



Universidad Tecnológica del Perú

Facultad de Ingeniería Electrónica y Mecatrónica

Curso : Microcontroladores

Profesor : Domínguez Jesús

Práctica Calificada Nro: 1

Tema : Práctica Calificada 1

Integrantes : Espinoza Valera, Jesús Alberto
Francisco
Alberto Francisco, Jaime Félix
Guerrero Isuiza, Mateo

Fecha del Experimento : Lunes, 24 y 25 de Septiembre del 2018

Hora : De 9:30 am a 11:00 am

Fecha de entrega del informe : Martes, 25 de Septiembre del 2018

Hora : De 8:00 am a 6:00 pm

2018 - III



Contenido

Introducción.....	3
Capítulo 1: Compendio Teórico	4
1.1. Lenguaje C.....	4
1.2. MPLAB	4
1.3. PicKit.....	4
1.4. Copilador XC8.....	4
1.5. Microcontrolador	4
Capítulo 2: Resultados Obtenidos	6
2.1. Indicador de frases por combinación	6
2.1.1. Diagrama de flujo	6
2.1.2. Código	6
2.1.3. Lenguaje y Simulación	8
2.1.4. Circuito Real	11
Link del Proyecto.....	12
Observaciones	13
Conclusiones.....	14
Bibliografía.....	15

Introducción

El interés del grupo de investigación se basó a partir de la observación de los distintos programas que se realizaron en las clases de Microcontroladores. En consecuencia, nos planteamos la interrogante: ¿En qué medida es posible obtener nombres a partir de una entrada mediante el PIC18F4550? Consideramos que es posible construir un mecanismo que permita al PIC18F4550 mostrar una palabra en un display letra por letra a partir de la lectura valor de entrada, que en este caso fue un switch.

De esta manera, el informe se dividió en cinco partes. En primer lugar se detallan los conceptos de los sistemas o circuitos usados en el proyecto. En segundo lugar se presenta la elaboración del proyecto: El diagrama de flujo, el código de programación en Lenguaje C, las fotos de las simulaciones en Mplab y Proteus, y finalmente las fotos del circuito real programado con el PicKIT 3. En tercer lugar, se muestra el link del video en Youtube. En cuarto lugar, se muestra las observaciones obtenidas en el proyecto. Finalmente, se muestran las conclusiones.

La importancia del proyecto radica en la relevancia de las aplicaciones del PIC en la lectura y escritura de datos a partir de instrucciones dada por el usuario. Siendo esta proyecto un medio eficaz para evidenciar las funcionalidades de los microcontroladores para la creación de sistemas y/o mecanismos capaces de realizar funciones precisas a partir de instrucciones y disminuir los riesgos del error humano en procesos repetitivos. Del mismo modo, el proyecto me permite desarrollar habilidades para la programación y construcción de sistemas eléctricos que serán de gran utilidad en mi carrera profesional como Ingeniero egresado de UTP.

Capítulo 1: Compendio Teórico

1.1. Lenguaje C

El lenguaje C, es un lenguaje que revoluciono ya que debido a su sencillez, y a su tamaño, pero que sobre todo no está dentro de una aplicación en específica, lo hace más potente. C trabaja con tipos de datos que son directamente tratables por el hardware de la mayoría de computadoras actuales, como son los caracteres, números y direcciones. Estos tipos de datos pueden ser manipulados por las operaciones aritméticas que proporcionan las computadoras.

1.2. MPLAB

Este es un programa que sirve para editar distintos micro controladores que este mismo programa puede soportar, sin embargo una de las cosas importantes es que se puede grabar circuitos integrados. Asimismo cabe recalcar que se debe a empezar a escribir el programa, respetando la directiva y/o parámetros necesarios para compilarlo y grabarlo al chip. Definimos directivas como palabras específicas o palabras que han sido separadas para ordenarle al Mplab que funciones debe configurar a la hora de compilar nuestro programa.

1.3. PicKit

El pickit hace posible la programación de micro controlador utilizando todo lo que incluye en el armado del código en el Mplab, desarrollado por la empresa Microchip. Este pickit es muy fácil de conectar ya que solo se conecta mediante una interfaz USB.

1.4. Copilador XC8

Un compilador es un programa informático que permite que el lenguaje avanzado usado por el programador (Lenguaje C, BASIC, etc) sea traducido al lenguaje de máquina, el único que puede leer el microcontrolador. El compilador XC8 es uno de los compiladores de la línea de compiladores MPLAB XC de Microchip, el cual es compatible con los microcontroladores PIC de 8 bits, utilizado en este proyecto.

1.5. Microcontrolador

Un microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y

ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas dentro del microcontrolador, y en conjunto forman lo que se le conoce como microcomputadora. Se puede decir con toda propiedad que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado.

Inicialmente todos los microcontroladores adoptaron la arquitectura clásica de Von Neumann, la cual se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de manera indistinta.

Después, se empezó a realizar microcontroladores con la nueva arquitectura Harvard, la cual se caracteriza por disponer de dos memorias independientes para datos y para instrucciones, permitiendo accesos simultáneos.

Estos microcontroladores necesitan ser programados para poder realizar las funciones. Debido a la gran dificultad de programar los microcontroladores en el código binario, la programación comúnmente se lleva a cabo a través de un lenguaje de alto nivel, es decir, un lenguaje que utilice frases o palabras semejantes o propias del lenguaje humano. Lenguajes como el C, utilizado en nuestro caso, o BASIC son comúnmente utilizados en la programación de los microcontroladores.

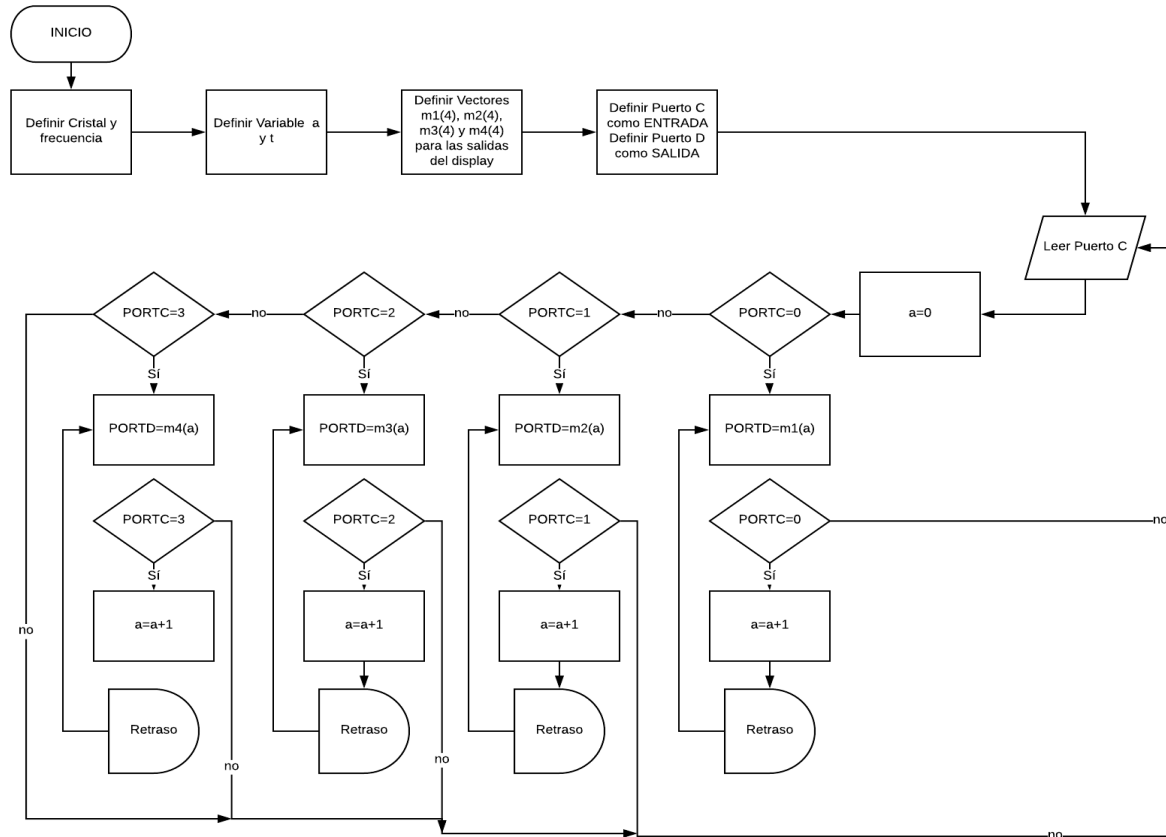
1.6. Display 7 segmentos

Un microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas dentro del microcontrolador, y en conjunto forman lo que se le conoce como microcomputadora. Se puede decir con toda propiedad que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado.

Capítulo 2: Resultados Obtenidos

2.1. Indicador de frases por combinación

2.1.1. Diagrama de flujo



2.1.2. Código

```
#define _XTAL_FREQ 4000000 //Definir el cristal y su frecuencia

int main (int argc, char** argv)
{
    int a,t; // Definir "a" y "t" como entero
    int m1[4]={0x0C,0x79,0x12,0x40}; // Definir vector m1 con 4 valores
    int m2[4]={0x47,0x41,0x2B,0x08}; // Definir vector m3 con 4 valores
    int m3[4]={0x07,0x2F,0x06,0x12}; // Definir vector m2 con 4 valores
    int m4[4]={0x21,0x79,0x40,0x12}; // Definir vector m4 con 4 valores

    TRISD=0; // Definir puerto D como salida
    TRISC=3; // Definir puerto C como entrada módulo 3

    INICIO: // Definir etiqueta INICIO
```



```
if(PORTC==0)goto UNO; //Condicional que dirige a parámetro UNO
if(PORTC==1)goto DOS; //Condicional que dirige a parámetro DOS
if(PORTC==2)goto TRES; //Condicional que dirige a parámetro TRES
if(PORTC==3)goto CUATRO; //Condicional que dirige a CUATRO
```

```
UNO: //Definir parámetro UNO
for (a=0;a<4;a++) // Establecer for como contador
{
    PORTD= m1[a]; // Salida es el valor a del vector m1
    if(PORTC==1)goto DOS; //Condicional que dirige a DOS
    if(PORTC==2)goto TRES; //Condicional que dirige a TRES
    if(PORTC==3)goto CUATRO; //Condicional que dirige a CUATRO
    for (t=0;t<10;t++) // Establecer for como contador
    {
        __delay_ms(100); // Definir retraso 100 microsegundos
    }
}
goto INICIO; // Ir al parámetro INICIO
```

```
DOS: //Definir parámetro DOS
for (a=0;a<4;a++) // Establecer for como contador
{
    PORTD= m2[a]; // Salida es el valor a del vector m2
    if(PORTC==0)goto UNO; //Condicional que dirige a UNO
    if(PORTC==2)goto TRES; //Condicional que dirige a TRES
    if(PORTC==3)goto CUATRO; //Condicional que dirige a CUATRO
    for (t=0;t<10;t++) // Establecer for como contador
    {
        __delay_ms(100); // Definir retraso 100 microsegundos
    }
}
goto INICIO; //Dirigir a parámetro INICIO
```

```

TRES:                                     //Definir parámetro TRES
for (a=0;a<4;a++)                         // Establecer for como contador
{
    PORTD= m3[a];                         // Salida es el valor a del vector m1
    if(PORTC==0)goto UNO; //Condicional que dirige a UNO
    if(PORTC==1)goto DOS; //Condicional que dirige a DOS
    if(PORTC==3)goto CUATRO; //Condicional que dirige a CUATRO
    for (t=0;t<10;t++)                    // Establecer for como contador
    {
        __delay_ms(100); // Definir retraso 100 microsegundos
    }
}
goto INICIO;                             //Dirigir a parámetro INICIO

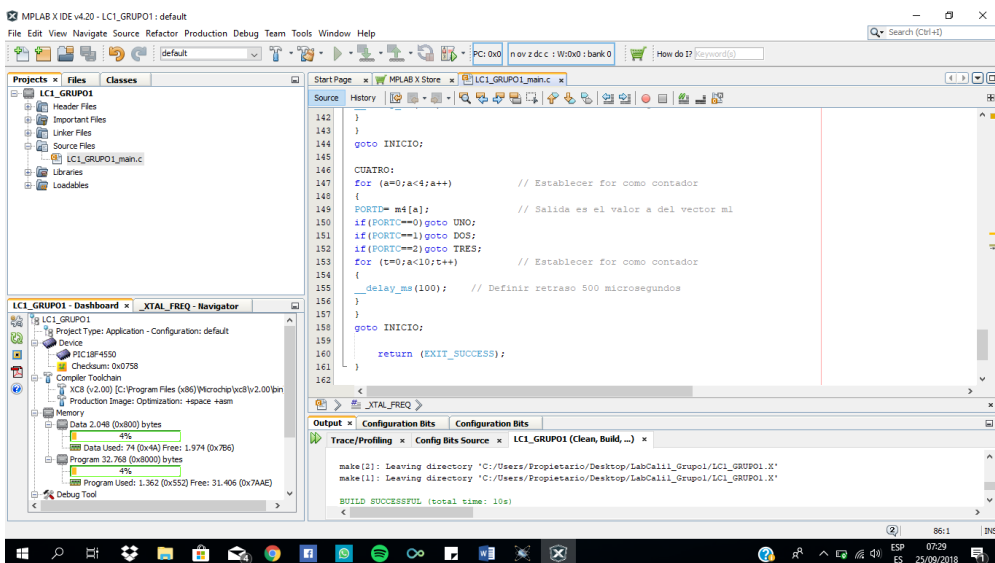
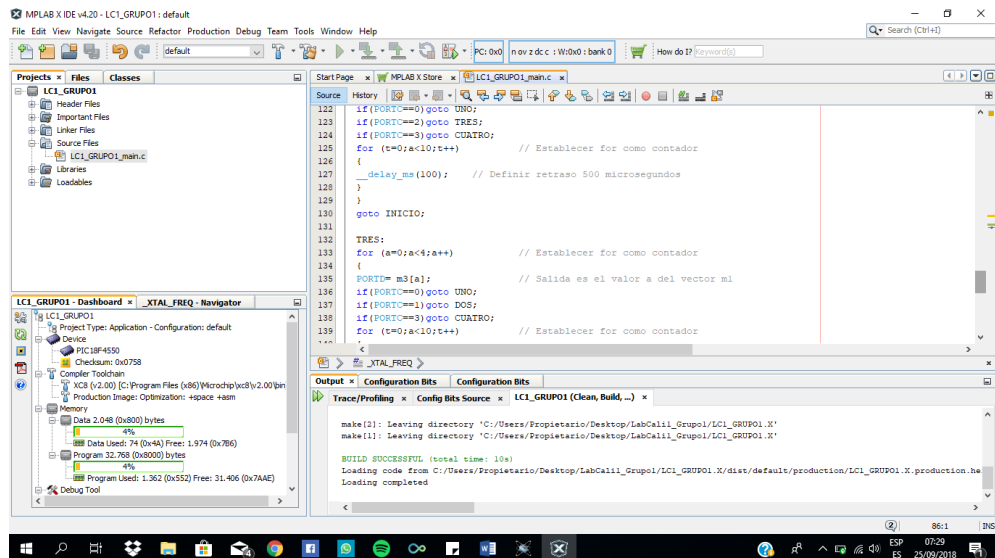
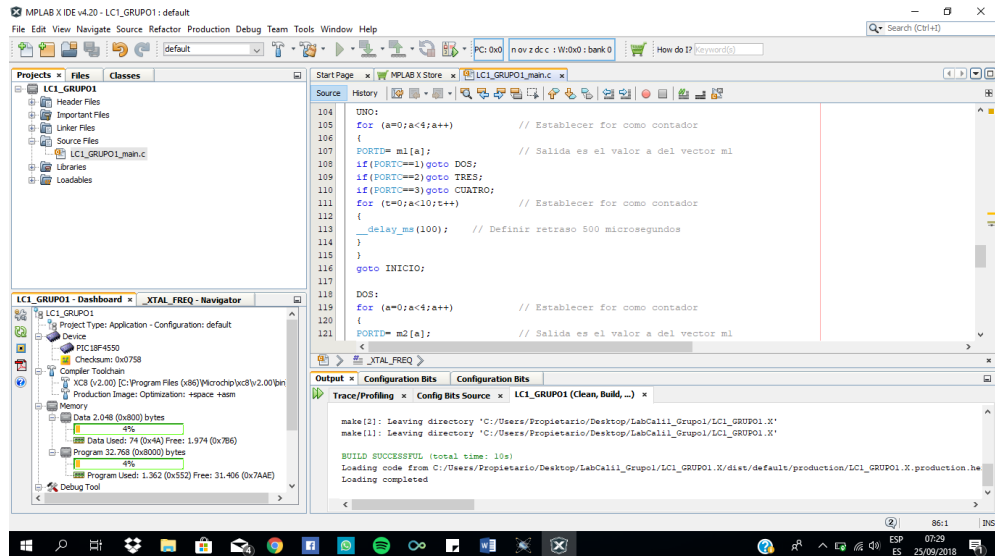
CUATRO:                                   //Definir parámetro CUATRO
for (a=0;a<4;a++)                         // Establecer for como contador
{
    PORTD= m4[a];                         // Salida es el valor a del vector m1
    if(PORTC==0)goto UNO; //Condicional que dirige a UNO
    if(PORTC==1)goto DOS; //Condicional que dirige a DOS
    if(PORTC==2)goto TRES; //Condicional que dirige a TRES
    for (t=0;t<10;t++)                    // Establecer for como contador
    {
        __delay_ms(100); // Definir retraso 100 microsegundos
    }
}
goto INICIO;                             //Dirigir a parámetro INICIO
return (EXIT_SUCESS);

}

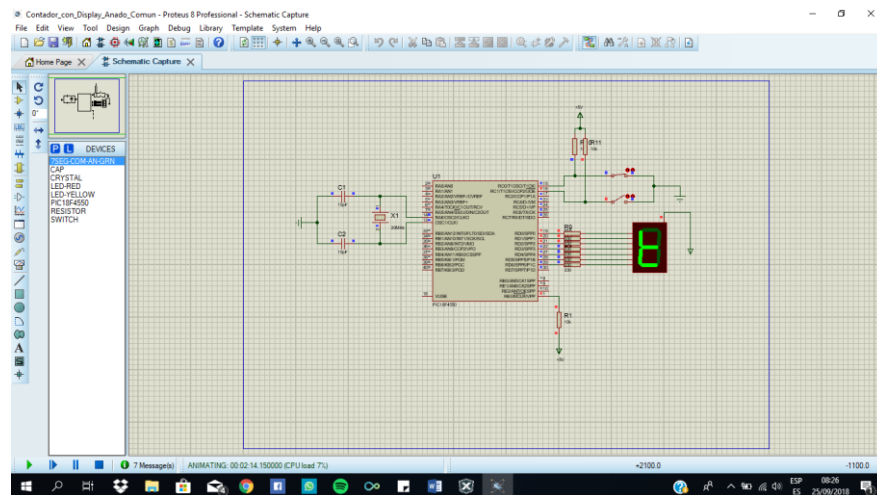
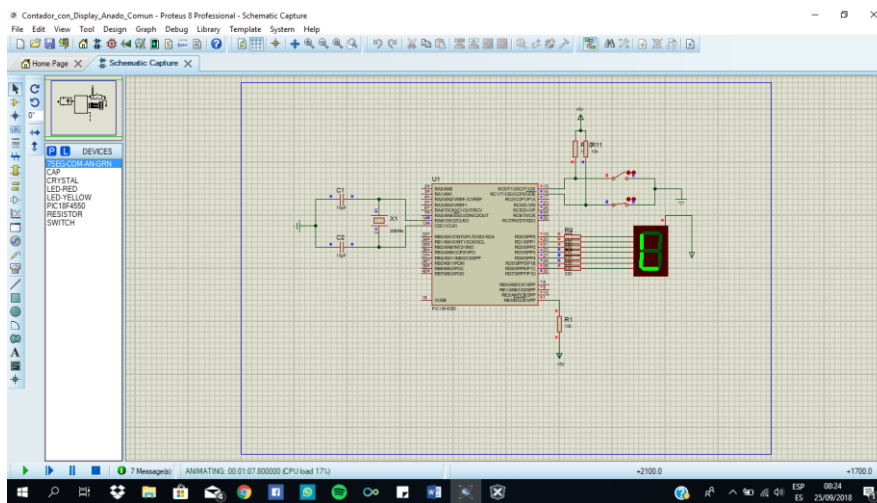
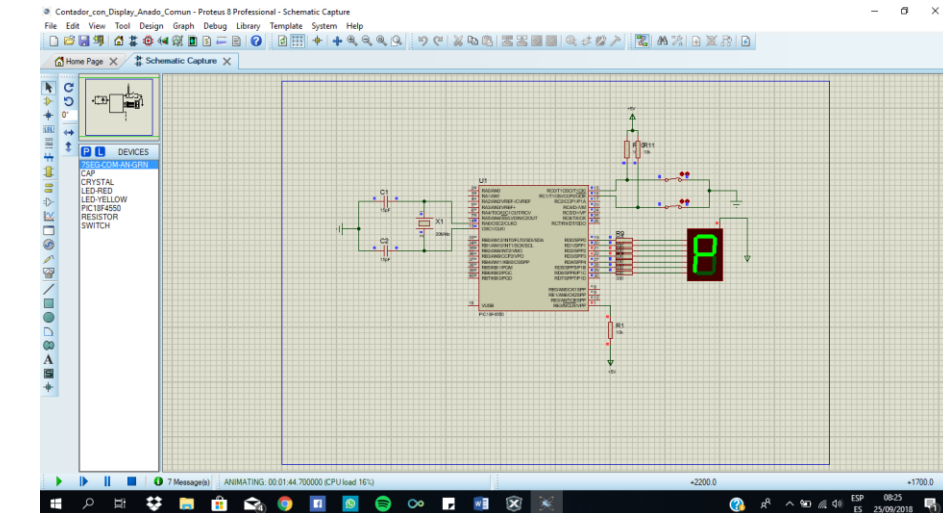
```

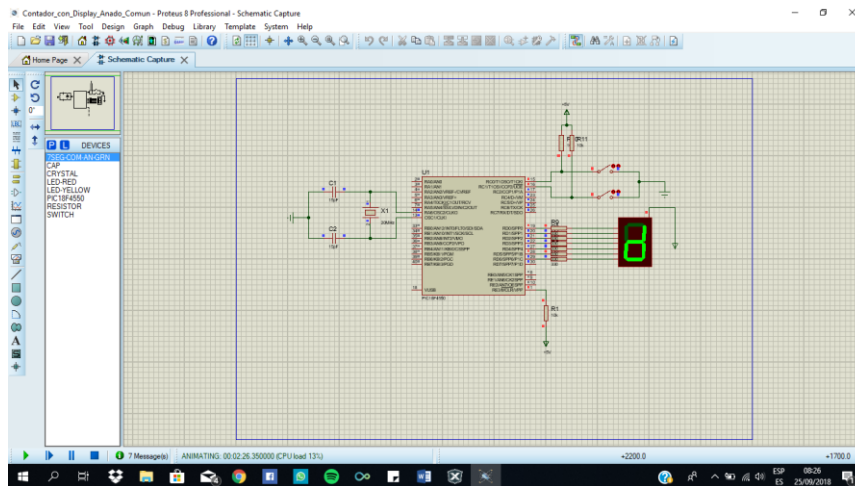
2.1.3. Lenguaje y Simulación

A continuación se muestra el código escrito en MPLAB V4.20:



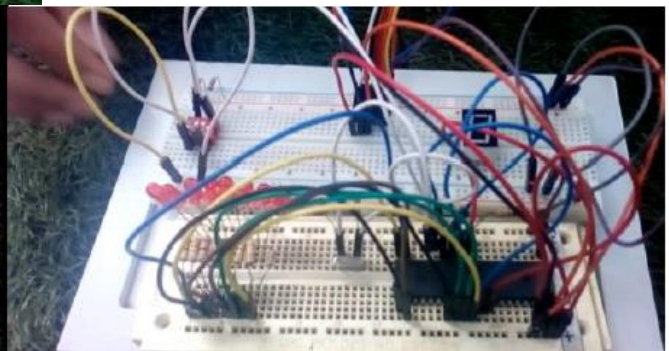
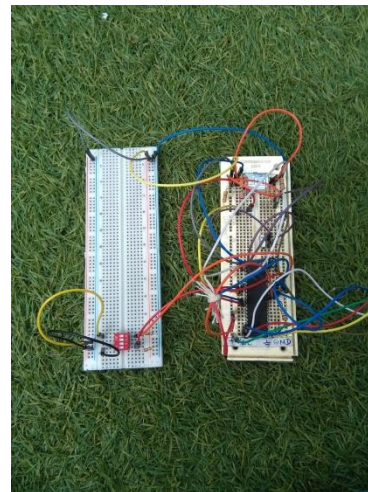
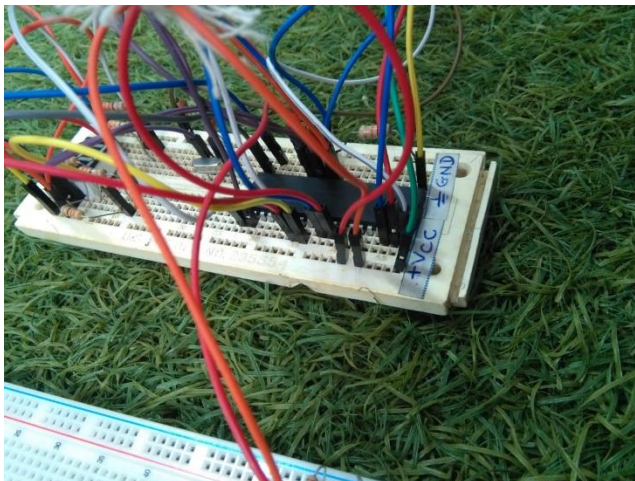
Del mismo modo, a continuación se muestra la simulación del circuito construido en Proteus 8:





2.1.4. Circuito Real

Para la primera simulación se realizó un circuito en Protoboard conectado a base de la simulación del Proteus y el manual del PIC18F4550:





Link del Proyecto

[https://www.youtu](https://www.youtube.com/watch?v=_u2VdnIEptY)
[be.com/watch?v=_](https://www.youtube.com/watch?v=_u2VdnIEptY)
[u2VdnIEptY](https://www.youtube.com/watch?v=_u2VdnIEptY)

Observaciones

- Al realizar el cambio del valor de entrada, en los switch, el cambio de la palabra no es inmediato debido a que se a colocado un delay de 1s entre cada letra, como indica el calificado. Por ello, al reducir el tiempo entre cada letra el cambio entre cada palabra es más rápido.
- Durante la realización del circuito en el laboratorio, el switch utilizado no entraba perfectamente en los orificios del protoboard lo cual generaba que la entrada no sea la requerida.
- Se configuró una función de salida especial para el display con la finalidad de que se mostrará el valor deseado.
- Para poder usar el retraso “delay” en microsegundos, fue necesario definir el cristal y su frecuencia

Conclusiones

- La configuración de los bits del PIC18F4550, en MPLABX, es esencial en la simulación ya que de hacerlo de manera correcta nos la facilitara, pero haciéndolo mal generará un error.
- Los vectores, punteros y funciones fueron fundamental en la escritura del programa del PIC para lograr el funcionamiento deseado.
- Es necesario definir el cristal y su frecuencia para usar el retraso delay.
- Al usar un display, se debe crear una función de salida especial para que se muestren los valores deseados, ya que el display posee una combinación de bits que no necesariamente coincide con el valor que se desea mostrar.
- Es fundamental escribir cada detalle del lenguaje C del programa de manera precisa para evitar errores.



Bibliografía

- MICROCHIP (2018) (<https://www.microchip.com/>) Sitio web oficial de Microchip; contiene información acerca del compilador XC8 y de MPLABX.
- ELECTRONICA ESTUDIO (2018)
(<http://www.electronicaestudio.com/index.htm>) Sitio web oficial de Electrónica estudio; contiene información acerca de los microcontroladores.
- WIKIPEDIA (2018)(<https://es.wikipedia.org/wiki/Wikipedia:Portada>) Sitio web oficial de Wikipedia; contiene información de interés.