

# Proyecto Final. Implementación de sensor de distancia ultrasónico, sensor de gas y sensor de temperatura

Jesus Salazar Vidal

Departamento de Estudios Multidisciplinarios

Universidad de Guanajuato

Yuriria Guanajuato, 38940, MX.

j.salazarvidal@ugto.mx

**Resumen:** En la siguiente practica se realizará un circuito en el cual se incluyan diferentes sensores como el sensor de distancia ultrasónico, el sensor de gas y el sensor de temperatura. Esto servirá para crear un prototipo el cual nos ayude a identificar un objeto que este generando calor, el prototipo nos mostrara la calidad del aire, la temperatura y la distancia a la que se encuentre este objeto, todos los datos se mostraran en una pantalla LCD.

## I. INTRODUCCIÓN

Durante el transcurso de la materia de Informática industrial se estuvieron utilizando varias herramientas como lo fue TinkerCard para la simulación del circuito con Arduino, vimos el software de Eagle para crear esquemas de los circuitos y la creación de repositorios en GitHub los cuales nos permiten llevar un control de las versiones de nuestro trabajo. Por lo que este proyecto final tiene la finalidad de utilizar todo lo visto en la clase por lo que se hará una simulación del circuito que en este caso es el prototipo con los diferentes sensores, se creara el esquema en Eagle y se documentarán los resultados los cuales se estarán subiendo al repositorio para su revisión por parte del profesor.

Arduino es una plataforma de hardware libre la cual nos permite desarrollar un gran número de proyectos ya que es bastante sencilla la programación del microcontrolador, yo elegí utilizar sensores ya que bastantes de los proyectos que he visto en la universidad por parte de mis compañeros los utilizan yo al ser de sistemas no he realizado muchos circuitos en Arduino, esta es la razón por la que elegí este sencillo prototipo.

## II. MATERIALES

A continuación, se mostrará la información de los materiales necesarios para realizar el prototipo.

**Sensor de movimiento ultrasónico HC-SR04:**

El sensor HC-SR04 está compuesto por un emisor y un receptor de ultrasonidos. Estos nos ayudan a medir la distancia a la que se encuentra un objeto justo frente a él, enviando un pulso de ultrasonidos y midiendo el tiempo que transcurre hasta que vuelve dicho pulso.

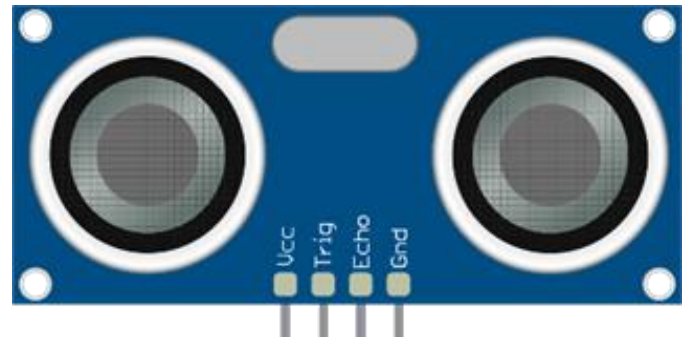


Fig. 1 Sensor ultrasónico

-Vcc: Pin de alimentación. (5V)

-Trigger: Pin de disparo. Este pin es una entrada, por lo que, en el sistema de control, por ejemplo, Arduino, se tiene que conectar a una salida.

-Echo: Este pin es una salida del sensor, por lo que ha de ser conectado a una entrada del sistema de control.

-Gnd: Pin negativo de alimentación.

**Sensor de calidad del aire MQ-135:**

Este sensor de control de calidad de aire es usado para la detección de contaminación en el medio ambiente, por lo general es implementado en circuitos de control como alarmas en las casas, sitios donde se desea prevenir altos niveles de contaminación a nivel aeróbico como industrias que manejan compuestos químicos que pueden ser nocivos también para la salud, especialmente en equipos controladores de calidad de aire en edificios/oficinas.



Fig. 2 Sensor de calidad del aire

### Sensor de temperatura TEMP-36

El TMP36 es un sensor de temperatura que en su salida los proporciona una lectura en grados centígrados de precisión y de bajo voltaje. Cuenta con una precisión de  $\pm 1^\circ\text{C}$  a  $+25^\circ\text{C}$  y  $\pm 2^\circ\text{C}$  por encima del rango de temperatura de  $-40^\circ\text{C}$  a  $+125^\circ\text{C}$ . Trabaja con un voltaje de funcionamiento de 2.7V a 5.5V.

Se utilizan el sensor TMP36 para medir la temperatura de los sistemas de control ambiental, protección térmica, control de procesos industriales, alarmas contra incendios, monitores de sistemas de potencia y gestión térmica de la CPU.



Fig. 3 Sensor de temperatura

- Pantalla LCD 16x2:

Pantalla de cristal líquido nombrada por sus siglas en inglés Liquid Crystal Display, que se utiliza para ver imágenes fijas y en movimiento. En nuestro prototipo nos mostrara la información de los sensores.

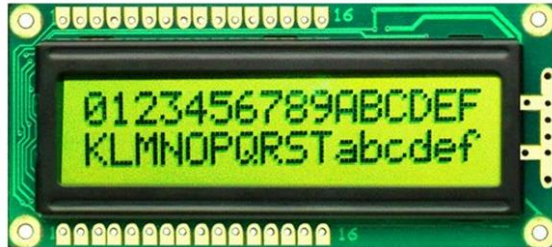


Fig. 4 Pantalla LCD 16x2

### III. METODOLOGÍA

Para elaborar el Proyecto se utilizó la placa de Arduino Uno en la cual se montó el siguiente circuito que se muestra en la Fig. 5 la cual corresponde al diagrama realizado en TinkerCad y en la Fig. 6 el esquemático realizado en Eagle.

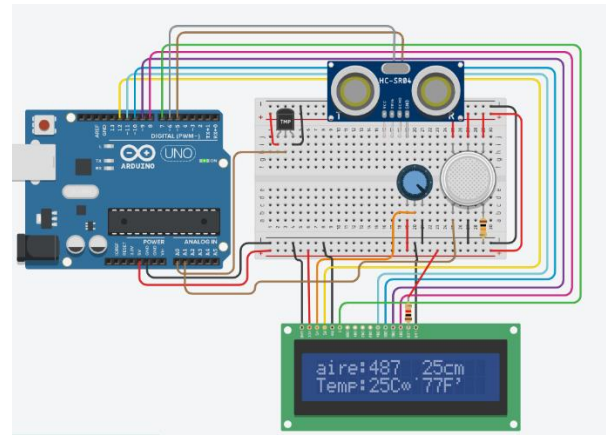


Fig. 5 Diagrama del circuito en TinkerCad.

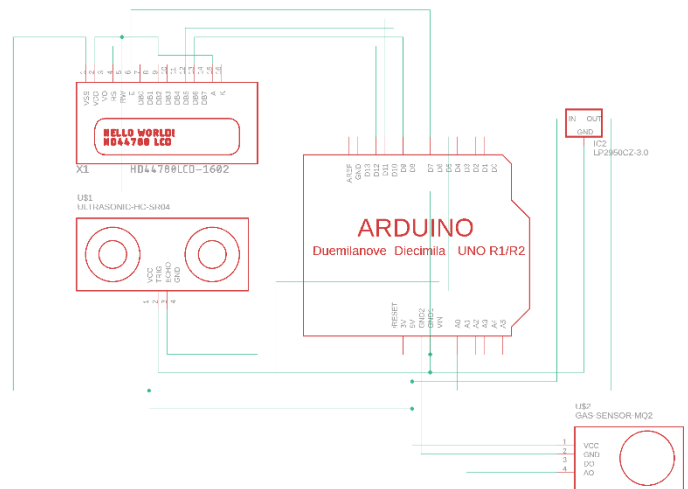


Fig. 6 Esquemático en Eagle.

Para programar el Arduino se utilizó la librería de LiquidCrystal la cual es una librería que permite que la placa de Arduino controle pantallas LiquidCrystal ( LCD ) basadas en el chipset Hitachi HD44780 ( o un compatible), que se encuentra en la mayoría de las pantallas LCD basadas en texto . La biblioteca funciona en modo de 4 u 8 bits (es decir, utilizando 4 u 8 líneas de datos además de rs, enable y, opcionalmente, las líneas de control rw).[1]

Para usar esta biblioteca  
`#include <LiquidCrystal.h>`

Para la lectura de la temperatura se utilizó la función map la cual es una función que nos permite mapear un numero de un rango a otro.

Sintaxis

`map(value, fromLow, fromHigh, toLow, toHigh)`

Parámetros

value: el número a mapear.

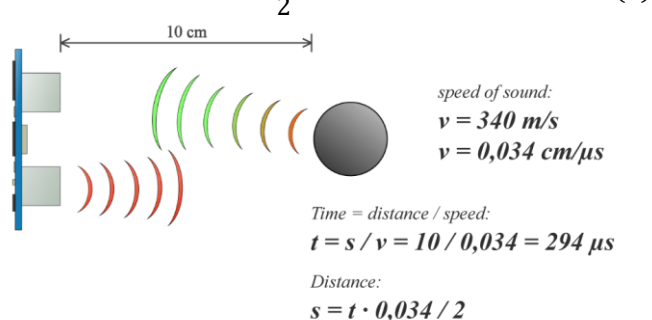
fromLow: el límite inferior del rango actual del valor.

fromHigh: el límite superior del rango actual del valor.  
 toLow: el límite inferior del rango objetivo del valor.  
 toHigh: el límite superior del rango objetivo del valor.

```
celsius = map (((analogRead (A0) - 20) * 3.04), 0, 1023, -40, 125);
```

Mapa para obtener la temperatura matemáticamente. Significa 0 = -40 grados y 1023 = 125 grados.

Para la lectura del sensor ultrasónico y poder mostrar el resultado en cm se utilizó la siguiente formula.

$$d = \frac{t \cdot 0.034}{2} \quad (1)$$


speed of sound:  
 $v = 340 \text{ m/s}$   
 $v = 0,034 \text{ cm/us}$

Time = distance / speed:  
 $t = s / v = 10 / 0,034 = 294 \mu\text{s}$

Distance:  
 $s = t \cdot 0,034 / 2$

Fig. 7 imagen tomada de howtomechatronics.com donde explican cómo se obtiene la fórmula 1.

La formula se divide entre dos ya que el tiempo total es el doble ya que la onda de sonido del sensor debe hacia adelante y después rebotar hacia atrás. Entonces, para obtener la distancia en cm, debemos multiplicar el valor del tiempo de viaje recibido desde el pin de eco por 0.034 y dividirlo por 2.[2]

Para la lectura del sensor de calidad del aire MQ-132 se utiliza nuevamente la función map ya que queremos mapear la lectura de la entrada analógica del sensor la cual va de 0 a 1023 y el sensor mide de 10ppm a 1000 ppm por lo que la función map queda de la siguiente manera.

```
Map(lectura, 0, 1023, 10, 1000)
```

A continuación, se muestra el código que se cargó en el Arduino.

```
//incluimos librería para el LCD:  
#include <LiquidCrystal.h>
```

```
// declarar pantalla LCD y asignar pines RS=12 E=11 DB4=5  
DB5=4 DB6=3 DB7=2  
LiquidCrystal lcd(12, 7, 11, 10, 9, 8);
```

```
unsigned long tiempoAhora = 0;  
//variable para la lectura analógica de gas  
int valor_Gas;
```

```
//variable para lectura de temperatura  
int baselineTemp = 0;
```

```
int celsius = 0;  
int fahrenheit = 0;
```

```
//sensor ultrasonico  
const int trigPin = 6;  
const int echoPin = 5;  
long tiempo;  
int distanciaCm;
```

```
void setup() {  
  // iniciar pantalla indicando número de columnas(columns) y  
  filas(rows)  
  lcd.begin(16, 2);  
  //escribimos título en la posición 0,0  
  lcd.setCursor(0,0);  
  lcd.print("aire:");// calidad del aire  
  
  pinMode(A0, INPUT);// pinbaseTMP  
  lcd.setCursor(0,1);  
  lcd.print("Temp:");//nivel de temperatura
```

```
  pinMode(trigPin, OUTPUT);  
  pinMode(echoPin, INPUT);
```

```
}
```

```
void loop() {  
  tiempo = millis();  
  valor_Gas = map(analogRead(A1),0,1023,10,1000);// valor  
  entre 0 y 1023  
  lcd.setCursor(5, 0);  
  lcd.print(valor_Gas);  
  //delay(500);  
  while (millis() < tiempoAhora + 500){  
    /* ESPERA */  
  }
```

```
  //temperatura  
  baselineTemp = 40;
```

```
  celsius = map(((analogRead(A0) - 20) * 3.04), 0, 1023, -40,  
  125);//mapa para obtener la temperatura matemáticamente.  
  Significa 0 = -40 grados y 1023 = 125 grados
```

```
  fahrenheit = ((celsius * 9) / 5 + 32);  
  lcd.setCursor(5, 1);  
  lcd.print(celsius);  
  lcd.print("C");
```

```

lcd.setCursor(10, 1);
lcd.print(fahrenheit);
lcd.print("F");

```

```

digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

```

```

tiempo = pulseIn(echoPin, HIGH);
distanciaCm= tiempo*0.034/2;

```

```

lcd.setCursor(10,0); // Definimos en que parte de la pantalla
mostrara la distancia

```

```

lcd.print(distanciaCm); // Muestra la distancia en la pantalla
LCD
lcd.print("cm ");
//delay(10);
while (millis() < tiempoAhora + 10){
  /* ESPERA */
}
}

```

#### IV. RESULTADOS

A continuación, se muestran algunas imágenes que se tomaron del circuito en distintas condiciones, primeramente, se puede dar a conocer que los sensores funcionan correctamente.

La Fig. 8 muestra la lectura de la temperatura y la calidad del aire en condiciones normales sin ningún agente que las esté alterando.



Fig. 8 medidas de los sensores en condiciones normales.

La Fig.9 muestra las lecturas de los sensores al ser alterados por un pedazo de cartón el cual se encuentra encendido por lo tanto emite humo, calor y se encuentra cerca del sensor de distancia.

La calidad del aire es de 745ppm por lo que es peor, la temperatura subió a 35° C y el objeto se encuentra a 6cm del sensor.

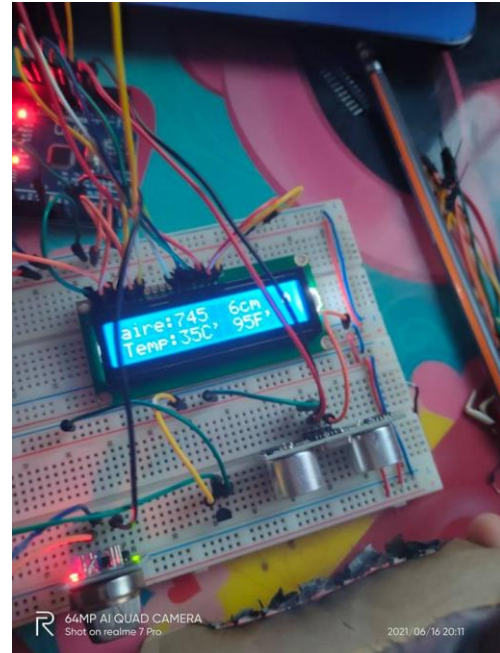


Fig. 9 medidas de los sensores afectadas por un objeto que está emitiendo humo, calor y se encuentra cerca del sensor de distancia

Por ultimo la Fig. 10 muestra las lecturas de los sensores una vez que se retiro el objeto que emitía humo y calor y se abrió la ventana para que el aire fuera mas puro y se espaciara el humo.

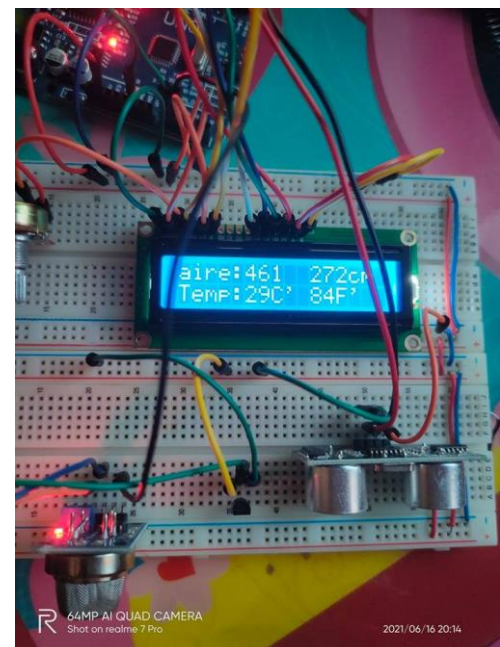




Fig. 10 medidas de los sensores una vez retirado el objeto que emitía humo y calor.

Como se puede observar una vez que se retiró el objeto las medias de los sensores comienzan a bajar o volver a la normalidad y en este caso el sensor de distancia no detecta ningún objeto cerca.

### CONCLUSIÓN

El Desarrollo del proyecto fue entretenido ya que es la primera vez que utilizo los sensores, y creo que con estos se pueden crear un gran numero de proyectos como un detector de humo el cual tenga una alarma, el sensor de calidad del aire se puede usar para hacer un prototipo que nos ayude a identificar si hay una fuga de gas o detectar si hay presencia de alcohol en un lugar determinado o en nuestras propias casas. La practica se concluyó con éxito y no se presentaron problemas durante el desarrollo del circuito.

### REFERENCIAS

- [1] LiquidCrystal                      Library                      URL.  
<https://www.arduino.cc/en/Reference/LiquidCrystal>
- [2] Dejan, "Ultrasonic Sensor HC-SR04 and Arduino Tutorial" URL:  
<https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>