



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA

Laboratorio de Microcomputadoras

**Práctica 7.**  
**Puerto Serie SCI (Asíncrono)**

Grupo de Laboratorio: 4

Sanjuan Aldape Diana Paola  
Grupo de Teoría: 5

Esparza Fuentes Jorge Luis  
Grupo de Teoría: 4

Fecha de realización: 19 de abril del 2022



## DESARROLLO

1.- Escribir, comentar y ensamblar el siguiente código.

```
processor 16f877
include<pl6f877.inc>
ORG 0
GOTO INICIO
ORG 5
INICIO:
    BSF STATUS,RP0 ;Cambio al banco 1
    BCF STATUS,RP1
    CLRF PORTB      ;PORTB = 0
    CLRF TRISB      ;Configuro puerto B como salida
    BSF TXSTA,BRGH   ;BRGH = 1
    MOVLW D'129'     ;W = 129
    MOVWF SPBRG      ;SPBRG = W = 129
    BCF TXSTA,SYNC   ;SYNC = 0
    BSF TXSTA,TXEN    ;TXEN = 1
    BCF STATUS,RP0   ;Cambio al banco 0
    BSF RCSTA,SPEN    ;SPEN = 1
    BSF RCSTA,CREN    ;CREN = 1
RECIBE:
    BTFSS PIR1,RCIF ;If (RCIF = 1) {salta}
    GOTO RECIBE      ;else {RECIBE}
    MOVF RCREG,W      ;W = RCREG
    MOVWF PORTB       ;PORTD = RCREG
    MOVWF TXREG        ;TXREG = W
    BSF STATUS,RP0    ;Cambio al banco 1
TRASMITE:
    BTFSS TXSTA,TRMT;if (TRMT = 1) {Salta}
    GOTO TRASMITE     ;else {TRANSMITE}
    BCF STATUS,RP0    ;Cambio al banco 0
    GOTO RECIBE       ;Va a RECIBE
END
```

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

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

HOLA UNAM



```
| processor 16f877
| include<pl6f877.inc>
ORG 0
GOTO INICIO
ORG 5
INICIO:
    BSF STATUS,RP0 ;Cambio al banco 1
    BCF STATUS,RP1
    CLRF PORTB ;PORTB = 0
    CLRF TRISB ;Configuro puerto B como salida
    BSF TXSTA,BRGH ;BRGH = 1
    MOVLW D'129' ;W = 129
    MOVWF SPBRG ;SPBRG = W = 129
    BCF TXSTA,SYNC ;SYNC = 0
    BSF TXSTA,TXEN ;TXEN = 1
    BCF STATUS,RP0 ;Cambio al banco 0
    BSF RCSTA,SPEN ;SPEN = 1
    BSF RCSTA,CREN ;CREN = 1
    MOVLW A'H'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'O'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'L'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'A'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A' '
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'U'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'N'
    MOVWF TXREG
    CALL TRANSMITE

    MOVLW A'A'
    MOVWF TXREG
    CALL TRANSMITE
    MOVLW A'M'
    MOVWF TXREG
    CALL TRANSMITE
TRANSMITE:
    BSF STATUS,RP0 ;Cambio al banco 1
TRANSMITE1:
    BTFSS TXSTA,TRMT;if (TRMT = 1) {Salta}
    GOTO TRANSMITE1 ;else {TRANSMITE}
    BCF STATUS,RP0 ;Cambio al banco 0
    RETURN
END
```



4.- Realizar un programa que ejecute 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

```
processor 16f877a
include <pl6f877a.inc>
AUX EQU H'25'
ORG 0
GOTO INICIO
ORG 5
INICIO:
    BSF STATUS,RP0      ;Cambio al banco 1
    BCF STATUS,RP1
    BSF TXSTA,BRGH      ;BRGH = 1
    MOVLW D'129'
    MOVWF SPBRG         ;Se configura una tasa de 9600 buds
    BCF TXSTA,SYNC      ;Comunicación de tipo asíncrona
    BSF TXSTA,TXEN      ;Se habilita el transmisor
    MOVWF TRISE        ;Se configura el puerto B como salida
    CLRF PORTE         ;Se limpia el puerto B
    BCF STATUS,RP0      ;Cambio al banco 0
    BSF RCSTA,SPEN      ;Se habilita el puerto Serie
    BSF RCSTA,CREN      ;Se habilita a recepción continua
RECIBE
    BTFSS PIR1,RCIF     ;If (RCIF = 1) {salta}
    GOTO RECIBE         ;else {RECIBE}
    MOVF RCREG,W        ;W = RCREG
    MOVWF AUX
    MOVWF TXREG         ;Vemos en la terminal lo que se tecleo
    BSF STATUS,RP0      ;Cambio al banco 1
TRANSMITE:
    BTFSS TXSTA,TRMT    ;if (TRMT = 1) {Salta}
    GOTO TRANSMITE      ;else {TRANSMITE}
    BCF STATUS,RP0      ;Cambio a banco 0
    MOVLW A'1'          ;W <- 1
    XORWF AUX,W         ;W XOR AUX
    BTFSC STATUS,Z      ;if (Z = 0) {salta}
    GOTO CASO_1         ;else {CASO_1}
    MOVLW A'0'          ;W <- 1
    XORWF AUX,W         ;W XOR AUX
    BTFSC STATUS,Z      ;if (Z = 0) {salta}
    GOTO CASO_0         ;else {CASO_1}
    GOTO RECIBE         ;Repite
```



```
CASO_0
  BCF  PORTE,0    ;Mandamos 0 al puerto B
  GOTO RECIBE    ;Regresa a RECIBE
CASO_1
  BSF  PORTE,0    ;Mandamos 1 al puerto B
  GOTO RECIBE    ;Regresa a RECIBE

END
```

5.- Realizar un programa que ejecute el control indicado; la secuencia será reconocida cada que sea recibido el comando, usar retardos de  $\frac{1}{2}$  segundo entre cada estado generado:

<b>DATO</b> Puerto Serie	<b>ACCION</b> 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



Práctica 7. Puerto Serie SCI (Asíncrono)  
Fecha de realización: 19/04/2022

```
processor 16f877
include<pl6f877.inc>
AUX equ 0x23
ORG 0
GOTO INICIO
ORG 5
INICIO:
    BSF STATUS,RP0 ;Cambio al banco 1
    BCF STATUS,RP1
    CLRF PORTB ;PORTB = 0
    CLRF TRISB ;Configuro puerto B como salida
    BSF TXSTA,BRGH ;BRGH = 1
    MOVLW D'129' ;W = 129
    MOVWF SPBRG ;SPBRG = W = 129
    BCF TXSTA,SYNC ;SYNC = 0
    BSF TXSTA,TXEN ;TXEN = 1
    BCF STATUS,RP0 ;Cambio al banco 0
    BSF RCSTA,SPEN ;SPEN = 1
    BSF RCSTA,CREN ;CREN = 1
RECIBE:
    BTFSS PIR1,RCIF ;If (RCIF = 1) {salta}
    GOTO RECIBE ;else {RECIBE}
    MOVF RCREG,W ;W <= RCREG
    MOVWF AUX
    MOVWF TXREG ;Vemos en la terminal lo que se tecleo
    BSF STATUS,RP0 ;Cambio banco 1

TRANSMITE:
    MOVF PORTB
    MOVWF TXREG ;TXREG = W
    BSF STATUS,RP0 ;Cambio al banco 1
    BTFSS TXSTA,TRMT;if (TRMT = 1) {Salta}
    GOTO TRANSMITE ;else {TRANSMITE}
    BCF STATUS,RP0 ;Cambio al banco 0
    ;CASO 'D'
    SUBLW 0x44 ;ASCII D = 68
    BTFSC STATUS, Z
    GOTO CASO_D
    ;CASO 'd'
    SUBLW 0x64 ;ASCII d = 100
    BTFSC STATUS, Z
    GOTO CASO_D
    ;CASO 'I'
    SUBLW 0x49 ;ASCII I = 73
    BTFSC STATUS, Z
    GOTO CASO_I
    ;CASO 'i'
    SUBLW 0x69 ;ASCII i = 105
    BTFSC STATUS, Z
    GOTO CASO_I
    GOTO RECIBE ;Va a RECIBE
CASO_D:
    MOVLW 0x80
    MOVWF PORTB
    BCF STATUS,C ;Limpiamos
    CALL RETARDO
CORRIMIENTO_DER:
    RRF PORTB,1
    CALL RETARDO
    BTFSS STATUS, C
    GOTO CORRIMIENTO_DER
    GOTO RECIBE
CASO_I:
    MOVLW 0x01
    MOVWF PORTB
    BCF STATUS, C
    CALL RETARDO
CORRIMIENTO_IZQ:
    RLF PORTB,1
    CALL RETARDO
    BTFSS STATUS, C
    GOTO CORRIMIENTO_IZQ
    GOTO RECIBE
;Rutina que genera un retardo
RETARDO:
    MOVLW CTE1
    MOVWF VALOR1
TRES:
    MOVWF CTE2
    MOVWF VALOR2
DOS:
    MOVLW CTE3
    MOVWF VALOR3
UNO:
    DECFSZ VALOR3
    GOTO UNO
    DECFSZ VALOR2
    GOTO DOS
    DECFSZ VALOR1
    GOTO TRES
    RETURN
END
```



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:

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

Notas importantes:

- El dato que recibe es el código ASCII del carácter transmitido.
- Para vincularse con el dispositivo Bluetooth deberá comprobar su identificador.
- Considerar la ubicación de las señales de control y los valores encontrados en la práctica 5.

```
processor 16f877a
include <pl6f877a.inc>
AUX EQU H'31'
ORG 0
GOTO INICIO
ORG 5
INICIO:
    BSF STATUS,RP0           ;Cambio al banco 01
    BCF STATUS,RP1
    BSF TXSTA,BRGH           ;Ponemos en 1 la bandera BRGH, tasa de tranferencia alta
    MOVLW D'129'
    MOVWF SPBRG              ;Configuramos a una tasa de 9600 buds
    MOVLW h'0'
    MOVWF TRISE              ;Configura el puerto B como salida
    CLRF PORTE               ;Limpia los bits del PUERTO B
    BCF TXSTA,SYNC           ;Comunicación de tipo asincrona
    BSF TXSTA,TXEN           ;habilita el transmisor
    BCF STATUS,RP0           ;Cambio al banco 0
    BSF RCSTA,SPEN           ;Habilita el puerto Serie
    BSF RCSTA,CREN           ;Habilita a recepci3n continua
    CLRF PORTB
RECIBIR:
    BTFSS PIR1,RCIF
    GOTO RECIBIR
    MOVF RCREG,W
    MOVWF AUX
    MOVWF TXREG
    BSF STATUS,RP0           ;Cambio banco 1
```



```
TRASMITIR:
    BTFSS TXSTA,TRMT      ;Revisar si se transmite informacion
    GOTO TRASMITIR
    BCF STATUS,RPO       ;Cambio a banco 0
    MOVLW  A'S'
    XORWF  var,W          ;Comparar Si es 'S'
    BTFSC  STATUS,Z
    GOTO   PARO_PARO     ;Si es igual ve a ESTADO_1
    MOVLW  A'A'
    XORWF  var,W          ;Comparar Si es 'A'
    BTFSC  STATUS,Z
    GOTO   DERECH_DERECH ;Si es igual ve a ESTADO_2
    MOVLW  A'T'
    XORWF  var,W          ;Comparar Si es 'T'
    BTFSC  STATUS,Z
    GOTO   IZQ_IZQ       ;Si es igual ve a ESTADO_3
    MOVLW  A'D'
    XORWF  var,W          ;Comparar Si es 'D'
    BTFSC  STATUS,Z
    GOTO   DER_IZQ       ;Si es igual ve a ESTADO_4
    MOVLW  A'I'
    XORWF  var,W          ;Comparar Si es 'I'
    BTFSC  STATUS,Z
    GOTO   IZQ_DER       ;Si es igual ve a ESTADO_5
    GOTO  RECIBIR

PARO_PARO:                ;PARO PARO
    MOVLW  0X00
    MOVWF  PORTB
    GOTO  RECIBIR

DERECH_DERECH:            ;DERECHA DERECHA
    MOVLW  0X05
    MOVWF  PORTB
    GOTO  RECIBIR

IZQ_IZQ:                  ;IZQUIERDA IZQUIERDA
    MOVLW  0X0A
    MOVWF  PORTB
    GOTO  RECIBIR

DER_IZQ:                  ;DERECHA IZQUIERDA
    MOVLW  0X09
    MOVWF  PORTB
    GOTO  RECIBIR

IZQ_DER:                  ;IZQUIERDA DERECHA
    MOVLW  0X06
    MOVWF  PORTB
    GOTO  RECIBIR
END
```

## CONCLUSIÓN

- Sanjuan Aldape Diana Paola

En esta práctica realizamos programas haciendo uso del puerto serie, utilizando registros de transmisión y recepción de datos. Para esto fue necesario hacer dos etiquetas, una para la transmisión y otra para la recepción. Depende de lo que se haya recibido la acción que se tomará en el motor posteriormente.

- Esparza Fuentes Jorge Luis





Se analizó e implementó la comunicación serial asíncrona a partir del puerto serie tanto del pic como del PC y el celular por vía bluetooth. Aunado a esto, se implementó el uso de un monitor serial para poder entablar comunicación desde el PC con el microcontrolador PIC, de esta forma se logró obtener los resultados esperados por esta práctica.