



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Departamento de Eléctrica y Electrónica

Carrera de Electrónica y Automatización

SISTEMAS BASADOS EN MCU

Práctica 2.4

**MANEJO DE PUERTOS COMPLETOS –
CONTADORES**

Autor:

Iza Tipanluisa Alex Paul

Docente:

Ing. Amparo Meythaler

NRC: 4891

1) OBJETIVOS

- Realizar ejercicios de contadores utilizando los Puertos completos del PIC16F877.
- Verificar el funcionamiento y la importancia de los Retardos para la utilización de los puertos.

2) MARCO TEORICO

CONTADORES

A menudo se requiere realizar contaje, este puede ser de subida o bajada, desde y hasta cualquier valor.

Pueden los contadores volver a empezar (reciclables) o terminar en un dato (parada propia). Cualquier contaje que se

escoja, únicamente se realiza con lazos.

RETARDOS

A menudo es necesario que los programas usen demoras o retardos.

Los retardos por Lazos consisten en que el procesador se quede ejecutando unos bucles que van decrementando o

incrementando unos contadores de acuerdo al tiempo requerido; una vez terminada esta acción, el programa sigue

ejecutando otros procesos.

Se aconseja realizar los retardos mediante subrutinas.

El tiempo del retardo se calcula contabilizando los ciclos de máquinas de las instrucciones empleadas, el valor del ciclo

de máquina (dato del integrado utilizado) y la frecuencia de trabajo (oscilador).

3) EQUIPOS Y MATERIALES

- PC con el paquete MPLAB IDE.
- PC con el paquete PROTEUS.

4) ACTIVIDADES

1) Trabajo Preparatorio:

a) Realizar el diagrama de flujo, la codificación e implementación correspondiente, de un programa que reciba un dato a través de 8 interruptores y realice un contador desde ese valor hasta FF o 00 (subida o bajada escoja Ud. al igual que los pasos de conteo que debe indicar al realizar el ejercicio). El contaje se visualiza en 8 leds. Entre cada dato de análisis debe estar un retardo de 0,4 milisegundos, calculados en base a un oscilador de 4 MHz.

2) Verifique el funcionamiento del MPLAB IDE y del PROTEUS.

3) Cálculo de Retardo

Realice los cálculos para obtener el tiempo solicitado.

$$T = \#CM(\text{que no se repite}) + \#CM(\text{que se repite})$$

$$T = (6 + 6 * 255) * \frac{4}{4 * 10^6}$$

$$T_{max} = 1.536ms$$

Por lo tanto, si me alcanza y no debo realizar otro lazo anidado de retardo

$$0.4ms = (6 + 6 * V_{dec}) * \frac{4}{4 * 10^6}$$

$$400 = 6 + 6 * V_{dec}$$

$$V_{dec} = 65.666_{10} \cong 66_{10}$$

$$V_{hex} = 42_{16}$$

4) Trabajo en el paquete MPLAB IDE.

Digite el programa

Compile el ejercicio hasta que obtenga 0 errores.

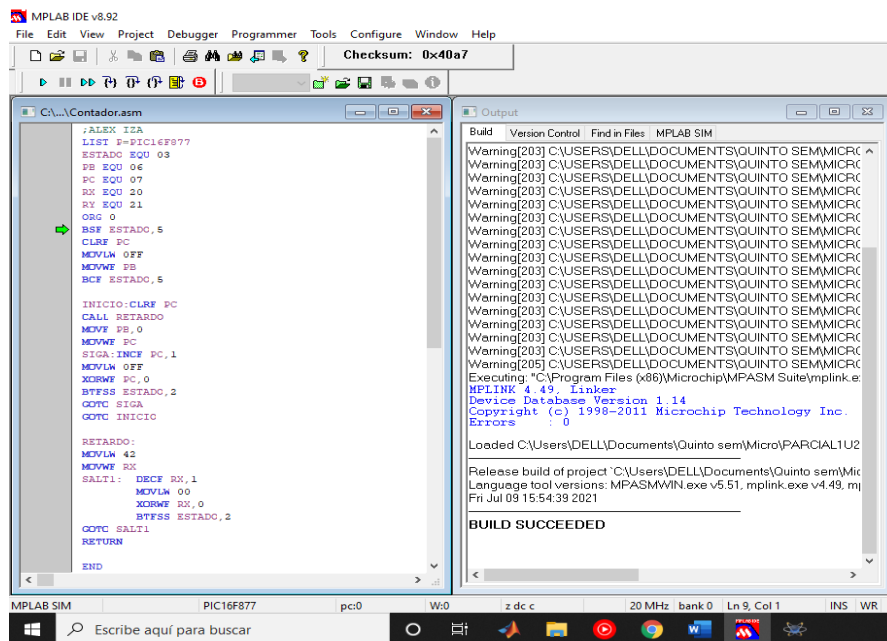


Ilustración 1

5) Trabajo en el paquete PROTEUS.

Realice el diagrama esquemático.

Cargue el programa compilado en el microcontrolador.

Corra el diseño y verifique el funcionamiento.

Ilustración 2

5) RESULTADOS

- 1) Explique los errores cometidos en el ejercicio realizado (si los tuvo) y la forma de corregirlos.

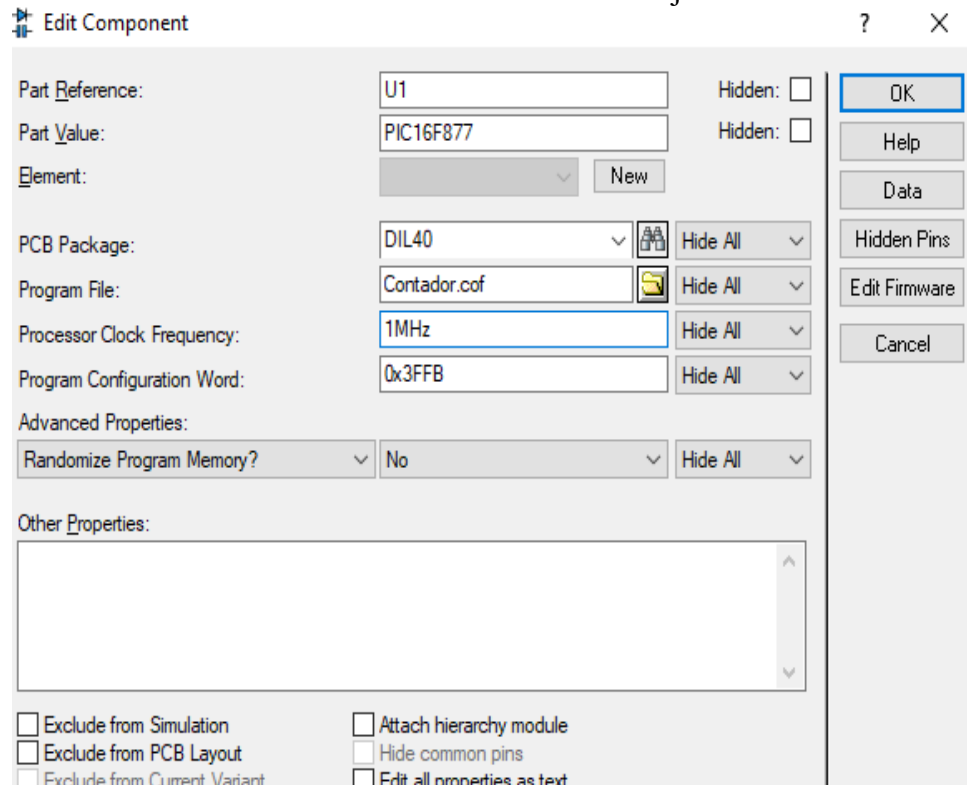
El único error que tuve fue el del retardo ya que estaba haciendo un retardo anidado y este no era necesario.

- 2) Explique, qué se debe hacer si se requiere un tiempo de 20 milisegundos entre datos

Ya que el tiempo máximo es $T_{max} = 1.536ms$ entonces este debe tener un retardo anidado ya que se necesitará un lazo externo para alcanzar un retardo de 20ms

- 3) Explique cómo simularía en el paquete PROTEUS a otra velocidad de trabajo sin cambiar el programa.

Cambiando la frecuencia del reloj del Pic16f877



6) DISEÑO

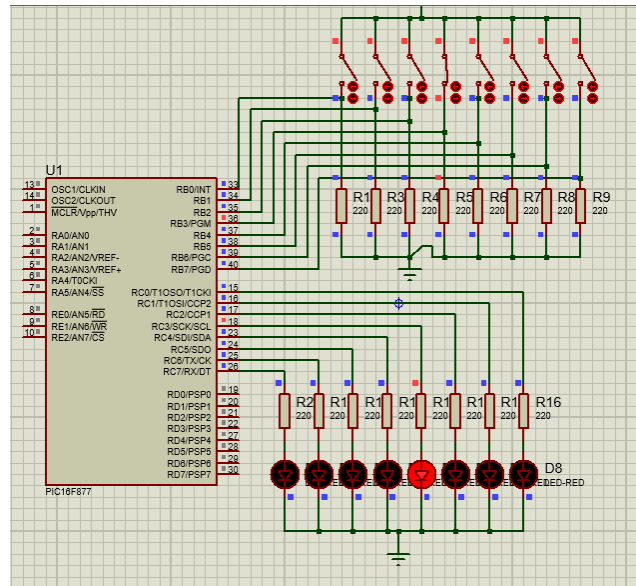


Ilustración 3

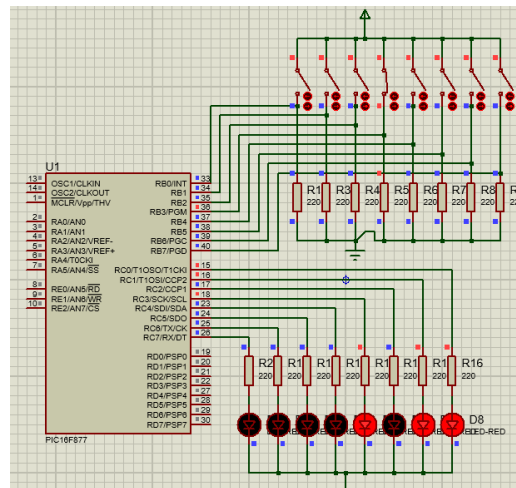


Ilustración 4

7) DIAGRAMA DE FLUJO

- Realizar el diagrama de flujo, la codificación e implementación correspondiente, de un programa que prenda y apague un led mediante la acción de cualquiera de los 2 interruptores colocados en el Microcontrolador PIC16F877.

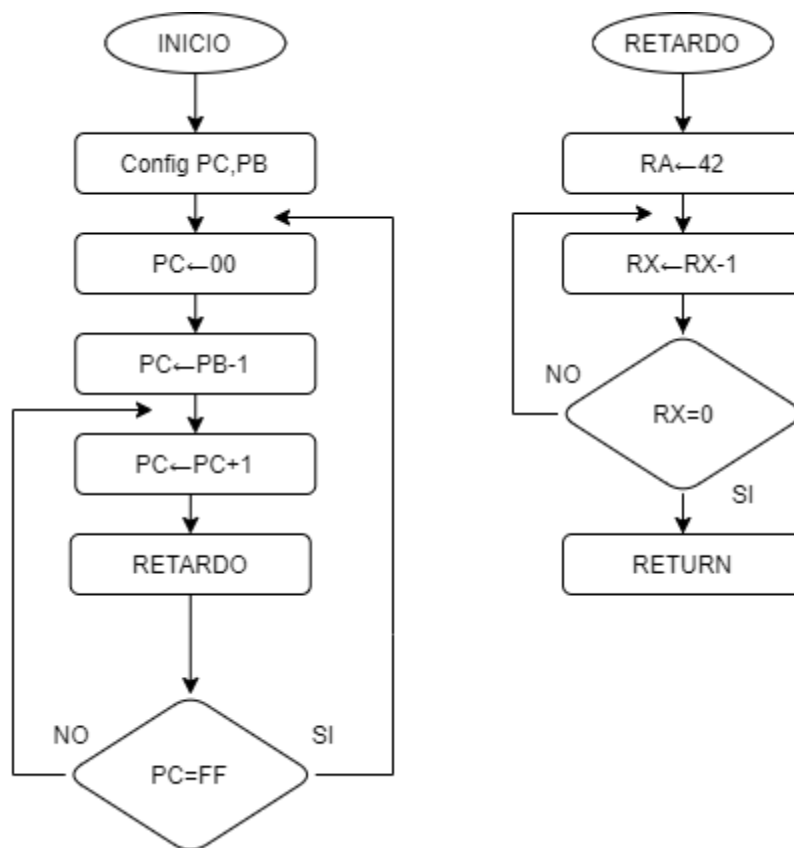


Diagrama 1

8) PROGRAMA

- a) Realizar el diagrama de flujo, la codificación e implementación correspondiente, de un programa que reciba un dato a través de 8 interruptores y realice un contador desde ese valor hasta FF o 00 (subida o bajada escoja Ud. al igual que los pasos de conteo que debe indicar al realizar el ejercicio). El conteo se visualiza en 8 leds. Entre cada dato de análisis debe estar un retardo de 0,4 milisegundos, calculados en base a un oscilador de 4 MHz.

```

;ALEX IZA
LIST P=PIC16F877
ESTADO EQU 03
PB EQU 06
PC EQU 07
RX EQU 20
RY EQU 21
ORG 0
BSF ESTADO,5
CLRF PC
MOVLW 0FF
MOVWF PB
BCF ESTADO,5

```

```

INICIO:CLRF PC
DECF PB,0
MOVWF PC

```

```
SIGA:INCF PC,1  
CALL RETARDO  
MOVLW 0FF  
XORWF PC,0  
BTFSS ESTADO,2  
GOTO SIGA  
GOTO INICIO
```

```
RETARDO:  
MOVLW 42  
MOVWF RX  
SALT1:  DECF RX,1  
        MOVLW 00  
        XORWF RX,0  
        BTFSS ESTADO,2  
GOTO SALT1  
RETURN
```

END

9) CONCLUSIONES

- Se ha concluido que la función del programa retardo es una ayuda para observar los cambios generados en la salida
- Se ha concluido que el tiempo de retardo se puede calcular mediante la formula

$$tret = \#CM * ValorCM$$

- Se ha concluido que el valor encontrado al aplicar la formula del retardo, este nos entrega un resultado en decimal, el mismo que para el programa se debe convertir a hexadecimal.

10) RECOMENDACIONES

- Se recomienda analizar correctamente las posibles respuestas que pueden darse en los resultados, para que se puedan solucionarse durante la digitación del programa.
- Se recomienda analizar correctamente el proceso que debe realizar el programa porque de lo contrario terminara realizando un proceso diferente al requerido.
- Se recomienda usar lazos cuando el programa requiera que el retardo se repita n veces, todo esto logrando que ocupe menos memoria y que el programa sea más rápido

11) BIBLIOGRAFIA

Meythaler, A. (2021). Sistemas basados en MCU. Ecuador: UFA ESPE