

Microcontroladores

Laboratorio Semana 1

Semestre: 2021-0

Profesor: Kalun José Lau Gan

1

Agenda

- Requerimientos de software:
 - El MPLAB X IDE
 - El Proteus VSM
- Requerimientos de hardware:
 - Lista de materiales
 - Instrumentos de laboratorio
- Requerimientos de documentos:
 - Hoja técnica del microcontrolador PIC18F4550 rev.E
 - <https://ww1.microchip.com/downloads/en/devicedoc/39632e.pdf>

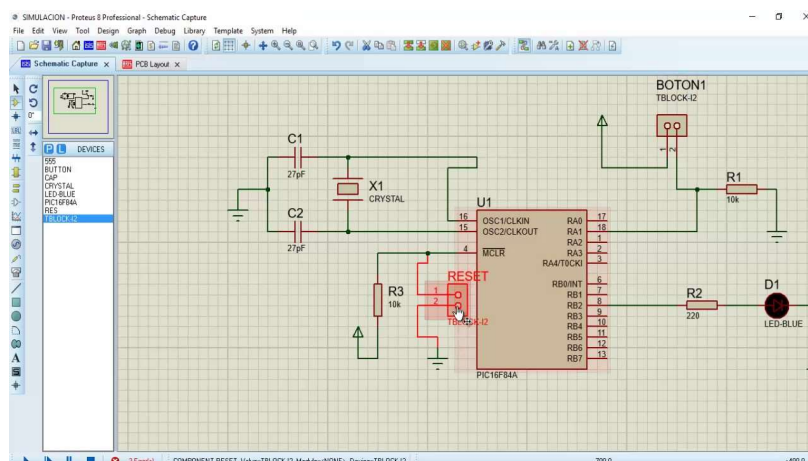
2

Verificación del funcionamiento de los softwares:

- ¿Se instaló correctamente el MPLAB X IDE?
- ¿Instalaste la versión v5.35 del MPLAB X IDE?
 - No instalar versión 5.45 debido a que usa XC8 Assembler en lugar de MPASM
 - Versión 5.35 ya no soporta PICKit2, si vas a usar este último tendrás que emplear la 5.30
- El Proteus. ¿Funciona correctamente?
- Verificar si el Proteus instalado tiene instalado la librería de simulación para el microcontrolador PIC18F4550

3

El Proteus VSM



- Simulador de circuitos

4

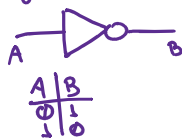
Procedimiento para desarrollar una aplicación con el microcontrolador PIC18F4550

1. Análisis del problema y ver los requerimientos (puertos E/S, tipo de señales, velocidad, consumo energético, etc)
2. Desarrollamos el hardware (el circuito)
3. Elaboramos el algoritmo en diagrama de flujo
4. Redactamos el código en un lenguaje de programación
5. Compilar y realizar la pruebas (simulación, emulación, programación)

7

Ejemplo: Desarrollar un negador lógico de 2 bits empleando el PIC18F4550

Negador lógico:



¿Qué es emular?

-Imitar el funcionamiento de un dispositivo en otra plataforma

Circuito prototipo:

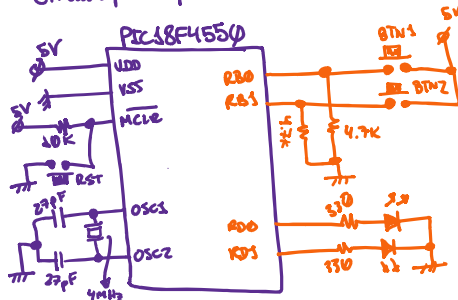
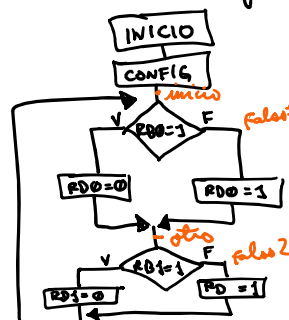


Diagrama de flujo:



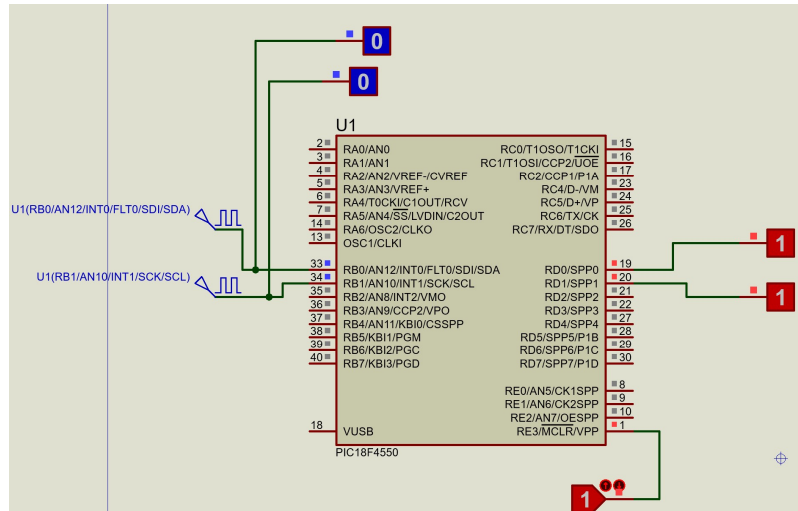
8

Ejemplo: Desarrollar un negador lógico de 2 bits empleando el PIC18F4550

```

1 ;Este es un comentario
2 list p=18f4550
3 #include <pic18f4550.inc> ;libreria de nombre de los
4
5 CONFIG FOSC = XT_XT ; Oscillator Selection
6 CONFIG PWRT = ON ; Power-up Timer Enable
7 CONFIG BOR = OFF ; Brown-out Reset Enable
8 CONFIG WDT = OFF ; Watchdog Timer Enable
9 CONFIG PBADEN = OFF ; PORTB A/D Enable bit
10 CONFIG LVP = OFF ; Single-Supply ICSP Enable
11
12 org 0x0000 ;Vector de reset
13 goto configuracion
14
15 org 0x0020 ;Zona de programa de usuario
16
17 configuracion:
18 bcf TRISD, 0 ;RD0 como salida
19 bcf TRISD, 1 ;RD1 como salida
20
21 inicio:
22 btfs PORTB, 0 ;pregunta si rb0 es uno
23 goto falso1 ;viene cuando es falso
24 bcf LATD, 0 ;viene aquí cuando es verdadero
25 goto otro
26 falso1:
27 bsf LATD, 0
28 otro:
29 btfs PORTB, 1
30 goto falso2
31 bcf LATD, 1
32 goto inicio
33 falso2:
34 bsf LATD, 1
35 goto inicio
36 end

```



9

Directivas de pre-procesador (bits de configuración)

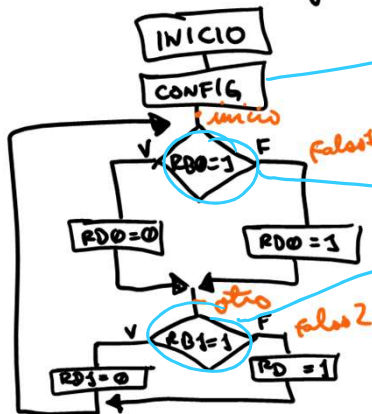
Los esenciales:

CONFIG FOSC = XT_XT → Tipo de fuente de reloj: cristal externo hasta 4MHz
 CONFIG PWRT = ON → Establece un tiempo de pausa al energizarse el uC
 CONFIG BOR = OFF → Detecta un límite mínimo en VDD, debajo del límite el uC estará en RST
 CONFIG WDT = OFF → Perro guardián
 CONFIG PBADEN = OFF → Puertos RB como digitales
 CONFIG LVP = OFF → Programación con 5V deshabilitado

10

Ejemplo: Desarrollar un negador lógico de 2 bits empleando el PIC18F4550

Diagrama de flujo:



```

1 ;Este es un comentario
2 list p=18f4550
3 #include <pic18f4550.inc> ;libreria de nombre de loc
4
5 CONFIG FOSC = XT_XT ; Oscillator Selection
6 CONFIG FWRT = ON ; Power-up Timer Enable
7 CONFIG BOR = OFF ; Brown-out Reset Enable
8 CONFIG WDT = OFF ; Watchdog Timer Enable
9 CONFIG PBADEN = OFF ; PORTB A/D Enable bit
10 CONFIG LVP = OFF ; Single-Supply ICSP Ena
11
12 org 0x0000 ;Vector de reset
13 goto configuracion
14
15 org 0x0020 ;Zona de programa de usuario
16
17 configuracion:
18 bcf TRISD, 0 ;RD0 como salida
19 bcf TRISD, 1 ;RD1 como salida
20
21 inicio:
22 btfss PORTD, 0 ;pregunta si rd0 es uno
23 goto falso1 ;viene cuando es falso
24 bcf LATD, 0 ;viene aqui cuando es verdadero
25 goto otro
26 falso1:
27 bsf LATD, 0
28
29 otro:
30 btfss PORTD, 1
31 goto falso2
32 bcf LATD, 1
33 goto inicio
34 falso2:
35 bsf LATD, 1
36 goto inicio
37 end
  
```