SMART ALARM SYSTEM

20CSL62 INTERNET OF THINGS AND CLOUD LABORATORY

A MINI PROJECT REPORT

Submitted by

RAMYA SHRI T
20CSR166

RITHIKA B
20CSR170

SANJAY B
20CSR183

in partial fulfillment of the requirements
for the award of the degree
of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Estd : 1984

KONGU ENGINEERING COLLEGE
(Autonomous)
PERUNDURAI ERODE – 638 060

ABSTRACT

In our home or in the parking areas while we are parking the vehicle there may be a chance of hitting into the walls or on any other things and also during night times there is a high possibility for accidents. To prevent this, an alarm system had been developed which helps people to know how close they are to the walls or anyother things. In this application the distance is calculated using ultrasonic sensor and it is stored in the thingspeak for analysis. Based on the distance value it will give alerts.

COMPONENTS REQUIRED

- **Ultrasonic Sensor** – a device that uses waves to detect the presence and proximity of objects.

- **Raspberry Pi** – microcontroller used for open-source hardware projects.

- **Resistor** – limits or regulates the flow of electric current in an electronic circuit.

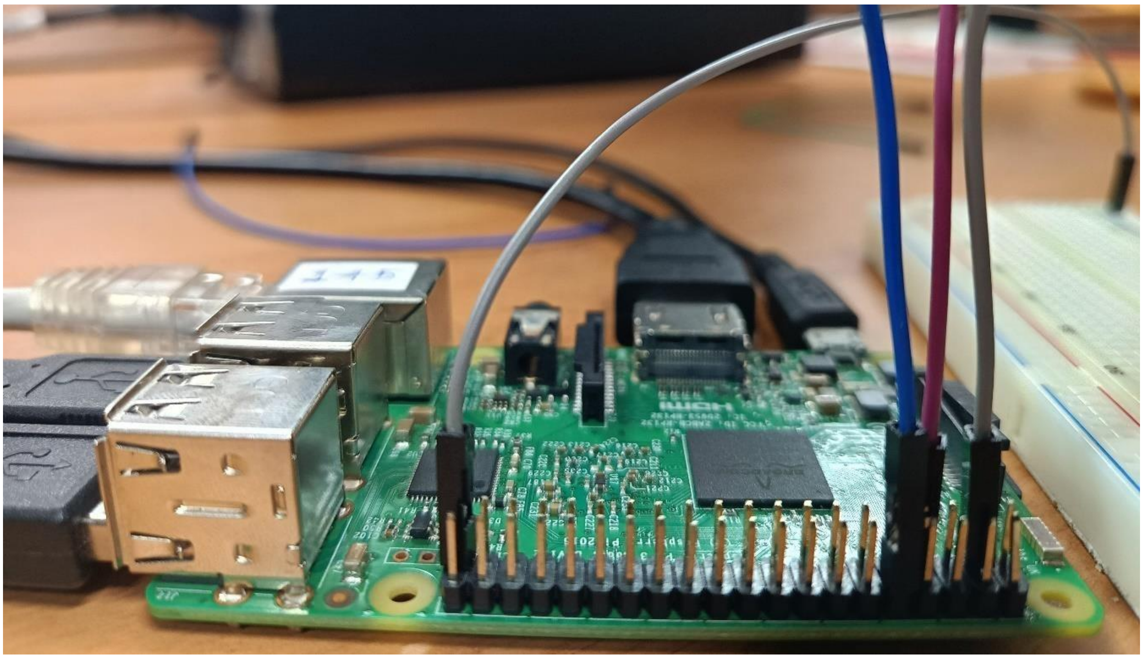
- **Jumper wires** (female to female and male to female) – to connect two points in an electrical circuit.

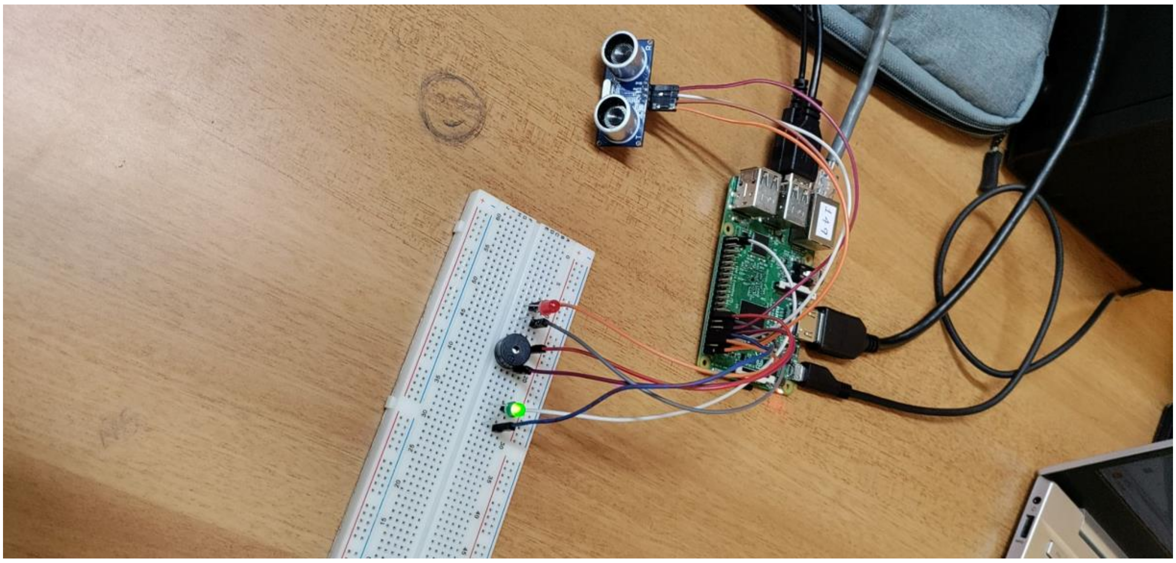
- **Light Emitting Diode (LED)** – semiconductor device, which can emit light when an electrical current passes through it.

- **Breadboard** – to form simple electrical connections among different components.

- **Buzzer -** electric sounding device that generate sounds.

HARDWARE SETUP





CODING

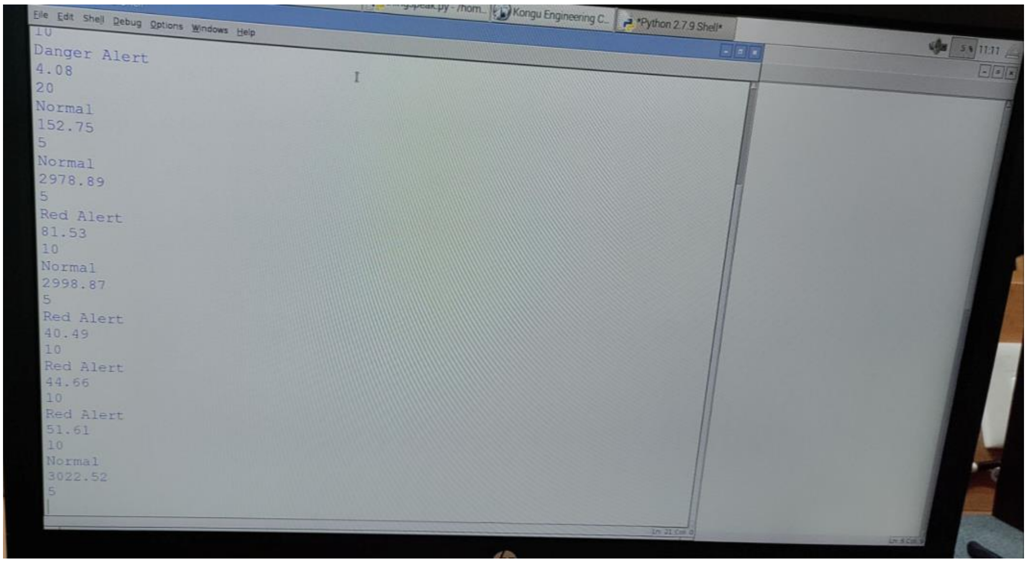
```
import time
import urllib2
import RPi.GPIO as GPIO
trig = 2
echo = 3
led1 = 17
led2 = 14
buzzer = 4
myapi="CNHNEA1BMIZA5O0W"
baseurl = "https://api.thingspeak.com/update?api_key=%s"%myapi
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(trig,GPIO.OUT)
GPIO.setup(echo,GPIO.IN)
GPIO.setup(buzzer,GPIO.OUT)
GPIO.setup(led1,GPIO.OUT)
GPIO.setup(led2,GPIO.OUT)
def read_distance():
    alert =0
    GPIO.output(trig,True)
    time.sleep(1)
    GPIO.output(trig,False)
    pulse_st=time.time()
    while GPIO.input(echo)==0:
```

```
    pulse_st= time.time()
pulse_end=time.time()
while GPIO.input(echo)==1:
    pulse_end=time.time()
pulse_durat = pulse_end - pulse_st
distance = pulse_durat*17150
distance = round(distance,2)

if(distance <10):
    alert=20
    GPIO.output(buzzer,True)
    GPIO.output(led2,False)
    GPIO.output(led1,False)
    print("Danger Alert")

elif(distance>10 and distance <100):
    alert=10
    GPIO.output(buzzer,False)
    print("Red Alert")
    GPIO.output(led1,True)
    GPIO.output(led2,False)

else:
    alert=5
    GPIO.output(buzzer,False)
```

```
        GPIO.output(led2,True)
        GPIO.output(led1,False)
        print("Normal")

    return distance,alert
while True:
    distance,alert = read_distance()

    print(distance)
    print(alert)
    conn = urllib2.urlopen(baseurl+'&field1=%f' %(distance)
+'&field2=%f' %(alert) )
    conn.close()

    time.sleep(0.5)
```

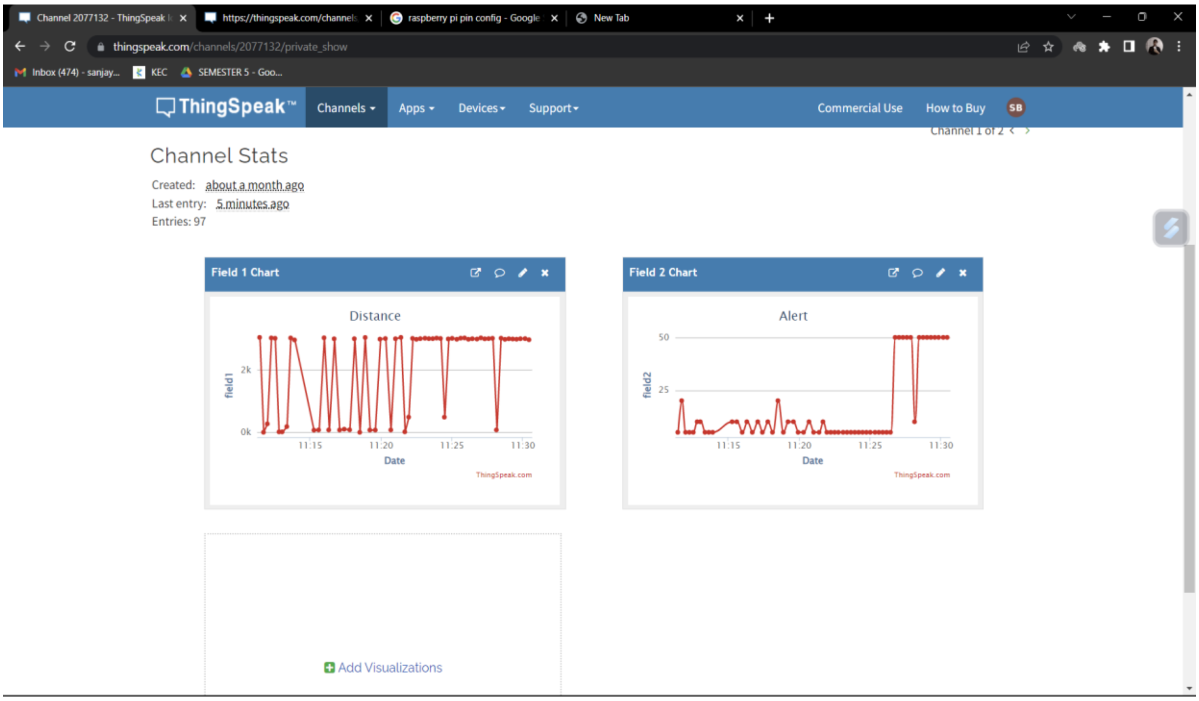
OUTPUT

TERMINAL :



```
10
Danger Alert
4.08
20
Normal
152.75
5
Normal
2978.89
5
Red Alert
81.53
10
Normal
2998.87
5
Red Alert
40.49
10
Red Alert
44.66
10
Red Alert
51.61
10
Normal
3022.52
5
```

THINGSPEAK :



Channel Stats

Created: about a month ago

Last entry: 5 minutes ago

Entries: 97

CONCLUSION

Overall, this IOT-based Smart Alarm System would help people in parking areas to help their vehicles to be parked in the respective slot without any damage to the vehicles. It also help us to avoid accidents and to detect thieves during night times.