

*Me fué como el culo, mirar última página
Lo nos lo explicaron al día
siguiente de la práctica :)*



Escola de Enxeñaría Industrial
Grado en Ingeniería en Electrónica Industrial y Automática
Prácticas de Electrónica Digital y Microcontroladores

PRÁCTICA 9

PERIFERICOS DE ENTRADA/SALIDA DIGITAL

1. INTRODUCCIÓN

Para que un microcontrolador pueda interactuar con un proceso debe disponer de periféricos de entrada/salida digital. Para que un diseñador de aplicaciones pueda utilizar un microcontrolador debe conocer las características eléctricas y funcionales de dichos periféricos así como los recursos de programación disponibles. En esta práctica se analizarán y utilizarán los recursos del PIC18F47Q10 disponibles en la tarjeta Curiosity_HPC.

2. OBJETIVOS DE APRENDIZAJE

Para el microcontrolador PIC18F47Q10:

- Identificar e interpretar los parámetros eléctricos que definen el comportamiento de un periférico de Entrada / Salida digital.
- Comprender los conceptos ligados a operaciones de entrada y salida digital.
- Identificar la problemática de la sincronización de las operaciones.
- Conocer y gestionar los recursos de programación de operaciones de Entrada/Salida.

3. TAREAS PREVIAS A LA REALIZACIÓN DE LA PRÁCTICA

Para una preparación adecuada de la práctica, el alumno debe realizar las siguientes tareas previas:

- a) Leer y entender el tema correspondiente a los recursos del PIC18F47Q10 destinados a operaciones de E/S digital.
- b) Revisar las características eléctricas de los terminales dedicados a E/S digital.
- c) Identificar en el esquema de la tarjeta Curiosity_HPC, los circuitos dedicados a E/S digital.
- d) Realizar el cuestionario de evaluación.
- e) Preparar una propuesta de listados de programa que respondan a los diagramas de flujo de cada una de las tareas de la práctica.

4. DESARROLLO DE LA PRÁCTICA

Tarea 9.1:

Escribir y depurar un programa capaz de activar el LED conectado al terminal RA5 (LED3) del PORTA cuando se active (PULSADOR_ON) el pulsador conectado al terminal RB4 del PORTB, y de activar el LED conectado al terminal RA4 (LED2) del PORTA cuando el pulsador no se active (PULSADOR_OFF).

Realizar la detección del estado del pulsador por consulta periódica. El

programa debe responder al siguiente diagrama de flujo:

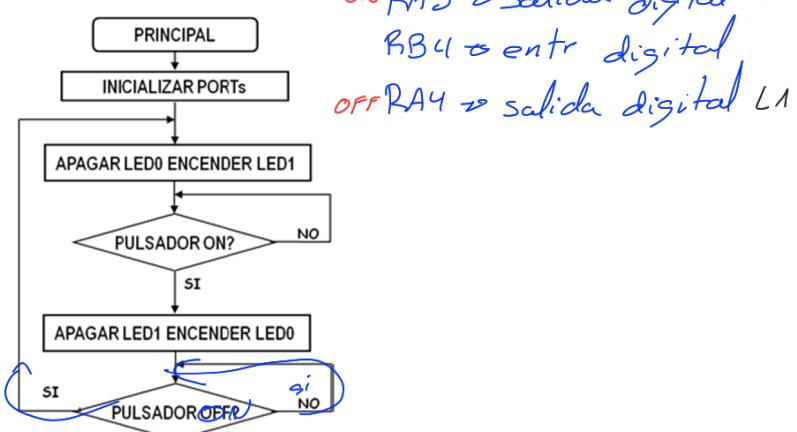


Figura 1. Diagrama de flujo de la tarea 9.1

Tarea 9.2:

Esta tarea está destinada poner de manifiesto la necesidad de sincronizar dos dispositivos cuya velocidad de operación es muy diferente. En este caso vamos a comprobar la velocidad de ejecución de instrucciones del microcontrolador respecto a la velocidad de activación de un pulsador. Para ello vamos a diseñar y depurar un programa que incremente el valor de una variable cada vez que se active el pulsador conectado a RB4 y presentar el valor de la variable en los LEDs conectados a los cuatro bits más significativos del PORTA.

Tarea 9.2.1: Escribir un programa que responda al siguiente diagrama de flujo:

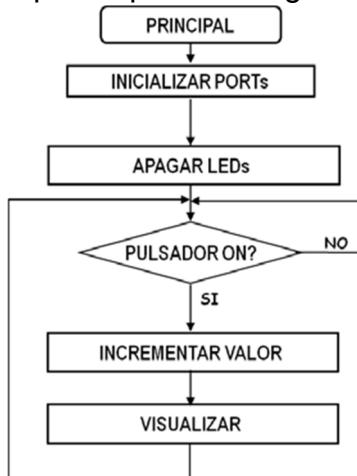


Figura 2. Diagrama de flujo de la tarea 9.2.1

Responer a la Cuestión 1



Tarea 9.2.2: Escribir un programa que detecte flancos de activación del pulsador y que responda al siguiente diagrama de flujo:

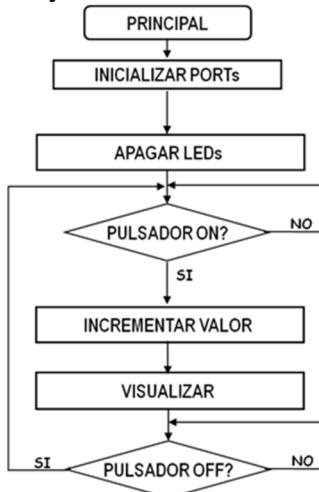


Figura 3. Diagrama de flujo de la tarea 9.2.2

Tarea 9.2.3: Escribir un programa que responda al siguiente diagrama de flujo para detectar flancos de activación en el pulsador con eliminación de rebotes:

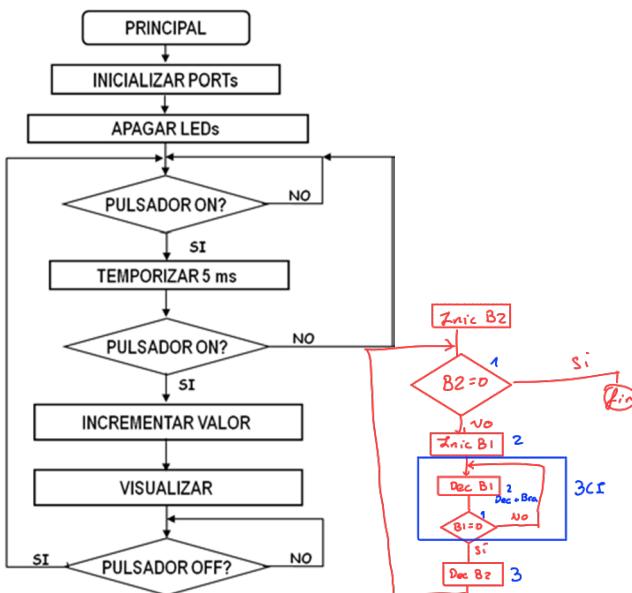


Figura 4. Diagrama de flujo de la tarea 9.2.3

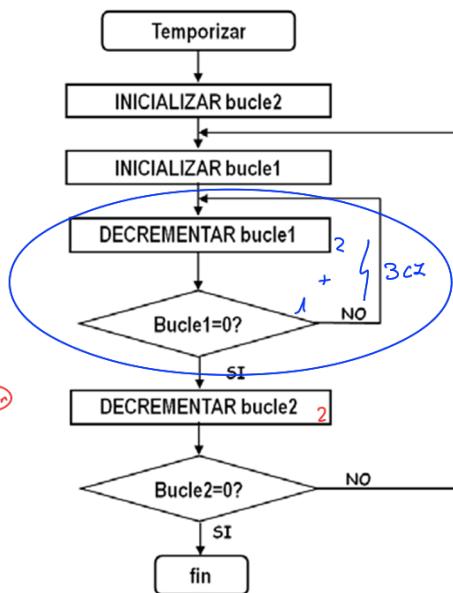


Figura 5. Diagrama de flujo de rutina de temporización

La temporización de 5 ms debe realizarse con una subrutina de programa que responda al diagrama de flujo de la Figura 5. Como puede comprobarse, se trata de dos bucles anidados en los que el tiempo T transcurrido desde el inicio de la rutina hasta el final depende del tiempo de ejecución de las instrucciones y del número de veces que se repitan cada uno de los bucles. La variable Bucle1 determina las veces que se repite el bucle interno y la variable Bucle2 determina las veces que se repite el bucle externo.

Para calcular el valor de las variables *bucle1* y *bucle2* se puede utilizar la expresión genérica mostrada a continuación, pero su validez depende de la estructura del programa

$$5 \times 10^{-3} = 4 \cdot 1 \mu s \cdot B2(9 + B1 \cdot 3) \Rightarrow B2 = 5 = (0x05)$$

$$T = \frac{4}{F_{OSC}} \times Bucle2 \times (N_{CM2} + Bucle1 \times N_{CM1})$$

En donde:

T -> es la temporización total

Fosc -> es la frecuencia del oscilador utilizado. Si se usa el oscilador interno del microcontrolador en su programación por defecto (reset) la frecuencia de oscilación (*Fosc*) es de 1MHz.

N_CM -> Es el número de Ciclos Máquina equivalente a las instrucciones utilizadas para implementar los bucles del programa. En el tema del Juego de Instrucciones se puede localizar esta información para cada instrucción. Si se tiene en cuenta que, en tiempo, un CM equivale a 4/*Fosc* y se usa *Fosc*=1MHz, entonces *CM*=4μs.

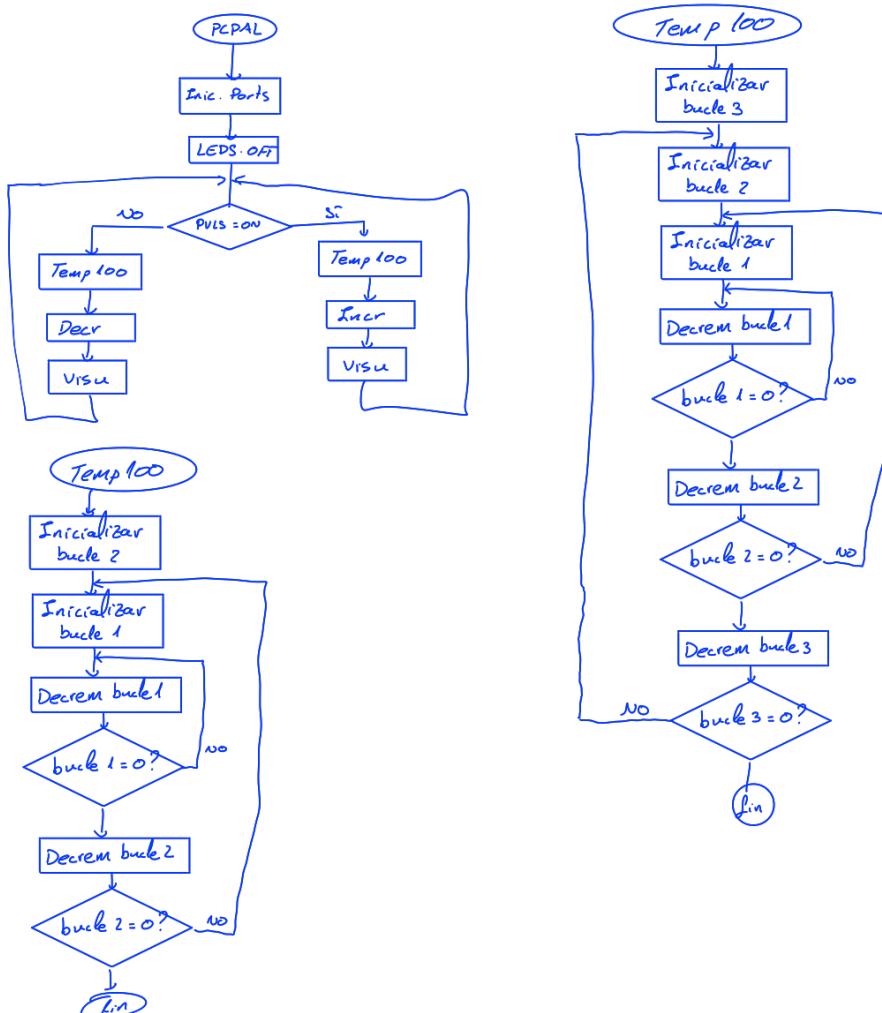
```

Movlw 1      bra 2
Movwf 1      return z
decfsz 1,2,03
  
```

Tarea 9.3: *→ Preguntar 6 puse 5 ciclos 1+1+1+2*

Escribir y depurar un programa capaz de incrementar de forma continua cada 100ms, los LEDs del conjunto de 4 conectados a los 4 bits más significativos del PORTA. La temporización debe realizarse por programa. La secuencia de encendido debe cambiar de incrementar a decrementar, y viceversa, cada vez que se active el pulsador conectado al terminal 4 del PORTB. La detección del estado del pulsador debe realizarse por consulta periódica.

5ms



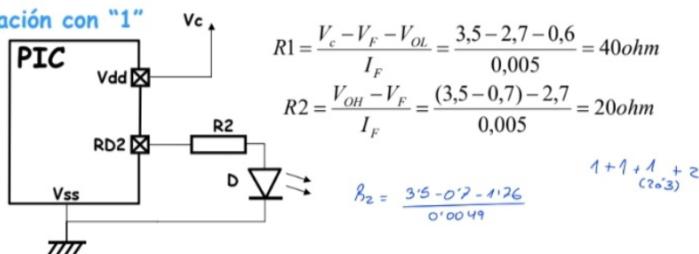
• Conexión de un LED V_C 

LED:
 $-V_{Fmax} \rightarrow 2,7V$
 $-I_{Fmax} \rightarrow 30mA$

Terminal:
 $-V_{OHmin} \rightarrow V_{DD} - 0,7V$
 $-V_{OLmax} \rightarrow 0,6V$
 $-I_{Omax} \rightarrow 25mA$

Si $V_c = 3,5V$ y $I_F = 5mA$

Activación con "1"



$$R1 = \frac{V_c - V_F - V_{OL}}{I_F} = \frac{3,5 - 2,7 - 0,6}{0,005} = 40\text{ohm}$$

$$R2 = \frac{V_{OH} - V_F}{I_F} = \frac{(3,5 - 0,7) - 2,7}{0,005} = 20\text{ohm}$$

$$R_2 = \frac{3,5 - 0,7 - 1,76}{0,0049}$$

$1 + 1 + 1 + 2$
 $(2 \cdot 3)$

MOVLW d

MOVWF bucle2 = 5

Ini_bucle2:

MOVLW d = 250 1.4us

MOVWF bucle1 1.4us

Ini_bucle1:

DECFSZ bucle1,f,a bucle1.4us+4us

bra Ini_bucle1 2.4us

DECFSZ bucle2,f,a bucle2.(Total, +4us)

bra Ini_bucle2 2.4us

$$T_{total} = (1.4\mu s) + (1.4\mu s) + bucle1.4\mu s + 4\mu s + 2.4\mu s$$

$$T_{total} = 20\mu s + bucle1.4\mu s$$

$$T_{total} = bucle2.(24\mu s + bucle1.4\mu s) + 8\mu s$$

→ Si $T_{total} = 5ms$

→ Dar valores a bucle1 y bucle2

Previo

	tiempo=2	tiempo=25	t=2 f=31kHz
MOVLW tiempo	1.4us	1.4us	1.0'13
MOVWF veces, a	1.4us	1.4us	1.0'13
bucle:			
DECFSZ veces, a	2.4us + 4us	25.4us + 4us	2.0'13 + 0'13
BRA bucle	2.4us	2.4us	2.0'13
	28	120	0'9us

inicio:

MOVLW 20

MOVWF bucle2,A

inicio_bucle2:

MOVLW 19 1.4us

MOVWF bucle1,A 1.4us

BRA inicio_bucle1 2.4us

inicio_bucle1:

DECFSZ bucle1,A b1.4us+4us

BRA inicio_bucle1 2.4us

DECFSZ bucle2,A b2.(Tu + 4us)

BRA inicio_bucle2 2.4us

NOP

$$5000 = b_2(32 + 4b_1) + 8$$

5000 =

```

20      goto    inicio
21
22      ORG 0x20
23      inicio
24          bcf ANSELH,4,ACCESS
25          bcf TRISB,0,ACCESS
26          bcf TRISD,0, ACCESS
27      lazo
28          btfsc PORTB,0,ACCESS
29          goto es_uno
30      es_cero
31          bcf PORTD,0,ACCESS
32          call retardo
33          bra lazo
34      es_uno
35          bsf PORTD,0,ACCESS
36          call retardo
37          bra lazo
38
39      retardo
40          movlw 0x4 = 64
41          movwf cont1
42          b2
43              movlw 104   4us
44              movwf cont2 4us
45          b1
46              decfsz cont2,f 4.104 + 4
47              bra b1
48              decfsz cont1,f 8ms
49              bra b2
50              + 8
51          return
52      END

```

temporizar:

```

    movlw b2
    movwf pos2,a

```

$$C_{bl} = 2$$

bucle1:

```

    movlw bl      4μs  1c
    movwf pos1,a  4μs  1c

```

$$T_{bucle1} = 20\mu s + bl \cdot 4\mu s$$

bucle2:

```

    decfsz pos1,f,a  bl · 4μs + 4μs = 2c
    bra bucle2      2 · 4μs   2c
    decfsz pos2,f,a  b2 · (T_{bucle1} + 4μs) = 2c
    bra bucle1      8μs
    return

```

$$bl \cdot 1c + 5c$$

$$T_{tot} = b2 \cdot (2\mu s + bl \cdot 4\mu s) + 8\mu s$$

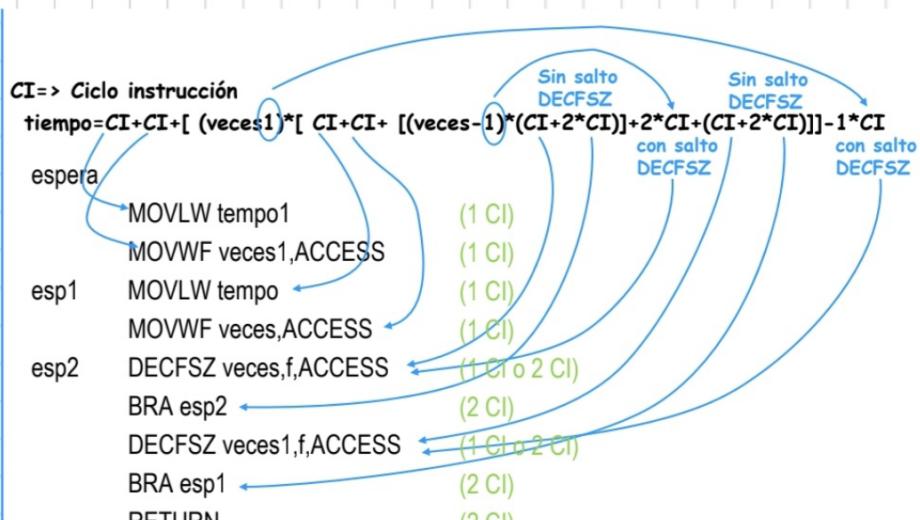
$$b2 \cdot (GC + bl \cdot 1c) + 8c$$

end mi_programa

$$T = \frac{4}{F_{OSC}} \times \text{Bucle2} \times (N_CM2 + \text{Bucle1} \times N_CM1)$$

$$100000\mu s = 4 \times 250 (8 + Bl \times 2) \rightarrow Bl = 196 \rightarrow 0xC4$$

$$B2 = 250 \rightarrow 0xFA$$



Si Fosc=1MHz \rightarrow CI=4us ; si veces1=2 y veces=10 \rightarrow Ciclos Intrucción=69
Tiempo=276us