CSE461: Introduction to Robotics

LAB REPORT 2

Name of the experiment:

Measuring distance using ultrasonic sensor

Group: 05

Name: Anindita Dutta

Student Id: 22101031

Section: 03

G-Suite: anindita.dutta@g.bracu.ac.bd

Date Of Submission: 1.06.2024

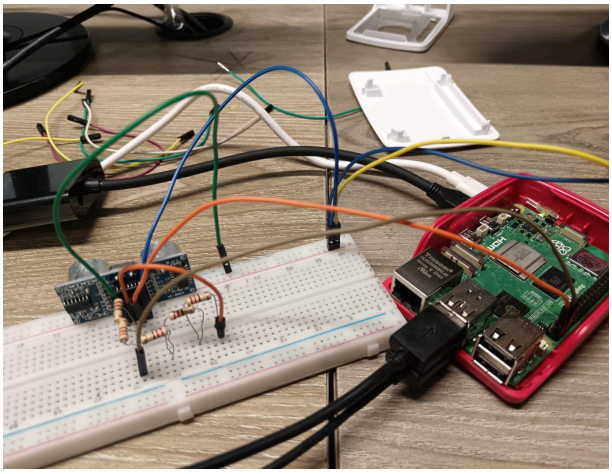
Objective:

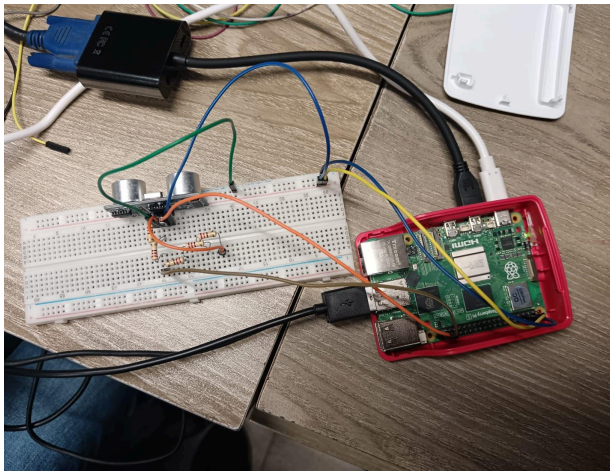
1. Understand how ultrasonic sensors measure distance.
2. Learn the components of the Raspberry Pi and connect sensors to GPIO pins.
3. Install Python libraries and run Python programs on the Raspberry Pi.
4. Measure distance using Python code and an ultrasonic sensor.
5. Test and troubleshoot the sensor's accuracy in different scenarios.

Equipment:

- Raspberry Pi
- Ultrasonic Sensor (HC-SR04)
- Breadboard
- Jumper Wires
- 5 resistors of 220 ohm
- MicroSD Card
- USB Cable
- Monitor, Keyboard, and Mouse (Optional).

Experimental Setup:





The Raspberry Pi is connected to the desktop through an HDMI connector, and the breadboard is connected to the Raspberry Pi via jumper wires. The desktop operates the Raspberry Pi, and we have written a few lines of code so that the Raspberry Pi reads the distance from the ultrasonic sensor and calculates it accordingly.

The ultrasonic sensor (HC-SR04) is connected with the trigger pin to GPIO21 (pin 40) and the echo pin to GPIO20 (pin 38) of the Raspberry Pi. A microSD card is used for the Raspberry Pi's operating system, and a USB cable powers the Raspberry Pi. We have connected ground to ground and VCC to 5v DC power.

Five 220-ohm resistors are used to protect the circuits. Two of the resistors were in series and others were also in series and these two sets were in parallel. A monitor, keyboard, and mouse

are connected for direct interaction with the Raspberry Pi if needed. This setup allows the Raspberry Pi to measure the distance using the ultrasonic sensor and display the readings.

Code:

```
#!/usr/bin/env python3
from gpiozero import DistanceSensor
from time import sleep


# Initialize the DistanceSensor using GPIO Zero library
# Trigger pin is connected to GPIO 21, Echo pin to GPIO 20
sensor = DistanceSensor(echo=20, trigger=21)


try:
    # Main loop to continuously measure and report distance
    while True:
        dis = sensor.distance * 100  # Measure distance and convert from meters to centimeters
        print('Distance: {:.2f} cm'.format(dis))  # Print the distance with two decimal precision
        sleep(0.3)  # Wait for 0.3 seconds before the next measurement


except KeyboardInterrupt:
    # Handle KeyboardInterrupt (Ctrl+C) to gracefully exit the loop
    pass
```

Results:

We finished the experimental setup and attached the Raspberry Pi's power line. Then, using a desktop monitor, we entered the code into the Raspberry Pi.. After completing all these steps, when we put our hands in front of the sensor as a tester ,we found the distances as the outputs on the monitor through our code.We waved our hands forward and backward to test if the sensor was capable of identifying the distances properly. And the sensor was identifying the distances properly every time.

Discussion:

In this lab, we learned how to use an Ultrasonic sensor with the help of Raspberry-Pi 4. We used an Ultrasonic sensor to measure distance by launching sound waves and timing how long it takes for them to return. We first gave the connection between the sensor and Raspberry Pi. For connection, we used jumper wires, resistors. After giving the connections, we wrote a few lines of code by Thone software so that the sensor could measure the distance . We tested by waving our hands in front of the sensor to check if the sensor could detect . We successfully completed the whole process. An updated code was given later to make the process more efficient. We have learned how theoretically and practically ultrasonic sensors work. We did not face any difficulties during the lab work. We were careful throughout the whole lab so that no component could be damaged and was successful to complete the task.

Question-Answers:

Question 1. Why are the resistors used?

Answer:

Raspberry Pi 4 operated at 3.3 V but this HC_SR04 Ultrasonic sensor operates at 5V. In order to step down the 5V signal from the sensor to a safe 3.3V level for the GPIO pins, resistors can be used to construct a voltage divider. And also for saving both sensor and Raspberry pi from excessive current flow , resistors can play an important role.