MRE 320 SENSORS AND ACTUATORS

INDIVIDUAL PROJECT MILESTONE 1
ULTRASONIC #1

UGOCHUKWU C. ELEONU

September 25, 2022

INTRODUCTION

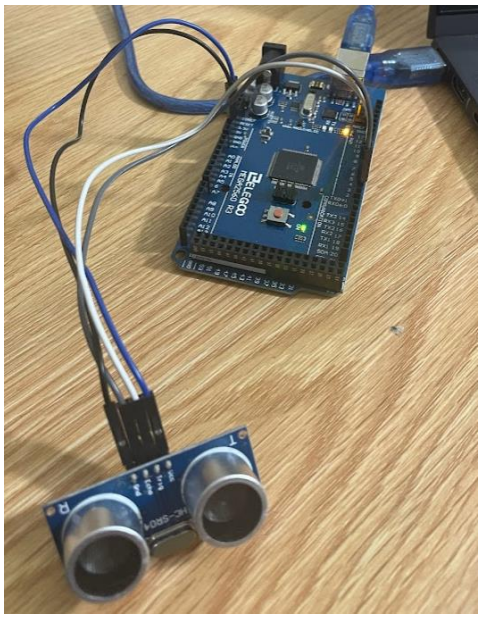
The assigned sensor was the Ultrasonic sensor HC-SR04, this is a sensor that sends an ultrasonic sound wave from one end which acts as its emitter and when this sound wave hits an obstacle, it is reflected to the sensor's receiver and the time traveled by this sound wave can be used to calculate the distance between the sensor and an object.

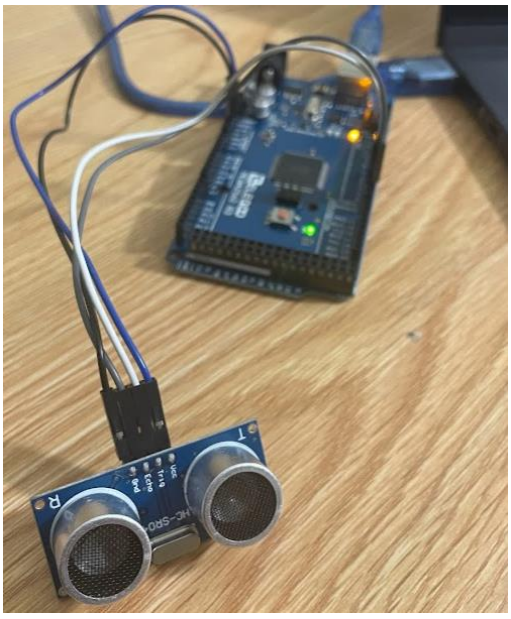
The HC-SR04 Ultrasonic Range Sensor Features:

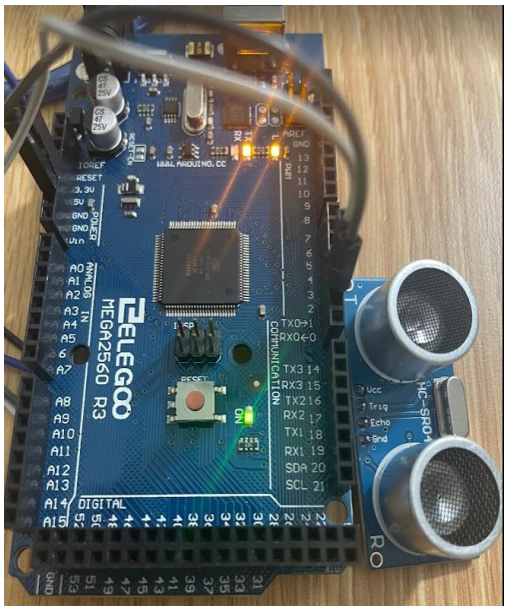
- Input Voltage: 5V
- Current Draw: 20mA (Max)
- Digital Output: 5V
- Digital Output: 0V (Low)
- Working Temperature: -15°C to 70°C
- Sensing Angle: 30° Cone
- Angle of Effect: 15° Cone
- Ultrasonic Frequency: 40kHz
- Range: 2cm - 400cm
- Resolution: 0.3cm
- Trigger input pulse width: 10µs
- Dimensions: 45mm x 20mm x 15mm

TEST BENCH

To test the ultrasonic range sensor, I connected it to an Arduino and using the IDE to code instructions to run the sensor. For the connections, the first pin Vcc which is the power supply of the sensor was connected to the 5V power source on the Arduino MEGA, the second pin which is the trigger pin and this is the pin for the transmitter was connected to port 3 of the digital PWM while the third pin which is the echo pin which is the pin for the receiver was connected to port 2 of the digital PWM of the Arduino, finally the last pin GND was connected to ground. These connections can be seen in the images below:







ARDUINO CODE

The Arduino code is divided into 3 main parts; The first part defines the pins used on the board and declares the variables to be measured, the second part is the setup which defines the input and output pins and while the last part which is the loop is the main part of the code that tests the sensor, it Initially sets the trigger to low to turn off any deviations or residue, then when set to high it saves the time taken for the ultrasonic sound to go and return to the receiver. This time is then multiplied with the speed of sound (since velocity = distance/time therefore distance = velocity * time) and divided by 2 since the distance covered is both to and from the destination object. This distance value is then shown as an output on the screen. The following code below was used to test the sensor, and the output is seen in the image below it.

```
// Arduino Ultrasonic Sensor HC-SRO4
// Arduino MEGA 2560 R3
// Arduino IDE 2.0.0

#define echo_pin 2 //Echo pin of sensor is attached to PWM 2 on arduino
#define trig_pin 3 //Trig pin of sensor is attached tp PWM 3 on arduino

long time_taken; //time taken for ultrasonic sound to travel to destination and
back to receiver
int distance_travelled; // distance travelled by ultrasonic sound

void setup() {

  pinMode(trig_pin, OUTPUT); // trig_pin is set as the output
  pinMode(echo_pin, INPUT); // echo_pin is set as the input
  Serial.begin(9600); //Baudrate set to 9600
  Serial.println("Ultrasonic Sensor HC-SRO4 Test");
  Serial.println("Arduino MEGA 2560 R3");
}

void loop() {

  // trig pin set to low
  digitalWrite(trig_pin, LOW);
  delayMicroseconds(10);

  // trig pin set to high
  digitalWrite(trig_pin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig_pin, LOW);

  //stores value gotten from echo pin
  time_taken = pulseIn(echo_pin, HIGH);
```

```
  //uses stored value to calculate distance
  distance_travelled = time_taken * 0.034/2;

  Serial.print("Distance: ");
  Serial.print(distance_travelled);
  Serial.println(" cm");
}
```

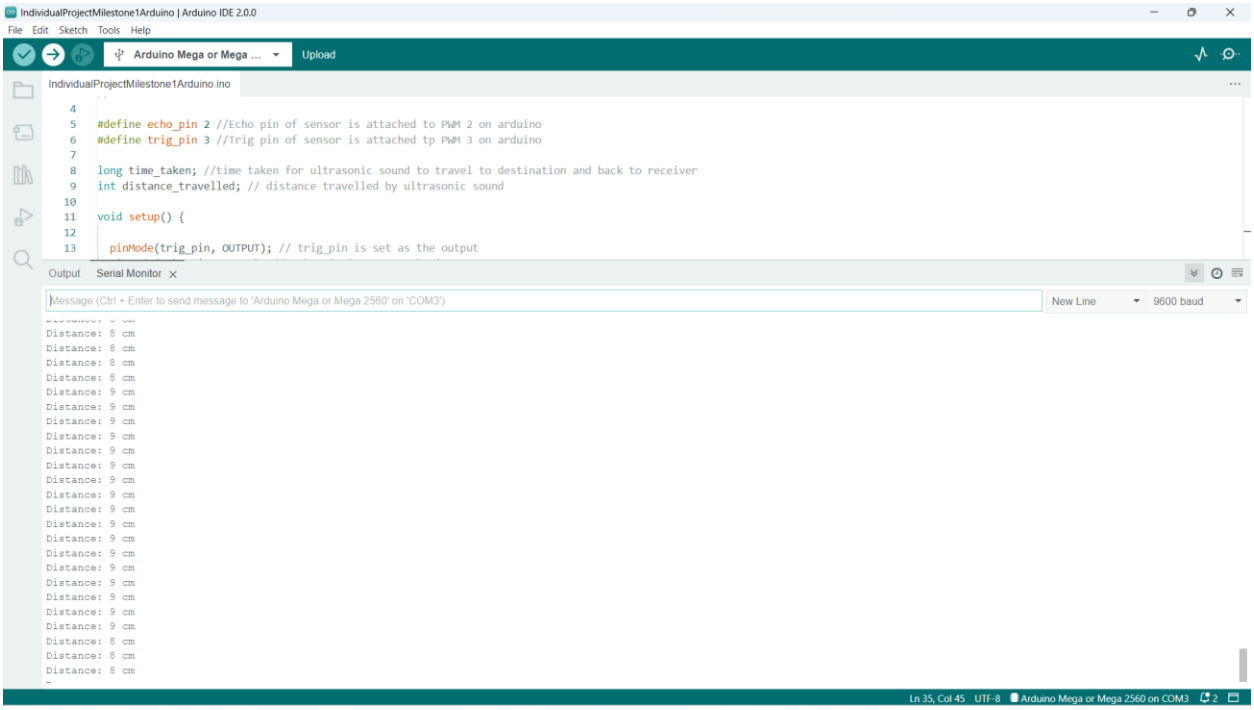


Figure 1 Arduino code output for Ultrasonic sensor

TEST PLAN

The following will be specifications of the sensor will be tested:

RANGE:

I will be testing the accuracy of the specified range by the manufacturer, the main tool to be used will be the same used in this section and a tape rule, after connecting the circuit as seen above, I will put an obstacle at several distances of certain increments (for example, every 10cm) this will be done and repeated and used to confirm at what distance the sensor readings stop working.

ERROR:

For this section the same equipment used in range testing will be used here, I will premeasure a specified distance between two points and record it, the sensor will be placed at one end of the measured distance while the obstacle placed at the other end, the recorded distance by the sensor will then be compared to the premeasured distance. This will be repeated for a range of values, the graph of true distance vs measured distance will be plotted and a relationship established which will be then used to find the error in the sensor.

REPEATABILITY

This will be tested by repeating the same measurement several times at a specified distance and recorded and compared. This will be done for different specified distances.

RESOLUTION

The resolution will also be tested, the required tools will be the same as the other tests above, this time the procedure will involve having minimal changes to the obstacle less than, equal to or greater than the company specified resolution, this will help verify the value of resolution given by the manufacturer.

SENSITIVITY

This will be done in several ways, first the sensor will be tested in different weather conditions to see if the same distance to an object can be detected in different conditions. Also, the surface of this object will be altered, and the sensor used to confirm if there is any change or if the measurements are the same. Finally, the size of the object will be altered to see if the sensor has a minimum required size for the object to be detected to be.