



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO.
FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN.



Proyecto: Sensor de frutas.

Autor: Chávez Ortiz Saúl David.

Docente: Arcelia Bernal Díaz.

Microprocesadores y Microcontroladores.

Facultad de Estudios Superiores Aragón (FES Aragón).

CDMX a 29 de noviembre del 2023.

RESUMEN.

En este proyecto se trabajo un sensor que identifica una fruta. Este se realizo con el microcontrolador PIC16F887, conectado a un sensor de color, el cual mide las frecuencias de los colores primarios RGB de la fruta, y pasa las frecuencias al microcontrolador para identificar la fruta. El circuito también contiene un led RGB, que prendera de acuerdo al color de la fruta. Los resultados son mostrados por una LCD que igual esta conectada al PIC, y que además el circuito contendrá un modulo HC-06 Bluetooth, el cual se comunicará con el celular para recibir los datos.

INTRODUCCIÓN.

Para demostrar lo aprendido en esta materia de “Microcontroladores y Microprocesadores”, debemos de desarrollar un proyecto, en el cual se deben de aplicar componentes como: un sensor, una pantalla LCD para mostrar los datos y resultados, un módulo de comunicación para enviar los resultados o datos a un celular y como es obvio, se debe de usar un microcontrolador para poder controlar estos componentes. El microcontrolador será programado en el

lenguaje C. Todo esto es lo que vamos aplicar para realizar nuestro proyecto, el sensor de frutas.

JUSTIFICACIÓN.

Para poder realizar este proyecto, 1ro se hizo una investigación, se busco encontrar un sensor que identifique cosas, haciendo la investigación, se hallo el sensor de color el cual obtiene las frecuencias de los colores primarios. Entonces surgió la idea de usarlo, ya que, podemos leer las frecuencias de los colores primarios de las frutas e identificarlas con ayuda del sensor de color.

Para poder trabajar y controlar este sensor, necesitamos un microcontrolador, por lo que, para este proyecto se trabajara con el PIC16F887. Los demás componentes que utilizaremos para el proyecto son los siguientes (se agrega el PIC y el sensor):

- PIC16F887.
- Sensor de color TCS230.
- Oscilador de cristal de 20Mhz.
- 2 capacitores de 22pf.
- LCD

- Módulo HC-06 Bluetooth.
- Potenciómetro de 10k ohms.
- Led RGB.
- 1 resistencia de 1k ohm, 3 resistencias de 330 ohms.
- Cable de calibre 22 de varios colores (para las conexiones).
- Jumpers.
- Programador Pic Kit 3 (para cargar los programas al microcontrolador).
- Software PIC C compiler (para programar).

DESARROLLO

Para poder desarrollar este proyecto, primero debemos de buscar los diagramas de circuito, para saber cómo deben de ir las conexiones de cada uno de los componentes. Las siguientes imágenes proporcionan el esquema del microcontrolador y cada uno de sus pines, y demuestran como debe de ir cada conexión de cada componente con los pines del microcontrolador, se ha de aclarar que, en las imágenes algunos componentes se conectan en los mismos puertos o pines, pero en el código y en la realización de este proyecto se configuraran puertos distintos para cada componente.

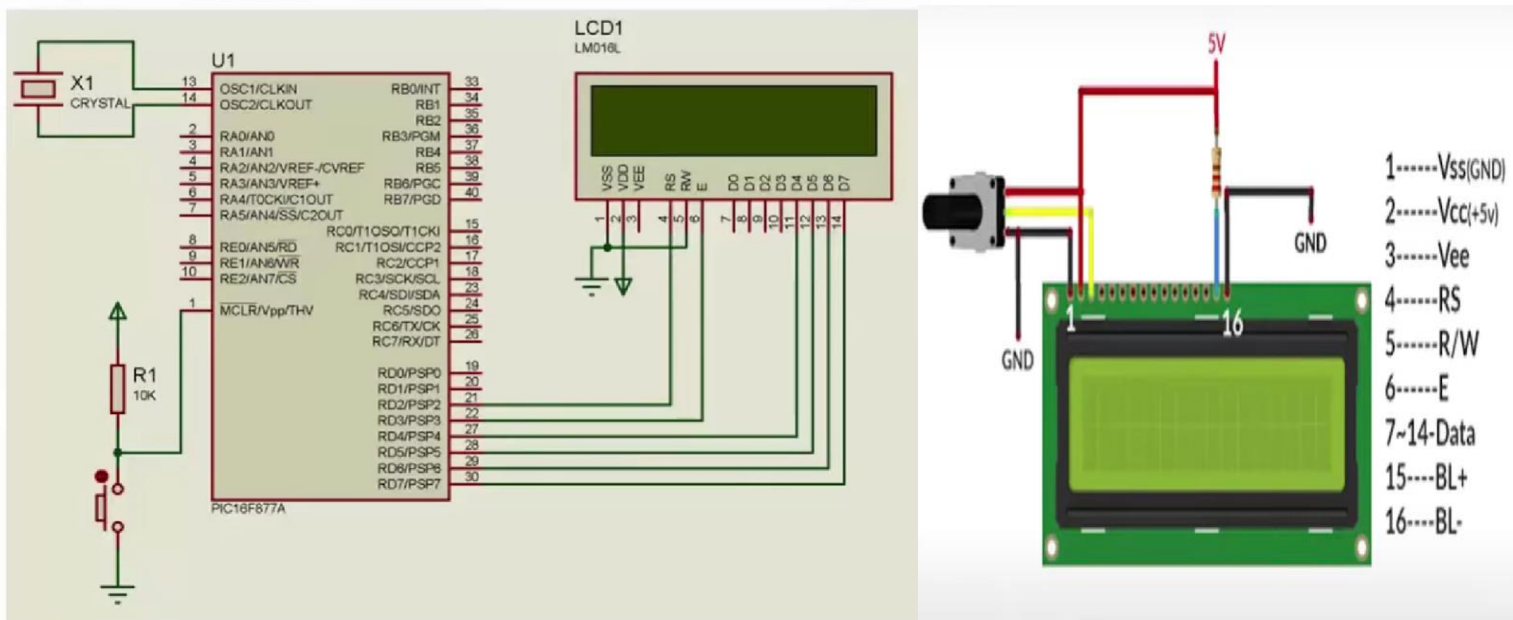


Imagen 1. Conexión de la LCD con el microcontrolador, conexiones de los demás pines de la LCD y la conexión con el potenciómetro.

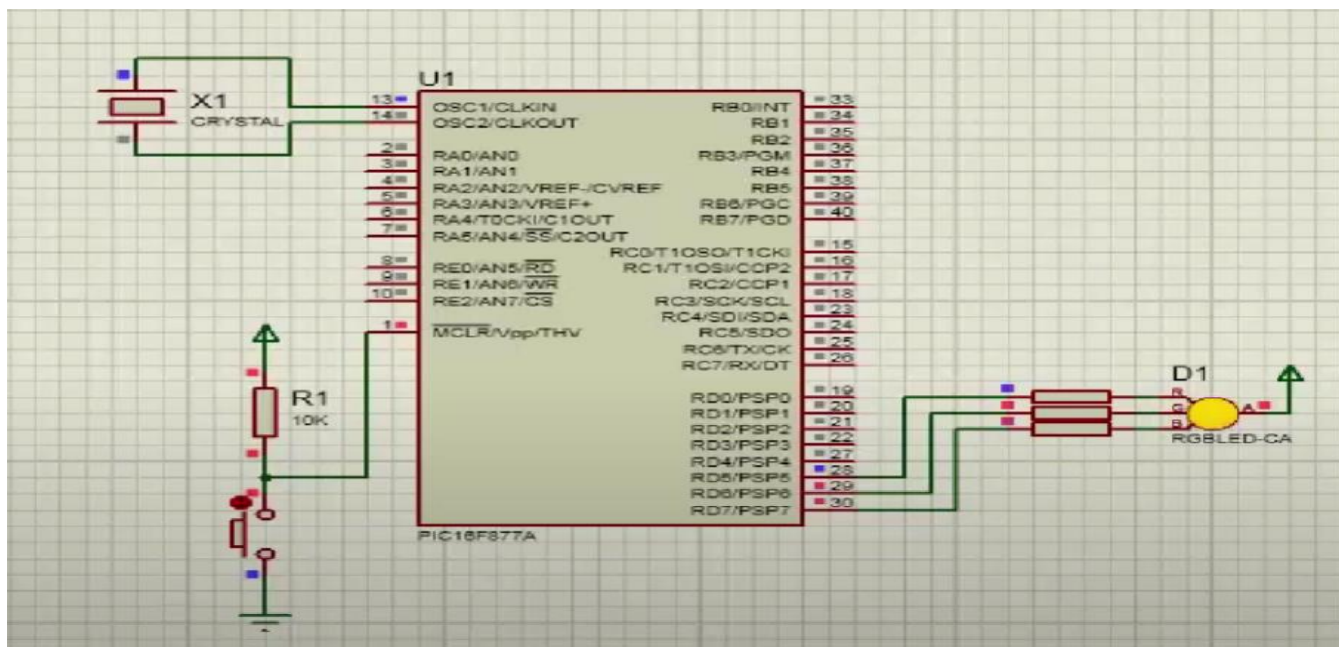


Imagen 2. Conexión del led RGB con el microcontrolador.

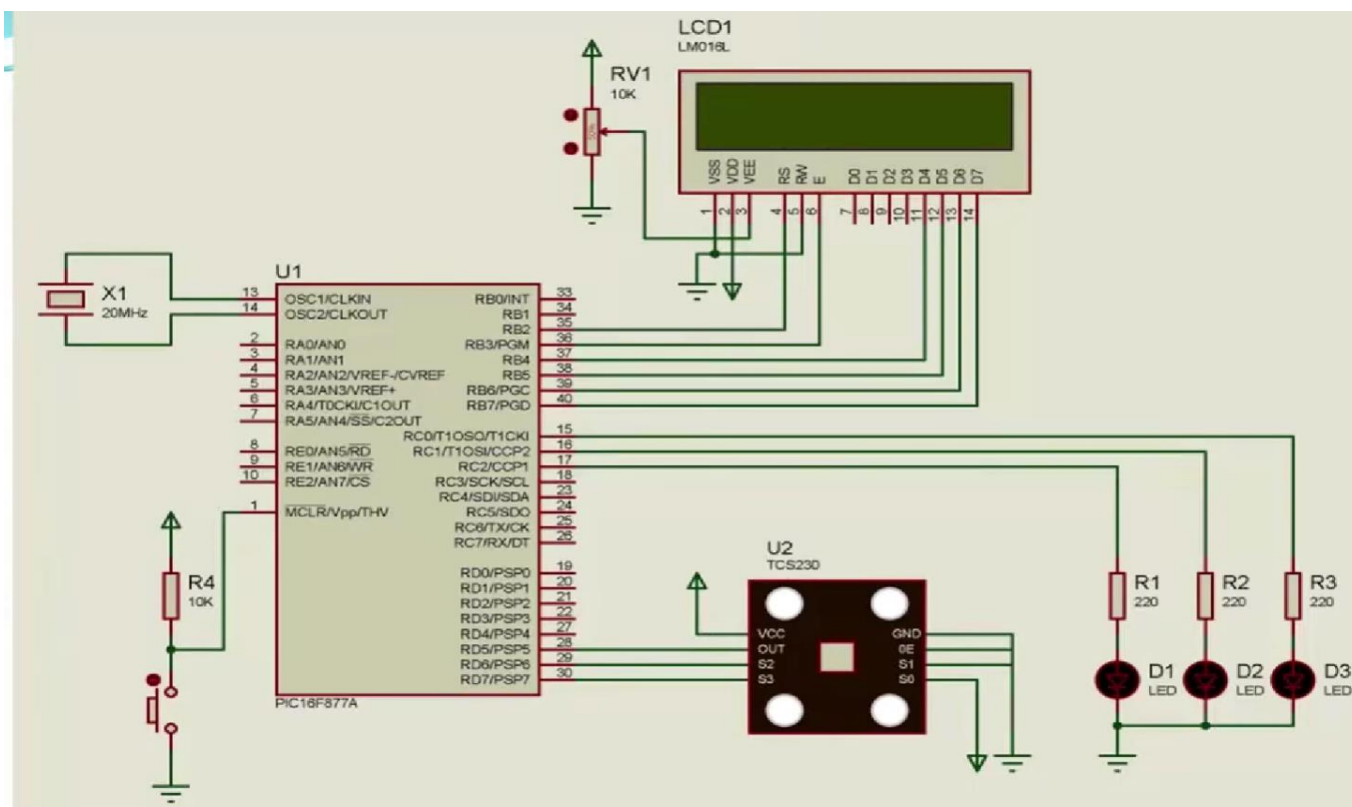


Imagen 3. Conexión del sensor con el microcontrolador.

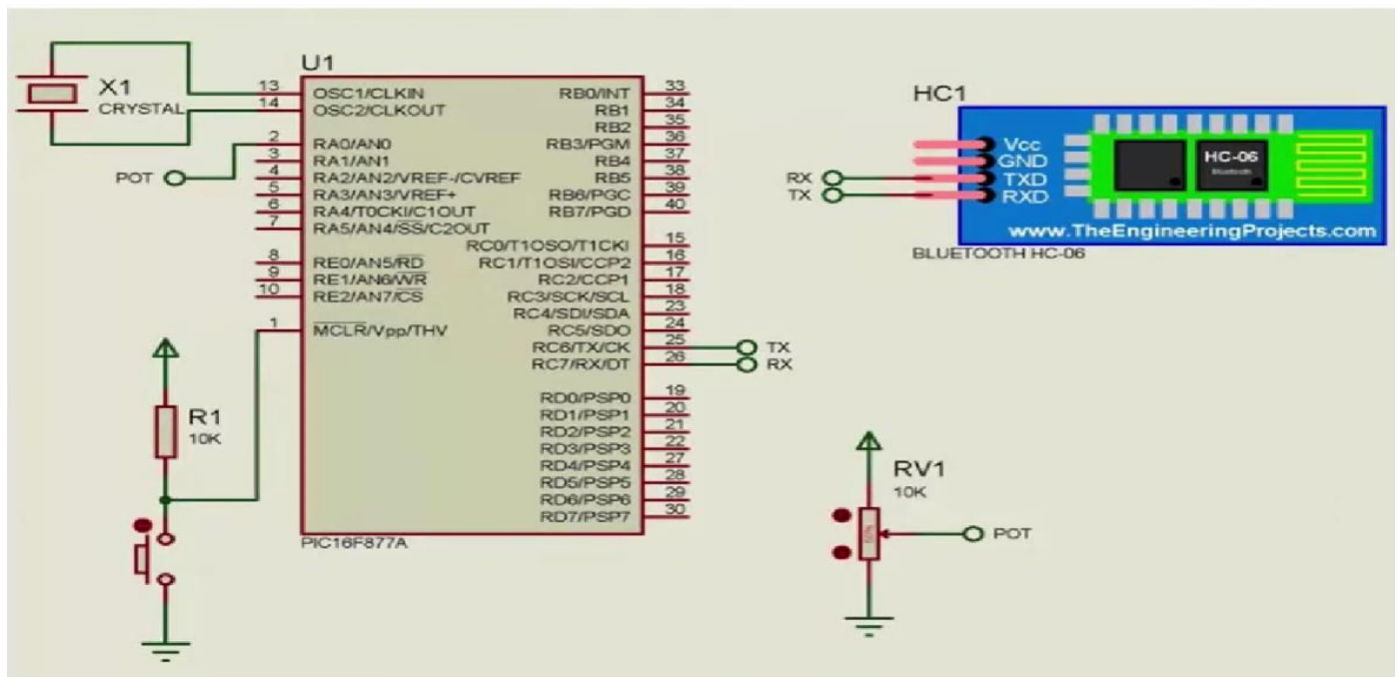


Imagen 4. Conexión del módulo HC-06 Bluetooth.

Teniendo los diagramas de conexión para saber como realizar las conexiones, ahora procederemos a realizar el código en el PIC C compiler. El código ya con los componentes programados queda de la siguiente manera:

```
#include <16f887.h>
#fuses HS, NOWDT, NOPROTECT, NOPUT, NOLVP, BROWNOUT
#use delay(clock=20M)
//CONFIGURACIÓN PARA LA COMUNICACIÓN DEL MÓDULO USART
#use RS232(BAUD=9600, BITS=8, PARITY=N, XMIT=PIN_C6, RCV=PIN_C7)
#include <stdio.h>
//INSTRUCCIONES PARA UTILIZAR LOS PUERTOS EN LAS DEFINICIONES
#use standard_io(D)
#use standard_io(C)
#use standard_io(B)

//CONFIGURACIÓN DE LA PANTALLA LCD
#define LCD_DB4 PIN_D4
#define LCD_DB5 PIN_D5
#define LCD_DB6 PIN_D6
#define LCD_DB7 PIN_D7
```

```

#define LCD_RS    PIN_D2
#define LCD_E     PIN_D3
#include <LCD_16X2.c>

//CONFIGURACION DEL LED RGB
#define led_r PIN_C0
#define led_g PIN_C1
#define led_b PIN_C2
#include <RGB_Control.c>

//CONFIGURAMOS EL SENSOR
#define P_TCS230 PIN_B7
#define S2 PIN_B6
#define S3 PIN_B5
#include <TCS230.c>

void main()
{
    //VARIABLES PARA GUARDAR LAS FRECUENCIAS DE CADA COLOR QUE CAPTA EL
    SENSOR
    long red = 0;
    long green = 0;
    long blue = 0;

    //VARIABLES PARA LOS RESULTADOS DE LA FRUTA IDENTIFICADA.
    char manzana[] = "Manzana";
    char fresa[] = "Fresa";
    char naranja[] = "Naranja";
    char mandarina[] = "Mandarina";

    //INICIALIZAMOS LA LCD, EL SENSOR Y EL LED RGB
    lcd_init();
    TCS230_init();
    rgb_init();
    rgb_write_value(0, 0, 0, COMMON_CATHODE); //INICIAMOS EL LED APAGADO

    while(true)
    {
        //ASIGNAMOS LAS FRECUENCIAS A NUESTRAS VARIABLES
        red = TCS230_getFrequency_red();
        green = TCS230_getFrequency_green();
        blue = TCS230_getFrequency_blue();
    }
}

```



```

//MOSTRAMOS EN PANTALLA LAS FRECUENCIAS DE LOS COLORES
lcd_clear();
lcd_gotoxy(1,1);
printf(lcd_putc, "R:%Lu", red);
lcd_gotoxy(6,1);
printf(lcd_putc, "G:%Lu", green);
lcd_gotoxy(12,1);
printf(lcd_putc, "B:%Lu", blue);
delay_ms(350);

//FRECUENCIAS DE MANZANA
if((red >= 14 && red <= 44) && (green >= 69 && green <= 114) && (blue
>= 45 && blue <= 90))
{
    //MOSTRAMOS EN PANTALLA LA FRUTA DETECTADA Y ENCENDEMOS EL LED RGB CON
EL COLOR DE LA FRUTA.
    lcd_gotoxy(1, 2);
    lcd_putc("Fr: Manzana");
    rgb_write_value(196, 2, 2, COMMON_CATHODE);
    printf("%s\r\n", manzana); //ENVIA LA FRUTA DETECTADA AL CELULAR
    delay_ms(500);
    lcd_clear();
}

//FRECUENCIA MANDARINA
if((red >= 0 && red <= 42) && (green >= 38 && green <= 44) && (blue >=
45 && blue <= 70))
{
    //MOSTRAMOS EN PANTALLA LA FRUTA DETECTADA Y ENCENDEMOS EL LED
RGB CON EL COLOR DE LA FRUTA.
    lcd_gotoxy(1, 2);
    lcd_putc("Fr: Mandarina");
    rgb_write_value(255, 89, 1, COMMON_CATHODE);
    printf("%s\r\n", mandarina); //ENVIA LA FRUTA DETECTADA AL CELULAR
    delay_ms(500);
    lcd_clear();
}

//FRECUENCIA NARANJA
if((red >= 43 && red <= 69) && (green >= 45 && green <= 75) && (blue
>= 46 && blue <= 90))
{
    //MOSTRAMOS EN PANTALLA LA FRUTA DETECTADA Y ENCENDEMOS EL LED
RGB CON EL COLOR DE LA FRUTA.
    lcd_gotoxy(1, 2);

```

```

        lcd_putc("Fr: Naranja");
        rgb_write_value(5, 162, 22, COMMON_CATHODE);
        printf("%s\r\n", naranja); //ENVIA LA FRUTA DETECTADA AL CELULAR
        delay_ms(500);
        lcd_clear();
    }

    //FRECUENCIA FRESA
    if((red >= 45 && red <= 70) && (green >= 95 && green <= 137) && (blue
    >= 87 && blue <= 119))
    {
        //MOSTRAMOS EN PANTALLA LA FRUTA DETECTADA Y ENCENDEMOS EL
        LED RGB CON EL COLOR DE LA FRUTA.
        lcd_gotoxy(1, 2);
        lcd_putc("Fr: Fresa");
        rgb_write_value(220, 16, 2, COMMON_CATHODE);
        printf("%s\r\n", fresa); //ENVIA LA FRUTA DETECTADA AL CELULAR
        delay_ms(500);
        lcd_clear();
    }

}

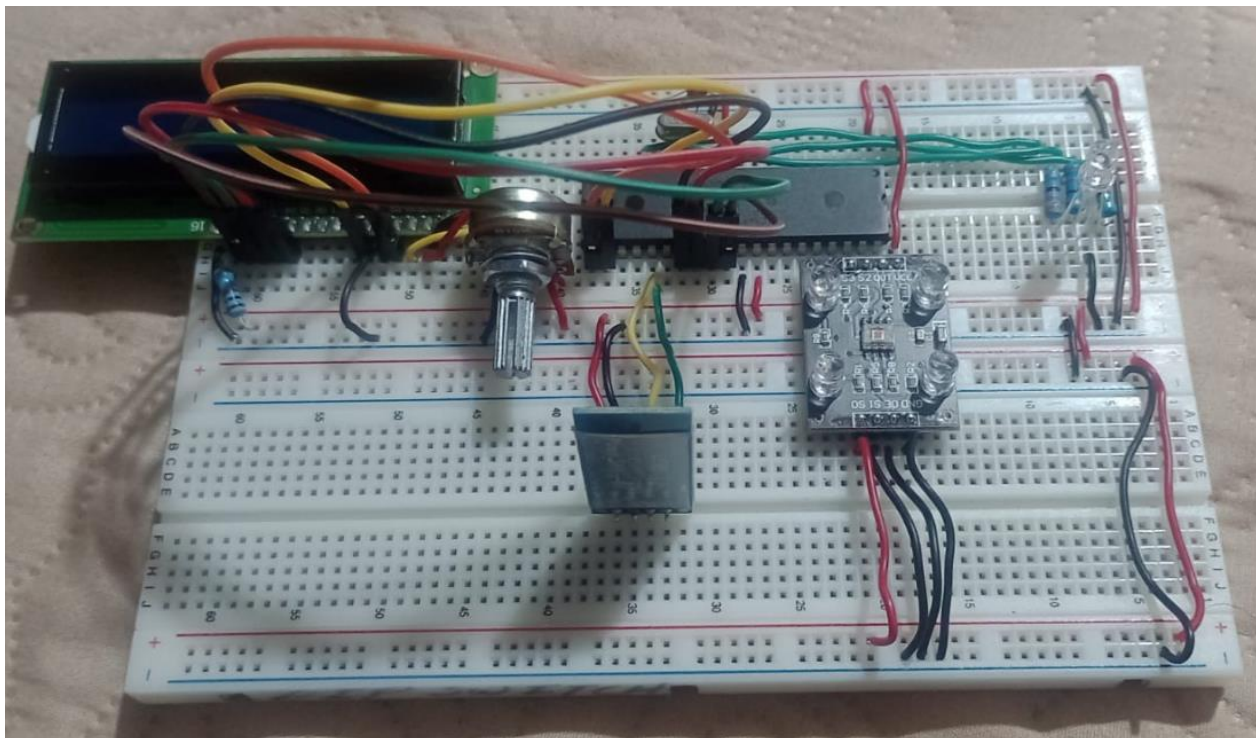
}

```

A grandes rasgos el código contiene las librerías para poder utilizar y controlar los componentes, librerías como la del sensor TCS230, LCD16x2, RGB, y la librería `stdio.h` para poder configurar el modulo “USART” para la comunicación Bluetooth. El código contiene las configuraciones de los pines del PIC para las conexiones de los componentes. La lógica del programa se encuentra en las 4 condicionales IF’S que contienen una expresión booleana

larga, ya que se asegura de que las frecuencias de la fruta correspondan a la fruta, nuestro programa puede detectar en este caso 4 frutas, una manzana, una fresa, una mandarina y una naranja.

Tomando en cuenta las configuraciones de cada uno de los pines y puertos del microcontrolador con los componentes, nuestro circuito alambrado queda de la siguiente manera:



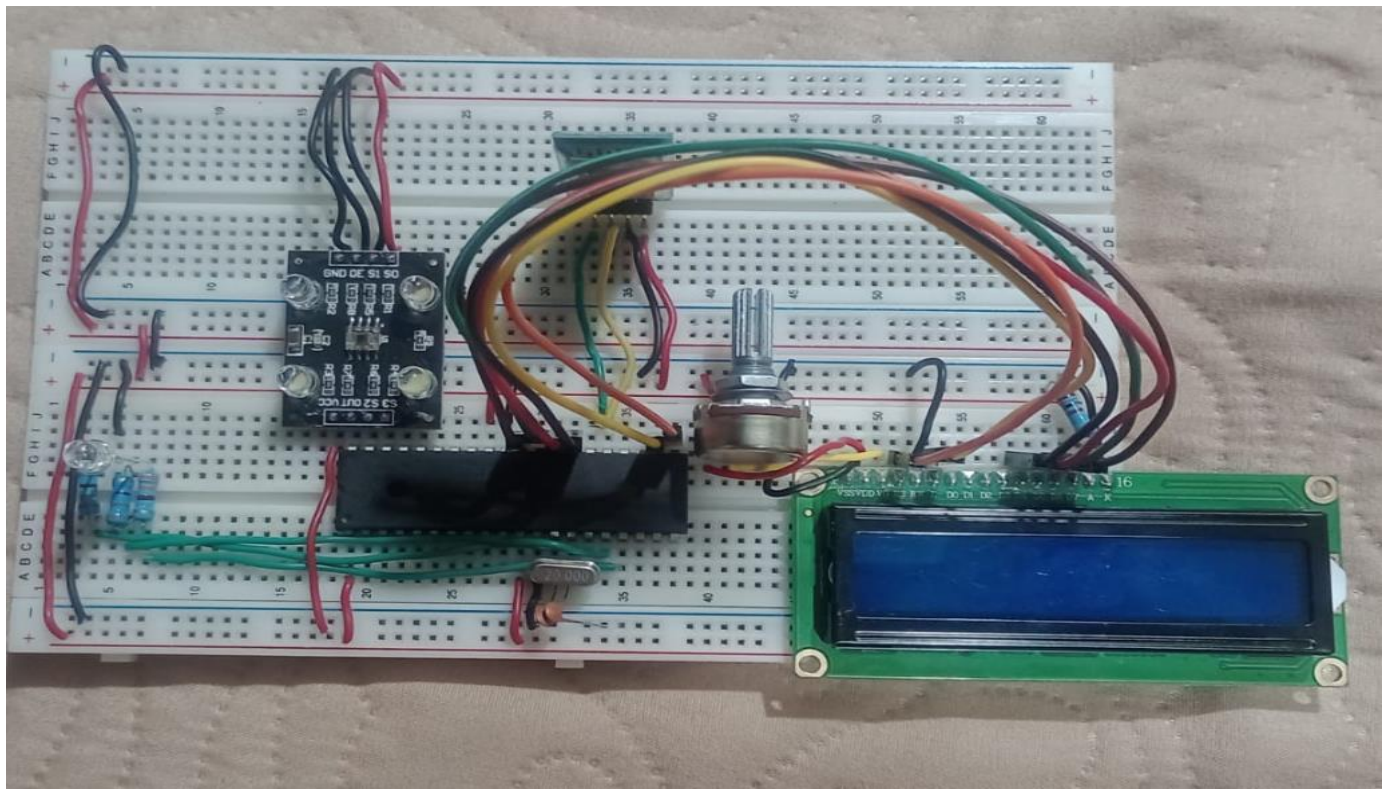


Imagen 5 y 6. Circuito sensor de frutas.

RESULTADO.

Probando el circuito, verificamos su funcionamiento, con la fruta manzana, veamos los resultados:

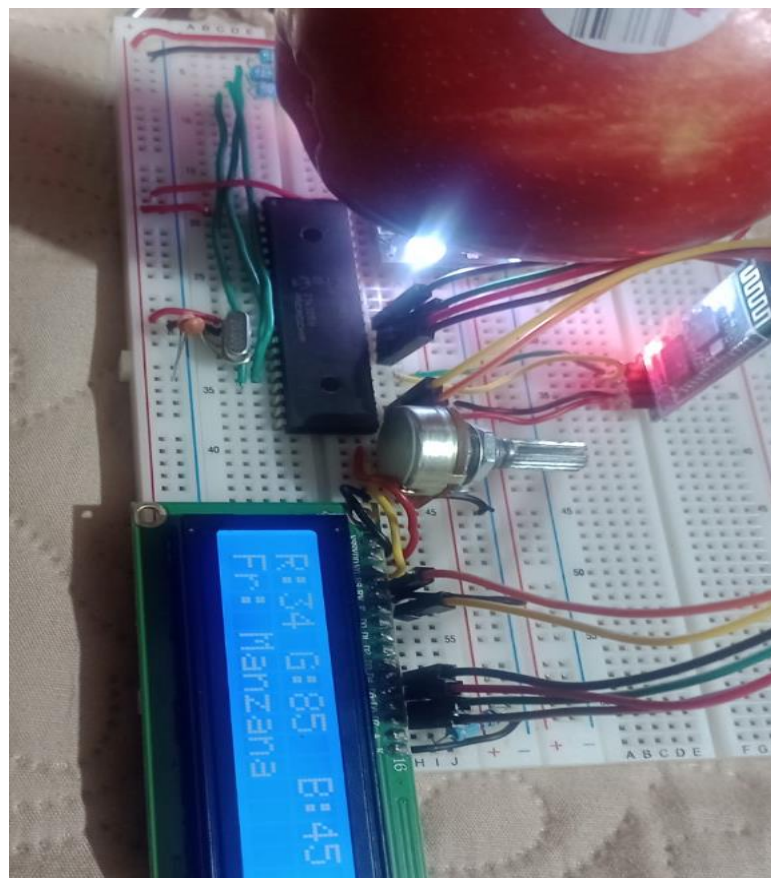
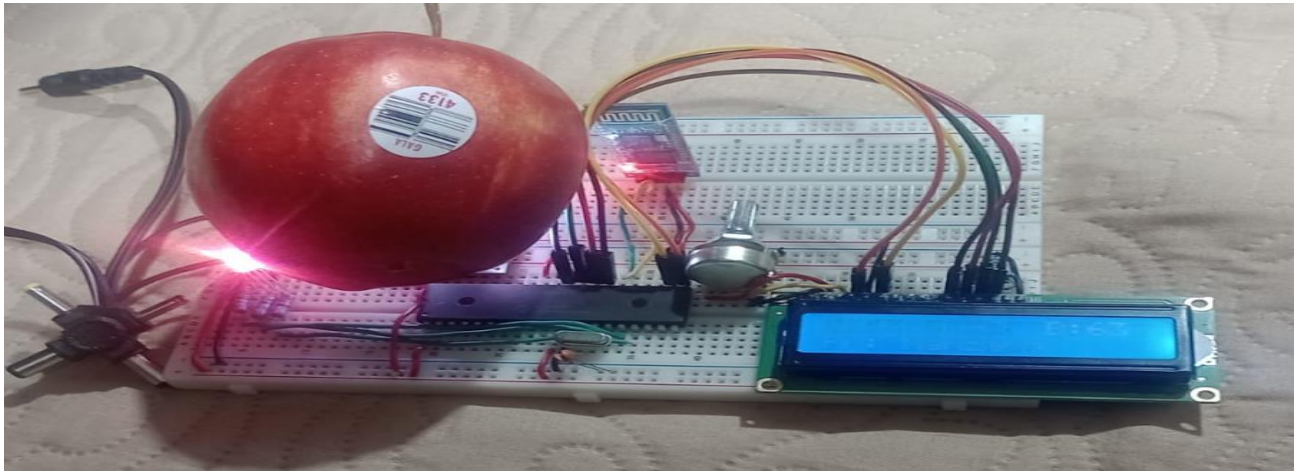


Imagen 7 y 8. Probando el sensor de frutas.

Como podemos observar, colocando la manzana en el sensor de color, vemos que la LCD nos arroja las frecuencias de la fruta y nos dice que es una manzana. Ahora con ayuda de la aplicación “Serial Bluetooth Terminal”, nos conectaremos con el módulo HC-06, y veremos que el microcontrolador nos manda el nombre de la fruta detectada.

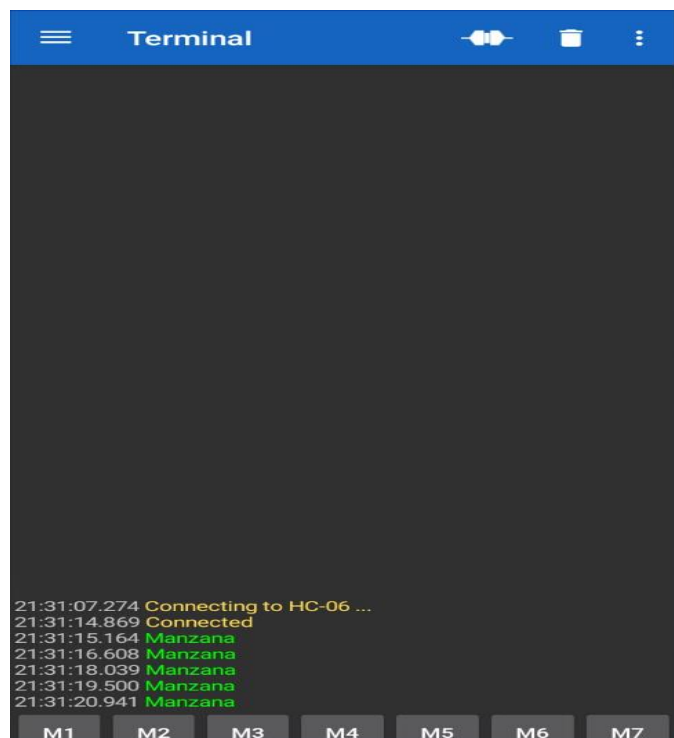


Imagen 9. Datos enviados al celular.

CONCLUSIÓN.

En este proyecto aplicamos todo lo aprendido en la materia “Microprocesadores y Microcontroladores”, utilizamos el microcontrolador PIC16F887, el sensor de color TCS230 para leer las frecuencias de las frutas, una LCD para mostrar los datos, un módulo HC-06 Bluetooth para enviar los datos al celular y demás componentes para poder desarrollar este sensor de frutas. Este proyecto fue programado en lenguaje C, con ayuda del IDE PIC C compiler, este sensor fue logrado gracias al “Curso de programación de microcontroladores Pic en lenguaje C”, del canal de YouTube llamado: “Electrónica y circuitos”.

BIBLIOGRAFÍA.

YouTube (2020.). *Curso de programación de microcontroladores Pic en lenguaje C*. https://youtube.com/playlist?list=PLONPO-iVba9nbY_KTCHt9GGj9dGSj1qYo&si=ZTz-crPqbxdk7wh7