



ARQUITECTURA DE COMPUTADORAS ITM7

Actividad 1 Unidad 4

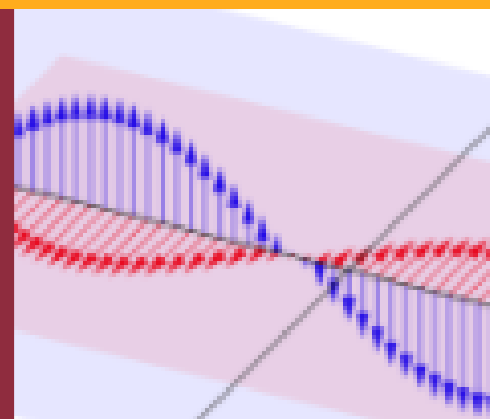
PROFESOR

Sergio Garduza

27 OCTUBRE DE 2021

ALUMNO

Itzel Berenice Cabrera Vazquez
2020640576



UNIDAD 4 ACTIVIDAD 1

Itzel Berenice Cabrera Vazquez

October 28, 2021

1 Introducción

El módulo Timer0 (TMR0) es un temporizador-contador de 8 bits, es decir, de 0 a 255. Se dice que es contador cuando la fuente de reloj es externa, para ello se usa el pin T0CKI y se dice que es temporizador cuando la fuente de reloj es interna, viene dada por el mismo reloj del microcontrolador. Por otra parte, puede ser preescalado, ya sea mediante un "watchdog" o el mismo módulo "TMR0". Ver fig.1 y fig. 2.

BIT VALUE	TMR0 RATE	WDT RATE
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Figure 1: Prescaler rate.

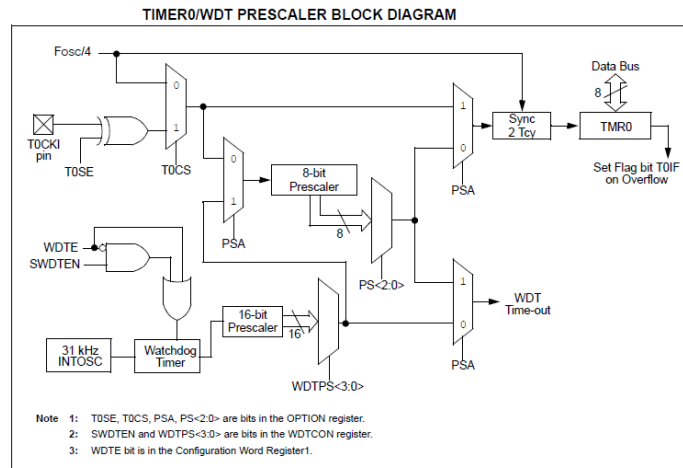


Figure 2: Prescaler diagram.

Cabe mencionar que las características del temporizador-contador se configuran mediante el registro OPTION REG. Ver fig.3

Por otro lado, para el uso de puertos se requiere especificar si son pines digitales o analógicos (esto mediante registros como ANSEL o ANSELH); además, se debe especificar si los pines son entradas o salidas (mediante el uso de registros como TRISA, TRISB, etc).

Finalmente, también hay que recordar el uso del comando RETLW para poder retornar valores de una "tabla", este funciona mediante el uso del registro PCL.

bit 7	RBP U : PORTB Pull-up Enable bit 1 = PORTB pull-ups are disabled 0 = PORTB pull-ups are enabled by individual PORT latch values
bit 6	INTEDG : Interrupt Edge Select bit 1 = Interrupt on rising edge of INT pin 0 = Interrupt on falling edge of INT pin
bit 5	T0CS : Timer0 Clock Source Select bit 1 = Transition on T0CKI pin 0 = Internal instruction cycle clock ($F_{osc}/4$)
bit 4	T0SE : Timer0 Source Edge Select bit 1 = Increment on high-to-low transition on T0CKI pin 0 = Increment on low-to-high transition on T0CKI 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

Figure 3: Bits de $OPTION_{REG}$.

2 Desarrollo y Resultados

2.1 Configuración de la máquina

Inicialmente se declaran los pines de los puertos A y B como digitales. Además, para el puerto A, los pines 0,1,2 se usan para ajustar el preescalador y el pin 4 se usa para la fuente de reloj externa; para el puerto B, todos son salidas, estas salidas irán conectadas a un display. Finalmente, se limpia el puerto A, el puerto B y el contador, esto para limpiar el buffer de posible "basura". Ver fig4.

```

;CONFIGURACION DE LA MAQUINA
BANKSEL    ANSEL
CLRF       ANSEL      ;PINES DIGITALES
CLRF       ANSELH     ;PINES DIGITALES
;PUERTO A: USA COMO SELECTOR DEL PREESCALADOR
;PUERTO B: SE USA PARA VER LO QUE ESTÉ EN TMR0
BANKSEL    TRISA
MOVLW     B'00010111' ;ENTRADAS => BIT<0:2> PREESCALADOR; BIT<4> PARA EL PIN T0CKI
MOVWF     TRISA
CLRF      TRISB       ;TODOS LOS PINES SON SALIDA
BANKSEL    PORTA
;LIMPIA EL BUFFER DE TMR0, PORTA Y PORTB
CLRF      TMR0
CLRF      PORTA
CLRF      PORTB

```

Figure 4: Aplicacion.

2.2 Aplicación

Primeramente, se lee en WREG lo que hay en el puerto A, ya que este nos indica el preescalador, del bit 0 al 2. Despues, se realiza un OR como máscara para poder actualizar los bits del preescalador. Luego, se extrae la información de TMR0 para que se lea en el puerto B, es decir, para que aparezca la información en el display. Posteriormente, se realiza un XOR con la información del TMR0, si la información en TMR0 es un '9', entonces el bit Z del registro Status es '1' (ya que $A \text{ XOR } A = 0$). Finalmente, si se llega al dígito '9', entonces se resetea el contador. Ver fig5.

2.3 Decodificación a 7 segmentos

Se usa el mecanismo de una tabla y del comando "RETLW" para obtener el código a usar en el display de 7 segmentos. Ver fig5.

ciclo:	NOP	
	BANKSEL	PORTA
	MOVWF	PORTA
	BANKSEL	OPTION_REG
	IORWF	OPTION_REG,1;APLICA LA MÁSCARA PARA ACTUALIZAR EL SELECTOR DE PREESCALADOR
	BANKSEL	PORTA
	MOVWF	TMR0
	CALL	DEC ;DECODIFICADOR A 7 SEGMENTOS
	MOVWF	PORTB ;SE CONFIGURA EL DISPLAY
	MOVWF	TMR0
	XORLW	B'01111011' ; (TMR0 XOR TMR0) = 0
	BTFSC	STATUS,2 ;SI EL BIT 2 DE STATUS ES 1 (LLEGO A 9), EJECUTA LA SIG LINEA
	CLRF	TMR0 ;VUELVE A EMPEZAR EL CONTEO
	GOTO	ciclo

Figure 5: Aplicación

DEC:	NOP	
		ADDWF
		PCL, 1
		RETLW
		H'3F'
		RETLW
		H'06'
		RETLW
		H'5B'
		RETLW
		H'4F'
		RETLW
		H'66'
		RETLW
		H'6D'
		RETLW
		H'7D'
		RETLW
		H'07'
		RETLW
		H'7F'
		RETLW
		H'67'

Figure 6: Decodificador a 7 segmentos

2.4 Circuito en Proteus

Se usó Proteus para simular el algoritmo hecho en ensamblador.

Se tienen 5 push button, 3 para modificar el prescalador, 1 para modificar el pin T0CK1 y otro para el RESET. Finalmente, el uso de display nos permite visualizar como el prescalador funciona. Ver fig7.

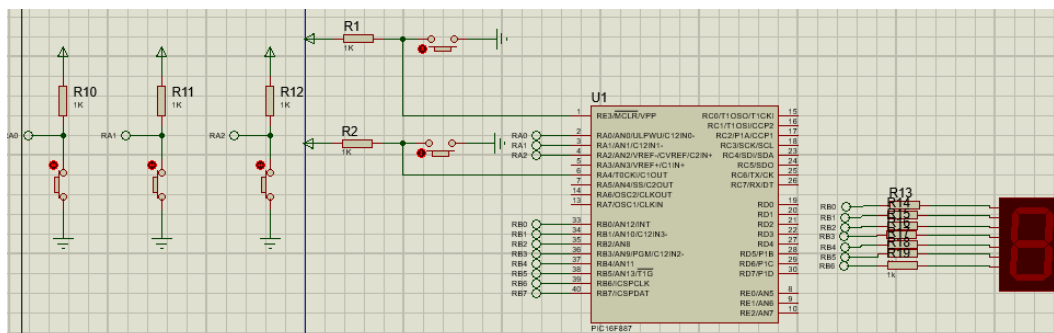


Figure 7: Circuito

3 Conclusiones

Con esta práctica pude estudiar el uso de puertos y del Timer0 en el PIC16F887. Además me ayudó a aprender a usar Proteus, nunca lo había usado y me resulta bastante útil para poder visualizar los resultados de los programas hechos en MPLAB.

Donde se me complicó fue en la parte de leer el puerto B, me confundí un poco con PORTB y TMRO, pero finalmente entendí que uno lee la información del otro.