

X A.3.1 Actividad de aprendizaje

Circuito sensor de detección de objetos, utilizando un Arduino, un sensor ultrasonico y un Display I2C LCD 16x2.

Instrucciones

- Realizar un sistema simulado, capaz de detectar la presencia y/o ausencia de un objeto, a través de un circuito electrónico, utilizando un Arduino, y un Sensor Ultrasonico HC-SR04.
- Toda actividad o reto se deberá realizar utilizando el estilo MarkDown con extension .md y el entorno de desarrollo VSCode, debiendo ser elaborado como un documento single page, es decir si el documento cuanta con imágenes, enlaces o cualquier documento externo debe ser accedido desde etiquetas y enlaces, y debe ser nombrado con la nomenclatura A3.1_NombreApellido_Equipo.pdf.
- Es requisito que el .md contenga una etiqueta del enlace al repositorio de su documento en GITHUB, por ejemplo Enlace a mi GitHub y al concluir el reto se deberá subir a github.
- Desde el archivo .md exporte un archivo .pdf que deberá subirse a classroom dentro de su apartado correspondiente, sirviendo como evidencia de su entrega, ya que siendo la plataforma oficial aquí se recibirá la calificación de su actividad.
- Considerando que el archivo .PDF, el cual fue obtenido desde archivo .MD, ambos deben ser idénticos.
- Su repositorio ademas de que debe contar con un archivo **readme**.md dentro de su directorio raíz, con la información como datos del estudiante, equipo de trabajo, materia, carrera, datos del asesor, e incluso logotipo o imágenes, debe tener un apartado de contenidos o indice, los cuales realmente son ligas o **enlaces a sus documentos .md**, evite utilizar texto para indicar enlaces internos o externo.
- Se propone una estructura tal como esta indicada abajo, sin embargo puede utilizarse cualquier otra que le apoye para organizar su repositorio.
- * readme.md
 - blog
 - C3.1 TituloActividad.md
 - C3.2 TituloActividad.md
 - C3.3 TituloActividad.md
 - C3.4_TituloActividad.md
 - C3.5_TituloActividad.md
 - C3.6_TituloActividad.md
 - C3.7_TituloActividad.md
 - C3.8_TituloActividad.md
 - img
 - docs
 - A3.1_TituloActividad.md
 - A3.2 TituloActividad.md
 - A3.3 TituloActividad.md

Desarrollo

1. Utilizar el siguiente listado de materiales para la elaboración de la actividad

Cantidad	Descripción
1	Sensor Ultrasonico HC-SR04
1	Display LCD de 16x2
1	Fuente de voltaje de 5V
1	1 Potenciómetro 10k
1	Arduino UNO

2. Basado en la imágen que se muestra, ensamble dentro del simulador a utilizar, el circuito electrónico indicado en la **Figura 1**.

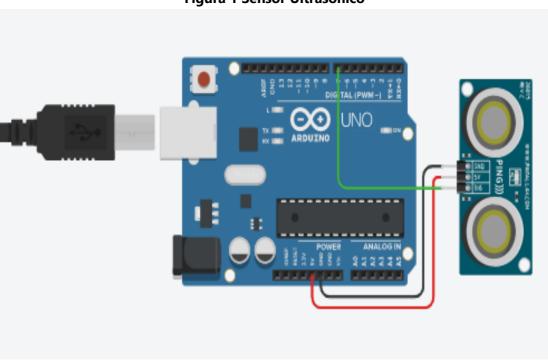
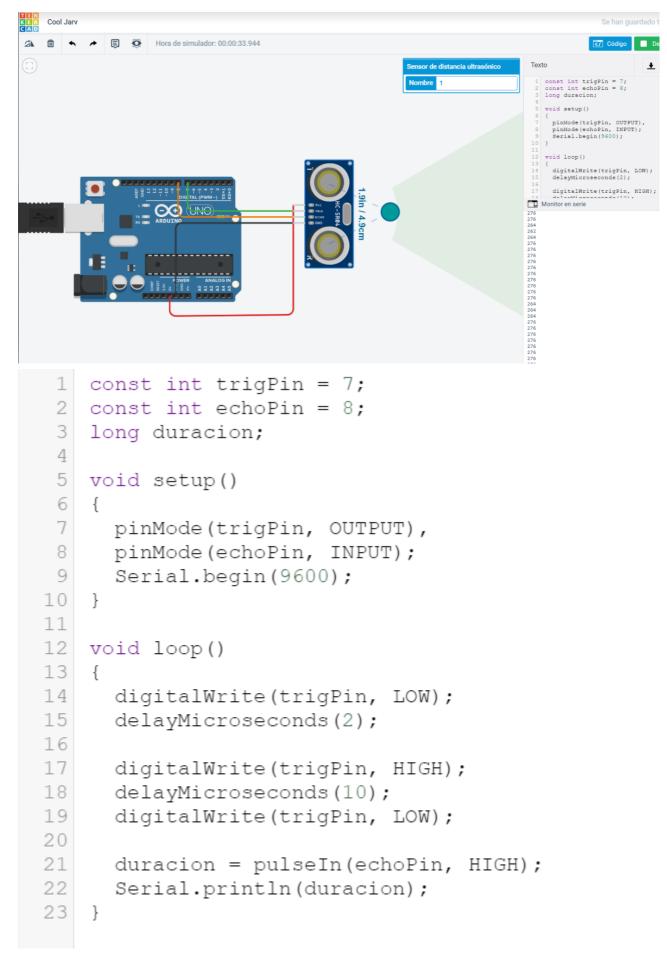


Figura 1 Sensor Ultrasonico

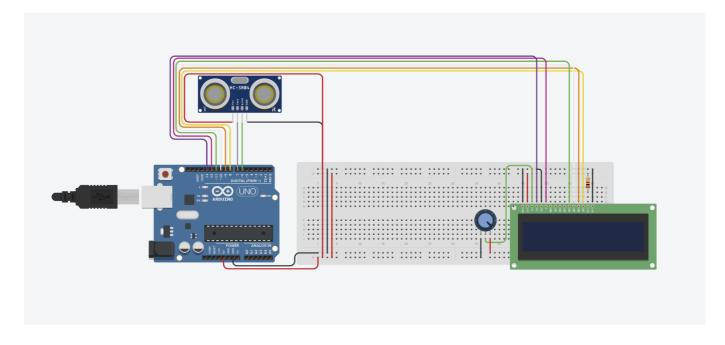
3. Realice el programa que permita a través de una de las entradas del Arduino, recibir el valor que registra el **Sensor Ultrasonico** al acercarse un objeto a distintas distancias.



4. Considerando que el sensor ultrasonico tiene un rango mínimo y máximo de detección basado en el tiempo de retorno de la señal sonica, que valores se obtienen en la simulación bajo las **siguientes** condiciones:

Numero	Condición 1	Condición2	El objeto es detectado?
1	5 cm de distancia al sensor	0 grados al eje perpendicular del sensor	Si se detecta
2	50 cm de distancia al sensor	35 grados al eje perpendicular del sensor	No se detecta
3	100 cm de distancia al sensor	-35 grados al eje perpendicular del sensor	No se detecta
4	5 cm de distancia al sensor	90 grados al eje perpendicular del sensor	No se detecta
5	50 cm de distancia al sensor	-60 grados al eje perpendicular del sensor	No se detecta
6	350 cm de distancia al sensor	0 grados al eje perpendicular del sensor	No se detecta

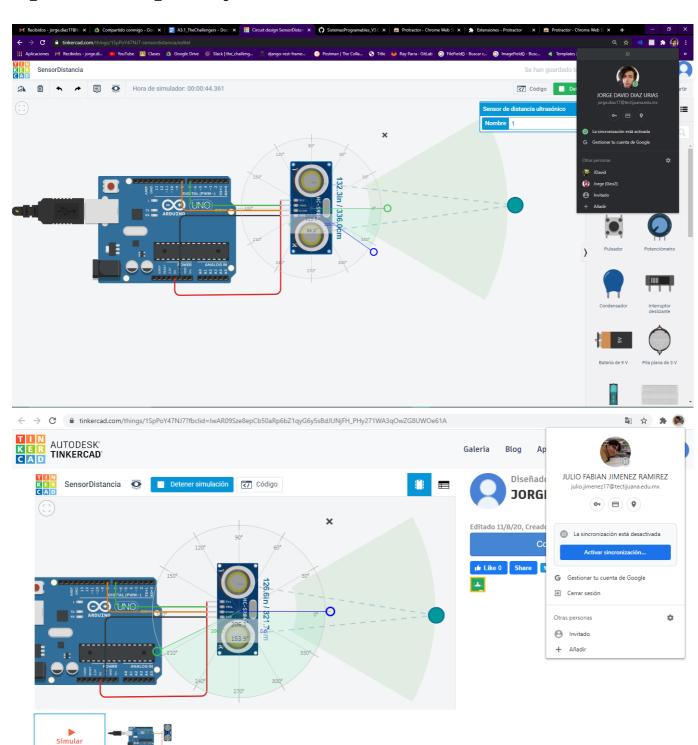
5. Una vez completados los puntos anteriores, agregue a la Figura 1, **un Display I2C 16x2 LCD**, y coloque la imagen del circuito completado hasta este apartado.

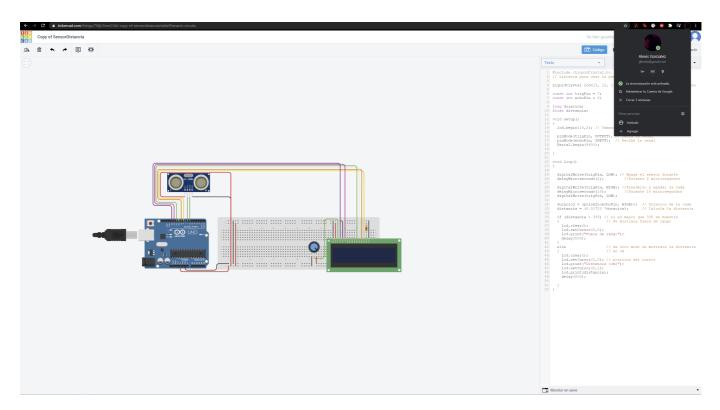


6. Al haber completado la integración del Display I2C, ajuste el programa que le permita a través del display mostrar el siguiente mensaje, "**Objetivo detectado a ? cm**, y en caso de no lograr la detección indicar el mensaje "**Objetivo fuera de rango**"

```
1 #include <LiquidCrystal.h>
   // Libreria para usar la pantalla
 3
 4 LiquidCrystal lcd(13, 12, 11, 10, 9, 8); // Declaracion de pines
 6 const int trigPin = 7;
 7 const int echoPin = 6;
 8
 9
   long duracion;
10 float distancia;
11
12 void setup()
13 {
14
     lcd.begin(16,2); // Tamano de la pantalla
15
16
    pinMode(trigPin, OUTPUT), // Lanza la senal
17
    pinMode(echoPin, INPUT); // Recibe la senal
18
     Serial.begin(9600);
19
20 }
21
22 void loop()
23 {
24
25
     digitalWrite(trigPin, LOW); // Apaga el sensor durante
26
     delayMicroseconds(2);
                                    //Durante 2 microsegundos
27
     digitalWrite(trigPin, HIGH); //Prenderlo y mandar la onda
29
     delayMicroseconds(10);
                                   //Durante 10 microsegundos
     digitalWrite(trigPin, LOW);
30
31
32
     duracion = (pulseIn(echoPin, HIGH)); // Duracion de la onda
33
     distancia = (0.01723 *duracion);
                                           // Calcula la distancia
34
35
     if (distancia > 335) // si es mayor que 335 se muestra
36
                            // Se mostrara fuera de rango
     {
37
       lcd.clear();
       lcd.setCursor(0,0);
38
39
       lcd.print("Fuera de rango");
40
       delay(500);
41
     }
42
     else
                            // de otro modo se mostrara la distancia
43
     {
                            // en cm
44
       lcd.clear();
45
       lcd.setCursor(1,0); // posicion del cursor
46
       lcd.print("Distancia (cm)");
47
       lcd.setCursor(0,1);
48
       lcd.print(distancia);
49
       delay(500);
50
51
     }
52
   }
```

7. Coloque aqui evidencias que considere importantes durante el desarrollo de la actividad.





8. Conclusiones

Alexis Gonzalez

Durante la elaboración de esta práctica se pudo observar el funcionamiento del sensor ultrasónico en conjunto con una pequeña pantalla lcd para mostrar los datos que deseamos. En este caso se utilizó para mostrar si el objetivo detectado por el sensor ultrasónico estaba fuera o dentro del rango, y si estaba dentro a cuantos cm se encontraba dicho objeto. También el funcionamiento del código para poder transformar los datos obtenidos por el sensor para después mostrarlos en la pantalla.

Jorge Diaz

Con el desarrollo de esta práctica nos podemos dar cuenta del funcionamiento del sensor ultrasónico, este sensor está compuesto con un emisor y un receptor ayudando a medir la distancia de un objeto mediante el envío de pulsos. Estos sensores manejan un alcance máximo y uno mínimo así como con un ángulo de detección de 15 a 20 grados de acuerdo al datasheet del sensor y al desarrollar la práctica pudimos observar que solamente una condición de las presentadas en la práctica pudo detectar el objeto.

Julio Jimenez

En conclusión se puede observar cómo funciona el sensor ultrasónico, y como se pueden manipular los datos obtenidos del sensor para nuestra conveniencia, por ejemplo, si queremos distancia o tiempo, etc. El sensor HC-SR04 tiene un límite de distancia aproximada de 336, y un punto ciego, entonces empieza a detectar desde los 2.5 cm aproximadamente.



Criterios Descripción Puntaje

Criterios	Descripción	Puntaje
Instrucciones	Se cumple con cada uno de los puntos indicados dentro del apartado Instrucciones?	10
Desarrollo	Se respondió a cada uno de los puntos solicitados dentro del desarrollo de la actividad?	60
Demostración	El alumno se presenta durante la explicación de la funcionalidad de la actividad?	20
Conclusiones	Se incluye una opinión personal de la actividad por cada uno de los integrantes del equipo?	10

🔝 Ir a GitHub de Julio Jimenez

🙉 Ir a GitHub de Jorge Diaz

🕮 Ir a GitHub de Alexis Gonzalez