Group - 11

Soumava Paul 16EE10056
Yerramsetty Rohit 16EE10055

Parking Assistant

Aim:

Developing an embedded parking assistant system with suitable audio-visual cues for driver assistance

Apparatus Required:

● Arduino Uno Board x3
● PC with Arduino installed
● Ultrasonic Sensor x3
● LCD x3
● RGB LED Module x3
● Buzzer x3

Procedure:

● We connect the circuit as shown in the figure below. Additionally, we also used an RGB LED module and buzzer.
● This is however, only a part of the total setup we had planned. Any further connections were not possible due to unavailability of sufficient pins on a single Arduino Board.
● The entire setup would consist of 3 such identical circuits with a separate Arduino Microcontroller for each part. However, we did not proceed with that due to repetitive nature of the design.
● The LCD is used for displaying the distance in centimetres and a corresponding bar calibrated with a maximum distance of 30 centimetres.
● The RGB module glowed with red light when the distance was less a predefined threshold of 8 cms, blue light for distance between 8 and 15 cms and green light beyond that.
● A simple buzzer was also incorporated to sound a beeping alarm in the red-light zone.

Circuit Diagram:
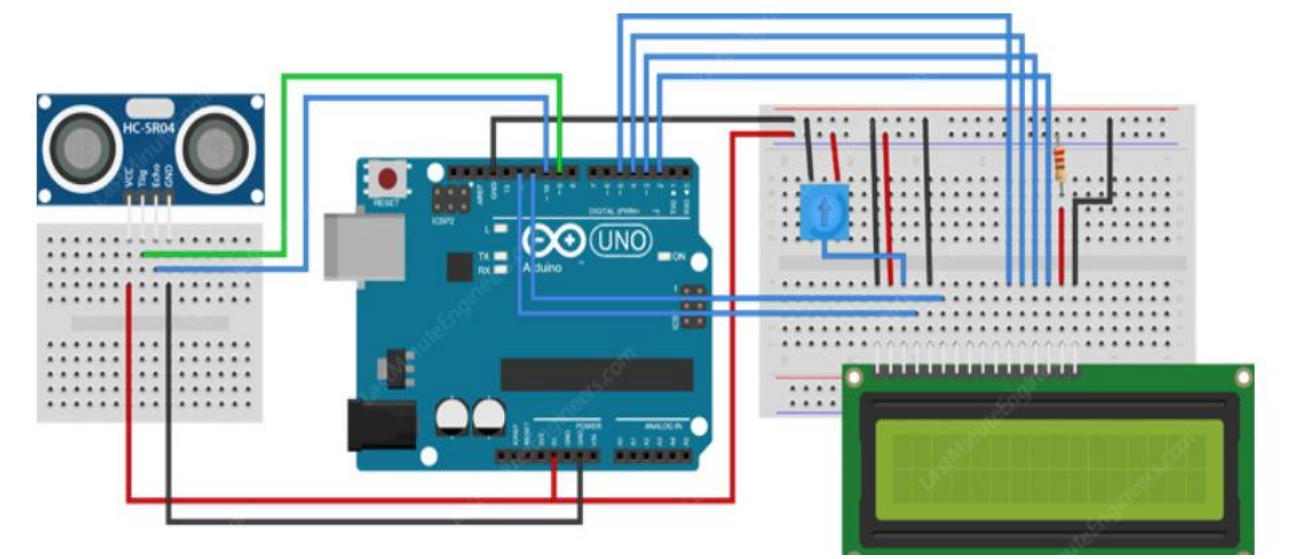
Code:

#include <LiquidCrystal.h>

#include <LcdBarGraph.h>

#define max_distance 30

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

LcdBarGraph lbg(&lcd, 16, 0, 1);

const int trigPin = 9;

const int echoPin = 10;

int redPin= 6;

int greenPin = 7;

int bluePin = 8;

```
int buzzer = 13;
long duration;
int distance;
const int freq1 = 220; //Low Note A
const int freq2 = freq1 * 2; //Med
const int freq3 = freq1 * 3; //Hi
const int freq4 = freq1 * 4; //Very Hi
boolean play=false;

void setup()
{
  lcd.begin(16,2);

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
}

void setColor(int redValue, int greenValue, int blueValue) {
  analogWrite(redPin, redValue);
  analogWrite(greenPin, greenValue);
  analogWrite(bluePin, blueValue);
}
```

```
void makeBeep(int intervalSeconds, int howlongSeconds, int freq) {
  tone(buzzer, freq, howlongSeconds * 1000);
}


void loop()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);


  duration = pulseIn(echoPin, HIGH, 5000);


  distance= duration*0.034/2;


  if(distance<=8){
   setColor(255, 0, 0);
   play=true;
   }
  else if(distance>=8 && distance<=15){
   setColor(0, 0, 255);
   play=false;
   }
  else{
   setColor(0, 255, 0);
```

```
    play=false;
    }
  if (play == true) {
   makeBeep(1, 1, freq1);
   makeBeep(5, 1, freq1);
   makeBeep(5, 1, freq2);
   makeBeep(1, 1, freq3);
   makeBeep(1, 1, freq4);
   }
  if(play==false){
   noTone(buzzer);
   }

  lcd.setCursor(0,0);
  lcd.print("Distance: ");
  lcd.print(distance);
  lcd.print(" cm");

  lcd.setCursor(0,1);
  lbg.drawValue(distance, max_distance);
  delay(500);
}
```

Links

The video demonstration and the arduino code file are available in the following links:

https://photos.app.goo.gl/GEA7mGzgUmnPhs2L7

https://drive.google.com/file/d/13qyGEcmvcI6N5VhhZde-tchgWX0G76Qa/view?usp=sharing

Conclusion

In this project, we have designed a part of a parking assistant system. The entire setup would consist of 3 such identical parts. Further enhancements might include adding waterproof sensors such as JSN SR-04 for navigation in rainy weather.