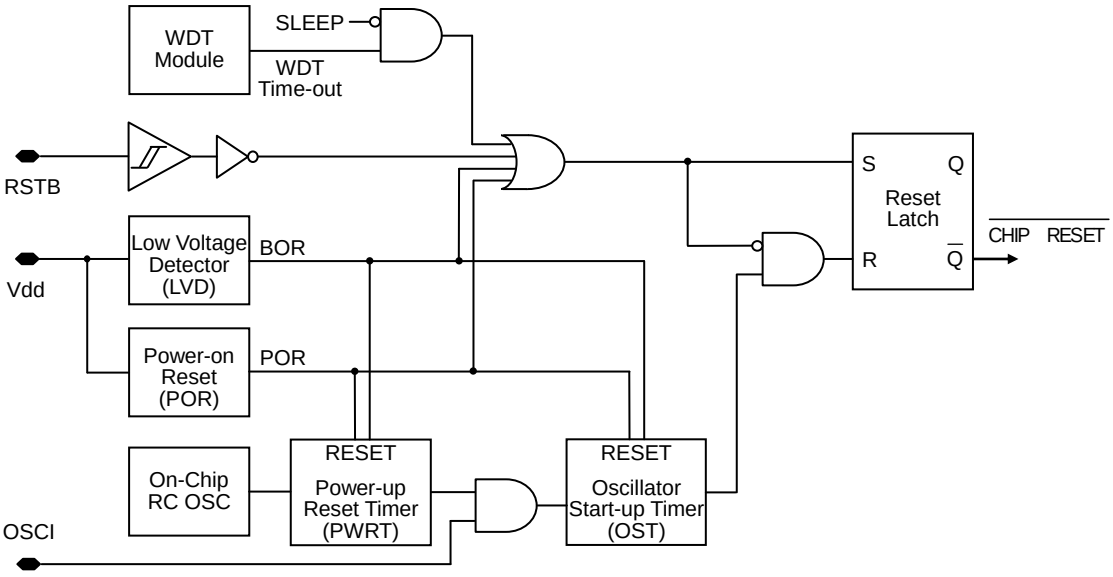


FIGURE 2.30: Time-out Sequence on Power-up (RSTB Pin Tied to Vdd)

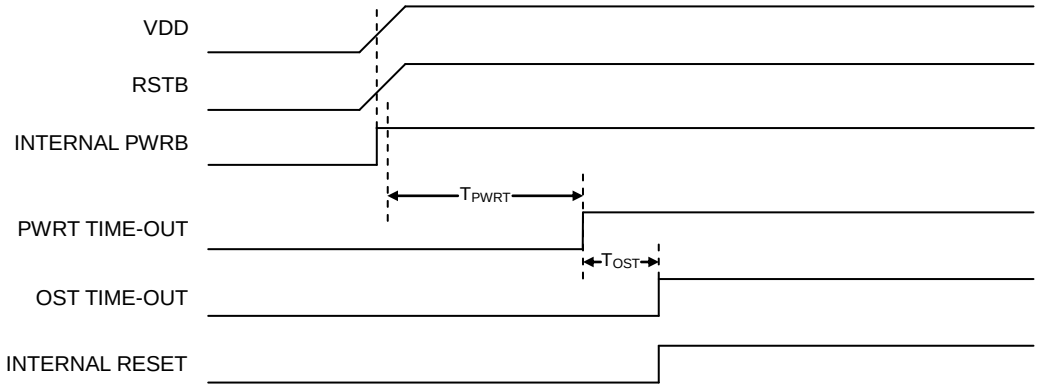


FIGURE 2.31: Time-out Sequence on Power-up (RSTB Pin Not Tied to Vdd): Condition 1

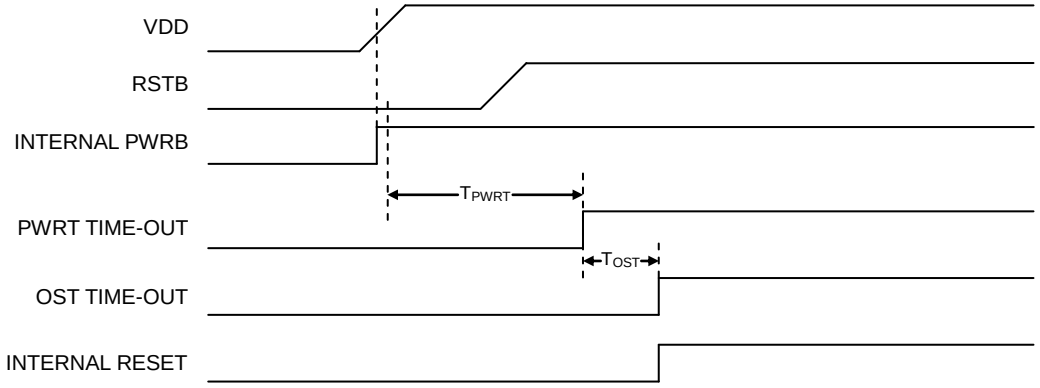
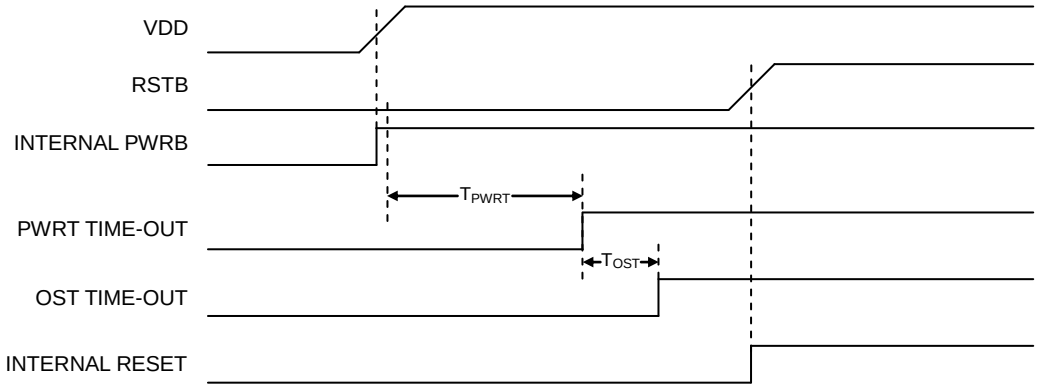


FIGURE 2.32: Time-out Sequence on Power-up (RSTB Pin Not Tied to Vdd) : Condition 2



Note: T_{PWRT} = 0/18/36/72 ms; T_{OST} = 0/1024 oscillator cycle time

TABLE 2.6: Reset Conditions for All Registers

Register	Address	Power-on Reset	RSTB Reset WDT Time-out Reset Brown-out Reset	Wake-up from SLEEP through Interrupt or WDT Time-out
ACC	N/A	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h	---- ----	---- ----	---- ----
TMR0	01h; Bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h	0000 0000	0000 0000	PC + 1 004h
STATUS	03h	0001 1xxx	000# #uuu	uuu# #uuu
FSR	04h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	05h; Bank 0	--xx xxxx	--uu uuuu	--uu uuuu
PORTC	07h; Bank 0	--xx xxxx	--uu uuuu	--uu uuuu
PCHBUF	0Ah	---- -000	---- -000	---- -uuu
INTCON	0Bh	0000 0000	0000 0000	uuuu uqqq
INTFLAG	0Ch; Bank 0	0000 0000	0000 0000	qqqq qqqq
TMR1L	0Eh; Bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh; Bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h; Bank 0	0000 0000	0000 0000	uuuu uuuu
TMR2	11h; Bank 0	0000 0000	0000 0000	uuuu uuuu
T2CON	12h; Bank 0	-000 0000	-000 0000	-uuu uuuu
CCPRL	13h; Bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPRH	14h; Bank 0	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPCON	15h; Bank 0	0000 0000	0000 0000	uuuu uuuu
PWMCON	16h; Bank 0	0000 0000	0000 0000	uuuu uuuu
CCPAS	17h; Bank 0	0000 0000	0000 0000	uuuu uuuu
WDTCON	18h; Bank 0	---0 1000	---0 1000	---u uuuu
CMPCON1	19h; Bank 0	0000 0000	0000 0000	uuuu uuuu
CMPCON2	1Ah; Bank 0	---- --10	---- --10	---- --uu
IRCYCLE	1Bh; Bank 0	0000 1100	0000 1100	uuuu uuuu
IRDUTY	1Ch; Bank 0	0000 0110	0000 0110	uuuu uuuu
IRCPR	1Dh; Bank 0	0000 0000	0000 0000	uuuu uuuu
IRCON	1Eh; Bank 0	0000 --00	0000 --00	uuuu --uu
OPTION_REG	01h; Bank 1	1111 1111	1111 1111	uuuu uuuu
IOSTA	05h; Bank 1	--11 1111	--11 1111	--uu uuuu
IOSTC	07h; Bank 1	--11 1111	--11 1111	--uu uuuu
INTEN	0Ch; Bank 1	0000 0000	0000 0000	uuuu uuuu
PCON	0Eh; Bank 1	--01 --0x	--01 --uu --01 --10	--uu --uu
OSCCON	0Fh; Bank 1	-110 x000	-110 x000	-uuu xuuu
OSCTUNE	10h; Bank 1	---0 0000	---0 0000	---u uuuu
PR2	12h; Bank 1	1111 1111	1111 1111	uuuu uuuu
APHCON	15h; Bank 1	--11 -111	--11 -111	--uu -uuu
AWUCON	16h; Bank 1	--00 0000	--00 0000	--uu uuuu
CMPCON3	19h; Bank 1	0-0- 0000	0-0- 0000	u-u- uuuu
EEDATA	1Ah; Bank 1	0000 0000	0000 0000	uuuu uuuu
EEADR	1Bh; Bank 1	0000 0000	0000 0000	uuuu uuuu
EECON1	1Ch; Bank 1	---0 x000	---0 x000	---u xuuu
EECON2	1Dh; Bank 1	---- ----	---- ----	---- ----
General Purpose Registers	20 ~ 7Fh	xxxx xxxx	xxxx xxxx	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented,
# = refer to the following table for possible values.
q = one or more bits in INTCON and/or INTFLAG will be set to cause wake-up.

TABLE 2.7: $\overline{POR}$ / $\overline{BOD}$ / $\overline{TO}$ / $\overline{PD}$ Status after Reset or Wake-up

$\overline{POR}$	$\overline{BOD}$	$\overline{TO}$	$\overline{PD}$	RESET was caused by
0	x	1	1	Power-on Reset
1	0	1	1	Brown-out reset
u	u	u	u	RSTB Reset during normal operation
u	u	1	0	RSTB Reset during SLEEP
u	u	0	1	WDT Reset during normal operation
u	u	0	0	WDT Wake-up during SLEEP
u	u	1	0	Wake-up through interrupt during SLEEP

Legend: u = unchanged, x = unknown

TABLE 2.8: Instructions Affecting $\overline{TO}$ / $\overline{PD}$ Status Bits

Event	$\overline{TO}$	$\overline{PD}$
SLEEP instruction	1	0
CLRWDT instruction	1	1

2.13 Hexadecimal Convert to Decimal (HCD)

Decimal format is another number format for FM8PE32F. When the content of the data memory has been assigned as decimal format, it is necessary to convert the results to decimal format after the execution of ALU instructions. When the decimal converting operation is processing, all of the operand data (including the contents of the data memory (RAM), accumulator (ACC), immediate data, and look-up table) should be in the decimal format, or the results of conversion will be incorrect.
Instruction DAA can convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC.
The conversion operation is illustrated in example 2.2.

EXAMPLE 2.5: DAA CONVERSION

```
MOVIA   90h     ;Set immediate data = decimal format number “90” (ACC ← 90h)
MOVAR   30h     ;Load immediate data “90” to data memory address 30H
MOVIA   10h     ;Set immediate data = decimal format number “10” (ACC ← 10h)
ADDAR   30h, 0  ;Contents of the data memory address 30H and ACC are binary-added
                ;the result loads to the ACC (ACC ← A0h, C ← 0)
DAA             ;Convert the content of ACC to decimal format, and restored to ACC
                ;The result in the ACC is “00” and the carry bit C is “1”. This represents the
                ;decimal number “100”
```

Instruction DAS can convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC.
The conversion operation is illustrated in example 2.3.

EXAMPLE 2.6: DAS CONVERSION

```
MOVIA   10h     ;Set immediate data = decimal format number “10” (ACC ← 10h)
MOVAR   30h     ;Load immediate data “10” to data memory address 30H
MOVIA   20h     ;Set immediate data = decimal format number “20” (ACC ← 20h)
SUBAR   30h, 0  ;Contents of the data memory address 30H and ACC are binary-subtracted
                ;the result loads to the ACC (ACC ← F0h, C ← 0)
DAS             ;Convert the content of ACC to decimal format, and restored to ACC
                ;The result in the ACC is “90” and the carry bit C is “0”. This represents the
                ;decimal number “ -10”
```

2.14 Oscillator Module (with Fail-Safe Clock Monitor)

The system clock sources can be configured from external oscillators, crystal/ceramic resonators and Resistor-Capacitor (RC) circuits. In addition, the system clock source can be configured from one of two internal oscillators, with a choice of speeds selectable via software. Additional clock features include:

- Selectable system clock source between external or internal via software.
- Two-Speed Start-up mode, which minimizes latency between external oscillator start-up and code execution.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LF, XT, HF, EC or ERC modes) and switch automatically to the internal oscillator.

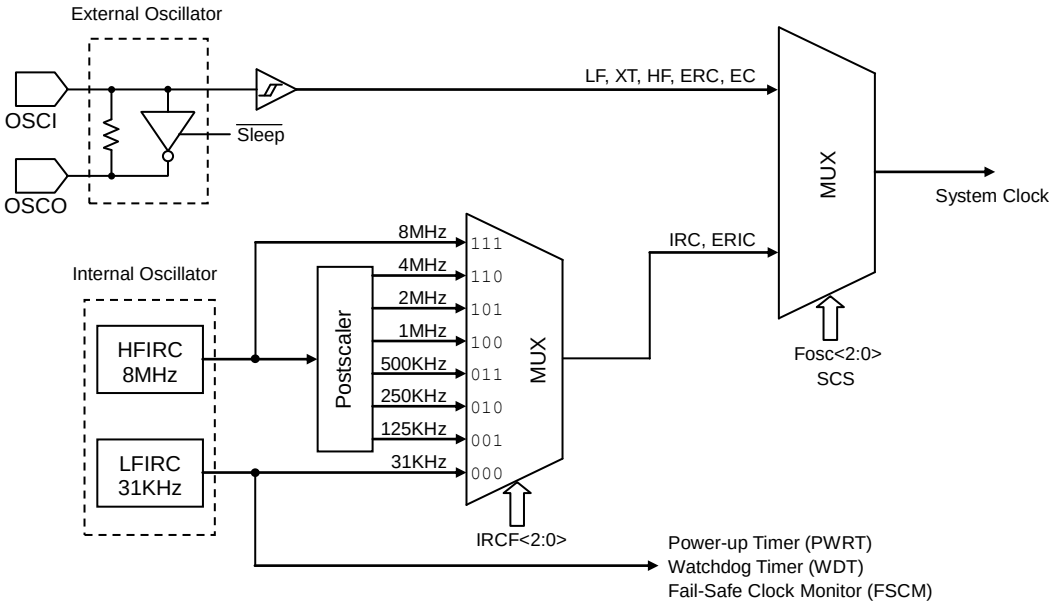
The Oscillator module can be configured in one of seven clock modes.

1. ERC: External Resistor/Capacitor Oscillator
2. HF: High Frequency Crystal/Resonator Oscillator
3. XT: Crystal/Resonator Oscillator
4. LF: Low Frequency Crystal Oscillator
5. IRC: Internal Resistor/Capacitor Oscillator
6. ERIC: External Resistor/Internal Capacitor Oscillator
7. EC: External Clock Input

Where in ERC, IRC ERIC and EC modes, the OSCO pin with the instruction cycle rate output or I/O can be selected.

Clock Source modes are configured by the Fosc<2:0> bits in the configuration word register (CONFIG). The internal clock can be generated from two internal oscillators. The HFIRC (8MHz) is a calibrated high-frequency oscillator. The LFIRC (31KHz) is an un-calibrated low-frequency oscillator.
The Oscillator Control register (OSCCON) controls the system clock and frequency selection options. The OSCCON register contains IRCF (frequency selection bits), HTS / LTS (frequency Status bits), & OSTS / SCS (system clock control bits) bits, see section 2.1.29 for detail description.

FIGURE 2.33: System Clock Source Block Diagram



2.14.1 External Clock Modes

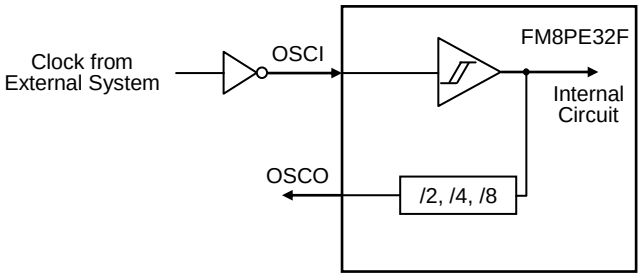
External Clock modes rely on external circuitry for the clock source. Examples are: Oscillator modules (EC mode), crystal resonators or ceramic resonators (LF, XT and HF modes) and Resistor-Capacitor (ERC) mode circuits

Note: 1. The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bit of the OSCCON register.
2. In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected.

2.14.1.1 EC Oscillator Mode

The External Clock (EC) mode allows an externally generated logic level as the system clock source.

FIGURE 2.34: EC Oscillator Mode (External Clock Input Operation)

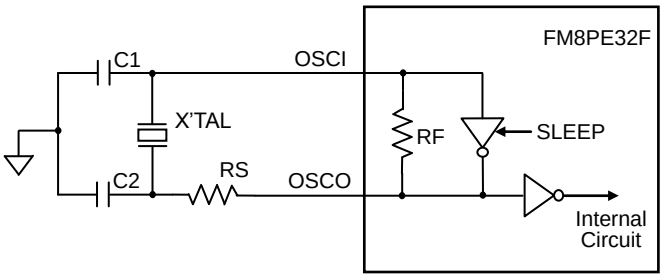


2.14.1.2 LF, XT, HF Modes

The LF, XT and HF modes support the use of crystal resonators or ceramic resonators connected to OSCI and OSCO. The mode selects a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

LF Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LF mode current consumption is the least of the three modes. This mode is designed to drive only 32.768kHz crystals.
XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.
HF Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HF mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

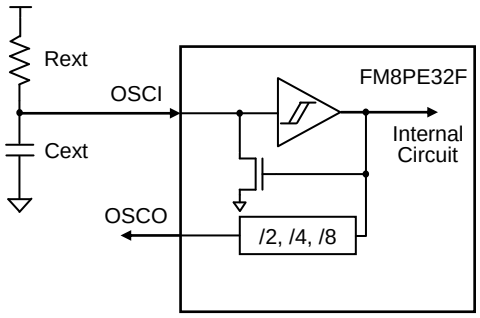
FIGURE 2.35: HF, XT, or LF Oscillator Modes (Crystal Operation or Ceramic Resonator)



2.14.1.3 External RC (ERC) Mode

The external Resistor-Capacitor (ERC) mode supports the use of an external RC circuit. This allows the designer maximum flexibility in frequency choice while keeping costs to a minimum when clock accuracy is not required. The ERC oscillator frequency is a function of the supply voltage, the resistor (Rext) and capacitor (Cext) values, the operating temperature, threshold voltage variation, component tolerances, and packaging variations in capacitance

FIGURE 2.36: ERC Oscillator Mode (External RC Oscillator)



2.14.2 Internal Clock Modes

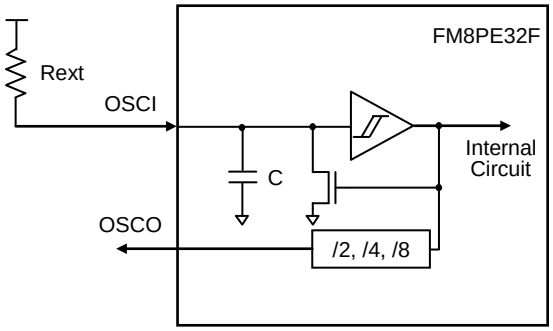
Internal clock sources are contained internally within the IRC/ERIC module. The IRC module has two internal oscillators: the 8MHz High-Frequency Internal Oscillator (HFIRC) and the 31kHz Low-Frequency Internal Oscillator (LFIRC) that can be configured or selected as the system clock source.

HFIRC (High-Frequency Internal Oscillator) mode is factory calibrated and operates at 8 MHz. The frequency of the HFIRC can be user-adjusted via software using the OSCTUNE register.
LFIRC (Low-Frequency Internal Oscillator) mode is un-calibrated and operates at 31 kHz.
ERIC (External R, Internal C Oscillator) mode is un-calibrated and operating frequency is based on the external resistor value.

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<2:0> of the OSCCON register.
Note: The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bit of the OSCCON register.

2.14.2.1 ERIC Mode

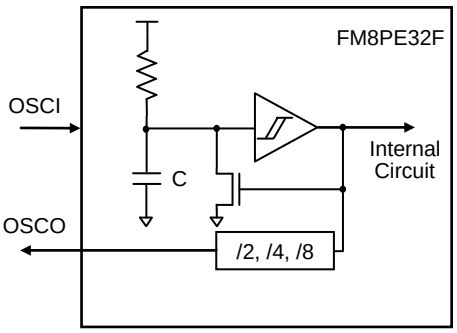
FIGURE 2.37: ERIC Oscillator Mode (External R, Internal C Oscillator)



2.14.2.2 HFIRC Mode

The High-Frequency Internal Oscillator (HFIRC) is a factory calibrated 8 MHz internal clock source. The frequency of the HFIRC can be altered via software using the OSCTUNE register.
The output of the HFIRC connects to a postscaler and multiplexer. One of seven frequencies can be selected via software using the IRCF<2:0> bits of the OSCCON register.
The HFIRC is enabled by selecting any frequency between 8 MHz and 125 kHz by setting the IRCF<2:0> bits of the OSCCON register ≠ 000. Then, set the System Clock Source (SCS) bit of the OSCCON register to '1' or enable Two-Speed Start-up by setting the IESO bit in the Configuration Word register (CONFIG) to '1'.
The HRIRC status bit (HTS) of the OSCCON register indicates whether the HFIRC is stable or not.

FIGURE 2.38: IRC Oscillator Mode (Internal R, Internal C Oscillator)



2.14.2.3 LFIRC Mode

The Low-Frequency Internal Oscillator (LFIRC) is an un-calibrated 31 kHz internal clock source.
The output of the LFIRC connects to a multiplexer. Select 31 kHz, via software, using the IRCF<2:0> bits of the OSCCON register. See section 2.1.29 for more information. The LFIRC is also the frequency for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).
The LFIRC is enabled by selecting 31 kHz (IRCF<2:0> bits of the OSCCON register = 000) as the system clock source (SCS bit of the OSCCON register = 1), or when any of the following are enabled:
• Two-Speed Start-up IESO bit of the CONFIG register = 1 and IRCF<2:0> bits of the OSCCON register = 000
• Power-up Timer (PWRT)
• Watchdog Timer (WDT)
• Fail-Safe Clock Monitor (FSCM)
The LFIRC status bit (LTS) of the OSCCON register indicates whether the LFIRC is stable or not.

2.14.2.4 IRC Frequency Select Bits (IRCF)

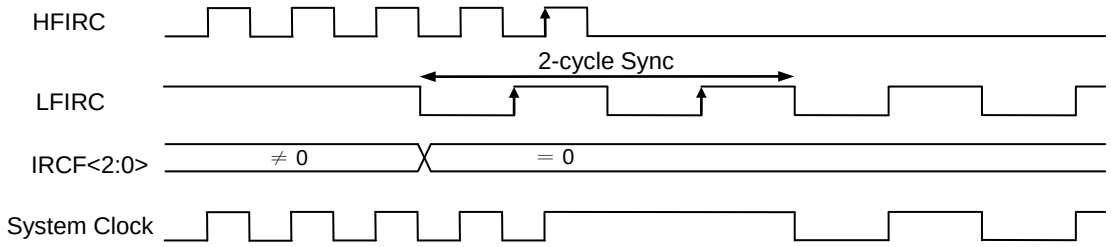
The output of the 8 MHz HFIRC and 31 kHz LFIRC connects to a postscaler and multiplexer (see Figure 2.33). The Internal Oscillator Frequency Select bits IRCF<2:0> of the OSCCON register select the frequency output of the internal oscillators. One of eight frequencies can be selected via software: 8MHz, 4MHz (Default after Reset), 2MHz, 1MHz, 500kHz, 250kHz, 125kHz, 31kHz (LFIRC). The frequency between 8 MHz and 125 kHz are derived from the HFIRC (8MHz) via the postscaler and multiplexer.

2.14.3 HFIRC and LFIRC Clock Switch Timing (IRCF<2:0> Bits Changing)

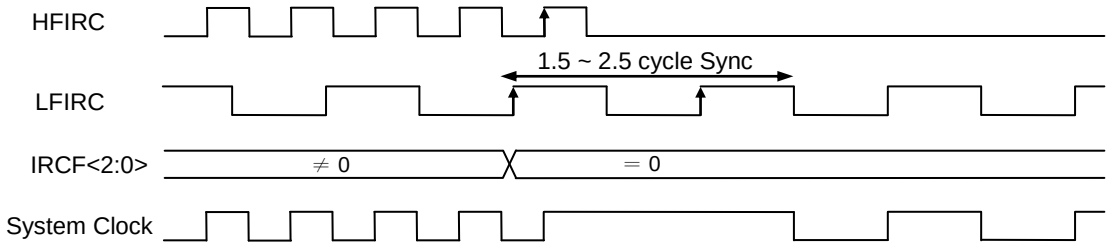
When switching between the LFIRC and the HFIRC, the new oscillator may already be shut down to save power. If this is the case, there is a delay after the IRCF<2:0> bits of the OSCCON register are modified before the frequency selection takes place. The LTS and HTS bits of the OSCCON register will reflect the current active status of the LFIRC and HFIRC oscillators. The timing of a frequency selection is as following figure 2.39.

FIGURE 2.39: Internal Oscillator Switch Timing

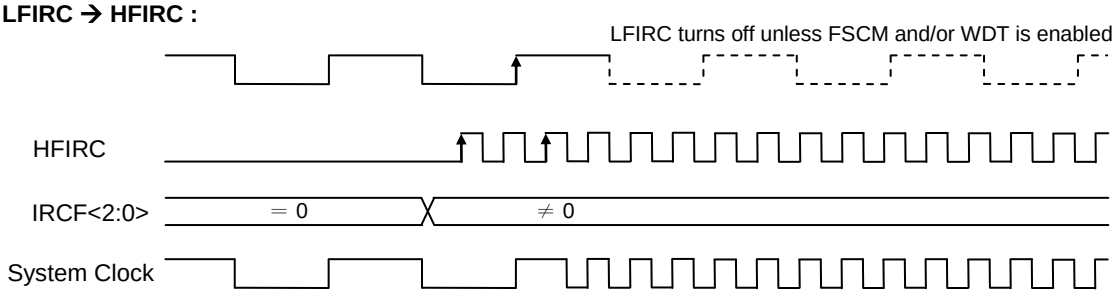
HFIRC → LFIRC (FSCM and WDT Disabled) :



HFIRC → LFIRC (Either FSCM or WDT Enabled) :



LFIRC → HFIRC :



2.14.4 Clock Switching (SCS bit Changing)

The system clock source can be switched between external and internal clock sources via software using the System Clock Select (SCS) bit of the OSCCON register.

The System Clock Select (SCS) bit of the OSCCON register selects the system clock source that is used for the CPU and peripherals. After a Reset, the SCS bit of the OSCCON register is always cleared.

- **SCS = 0**, the system clock source is determined by configuration of the Fosc<2:0> bits in the Configuration Word register (CONFIG).
- **SCS = 1**, the system clock source is chosen by the internal oscillator frequency selected by the IRCF<2:0> bits of the OSCCON register.

The Oscillator Start-up Time-out Status (OSTS) bit of the OSCCON register indicates that the system clock is running from the external clock source, as defined by the Fosc<2:0> bits in the Configuration Word register (CONFIG) if OSTS = 1, or from the internal clock source if OSTS = 0. In particular, OSTS indicates that the Oscillator Start-up Timer (OST) has timed out for LF, XT or HF modes.

Any automatic clock switch, which may occur from Two-Speed Start-up or Fail-Safe Clock Monitor, does not update the SCS bit of the OSCCON register. The user can monitor the OSTS bit of the OSCCON register to determine the current system clock source.

2.14.5 Two-Speed Clock Start-up Mode

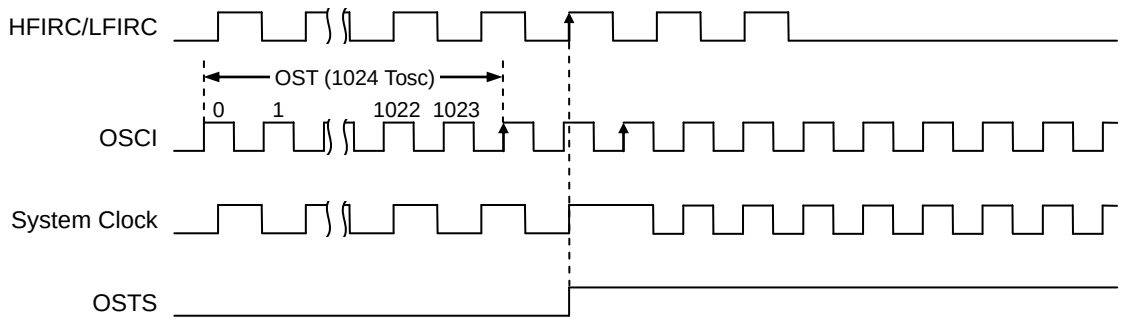
Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device.
This mode allows the application to wake-up from Sleep, perform a few instructions using the IRC as the clock source and go back to sleep without waiting for the primary oscillator to become stable.
When the Oscillator module is configured for LF, XT or HF modes, the Oscillator Start-up Timer (OST) is enabled. The OST will suspend program execution until 1024 oscillations are counted. Two-Speed Start-up mode minimizes the delay in code execution by operating from the internal oscillator as the OST is counting. When the OST count reaches 1024 and the OSTS bit of the OSCCON register is set, program execution switches to the external oscillator. Two-Speed Start-up mode is entered after power-on reset (POR) and, if enabled, after power-up timer (PWRT) has expired, or wake-up from sleep.
Two-Speed Start-up mode is configured by the following settings:
• IESO (of the Configuration Word register) = 1.
• SCS (of the OSCCON register) = 0.
• Fosc<2:0> bits in the Configuration Word register (CONFIG) configured for LF, XT or HF mode.

If the external clock oscillator is configured to be anything other than LF, XT or HF mode, then Two-Speed Start-up is disabled. This is because the external clock oscillator does not require any stabilization time after POR or an exit from Sleep.
Checking the state of the OSTS bit of the OSCCON register will confirm if the microcontroller is running from the external clock source, as defined by the Fosc<2:0> bits in the Configuration Word register (CONFIG), or the internal oscillator.

2.14.5.1 Two-Speed Start-up Sequence

1. Wake-up from Power-on Reset or Sleep.
2. Instructions begin execution by the internal oscillator at the frequency set in the IRCF<2:0> bits of the OSCCON register.
3. OST enabled to count 1024 external clock cycles (LF, XT or HF mode).
4. OST timed out, waiting for rising edge of the internal oscillator.
5. OSTS is set.
6. System clock held high until the next rising edge of external oscillator.
7. System clock is switched to external clock source.

FIGURE 2.40: Two-Speed Start-up



2.14.6 Fail-Safe Clock Monitor (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue operating while the external oscillator is fail. The FSCM can detect oscillator failure any time after the Oscillator Start-up Timer (OST) has expired. The FSCM is enabled by setting the FCMEN bit in the Configuration Word register (CONFIG). The FSCM is applicable to all external oscillator modes (LF, XT, HF, EC, ERC).
The FSCM module detects a failed oscillator by comparing the external oscillator to the FSCM sample clock. The sample clock is generated by dividing the LFIRC by 64. A failure is detected when no oscillator clock appears in the high level of the sample clock.

FIGURE 2.41: FSCM Block Diagram

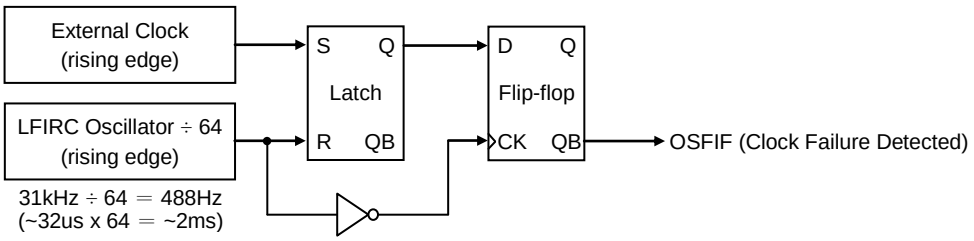
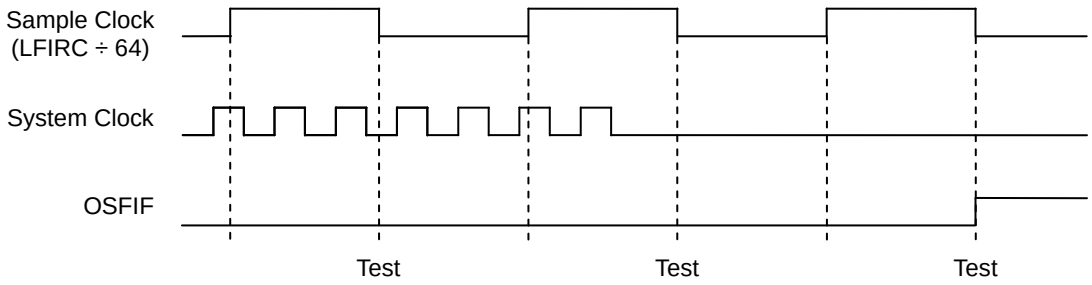


FIGURE 2.42: FSCM Timing Diagram



When the external clock fails, the FSCM switches the device clock to an internal clock source and sets the bit flag OSFIF of the INTFLAG register, also an interrupt will occur if the OSFIE bit of the INTEN register is also set. The system clock will continue to be sourced from the internal clock source until the device firmware successfully restarts the external oscillator and switches back to external operation.
The internal clock source chosen by the FSCM is determined by the IRCF<2:0> bits of the OSCCON register. This allows the internal oscillator to be configured before a failure occurs.
The FSCM is designed to detect an oscillator failure after the Oscillator Start-up Timer (OST) has expired. The OST is used after waking up from sleep and after any type of reset. The OST is not used with the EC or RC Clock modes so that the FSCM will be active as soon as the reset or wake-up has completed. When the FSCM is enabled, the Two-Speed Start-up is also enabled. Therefore, the device will always be executing code while the OST is operating.
The Fail-Safe circuit is not active during oscillator start-up (i.e., after exiting Reset or Sleep). After an appropriate amount of time, the user should check the OSTS bit of the OSCCON register to verify the oscillator start-up and that the system clock switchover has successfully completed.
The Fail-Safe condition is cleared after a Reset, executing a SLEEP instruction or toggling the SCS bit of the OSCCON register. When the SCS bit is toggled, the OST is restarted. While the OST is running, the device continues to operate from the IRC selected in OSCCON. When the OST times out, the Fail-Safe condition is cleared and the device will be operating from the external clock source. The Fail-Safe condition must be cleared before the OSFIF flag can be cleared.

2.15 Configurations Word

TABLE 2.9: Configurations Word

Name	Description
Fosc<2:0>	Oscillator Selection Bits = 1, 1, 1 → ERC mode (external R & C) = 1, 1, 0 → HF mode = 1, 0, 1 → XT mode = 1, 0, 0 → LF mode = 0, 1, 1 → IRC mode (internal R & C) = 0, 1, 0 → ERIC mode (external R & internal C) = 0, 0, 0 → EC mode (external clock input) (default)
LVDT<2:0>	Low Voltage Detector Selection Bits = 1, 1, 1 → enable, LVDT voltage = 2.7V = 1, 1, 0 → enable, LVDT voltage = 2.0V = 1, 0, 1 → enable, LVDT voltage = 2.7V, controlled by LVDTE bit of PCON = 1, 0, 0 → enable, LVDT voltage = 2.0V, controlled by LVDTE bit of PCON = 0, 1, 1 → enable, LVDT voltage = 2.7V, disabled in SLEEP = 0, 1, 0 → enable, LVDT voltage = 2.0V, disabled in SLEEP = 0, 0, 0 → disable (default)
PWRT<1:0>	Power-up Reset Timer Selection Bits = 1, 1 → disabled = 1, 0 → 18ms = 0, 1 → 36ms = 0, 0 → 72ms (default)
IESO	Two Speed Start Mode Enable Bit = 1, Disable two speed start mode = 0, Enable two speed start mode (default)
FCMEN	Fail-Safe Clock Monitor Enable Bit = 1, Disable fail-safe clock monitor = 0, Enable fail-safe clock monitor, IESO function will be forced to enable (default)
WRONLYE	EEPROM Write-only Function Enable Bit = 1, Allows EEPROM Write-only Function = 0, Inhibits EEPROM Write-only Function (default)
OSCOUT	IOA4/OSCO Pin Selection Bit for ERC/IRC/ERIC/EC Mode = 1, OSCO pin is selected; Instruction clock will be output = 0, IOA4 pin is selected (default)
RSTBIN	IOA3/RSTB Pin Selection Bit = 1, IOA3 pin is selected = 0, RSTB pin is selected (default)
WDTEN	Watchdog Timer Enable Bit = 1, WDT fixed to be enabled = 0, WDT disabled and can be enabled by WDTE of WDTCON register (default)
CPROTECT	Program Code Protection Bit = 1, Program code protection on = 0, Program code protection off (default)
DPROTECT	EEPROM Data Protection Bit = 1, EEPROM data protection on = 0, EEPROM data protection off (default)
OSCD<1:0>	Instruction Period Selection Bits = 1, 1 → Four oscillator periods = 1, 0 → Two oscillator periods = 0, 0 → Eight oscillator periods (default)

Name	Description
PKG	Operation Package/Voltage Selection Bit = 1, HV package (**FM8PE32AF**) is selected, valid only on Vdd ≧ 3.3V = 0, LV package (**FM8PE32BF**) is selected, valid only on Vdd ≦ 3.3V (default)
RAIF	Port A Change Interrupt Flag Option Bit = 1, Edge trigger = 0, Level trigger (default)
CAL<6:0>	Calibration Selection Bits for IRC Mode

3.0 INSTRUCTION SET

Mnemonic, Operands		Description	Operation	Cycles	Status Affected
BCR	**R, bit**	Clear bit in R	0 → R<b>	1	-
BSR	**R, bit**	Set bit in R	1 → R<b>	1	-
BTRSC	**R, bit**	Test bit in R, Skip if Clear	Skip if R<b> = 0	1/2 (1)	-
BTRSS	**R, bit**	Test bit in R, Skip if Set	Skip if R<b> = 1	1/2 (1)	-
NOP		No Operation	No operation	1	-
CLRWDT		Clear Watchdog Timer	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$, $\overline{PD}$
OPTION		Load OPTION_REG register	ACC → OPTION_REG	1	-
SLEEP		Go into power-down mode	00h → WDT, 00h → WDT prescaler	1	$\overline{TO}$, $\overline{PD}$
IOST	**R**	Load IOST register	ACC → IOST register	1	-
DAA		Adjust ACC's data format from HEX to DEC after any addition operation	ACC(hex) → ACC(dec)	1	C
DAS		Adjust ACC's data format from HEX to DEC after any subtraction operation	ACC(hex) → ACC(dec)	1	-
INT		S/W interrupt	PC + 1 → Top of Stack, 002h → PC	2	-
RETURN		Return from subroutine	Top of Stack → PC	2	-
RETFIE		Return from interrupt, set GIE bit	Top of Stack → PC, 1 → GIE	2	-
CLRA		Clear ACC	00h → ACC	1	Z
CLRR	**R**	Clear R	00h → R	1	Z
MOVAR	**R**	Move ACC to R	ACC → R	1	-
MOVR	**R, d**	Move R	R → dest	1	Z
DECR	**R, d**	Decrement R	R - 1 → dest	1	Z
DECRSZ	**R, d**	Decrement R, Skip if 0	R - 1 → dest, Skip if result = 0	1/2 (1)	-
INCR	**R, d**	Increment R	R + 1 → dest	1	Z
INCRSZ	**R, d**	Increment R, Skip if 0	R + 1 → dest, Skip if result = 0	1/2 (1)	-
ADDAR	**R, d**	Add ACC and R	R + ACC → dest	1	C, DC, Z
SUBAR	**R, d**	Subtract ACC from R	R - ACC → dest	1	C, DC, Z
ADCAR	**R, d**	Add ACC and R with Carry	R + ACC + C → dest	1	C, DC, Z
SBCAR	**R, d**	Subtract ACC from R with Carry	R + $\overline{ACC}$ + C → dest	1	C, DC, Z
ANDAR	**R, d**	AND ACC with R	ACC and R → dest	1	Z
IORAR	**R, d**	Inclusive OR ACC with R	ACC or R → dest	1	Z
XORAR	**R, d**	Exclusive OR ACC with R	R xor ACC → dest	1	Z
COMR	**R, d**	Complement R	$\overline{R}$ → dest	1	Z
RLR	**R, d**	Rotate left R through Carry	R<7> → C, R<6:0> → dest<7:1>, C → dest<0>	1	C
RRR	**R, d**	Rotate right R through Carry	C → dest<7>, R<7:1> → dest<6:0>, R<0> → C	1	C

Mnemonic, Operands	Description	Operation	Cycles	Status Affected
SWAPR R, d	Swap R	R<3:0> → dest<7:4>, R<7:4> → dest<3:0>	1	-
MOVIA I	Move Immediate to ACC	I → ACC	1	-
ADDIA I	Add ACC and Immediate	I + ACC → ACC	1	C, DC, Z
SUBIA I	Subtract ACC from Immediate	I - ACC → ACC	1	C, DC, Z
ADCIA I	Add ACC and Immediate with Carry	I + ACC + C → ACC	1	C, DC, Z
SBCIA I	Subtract ACC from Immediate with Carry	I + $\overline{\text{ACC}}$ + C → ACC	1	C, DC, Z
ANDIA I	AND Immediate with ACC	ACC and I → ACC	1	Z
IORIA I	OR Immediate with ACC	ACC or I → ACC	1	Z
XORIA I	Exclusive OR Immediate to ACC	ACC xor I → ACC	1	Z
RETIA I	Return, place Immediate in ACC	I → ACC, Top of Stack → PC	2	-
CALL I	Call subroutine	PC + 1 → Top of Stack, I → PC	2	-
GOTO I	Unconditional branch	I → PC	2	-

Note: 1. 2 cycles for skip, else 1 cycle

2. bit : Bit address within an 8-bit register R
   R : Register address (00h to 7Fh)
   I : Immediate data
   ACC : Accumulator
   d : Destination select;
   =0 (store result in ACC)
   =1 (store result in file register R)
   dest : Destination
   PC : Program Counter
   PCHBUF : High Byte Buffer of Program Counter
   WDT : Watchdog Timer Counter
   GIE : Global interrupt enable bit
   $\overline{\text{TO}}$: Time-out bit
   $\overline{\text{PD}}$: Power-down bit
   C : Carry bit
   DC : Digital carry bit
   Z : Zero bit

ADCAR	**Add ACC and R with Carry**
Syntax:	ADCAR R, d
Operands:	0≤R≤0x7F d∈[0,1]
Operation:	R + ACC + C → dest
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ADCIA	**Add ACC and Immediate with Carry**
Syntax:	ADCIA I
Operands:	0≤I≤0xFF
Operation:	ACC + I + C → ACC
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register with the 8-bit immediate 'I' with Carry. The result is placed in the ACC register.
Cycles:	1

ADDAR	**Add ACC and R**
Syntax:	ADDAR R, d
Operands:	0≤R≤0x7F d∈[0,1]
Operation:	ACC + R → dest
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register and register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ADDIA	**Add ACC and Immediate**
Syntax:	ADDIA I
Operands:	0≤I≤0xFF
Operation:	ACC + I → ACC
Status Affected:	C, DC, Z
Description:	Add the contents of the ACC register with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

ANDAR	**AND ACC and R**
Syntax:	ANDAR R, d
Operands:	0≤R≤0x7F d∈[0,1]
Operation:	ACC and R → dest
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is '1' the result is stored back in register 'R'.
Cycles:	1

ANDIA	**AND Immediate with ACC**
Syntax:	ANDIA I
Operands:	0≤I≤0xFF
Operation:	ACC AND I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are AND'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

BCR	**Clear Bit in R**
Syntax:	BCF R, b
Operands:	0≤R≤0x7F 0≤b≤7
Operation:	0 → R<b>
Status Affected:	None
Description:	Clear bit 'b' in register 'R'.
Cycles:	1

BSR	**Set Bit in R**
Syntax:	BSR R, b
Operands:	0≤R≤0x7F 0≤b≤7
Operation:	1 → R<b>
Status Affected:	None
Description:	Set bit 'b' in register 'R'.
Cycles:	1

BTRSC	**Test Bit in R, Skip if Clear**
Syntax:	BTRSC R, b
Operands:	0≤R≤0x7F 0≤b≤7
Operation:	Skip if R<b> = 0
Status Affected:	None
Description:	If bit 'b' in register 'R' is 0 then the next instruction is skipped. If bit 'b' is 0 then next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead making this a 2-cycle instruction.
Cycles:	1/2

BTRSS	**Test Bit in R, Skip if Set**
Syntax:	BTRSS R, b
Operands:	0≤R≤0x7F 0≤b≤7
Operation:	Skip if R<b> = 1
Status Affected:	None
Description:	If bit 'b' in register 'R' is '1' then the next instruction is skipped. If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2-cycle instruction.
Cycles:	1/2

CALL	**Subroutine Call**
Syntax:	CALL I
Operands:	$0 \leq I \leq 0x7FF$
Operation:	PC +1 → Top of Stack; I → PC
Status Affected:	None
Description:	Subroutine call. First, return address (PC+1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. CALL is a two-cycle instruction.
Cycles:	2

CLRA	**Clear ACC**
Syntax:	CLRA
Operands:	None
Operation:	00h → ACC; 1 → Z
Status Affected:	Z
Description:	The ACC register is cleared. Zero bit (Z) is set.
Cycles:	1

CLRR	**Clear R**
Syntax:	CLRR R
Operands:	$0 \leq R \leq 0x7F$
Operation:	00h → R; 1 → Z
Status Affected:	Z
Description:	The contents of register 'R' are cleared and the Z bit is set.
Cycles:	1

CLRWDT	**Clear Watchdog Timer**
Syntax:	CLRWDT
Operands:	None
Operation:	00h → WDT; 00h → WDT prescaler (if assigned); 1 → $\overline{TO}$; 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$ $\overline{PD}$
Description:	The CLRWDT instruction resets the WDT. It also resets the prescaler, if the prescaler is assigned to the WDT and not Timer0. Status bits $\overline{TO}$ and $\overline{PD}$ are set.
Cycles:	1

COMR	**Complement R**
Syntax:	COMR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	$\overline{R}$ → dest
Status Affected:	Z
Description:	The contents of register 'R' are complemented. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DAA	**Adjust ACC's data format from HEX to DEC**
Syntax:	DAA
Operands:	None
Operation:	ACC(hex) → ACC(dec)
Status Affected:	C
Description:	Convert the ACC data from hexadecimal to decimal format after any addition operation and restored to ACC.
Cycles:	1

DAS	**Adjust ACC's data format from HEX to DEC**
Syntax:	DAS
Operands:	None
Operation:	ACC(hex) → ACC(dec)
Status Affected:	None
Description:	Convert the ACC data from hexadecimal to decimal format after any subtraction operation and restored to ACC.
Cycles:	1

DECR	**Decrement R**
Syntax:	DECR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R - 1 → dest
Status Affected:	Z
Description:	Decrement register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

DECRSZ	**Decrement R, Skip if 0**
Syntax:	DECRSZ R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R - 1 → dest; skip if result =0
Status Affected:	None
Description:	The contents of register 'R' are decremented. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'. If the result is 0, the next instruction, which is already fetched, is discarded and a NOP is executed instead making it a two-cycle instruction.
Cycles:	1/2

GOTO	**Unconditional Branch**
Syntax:	GOTO I
Operands:	$0 \leq I \leq 0x7FF$
Operation:	I → PC
Status Affected:	None
Description:	GOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>. GOTO is a two-cycle instruction.
Cycles:	2

INCR	**Increment R**
Syntax:	INCR R, d
Operands:	0 ≤ R ≤ 0x7F d ∈ [0,1]
Operation:	R + 1 → dest
Status Affected:	Z
Description:	The contents of register 'R' are incremented. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

INCRSZ	**Increment R, Skip if 0**
Syntax:	INCRSZ R, d
Operands:	0 ≤ R ≤ 0x7F d ∈ [0,1]
Operation:	R + 1 → dest, skip if result = 0
Status Affected:	None
Description:	The contents of register 'R' are incremented. If 'd' is 0 the result is placed in the ACC register. If 'd' is the result is placed back in register 'R'. If the result is 0, then the next instruction, which is already fetched, is discarded and a NOP is executed instead making it a two-cycle instruction.
Cycles:	1/2

INT	**S/W Interrupt**
Syntax:	INT
Operands:	None
Operation:	PC + 1 → Top of Stack, 002h → PC
Status Affected:	None
Description:	Interrupt subroutine call. First, return address (PC+1) is pushed onto the stack. The address 002h is loaded into PC bits <10:0>.
Cycles:	2

IORAR	**OR ACC with R**
Syntax:	IORAR R, d
Operands:	0 ≤ R ≤ 0x7F d ∈ [0,1]
Operation:	ACC or R → dest
Status Affected:	Z
Description:	Inclusive OR the ACC register with register 'R'. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

IORIA	**OR Immediate with ACC**
Syntax:	IORIA I
Operands:	0 ≤ I ≤ 0xFF
Operation:	ACC or I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are OR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

IOST	**Load IOST Register**
Syntax:	IOST R
Operands:	R = 0x05 or 0x07
Operation:	ACC → IOST register R
Status Affected:	None
Description:	IOST register 'R' (R= 0x05 or 0x07) is loaded with the contents of the ACC register.
Cycles:	1

MOVAR	**Move ACC to R**
Syntax:	MOVAR R
Operands:	$0 \leq R \leq 0x7F$
Operation:	ACC → R
Status Affected:	None
Description:	Move data from the ACC register to register 'R'.
Cycles:	1

MOVIA	**Move Immediate to ACC**
Syntax:	MOVIA I
Operands:	$0 \leq I \leq 0xFF$
Operation:	I → ACC
Status Affected:	None
Description:	The 8-bit immediate 'I' is loaded into the ACC register. The don't cares will assemble as 0s.
Cycles:	1

MOVR	**Move R**
Syntax:	MOVR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R → dest
Status Affected:	Z
Description:	The contents of register 'R' is moved to destination 'd'. If 'd' is 0, destination is the ACC register. If 'd' is 1, the destination is file register 'R'. 'd' is 1 is useful to test a file register since status flag Z is affected.
Cycles:	1

NOP	**No Operation**
Syntax:	NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.
Cycles:	1

OPTION	**Load OPTION_REG Register**
Syntax:	OPTION
Operands:	None
Operation:	ACC → OPTION_REG
Status Affected:	None
Description:	The content of the ACC register is loaded into the OPTION_REG register.
Cycles:	1

RETFIE	**Return from Interrupt, Set 'GIE' Bit**
Syntax:	RETFIE
Operands:	None
Operation:	Top of Stack → PC
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). The 'GIE' bit is set to 1. This is a two-cycle instruction.
Cycles:	2

RETIA	**Return with Immediate in ACC**
Syntax:	RETIA I
Operands:	$0 \leq I \leq 0xFF$
Operation:	I → ACC; Top of Stack → PC
Status Affected:	None
Description:	The ACC register is loaded with the 8-bit immediate 'I'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles:	2

RETURN	**Return from Subroutine**
Syntax:	RETURN
Operands:	None
Operation:	Top of Stack → PC
Status Affected:	None
Description:	The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Cycles:	2

RLR	**Rotate Left R through Carry**
Syntax:	RLR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R<7> → C; R<6:0> → dest<7:1>; C → dest<0>
Status Affected:	C
Description:	The contents of register 'R' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

RRR	**Rotate Right R through Carry**
Syntax:	RRR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	C → dest<7>; R<7:1> → dest<6:0>; R<0> → C
Status Affected:	C
Description:	The contents of register 'R' are rotated one bit to the right through the Carry Flag. If 'd' is 0 the result is placed in the ACC register. If 'd' is 1 the result is placed back in register 'R'.
Cycles:	1

SLEEP	**Enter SLEEP Mode**
Syntax:	SLEEP
Operands:	None
Operation:	00h → WDT; 00h → WDT prescaler; 1 → $\overline{TO}$; 0 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	Time-out status bit ($\overline{TO}$) is set. The power-down status bit ($\overline{PD}$) is cleared. The WDT and its prescaler are cleared. The processor is put into SLEEP mode.
Cycles:	1

SBCAR	**Subtract ACC from R with Carry**
Syntax:	SBCAR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R + $\overline{ACC}$ + C → dest
Status Affected:	C, DC, Z
Description:	Add the 2's complement data of the ACC register from register 'R' with Carry. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

SBCIA	**Subtract ACC from Immediate with Carry**
Syntax:	SBCIA I
Operands:	$0 \leq I \leq 0xFF$
Operation:	I + $\overline{ACC}$ + C → ACC
Status Affected:	C, DC, Z
Description:	Add the 2's complement data of the ACC register from the 8-bit immediate 'I' with Carry. The result is placed in the ACC register.
Cycles:	1

SUBAR	**Subtract ACC from R**
Syntax:	SUBAR R, d
Operands:	$0 \leq R \leq 0x7F$ $d \in [0,1]$
Operation:	R - ACC → dest
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

SUBIA	**Subtract ACC from Immediate**
Syntax:	SUBIA I
Operands:	$0 \leq I \leq 0xFF$
Operation:	I - ACC → ACC
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) the ACC register from the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

SWAPR	**Swap nibbles in R**
Syntax:	SWAPR R, d
Operands:	0 ≤ R ≤ 0x7F d ∈ [0,1]
Operation:	R<3:0> → dest<7:4>; R<7:4> → dest<3:0>
Status Affected:	None
Description:	The upper and lower nibbles of register 'R' are exchanged. If 'd' is 0 the result is placed in ACC register. If 'd' is 1 the result in placed in register 'R'.
Cycles:	1

XORAR	**Exclusive OR ACC with R**
Syntax:	XORAR R, d
Operands:	0 ≤ R ≤ 0x7F d ∈ [0,1]
Operation:	ACC xor R → dest
Status Affected:	Z
Description:	Exclusive OR the contents of the ACC register with register 'R'. If 'd' is 0 the result is stored in the ACC register. If 'd' is 1 the result is stored back in register 'R'.
Cycles:	1

XORIA	**Exclusive OR Immediate with ACC**
Syntax:	XORIA I
Operands:	0 ≤ I ≤ 0xFF
Operation:	ACC xor I → ACC
Status Affected:	Z
Description:	The contents of the ACC register are XOR'ed with the 8-bit immediate 'I'. The result is placed in the ACC register.
Cycles:	1

4.0 ABSOLUTE MAXIMUM RATINGS

Ambient Operating Temperature	0℃ to +70℃
Store Temperature	-65℃ to +150℃
DC Supply Voltage (Vdd)	0V to +6.0V
Input Voltage with respect to Ground (Vss)	-0.3V to (Vdd + 0.3)V

5.0 OPERATING CONDITIONS

DC Supply Voltage	+2.3V to +5.5V
Operating Temperature	0℃ to +70℃

6.0 ELECTRICAL CHARACTERISTICS

TA=25℃

Under Operating Conditions, at four clock instruction cycles and WDT & LVDT are disabled

Sym	Description	Conditions	Min.	Typ.	Max.	Unit
F_{HF}	X'tal oscillation range	HF mode, Vdd=5V	1		20	MHz
		HF mode, Vdd=3V	1		15	
F_{XT}	X'tal oscillation range	XT mode, Vdd=5V	1		20	MHz
		XT mode, Vdd=3V	1		15	
F_{LF}	X'tal oscillation range	LF mode, Vdd=5V	32		1000	KHZ
		LF mode, Vdd=3V	32		500	
F_{ERC}	RC oscillation range	ERC mode, Vdd=5V	DC		10	MHz
		ERC mode, Vdd=3V	DC		7	
$F_{IRC/ERIC}$	RC oscillation range	ERIC mode, external R, Vdd=5V	DC		15	MHz
		ERIC mode, external R, Vdd=3V	DC		7	
		IRC mode, internal R, Vdd=5V			8	
		IRC mode, internal R, Vdd=3V			8	
V_{IH}	Input high voltage	I/O ports, Vdd=5V	2.0			V
		T0CKI pins, Vdd=5V	3.0			
		I/O ports, Vdd=3V	1.5			
		T0CKI pins, Vdd=3V	1.5			
V_{IL}	Input low voltage	I/O ports, Vdd=5V			1.0	V
		T0CKI pins, Vdd=5V			1.0	
		I/O ports, Vdd=3V			0.6	
		T0CKI pins, Vdd=3V			0.6	
V_{OH}	Output high voltage	I_{OH}=-5.4mA, Vdd=5V	3.6			V
V_{OL}	Output low voltage	I_{OL}=8.7mA, Vdd=5V			0.6	V
I_{PH}	Pull-high current	Input pin at Vss, Vdd=5V		-200		uA
I_{ULP}	Ultra low-power wake-up current			200		nA
I_{WDT}	WDT current	Vdd=5V		9	12	uA
		Vdd=3V		2	4	
T_{WDT}	WDT period	Vdd=3V		17.6		mS
		Vdd=4V		16.5		
		Vdd=5V		16.4		
I_{LVDT}	LVDT current	Vdd=5V, LVDT = 2.7V		30	40	uA
		Vdd=5V, LVDT = 2V		23	30	
		Vdd=3V, LVDT = 2V		8.6	16	
I_{SB}	Power down current	Sleep mode, Vdd=5V, WDT disable		3		uA
		Sleep mode, Vdd=3V, WDT disable		1.1		
T_{WR}	Data EEPROM write cycle time	Under Operating Conditions		10.5		mS
T_{WRONLY}	Data EEPROM write-only cycle time	Under Operating Conditions		0.7		mS
E_D	Data EEPROM endurance	Under Operating Conditions	100K			E/W
E_P	Program Flash endurance	Under Operating Conditions	10K			E/W

Sym	Description	Conditions	Min.	Typ.	Max.	Unit
I_{DD}	Operating current	HF mode, Vdd=5V, 4 clock instruction, OSCI / OSCO = 3pF / 3pF				mA
		20MHz		2.495		
		16MHz		2.119		
		10MHz		1.533		
		4MHz		0.827		
		2MHz		0.597		
		HF mode, Vdd=3V, 4 clock instruction, OSCI / OSCO = 3pF / 3pF				
		20MHz		1.523		
		16MHz		1.297		
		10MHz		0.878		
		4MHz		0.454		
		2MHz		0.288		
		HF mode, Vdd=5V, 2 clock instruction, OSCI / OSCO = 3pF / 3pF				
		20MHz		3.534		
		16MHz		2.983		
		10MHz		2.066		
		4MHz		1.062		
		2MHz		0.702		
		HF mode, Vdd=3V, 2 clock instruction, OSCI / OSCO = 3pF / 3pF				
		20MHz		2.277		
		16MHz		1.914		
		10MHz		1.316		
		4MHz		0.608		
		2MHz		0.368		
I_{DD}	Operating current	XT mode, Vdd=5V, 4 clock instruction, OSCI / OSCO =20pF / 20pF				mA
		20MHz		2.074		
		16MHz		1.722		
		10MHz		1.205		
		4MHz		0.68		
		2MHz		0.514		
		XT mode, Vdd=5V, 4 clock instruction, OSCI / OSCO =20pF / 100pF				
		455KHz		0.423		
		100KHz		0.675		
		32KHz		0.689		
		XT mode, Vdd=3V, 4 clock instruction, OSCI / OSCO =20pF / 20pF				
		20MHz		1.378		
		16MHz		1.141		
		10MHz		0.743		
		4MHz		0.383		
		2MHz		0.272		

Sym	Description	Conditions	Min.	Typ.	Max.	Unit
I_{DD}	Operating current	XT mode, Vdd=3V, 4 clock instruction, OSCI / OSCO =20pF / 100pF				mA
		455KHz		0.199		
		100KHz		0.152		
		32KHz		0.135		
		XT mode, Vdd=5V, 2 clock instruction, OSCI / OSCO =20pF / 20pF				
		20MHz		2.973		
		16MHz		2.461		
		10MHz		1.664		
		4MHz		0.859		
		2MHz		0.602		
		XT mode, Vdd=5V, 2 clock instruction, OSCI / OSCO =20pF / 100pF				
		455KHz		0.439		
		100KHz		0.674		
		32KHz		0.686		
		XT mode, Vdd=3V, 2 clock instruction, OSCI / OSCO =20pF / 20pF				
		20MHz		2.123		
		16MHz		1.751		
		10MHz		1.165		
		4MHz		0.539		
		2MHz		0.352		
		XT mode, Vdd=3V, 2 clock instruction, OSCI / OSCO =20pF / 100pF				
		455KHz		0.217		
		100KHz		0.155		
		32KHz		0.137		
I_{DD}	Operating current	LF mode, Vdd=5V, 4 clock instruction, OSCI / OSCO =10pF / 20pF				uA
		2MHz		-		
		1MHz		198.61		
		455KHz		142.88		
		100KHz		106.97		
		32KHz		96.88		
		LF mode, Vdd=3V, 4 clock instruction, OSCI / OSCO =10pF / 20pF				
		2MHz		-		
		1MHz		-		
		455KHz		94.39		
		100KHz		60.69		
		32KHz		52.04		
		LF mode, Vdd=5V, 2 clock instruction, OSCI / OSCO =10pF / 20pF				
		2MHz		-		
		1MHz		249.47		
		455KHz		166.54		
		100KHz		112.18		
		32KHz		98.85		

Sym	Description	Conditions			Min.	Typ.	Max.	Unit
I_{DD}	Operating current	LF mode, Vdd=3V, 2 clock instruction, OSCI / OSCO =10pF / 20pF						uA
		2MHz				-		
		1MHz				-		
		455KHz				113.09		
		100KHz				64.75		
		32KHz				53.41		
I_{DD}	Operating current	ERC mode, Vdd=5V, 4 clock instruction						mA
		C=3P	R=1Kohm	F=11.76MHz		3.945		
			R=3.3Kohm	F=9.14MHz		1.949		
			R=10Kohm	F=5.3MHz		0.961		
			R=100Kohm	F=848KHz		0.216		
			R=300Kohm	F=298KHz		0.125		
		C=20P	R=1Kohm	F=9.11MHz		3.807		
			R=3.3Kohm	F=5.62MHz		1.594		
			R=10Kohm	F=2.61MHz		0.674		
			R=100Kohm	F=336KHz		0.161		
			R=300Kohm	F=113KHz		0.115		
		C=100P	R=1Kohm	F=4.59MHz		3.448		
			R=3.3Kohm	F=2.1MHz		1.223		
			R=10Kohm	F=823KHz		0.486		
			R=100Kohm	F=92KHz		0.133		
			R=300Kohm	F=28KHz		0.104		
		C=300P	R=1Kohm	F=2.26MHz		3.212		
			R=3.3Kohm	F=944KHz		1.08		
			R=10Kohm	F=344KHz		0.428		
			R=100Kohm	F=36KHz		0.126		
			R=300Kohm	F=12KHz		0.102		
		ERC mode, Vdd=3V, 4 clock instruction						
		C=3P	R=1Kohm	F=7.01MHz		2.177		
			R=3.3Kohm	F=5.91MHz		0.986		
			R=10Kohm	F=4.25MHz		0.553		
			R=100Kohm	F=896KHz		0.137		
			R=300Kohm	F=332KHz		0.08		
		C=20P	R=1Kohm	F=5.96MHz		2.144		
			R=3.3Kohm	F=4.37MHz		0.907		
			R=10Kohm	F=2.48MHz		0.426		
			R=100Kohm	F=378KHz		0.097		
			R=300Kohm	F=132KHz		0.065		
		C=100P	R=1Kohm	F=3.87MHz		2.104		
			R=3.3Kohm	F=2.07MHz		0.759		
			R=10Kohm	F=889KHz		0.309		
			R=100Kohm	F=104KHz		0.076		
			R=300Kohm	F=36KHz		0.057		

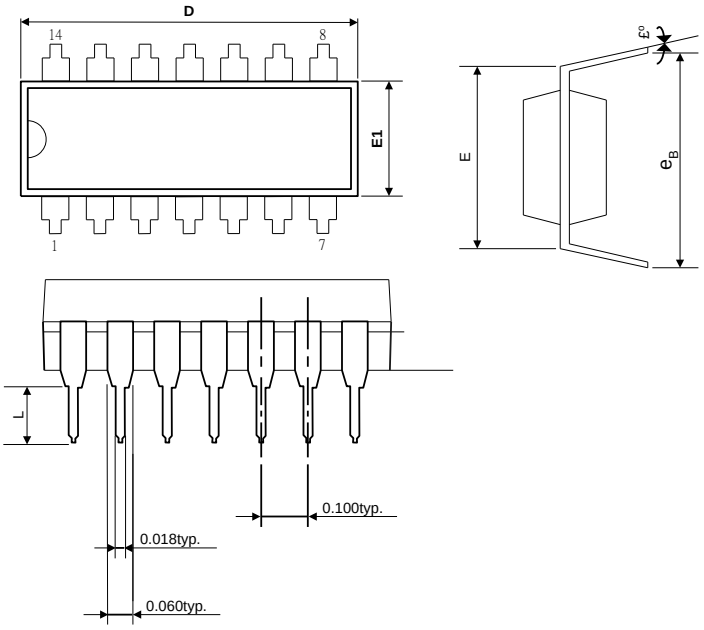
Sym	Description	Conditions			Min.	Typ.	Max.	Unit
I_{DD}	Operating current	C=300P	R=1Kohm	F=2.25MHz		2.016		mA
			R=3.3Kohm	F=997KHz		0.681		
			R=10Kohm	F=384KHz		0.27		
			R=100Kohm	F=40KHz		0.07		
			R=300Kohm	F=12KHz		0.055		
I_{DD}	Operating current	ERC mode, Vdd=5V, 2 clock instruction						mA
		C=3P	R=1Kohm	F=11.45MHz		4.811		
			R=3.3Kohm	F=9MHz		2.671		
			R=10Kohm	F=5.45MHz		1.43		
			R=100Kohm	F=910KHz		0.293		
			R=300Kohm	F=326KHz		0.163		
		C=20P	R=1Kohm	F=9.06MHz		4.527		
			R=3.3Kohm	F=5.67MHz		2.065		
			R=10Kohm	F=2.62MHz		0.883		
			R=100Kohm	F=339KHz		0.188		
			R=300Kohm	F=116KHz		0.124		
		C=100P	R=1Kohm	F=4.61MHz		3.828		
			R=3.3Kohm	F=2.1MHz		1.393		
			R=10Kohm	F=823KHz		0.55		
			R=100Kohm	F=92KHz		0.14		
			R=300Kohm	F=30KHz		0.107		
		C=300P	R=1Kohm	F=2.27MHz		3.395		
			R=3.3Kohm	F=938KHz		1.158		
			R=10Kohm	F=344KHz		0.455		
			R=100Kohm	F=36KHz		0.129		
			R=300Kohm	F=12KHz		0.104		
		ERC mode, Vdd=3V, 2 clock instruction						
		C=3P	R=1Kohm	F=6.92MHz		2.547		
			R=3.3Kohm	F=6.02MHz		1.372		
			R=10Kohm	F=4.31MHz		0.794		
			R=100Kohm	F=965KHz		0.197		
			R=300Kohm	F=366KHz		0.104		
		C=20P	R=1Kohm	F=5.99MHz		2.474		
			R=3.3Kohm	F=4.51MHz		1.211		
			R=10Kohm	F=2.49MHz		0.565		
			R=100Kohm	F=380KHz		0.119		
			R=300Kohm	F=133KHz		0.073		
		C=100P	R=1Kohm	F=3.88MHz		2.33		
			R=3.3Kohm	F=2.07MHz		0.872		
			R=10Kohm	F=890KHz		0.358		
			R=100Kohm	F=107KHz		0.082		
			R=300Kohm	F=36KHz		0.059		

Sym	Description	Conditions			Min.	Typ.	Max.	Unit
I_{DD}	Operating current	C=300P	R=1Kohm	F=2.25MHz		2.146		mA
			R=3.3Kohm	F=998KHz		0.736		
			R=10Kohm	F=384KHz		0.291		
			R=100Kohm	F=42KHz		0.073		
			R=300Kohm	F=14KHz		0.056		
I_{DD}	Operating current	ERIC mode, external R, Vdd=5V, 4 clock instruction						mA
		F=8MHz	R=88Kohm			1.197		
		F=4MHz	R=179.9Kohm			0.644		
		F=2MHz	R=349Kohm			0.377		
		F=1MHz	R=650.8Kohm			0.24		
		F=455KHz	R=1222Kohm			0.164		
		ERIC mode, external R, Vdd=3V, 4 clock instruction						
		F=8MHz	R=83.5Kohm			0.77		
		F=4MHz	R=166.2Kohm			0.425		
		F=2MHz	R=315.8Kohm			0.242		
		F=1MHz	R=565.8Kohm			0.147		
		F=455KHz	R=992.8Kohm			0.094		
		ERIC mode, external R, Vdd=5V, 2 clock instruction						
		F=8MHz	R=88.3Kohm			1.237		
		F=4MHz	R=197.2Kohm			0.896		
		F=2MHz	R=349.3Kohm			0.531		
		F=1MHz	R=650.4Kohm			0.316		
		F=455KHz	R=1224Kohm			0.198		
		ERIC mode, external R, Vdd=3V, 2 clock instruction						
		F=8MHz	R=81.1Kohm			1.237		
		F=4MHz	R=167.7Kohm			0.641		
		F=2MHz	R=315.7Kohm			0.354		
		F=1MHz	R=585.2Kohm			0.203		
		F=455KHz	R=991.3Kohm			0.12		
I_{DD}	Operating current	IRC mode, internal R, Vdd=5V, 4 clock instruction						mA
		F=8MHz				1.122		
		F=31KHz*				0.089		
		IRC mode, internal R, Vdd=3V, 4 clock instruction						
		F=8MHz				0.811		
		F=31KHz*				0.049		
		IRC mode, internal R,Vdd=5V, 2 clock instruction						
		F=8MHz				1.764		
		F=31KHz*				0.091		
		IRC mode, internal R,Vdd=3V, 2 clock instruction						
		F=8MHz				1.276		
		F=31KHz*				0.051		

Note: IRC 31KHz no calibration mechanism, current value is for reference only.

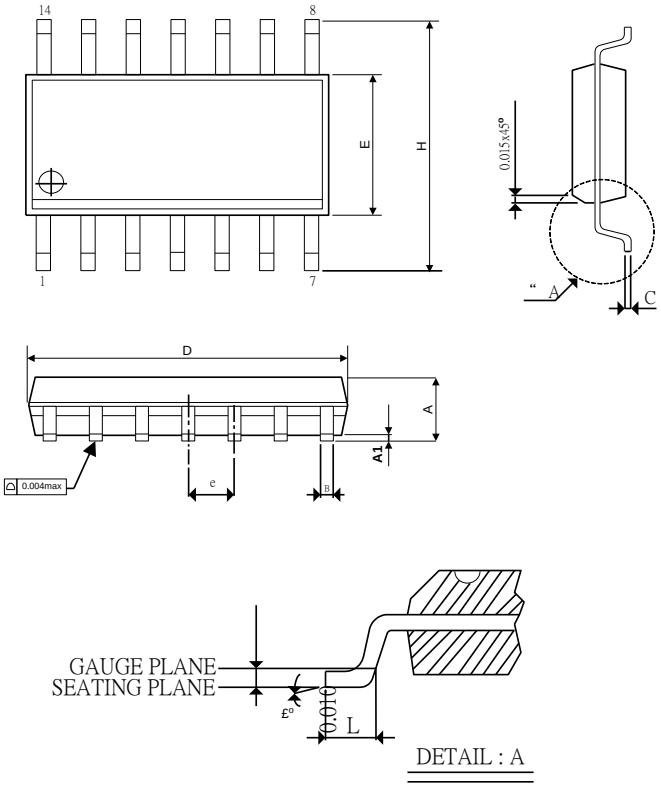
7.0 PACKAGE DIMENSION

7.1 14-PIN PDIP 300mil



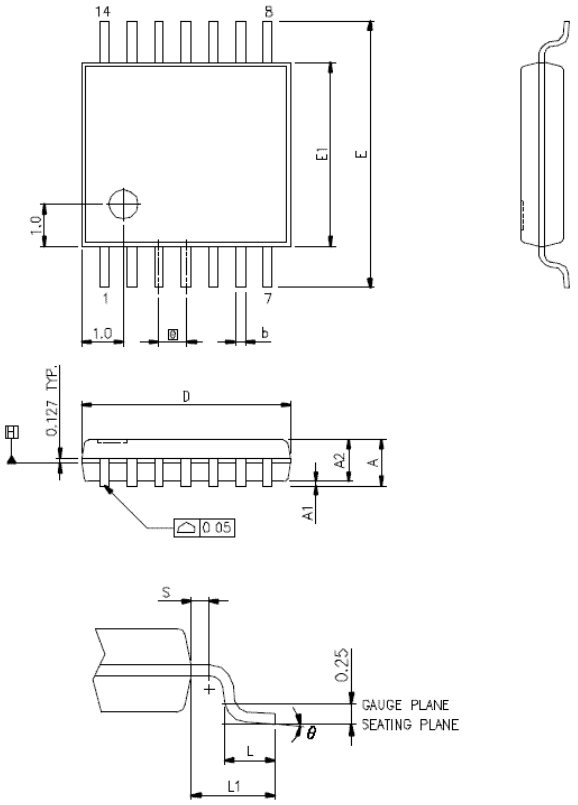
Symbols	Dimension In Inches		
	Min	Nom	Max
A	-	-	0.210
A1	0.015	-	-
A2	0.125	0.130	0.135
D	0.735	0.750	0.775
E		0.300 BSC.	
E1	0.245	0.250	0.255
L	0.115	0.130	0.150
eB	0.335	0.355	0.375
θ°	0°	7°	15°

7.2 14-PIN SOP 150mil



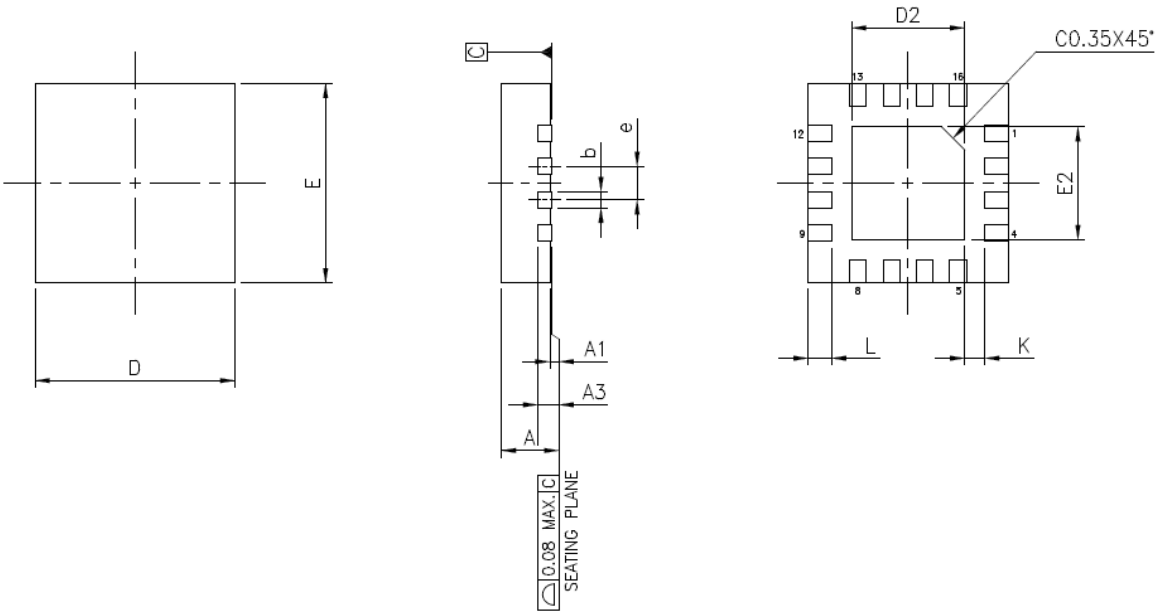
Symbols	Dimension In Inches		
	Min	Nom	Max
A	0.058	0.064	0.068
A1	0.004	-	0.010
B	0.013	0.016	0.020
C	0.0075	0.008	0.0098
D	0.336	0.341	0.344
E	0.150	0.154	0.157
e	-	0.050	-
H	0.228	0.236	0.244
L	0.015	0.025	0.050
θ°	0°	-	8°

7.3 14-PIN TSSOP 4.4mm



Symbols	Dimension In Millimeters		
	Min	Nom	Max
A	-	-	1.20
A1	0.05	-	0.15
A2	0.80	1.00	1.05
b	0.19	-	0.30
D	4.90	5.00	5.10
E1	4.30	4.40	4.50
E	6.40 BSC		
e	0.65 BSC		
L1	1.00 REF		
L	0.45	0.60	0.75
S	0.20	-	-
θ°	0°	-	8°

7.4 16-PIN QFN



Symbols	Dimension In Millimeters		
	Min	Nom	Max
A	0.70 0.80	0.75 0.85	0.80 0.90
A1	0.00	0.02	0.005
A3		0.20 REF	
b	0.018	0.025	0.030
D		3.00 BSC	
E		3.00 BSC	
e		0.50 BSC	
L	0.30	0.35	0.40
K	0.20	-	-
E2	1.60	1.70	1.75
D2	1.60	1.70	1.75

8.0 ORDERING INFORMATION

Flash Type MCU	Package Type	Pin Count	Package Size	Note
FM8PE32AFP	PDIP	14	300 mil	HV package
FM8PE32AFD	SOP	14	150 mil	HV package
FM8PE32AFW	TSSOP	14	173 mil	HV package
FM8PE32AFe	QFN	16	(3X3)	HV package
FM8PE32BFP	PDIP	14	300 mil	LV package
FM8PE32BFD	SOP	14	150 mil	LV package
FM8PE32BFW	TSSOP	14	173 mil	LV package
FM8PE32BFe	QFN	16	(3X3)	LV package