Theory Group Assignment

Only for course Teacher						
		Needs Improvement	Developing	Sufficient	Above Average	Total Mark
Allocate mark & Percentage		25%	50%	75%	100%	5
Clarity	1					
Content Quality	2					
Spelling & Grammar	1					
Organization and Formatting	1					
					Total obtained mark	
Comments						

Semester: Spring 2024

Student Name: Nur Ahmed, Md. Emon Matubbor

Student ID: 222-35-1111, 222-35-1155

Batch: 38 **Section: B**

Course Code: SE 532

Course Name: Introduction to Robotics

Course Teacher Name: Md. Hafizul Imran

Designation: Lecturer (Senior Scale)

Submission Date: 07/03/2024

Title: Smart Doorbell with Distance Sensor

Project Description:

The Smart Doorbell with Distance Sensor project aims to create a unique doorbell system that not only alerts you when someone is at your door but also provides an indication of the distance from the doorbell to the person approaching. This project utilizes an Arduino Uno R3 board along with an HC-SR04 ultrasonic distance sensor. When someone approaches the doorbell, the ultrasonic sensor detects their presence and triggers a notification, while also displaying the approximate distance from the sensor to the person. This project enhances traditional doorbell functionality with distance sensing capabilities.

Equipment List:

Arduino Uno R3 board x1

HC-SR04 Ultrasonic Sensor x1

Passive Buzzer x1

Breadboard x1

Jumper wires

Circuit:

Connect the VCC pin of the HC-SR04 sensor to the 5V pin on the Arduino.

Connect the GND pin of the HC-SR04 sensor to the GND pin on the Arduino.

Connect the TRIG pin of the HC-SR04 sensor to digital pin 7 on the Arduino.

Connect the ECHO pin of the HC-SR04 sensor to digital pin 6 on the Arduino.

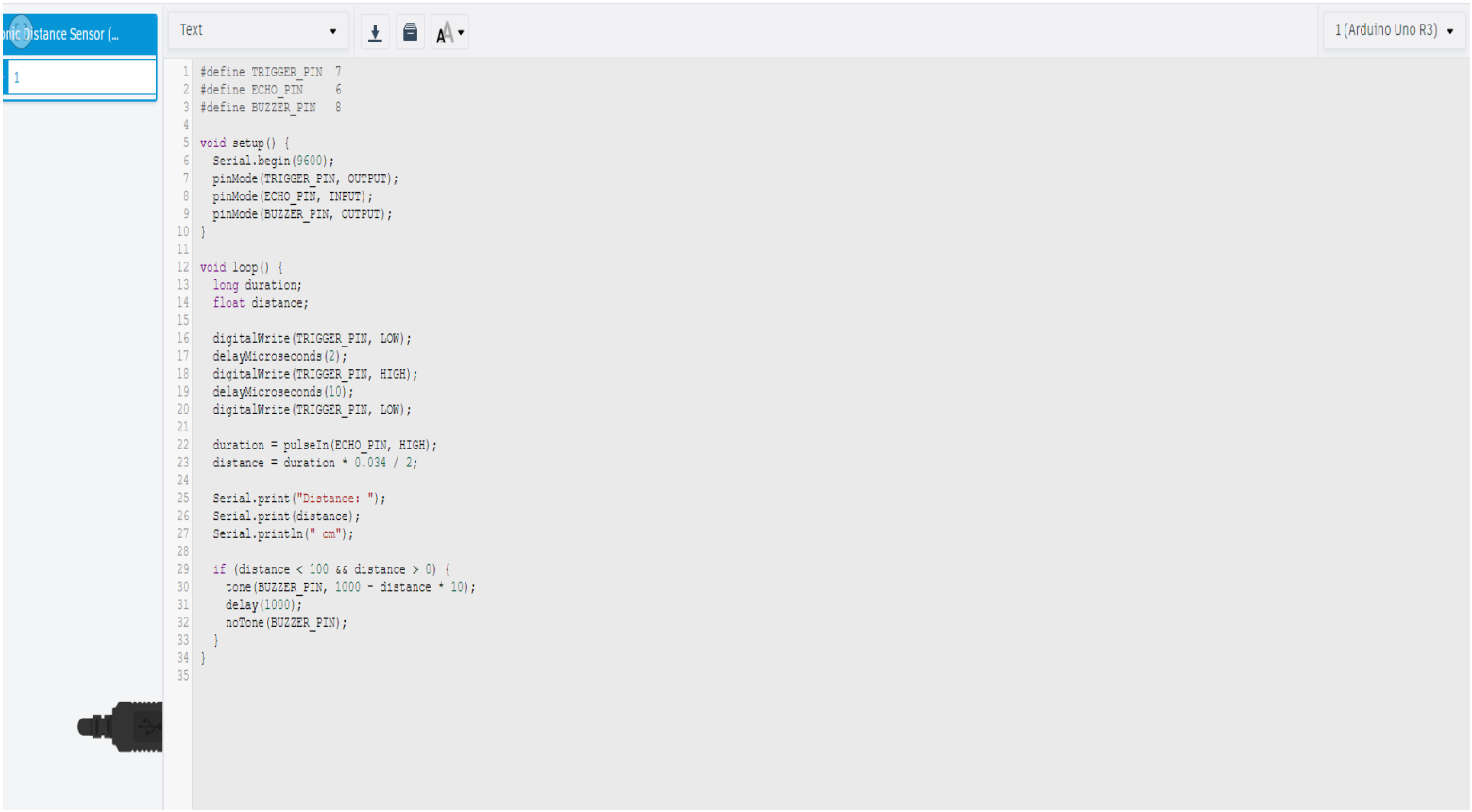
Connect the positive leg of the passive buzzer to digital pin 8 on the Arduino.

Connect the negative leg of the passive buzzer to the GND pin on the Arduino.

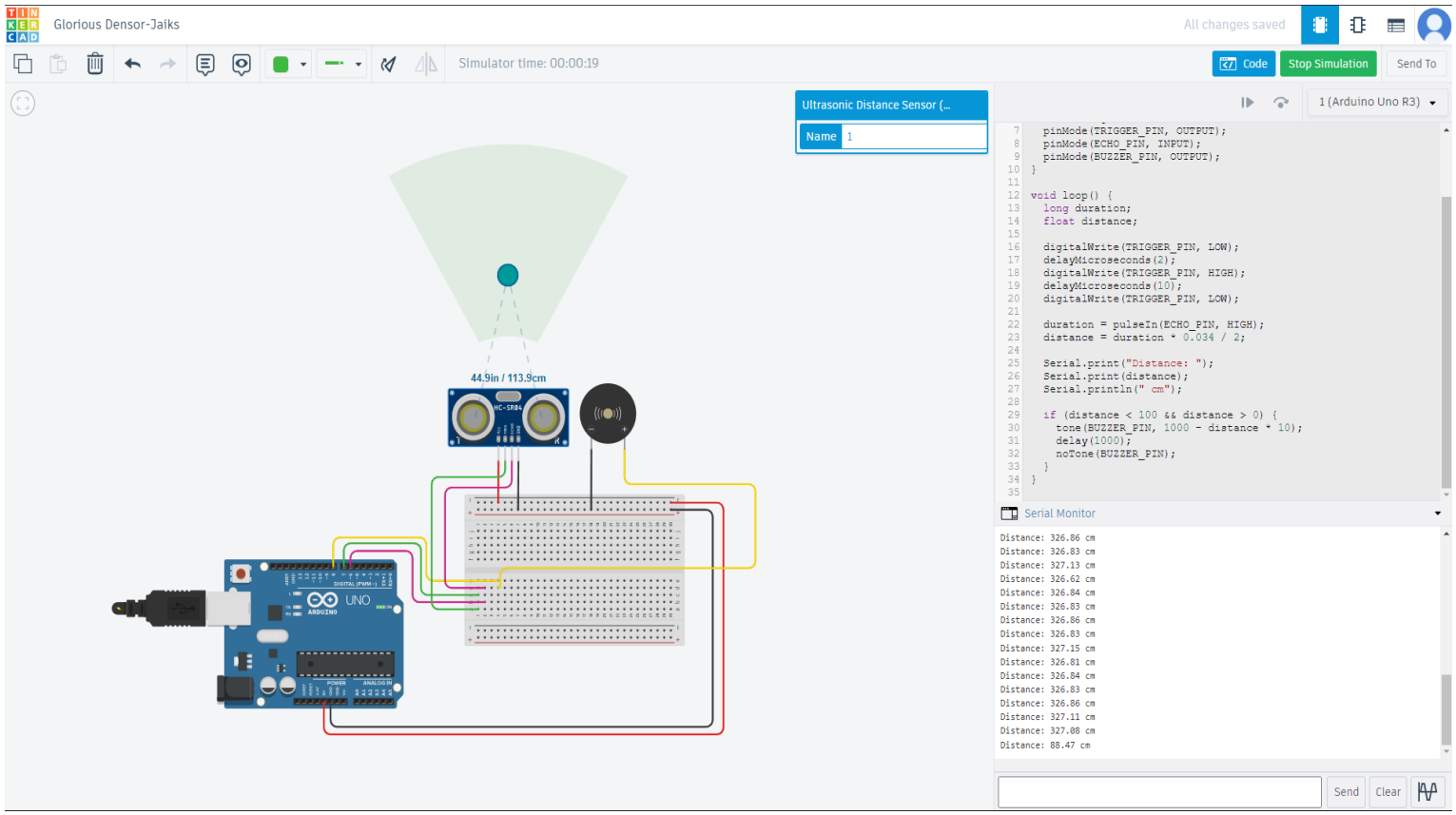
Explanation1:

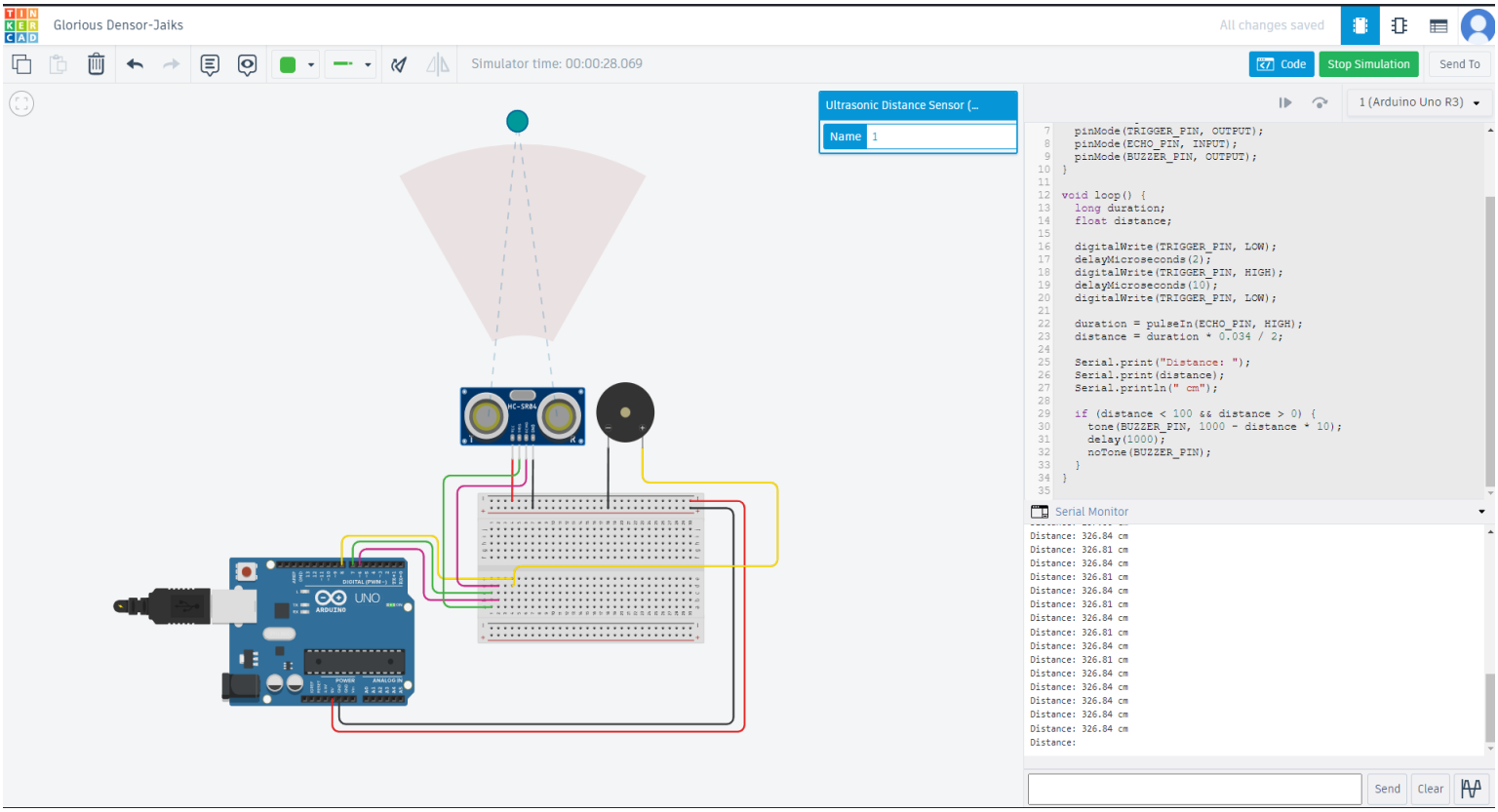
The HC-SR04 sensor detects the presence of someone approaching the doorbell by sending out ultrasonic waves and measuring the time taken for the waves to bounce back. When someone is detected within a certain range, the Arduino triggers the passive buzzer to produce an alert sound, indicating someone is at the door. Additionally, the Arduino calculates the distance based on the time taken for the ultrasonic waves to return and displays this distance through the buzzer's sound frequency.

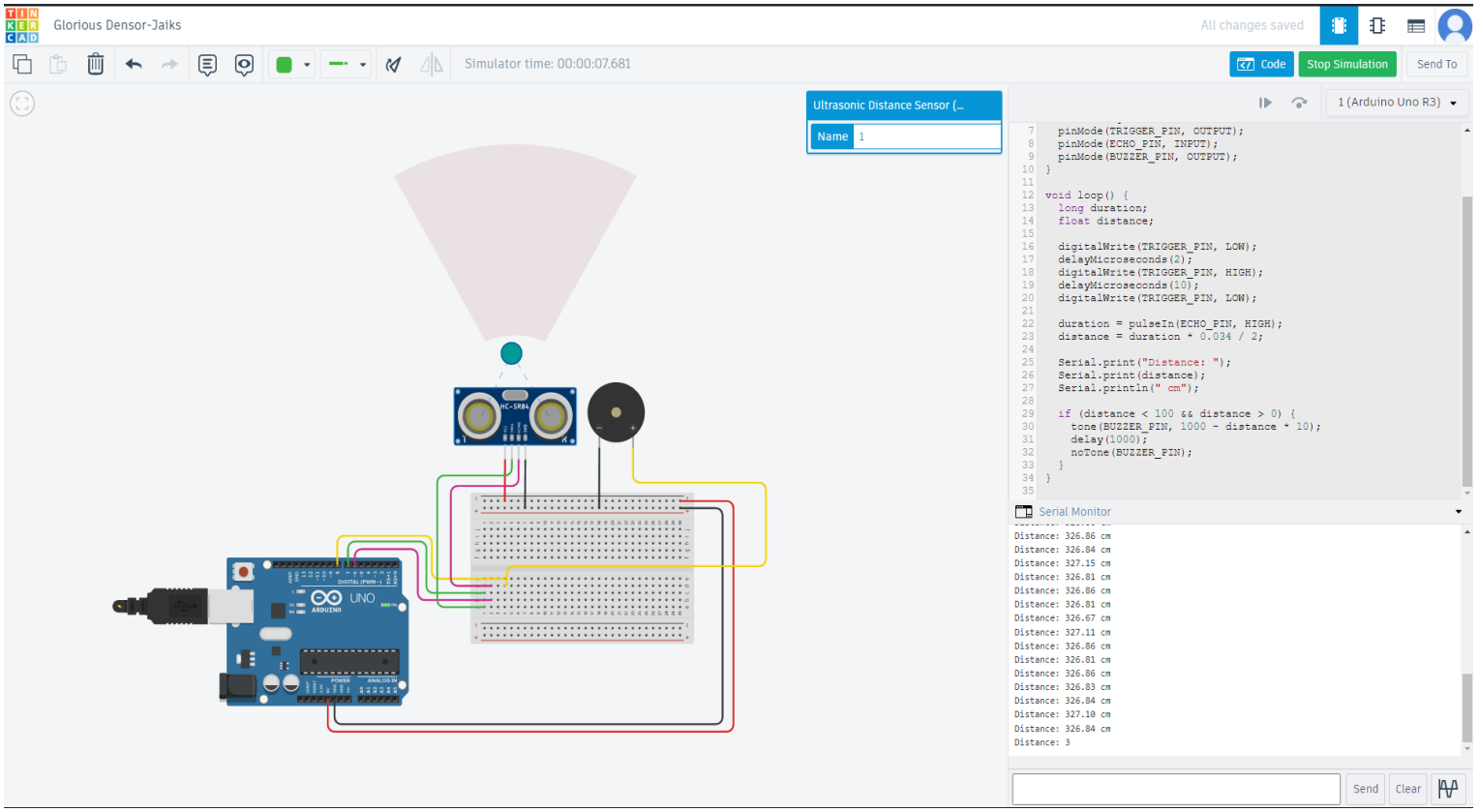
Code:

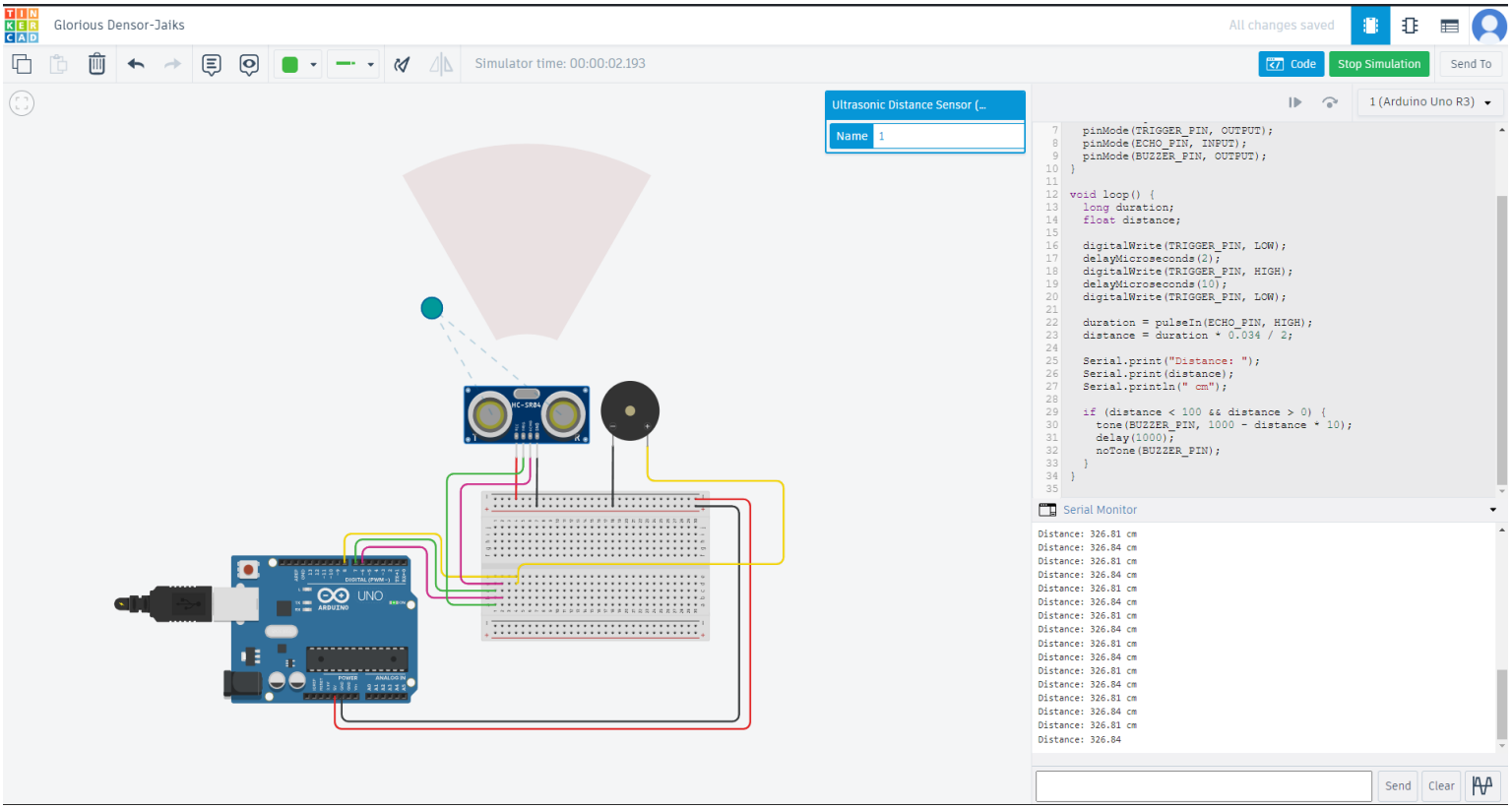

Distance Sensor (...
1
Text
1 (Arduino Uno R3)
#define TRIGGER_PIN 7
#define ECHO_PIN 6
#define BUZZER_PIN 8
void setup() {
Serial.begin(9600);
pinMode(TRIGGER_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
pinMode(BUZZER_PIN, OUTPUT);
}
void loop() {
long duration;
float distance;
digitalWrite(TRIGGER_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIGGER_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGER_PIN, LOW);
duration = pulseIn(ECHO_PIN, HIGH);
distance = duration * 0.034 / 2;
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");
if (distance < 100 && distance > 0) {
tone(BUZZER_PIN, 1000 - distance * 10);
delay(1000);
noTone(BUZZER_PIN);
}
}

Screenshots:






Glorious Densor-Jaiks
All changes saved
Simulator time: 00:00:07.681
Code
Stop Simulation
Send To
Ultrasonic Distance Sensor (...
Name 1
1 (Arduino Uno R3)
pinMode(TRIGGER_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
pinMode(BUZZER_PIN, OUTPUT);
}
void loop() {
long duration;
float distance;
digitalWrite(TRIGGER_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIGGER_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGER_PIN, LOW);
duration = pulseIn(ECHO_PIN, HIGH);
distance = duration * 0.034 / 2;
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");
if (distance < 100 && distance > 0) {
tone(BUZZER_PIN, 1000 - distance * 10);
delay(1000);
noTone(BUZZER_PIN);
}
}
Serial Monitor
Distance: 326.86 cm
Distance: 326.84 cm
Distance: 327.15 cm
Distance: 326.81 cm
Distance: 326.86 cm
Distance: 326.81 cm
Distance: 326.67 cm
Distance: 327.11 cm
Distance: 326.86 cm
Distance: 326.81 cm
Distance: 326.86 cm
Distance: 326.83 cm
Distance: 326.84 cm
Distance: 327.10 cm
Distance: 326.84 cm
Distance: 3
Send
Clear


Glorious Densor-Jaiks
All changes saved
Simulator time: 00:00:02.193
Code
Stop Simulation
Send To
Ultrasonic Distance Sensor (...
Name 1
1 (Arduino Uno R3)
pinMode(TRIGGER_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
pinMode(BUZZER_PIN, OUTPUT);
}
void loop() {
long duration;
float distance;
digitalWrite(TRIGGER_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIGGER_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGER_PIN, LOW);
duration = pulseIn(ECHO_PIN, HIGH);
distance = duration * 0.034 / 2;
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");
if (distance < 100 && distance > 0) {
tone(BUZZER_PIN, 1000 - distance * 10);
delay(1000);
noTone(BUZZER_PIN);
}
}
Serial Monitor
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84 cm
Distance: 326.81 cm
Distance: 326.84
Send
Clear

Explanation2:

We define the pins for the ultrasonic sensor's trigger, echo, and the passive buzzer.

In the setup() function, we initialize serial communication, set the trigger pin as an output, the echo pin as an input, and the buzzer pin as an output.

In the loop() function, we measure the distance using the ultrasonic sensor. If the distance is within a certain range (here, less than 100 cm), we generate a sound through the buzzer.

The frequency of the sound generated by the buzzer is inversely proportional to the distance, providing a unique indication of the distance to the person approaching the doorbell.