

EET250 Lab PIC TIMER0

This lab demonstrates the use of PIC TIMER0 peripheral used in both Interrupt and non-interrupt modes to help display a counting values to LEDs. The Timer essentially functions in hardware as a delay. In both cases the timer is initialized and an overflow condition is monitored and used as a trigger to increment a software counter. Timer0 is automatically interrupts (if interrupts are enabled) at overflow condition.

OPTION REG:

The Timer0 module timer/counter has the following features:

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

Figure 1 Timer0 features

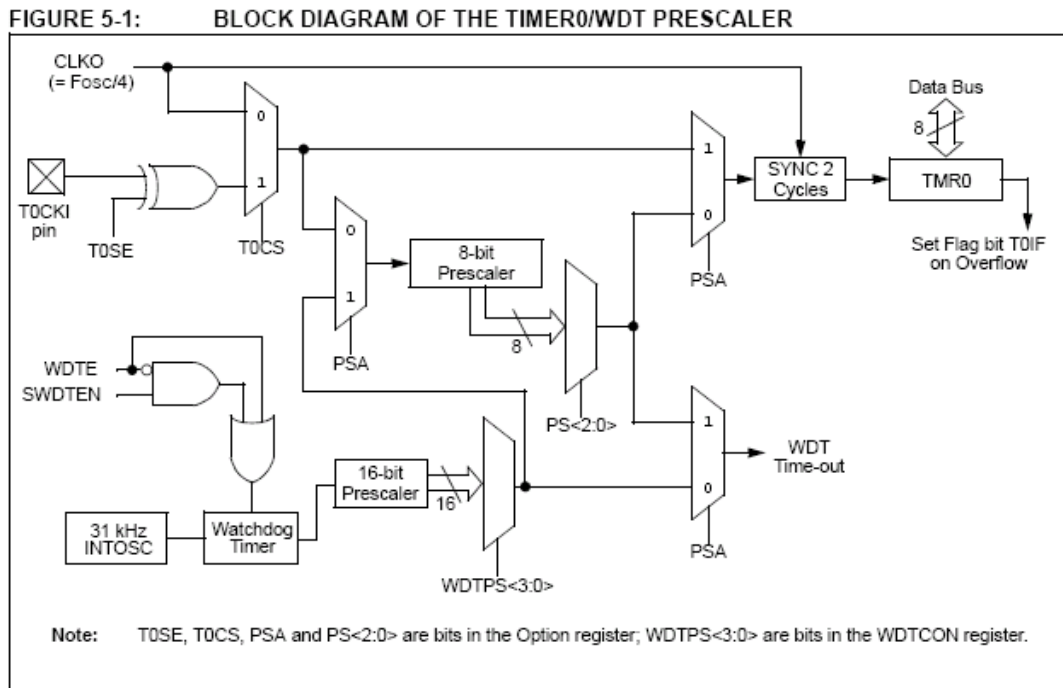


Figure 2 Timer0 Block Diagram

REGISTER 5-1: OPTION_REG – OPTION REGISTER (ADDRESS: 81h OR 181h)

	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	
	RBP \bar{U}	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
							bit 7	bit 0
bit 7	RBP\bar{U}: PORTB Pull-up Enable bit 1 = PORTB pull-ups are disabled 0 = PORTB pull-ups are enabled by individual port latch values in WPUA register							
bit 6	INTEDG: Interrupt Edge Select bit 1 = Interrupt on rising edge of RB0/INT/SEG0 pin 0 = Interrupt on falling edge of RB0/INT/SEG0 pin							
bit 5	T0CS: TMR0 Clock Source Select bit 1 = Transition on RA4/C1OUT/T0CKI/SEG4 pin 0 = Internal instruction cycle clock (CLKO)							
bit 4	T0SE: TMR0 Source Edge Select bit 1 = Increment on high-to-low transition on RA4/C1OUT/T0CKI/SEG4 pin 0 = Increment on low-to-high transition on RA4/C1OUT/T0CKI/SEG4 pin							
bit 3	PSA: Prescaler Assignment bit 1 = Prescaler is assigned to the WDT 0 = Prescaler is assigned to the Timer0 module							
bit 2-0	PS<2:0>: Prescaler Rate Select bits							
	Bit Value	TMR0 Rate	WDT Rate ⁽¹⁾					
	000	1 : 2	1 : 1					
	001	1 : 4	1 : 2					
	010	1 : 8	1 : 4					
	011	1 : 16	1 : 8					
	100	1 : 32	1 : 16					
	101	1 : 64	1 : 32					
	110	1 : 128	1 : 64					
	111	1 : 256	1 : 128					

Note 1: A dedicated 16-bit WDT postscaler is available for the PIC16F917/916/914/913. See Section 16.6 “Watchdog Timer (WDT)” for more information.

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

Figure 3 OPTION_REG

2.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, PORTB change and external RB0/INT/SEG0 pin interrupts.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON – INTERRUPT CONTROL REGISTER (ADDRESS: 0Bh, 8Bh, 10Bh OR 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	TOIE	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

- bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts
 - bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts
 - bit 5 **TOIE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt
 - bit 4 **INTE:** RB0/INT/SEG0 External Interrupt Enable bit
1 = Enables the RB0/INT/SEG0 external interrupt
0 = Disables the RB0/INT/SEG0 external interrupt
 - bit 3 **RBIE:** PORTB Change Interrupt Enable bit⁽¹⁾
1 = Enables the PORTB change interrupt
0 = Disables the PORTB change interrupt
 - bit 2 **T0IF:** TMR0 Overflow Interrupt Flag bit⁽²⁾
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow
 - bit 1 **INTF:** RB0/INT/SEG0 External Interrupt Flag bit
1 = The RB0/INT/SEG0 external interrupt occurred (must be cleared in software)
0 = The RB0/INT/SEG0 external interrupt did not occur
 - bit 0 **RBIF:** PORTB Change Interrupt Flag bit
1 = When at least one of the PORTB <5:0> pins changed state (must be cleared in software)
0 = None of the PORTB <7:4> pins have changed state
- Note 1:** IOCB register must also be enabled.
- 2:** T0IF bit is set when Timer0 rolls over. Timer0 is unchanged on Reset and should be initialized before clearing T0IF bit.

Legend:

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

Figure 4 INTCON

1.0 Program #1: Non-Interrupt Code

Use supplied program, enter and build program and program and debug using PICKIT2. Flowchart and demo to instructor. Review all control settings. Why is GIE bit in INTCON need to be set?

```
#include <pl6F917.inc>
    __config (_INTRC_OSC_NOCLKOUT & _WDT_OFF & _PWRTE_OFF & _MCLRE_OFF & _CP_OFF &

; Register Usage
CBLOCK 0x020                ; Start Registers at End of the Values
counter
ENDC

; Mainline of TMRO
org 0
    movlw 0
    movwf counter
    bsf STATUS, RPO
; clrf TRISD                ; Make PortD all output
bcf INTCON,TOIE
bcf INTCON,TOIF
|
    movlw 0xd7              ; configure Timer0. Sourced from the Processor clock;
    movwf OPTION_REG        ; Maximum Prescaler
    clrf TRISD              ; Make PortD all output
    bcf STATUS, RPO

Loop                ; Loop Here

;movf TMRO, w            ; Output the TMRO Value
btss INTCON,TOIF
goto Loop
    incf counter
movf counter,w
    movwf PORTD
bcf INTCON,TOIF
goto Loop
```

2.0 Program #2 Interrupt Driven code

Use supplied program, enter and build program and program and debug using PICKIT2. Flowchart and demo to instructor. Where is the Interrupt service routine located on the PIC?

```

; Mainline of TMRO
org 0
goto main
org 4
Int ; TMRO has Overflowed - New Input

bcf INTCON, TOIF ; Reset the Interrupt Flag
incf counter
movf counter, w
movwf PORTD

retfie
main movlw 0
movwf counter
bsf STATUS, RPO

bcf INTCON, TOIF
movlw 0xd7 ; configure Timer0. Sourced from the Processor clock;
movwf OPTION_REG ; Maximum Prescaler
clrf TRISD ; Make PortD all output
bsf INTCON, TOIE
bsf INTCON, GIE
bcf STATUS, RPO

Loop ; Loop Here

goto Loop

end

```