



UNIVERSIDAD DE GUANAJUATO

D. I. C. I. S.

Laboratorio de Microprocesadores y Microcontroladores

Práctica de Laboratorio No. 1:

Introducción al Microcontrolador PIC18F45K50

Práctica de Laboratorio No. 1

Introducción al Microcontrolador PIC18F45K50

Objetivos:

Implementar un sistema básico para operar el microcontrolador PIC 18F45K50
Utilizar el entorno gráfico MPLAB X IDE para editar, compilar y depurar los primeros programas en lenguaje para programar el PIC 18F45K50

Componentes:

- 1 Microcontrolador PIC 18F45K50
- 8 LEDS
- 8 resistores de 220 Ω

Equipo:

- 1 Fuente de +5V
 - 1 Punta lógica
 - 1 Multímetro
 - 1 Plantilla
 - 1 Equipo de cómputo con paquete de software MPLAB X IDE
 - 1 Programador para el PIC18F45K50
 - 1 Hoja de datos del PIC18F45K50
-

1.1 Introducción.

Los microcontroladores son dispositivos versátiles que tienen una infinidad de aplicaciones. Son pequeños, de bajo costo y con prestaciones reducidas. Esto sin embargo no limita su uso. Al contrario, los microcontroladores son los dispositivos de procesamiento de información más utilizados en el mundo. En este curso se utilizará el microcontrolador PIC18F45K50 de Microchip Technology Inc, una de las empresas líderes en el mercado de los microcontroladores.

El PIC18F45K50 es fabricado en varios empaquetamientos: TQFP (Thin Quad Flat Pack) con 44 pines, UQFN (Ultra-Quad-Flat No-leads) con 40 pines, SOIC (plastic Small Outline Integrated Circuit), SSOP (Small Shrink Outline Package) y DIP (Dual in Line Package) con 28 pines. El diagrama de los pines para el empaquetamiento DIP con 40 pines, que es el que se utilizará en este curso, es el siguiente:

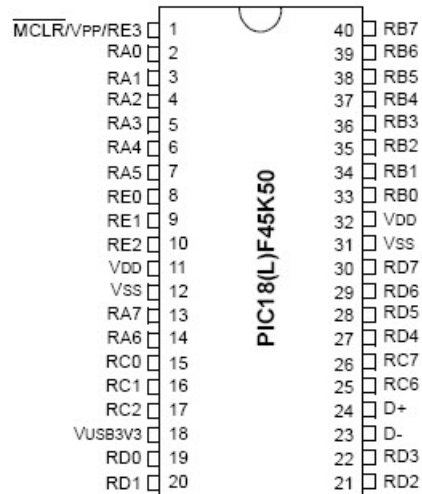


Diagrama de 40 pines DIP

Las características principales de este PIC están resumidas en el siguiente cuadro:

Dispositivo	Memoria de datos		Memoria de programa (palabras)	MSSP	I/O	10-bit A/D (canales)	Temporizadores 8-bit/16-bit	Comparadores
	SRAM (bytes)	EEPROM (bytes)						
PIC18F45K50	2048	256	16384	1	A, B, C, D, E	25	2/2	2

Todas las características de este microcontrolador se pueden consultar en la hoja de datos que se puede descargar de la página www.microchip.com.

1.2 Desarrollo.

A. Para editar, depurar y simular el código con el que se programará el microcontrolador se utilizará el entorno gráfico MPLAB X IDE (Integrated Development Environment). Se abre el programa y se selecciona la opción **New Project** del menú **File**. En la siguiente ventana emergente se selecciona “**Microchip Embedded**” y de esta opción “**Standalone Project**”. Se da click en next y en la siguiente ventana se selecciona el microcontrolador a usar. Después del click en next se puede seleccionar la herramienta para programar/depurar el microcontrolador. Si no se tiene alguna de las indicadas, se selecciona “Simulator”. Después de otro click en Next, se selecciona el compilador “xc8” para trabajar código en lenguaje C. En el siguiente click en Next aparece la ventana de selección del nombre y ubicación del proyecto, así como otras opciones como la de establecer el proyecto como proyecto principal. Al llenar las casillas con la información adecuada se activa el botón “Finish” con el que se pasa al entorno de trabajo

Se procede a crear un archivo con extensión “.c”, para ello se puede seleccionar la carpeta “Source Files” en la ventana superior (Projects) y se selecciona “New” – Main.c. Se abre entonces una nueva ventana donde se introducirá el nombre del archivo y la opción del folder donde se guardará. Inmediatamente se abre una ventana en la parte superior centro-derecha donde se escribirá el siguiente código

```
; Programa que genera el parpadeo de un LED con retardo
```

```
#pragma config FOSC = INTOSCIO // Oscillator Selection (Internal oscillator)
#pragma config WDTEN = OFF // Watchdog Timer Enable bits (WDT disabled in hardware)
#pragma config MCLRE = ON // Master Clear Reset Pin Enable (MCLR pin enabled; RE3 input disabled)
#pragma config LVP = OFF // Single-Supply ICSP Enable bit (Single-Supply ICSP disabled)
#pragma config ICPRT = OFF // Dedicated In-Circuit Debug/Programming Port Enable (ICPORT disabled)
```

```
#include <xc.h>

#define _XTAL_FREQ 1000000    // Frecuencia por default

void main(void) {

    TRISD=0;
    ANSELD=0;

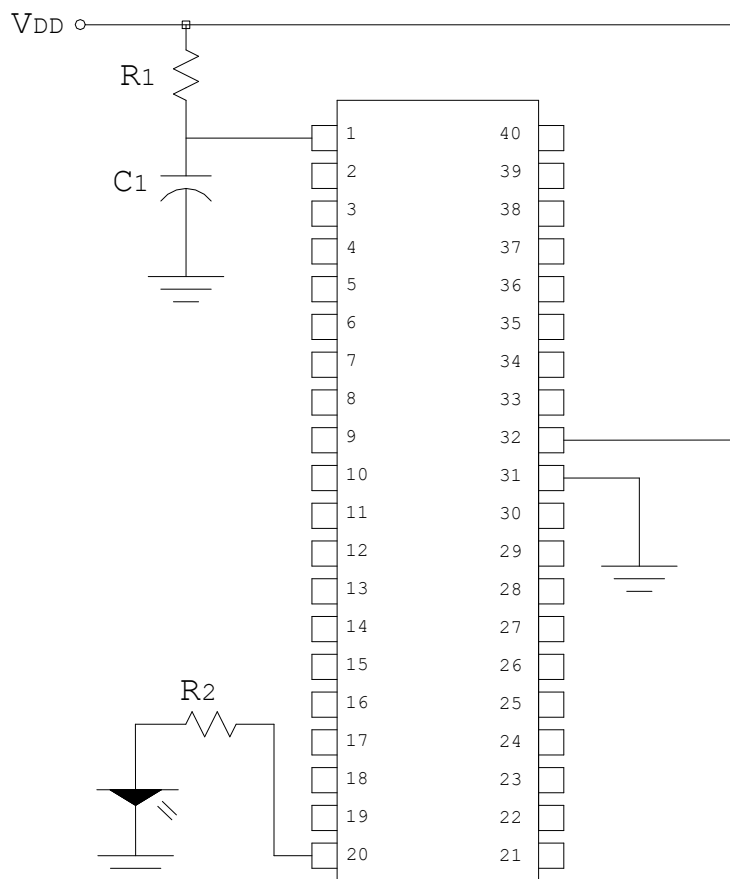
    while(1)
    {
        LATDbits.LD1 ^= 1;
        __delay_ms(500);
    }

    return;
}
```

Ya que se haya escrito este código se seleccionará la opción Build Main Proyecto en el menú RUN o el ícono en la barra de botones.

Al haberse compilado el código sin errores se utilizará el programa que indique el profesor o encargado para programar el PIC. Solicite la ayuda del profesor para realizar esta operación.

B. Ya que se haya programado el PIC, este se pasa a la plantilla de trabajo y se conectará de acuerdo con el siguiente diagrama. Tener cuidado al manejar el PIC de no dañarlo con electricidad estática (utilizar pinzas o pulsera antiestática para su manejo). El voltaje que se utilizará es de 5V para V_{DD} y 0V para V_{SS} . La corriente máxima de salida de cada pin es de 25 mA. Si los LEDs con los que se va a trabajar no soportan esa corriente entonces puede conectar en serie una resistencia de 220Ω con cada LED.



Donde los valores para los componentes se dan en la siguiente tabla:

Símbolo	Valor Mín	Valor Max
V_{DD}	5V	5V
R_1	1K Ω	-
R_2	100 Ω	220 Ω
C_1	10 μ F	-

La resistencia R_1 y el capacitor C_1 se recomiendan para evitar que voltajes fuera de rango lleven el dispositivo a reinicios o consumos de energía más allá de los especificados para el dispositivo. En caso de utilizar una fuente regulada puede prescindir de este arreglo.

1.3 Ejercicios en el laboratorio (para incluir en el reporte de la práctica)

1. Si se quiere generar la salida del LED por el pin RA1 del puerto A (pin 3), escriba las instrucciones que se tienen que cambiar en la sección de configuración del microcontrolador y en la sección del ciclo infinito. Cambie el código, compile y programe el microcontrolador. Muestre el cambio al profesor.

2. Modifique el código anterior y, que en lugar de un LED que este prendiendo y apagando, en las salidas del puerto A se presente una secuencia de valores en binario descendente de forma continua (FF, FE, FD...00, FF, FE...), para observar el funcionamiento de la secuencia esta se debe mostrar mediante 8 LEDs conectados a

dicho puerto. Utilice, de preferencia una variable tipo char que lleve el valor descendente. Presente el comportamiento del microcontrolador al profesor.

2. ¿Como cambia el comportamiento del programa si, en el programa principal se agrega, inmediatamente después de la función “__delay_ms(500);” la misma función “__delay_ms(500);”? Determine la diferencia reprogramando el microcontrolador y mostrándole el nuevo comportamiento al profesor.

1.4 Ejercicios de tarea (para incluir en el reporte de la práctica)

En la hoja de datos del microcontrolador busque la siguiente información:

1. Complete la siguiente tabla que indica el número de pin del microcontrolador que está ligado con cada canal analógico (incluir los 25 canales analógicos con que cuenta el microcontrolador).

Canal analógico	PIN
AN0	2
AN1	3
AN2	4
⋮	⋮

2. Escriba todas las frecuencias de reloj que se pueden utilizar para el CPU con los osciladores internos.

3. Encuentre en la hoja de especificaciones y reporte la corriente que consume el microcontrolador en modo Sleep.