|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | Proyecto Final con Arduino y su aplicación en la vida real | | | **No.** | **3** |
| **Asignatura:** | **Lenguaje Interfaz** | **Carrera:** | **ISIC** | **Duración de la práctica (Hrs)** | **2** |

**Nombres:**

- Jehiel Andru Medina Álvarez

- Sonia Cuevas García

- Raúl Ciriaco Castillo

- Jonathan Ordoñez Cruz

**Grupo:** 3501

**I. Competencia(s) específica(s):** El objetivo de este proyecto es medir la distancia utilizando un sensor ultrasónico y mostrar el resultado en una pantalla LCD. Además, el sistema controlará dos LEDs (verde y rojo) y un buzzer, dependiendo de la distancia medida.

**II. Lugar de realización de la práctica:** Aula

**III. Material empleado:** **Componentes Electrónicos:**

* 1 Arduino UNO (u otro compatible).
* 1 Pantalla LCD 16x2 con Interfaz I2C - Fondo Azul
* 1 Sensor ultrasónico HC-SR04.
* 2 LEDs (1 rojo, 1 verde).
* 2 resistencias de 220 ohmios (para proteger los LEDs).
* 1 Protoboard.
* Cables de conexión (jumper wires).

**Herramientas:**

* Cable USB para programar el Arduino.
* Computadora con Arduino IDE instalado.

**IV. Desarrollo de la práctica:**

**Medición de Distancia**: El sensor ultrasónico emite un pulso ultrasónico cuando se activa el pin TRIG. Luego, mide el tiempo que tarda el pulso en regresar al pin ECHO. Este tiempo se utiliza para calcular la distancia entre el sensor y el objeto.

**Visualización en la Pantalla LCD**: La distancia medida se muestra en tiempo real en una pantalla LCD de 16x2, actualizándose cada vez que se toma una nueva medición.

**Control de LEDs y Buzzer**:

**LED verde**: Se enciende cuando la distancia medida es mayor o igual a 10 cm.

**LED rojo**: Se enciende cuando la distancia es menor a 10 cm.

**Buzzer**: Suena cuando la distancia es menor a 10 cm, emitiendo un tono de 1 kHz.

**Esquema de Conexiones**

1. **Sensor Ultrasónico (HC-SR04)**:
   * **VCC** → **5V** en Arduino.
   * **GND** → **GND** en Arduino.
   * **TRIG** → Pin **12** en Arduino.
   * **ECHO** → Pin **13** en Arduino.
2. **Pantalla LCD (con módulo I2C)**:
   * **VCC** → **5V** en Arduino.
   * **GND** → **GND** en Arduino.
   * **SDA** → Pin **A4** en Arduino (SDA).
   * **SCL** → Pin **A5** en Arduino (SCL).
3. **LED Verde y LED Rojo**:
   * **Anodo (+)** de cada LED a los pines **2** (verde) y **4** (rojo) en Arduino.
   * **Cátodo (-)** de cada LED a **GND** a través de una resistencia de 220 ohm.
4. **Buzzer**:
   * **Pin positivo (+)** del buzzer a **pin 8** en Arduino.
   * **Pin negativo (-)** del buzzer a **GND** en Arduino.

**Montaje del Circuito:**

1. **Sensor Ultrasónico:**
   * Conecta VCC a 5V del Arduino.
   * Conecta GND a GND del Arduino.
   * Conecta Trig al pin 9 del Arduino.
   * Conecta Echo al pin 10 del Arduino.
2. **LEDs:**
   * Conecta el ánodo (patilla larga) de cada LED al pin digital correspondiente del Arduino.
   * Coloca una resistencia de 220 ohmios en serie con el ánodo para limitar la corriente.
   * Conecta el cátodo (patilla corta) de cada LED al GND de la protoboard.
3. **GND:**
   * Conecta el GND del Arduino a la línea de tierra de la protoboard.

**Código Arduino:**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

// Inicializar la pantalla LCD con dirección I2C 0x27 y tamaño 16x2

LiquidCrystal\_I2C lcd(0x27, 16, 2);

// Definición de pines

const int TRIG = 12;

const int ECHO = 13;

const int LEDV = 2;

const int LEDR = 4;

const int BUZZER = 8;

// Variables para medir la distancia

long duracion;

int distancia;

void setup() {

// Configuración de pines

pinMode(TRIG, OUTPUT);

pinMode(ECHO, INPUT);

pinMode(LEDR, OUTPUT);

pinMode(LEDV, OUTPUT);

pinMode(BUZZER, OUTPUT);

// Inicializar comunicación serie para depuración

Serial.begin(9600);

// Inicializar la pantalla LCD

lcd.init();

lcd.backlight(); // Encender la luz de fondo

}

void loop() {

// Generar el pulso de activación del sensor ultrasónico

digitalWrite(TRIG, LOW);

delayMicroseconds(2);

digitalWrite(TRIG, HIGH);

delayMicroseconds(10);

digitalWrite(TRIG, LOW);

// Leer la duración del pulso de retorno

duracion = pulseIn(ECHO, HIGH);

// Calcular la distancia en centímetros

distancia = duracion / 58.2;

// Mostrar la distancia en el monitor serie

Serial.print("Distancia: ");

Serial.print(distancia);

Serial.println(" cm");

// Mostrar la distancia en la pantalla LCD en las dos filas

lcd.clear();

lcd.setCursor(0, 0); // Posicionar el cursor en la primera fila

lcd.print("Distancia:");

lcd.setCursor(0, 1); // Posicionar el cursor en la segunda fila

lcd.print(distancia);

lcd.print(" cm");

// Control de LEDs y buzzer según la distancia

if (distancia < 10) {

digitalWrite(LEDR, HIGH);

digitalWrite(LEDV, LOW);

tone(BUZZER, 1000); // Activar buzzer

} else {

digitalWrite(LEDR, LOW);

digitalWrite(LEDV, HIGH);

noTone(BUZZER); // Desactivar buzzer

}

delay(100); // Pequeño retraso para evitar lecturas muy frecuentes

}

**Descripción del Código**

**Inicialización de Pines**:

Configura los pines de los LEDs, el buzzer, y el sensor ultrasónico en el setup().

**Medición de Distancia**:

Se genera un pulso en el pin TRIG del sensor ultrasónico, y se mide el tiempo de retorno del pulso con pulseIn() en el pin ECHO.

La distancia se calcula usando la fórmula: distancia = duracion / 58.2; (donde 58.2 es un factor de conversión de microsegundos a centímetros).

**Control de LEDs y Buzzer**:

Si la distancia es menor a 10 cm, se enciende el LED rojo y se activa el buzzer.

Si la distancia es mayor o igual a 10 cm, se enciende el LED verde y se apaga el buzzer.

**Pantalla LCD**:

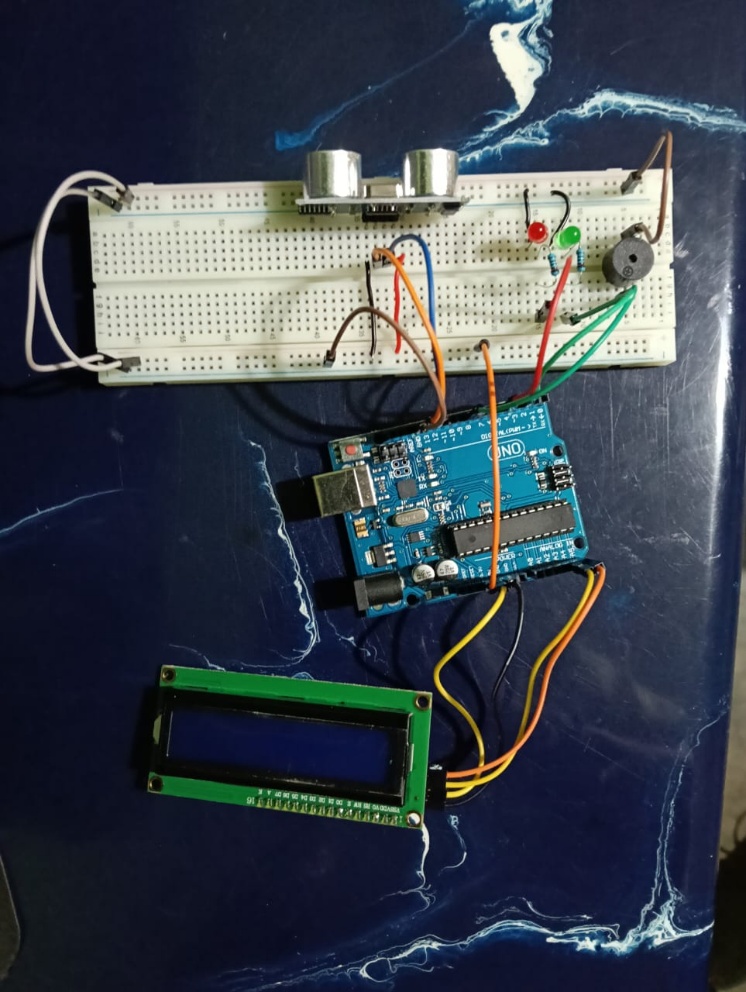
Se limpia la pantalla LCD con lcd.clear() y se actualiza la distancia en tiempo real en ambas filas de la pantalla.

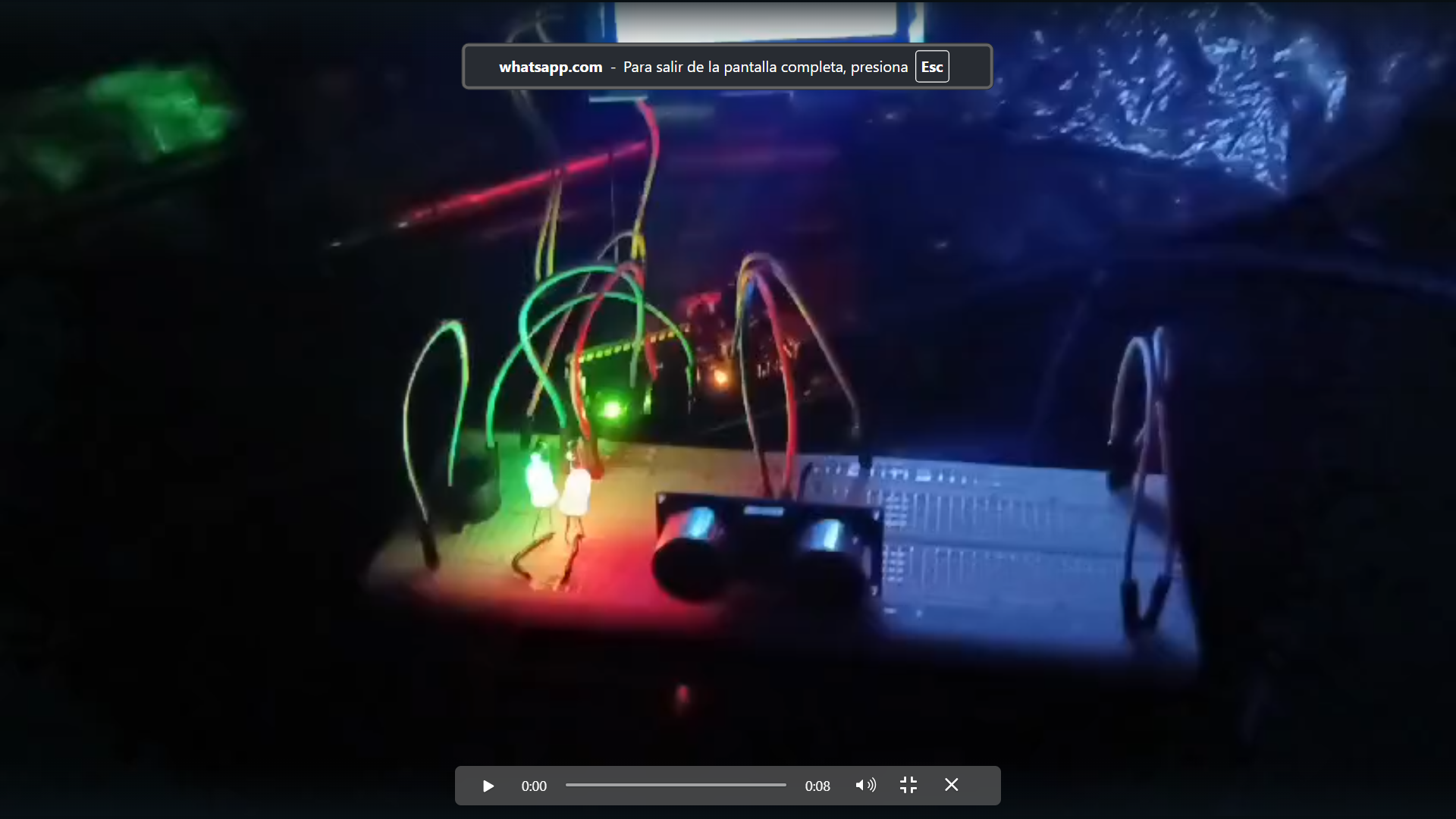
**Posibles Mejoras**

**Alarma sonora**: Si deseas una alarma más fuerte, puedes ajustar la frecuencia del buzzer o usar diferentes tonos para diferentes rangos de distancia.

**Medición continua**: Puedes agregar un botón para activar o desactivar el sistema de medición, o agregar más sensores ultrasónicos para medir en diferentes direcciones.

**Realización de Practica:**





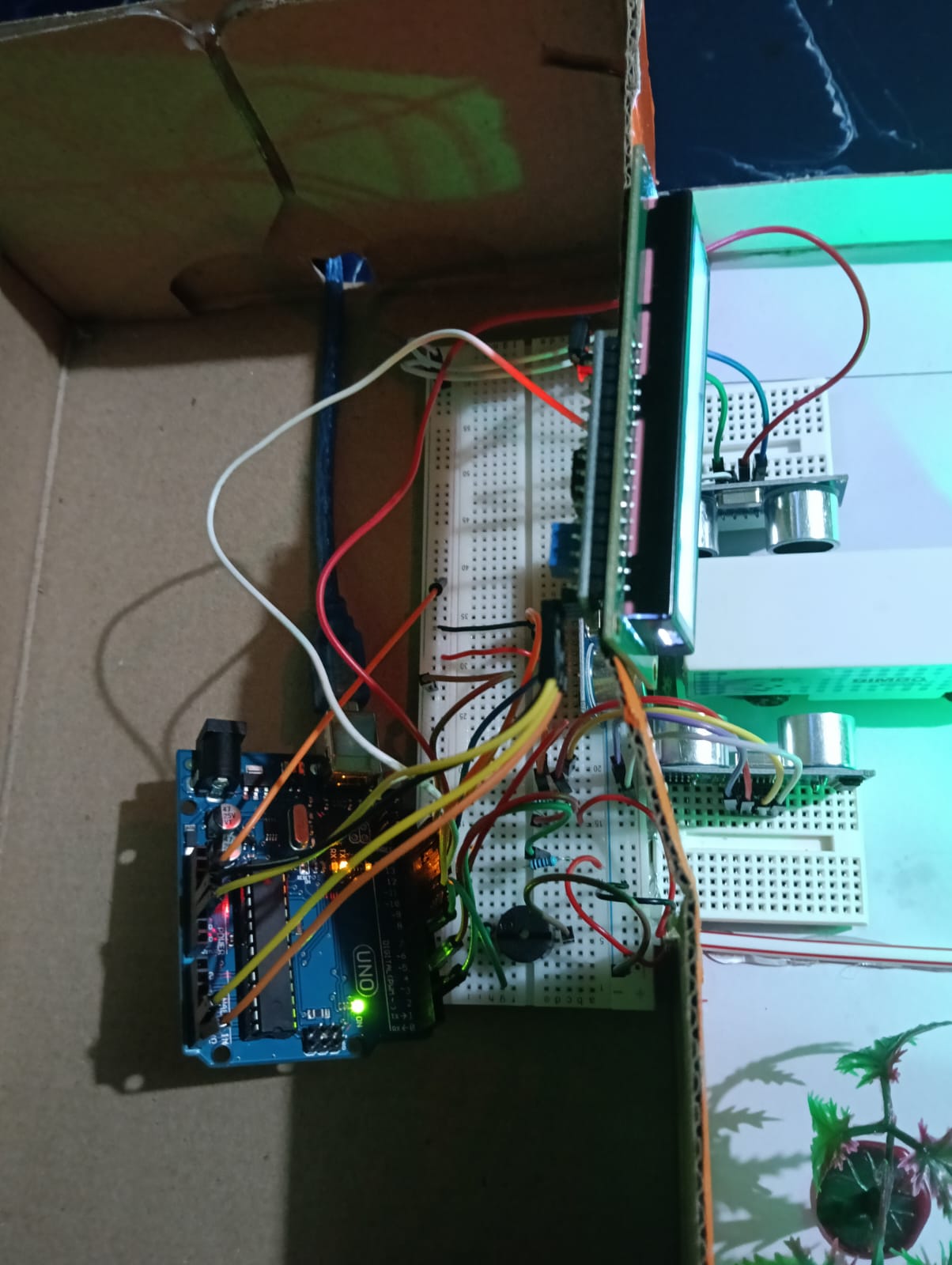


Una vez que tenemos la base de nuestro proyecto implementamos la maqueta o diseño para su presentación:

Como lo habíamos mencionado nuestro proyecto principalmente está enfocado para la para ayudar a las empresas a lo que viene siendo los vehículos de carga estos pues principalmente la mayoría de las veces tienen que ocupar de chalanes o personas externas que indiquen cuando un carro esté estacionando en un lugar lo que nuestro proyecto hace sirve para facilitar el estacionamiento de un vehículo dentro de un carril donde nuestros sensores indican pues una cierta distancia en la cual se debe de acomodar y de igual manera la posición correcta en la que debe de permanecer el vehículo dentro de lo que viene siendo un carril eso indicándonos con focos leds rojo y verde si el carro se está estacionando correctamente o no de igual manera con una alarma que viene siendo el buzzer que usamos el cual nos indica un sonido cuando realmente se está acomodando correctamente y si no para que se estacione de la manera que es posteriormente también tenemos la pantalla LCD la cual pues nos muestra que el carro al final está estacionado correctamente al igual que la distancia a la que se encuentra de la pared.



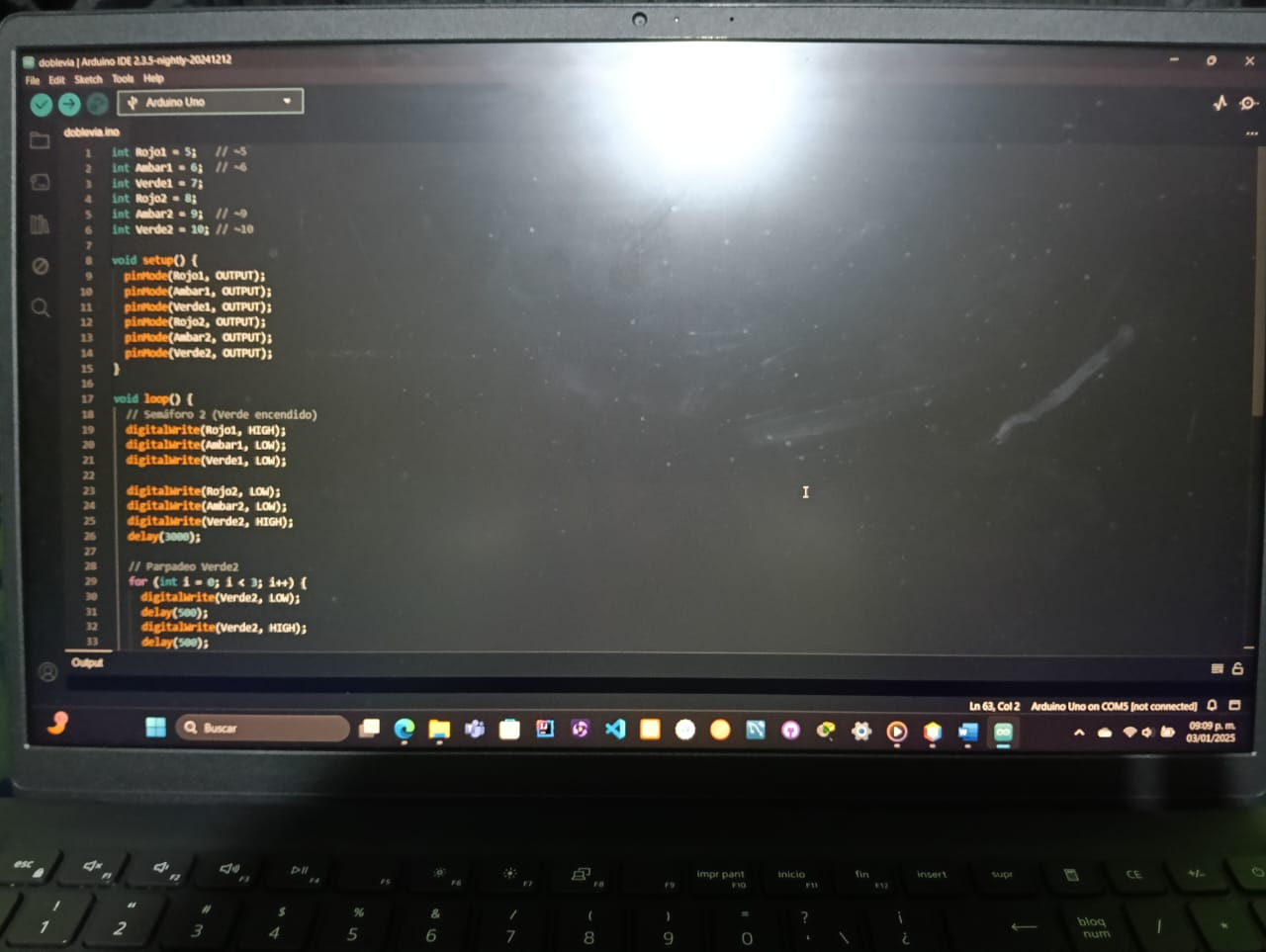


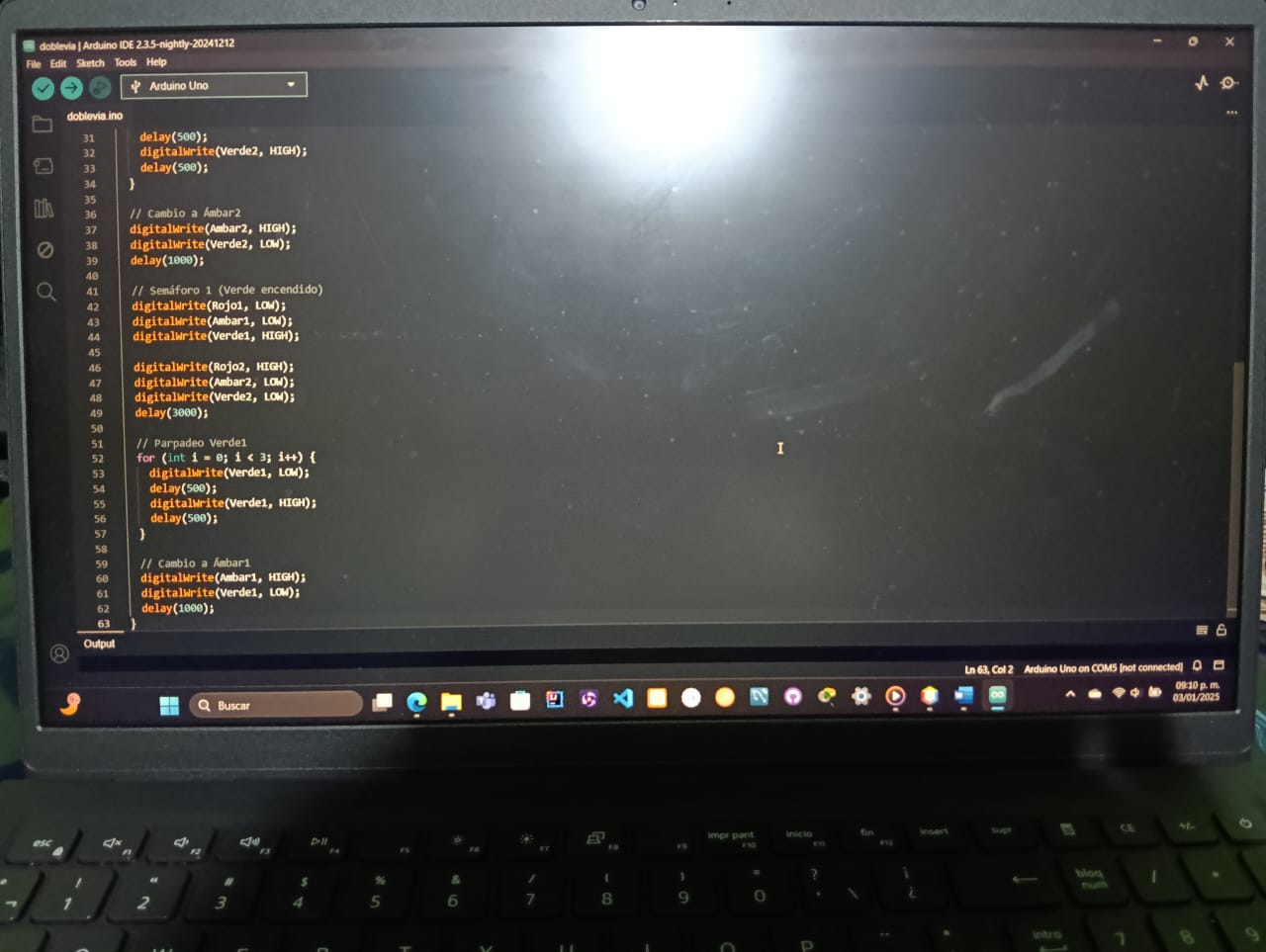


**Link del video del circuito funcionando:**

<https://youtu.be/A6Xz05LrTNQ?si=KcHwBExfPqpQZfbb>

Código ejecutado en computadora:





**V. Conclusiones**

Este proyecto representa una solución simple pero poderosa para medir distancias utilizando un sensor ultrasónico con Arduino, permitiendo realizar un seguimiento preciso de la distancia entre un objeto y el sensor. La visualización de los resultados en tiempo real a través de una pantalla LCD ofrece una forma clara y directa de monitorear las mediciones sin necesidad de un dispositivo adicional como una computadora o un monitor serial. Este aspecto es especialmente útil en aplicaciones donde se necesita una interfaz de usuario autónoma.

Además de la visualización, el sistema también integra un control de LED y buzzer, lo que agrega una capa adicional de interactividad al proyecto. Los LEDs indican visualmente el estado del sistema, cambiando de verde a rojo según la distancia medida, lo cual puede ser útil para alertas inmediatas. El buzzer complementa esta funcionalidad, emitiendo un sonido cuando la distancia es inferior a un umbral predefinido (en este caso, 10 cm), lo cual puede servir como una alerta sonora en aplicaciones críticas donde la distancia debe mantenerse dentro de ciertos rangos.

Este tipo de proyecto es altamente adaptable y puede aplicarse en diversas áreas como:

* **Automatización del hogar**: Integrando sensores de proximidad para activar o desactivar dispositivos dependiendo de la distancia.
* **Seguridad**: Utilizando el sensor para detectar intrusos o medir distancias en sistemas de monitoreo.
* **Robótica**: Empleando este sistema en robots móviles para evitar obstáculos o medir distancias en entornos dinámicos.
* **Control de tráfico o acceso**: Usando el sensor para medir la proximidad de objetos o personas, activando alertas en entornos como fábricas o almacenes.

La implementación de este sistema también facilita la introducción de conceptos clave de electrónica y programación con Arduino, proporcionando una base sólida para proyectos más complejos en los que se necesite medir y reaccionar ante las distancias en función de distintos umbrales. Además, el uso de la pantalla LCD y la integración con componentes adicionales como LEDs y buzzers abre la puerta a más posibilidades de expansión, como la integración con sistemas de control remoto, interfaces gráficas o comunicaciones inalámbricas.

En resumen, este proyecto no solo ofrece una solución eficiente para medir distancias, sino que también proporciona una excelente base para aprender y experimentar con sensores, pantallas y otros componentes electrónicos. Gracias a su versatilidad, puede adaptarse y expandirse para satisfacer las necesidades de una variedad de aplicaciones en la vida real, lo que lo convierte en un excelente punto de partida para quienes deseen adentrarse en el mundo de la automatización, la seguridad y la robótica.

