

**Universidad Nacional Autónoma de México**

**Facultad de Ingeniería**

**Laboratorio de Microcomputadoras**

**Práctica No.7: Puerto Serie SCI (Asíncrono)**

Profesor: Rubén Anaya García

Alumnos:

- Murrieta Villegas Alfonso
- Reza Chavarría, Sergio Gabriel
- Valdespino Mendieta Joaquín

Grupo: 4

Semestre: 2021-2

## Práctica 07: Puerto Serie SCI (Asíncrono)

### Objetivo

Familiarizar al alumno en el uso de una Interfaz de Comunicación Serie Asíncrona de un microcontrolador.

### Desarrollo

1. Escribir, comentar y ensamblar el siguiente código

```
processor 16f877
include<pl6f877.inc>

ORG 0
GOTO INICIO
ORG 5

INICIO:
;Cambio al banco 01
BSF STATUS,RP0
BCF STATUS,RP1

BSF TXSTA,BRGH ;SELECCIÓN DE ALTA VELOCIDAD DE BAUDIOS
MOVLW D'129'
MOVWF SPBRG ;Asignar 9600 BAUDS

BCF TXSTA, SYNC ;Modo de comunicación=0. Asíncrona
BSF TXSTA, TXEN ;Activación de transmisión

BCF STATUS, RP0 ;Cambio al banco 0

BSF RCSTA,SPEN ;Habilita el puerto Serie
BSF RCSTA,CREN ;Activa la recepción continua en modo de comunicación asíncrona

RECIBE:
BTFSS PIR1,RCIF ;Revisa si la recepción ha sido completada
GOTO RECIBE ;Si aun está en recepción de proceso repite

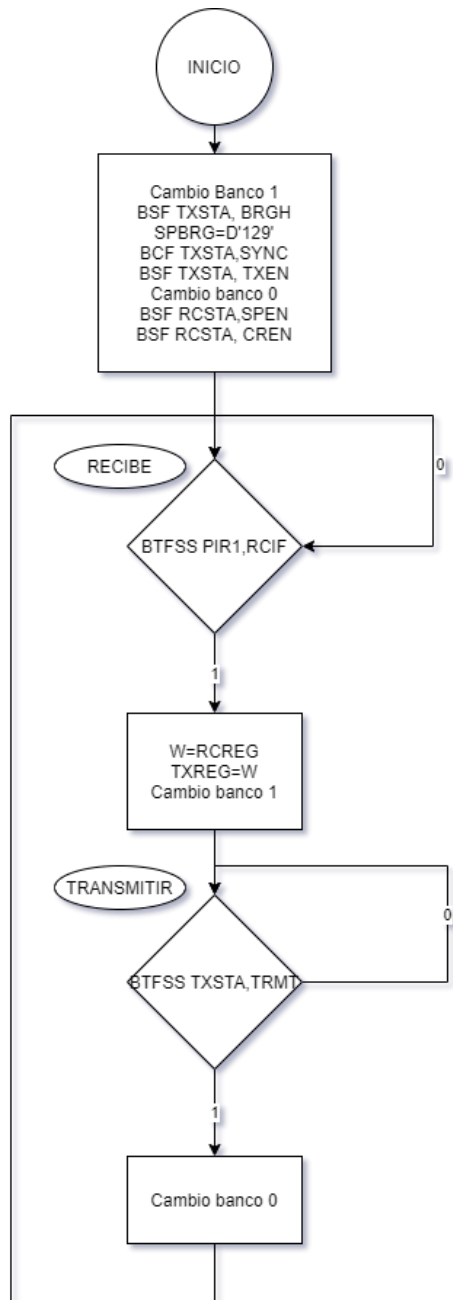
MOVWF RCREG,W ;En la recepción completada,
;se puede obtener la información por medio de RCREG
;W=RCREG
MOVWF TXREG ;TXREG=W
;Registro para mandar a transmitir

BSF STATUS,RP0 ;Cambio a banco 1

TRANSMITE:
BTFSS TXSTA,TRMT ;Revisa si ha transmitido el dato
GOTO TRANSMITE ;Si no repite el proceso
BCF STATUS,RP0 ;Cambio al banco 0
GOTO RECIBE ;Repite la recepción

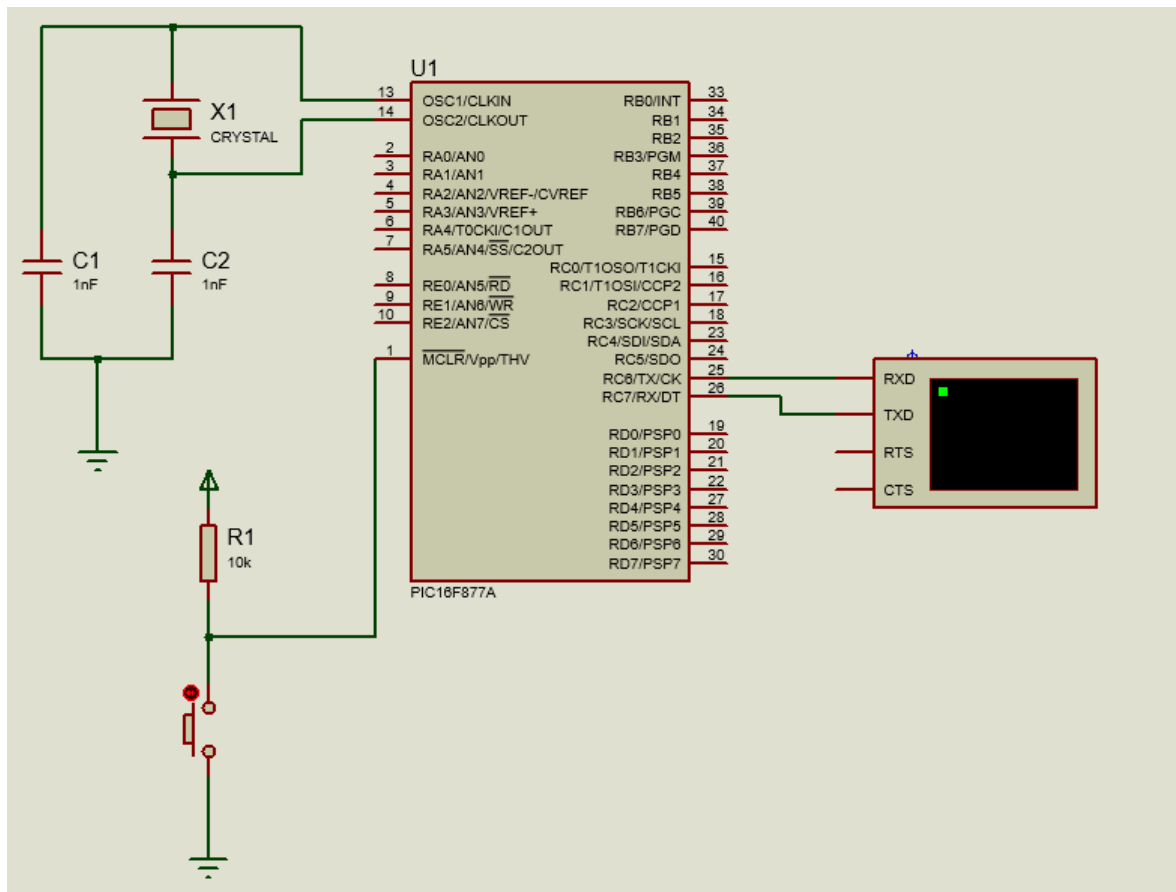
END
```

*Código 1: Código de transmisión y recepción de datos*



Modelo 1: Diagrama de flujo de Trasmisión y recepción

2. Abrir la terminal de su elección, configurar está a 9600 Bauds y un protocolo 8 bits de inicio, 8 bits de datos y un 1 de paro; comprobar su funcionamiento.



Simulación 1: Conexión del PIC16F877 con terminal

3. Realizar un programa que despliegue la siguiente cadena en una terminal.

HOLA UNAM

### Código

El código realiza el procedimiento para la configuración de transmisión, pasa al banco 1 (configuración de transferencia), asigna 9600 Bauds de transferencia, configura la comunicación asíncrona, se activa la transmisión, cambia al banco 00 (recepción) Habilita el puerto serie y activa la recepción.

Para el procedimiento se realizó una subrutina en donde se asigna al registro W el código ASCII de H y llama a la subrutina de configuración, para cambiar al banco 01. Se

continúa en la sección de transmisión, donde se revisa si el envío de información ya fue realizado, si es así se cambia al banco 0 y regresa a la sección del escrito de las letras. Si no es así, se seguirá revisando el estado del registro TRMT.

El proceso se realizará con todas las letras de la frase “HOLA UNAM”.

```
processor 16f877
include <16f877.inc>
ORG 0
GOTO INICIO
ORG 5
INICIO:
    ;Cambio al banco 01
    BSF STATUS,RP0
    BCF STATUS,RP1

    BSF TXSTA,BRGH      ;SELECCIÓN DE ALTA VELOCIDAD DE BAUDIOS
    MOVLW D'129'
    MOVWF SPBRG         ;Asignar 9600 BAUDS

    BCF TXSTA,SYNC      ;Modo de comunicación=0. Asíncrona
    BSF TXSTA,TXEN      ;Activación de transmisión

    BCF STATUS,RP0      ;Cambio al banco 0

    BSF RCSTA,SPEN      ;Habilita el puerto Serie
    BSF RCSTA,CREN      ;Activa la recepción continua en modo de comunicación asíncrona

INFO:
    ;Transmisión de los caracteres de 'HOLA UNAM'
    ;El valor en ASCII se envía a partir del registro TXREG.

    MOVLW A'H'
    MOVWF TXREG
    CALL CONF_TRANSM    ;Subrutina del cambio de banco 01
    MOVLW A'O'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'L'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'A'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A' '
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'U'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'N'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'A'
    MOVWF TXREG
    CALL CONF_TRANSM
    MOVLW A'M'
    MOVWF TXREG
    CALL CONF_TRANSM

GOTO $
```

```

CONF_TRANSM
    BSF      STATUS,RPO          ; Cambio al banco 1

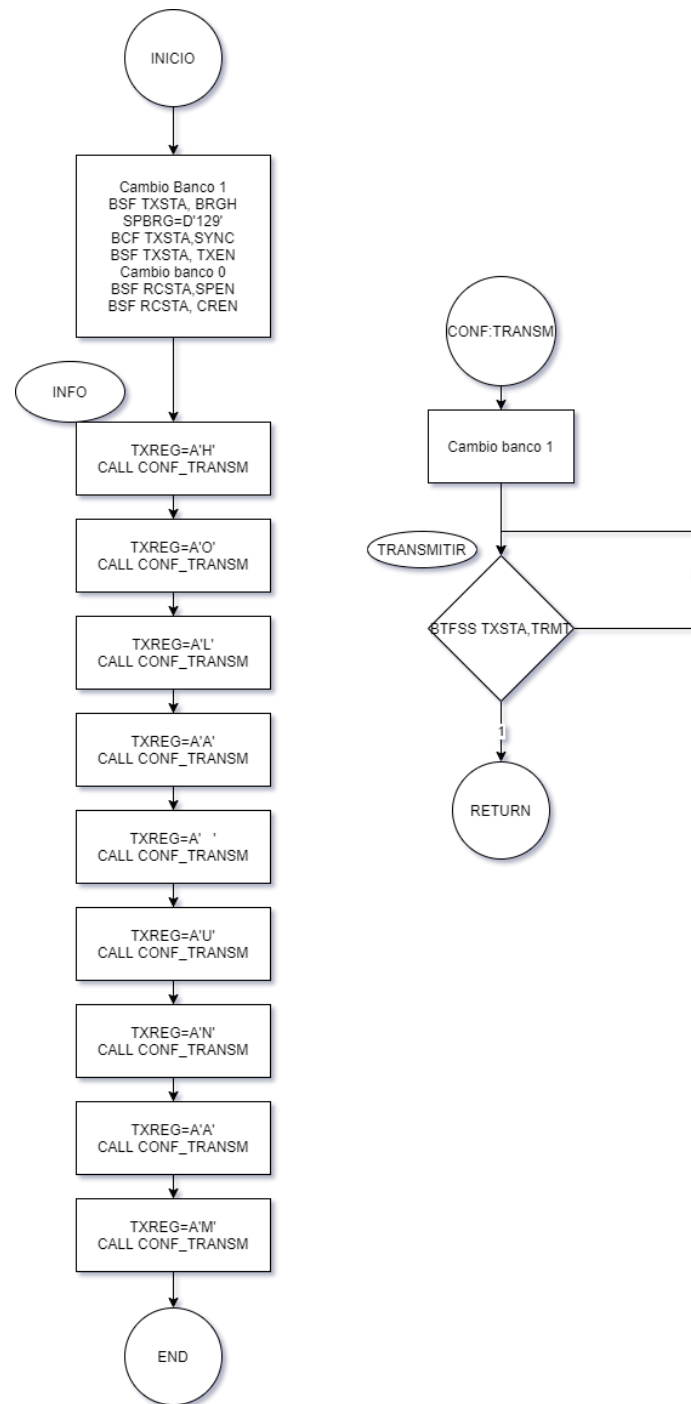
TRANSMITIR:
    BTFSS   TXSTA,TRMT          ;Revisión de transmisión exitosa
    GOTC   TRANSMITIR
    BCF     STATUS,RPO          ;Cambio al banco 0
    RETURN                                ;Regreso al último CALL

END

```

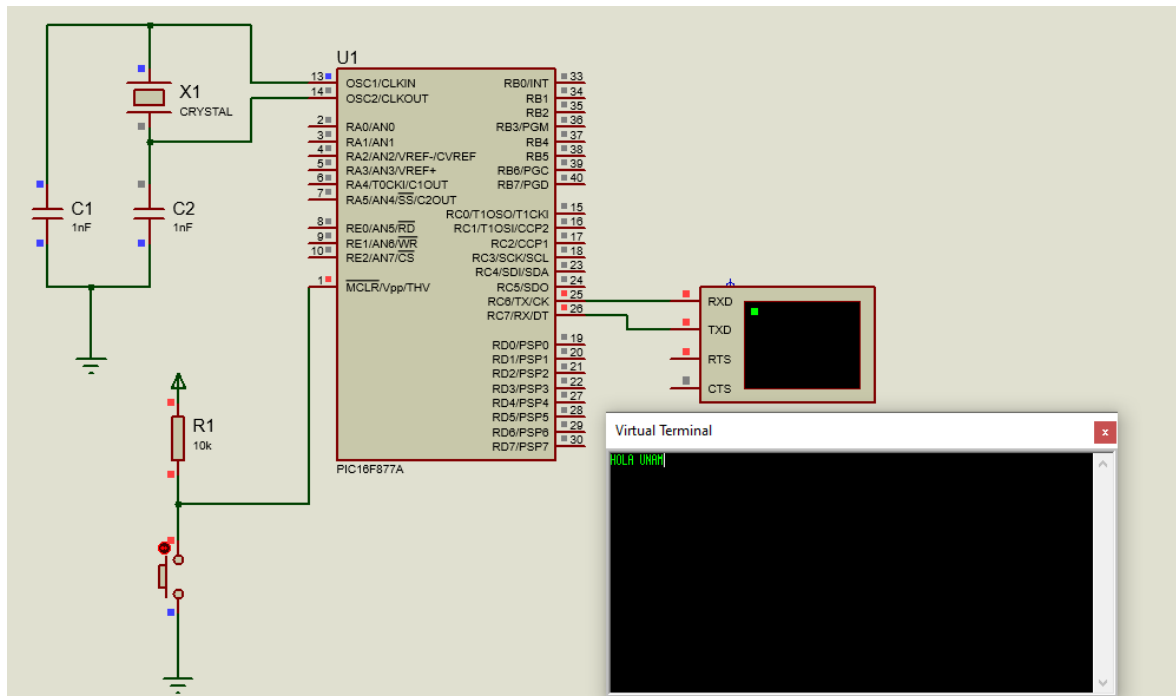
*Código 2: Ejercicio de transmitir "HOLA UNAM"*

## Diagrama de flujo



Modelo 2: Diagrama de flujo escritura de 'HOLA UNAM'

## Pruebas



Simulación 2: Simulación de transmisión de "HOLA UNAM" en terminal

- Realizar un programa que realice el control indicado; el dato proviene a través del puerto serie:

DATO	ACCION
Puerto Serie	Terminal 0 del puerto B (PB0)
'0'	0
'1'	1

Tabla 7.1 Control para activar y desactivar una señal

## Código

Al igual que el código anterior se maneja la configuración de la transmisión y la recepción de información asíncrona. También se manejó la configuración del puerto B como salida de información, para tener la salida por medio de leds.

Se recibe la información proporcionada por la terminal. Hasta que obtenga un carácter se guardará en la terminal y el código ASCII se guardará en una variable extra. La



variable se comparará con '1' y '0' por medio de XOR. Si es en alguno de los casos iguales se mandará al puerto B la salida correspondiente y se volverá a la sección de recepción. Si no obtiene alguno de los caracteres seguirá recibiendo información.

```
processor 16f877
include <pl6f877.inc>
VALOR EQU H'20'
ORG 0
GOTO INICIO
ORG 5
INICIO:
    ;Cambio al banco 01
    BSF STATUS,RP0
    BCF STATUS,RP1

    MOVLW h'0'
    MOVWF TRISE      ;Configura el puerto B como salida
    CLRF PORTE       ;Limpia los bits del PUERTO B

    BSF TXSTA,BRGH    ;SELECCIÓN DE ALTA VELOCIDAD DE BAUDIOS
    MOVLW D'129'
    MOVWF SPBRG       ;Asignar 9600 BAUDS

    BCF TXSTA,SYNC    ;Modo de comunicación=0. Asíncrona
    BSF TXSTA,TXEN    ;Activación de transmisión

    BCF STATUS,RP0    ;Cambio al banco 0

    BSF RCSTA,SPEN    ;Habilita el puerto Serie
    BSF RCSTA,CREN    ;Activa la recepción continua en modo de comunicación asíncrona

RECIBE
    BTFSS PIR1,RCIF   ;Revisa si se esta recibiendo datos
    GOTO RECIBE       ;Repetir

    MOVF RCREG,W      ;w=RCREG
    MOVWF VALOR       ;Valor a comparar
    MOVWF TXREG       ;TXREG=W Visualizar el dato en la terminal
    BSF STATUS,RP0    ;Cambio banco 1
```

```

TRASMITE:
    BTFSS TXSTA,TRMT    ;Revisar si se transmite información
    GOTO TRASMITE       ;Repetir
    BCF STATUS,RP0      ;Cambio a banco 0

SALIDA
                                ;Se revisará si entra a la terminal un 0 o 1
    MOVLW    A'1'
    XORWF    VALOR,W      ;Comparar Si es '1'
    BTFSC    STATUS,Z
    GOTO     SAL1         ;Si es igual ve a SAL1

    MOVLW    A'0'
    XORWF    VALOR,W      ;Comparar si es '0'
    BTFSC    STATUS,Z
    GOTO     SAL0         ;Si es igual ve a SAL0

    GOTO RECIBE           ;Si no es igual a alguno seguir recibiendo

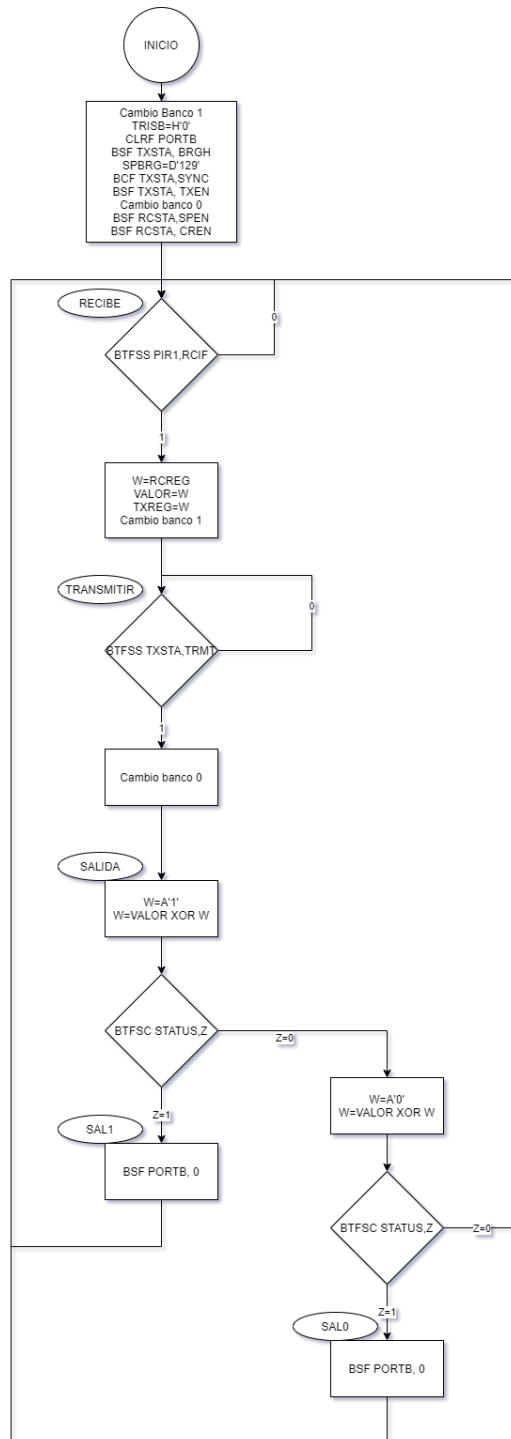
SAL1
    BSF      PORTB,0      ;Asignar 0x01 al puerto B
    GOTO     RECIBE       ;Seguir recibiendo
SAL0
    BCF      PORTB,0      ;Asignar 0x00 al puerto B
    GOTO     RECIBE       ;Seguir recibiendo

END

```

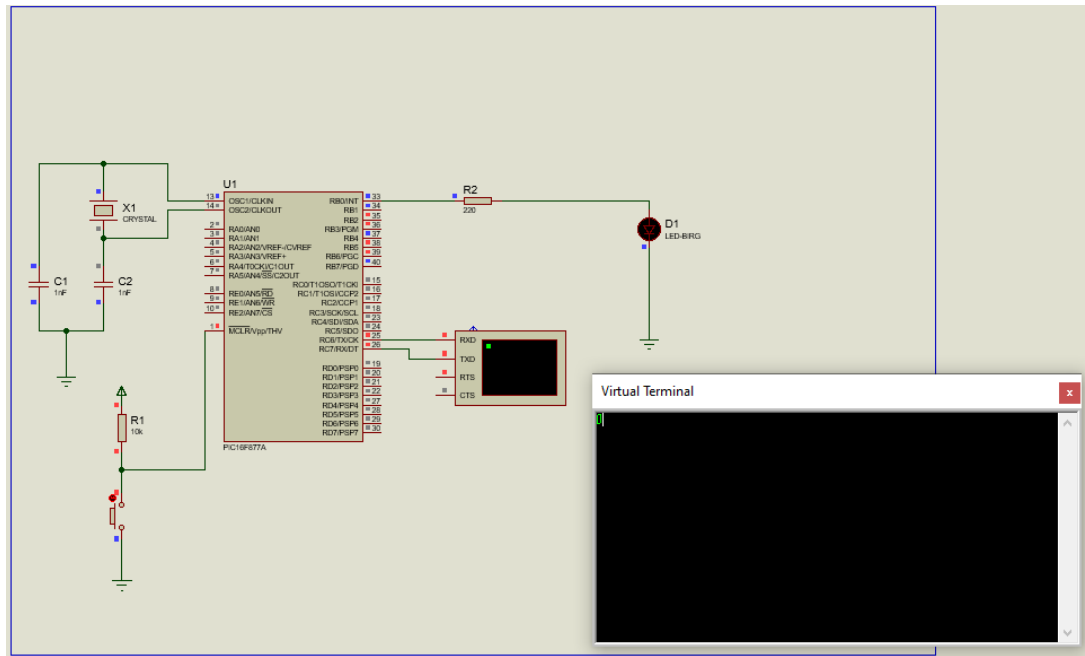
*Código 3: Código de salida del led*

## Diagrama de flujo

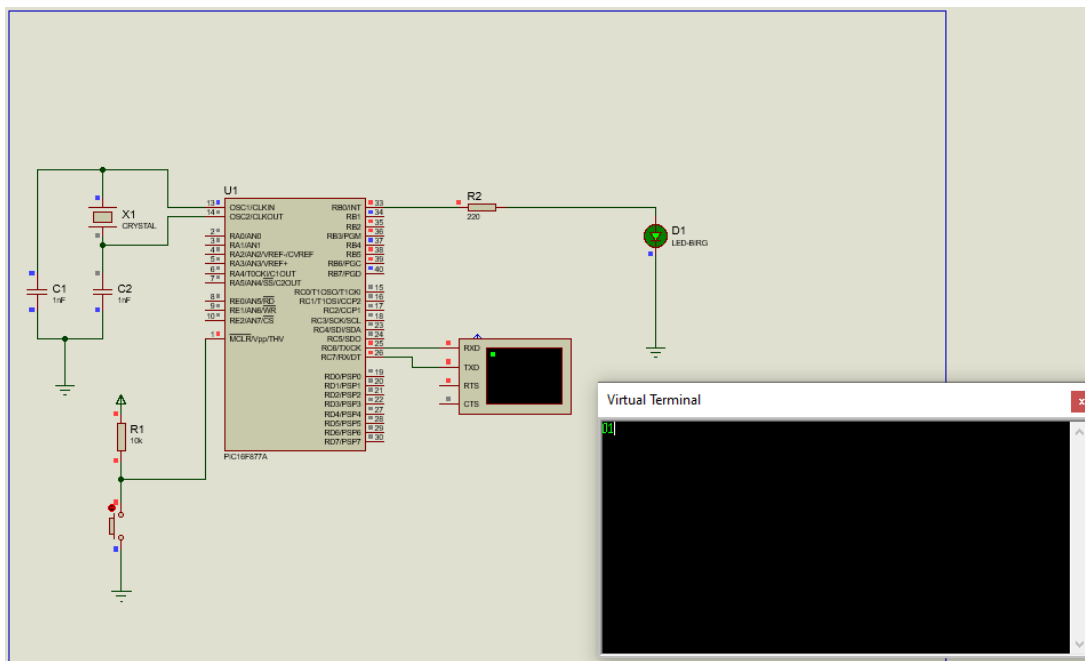


Modelo 3: Diagrama de flujo salida de 0 y 1 por el puerto B

## Prueba



Pruebas 1: Salida 0 con entrada en terminal de '0'



Pruebas 2: Salida 1 con entrada en terminal de '1'

- Realizar un programa que realice el control indicado; la secuencia será ejecutada cada que sea recibido el comando, usar retardos de ½ segundo entre cada estado generado

DATO	ACCION
Puerto Serie	Terminal 0 del puerto B (PB0)
'D' ó 'd'	10000000
	01000000
	00100000
	00010000
	00001000
	00000100
	00000010
	00000001
'I' ó 'i'	00000001
	00000010
	00000100
	00001000
	00010000
	00100000
	01000000
	10000000

Tabla 7.2 Secuencia de control

### Código

Para el código se manejó la configuración de la comunicación asíncrona y la configuración del puerto B como salida, esta vez se usarán todos los bits del puerto.

A partir de la recepción de datos se va a comparar el código ASCII obtenido. En el caso que sea 'D' o 'd', se hacen comparaciones por separado, se dirigirán a la sección del Recorrimiento a la derecha. En el caso que sea 'I' o 'i', se dirigirá a la sección de Recorrimiento a la izquierda. Si no coincide el ASCII se limpiará el puerto B.

El Recorrimiento a la derecha asigna el valor de 0X80 y se le asignará un retardo. Se utilizará la instrucción RRF y cada vez que se recorra habrá un retardo. Se revisa el estado del bit de acarreo. Si es C=0 seguirá repitiendo el Recorrimiento, si no es así se dirigirá a seguir recibiendo información.

El Recorrimiento a la izquierda asigna el valor de 0X01 y se le asignará un retardo. Se utilizará la instrucción RLF y cada vez que se recorra habrá un retardo. Se revisa el estado del bit de acarreo. Si es C=0 seguirá repitiendo el Recorrimiento, si no es así se dirigirá a seguir recibiendo información.

```

processor 16f877
include <pl6f877.inc>
VALOR EQU H'20'
VALOR1 EQU H'51'
VALOR2 EQU H'52'
VALOR3 EQU H'53'
CTE1 equ 70h      ;cte1=70h
CTE2 equ 70h      ;cte2=70h
CTE3 equ 70h      ;cte3=70h
ORG 0
GOTO INICIO
ORG 5
INICIO:
;Cambio al banco 01
BSF STATUS,RP0
BCF STATUS,RP1

MOVLW h'0'
MOVWF TRISE      ;Configura el puerto B como salida
CLRF PORTE       ;Limpia los bits del PUERTO B

BSF TXSTA,BRGH    ;SELECCIÓN DE ALTA VELOCIDAD DE BAUDIOS
MOVLW D'129'
MOVWF SPBRG       ;Asignar 9600 BAUDS

BCF TXSTA,SYNC    ;Modo de comunicación=0. Asíncrona
BSF TXSTA,TXEN    ;Activación de transmisión

BCF STATUS,RP0    ;Cambio al banco 0

BSF RCSTA,SPEN    ;Habilita el puerto Serie
BSF RCSTA,CREN    ;Activa la recepción continua en modo de comunicación asíncrona

RECIBE
BTFSS PIR1,RCIF   ;Revisa si se esta recibiendo datos
GOTO RECIBE       ;Repetir

MOVF RCREG,W      ;w=RCREG
MOVWF VALOR       ;Valor a comparar
MOVWF TXREG       ;TXREG=W Visualizar el dato en la terminal
BSF STATUS,RP0    ;Cambio banco 1

TRANSMITE:
BTFSS TXSTA,TRMT  ;Revisar si se transmite información
GOTO TRANSMITE    ;Repetir
BCF STATUS,RP0    ;Cambio a banco 0

SALIDA
;Se revisará si entra a la terminal un D O I

MOVLW A'D'
XORWF VALOR,W     ;Comparar Si es 'D'
BTFSC STATUS,Z
GOTO SAL_D        ;Si es igual ve a SAL_D

MOVLW A'd'
XORWF VALOR,W     ;Comparar Si es 'd'
BTFSC STATUS,Z
GOTO SAL_D        ;Si es igual ve a SAL_D

MOVLW A'I'
XORWF VALOR,W     ;Comparar si es 'I'
BTFSC STATUS,Z
GOTO SAL_I        ;Si es igual ve a SAL_I

MOVLW A'i'
XORWF VALOR,W     ;Comparar si es 'i'
BTFSC STATUS,Z
GOTO SAL_I        ;Si es igual ve a SAL_I

NONE
CLRF PORTE
GOTO RECIBE

```

```

SAL_D

    MOVLW    0X80
    MOVWF    PORTB          ;PORTB=0X80
    BCF      STATUS,C        ;Limpiar C
    CALL     RETARDO
LOOP_D
    RRF      PORTB           ;Recorrimiento a la derecha
    CALL     RETARDO
    BTFSS    STATUS,C        ;Si el C=1?
    GOTO     LOOP_D          ;Si C=0 repetir recorrimiento
    GOTO     RECIBE          ;Seguir recibiendo

SAL_I

    MOVLW    0X01
    MOVWF    PORTB          ;PORTB=0X01
    BCF      STATUS,C        ;Limpiar C
    CALL     RETARDO
LOOP_L
    RLF      PORTB           ;Recorrimiento a la izquierda
    CALL     RETARDO
    BTFSS    STATUS,C        ;Si el C=1?
    GOTO     LOOP_L          ;Si C=0 repite recorrimiento
    GOTO     RECIBE          ;Si C=1 Seguir recibiendo

RETARDO
    MOVLW    CTE1            ;W=20H
    MOVWF    VALOR1          ;valor1=20H
TRES
    MOVLW    CTE2            ;W=50H
    MOVWF    VALOR2          ;valor2=50H
DOS
    MOVLW    CTE3            ;W=60h
    MOVWF    VALOR3          ;valor3=60H
UNO
    DECFSZ   VALOR3          ;Decementa valor3 -1
    GOTO     UNO             ;Si el resultado es diferente de 0 ir a uno
    DECFSZ   VALOR2          ;Decementa valor2 -1
    GOTO     DOS             ;Si el resultado es diferente de 0 ir a dos
    DECFSZ   VALOR1          ;Decementa valor1 -1
    GOTO     TRES            ;Si el resultado es diferente de 0 ir a tres
RETURN
END

```

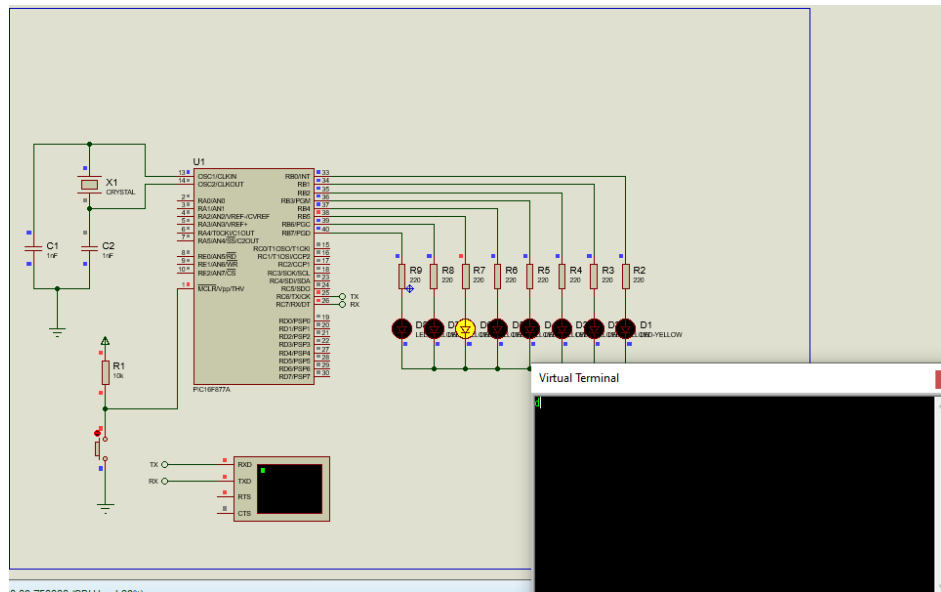
*Código 4: Recorrimiento de salidas por medio de la terminal*

[illegible]

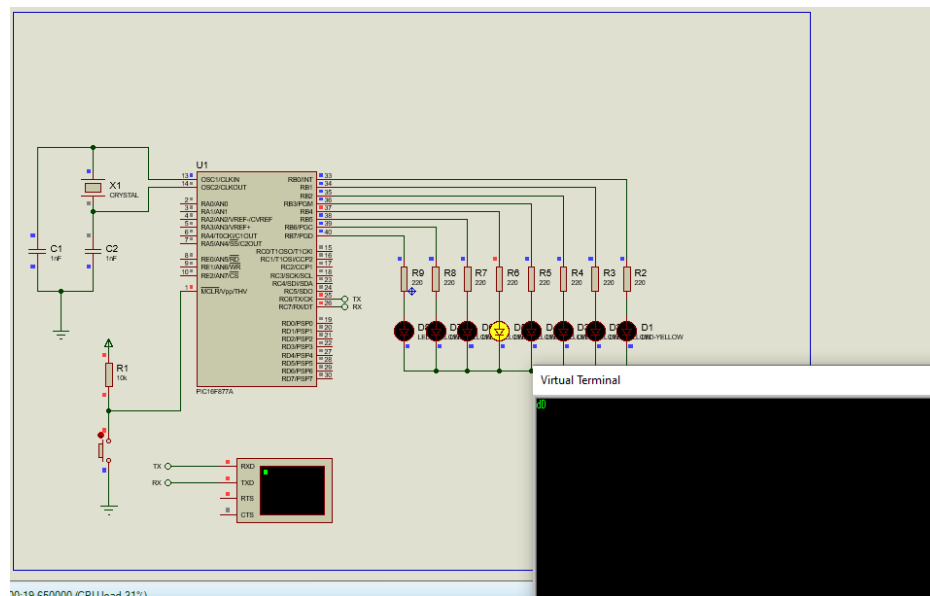
## PRÁCTICA 07



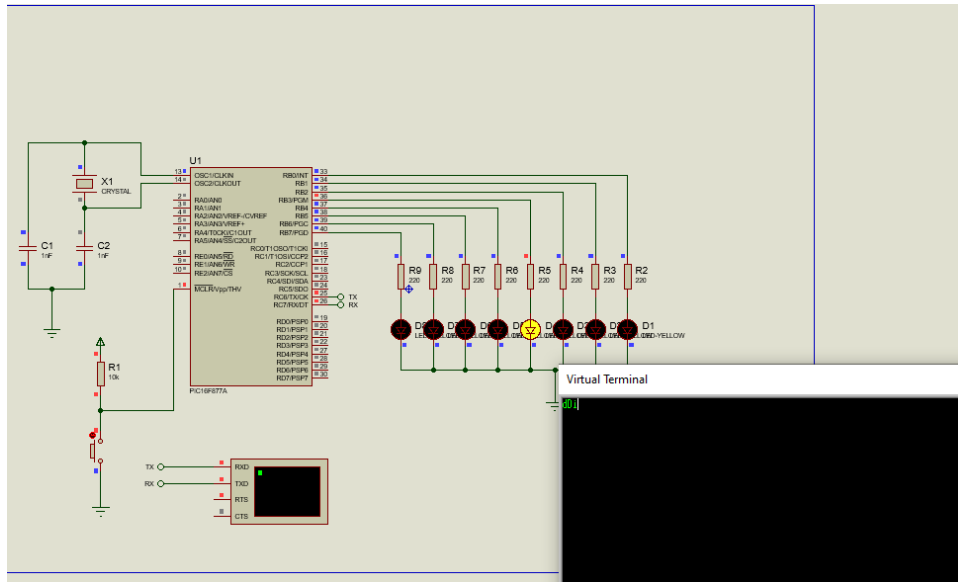
## Pruebas



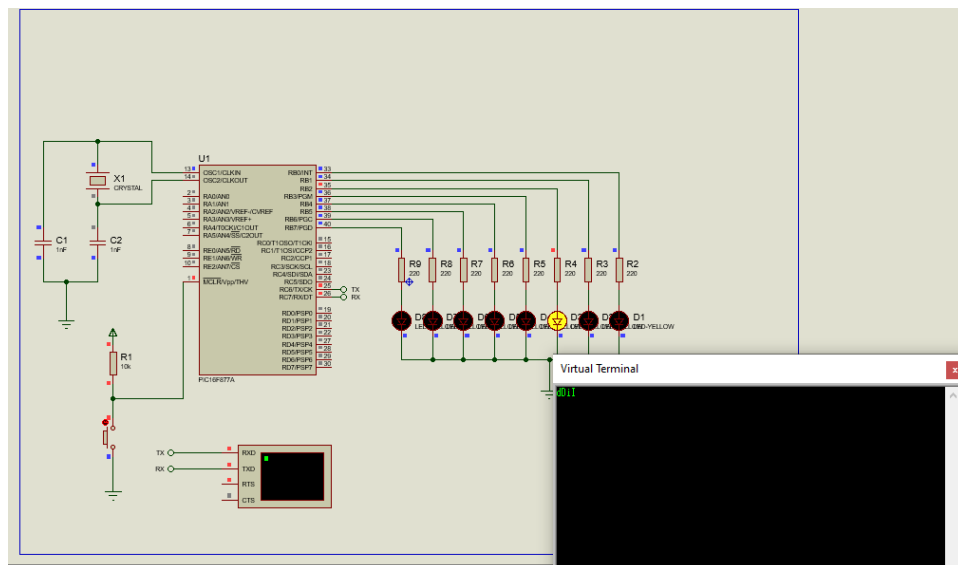
Pruebas 3: Recorrimiento a la derecha con 'd'



Pruebas 4: Recorrimiento a la derecha con 'D'



Pruebas 5: Recorrimiento a la izquierda con 'i'



Pruebas 6: Recorrimiento a la izquierda con 'T'

6. Descargar la aplicación practica7.apk e instalar en su dispositivo móvil (Android), realizar un programa para el microcontrolador, de manera que reciba el comando a través del puerto serie, con conexión inalámbrica (bluetooth), par que genere el control indicado en la tabla 7.3.

Comando Puerto serie	ACCION	
	MOTOR M1	MOTOR M2
'S'	PARO	PARO
'A'	DERECHA	DERECHA
'T'	IZQUIERDA	IZQUIERDA
'D'	DERECHA	IZQUIERDA
'I'	IZQUIERDA	DERECHA

### Código

Una vez configurada la transmisión, recepción de datos y el puerto de salida B se empezará a recibir y transmitir información. En el caso que el microprocesador obtenga algún carácter se revisará el código ASCII de la información obtenida.

Si obtiene el carácter 'S' se enviará al estado 1, donde se mandará al puerto B el valor de 0X00 (paro en ambas ruedas). Si obtiene una 'A' se envía al estado 2, donde se manda al puerto de salida 0X36 (giros de los motores a la derecha). Si obtiene una 'T' se dirige al estado 3, se envía al puerto B 0x2D (giros de los motores a la izquierda). Si se obtiene una D se dirige al estado 4, donde se envía al puerto B 0X35 (el motor 1 gira a la derecha y el 2 a la izquierda). Si obtiene el 'I' se envía al estado 5, donde sale el valor de 0X2E (el motor 1 gira a la izquierda y el 2 a la derecha). Si el carácter obtenido es diferente se volverá a esperar un nuevo carácter.

```

processor 16f877
include <pl6f877.inc>
VALOR EQU H'20'

ORG 0
GOTO INICIO
ORG 5
INICIO:
;Cambio al banco 01
BSF STATUS,RP0
BCF STATUS,RP1

MOVLW h'0'
MOVWF TRISE ;Configura el puerto B como salida
CLRF PORTB ;Limpia los bits del PUERTO B

BSF TXSTA,BRGH ;SELECCIÓN DE ALTA VELOCIDAD DE BAUDIOS
MOVLW D'129'
MOVWF SPBRG ;Asignar 9600 BAUDS

BCF TXSTA,SYNC ;Modo de comunicación=0. Asíncrona
BSF TXSTA,TKEN ;Activación de transmisión

BCF STATUS,RP0 ;Cambio al banco 0

BSF RCSTA,SPEN ;Habilita el puerto Serie
BSF RCSTA,CREN ;Activa la recepción continua en modo de comunicación asíncrona

CLRF PORTB

```

```

RECIBE
    BTFSS PIR1,RCIF      ;Revisa si se esta recibiendo datos
    GOTO RECIBE          ;Repetir

    MOVF RCREG,W         ;w=RCREG
    MOVWF VALOR          ;Valor a comparar
    MOVWF TXREG          ;TXREG=W Visualizar el dato en la terminal
    BSF STATUS,RP0       ;Cambio banco 1

TRASMITE:
    BTFSS TXSTA,TRMT     ;Revisar si se transmite información
    GOTO TRASMITE        ;Repetir
    BCF STATUS,RP0       ;Cambio a banco 0

GIROS
    MOVLW A'S'
    XORWF VALOR,W        ;Comparar Si es 'S'
    BTFSC STATUS,Z       ;Si es igual ve a ESTADO_1
    GOTO ESTADO_1

    MOVLW A'A'
    XORWF VALOR,W        ;Comparar Si es 'A'
    BTFSC STATUS,Z       ;Si es igual ve a ESTADO_2
    GOTO ESTADO_2

    MOVLW A'T'
    XORWF VALOR,W        ;Comparar Si es 'T'
    BTFSC STATUS,Z       ;Si es igual ve a ESTADO_3
    GOTO ESTADO_3

    MOVLW A'D'
    XORWF VALOR,W        ;Comparar Si es 'D'
    BTFSC STATUS,Z       ;Si es igual ve a ESTADO_4
    GOTO ESTADO_4

    MOVLW A'I'
    XORWF VALOR,W        ;Comparar Si es 'I'
    BTFSC STATUS,Z       ;Si es igual ve a ESTADO_5
    GOTO ESTADO_5

    GOTO RECIBE

ESTADO_1                ;PARO PARO
    MOVLW 0X00
    MOVWF PORTB
    GOTO RECIBE

ESTADO_2                ;DERECHA DERECHA
    MOVLW 0X36
    MOVWF PORTB
    GOTO RECIBE

ESTADO_3                ;IZQUIERDA IZQUIERDA
    MOVLW 0X2D
    MOVWF PORTB
    GOTO RECIBE

ESTADO_4                ;DERECHA IZQUIERDA
    MOVLW 0X35
    MOVWF PORTB
    GOTO RECIBE

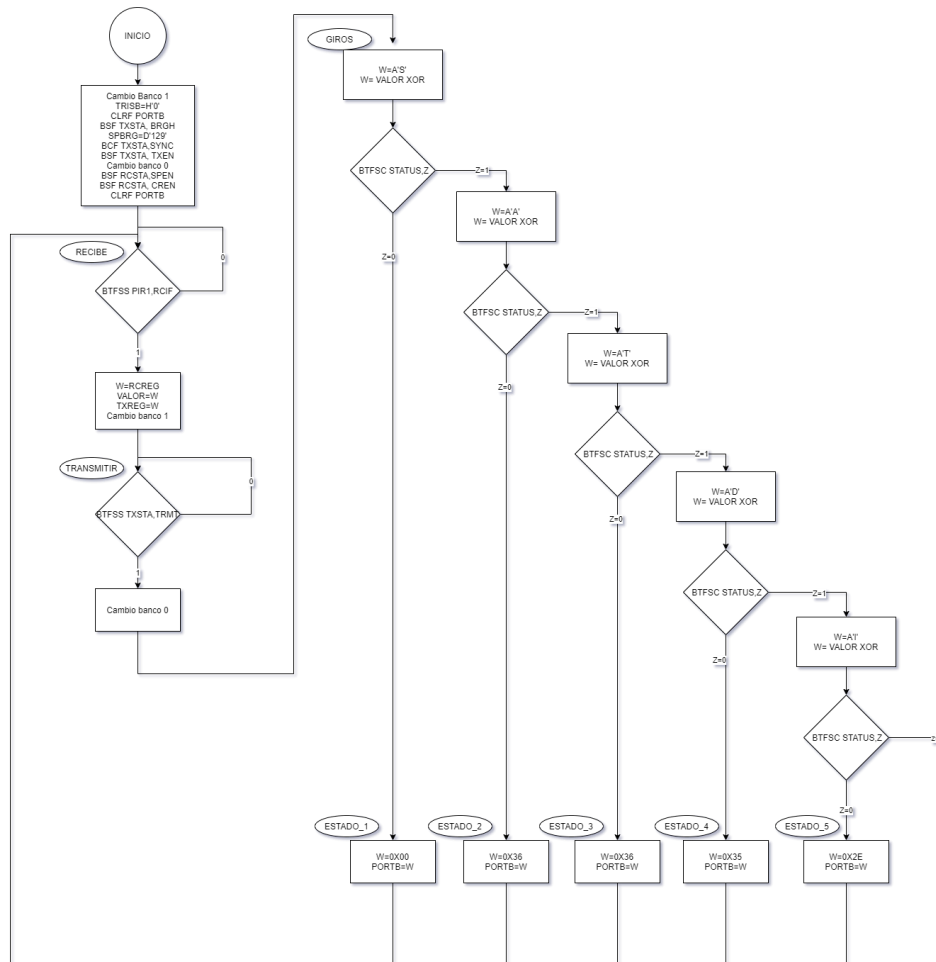
ESTADO_5                ;IZQUIERDA DERECHA
    MOVLW 0X2E
    MOVWF PORTB
    GOTO RECIBE

END

```

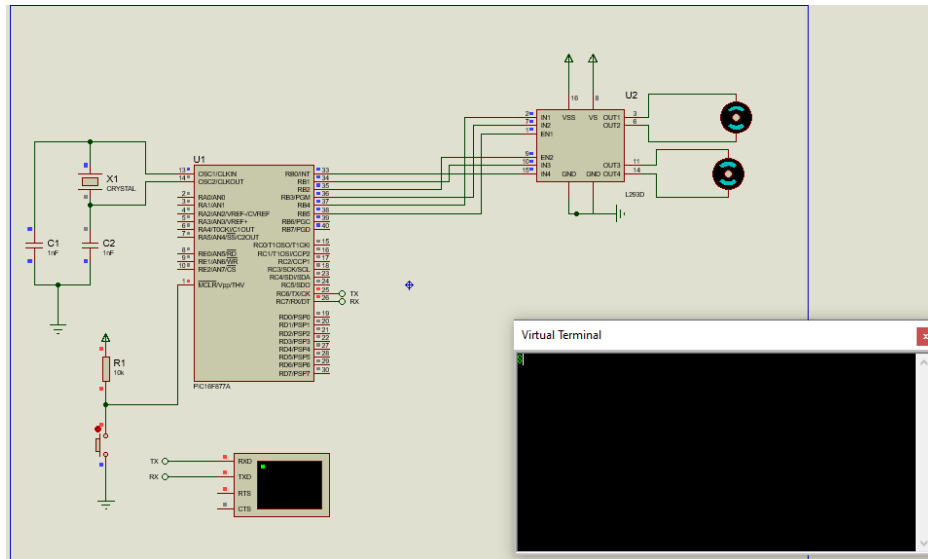
*Código 5: Cambio de dirección de motores por transmisión*

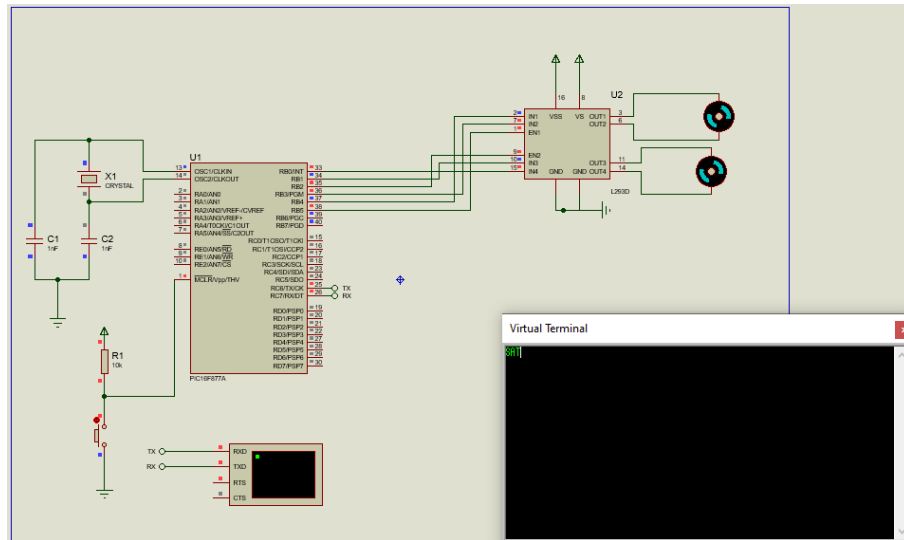
## Diagrama de flujo



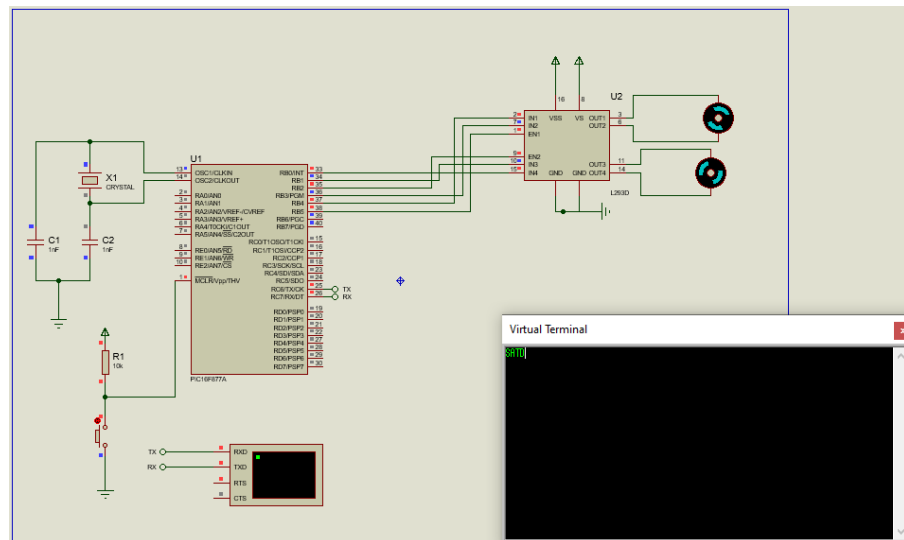
Modelo 5: Movimiento de motores

## Pruebas





Pruebas 9: Estado 'T' (IZQUIERDA, IZQUIERDA)



Pruebas 10: Estado 'D' (DERECHA, IZQUIERDA)





Limite	Hexadecimal	Binario
100	33	0011-0011
90	2E	0010-1110
80	29	0010-1001
70	24	0010-0100
60	1E	0001-1110
50	19	0001-1001
40	14	0001-0100
30	0F	0000-1111
20	0A	0000-1010
10	05	0000-0101

Limite	Hexadecimal	Binario	Estado 2 ADRESL
9	04	0000-0100	10
8	04	0000-0100	00
7	03	0000-0011	10
6	03	0000-0011	00
5	02	0000-0010	10
4	02	0000-0010	00
3	01	0000-0001	10
2	01	0000-0001	00
1	00	0000-0000	10
0	00	0000-0000	00

### ***Código***

Al configurar el convertidor A/D (para obtener la información de termómetro LM35) y la configuración de transmisión y recepción se empezará a obtener la información digital de la señal. El manejo de obtener el número es por medio de los rangos obtenidos en las tablas anteriores.

Una vez obtenida la información de ADRESH se comparará con los rangos de las centenas, decenas y unidades. Para las comparaciones de centenas se manejará la resta del valor obtenido y 0x33 (100) para saber si es mayor o menor. Si es menor a 100 en las centenas aparece un 0, si es mayor aparece un 1 en la terminal y se le restará 100 al valor obtenido para la obtención de las centenas.

Para las decenas se compara el valor obtenido con los rangos de decenas de la tabla y se le restará el valor correspondiente. Y para las unidades se manejará la comparación del valor entre los valores 0x00, 0x01, 0x02, 0x03, 0x04. Esto se hace ya que 2 números tienen esa codificación, en la parte de ADRESH. Así que se comparará los 2 bits obtenidos en ADRESL. Se compara el segundo bit para revisar si es una unidad par o impar.

Lo último para el formato se manejará la salida de los caracteres 'C', '°' y 'enter'. El proceso se repetirá la cantidad correspondiente al contador. Esto se hace ya que la señal del termómetro tarda en llegar a la señal correspondiente.

```

processor 16f877
include<pl6f877.inc>

;Variables para el DELAY
TEMP EQU H'30'
B_TEMP EQU H'31'
CONT EQU h'32'
valor1 equ h'21'
valor2 equ h'22'
valor3 equ h'23'
cte1 equ 50h
cte2 equ 50h
cte3 equ 60h

ORG 0
GOTO INICIO
ORG 5
INICIO:
;Configuración Convertidor A/D
CLRF PORTA ;Algoritmo para generar los registros analógicos.
BSF STATUS,RP0 ;Cambio al Banco 1
BCF STATUS,RP1

MOVLW 00h ;Configura puertos A y E como analógicos 00->analógicos
MOVWF ADCON1

BCF STATUS,RP0

MOVLW B'11000001' ;Configuración del registro analógico
;Se configura el canal 0->
;Frecuencia del reloj:11
;CHS2-0:000
;GO/DONE:0 Termina la conversión
;-:0
;adon:0 enciende el convertidor

MOVWF ADCON0 ;Asigna la conf. al adcon0

BSF STATUS,RP0 ;Cambio al Banco 1
BCF STATUS,RP1

BSF TXSTA,BRGH
MOVLW D'129'

MOVWF SPBRG ;Asignar 9600 BAUDS

BCF TXSTA,SYNC ;Modo de comunicación=0. Asíncrona
BSF TXSTA,TXEN ;Activación de transmisión

BCF STATUS,RP0 ;Cambio al banco 0

BSF RCSTA,SPEN ;Habilita el puerto Serie
BSF RCSTA,CREN ;Activa la recepción continua en modo de comunicación asíncrona

CLRF TEMP
CLRF B_TEMP

MOVLW H'70'
MOVWF CONT

```

```

LECTURA
    BSF ADCON0,2
    CALL RETARDO

ESPERA:
    CLRF TEMP
    CLRF B_TEMP
    BTFSC ADCON0,2 ;Si está prendido el convertidor
    GOTO ESPERA
    MOVF ADRESH,W ;Registro de los resultados en la parte alta
    MOVWF TEMP

    CALL COMP_CENT
    CALL COMP_DEC
    CALL COMP_UNI
    CALL FORM
    CALL FORM_C
    CALL ENTER
    CALL RETARDO
    DECFSZ CONT
    GOTO LECTURA
    GOTO LOOP

LOOP
    GOTO LOOP

FORM
    MOVLW H'F8'
    GOTO CONF

FORM_C
    MOVLW A'C'
    GOTO CONF

ENTER
    MOVLW H'0D'
    GOTO CONF

COMP_CENT
    MOVLW 0X33
    SUBWF TEMP,0
    BTFSS STATUS,C
    GOTO CIEN_0
    GOTO CIEN_1

;Comparación Decenas
COMP_DEC
    MOVLW 0X05
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO ZERO

    MOVLW 0X0A
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO ONE

    MOVLW 0X0F
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO TWO

    MOVLW 0X14
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO THREE

    MOVLW 0X19
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO FOUR

    MOVLW 0X1E
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO FIVE

    MOVLW 0X24
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO SIX

    MOVLW 0X29
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO SEVEN

    MOVLW 0X2E
    SUBWF TEMP,W
    BTFSS STATUS,C
    GOTO EIGHT

    GOTO NINE

```

```

;Comparación unidades
COMP_UNI

        MOVLW    0X04
        XORWF    TEMP,W
        BTFSK    STATUS,Z
        GOTO     EST_8_9

        MOVLW    0X03
        XORWF    TEMP,W
        BTFSK    STATUS,Z
        GOTO     EST_6_7

        MOVLW    0X02
        XORWF    TEMP,W
        BTFSK    STATUS,Z
        GOTO     EST_4_5

        MOVLW    0X01
        XORWF    TEMP,W
        BTFSK    STATUS,Z
        GOTO     EST_2_3

        GOTO     EST_0_1
;Revisar parte baja para unidades
EST_8_9    BSF STATUS,RP0    ;Cambio al Banco 1
        MOVF     ADRESL,W
        MOVWF    B_TEMP
        SWAPF    B_TEMP
        RRF      B_TEMP
        RRF      B_TEMP
        BTFSK    B_TEMP,1
        GOTO     EIGHT
        GOTO     NINE

EST_6_7    BSF STATUS,RP0    ;Cambio al Banco 1
        MOVF     ADRESL,W
        MOVWF    B_TEMP
        SWAPF    B_TEMP
        RRF      B_TEMP
        RRF      B_TEMP
        BTFSK    B_TEMP,1
        GOTO     SIX
        GOTO     SEVEN

EST_4_5    BSF STATUS,RP0    ;Cambio al Banco 1
        MOVF     ADRESL,W
        MOVWF    B_TEMP
        SWAPF    B_TEMP
        RRF      B_TEMP
        RRF      B_TEMP
        BTFSK    B_TEMP,1
        GOTO     FOUR
        GOTO     FIVE

EST_2_3    BSF STATUS,RP0    ;Cambio al Banco 1
        MOVF     ADRESL,W
        MOVWF    B_TEMP
        SWAPF    B_TEMP
        RRF      B_TEMP
        RRF      B_TEMP
        BTFSK    B_TEMP,1
        GOTO     TWO
        GOTO     THREE

EST_0_1    BSF STATUS,RP0    ;Cambio al Banco 1
        MOVF     ADRESL,W
        MOVWF    B_TEMP
        SWAPF    B_TEMP
        RRF      B_TEMP
        RRF      B_TEMP
        BTFSK    B_TEMP,1
        GOTO     ZERO
        GOTO     ONE

```

```

CIEN_1
    MOVLW    0X33
    SUBWF    TEMP
    MOVLW    A'1'
    GOTO     CONF

CIEN_0
    MOVLW    A'0'
    GOTO     CONF

NINE
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X2E
    SUBWF    TEMP,1
    MOVLW    A'9'
    GOTO     CONF

EIGHT
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X29
    SUBWF    TEMP,1
    MOVLW    A'8'
    GOTO     CONF

SEVEN
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X24
    SUBWF    TEMP,1
    MOVLW    A'7'
    GOTO     CONF

SIX
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X1E
    SUBWF    TEMP,1
    MOVLW    A'6'
    GOTO     CONF

FIVE
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X19
    SUBWF    TEMP,1
    MOVLW    A'5'
    GOTO     CONF

FOUR
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X14
    SUBWF    TEMP,1
    MOVLW    A'4'
    GOTO     CONF

THREE
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X0F
    SUBWF    TEMP,1
    MOVLW    A'3'
    GOTO     CONF

TWO
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X0A
    SUBWF    TEMP,1
    MOVLW    A'2'
    GOTO     CONF

ONE
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    0X05
    SUBWF    TEMP,1
    MOVLW    A'1'
    GOTO     CONF

ZERO
    BCF STATUS,RP0 ;Cambio al Banco 0
    MOVLW    A'0'
    GOTO     CONF

CONF
    MOVWF    TXREG
    BSF      STATUS,RP0 ;Cambio al banco 1

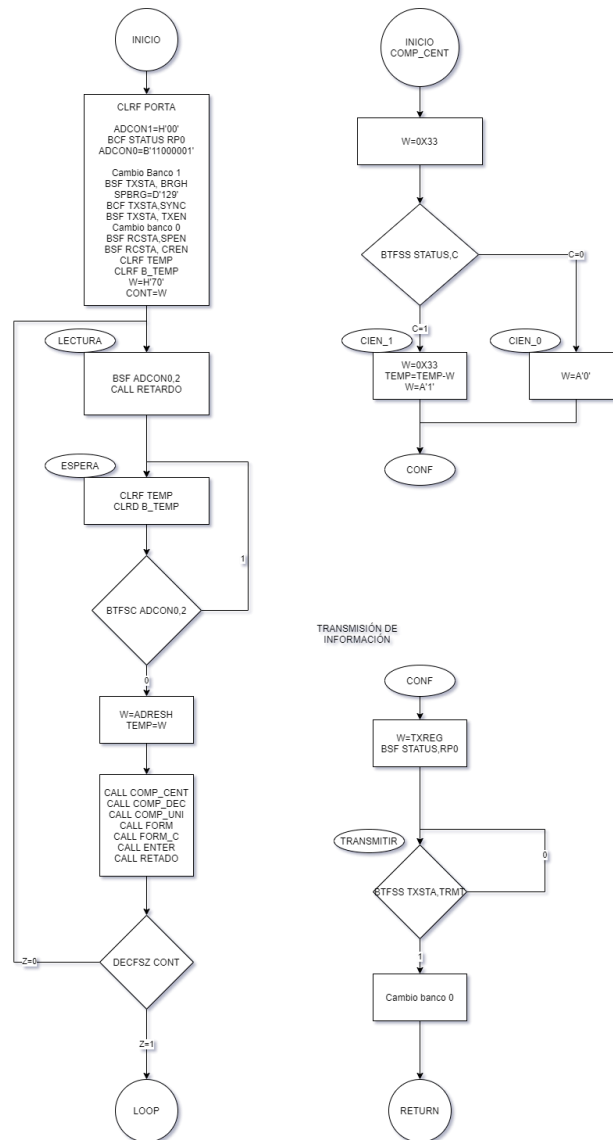
TRANSMITIR
    BTFSS    TXSTA,TRMT ;Revisión de transmisión exitosa
    GOTO     TRANSMITIR
    BCF      STATUS,RP0 ;Cambio al banco 0
    RETURN   ;Regreso al último CALL

RETARDO
    ; retardo de 20 microseg
    MOVLW    0x20
    MOVWF    valor1
uno
    DECFSZ   valor1
    GOTO     uno
    RETURN
END

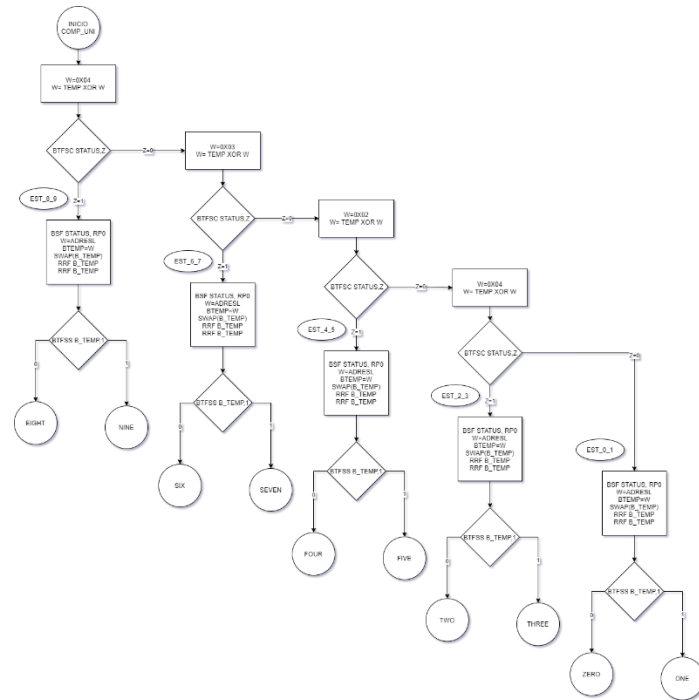
```

Código 6: Código de la transmisión de temperatura del LM35

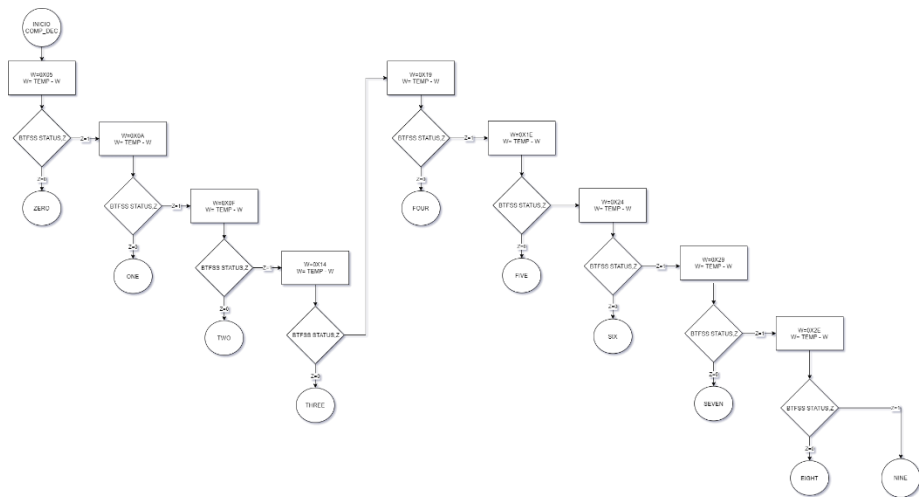
## Diagrama de flujo



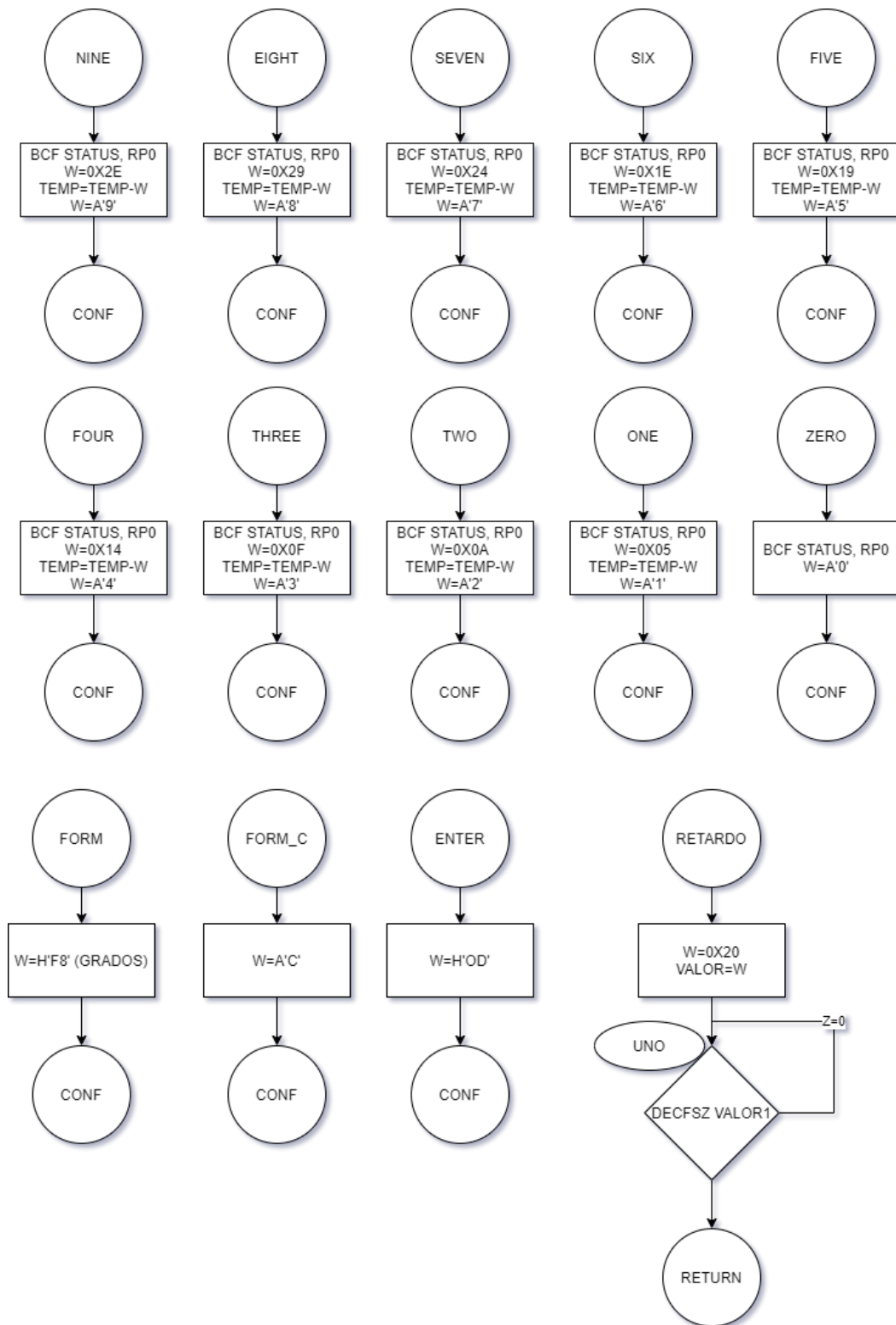
Modelo 6: Diagrama general, obtención de centenas y transmisión de información



Modelo 7: Obtención de decenas



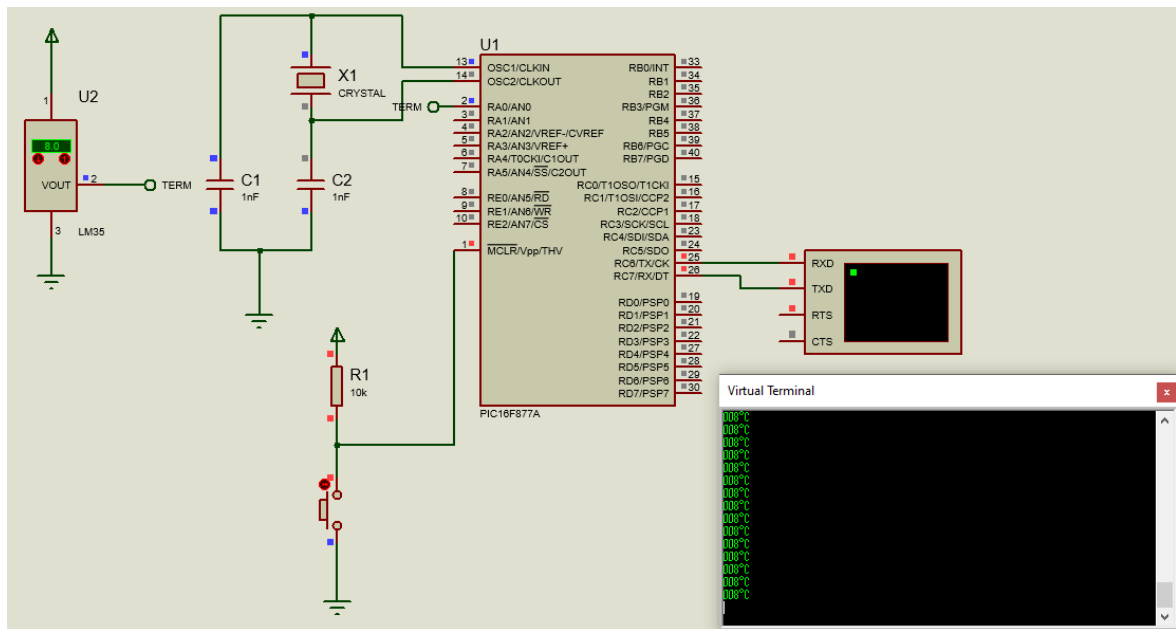
Modelo 8: Obtención de unidades



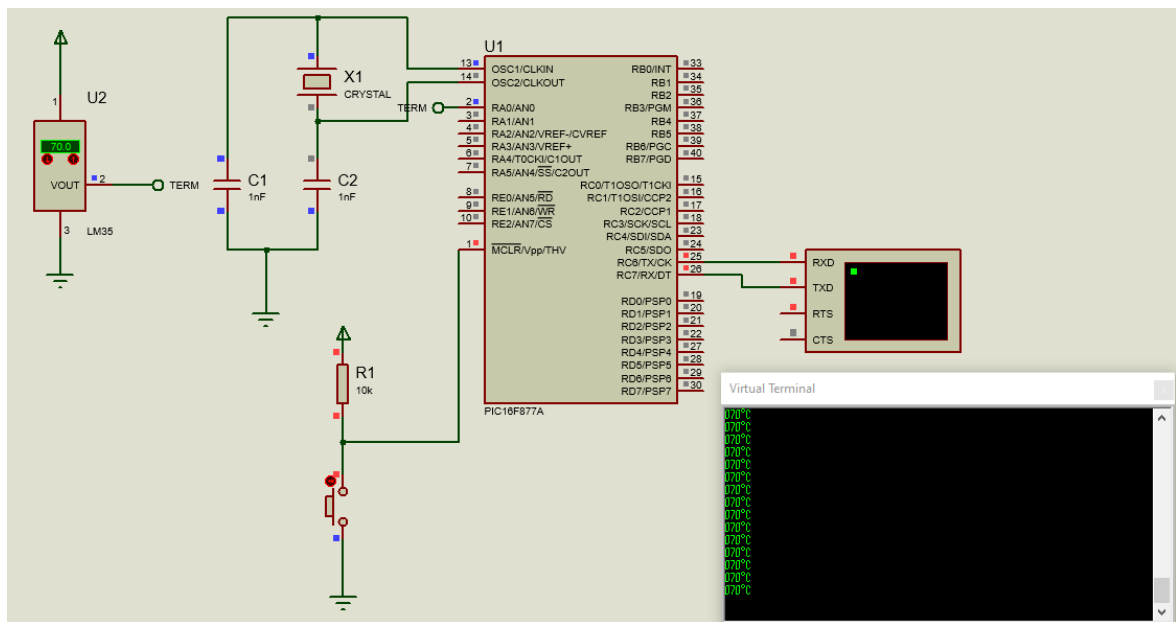
Modelo 9: Trasmisión de números, grados, C, enter y retardo



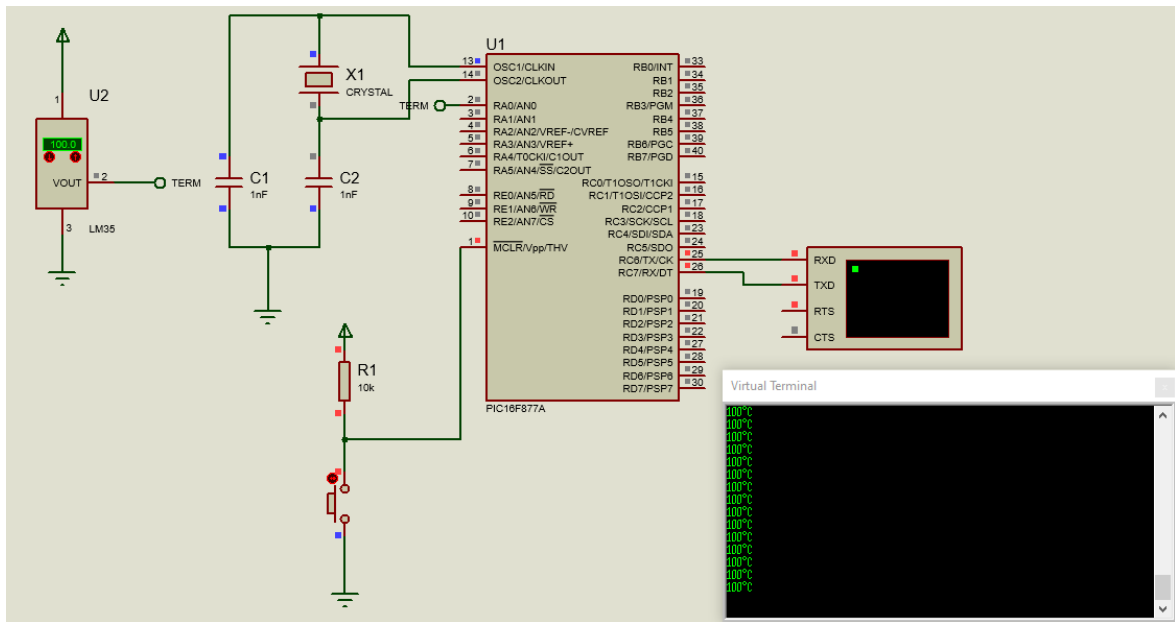
## Pruebas



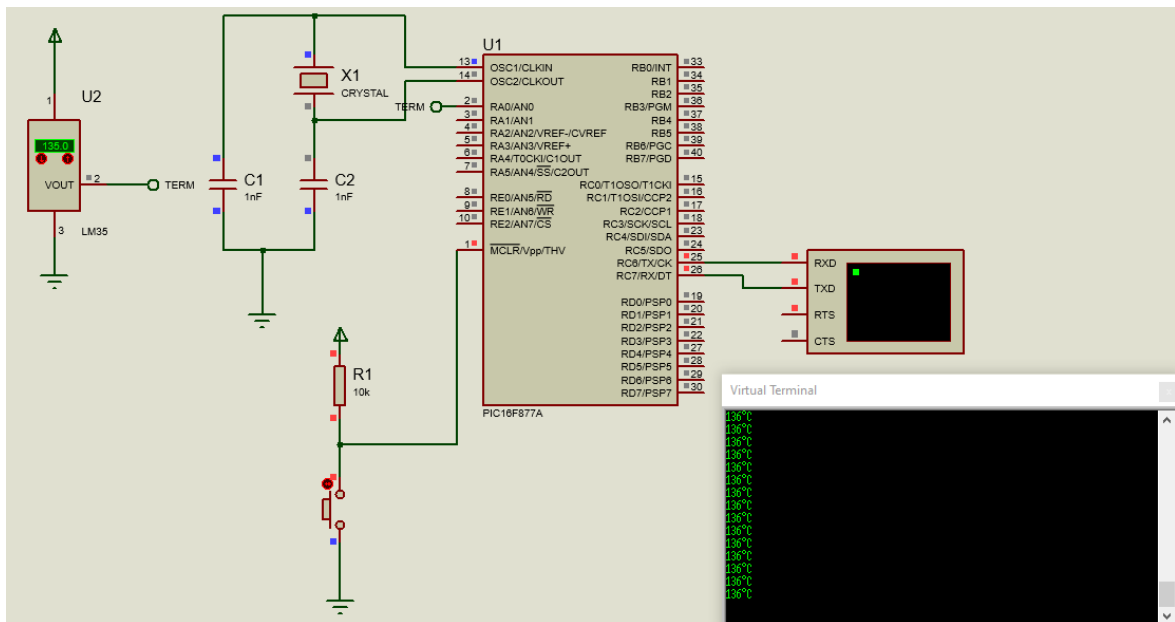
Pruebas 12: Temperatura de 8°C



Pruebas 13: Temperatura de 70°C



Pruebas 14: Temperatura de 100°C



Pruebas 15: Temperatura de 135°C

## Conclusiones

*Reza Chavarria Sergio Gabriel*

Para la práctica se comprendió el manejo de transmisión y recepción de datos de manera asíncrona y con esto se puede tener una interacción con el usuario de una maneja fluida a partir de la comunicación por medio de una terminal.

Además, con los ejercicios se puede dar a la ampliación del manejo de los conceptos obtenidos de las prácticas anteriores, como en el manejo de puertos de entrada y salida, manejo de elementos externos como motores y el manejo de señales analógicas por medio del convertidor A/D. Esto proporciona versatilidad en la realización de proyectos más complejos.

*Murrieta Villegas Alfonso*

En la presente práctica aprendimos como manejar de forma asíncrona la recepción de datos mediante ejercicios prácticos que con llevaron el manejo puertos tanto de entrada como de salida, fue el caso específico de señales analógicas como el manejo externo de motores de DC además del uso de estas mediante el convertidor A/D de nuestro microcontrolador.

Por último, empleamos una terminal virtual para el manejo del microcontrolador la cual nos dio una manera viable de poder escalar y resolver problemas más complejos en nuestros futuros proyectos o prácticas.

*Valdespino Mendieta Joaquin*

En esta práctica pudimos aplicar el manejo del flujo de datos de manera asíncrona, teniendo un aplicativo con funcionalidad manteniendo la interacción humano-maquina, como una forma de comunicación. A partir de ejercicios que integran conocimientos a lo largo del semestre, como puertos, manejo de entidades externas, señales analógicas, que ofrecen una automatización en el proceso de creación del proyecto, ademas de la capacidad de resolver problemas y aplicativos mas elaborados, como la terminal virtual.

