

Microcontroladores

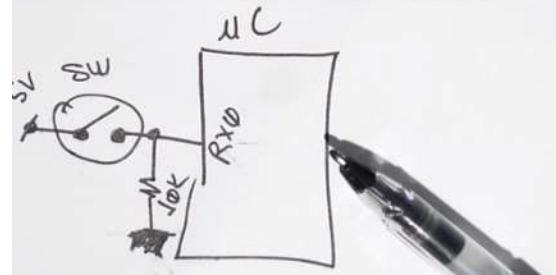
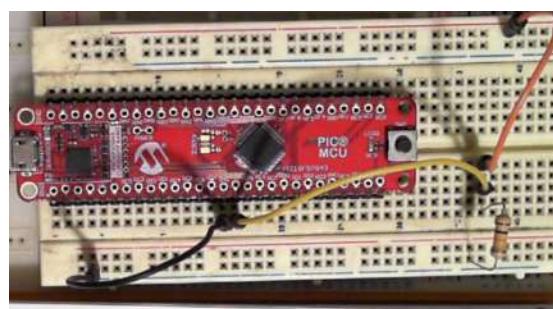
Semana 5

Semestre 2024-2
Por Kalun José Lau Gan

1

Preguntas previas

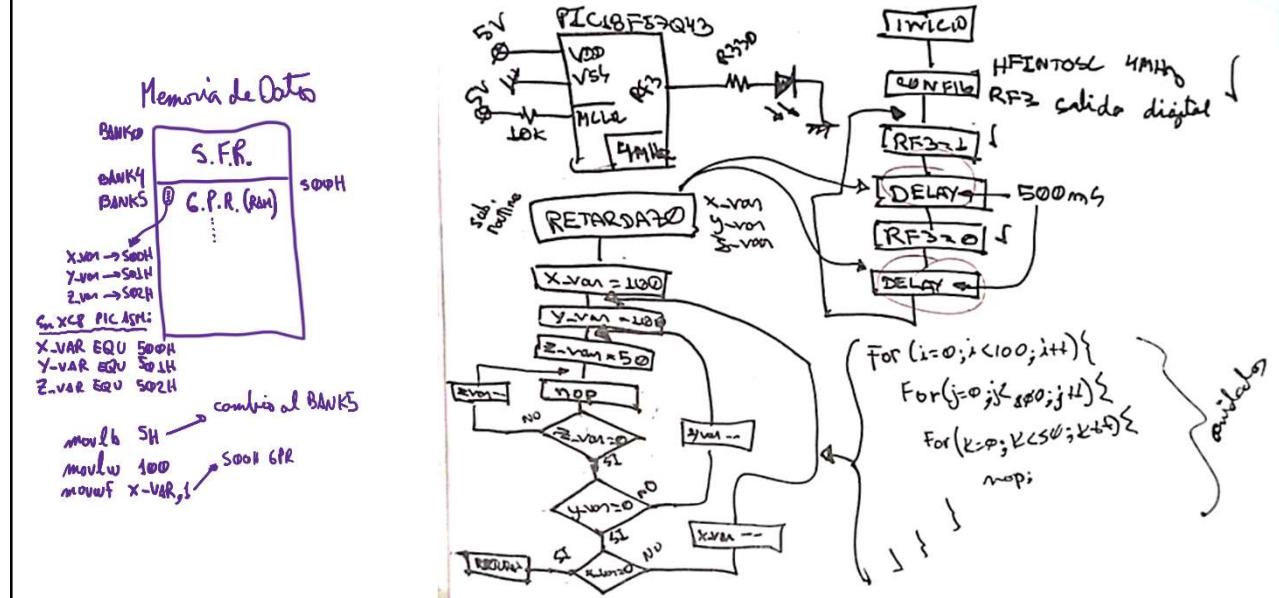
- ¿Reemplazar un switch con solamente un cable jumper?



2

Preguntas previas

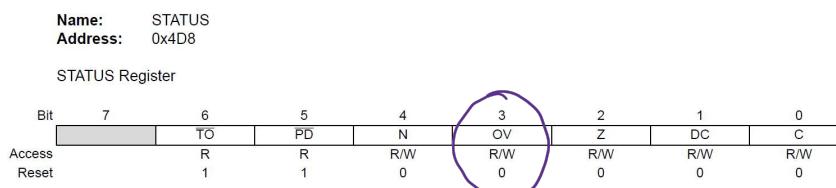
- ¿Retardo prolongado para prender y apagar un LED?



3

Preguntas previas

- ¿Cómo se trabaja con el bit de overflow?



cuenta EQU 500H

```

  inicio:
    movlb 5H
    movlw 250
    movwf cuenta, 1
    incf cuenta, 1, 1
    movlb 4H
    btfss STATUS, 3, 1
    bra aunno
    return
  aunno:
    movlb 5H
    bra bucle
  
```

numero EQU 508H

```

    moulw 180
    movwf numero, 1
    addwf numero, 1
    summa 1º operando
    2º operando
    resultado
    0x00
    0x00
  
```

mem.Datos - 8 bits (0~255)

se llevanta los bandos
C y OVF

4

Preguntas previas

- ¿Qué significa BRA \$-2 ó GOTO \$-2?
 - Indica salto a una posición de 2 bytes anteriores (una instrucción simple)

```

        movlw 55H           ;01010101B
        movwf LATB, 1
        comf LATB, 1, 1    ;10101010B
        bra $-6

bucle:   ---
retardon: ---          ;subrutina
--- 
--- 
--- 
--- 
--- 
return

```

```

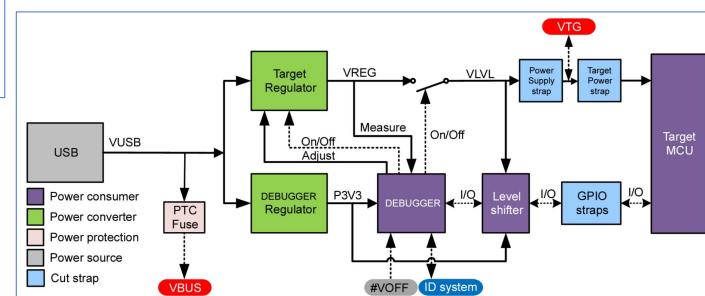
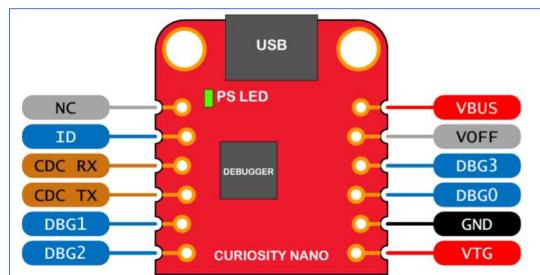
inicio:
btfs PORTB, 4, 1
bra $-2
btg LATF, 3, 1
btfsc PORTB, 4, 1
bra $-2
bra inicio

```

5

Preguntas previas

- Para alimentar el protoboard desde el Curiosity Nano. ¿Se conectan VBUS y VTG a la línea roja?
 - No, ni lo intentes!



6

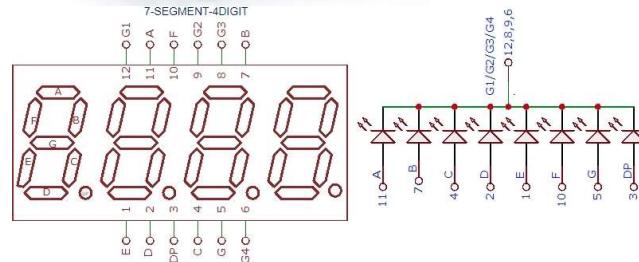
Agenda

- Multiplexación en displays de siete segmentos
- El display multiplexado de 4 dígitos de siete segmentos
- Ejemplos de aplicaciones con multiplexación de displays de siete segmentos

7

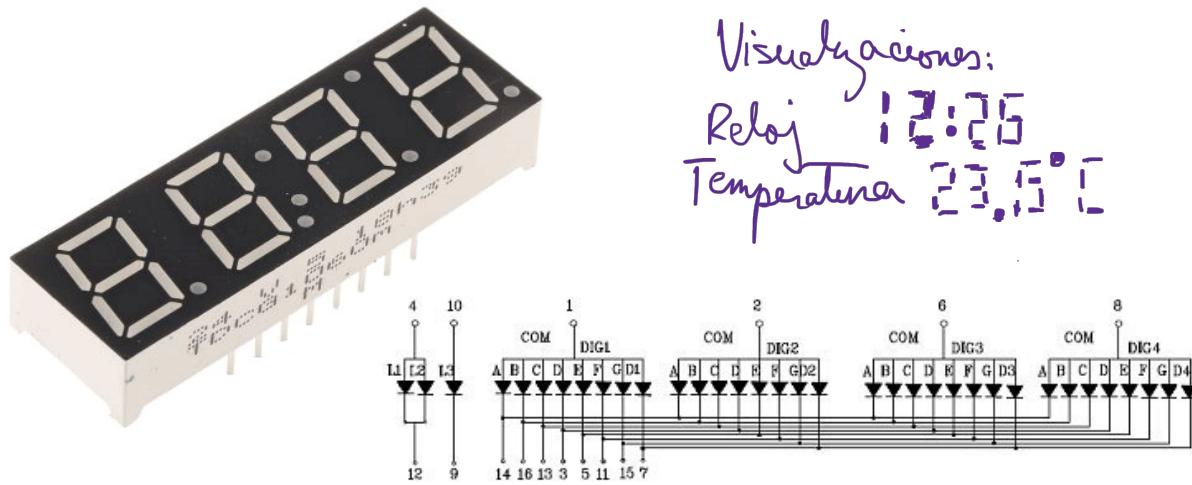
El display de cuatro dígitos de siete segmentos multiplexado:

- Verificar si el display es de cátodo común o ánodo común empleando el multímetro.
- Para la multiplexación se requerirá del uso de transistores para evitar que los pines del microcontrolador trabajen con corrientes excesivas.



8

Ejemplo: display dedicado para aplicaciones de reloj y temperatura



9

Antecesores a los displays de siete segmentos

- Tubos Nixie:



10

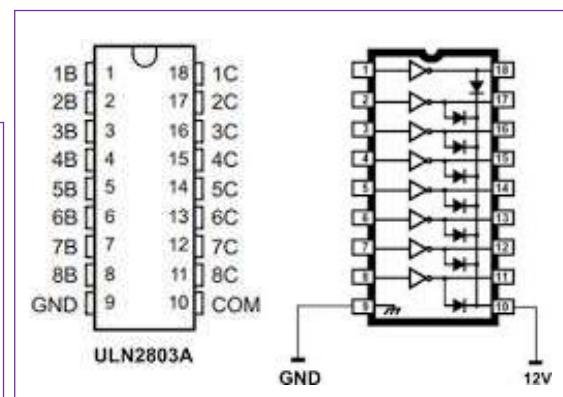
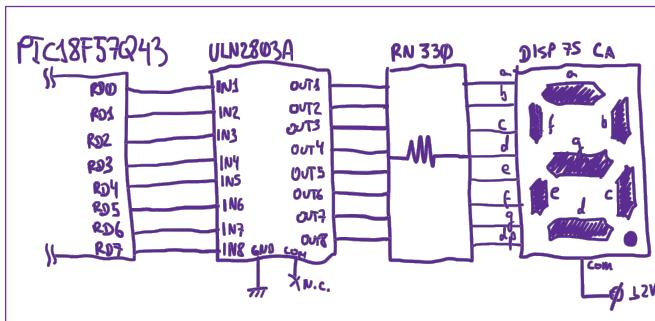
Multiplexación en displays de siete segmentos

- Los segmentos de cada dígito se encuentran conectados (Sa-Dig1 esta conectado con Sa-Dig2, con Sa-Dig3 y con Sa-Dig4, y demás segmentos)
- Se visualiza un dígito a la vez en el display multiplexado y en forma cíclica. Un ciclo de refresco (refresh rate) involucra haber visualizado la información en todos los dígitos que componen el display.
- Por medio de un refresco a alta frecuencia (>50Hz) podremos ver los cuatro displays encendidos al mismo tiempo por el efecto de “persistencia visual”
- Si se detiene la multiplexación, se detiene la visualización en el display.
- Reducimos la cantidad de E/S empleados en el microcontrolador (7 salidas para los segmentos excluyendo el punto decimal y 4 salidas para la selección del dígito)

11

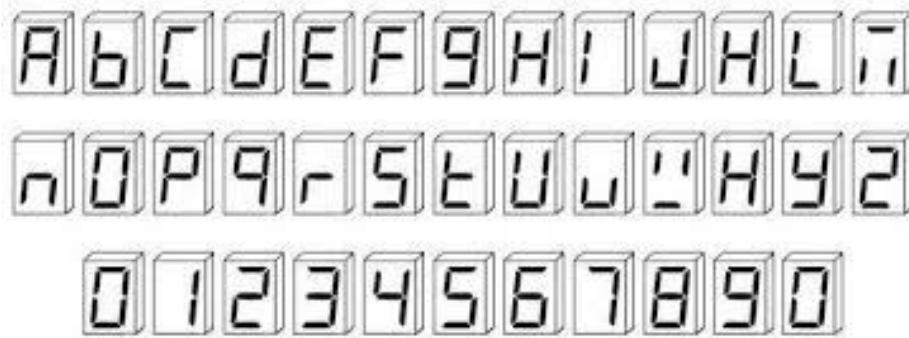
Arreglo Darlington para displays de mas de 5V

- En caso de hacer visualización con displays de mayor voltaje, se recomienda emplear el IC ULN2803A, solo para casos de ánodo común.



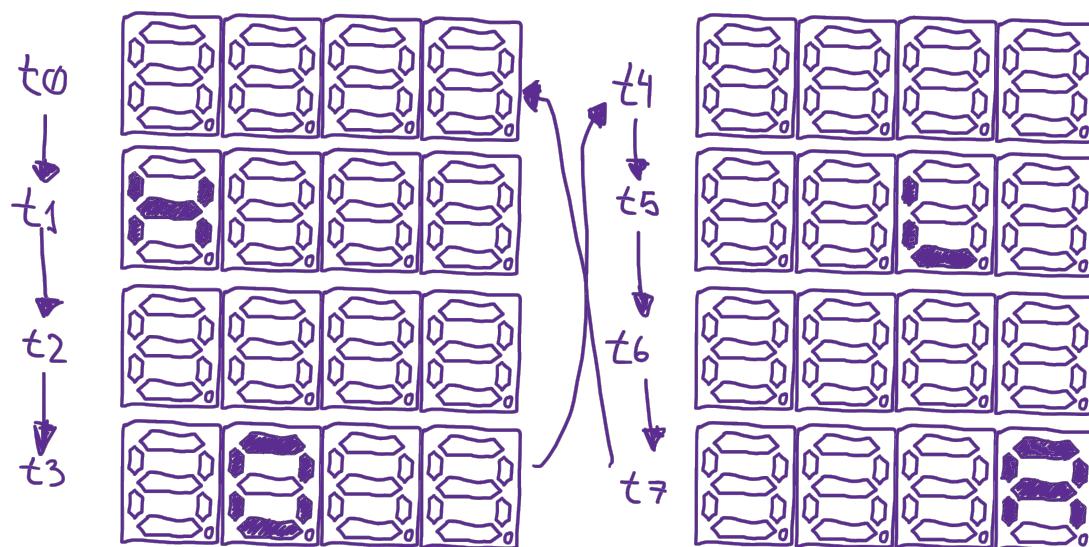
12

Possible visualización de las letras en el display de siete segmentos



13

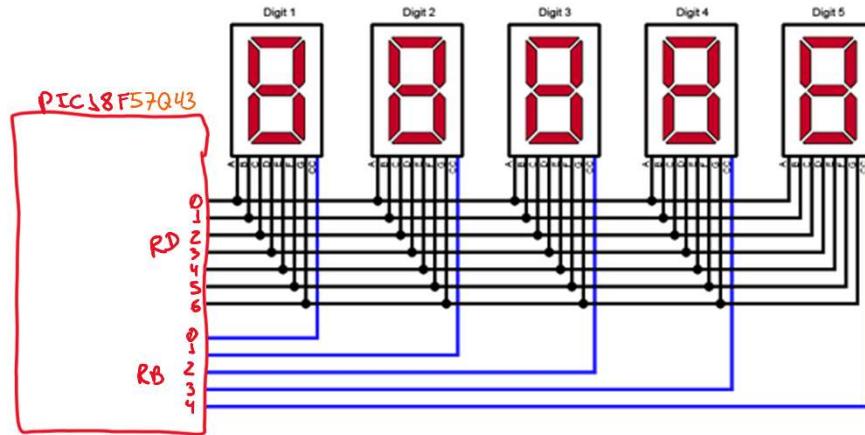
Funcionamiento de la multiplexación: Visualización de la palabra “HOLA”



14

Caso: Malos diseños electrónicos en la multiplexación de displays de siete segmentos:

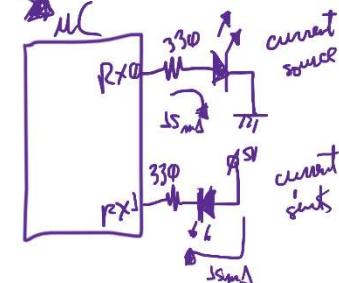
- Al igual que en displays de siete segmentos individuales, se deben de colocar resistencias en cada uno de los segmentos para limitar la corriente que pasa por ellos.
- Se recomienda no conectar los habilitadores de frente a los I/O del microcontrolador ya que dichos I/O no soportan entregar/recepcionar corrientes altas.



15

Consideraciones eléctricas al usar el microcontrolador PIC18F57Q43

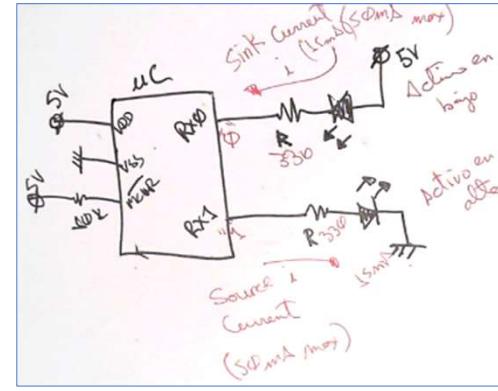
En la hoja técnica: $\pm 50\text{mA}$ máximo por I/O
 total el PIC18F57Q43 tiene 44 I/O
~~¿Total $\pm 2200\text{mA}$?~~ \rightarrow ¿Será cierto?
 $V_{DD} \rightarrow$ máx permite 350mA



16

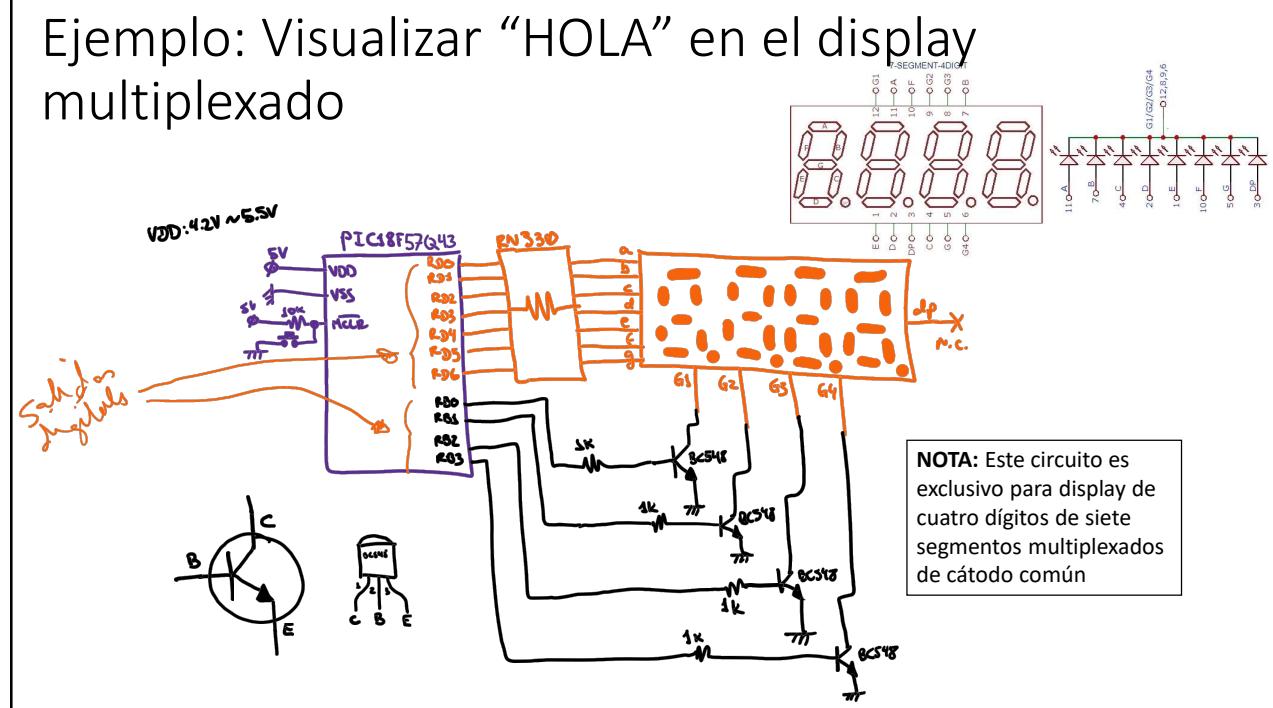
¿Corriente máxima por I/O en el PIC18F57Q43?

Absolute Maximum Ratings ⁽¹⁾	
Parameter	Rating
Ambient temperature under bias	-40°C to +125°C
Storage temperature	-65°C to +150°C
Voltage on pins with respect to V _{SS}	-0.3V to +6.5V
• on V _{DD} pin:	-0.3V to +9.0V
• on MCLR pin:	-0.3V to (V _{DD} + 0.3V)
Maximum current ⁽¹⁾	
• on V _{SS} pin	-40°C ≤ T _A ≤ +85°C 85°C < T _A ≤ +125°C 350 mA 120 mA
• on V _{DD} pin (28-pin devices)	-40°C ≤ T _A ≤ +85°C 85°C < T _A ≤ +125°C 250 mA 85 mA
• on V _{DD} pin (40-pin devices)	-40°C ≤ T _A ≤ +85°C 85°C < T _A ≤ +125°C 350 mA 120 mA
• on any standard I/O pin	±50 mA
Clamp current, I _K (V _{PIN} < 0 or V _{PIN} > V _{DD})	±20 mA
Total power dissipation ⁽²⁾	800 mW



17

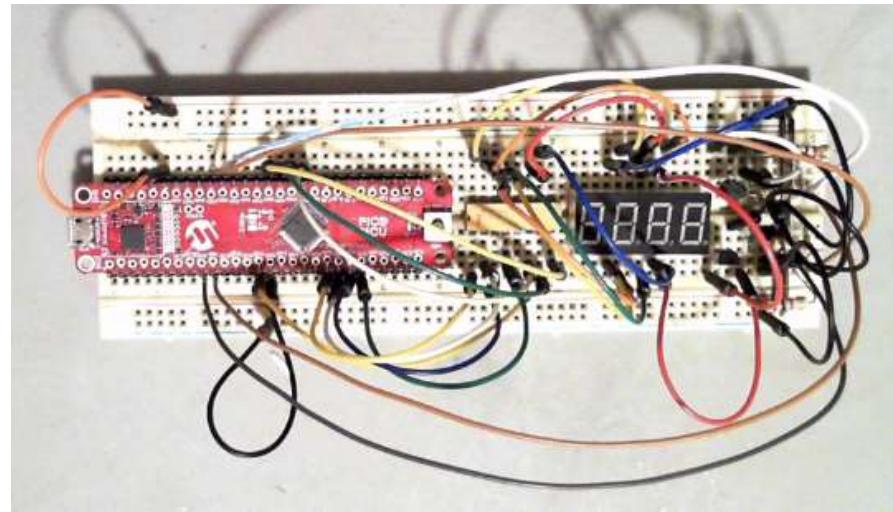
Ejemplo: Visualizar "HOLA" en el display multiplexado



18

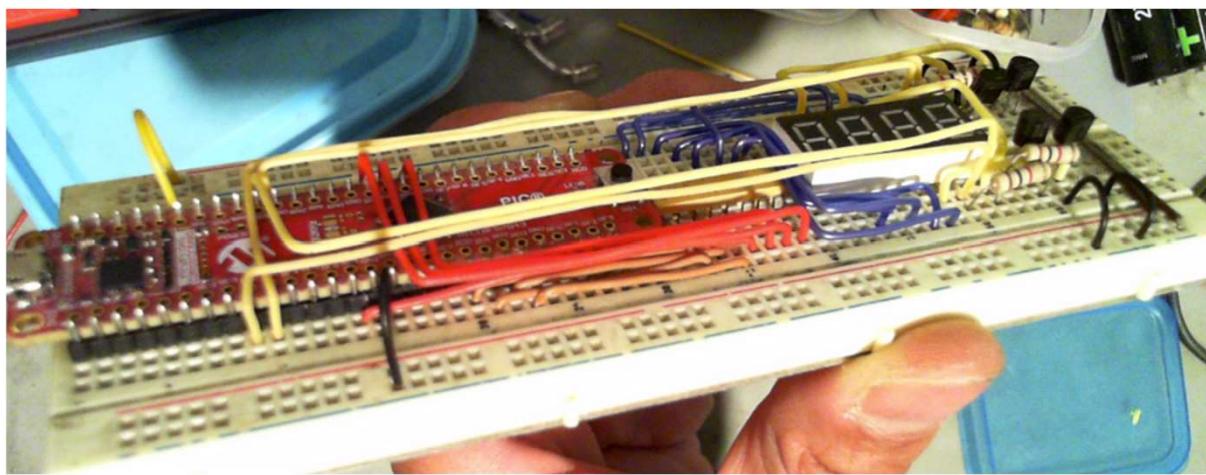
Circuito implementado con display de cuatro dígitos de siete segmentos cátodo común multiplexados

Circuito implementado



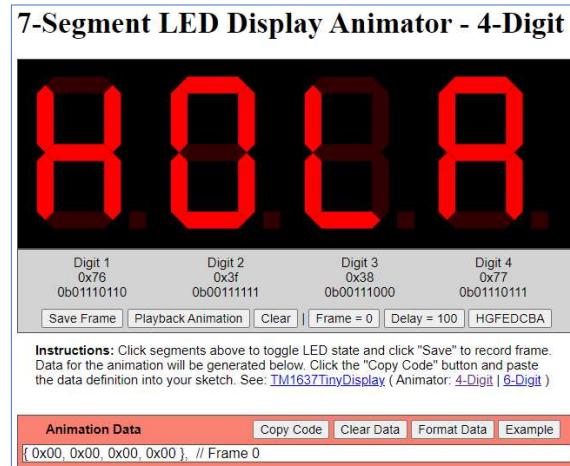
19

Implementación 2024-1



20

Circuito implementado con display de cuatro dígitos de siete segmentos cátodo común multiplexados

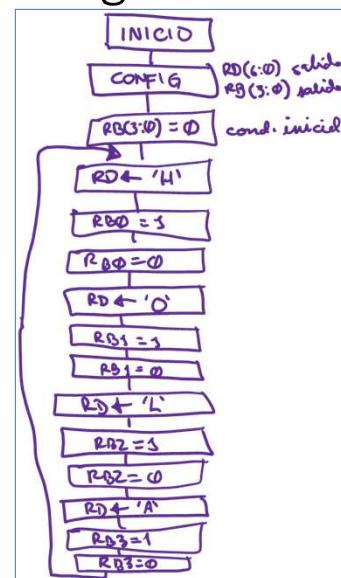


<https://jasonacox.github.io/TM1637TinyDisplay/examples/7-segment-animator.html>

21

Algoritmo para la multiplexación de los dígitos del display de siete segmentos

- “visualización dinámica”
- Se debe de enviar el dato cuando se encuentren deshabilitados los dígitos (para no tener “ghosting”)
- Luego de enviar el dato se hará la habilitación del dígito respectivo, antes de cargar otro dato se deberá de deshabilitar el dígito activo.



22

Código en XC8 PIC Assembler

```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6      ORG 000000H
7      bra configuro
8
9      ORG 000100H
10 configuro:
11     movlb 0H
12     movlw 60H
13     movwf OSCCON1, 1
14     movlw 02H
15     movwf OSCFRQ, 1
16     movlw 40H
17     movwf OSCEN, 1
18     movlb 4H
19     movlw 80H
20     movwf TRISD, 1      ;RD6-RD0 como salidas
21     movwf ANSEL0, 1      ;RD6-RD0 como digitales
22     movlw OF0H
23     movwf TRISB, 1      ;RB3-RB0 como salidas
24     movwf ANSELB, 1      ;RB3-RB0 como digitales
25     clrf LATB, 1        ;Condicion inicial: habilitadores todos en cero
26
27 inicio:
28     movlw 76H
29     movwf LATD, 1        ;cargando H en RD
30     bsf LATB, 0, 1       ;habilito digito 0
31     call nopes
32     bcf LATB, 0, 1       ;deshabilito digito 0
33     movlw 3FH
34     movwf LATD, 1        ;cargando O en RD
35     bsf LATB, 1, 1       ;habilito digito 1
36     call nopes
37     bcf LATB, 1, 1       ;deshabilito digito 1
38     movlw 38H
39     movwf LATD, 1        ;cargando L en RD
40     bsf LATB, 2, 1       ;habilito digito 2
41     call nopes
42     bcf LATB, 2, 1       ;deshabilito digito 2
43     movlw 77H
44     movwf LATD, 1        ;cargando A en RD
45     bsf LATB, 3, 1       ;habilito digito 3
46     call nopes
47     bcf LATB, 3, 1       ;deshabilito digito 3
48     bra inicio
49 nopes:
50     nop
51     nop
52     nop
53     nop
54     nop
55     nop
56     return
57
58 end upcino

```

23

Código 2024-1

- Visualización de “HOLA”
- Frecuencia 8MHz al CPU
- Se evidencia “ghosting” por lo que se requerirá bajar la frecuencia de trabajo del CPU

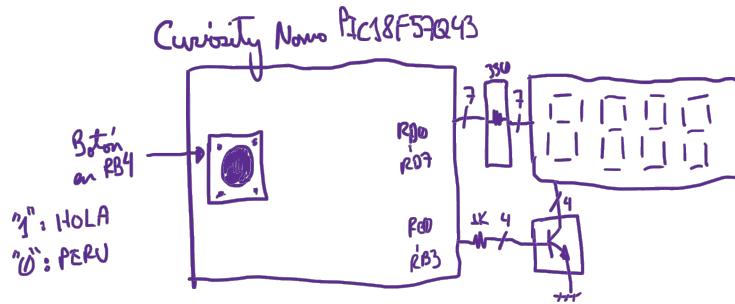
```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSECT upcino, class=CODE, reloc=2, abs
5  upcino:
6      ORG 000000H
7      bra configuro
8
9      ORG 000100H
10 configuro:
11     movlb 0H
12     movlw 60H
13     movwf OSCCON1, 1
14     movlw 03H
15     movwf OSCFRQ, 1
16     movlw 40H
17     movwf OSCEN, 1
18     movlb 4H
19     clrf TRISD, 1
20     clrf ANSEL0, 1
21     movlw OF0H
22     movwf TRISB, 1
23     movwf ANSELB, 1
24     clrf LATB, 1
25
26 inicio:
27     movlw 76H
28     movwf LATD, 1
29     bsf LATB, 0, 1
30     call nopx16
31     bcf LATB, 0, 1
32     movlw 3FH
33     movwf LATD, 1
34     bsf LATB, 1, 1
35     call nopx16
36     bcf LATB, 1, 1
37     movlw 38H
38     movwf LATD, 1
39     bsf LATB, 2, 1
40     call nopx16
41     bcf LATB, 2, 1
42     movlw 77H
43     movwf LATD, 1
44     bsf LATB, 3, 1
45     call nopx16
46     bcf LATB, 3, 1
47     bra inicio
48 nopx16:
49     nop
50     nop
51     nop
52     nop
53     nop
54     nop
55     nop
56     nop
57     nop
58     nop
59     nop
60     nop
61     nop
62     nop
63     nop
64     nop
65     nop
66     return
67
68 end upcino

```

24

¿Cómo hacer para que con el pulsador integrado en RB4 cambie de HOLA a PERU?



25

Código 2024-1

- El botón integrado del Curiosity Nano (RB4) se está utilizando para cambiar la visualización entre la palabra HOLA y PERU.
- Tener en cuenta que se está empleando dos sectores de la memoria de programa para almacenar las palabras a visualizar, y se utiliza el puntero de tabla (TBLPTR) para poder acceder a los datos de decodificación

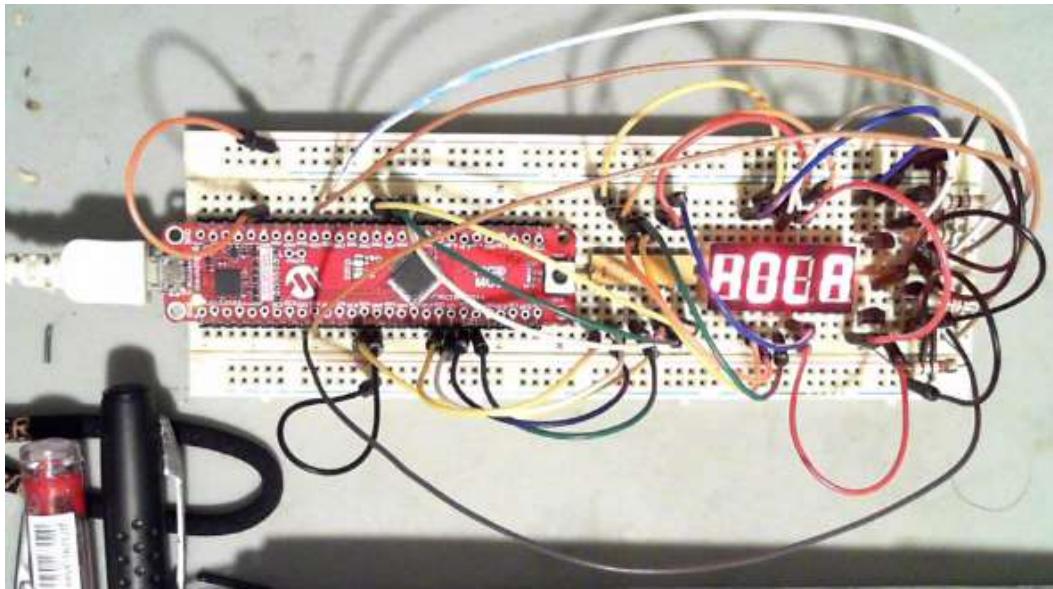
```

1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT upcino, class=CODE, reloc=2, abs
5      upcino:
6          ORG 000000H
7          bra configuro
8
9          ORG 000500H
10         tabla_mensaje1: DB 73H, 79H, 50H, 3EH ;PERU
11
12         ORG 000600H
13         tabla_mensaje2: DB 76H, 3FH, 38H, 77H ;HOLA
14
15         ORG 000100H
16         configuro:
17             movlw 0H
18             movlw 60H
19             movwf OSCCON1, 1
20             movlw 00H
21             movwf OSCFRC, 1
22             movlw 40H
23             movwf OSCEN, 1
24             movlw 4H
25             clrf TRISD, 1
26             clrf ANSELD, 1
27             movlw 0FOH
28             movwf TRISB, 1
29             movlw 0EOH
30             movwf ANSELB, 1
31             bsf WPUB, 4, 1
32             clrf LATB, 1
33             clrf TBLPTRU, 1
34
35         inicio:
36             call pregunta_btn
37             clrf TBLPTRL, 1
38             TBLRD+
39             movff TABLAT, LATD
40             bsf LATB, 0, 1
41             call nopx16
42             bcf LATB, 0, 1
43             TBLRD+
44             movff TABLAT, LATD
45             bsf LATB, 1, 1
46             call nopx16
47             bcf LATB, 1, 1
48             TBLRD+
49             movff TABLAT, LATD
50             bsf LATB, 2, 1
51             call nopx16
52             bcf LATB, 2, 1
53             TBLRD+
54             movff TABLAT, LATD
55             bsf LATB, 3, 1
56             call nopx16
57             bcf LATB, 3, 1
58             bra inicio
59
60         pregunta_btn:
61             btfsc PORTB, 4, 1
62             bra msg_hola
63             bra msg_peru
64
65         msg_hola:
66             movlw 06H
67             movwf TBLPTRH, 1
68
69         return
70
71         msg_peru:
72             movlw 05H
73             movwf TBLPTRH, 1
74             return
75
76         nopx16:
77             nop
78             nop
79             nop
80             nop
81             nop
82             nop
83             nop
84             nop
85             nop
86             nop
87             nop
88             nop
89             nop
90             nop
91             nop
92             return
93
94         end upcino

```

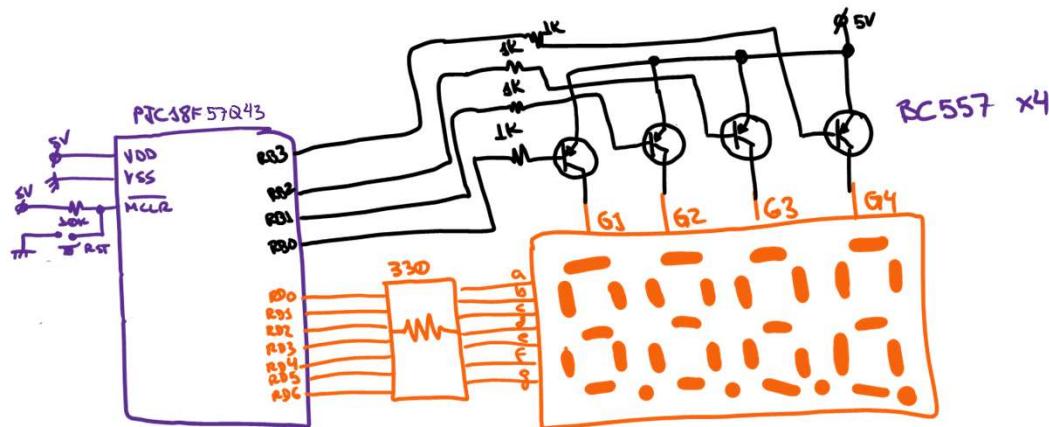
26

Pruebas en circuito:



27

Observación: Circuito para display multiplexado de ánodo común



28

Código mejorado empleando TBLPTR (XC8 PIC Assembler)

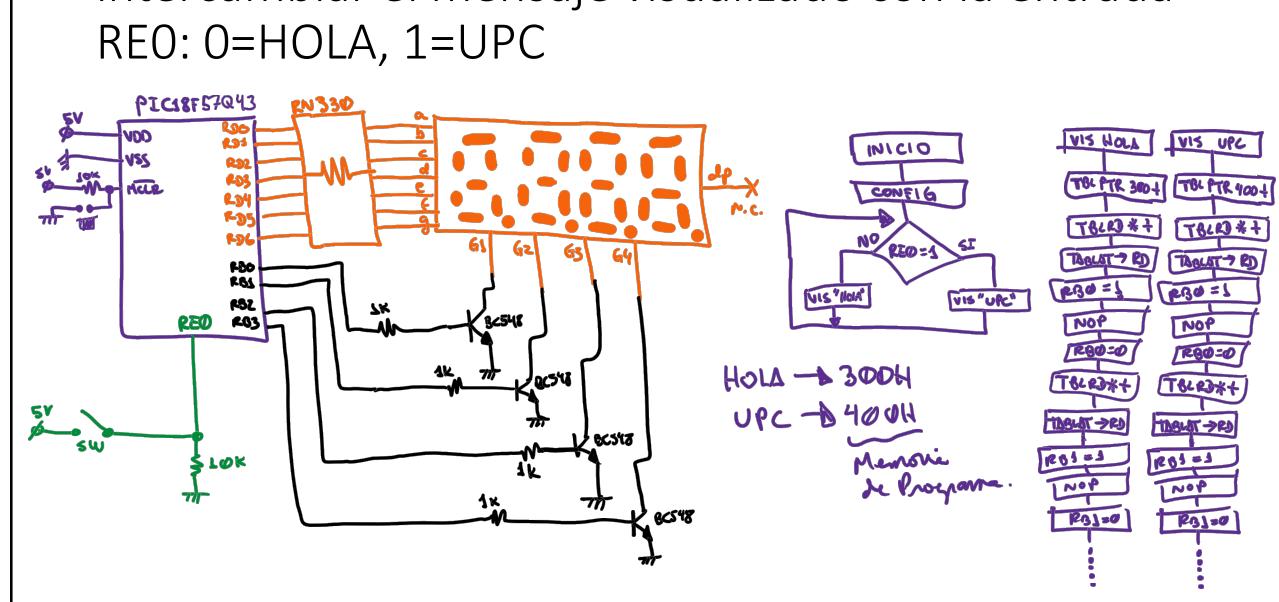
```

1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT upcino, class=CODE, reloc=2, abs
5      upcino:
6          ORG 000000H
7          bra configuro
8
9          ORG 000700H
10         mensaje: DB 76H, 3FH, 38H, 77H
11
12         ORG 000100H
13 configuro:
14         movlb 0H
15         movlw 60H
16         movwf OSCCON1, 1
17         movlw 02H
18         movwf OSCFRQ, 1
19         movlw 40H
20         movwf OSCEN, 1
21         movlb 4H
22         movlw 80H
23         movwf TRISD, 1 ;RD6-RD0 como salidas
24         movwf ANSELD, 1 ;RD6-RD0 como digitales
25         movlw OFOH
26         movwf TRISB, 1 ;RB3-RB0 como salidas
27         movwf ANSELB, 1 ;RB3-RB0 como digitales
28         clrf LATB, 1 ;Condicion inicial: habilitadores todos en cero
29
30         inicio:           ;TBLPTR esta apuntando a la direccion 700H de mem prog
31             clrf TBLPTRU, 1
32             movlw 07H
33             movwf TBLPTRH, 1
34             clrf TBLPTRL, 1 ;Lee el contenido de TBLPTR (700H) y luego lo incrementa
35             TBLRD*+
36             movff TABLAT, LATD ;Mueve el contenido de TABLAT hacia LATD
37             bsf LATB, 0, 1 ;habilito digito 0
38             call nopes
39             bcf LATB, 0, 1 ;deshabilito digito 0
40             TBLRD*+
41             movff TABLAT, LATD ;Lee el contenido de TBLPTR (701H) y luego lo incrementa
42             bsf LATB, 1, 1 ;Mueve el contenido de TABLAT hacia LATD
43             call nopes
44             bcf LATB, 1, 1 ;deshabilito digito 1
45             TBLRD*+
46             movff TABLAT, LATD ;Lee el contenido de TBLPTR (702H) y luego lo incrementa
47             bsf LATB, 2, 1 ;Mueve el contenido de TABLAT hacia LATD
48             call nopes
49             bcf LATB, 2, 1 ;deshabilito digito 2
50             TBLRD*+
51             movff TABLAT, LATD ;Lee el contenido de TBLPTR (703H) y luego lo incrementa
52             bsf LATB, 3, 1 ;Mueve el contenido de TABLAT hacia LATD
53             call nopes
54             bcf LATB, 3, 1 ;deshabilito digito 3
55             bra inicio
56
57             nopes:           ;Se ha modificado para
58                 nop           que el programa obtenga
59                 nop           los datos a visualizar
60                 nop           desde la memoria de
61                 nop           programa empleando el
62                 nop           TBLPTR
63             return
64
65             end upcino

```

29

Modificando el ejemplo para que pueda intercambiar el mensaje visualizado con la entrada RE0: 0=HOLA, 1=UPC



30

Modificando el ejemplo para que pueda intercambiar el mensaje visualizado con la entrada RE0: 0=HOLA, 1=UPC

```

5      #include "cabecera.inc"
6
7      PSECT multiplexacion, class=CODE, reloc=2, abs
8
9      ORG 00300H
10     ;          H   O   L   A
11     mensaje1: db    76H, 3FH, 38H, 77H
12     ;          O   R   G   00400H
13     ;          U   P   C
14     mensaje2: db    00H, 3EH, 73H, 39H
15
16     ORG 00000H
17     multiplexacion: goto configuracion
18
19     ORG 00020H
20     configuracion:
21     movlw 0x80
22     movwf TRISD      ;RD6-RD0 como salidas
23     movlw 0xF0
24     movwf TRISB      ;RB3-RB0 como salidas
25     clrf LATB        ;Condicion inicial de los habilitadores
26     movlw 0x0F
27     movwf ADCON1      ;Para que RE0 sea entrada digital
28
29     inicio:
30     btfss PORTE, 0    ;Pregunta si RE0=1
31     goto msg_hola
32     goto msg_upc
33
34     msg_hola:
35     movlw 03H
36     movwf TBLPTRH
37     clrf TBLPTRL
38     goto multiplex
39
40     msg_upc:
41     movlw 04H
42     movwf TBLPTRH
43     clrf TBLPTRL
44     goto multiplex
45
46     multiplex:
47     TBLRD*+
48     movff TABLAT, LATD
49     bsf LATB, 0
50     nop
51     bcf LATB, 0
52     TBLRD*+
53     movff TABLAT, LATD
54     bsf LATB, 1
55     nop
56     bcf LATB, 1
57     TBLRD*+
58     movff TABLAT, LATD
59     bsf LATB, 2
60     nop
61     bcf LATB, 2
62     TBLRD*
63     movff TABLAT, LATD
64     bsf LATB, 3
65     nop
66     bcf LATB, 3
67     goto inicio
68
69     end multiplexacion

```

31

Modificando el ejemplo para que pueda intercambiar el mensaje con la entrada RB4: 0=HOLA, 1=PATO

```

1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT upcino, class=CODE, reloc=2, abs
5      upcino:
6      ORG 000000H
7      bra configuro
8
9      ORG 000700H
10     ;          H   O   L   A
11     mensaje: DB 76H, 3FH, 38H, 77H
12     ORG 000750H
13     ;          P   A   T   O
14     mensaje2: DB 73H, 77H, 78H, 3FH
15
16     ORG 000100H
17     configuro:
18     movlb 0H
19     movlw 60H
20     movwf OSCCON1, 1
21     movlw 02H
22     movwf OSCFRC, 1
23     movlw 40H
24     movwf OSCEN, 1
25     movlb 4H
26     movlw 80H
27     movwf TRISB, 1    ;RD6:RD0 como salidas
28     movwf ANSELB, 1   ;RD6:RD0 como digital
29     movlw 0FOH
30     ;11110000B
31     movwf TRISB, 1    ;RB3:RB0 como salidas
32     movwf ANSELB, 1   ;RB3:RB0 como digital
33     bcf ANSELB, 4, 1  ;RB4 como entrada
34     bcf WPUB, 4, 1   ;RB4 como pullup
35     clrf LATB, 1      ;condicion inicial
36
37     inicio:
38     btfsc PORTB, 4, 1 ;pregunto si
39     bra nropresione
40     bra sropresione
41
42     nropresione:
43     clrf TBLPTRU, 1
44     movlw 07H
45     movwf TBLPTRH, 1
46     clrf TBLPTRL, 1
47     ;El puntero d
48     TBLRD*+
49     ;lectura del
50     movff TABLAT, LATD ;mueve el con
51     bsf LATB, 0, 1
52     call nopes
53     bcf LATB, 0, 1
54     TBLRD*+
55     ;lectura del
56     movff TABLAT, LATD ;mueve el con
57     bsf LATB, 1, 1
58     call nopes
59     bcf LATB, 1, 1
60     TBLRD*+
61     ;lectura del
62     movff TABLAT, LATD ;mueve el con
63     bsf LATB, 2, 1
64     call nopes
65     bcf LATB, 2, 1
66     TBLRD*+
67     ;lectura del
68     movff TABLAT, LATD ;mueve el con
69     bsf LATB, 3, 1
70     call nopes
71     bcf LATB, 3, 1
72
73     sropresione:
74     clrf TBLPTRU, 1
75     movlw 07H
76     movwf TBLPTRH, 1
77     movlw 50H
78     movwf TBLPTRL, 1
79     ;El puntero e
80     TBLRD*+
81     ;lectura del
82     movff TABLAT, LATD ;mueve el con
83     bsf LATB, 0, 1
84     call nopes
85     bcf LATB, 0, 1
86     TBLRD*+
87     ;lectura del
88     movff TABLAT, LATD ;mueve el con
89     bsf LATB, 1, 1
90     call nopes
91     bcf LATB, 1, 1
92     TBLRD*+
93     ;lectura del
94     movff TABLAT, LATD ;mueve el con
95     bsf LATB, 2, 1
96     call nopes
97     bcf LATB, 2, 1
98     TBLRD*+
99     ;lectu
100    movff TABLAT, LATD ;mueve el con
101    bsf LATB, 3, 1
102    call nopes
103    bcf LATB, 3, 1
104    TBLRD*+
105    ;lectu
106    movff TABLAT, LATD ;mueve el con
107    bsf LATB, 3, 1
108    call nopes
109    bcf LATB, 3, 1
110    bra inicio
111
112    nopes:
113    nop
114
115    end upcino

```

32

Ejemplo para visualizar HOLA y CERO con control en RB4

```

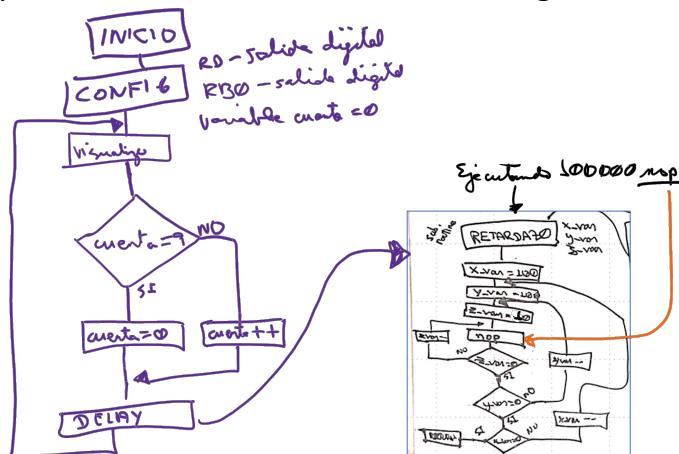
1      PROCESSOR 18F57Q43
2      #include "cabecera.inc"
3
4      PSECT upcino, class=CODE,reloc=2, abs
5      upcino:
6          ORG 000000H      ;Vector de RESET
7          bra configuro
8
9          ORG 000200H      ;Direccion de memoria de programacion
10         ;H   O   L   A
11         mensajes1: DB 76H, 3FH, 38H, 77H
12         ORG 000220H      ;Direccion de memoria de programacion
13         ;C   E   R   O
14         mensajes2: DB 39H, 79H, 50H, 3FH
15
16         ORG 000100H
17 configuro:
18         movlb 0H
19         movlw 60H
20         movwf OSCCON1, 1
21         movlw 02H
22         movwf OSCFRQ, 1
23         movlw 40H
24         movwf OSCEN, 1
25         movlb 4H
26         movlw 80H
27         movwf TRISD, 1    ;RD6-RD0 como salidas
28         movwf ANSEL0, 1    ;RD6-RD0 como digitales
29         movlw 0FOH
30         movwf TRISEB, 1   ;RB3-RB0 como salidas
31         movwf ANSELB, 1   ;RB3-RB0 como digitales
32         bsf TRISB, 4, 1   ;RB4 como entrada
33         bcf ANSELB, 4, 1  ;RB4 como digital
34         bsf WPUB, 4, 1    ;RB4 con pullup activado
35         clrf LATB, 1       ;Habilitadores en cero (cond 1)
36
37         inicio:
38             btfsc PORTB, 4, 1  ;Pregunto si presione el boton
39             bra noperacion     ;viene aqui cuando es falso
40             clrf TBLPTRU, 1      ;viene aqui cuando es verdadero
41             movlw 02H
42             movwf TBLPTRH, 1
43             movlw 20H
44             movwf TBLPTRL, 1      ;El TBLPTR esta mirando a direccion 20
45             bra siguiente
46
47         noperacion:
48             clrf TBLPTRU, 1
49             movlw 02H
50             movwf TBLPTRH, 1
51             clrf TBLPTRL, 1      ;El TBLPTR esta mirando a direccion 20
52             siguiente:
53                 TBLRD*+           ;Lectura del contenido apuntado por TB
54                 movff TABLAT, LATD
55                 bsf LATB, 0, 1
56                 call nopes
57                 bcf LATB, 0, 1
58                 TBLRD*+           ;Lectura del contenido apuntado por TB
59                 movff TABLAT, LATD
60                 bsf LATB, 1, 1
61                 call nopes
62                 bcf LATB, 1, 1
63                 TBLRD*+           ;Lectura del contenido apuntado por TBLPTR y post incremento
64                 movff TABLAT, LATD
65                 bsf LATB, 2, 1
66                 call nopes
67                 bcf LATB, 2, 1
68                 TBLRD*+           ;Lectura del contenido apuntado por TBLPTR y post incremento
69                 movff TABLAT, LATD
70                 bsf LATB, 3, 1
71                 call nopes
72                 bcf LATB, 3, 1
73             bra inicio
74
75         nopes:
76             nop
77             nop
78             nop
79             nop
80             nop
81             nop
82             nop
83             nop
84             nop
85             return
86
87         end upcino

```

33

Ejemplo: Contador 0-9 (BCD) autoincremental con periodo de incremento de aprox 100ms

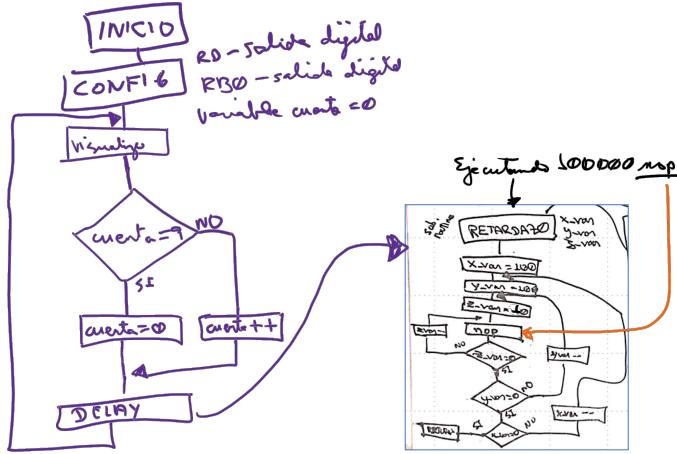
- El mismo hardware que el ejemplo anterior
- No habrá multiplexación, se usara solamente un dígito del display



34

Ejemplo: Contador 0-9 (BCD) autoincremental con periodo de incremento de aprox 100ms

- El mismo hardware que el ejemplo anterior
- No habrá multiplexación, se usará solamente un dígito del display



35

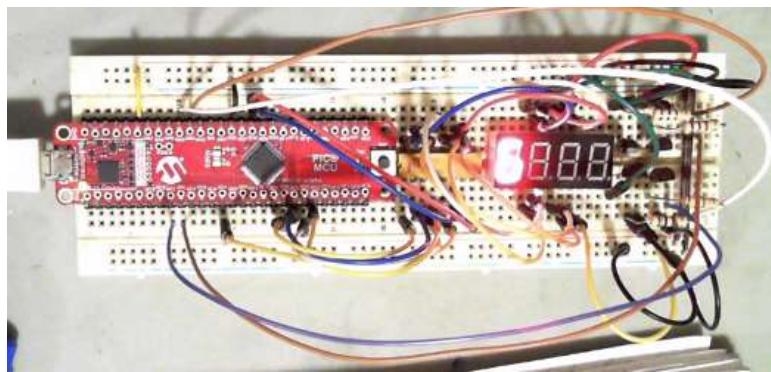
Ejemplo: Contador 0-9 (BCD) autoincremental con periodo de incremento de aprox 100ms

1	PROCESSOR 18F57Q43	40	inicio:	67	retardazo:
2	#include "cabecera.inc"	41	movlb 4H	68	movlb 5H
3		42	clrf TBLPTRU, 1	69	movlw 100
4	PSECT upcino, class=CODE, reloc=2, abs	43	movlw 03H	70	movwf x_var, 1
5		44	movwf TBLPTRH, 1	71	bucle3:
6	REG CUENTA EQU 500H ;etiqueta para la cuenta del contador	45	clrf TBLPTRL, 1 ;puntero de tabla apuntando a 300H	72	movlw 100
7	x_var EQU 501H	46	movlb 5H	73	movwf y_var, 1
8	y_var EQU 502H	47	movwf REG CUENTA, 0, 1 ;recabo cuenta	74	bucle2:
9	z_var EQU 503H	48	movlw 4H	75	movlw 10
10		49	movwf TBLPTRL, 1 ;mueve contenido de Wreg a TBLPTR	76	movwf z_var, 1
11	upcino:	50	TBLRD* ;lectura de lo que esta apuntando	77	bucle1:
12	ORG 000000H	51	movwf TABLAT, LATD ;mueve lo leido hacia RD	78	nop
13	bra configuro	52	movlb 5H	79	decfsz z_var, 1, 1
14		53	movlw 9	80	bra bucle1
15	ORG 000300H	54	cpfseq REG CUENTA, 1 ;pregunto si la cuenta = 9	81	decfsz y_var, 1, 1
16	; 0 1 2 3 4 5 6 7 8 9	55	bra noescierto ;cuenta esta entre 0 y 8	82	bra bucle2
17	tabla_deco: DB 3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H, 7FH, 67H	56	bra siescierto ;cuenta = 9	83	decfsz x_var, 1, 1
18		57	noescierto:	84	bra bucle3
19	ORG 000100H	58	inof REG CUENTA, 1, 1 ;incremento la cuenta	85	return
20	configuro:	59	bra siguiente	86	
21	movlb 0H	60	siescierto:	87	
22	movlw 60H	61	clrf REG CUENTA, 1 ;limpiando la cuenta	end upcino	
23	movwf OSCCON1, 1	62	bra siguiente		
24	movlw 02H	63	siguiente:		
25	movwf OSCFRC, 1 ;HFINTOSC = 4MHz	64	call retardazo		
26	movlw 40H	65	bra inicio		
27	movwf OSCEN, 1				
28	movib 5H				
29	clrf REG CUENTA, 1				
30	movlb 4H				
31	movlw 80H				
32	movwf TRISED, 1 ;RD6 al RDO como salidas				
33	movwf ANSELB, 1 ;RD6 al RDO como digitales				
34	moviw 0F0H				
35	movwf TRISB, 1 ;RB3 al RB0 como salidas				
36	movwf ANSELB, 1 ;RB3 al RB0 como digitales				
37	movlw 0BH				
38	movwf LATB, 1 ;Esta habilitado el cuarto dígito del display				

36

Ejemplo: Contador 0-9 (BCD) autoincremental con periodo de incremento de aprox 100ms

- Implementación



37

Ejercicios:

- Ampliar la aplicación de visualización de dos mensajes a cuatro mensajes (empleando dos pulsadores o switches para las cuatro combinaciones)
- Modificar la aplicación de visualización de dos mensajes con cambio manual en base a una entrada, hacia una visualización automática de ambos mensajes con periodo de cambio de 2 segundos.
- Realizar un contador 0000-9999 ya sea autoincremental con periodo de cuenta de un segundo aproximadamente como de manera manual a través de una entrada de reloj.
- Elaborar una estrategia para que los mensajes a visualizar tengan un efecto de desplazamiento de derecha a izquierda.

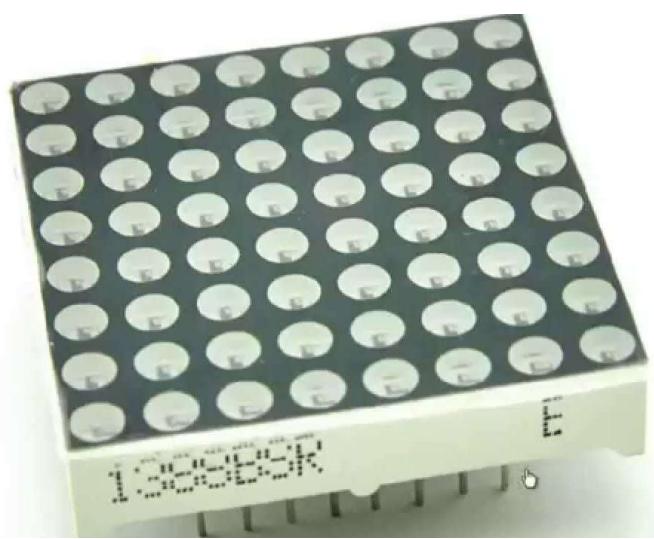
38

Displays alfanuméricos



39

Displays matriciales 8x8



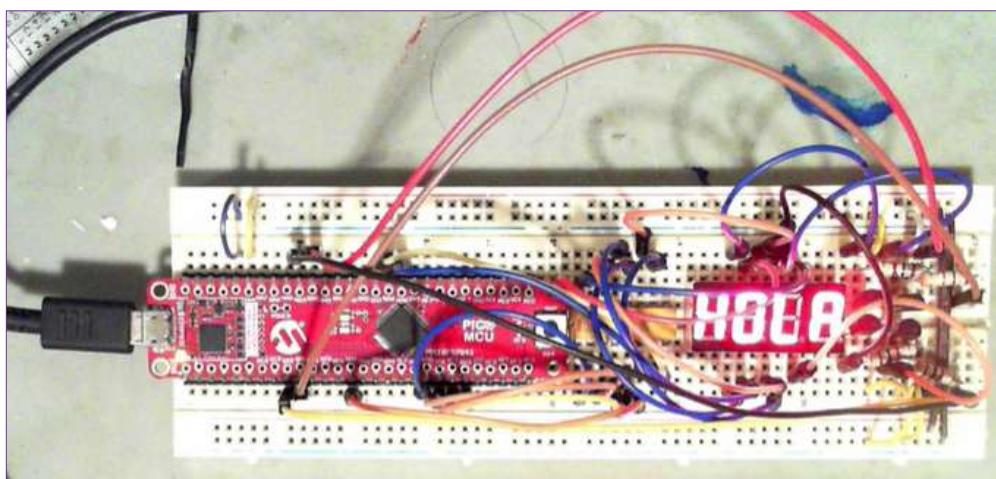
40

Laboratorio 2024-2

41

Ejemplo 2024-2

- Pruebas



42

Para visualizar hola con TBLPTR

<pre> 1 PROCESSOR 18F57Q43 2 #include "cabecera.inc" 3 4 PSECT upcino, class=CODE, reloc=2, abs 5 upcino: 6 ORG 000000H 7 bra configuro 8 9 ORG 001000H 10 p_holis: DB 76H, 3FH, 38H, 77H 11 12 ORG 000100H 13 configuro: 14 ;configuro la fuente de reloj 15 movlb 0H 16 movlw 69H 17 movwf OSCCON1, 1 ;NOSC=HFINTOSC NDIV=1:512 18 movlw 00H 19 movwf OSCFRC, 1 ;HFINTOSC=1MHZ, CPU=1.95KHz 20 movlw 40H 21 movwf OSCEN, 1 ;HFINTOSC enabled 22 ;configuro las E/S 23 movlb 4H 24 movlw 80H 25 movwf TRISD, 1 ;RD[6:0] como salidas 26 movwf ANSELD, 1 ;RD[6:0] como digitales 27 movlw 0FOH ;11110000 28 movwf TRISB, 1 ;RB[3:0] como salidas 29 movwf ANSELB, 1 ;RB[3:0] como digitales 30 ;configuraciones iniciales adicionales 31 clrf LATB, 1 ;habilitadores de los digitos en cero 32 clrf TBLPTRU, 1 33 movlw 50H 34 movwf TBLPTRH, 1 </pre>	<pre> 53 bcf LATB, 2, 1 ;deshabilito primer digito 54 TBLRD++ ;Leo la dir apuntada por TBLPTR 55 movff TABLAT, LATD ;muevo el contenido leido a LATD 56 bsf LATB, 3, 1 ;habilito primer digito 57 call nopesx32 58 bcf LATB, 3, 1 ;deshabilito primer digito 59 bra inicio_mux 60 61 nopesx32: 62 nop 63 nop 64 nop 65 nop 66 nop 67 nop 68 nop 69 nop 70 nop 71 nop 72 nop 73 nop 74 nop 75 nop 76 nop 77 nop 78 nop 79 nop 80 nop 81 nop 82 nop 83 nop 84 nop 85 nop 86 nop 87 nop 88 nop 89 nop 90 nop 91 nop 92 nop 93 nop 94 return 95 nop 96 end upcino </pre>
---	---

43

Para visualizar hola con TBLPTR

<pre> 1 PROCESSOR 18F57Q43 2 #include "cabecera.inc" 3 4 PSECT upcino, class=CODE, reloc=2, abs 5 upcino: 6 ORG 000000H 7 bra configuro 8 9 ORG 005000H 10 p_hola: DB 76H, 3FH, 38H, 77H 11 12 ORG 000100H 13 configuro: 14 ;fuente de reloj 15 movlb 0H 16 movlw 69H 17 movwf OSCCON1, 1 ;NOSC=HFINTOSC NDIV=1:512 18 movlw 00H 19 movwf OSCFRC, 1 ;HFINTOSC=1MHZ, CPU=1.95KHz 20 movlw 40H 21 movwf OSCEN, 1 ;HFINTOSC enabled 22 ;conf de las E/S 23 movlb 4H 24 movlw 80H ;10000000 25 movwf TRISD, 1 ;RD[6:0] como salidas 26 movwf ANSELD, 1 ;RD[6:0] como digitales 27 movlw 0FOH ;11110000 28 movwf TRISB, 1 ;RB[3:0] como salidas 29 movwf ANSELB, 1 ;RB[3:0] como digitales 30 ;configuraciones iniciales adicionales 31 clrf LATB, 1 ;habilitadores de los digitos en cero 32 clrf TBLPTRU, 1 33 movlw 50H 34 movwf TBLPTRH, 1 </pre>	<pre> 36 inicio_mux: 37 movlw LOW p_hola 38 movwf TBLPTRL, 1 ;TBLPTR apuntando a 005000H 39 TBLRD++ ;leo lo apuntado por TBLPTR post incremento 40 movff TABLAT, LATD ;mando a RD lo leido 41 bsf LATB, 0, 1 ;habilito primer digito 42 call nopesx32 43 bcf LATB, 0, 1 ;deshabilitamos el primer digito 44 TBLRD++ ;leo lo apuntado por TBLPTR post incremento 45 movff TABLAT, LATD ;mando a RD lo leido 46 bsf LATB, 1, 1 ;habilito segundo digito 47 call nopesx32 48 bcf LATB, 1, 1 ;deshabilitamos el segundo digito 49 TBLRD++ ;leo lo apuntado por TBLPTR post 50 movff TABLAT, LATD ;mando a RD lo leido 51 bsf LATB, 2, 1 ;habilito tercer digito 52 call nopesx32 53 bcf LATB, 2, 1 ;deshabilitamos el tercer digito 54 TBLRD++ ;leo lo apuntado por TBLPTR post 55 movff TABLAT, LATD ;mando a RD lo leido 56 bsf LATB, 3, 1 ;habilito cuarto digito 57 call nopesx32 58 bcf LATB, 3, 1 ;deshabilitamos el cuarto digito 59 bra inicio_mux 60 61 nopesx32: 62 nop 63 nop 64 nop 65 nop 66 nop 67 nop 68 nop 69 nop 70 nop 71 nop 72 nop 73 nop 74 nop 75 nop 76 nop 77 nop 78 nop 79 nop 80 nop 81 nop 82 nop 83 nop 84 nop 85 nop 86 nop 87 nop 88 nop 89 nop 90 nop 91 nop 92 nop 93 nop 94 return 95 nop 96 end upcino </pre>
---	--

44

```

Palabra HOLÀ con LFINTOSC y 1:64 NDIV

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSELECT upcino, class=CODE, reloc=2, abs
5  upcino:
6  ORG 000000H
7  bra configuro
8
9  ORG 003000H
10 p_hola: DB 76H, 3FH, 38H, 77H
11
12 ORG 000100H
13 configuro:
14 ;configuro la fuente de reloj
15 movlb 0H
16 movlw 56H
17 movwf OSCCON1, 1 ;NOSC=LFINTOSC, NDIV=1:64
18 movlw 10H
19 movwf OSCEN, 1 ;LFINTOSC enabled
20 ;configuro E/S
21 movlb 4H
22 movlw 0F0H ;11110000
23 movwf TRISB, 1 ;RB[3:0] como salidas
24 movwf ANSELB, 1 ;RB[3:0] como digitales
25 movlw 80H
26 movwf TRISSD, 1 ;RD[6:0] como salidas
27 movwf ANSELD, 1 ;RD[6:0] como digitales
28 ;configuraciones iniciales adicionales
29 clrf LATB, 1 ;RB[3:0] en cero (digitos disabled)
30 clrf TBLPTRU, 1
31 movlw 30H
32 movwf TBLPTRH, 1
33
34 inicio_muxi:
35     movlw LOW p_hola
36     movwf TBLPTRL, 1 ;TBLPTR apuntando a dirección 003000H
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72

;leo lo apuntado con posterior incremento de TBLPTR
;muevo contenido leido a RD
;habilitamos primer digito
;unos cuantos nops
;deshabilitamos primer digito
;leo lo apuntado con posterior incremento de TBLPTR
;muevo contenido leido a RD
;habilitamos segundo digito
;unos cuantos nops
;deshabilitamos segundo digito
;leo lo apuntado con posterior incremento de TBLPTR
;muevo contenido leido a RD
;habilitamos tercer digito
;unos cuantos nops
;deshabilitamos tercer digito
;leo lo apuntado con posterior incremento de TBLPTR
;muevo contenido leido a RD
;habilitamos cuarto digito
;unos cuantos nops
;deshabilitamos cuarto digito

nopes:
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    nop
    73    nop
    74    nop
    75    nop
    76    return
    77
    78
    79
    80
    81
    82
    83
    84
    85
    86
    87
    88
    89
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99    end upcino

```

45

Ejemplo 2024-2

• Visualizar PESO o HOLÀ empleando TBLPTR

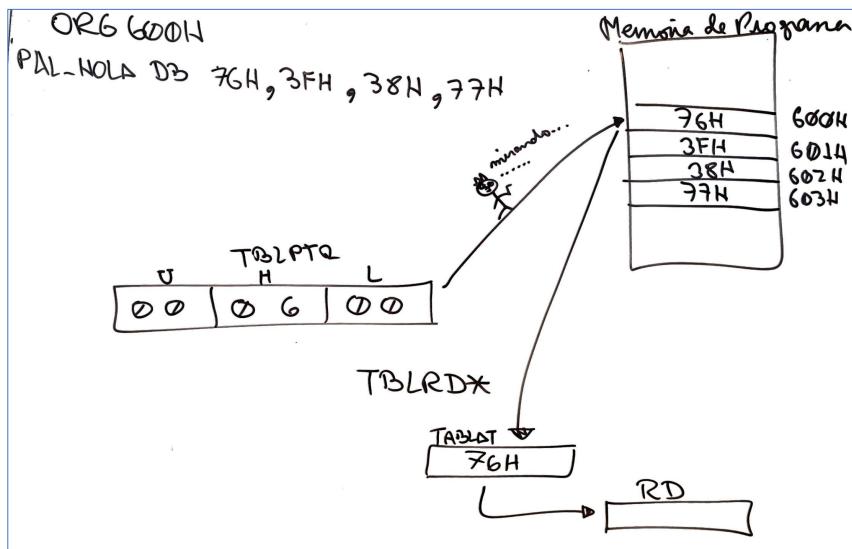
```

1  PROCESSOR 18F57Q43
2  #include "cabecera.inc"
3
4  PSELECT upcino, class=CODE, reloc=2, abs
5  upcino:
6  ORG 000000H
7  bra configuro
8
9  ORG 001000H
10 p_hola:   DB 76H, 3FH, 38H, 77H
11 p_peso:  DB 73H, 79H, 6DH, 3FH
12
13 ORG 000100H
14 configuro:
15 ;configuro fuente de reloj
16 movlb 0H
17 movlw 60H ;NOSC=HFINTOSC NDIV=1:1
18 movwf OSCCON1, 1
19 movlw 02H
20 movwf OSCFRQ, 1 ;HFINTOSC 4MHz, CPU a 4MHz
21 movlw 40H
22 movwf OSCEN, 1
23 ;configuro las E/S
24 movlb 4H
25 movlw 80H
26 movwf TRISSD, 1 ;RD[6:0] sean salidas
27 movwf ANSELD, 1 ;RD[6:0] sean digitales
28 movlw 0F0H ;11110000
29 movwf TRISB, 1 ;RB[3:0] sean salidas
30 movwf ANSELB, 1 ;RB[3:0] sean digitales
31 ;condiciones iniciales
32 clrf LATB, 1 ;Todo RB en cero
33 clrf TBLPTRU, 1
34 movlw HIGH p_peso
35 movwf TBLPTRH, 1
36 movlw LOW p_peso
37     movwf TBLPTRL, 1 ;TBLPTR apuntando a dirección 003000H
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76    nop
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99    end upcino

```

46

Funcionamiento del TBLPTR en el ejemplo



47

Ejemplo 2024-2

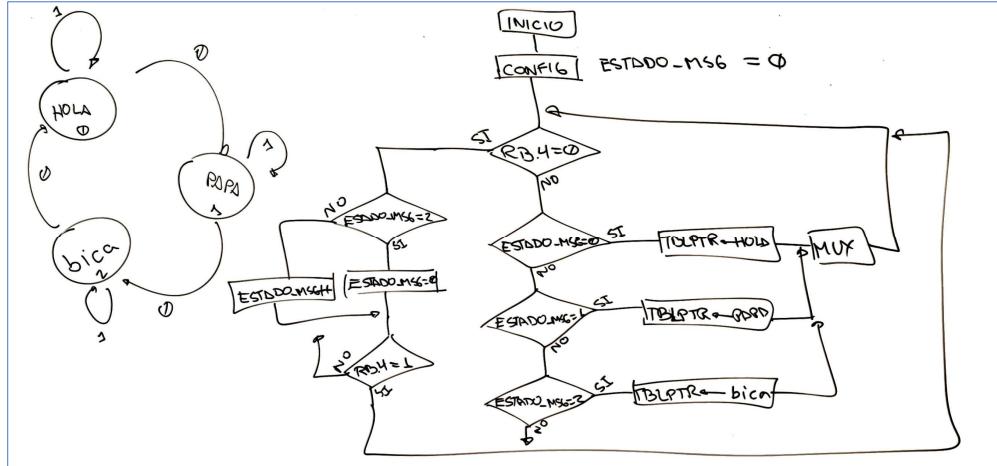
- Intercambio entre HOLA y PESO con pulsador en RB4

1	PROCESSOR 18F57Q43	34	;condiciones iniciales	68	TBLRD*+ ;lectura de TBLPTR con
2	#include "cabecera.inc"	35	clrf LATB, 1 ;Todo RB en cero	69	movff TABLAT, LATD
3		36	clrf TBLPTRU, 1	70	bsf LATB, 3, 1 ;Habilitamos cuarto digito
4	PSECT upcino, class=CODE, reloc=2, abs	37	movlw 10H	71	call nopsex32
5	upcino:	38	movwf TBLPTRH, 1	72	bcf LATB, 3, 1 ;Deshabilitamos cuarto digito
6	ORG 000000H	39	inicio_mux:	73	bra inicio_mux
7	bra configuro	40	btfscl PORTB, 4, 1 ;pregunto si presione	74	nopex32:
8	ORG 001000H	41	bra no_presione ;Falso	75	nop
9	p_hola: DB 76H, 3FH, 28H, 77H	42	bra si_presione ;Verdad	76	nop
10	p_peso: DB 73H, 79H, 6DH, 3FH	43	no_presione:	77	nop
11	ORG 000100H	44	movlw LOW p_hola	78	nop
12	configuro:	45	movwf TBLPTRL, 1 ;TBLPTR apunta a direcc	79	nop
13	;configuro fuente de reloj	46	bra siguiente	80	nop
14	movlw 0H	47	si_presione:	81	nop
15	movlw 60H ;NOSC=HFINTOSC NDIV=1	48	movlw LOW p_peso	82	nop
16	movwf OSCCON1, 1	49	movwf TBLPTRL, 1 ;TBLPTR apunta a direcc	83	102 nop
17	movlw 02H	50	siguiente:	84	103 nop
18	movwf HFINTFRQ, 1 ;HFINTOSC 4MHz, CPU a	51	TBLRD*+ ;lectura de TBLPTR con	85	104 nop
19	movlw 40H	52	movff TABLAT, LATD	86	105 nop
20	movwf OSCEN, 1	53	bsf LATB, 0, 1 ;Habilitamos primer digito	87	106 nop
21	;configuro las E/S	54	call nopsex32	88	107 nop
22	movlb 4H	55	bcf LATB, 0, 1 ;Deshabilitamos primer digito	89	108 return
23	movlw 80H	56	TBLRD*+ ;lectura de TBLPTR con	90	109 end upcino
24	movwf TRIISD, 1 ;RD[6:0] sean salidas	57	movff TABLAT, LATD	91	110 nop
25	movwf ANSELD, 1 ;RD[6:0] sean digitales	58	bsf LATB, 1, 1 ;Habilitamos segundo digito	92	
26	movlw 0FFH ;11110000	59	call nopsex32	93	
27	movwf TRIISB, 1 ;RB[3:0] sean salidas	60	bcf LATB, 1, 1 ;Deshabilitamos segundo digito	94	
28	movwf ANSELB, 1 ;RB[3:0] sean digitales	61	TBLRD*+ ;lectura de TBLPTR con	95	
29	bsf TRIISB, 4 ;RB4 es entrada	62	movff TABLAT, LATD	96	
30	bcf ANSELB, 4 ;RB4 es digital	63	bsf LATB, 2, 1 ;Habilitamos tercer digito	97	
31	bsf RFUB, 4 ;RB4 pullup activado	64	call nopsex32	98	
32		65	bcf LATB, 2, 1 ;Deshabilitamos tercer digito	99	
33		66		100	
		67		101	

48

¿Cómo se puede hacer que cambie tres mensajes usando un solo pulsador?

- Hay que modelar en máquina de estado la acción del pulsador que va a intercambiar el mensaje en el display



49

¿Cómo se puede hacer que cambie tres mensajes usando un solo pulsador?

Line	Processor 18F57Q43	Assembly Code
1	PROCESSOR 18F57Q43	
2	#include "cabecera.inc"	
3		
4	estado_msg EQU 500H ;etiqueta a GPR 500H	
5		
6	PSECT upcino, class=CODE, reloc=2, abs	
7	upcino:	baf NWUB, 4, 1 ;RB4
8	ORG 000000H	;condiciones iniciales
9	bra configuro	cirf LATB, 1 ;RB
10	ORG 000600H	cirf TBLPTRH, 1 ;TBD
11	pal_hola: DB 76H, 3FH, 38H, 77H	movlw 0EH
12	pal_papa: DB 73H, 77H, 73H, 77H	movwf TBLPTRH, 1 ;TBD
13	pal_bica: DB 7CH, 10H, 58H, 5FH	;clrf TBLPTRL, 1 ;TBL
14		movib SH
15		clrf estado_msg, 1 ;est
16	ORG 000100H	movib 4H
17	configuro:	bsf LATB, 2, 1 ;habilitamos tercero
18	;configuro la fuente de reloj	call nopesx32
19	movlb 0H	bcf LATB, 2, 1 ;deshabilitamos tercero
20	movlw 60H	TBLRD++
21	movwf OSCCON1, 1 ;NOSC=HFIN	movff TABLAT, LATD ;nuevo contenido de TABLAT
22	movlw 04H	bsf LATB, 3, 1 ;habilitamos cuarto
23	movwf OSCFRCQ, 1 ;HFINTOSC	call nopesx32
24	movlw 40H	bcf LATB, 3, 1 ;deshabilitamos cuarto
25	movwf OSCEN1, 1 ;HFINTOSC	bra inicio_mux
26	;configuro las E/S	
27	movlb 4H	
28	movlw 80H	
29	movwf TRIISD, 1 ;RD[6:0]	siguiente:
30	movwf ANSELB, 1 ;RD[6:0]	movlw LOW pal_bica
31	movlw OFOH	bra siguiente
32	movwf TRISB, 1 ;RB[5:0]	si_presione:
33	movwf ANSELB, 1 ;RB[3:0]	;yo quiero que salga 14
34	baf TRISE, 4, 1 ;RB4 como	movib SH
35	bcf ANSELB, 4, 1 ;RB4 como	movlw 02H
36		cpfseq estado_msg, 1
37		bra no_es_dos
38		bra si_es_dos
39		no_es_dos:
40		incf estado_msg, 1, 1
41	pal_hola:	bra siguiente
42		si_presione:
43		;yo quiero que salga 14
44		movib SH
45		movlw 02H
46		cpfseq estado_msg, 1
47	configuro:	bra no_es_dos
48	;configuro la fuente de reloj	si_es_dos:
49	movlb 0H	clrf estado_msg, 1 ;pd
50	movlw 60H	siguiente2:
51	movwf OSCCON1, 1 ;NOSC=HFIN	movib 4H
52	movlw 04H	btfs PORTB, 4, 1 ;Pr
53	movwf OSCFRCQ, 1 ;HFINTOSC	bra no_presione
54	movlw 40H	bra si_presione
55	movwf OSCEN1, 1 ;HFINTOSC	nop
56	;configuro las E/S	siguiente:
57	movlb 4H	movwf TBLPTRH, 1 ;TBD
58	movlw 80H	TBLRD++
59	sig01:	movff TABLAT, LATD ;mu
60	movlw 1	bsf LATB, 0, 1 ;ha
61	cpfseq estado_msg, 1	call nopesx32
62	bra sig02	bcf LATB, 0, 1 ;de
63	movib 4H	TBLRD++
64	movwf LOW pal_papa	movff TABLAT, LATD ;mu
65	bra siguiente	bsf LATB, 1, 1 ;ha
66	sig02:	call nopesx32
67	movib 2	bcf LATB, 1, 1 ;de
68	cpfseq estado_msg, 1	TBLRD++
69	bra inicio_mux	movff TABLAT, LATD ;mu
70	movib 4H	bsf LATB, 2, 1 ;habilitamos tercero

50

Asignación 2024-2

- Modificar el ejemplo para que tenga la opción de cambiar cuatro mensajes en el display de cuatro dígitos de siete segmentos multiplexado, empleando dos entradas.
- Palabras como POLO, PALO, TEMP, UPEC, PAPA, MAMA, COLA, LAZO, PEPE, MANO, P18F, NANO, etc
- Se recomienda usar CPFSEQ en lugar de muchos BTFSS/BTFSC
- Si ya lo lograste, intenta ahora intercambiar los mensajes empleando un solo pulsador

51

Fin de la sesión

52