|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Semáforo** | | | **No.** | **5** |
| **Asignatura:** | **Arquitectura de computadoras** | **Carrera:** | **Ingeniería en Sistemas Computacionales** | **Duración de la práctica (Hrs)** | **50** |

**NOMBRE DEL ALUMNO:**

**Diana Mondragón de Jesús**

**GRUPO: 3012**

**I. Competencia(s) específica(s):**

**II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):**

Aula de clases y laboratorio independiente

**III. Material empleado:**

MPLAB

Proteus

PIC16F84A

Tabla Protoboard

Cable para Protoboard

21 resistencias de 320 Ohms

Leds ultrabrillantes en color verde (8), rojo (8) y ámbar (4)

Cautín

Soldadura

Pasta para soldar

¼ de papel cascaron

Pinturas

Pistola de silicón

Barras de silicón

Popotes

Cristal oscilar de 4 MHz

2 capacitores de 22 picofaradios

8 botecitos de plástico

**IV. Desarrollo de la práctica:**

* En este proyecto se inició con el código en el lenguaje ensamblador y fue desarrollado en MPLAB

\_\_CONFIG \_CP\_OFF & \_PWRTE\_OFF & \_WDT\_ON & \_XT\_OSC

LIST P=P16F84A, R=DEC

#INCLUDE <P16F84A.INC>

ORG 0

GOTO PRINCIPAL ;SALTO AL PROGRAMA PRINCIPAL

ORG 4

GOTO IRQ ;SALTO A RUTINA DE INTERRUPCION

;DEFINIR CONSTANTES

CUENTA EQU 16 ;VALOR DE CUENTA DEL TMR0()256-240)

RP2 EQU 7 ;BITS DE LOS SEMAFOROS DE LOS PEATONES

RP1 EQU 6

VP2 EQU 1

VP1 EQU 0

;DECLARACION DE LAS VARIABLES Y LAS POSICIONES DE MEMORIA QUE USARAN

CONTADOR EQU 0X10

RETARDO EQU 0X11

;PROGRAMA PRINCIPAL

;

;INICIALIZAR EL PIC16F84A, REGISTROS DE CONTROL, PUERTOS E INTERRUPCIONES

;EL PROGRAMA LANZARA EL TEMPORIZADOR PARA QUE SE ACTIVE LA INTERRUPCION

;Y SE QUEDARA EN ESPERA DE UN BUCLE DE UN CICLO INFINITO

;

PRINCIPAL:

;SE INICIA EL PIC Y PROGRAMAR EL REGISTRO OPTION

;EL BIT 7 SERA RPBU=1 RESISTENCIAS PULL-UP INTERNAS DEL PORTB

;LAS CUALES NO SERAN NECESARIO YA QUE SE DESABILITA AL USAR EL PORTB COMO

;SALIDA.

;BIT6 INTEDG=0 DA LO MISMO PORQUE NO USAREMOS INT

;BIT5 TOCS=0 TEMPORIZADOR DEL RELOJ INTERNO

;BIT4 TOSE=0

;BIT3 PSA=0 PRE-ESCALAR EN EL TMR0

;BIT2-0=111 AJUSTA EL PRE-ESCALAR DEL TMR0 A 256

MOVLW B'10000111' ;EL VALOR DE LA PALABRA PARA OPTION SE MODIFICA LOS BIS 2 AL 0

BSF STATUS,RP0 ;BANCO DE MEMORIA 1

MOVWF OPTION\_REG ;ESCRIBE LA PALABRA EN EL REGISTRO OPTION

BCF STATUS,RP0 ;BANCO 0

;PROGRAMACION DE LOS PUERTOS

;EL PORTB=SALIDAS=> TODO A CERO

;ESTRUCTURA DEL PORTB: 7 6 5 4 3 2 1 0

; RP2 RP1 R2 A2 V2 R1 A1 V1

; \_\_^\_\_ \_\_^\_\_

; S2 S1

BSF STATUS, RP0 ;BANCO 1

CLRF TRISB ;BORRA TODOS LOS BITS DEL PORTB

CLRF TRISA ;HACE LO MISMO QUE EL ANTERIOR

BCF STATUS, RP0 ;BANCO 0

; INICIALIZAR EL CONTADOR Y EL PUERTO

CLRF CONTADOR ;PONER EL CONTador en 0

MOVLW B'10011110' ;S2 ROJO,S1 VERDE, PEATONES,

MOVWF PORTB ;LO SACA POR EL PUERTO

BCF PORTA, VP2 ;PEATONES, VERDE PARA EL S2

BSF PORTA, VP1

;PROGRAMAR EL TEMPORIZADOR

MOVLW CUENTA

MOVWF TMR0 ;LANZA LA CUENTA INICIAL DEL TEMPORIZADOR

;PROGRAMACION DE LAS INTERRUPCIONES

;BIT7 GIE=1 PARA HABILITAR LAS INTERRUPCIONES

;BIT5 TOIE=1 ARA HABILITAR LA INTERRUPCION DEL TEMPORIZADOR

MOVLW B'10100000'

MOVWF INTCON ;PROGRAMA LAS INTERRUPCIONES

ESPERA: ;BUCLE INFINITO QUE BORRA CONTINUAMENTE EL PERRO GUARDIAN

CLRWDT ;EL TRABAJO LO HACE LA IRG. SOLO SI HAY UN PROBLEMA, SE

GOTO ESPERA ;REINICIA EL MICRO

;\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

;RUTINA DE ATENCION A LA INTERRUPCION DEL TEMPORIZADOR

;

;LA INTERRUPCION SE PRODUCIRA CADA 0'05S.

;CONTAREMOS 5 LLAMADAS

;SE INCREMENTARA UN CONTADOR DE SEGUNDOS PARA CONTROLAR LA TEMPORIZACION DEL SEMAFORO,Y

;SE MODIFICARA EL ESTADO DE LOS SEMAFOROS, ESCRIBIENDO EN EL PUERTO SEGUN EL TIEMPO PASADO

;

;NO SE COMPRUEBA CUAL ES LA FUENTE DE INTERRUPCION, PORQUE SOLO ESTA HABILITADA

;LA INTERRUPCION DEL TEMORIZADOR,

IRQ:

;EMPEZAMOS PROGRAMANDO EL TEMPORIZADOR, PARA QUE NO VAYA CONTANDO YA, Y NO INTRUDUCIR

;UN RETARDO EN LA TEMPORIZACION DEBIDO AL PROCESO DE VARIACION

MOVLW CUENTA

MOVWF TMR0 ;REPROGRAMACION DEL TIMER

INCF RETARDO

MOVLW 5

SUBWF RETARDO,W ;COMPARA RETARDO CON 5

BTFSS STATUS,Z ;SI NO ES 5

GOTO FINSWITCH ;SALTA AL FINAL DE RUTINA DE INTERRUPCIONES

CLRF RETARDO

;ESTE CODIGO SE EJECUTA CADA 0'5S. EL CONTADOR CUENTA MEDIOS SEGUNDOS

INCF CONTADOR,F ;CONTADOR=CONTADOR+1

;IMPREMENTACION DE LA SENTENCIA SWITCH-CASE

CASO20: ;CONTADOR<20?

MOVLW 40

SUBWF CONTADOR,W ;COMPARA CINTADOR CON 40

BTFSC STATUS,C ;ACARREO

GOTO FINCASO20 ;NO HAY ACAREO=> CONTADPR>=40

;CONTADOR<40 (NO HAY ACARREO)

MOVLW B'10011110' ;COCHES; S2 ROJO,S1 VERDE,PEATONES RP2

MOVWF PORTB ; LO ESCRIBES EN EL PUERTO

BCF PORTA, VP2 ;PEATONES, VERDE PARA S2

BSF PORTA, VP1 ;PEATONES, VERDE PARA S1

GOTO FINSWITCH

FINCASO20:

CASO25: ;CONTADOR<25

MOVLW 50

SUBWF CONTADOR,W ;COMPARA SI EL CONTADOR ES 50

BTFSC STATUS,C ;OYE GUTIERREZ HAY ACARREO

GOTO FINCASO25 ;NO HYA CARREO=> EL CONTADOR >=50

;CONTADOR SEA MENOR QUE 25(NO HAY ACARREO)

MOVLW B'10011110' ;LOS AUTOS RS2, VS1, RP2 OFF, RP1 ON.

MOVWF PORTB ;LA MONDAMOS AL PUERTO DE SALIDA

BTFSC CONTADOR,0 ;SI CONTADOR ES PAR ENTONCES ES IGUAL A UN SEGUNDO

GOTO FINSWITCH

MOVLW B'10' ;MASCARA XOR PARA CONMUTAR VP2 SIN MODIFICAR EL RESTO

XORWF PORTA,F ;CONMUTA VP2 (CADA CADA 1 SEGUNDO)

GOTO FINSWITCH

FINCASO25:

CASO30: ;25>=CONTADOR<30

MOVLW 60

SUBWF CONTADOR,W ;COMPARA EL CONTADR CON 60

BTFSC STATUS,C ;PYE GUTIERREZ ES > QUE 60??

GOTO FINCASO30 ;NO HAY ACARREO CONTADOR=>60

;CONTADOR <30 (NO HAY ACARREO)

MOVLW B'00000010' ;MASCARA PARA CONMUTAR ASI CADA 0.5s ENTRE CADA

XORWF PORTB,F

BSF PORTB,0 ;APAGA EL VS1

BTFSC CONTADOR,0 ;SI CONTADOR ES PAR ENTONCES

GOTO FINSWITCH

MOVLW B'10' ;MASCARA PARA EL SEMAFORO DEL PEATON 2 EN VERDE

XORWF PORTA,F ;CONMUTA VP2

GOTO FINSWITCH

FINCASO30:

CASO50: ;30>=CONTADOR<50 ENTONCES

MOVLW 100

SUBWF CONTADOR,W ;COMPARA EL VALOR DEL CONTADOR CON 100

BTFSC STATUS,C ;驴HAY ACARREO?

GOTO FINCASO50 ;NO HAY ACARREO CONTADOR>=100

;C0NTADOR<50 CUANDO NO HAY ACARREO

MOVLW B'01110011' ;VS2, RS1, RP2 ON, RP1 ON.

MOVWF PORTB ;LO ENVIAMOS AL PUERTO B

BSF PORTA,VP2 ;VP2

BCF PORTA,VP1 ;VP1

GOTO FINSWITCH

FINCASO50:

CASO55 ;30>=30>55

MOVLW 110

SUBWF CONTADOR,W ;COMPARA EL CONTADOR CON 110

BTFSC STATUS,C ;HAY ACARREO

GOTO FINCASO55

;CONTADOR <55(NO HAY ACARREO)

MOVLW B'01110011' ;VS2, RS1, RP2 ON, RP11 OFF

MOVWF PORTB ;LO ESCRIBE EN EL PUERTO

BTFSS CONTADOR,0 ;SI EL CONTADOR ES PAR ENTONCES =1s

GOTO FINSWITCH

MOVLW B'01' ;ENMASCARA XOR PARA CONMUTAR VP1 SIN MODIFICAR EL RESTO

XORWF PORTA,F ;CONMUTA VP1 CADA 2 UNIDADES 1s

GOTO FINSWITCH

FINCASO55:

CASO60: ;55<=CONTADOR>60

MOVLW 120

SUBWF CONTADOR,W ;COMPARA CONTADOR CON 120

BTFSC STATUS,C ;HAY ACARREO

GOTO FINCASO60 ;NO HAY ACARREO => CONTADOR >=120

;CONTADOR<60 NO HAY ACARREO

MOVLW B'00010000' ;ENMASCARA XOR PARA CONMUTAR AS2 SIN MODIFICAR EL RESTO

BTFSS CONTADOR,0 ;SI EL CONTADOR ES PAR CADA SEGUNDO

BSF W,VP1 ;ACTIVA LA MASCARA TAMBIEN PARA VP1

XORWF PORTB,F ;COMUTA AS2 CADA, O.5s DURANTE EL TIEMPO ENTE 55 Y 60

BSF PORTB,3 ;APAGA VS2

BTFSC CONTADOR,0 ;SI EL CONTADOR ES PARA CADA SEGUNDO

GOTO FINSWITCH

MOVLW B'01' ;MASCARA PARA VP1

XORWF PORTA,F ;CONMUTA VP1

GOTO FINSWITCH

FINCASO60:

OTHERWISE: ;LLEGO A 60

MOVLW B'10011110' ;AUTOS RS2,VS2,RP2 OFF,RP1 ON

MOVWF PORTB ;L ENVIAMOS AL PUERTO

BCF PORTA,VP2 ;VP2

BSF PORTA,VP1

CLRF CONTADOR ;EMPIEZA TODO DE NUEZ

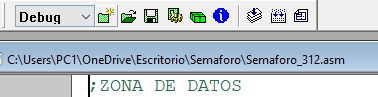
FINSWITCH: ;FIN DE TODA LA SENTENCIA SWITCH-CASE

MOVLW B'10100000'

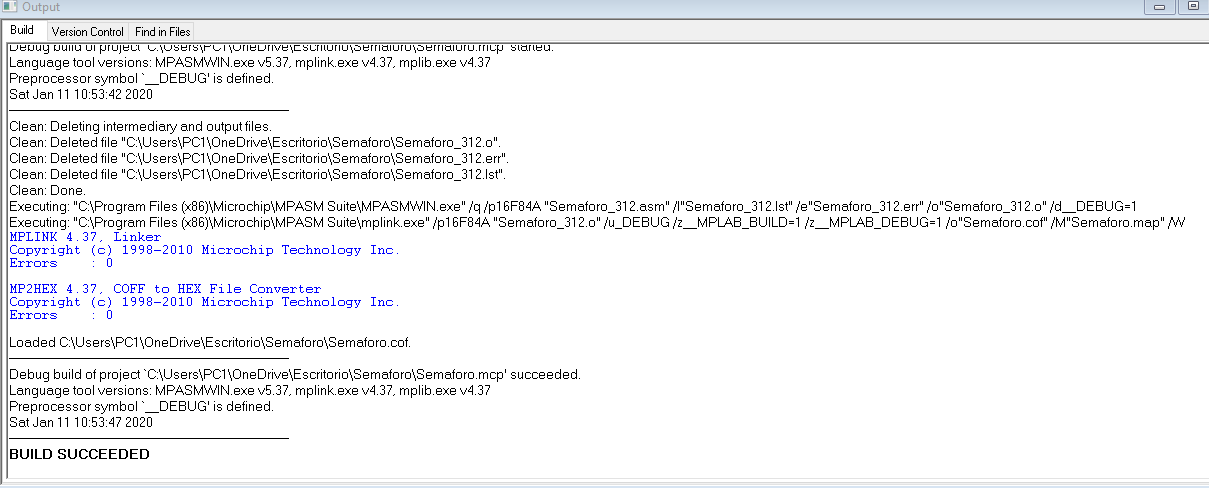
MOVWF INTCON

RETFIE ;RETORNO DE LA INTERRUPCION

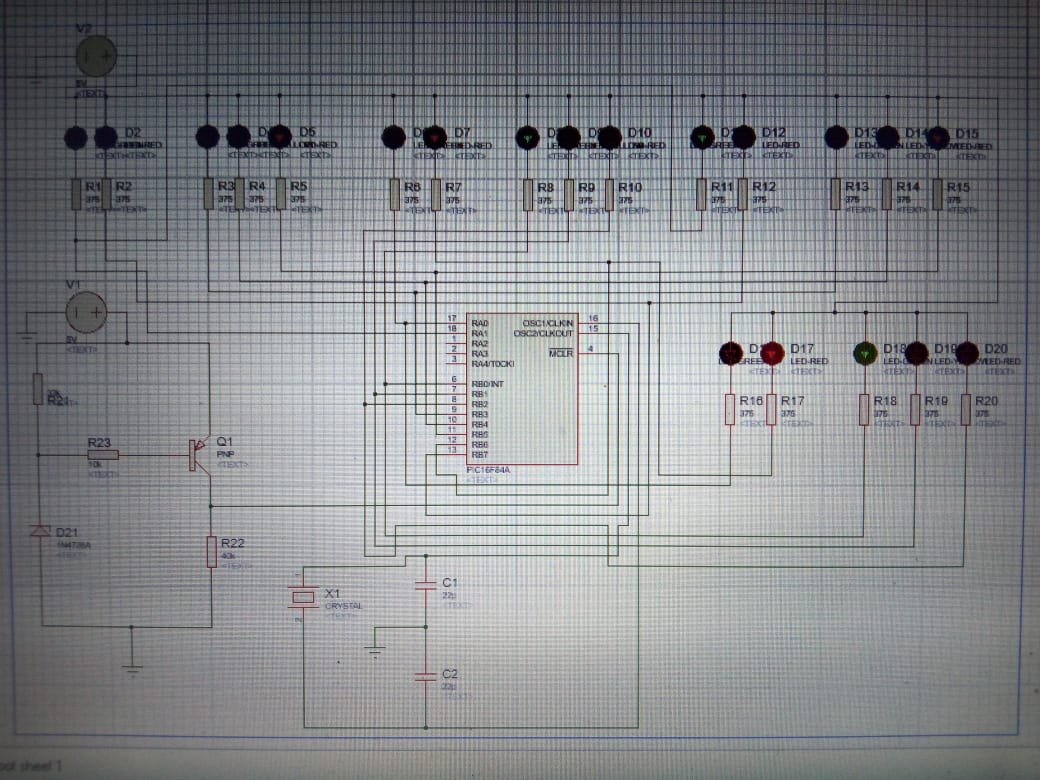
END

El código estaría acabado

* Para compilar el código debemos dar clic en este botón
* Mandara un mensaje donde se compilo correctamente y abra generado un **archivo .hex**



* Teniendo esto, pasamos a realizar el circuito de la siguiente manera



S1 y S3

Verde carro-RB3

Rojo carro-RB5

Amarillo carro-RB4

Verde peatón-RB6

Rojo peatón-RB7

S2 y S4

Verde carro-RB0

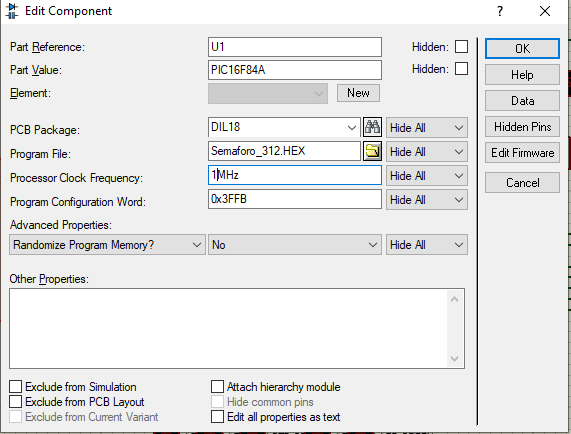
Rojo carro-RB2

Amarillo carro-RB1

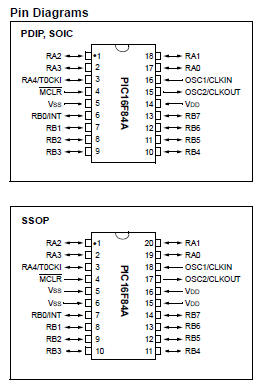
Verde peatón-RB7

Rojo peatón-RB

Damos doble clic en nuestro PIC y agregamos el archivo .hex que generamos anteriormente



Comprobamos que funcione y listo comenzaremos a cablear en la tabla Protoboard esto lo podemos hacer con ayuda del DATA SHEET DEL PIC16 f84A







**V. CONCLUSION:**

En este proyecto, aprendi varias cosas como a cablear, a soldar y a codificar en lenjuage enzamblador, tambien este proyecto fue un poco laborioso ya que requiere de tiempo y paciciencia, pues el cableado en la protoboard fue un poco complicado y al poner un cable mal no hacia su funcion o simplemente no prendia, pero a pesar de las dificultades el proyecto fue un éxito, ya que los semaforos encendieron de manera correcta.

