**ARQUITECTURA DE COMPUTADORAS**

**Practica 01. Salida por Grupos**

**OBJETIVO:**

Comprender la configuración de los puertos del microcontrolador. Configurar varios puertos del PIC como salida.

**MATERIALES Y EQUIPO:**

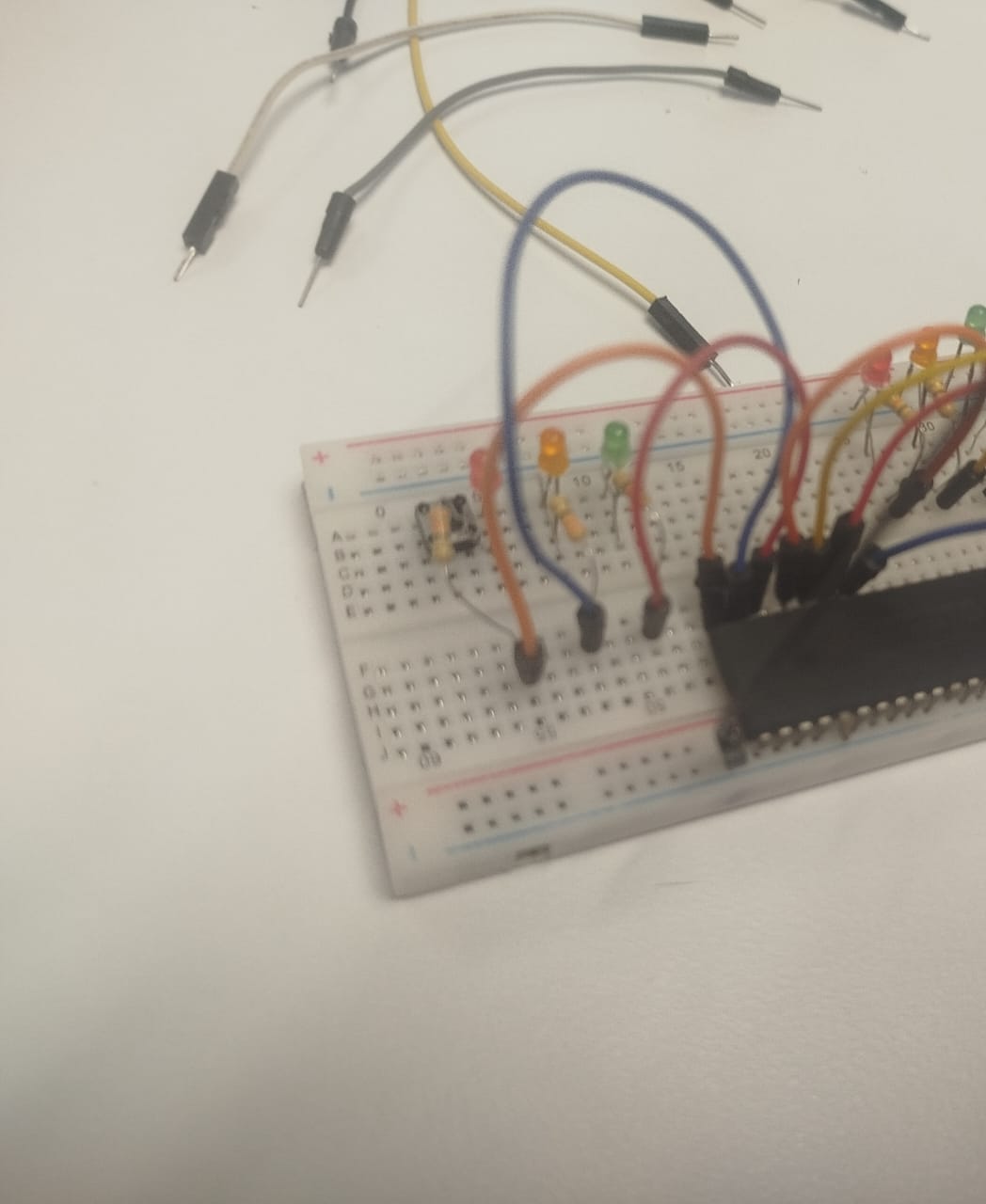
|  |  |
| --- | --- |
| Microcontrolador 16F877A con cristal (16 MHz) y capacitores (22 pF) |  |
| Protoboard (mínimo de 400 puntos) |  |
| 1 Resistencia de 10Kohm |  |
| Cables de conexión (UTP o Dupont) |  |
| Entrenadora Digital (IDL-800) |  |
| Grabadora de Microcontroladores (Master PROG) con su cable USB y el software de grabación. | Resultado de imagen de master prog mercadolibre |
| LEDS 8 |  |

**INTRODUCCIÓN:**

Un microcontrolador posee puertos de entrada salida o mejor dicho pines de entrada / salida (I/O) de propósito general, mediante ellos el PIC puede monitorear y controlar otros dispositivos. Por consiguiente es necesario configurarlos apropiadamente. Para este microcontrolador el puerto A tiene funciones analógicas que en prácticas posteriores se revisara a detalle su configuración.

**PROCEDIMIENTO:**

1. Realice la conexión básica del microcontrolador. Conecte en orden los pines del puerto C a los led del entrenador digital.



2. Haga un proyecto nuevo (practica01) codifique lo siguiente:

/\* ARQUITECTURA DE COMPUTADORAS Ing. Daniel López Piña UAMM - UAT

\* Practica 01: Salida por grupo

\* Integrantes:

\* La instrucción TRIS (TRISA, TRISB, TRISC, TRISD, TRISE) configura el puerto como entrada o salida

\* Se asigna un valor de 0 para salida y un valor de 1 para entrada

\* PIC16F877A 16Mhz

\*

\* Conectar todos los pines del puerto C a los Leds de la Entrenadora Digital

\*/

void main(){

TRISC=0; //Configura todo el puerto (RC0, RC1, RC2, RC3, RC4, RC5, RC6 y RC7) C como Salida

// TRISC=0x00; //Es el equivalente a la línea anterior, pero asignándole valor Hexadecimales

// TRISC=0b00000000; //Es otro equivalente, pero asignándole valor Binario

PORTC=0; // Manda un 0 lógico (LOW) a todos los pines del puerto C. en otras palabras apaga todo el puerto.

//PORTC=0x00; //Es el equivalente a la línea anterior, pero asignándole valor Hexadecimales

//PORTC=0b00000000; //Es otro equivalente, pero asignándole valor Binario

while(1){ //Inicia un ciclo infinito

PORTC=~PORTC; // "~" Invierte el valor del puerto completo: si está apagado lo enciende; si esta encendido lo apaga.

}

3. Compilar y grabar el proyecto.

4. Realice otro proyecto y configure los puertos B, C, D y E como salida. Encienda y apague todos los puertos al mismo tiempo como en el código de ejemplo. Conecte los siguientes pines a los led del entrenador digital: RB0, RB1, RC2, RC3, RD4, RD5, RE0 y RE1.

5. Genere un reporte de la práctica conteniendo los siguientes puntos: Introducción, Desarrollo (incluir esquemas de conexión y código) y Conclusiones.

**CONCLUSION**

Primera vez que ejecuta PORTC=~PORTC;  
Todos los LEDs se encienden (porque ~0 = 1).

Segunda vez que ejecuta PORTC=~PORTC;  
 Todos los LEDs se apagan (porque ~1 = 0).

Tercera vez → Todos encienden.

Cuarta vez → Todos apagan.

El programa hace que **todos los LEDs enciendan y apaguen al mismo tiempo**, en forma de **parpadeo general**.