EEE-204 Project Report (Spring 2020)

Project Title: Social Distancing Alert System for Covid-19

Course Title: Electronic Devices and Circuit & Pulse Techniques lab

Course Code: EEE-204
Section: DA
Program: B.Sc. In Computer Science & Engineering
Date of Submission: 16 May, 2021

Submitted To:	Submitted By:
Mr. Sharif Nafis Mahmood Lecturer Department: EEE **Green University Of Bangladesh**	Md. Torikul Islam ID:192002131 Department: CSE **Green University Of Bangladesh**

Table of Contents

Title: Social Distancing Alert System for Covid-19

Introduction:

Social Distancing Project using Arduino, Ultrasonic Sensor, and LEDs and buzzer. As per the World Health Organization "WHO" the safe distance is at least maintain 1 meter or 3 feet, while some people suggests that this distance should be around 6 feet.

Here is an Arduino based social distancing alarm cap meant for Covid-19 pandemic or similar disease outbreak. Maintaining the right gap for social or physical distancing is not always feasible. When we are outside, we forget to maintain the distance as soon as we get involved in our daily work.

This novel circuit alerts you if you come too close to someone either from the front, back, right, or left direction. A cap fitted with four ultrasonic distance sensors connected to Arduino Uno senses the minimum safe distance and alerts you through a buzzer.

Objective:

Social distancing also called "physical distancing," means keeping a safe space between yourself and other people who are not from your household. To practice social or physical

distancing, stay at least 3 to 6 feet (about 2 arms' length) from other people who are not from your household in both indoor and outdoor spaces.

We can reduce your chances of being infected or spreading COVID-19 by taking some simple precautions:

- Regularly and thoroughly clean your hands with an alcohol-based hand rub or wash them with soap and water.
- Maintain at least 1 meter (3 feet) distance between yourself and others. Why? When someone coughs, sneezes, or speaks they spray small liquid droplets from their nose or mouth which may contain virus. If you are too close, you can breathe in the droplets, including the COVID-19 virus if the person has the disease.

In this project we are designing a digital circuit which is performed by our programming to keep distance and reduce this pandemic COVID-19. Only Self-awareness can save us from COVID-19. So this project is much effective in this pandemic situation.

Motivation:

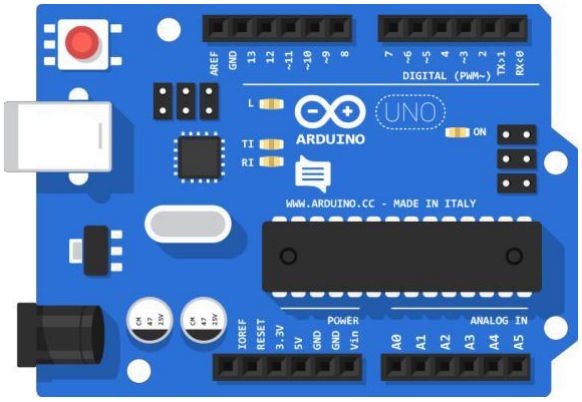
The current COVID-19 pandemic shows negative effects on health as well as social and economic life. The most critical and challenging task is to revive public life while minimizing

the risk of an infection. Reducing interactions between people by means of social distancing is an effective and prevalent measure to reduce the risk of an infection and spread of the virus within a community. Current developments in several countries show that this measure may be technologically accompanied by mobile applications (apps).We build not as an app, but physically it will help you to keep distance from anonymous.

Required Components:

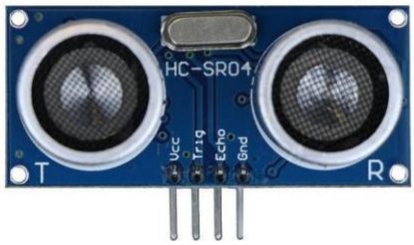
- ✓ Tinkercad

- ✓ Arduino Uno



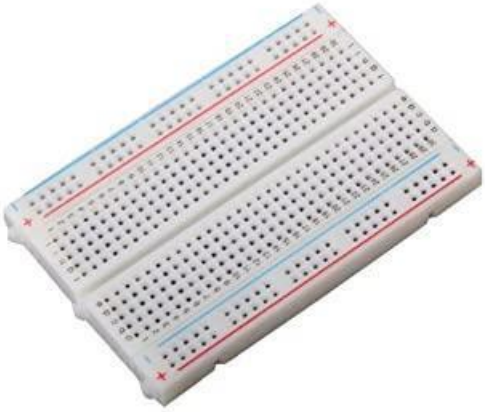
✓ Neopixel 12 Ring

✓ Ultrasonic Sensor (HC-SR04)



✓ Piezo Buzzer

✓ Breadboard



Procedure:

At first we go to the online simulation site named Tinkercad. Create a new project named Social Distancing Alert System.

Than find the component from search bar. Now connection is given as it is:

At first we take (+5v) from ardunio nano and ground and connect it breadboard. This is the RGB LED 12 module from the PCB Way Company which can be powered up using the 5v cell and can also be powered up using 5 Volts from the

Arduino board. As I will power up this LED module using the Arduino's 5V, so I will use the 5V cell. The metal part is the + which connects with the 5V side of the cell, while the other point is on the PCB and is the ground point.

HC-SR04 ultrasonic sensor utilizes sonar to determine distance from an object (or humans as in this case). It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. Its range is from 2cm to 400cm with an accuracy of 5mm. The module consists of an ultrasonic transmitter, receiver, and control circuit. HC-SR04 ultrasonic sensor is shown in Fig. 3. It has the following four pins, namely:

VCC: +5V DC, Trig: Trigger (input), Echo: Output ,GND: Ground.

It can be easily interfaced with Arduino Uno.

The working of HC-SR04 is simple. When the sensor is triggered (a short 10μs pulse to the trigger input pin 2 from Arduino Uno board), the transmitter sends a high-frequency sound signal at 40kHz. The transmitted signal reflects back from the nearby object and is picked up by the receiver.

The output of HC-SR04 (echo-pin 3) is a pulse where the width of the pulse is proportional to the distance of the object. The

width of the pulse can be calculated by the pulse In( ) function in the code. Arduino program calculates the distance of the objects in all its four directions.

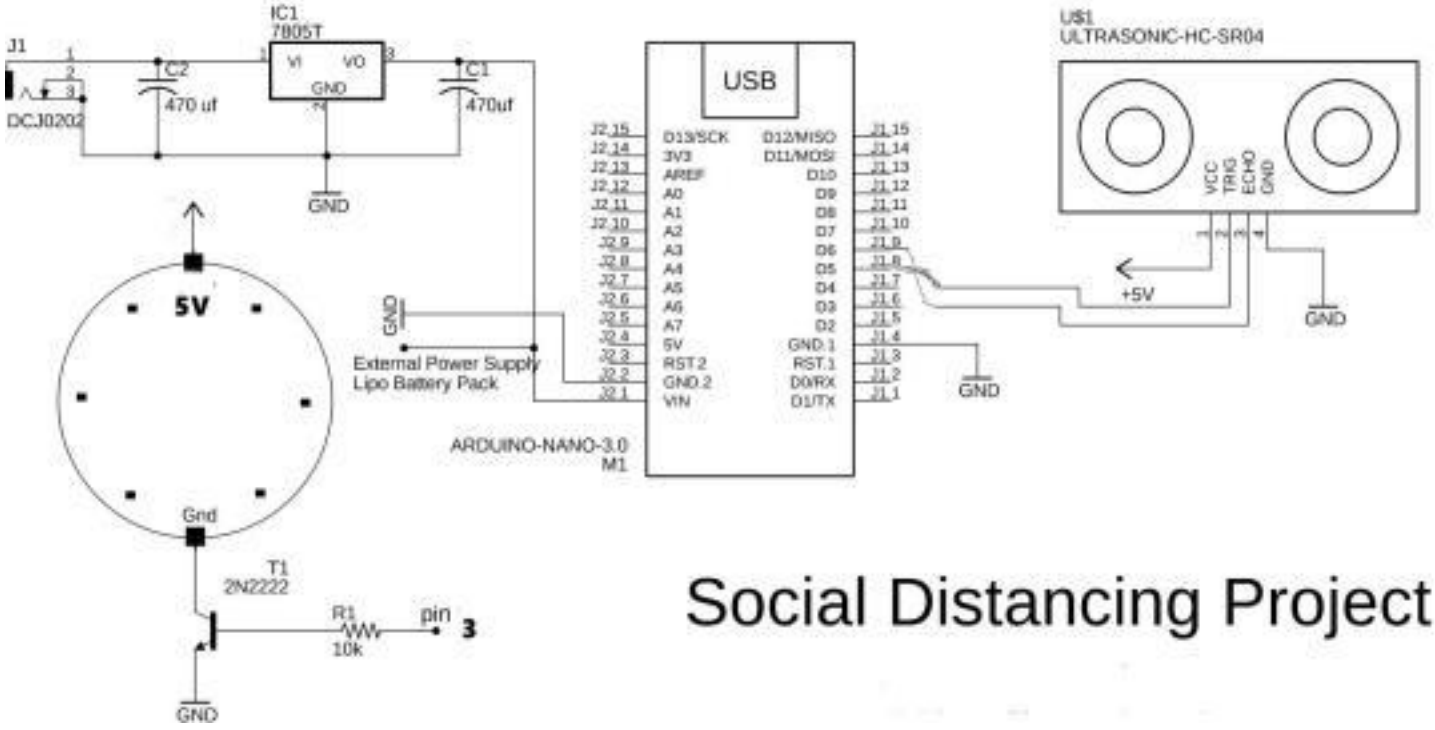
Echo pin is connected to digital pin 6 and trigger pin is connected to digital pin 5. And Vcc is connected to breadboard 5V and GND is connect to GDN.

Neopixel 12 has 4 pin. Power is connected to +5v and GND with GND in breadboard. Another pin Vin is connected to Digital pin 3.

Now pieozo buzzer has two pin. Positive is connected to digital pin 2 and another is connected to GND.

That's all.

Circuit Diagram:



Required Calculation and Coding:

```
#include <Adafruit_NeoPixel.h>

// Project Created By Md Torikul

Islam int ledPin= 3;

int ledNo= 12;

Adafruit_NeoPixel strip=
Adafruit_NeoPixel(ledNo,ledPin,NEO_RGB+NEO_KHZ800);
```

```
int buzzerPin= 2;
int echoPin= 6;
int trigPin= 5;
int minDistance = 100;
int maxDistance = 300;
void setup()
{
  pinMode(buzzerPin, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial. begin(9600);
  strip.begin();
  for(int i = 0; i < ledNo; i++)
  {
   strip.setPixelColor(i,strip.Color(0,0,0));
  }
  strip.show();
}


void loop()
{
  int distance = calcDistance();
  Serial.println(distance);
  int ledsToGlow = map(distance, minDistance, maxDistance,
ledNo, 1); // Mapping
```

```
  Serial.println(ledsToGlow);
  if(ledsToGlow == 12)
  {
    digitalWrite(buzzerPin, HIGH);
  }
  else
  {
    digitalWrite(buzzerPin, LOW);
  }
  for(int i = 0; i < ledsToGlow; i++)
  {
    if(i < 4)
    {
strip.setPixelColor(i,strip.Color(50,0,0));//green,red,blue
    }
    else if(i >= 4 && i < 8)
    {
strip.setPixelColor(i,strip.Color(50,50,0));//green,red,blue
    }
    else if(i >= 8 && i < 12)
    {
strip.setPixelColor(i,strip.Color(0,50,0));//green,red,blue
    }
```

```
  }
  for(int i = ledsToGlow; i < ledNo; i++)
  {
    strip.setPixelColor(i,strip.Color(0,0,0));
  }
  strip.show();
  delay(50);
}
            //----------------------Calculation Part------------
- --------------//


int calcDistance()
{
  long distance,duration;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);



  distance = (duration)2/29.1; // Distances in Centimeter


  Serial.println("Distance in Centimeter:");
```

```
if(distance >= maxDistance)
  {
    distance = maxDistance;
  }
  if(distance <= minDistance)
  {
    distance = minDistance;
  }
  return distance;
}
```

Result analysis & Discussion:

When we start simulation to show our devise active one led is glow from any distance. We initial it 300cm. Which color is green. Than we deal with reaming 11 RGB leds. The number of leds glow is depend on distance. First condition we check all led are glow or not? If glow than buzzer is high and it will alarm when distance is close to 100cm.

First led glow green than yellow finally red, when 4 red led is glowing than it will detect someone close to him than buzzer is ringing but if the distance increase than buzzer will stop ringing.

Project Image:

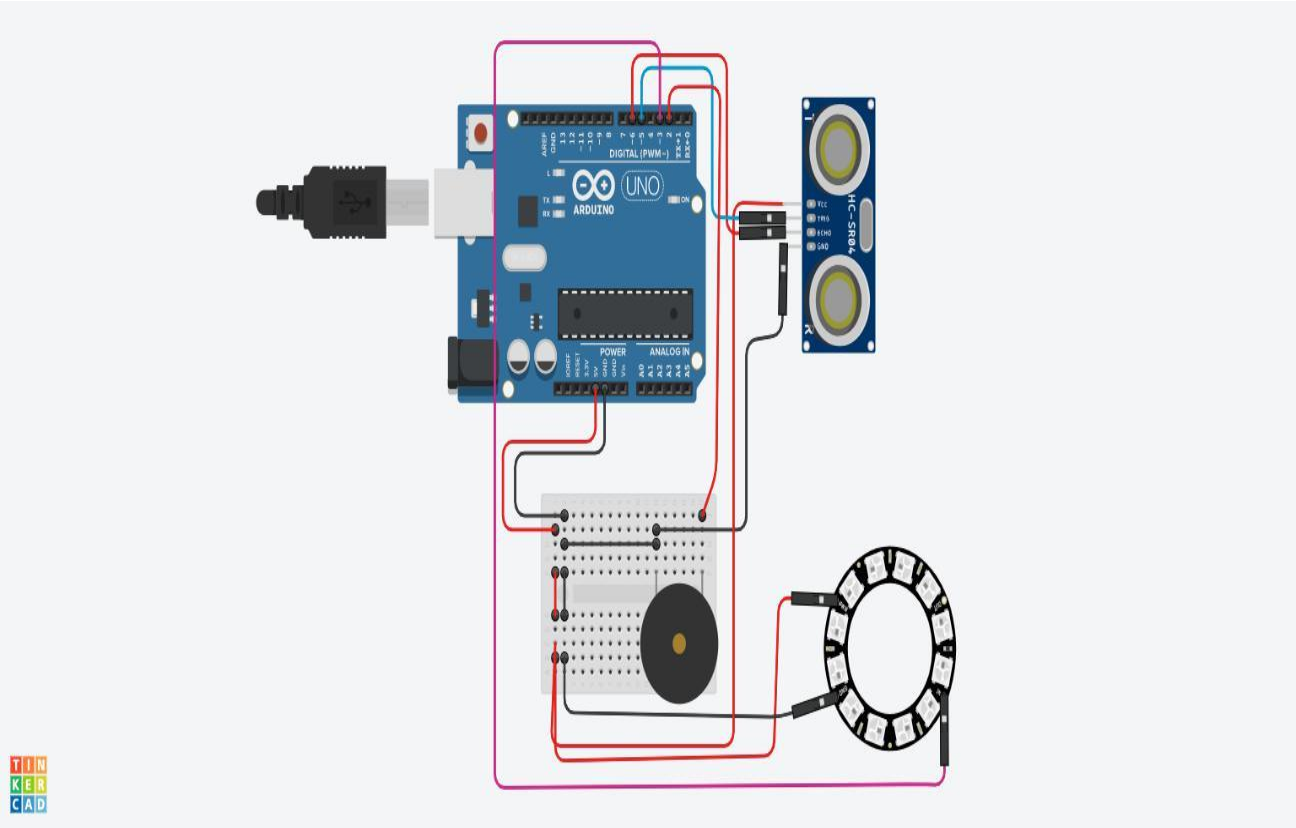


Figure: 1

Conclusion with Future Scopes:

Before assembling the circuit, make sure the source code (social_distancing.ino) is uploaded to the Arduino board using Arduino IDE. As shown in the prototype (Fig. 1), all the four HC-SR04 sensors along with Arduino are placed on the cap in all four directions. To power the circuit, a 5V PP3 battery is connected to the Arduino through its DC jack.

For testing purposes, a safe distance of 100cm is used in the program. It can be changed as per the norms. The serial monitor utility in Arduino IDE can be used to check the distance from all four sensors. You can remove the cable after verifying the working of each sensor on the serial monitor.

In order to reduce the bulkiness of the circuit, you can use smaller ultrasonic sensor modules and replace the Arduino Uno with an Arduino Nano board. It can be developed futures and it has a large uses potential in this pandemic situation.

References:

https://www.tinkercad.com/things/j959KmoZs5d-social-distance-alert-system/editel?sharecode=Kp0-j0jlvs6mA2P8EhbU7Z2I1osEe9acRLJUGmfwvgo