



# 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 3.1**

**INTERRUPCIONES DEL PIC16F877**

**Autor:**

**Iza Tipanluisa Alex Paul**

**Docente:**

**Ing. Amparo Meythaler**

**NRC: 4891**

## **1) OBJETIVOS**

- Identificar la forma de programar las interrupciones en el PIC16F877.
- Identificar el funcionamiento de los registros que controlan las interrupciones del PIC16F877.

## **2) MARCO TEORICO**

### **INTERRUPCIONES**

Una interrupción es una situación especial que suspende la ejecución de un programa de modo que la CPU pueda realizar otra acción por un instante, antes de continuar con la ejecución normal del programa. Tal situación se da, por ejemplo, cuando un periférico requiere la atención del procesador para realizar una operación de E/S.

La familia Pic16F877 tiene 14 fuentes de interrupción. Al aceptarse una interrupción se salva el valor del PC contador de programa en la pila y se carga aquel con el valor 0004h, que es el Vector de Interrupción. La mayoría de los recursos adicionales son capaces de ocasionar una interrupción.

Para habilitar el uso de las interrupciones con el PIC se utiliza el registro INTCON, cada microcontrolador PIC tiene su registro INTCON, algunos tendrán más de uno, es necesario revisar su hoja de datos para verificar para que se utilizan cada uno de sus bits, pero lo que sí es común en ellos es el bit7 o bit GIE.

## **3) OBJETIVOS**

- Identificar el trabajo de los timers del PIC16F877 en la función temporizador.
- Identificar el trabajo de los timers del PIC16F877 en la función contador.

## **4) EQUIPOS Y MATERIALES**

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

## **5) ACTIVIDADES**

### **1) Trabajo Preparatorio:**

a) Realice el diagrama de flujo, la codificación y la implementación correspondientes de un programa que maneje las tres luces de cada uno de los dos semáforos conectados en una esquina, estos semáforos controlan la circulación de vehículos que pueden ir en una sola vía (Norte a Sur) y (Este a Oeste)

Las luces cumplen los siguientes tiempos:

Tiempo verde = 0.8 seg

Tiempo amarillo = 0,2 seg

En la esquina también hay un display que indica la S (Norte a Sur) o la E (Este a Oeste), según el sentido de circulación de los vehículos. Eso se verifica (o tiene relación) según las luces de los semáforos únicamente.

Si un peatón debe pasar por emergencia, acciona un pulsante que está colocado en la esquina y debe ser atendido en ese instante. En éste caso los dos carriles detienen la circulación de los vehículos. Este proceso inicia prendiendo los amarillos y luego los rojos que titilan (o parpadean) tres veces; el display presentará la P de Peatón.

Todos los tiempos deberán ser regulados en base al Temporizador 1 trabajando a la frecuencia de 2 MHz. NO requiere usar interrupción.

#### 1) Trabajo en el paquete MPLAB IDE.

- Digite el programa.
- Compile el ejercicio hasta que obtenga 0 errores.

#### 2) Trabajo en el paquete PROTEUS.

- Realice el diagrama esquemático.
- Cargue el programa compilado en el microcontrolador.
- Corra el diseño y verifique el funcionamiento.

#### 3) Presente el cálculo de los tiempos.

$$tr = (\overline{DI} + 1) CM \cdot DF$$

$$0.2 = (\overline{DI} + 1) \left( \frac{4}{2 \times 10^6} \right) (4)$$

$$25000 = (\overline{DI} + 1)$$

$$\overline{DI} = 24999(10)$$

$$\overline{DI} = 61A7$$

$$\#lazos = \frac{quiero}{tengo}$$

$$\#lazos = \frac{0.8}{0.2}$$

$$\#lazos = 4 \text{ lazos}$$

### 6) RESULTADOS

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

Tuve problemas con el análisis del guardado y cambio de los leds de los semáforos, lo solucioné guardando los valores en localidades los mismos que se iban a mostrar en los semáforos.

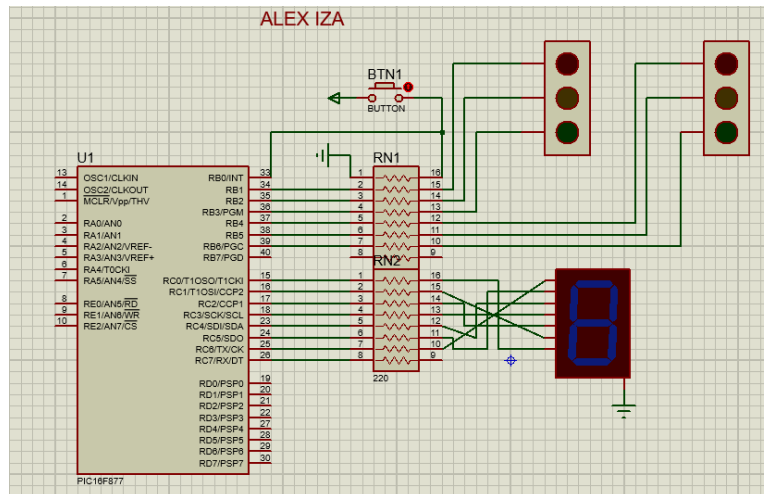
- En qué localidad de memoria debe estar el código de una subrutina de interrupción.

En la localidad 04

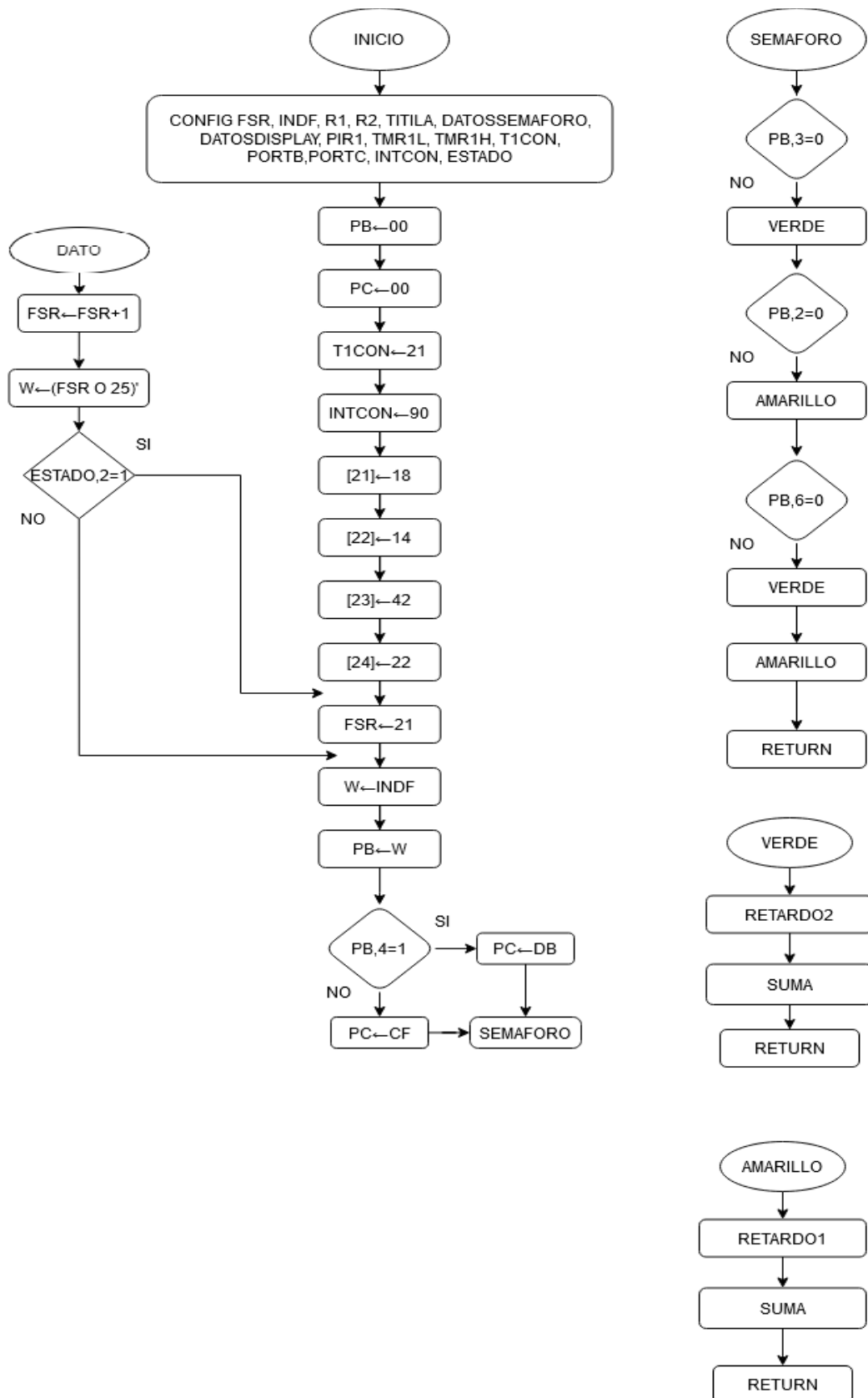
- ¿Para qué sirve la instrucción RETFIE?

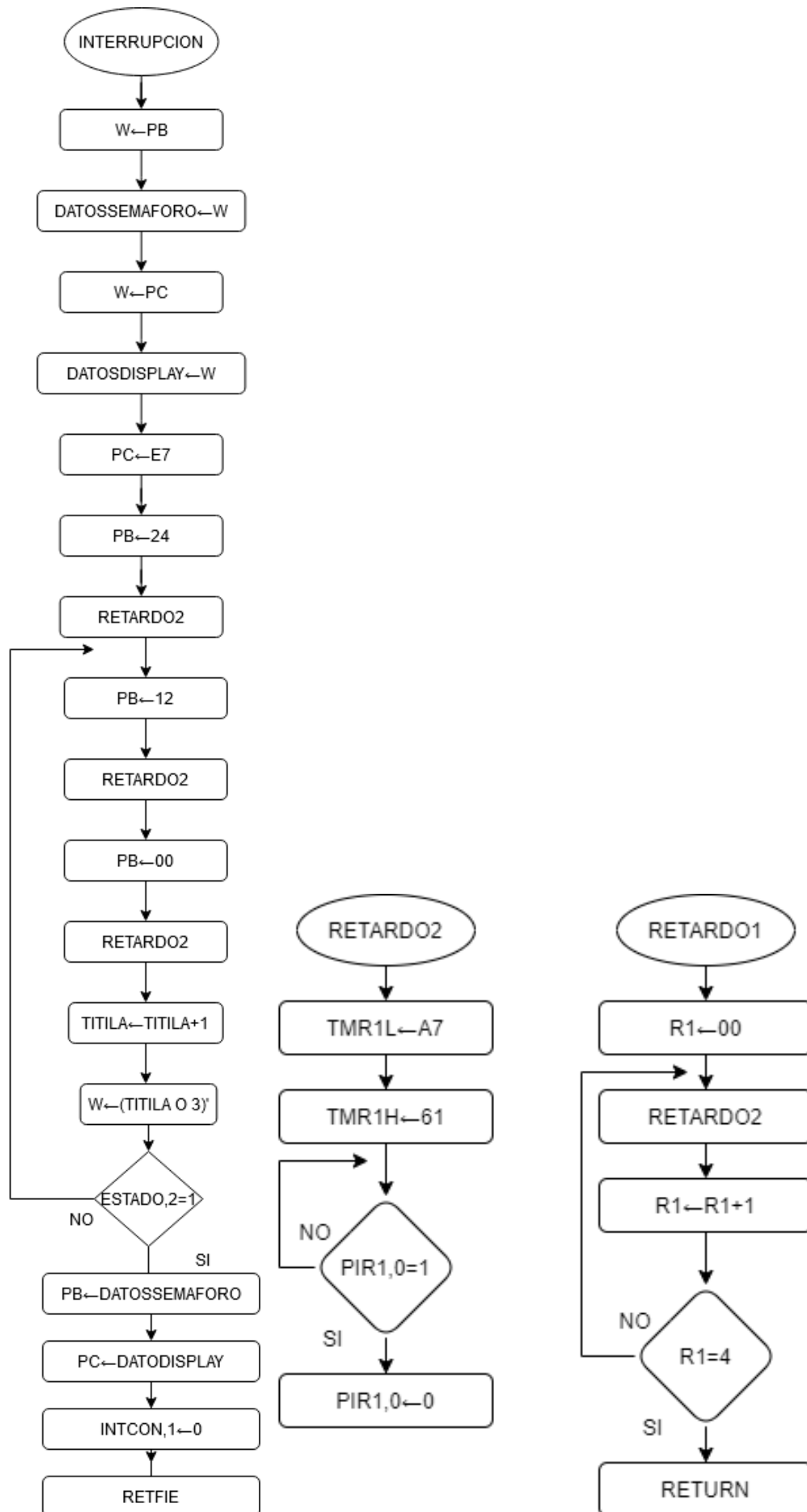
La instrucción RETFIE nos permite regresar de una interrupción habilitando el GIE

## 7) DISEÑO



## 8) DIAGRAMA DE FLUJO





## 9) PROGRAMA

;ALEX IZA

LIST P=PIC16F877

FSR EQU 04

ESTADO EQU 03

PB EQU 06

PC EQU 07

INDF EQU 00

TITILA EQU 20

R1 EQU 25

R2 EQU 26

DATOSSEMAFORO EQU 27

DATODISPLAY EQU 28

PIR1 EQU 0C

TMR1L EQU 0E

TMR1H EQU 0F

T1CON EQU 10

INTCON EQU 0B

ORG 0

GOTO INICIO

ORG 4

GOTO INTERRUPCION

INICIO:

BSF ESTADO,5

MOVLW 01

MOVWF PB

CLRF PC  
BCF ESTADO,5  
CLRF PB  
CLRF PC  
MOVLW 21  
MOVWF T1CON  
MOVLW 90  
MOVWF INTCON

MOVLW 18  
MOVWF 21  
MOVLW 14  
MOVWF 22  
MOVLW 42  
MOVWF 23  
MOVLW 22  
MOVWF 24

LEEDATOS:  
MOVLW 21  
MOVWF FSR  
IMPRIMEDATOS:  
MOVF INDF,0  
MOVWF PB  
BTFSS PB,4  
GOTO E  
GOTO S

S: MOVLW 0DB



```
MOVWF PC
CALL SEMAFORO
E: MOVLW 0CF
MOVWF PC
CALL SEMAFORO
```

```
VERDE:
CALL RETARDO2
CALL DATOS
RETURN
```

```
AMARILLO:
CALL RETARDO1
CALL DATOS
RETURN
```

```
DATOS:
INCF FSR,1
MOVLW 25
XORWF FSR,0
BTFSS ESTADO,2
GOTO IMPRIMEDATOS
GOTO LEEDATOS
RETURN
```

```
SEMAFORO:
BTFSC PB,3
CALL VERDE
BTFSC PB,2
```

```
CALL AMARILLO  
BTFSC PB,6  
CALL VERDE  
CALL AMARILLO  
RETURN
```

```
RETARDO1:  
CLRF R1  
SIGA:  
CALL RETARDO2  
INCF R1,1  
MOVLW 04  
XORWF R1,0  
BTFSS ESTADO,2  
GOTO SIGA  
RETURN
```

```
RETARDO2:  
MOVLW 0A7  
MOVWF TMR1L  
MOVLW 61  
MOVWF TMR1H  
SIGA1: BTFSS PIR1,0  
GOTO SIGA1  
BCF PIR1,0  
RETURN
```

```
INTERRUPCION:  
MOVF PB,0
```

```
MOVWF DATOSSEMAFORO
MOVF PC,0
MOVWF DATODISPLAY
MOVLW 0E7
MOVWF PC
MOVLW 24
MOVWF PB
CLRF TITILA
CALL RETARDO2
REPITETITIL:
MOVLW 12
MOVWF PB
CALL RETARDO2
MOVLW 00
MOVWF PB
CALL RETARDO2
INCF TITILA,1
MOVLW 03
XORWF TITILA,0
BTFSS ESTADO,2
GOTO REPITETITIL
MOVF DATOSSEMAFORO,0
MOVWF PB
MOVF DATODISPLAY,0
MOVWF PC
BCF INTCON,1
RETFIE
END
```

## **10) CONCLUSIONES**

- Se ha concluido que las banderas del INTCON deben estar habilitadas para poder usar las interrupciones en este caso el GIE
- Se ha concluido que el Timer1 es mas grande ya que abarca valores hasta FFFF mientras que los timers 0 y 2 abarcan valores hasta FF
- Se ha concluido que el PIC16F877 tiene nos permite generar una interrupción externa mediante el pin 0 del puerto B

## **11) RECOMENDACIONES**

- Se recomienda verificar correctamente las distribuciones de los pines y distribuirlos en entradas o salidas según correspondan
- Se recomienda que al final del retardo se debe deshabilitar la bandera INTCON para que se pueda volver a usar.

## **12) BIBLIOGRAFIA**

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