Automation of the room door control with a non-contact Temperature Device

Ahmed Garad Wafa Alhaaj

Abdalraheem A. Almagrahi , Nouralddeen A. Amyn , Moaad Ali Albousefi

ABSTRACT

Following recent events, the world has changed, and people are beginning to reconsider their involvement in crowded areas. High body temperature is one of the symptoms of fever and more recently Covid-19. An Automatic Temperature Device is a non-contact device used to check the body temperature of any intruder because it would be difficult and even dangerous to check everyone's temperature when in contact with the person. The device's primary function is to measure the user's body temperature by using an infrared sensor and a PIR sensor to detect the entry. If the temperature is higher than (37.5 degrees Celsius) the door will not open, and a buzzer will sound to notify that a symptom has been detected. The door's opening and closing mechanisms are controlled by a servo motor. Furthermore, the LCDs each person's temperature as well as the number of people permitted to enter.

Key Words: Covid-19, non-contact device, PIR sensor.

1- INTRODUCTION

Trying to keep track of people's health is a worldwide issue nowadays. Health conditions which have been affected by environmental and medical factors are essential for life's comfort. Using technology and sensor advancement the size of electronic components is reducing day by day, so embedded systems are used for many different uses. The use of such an application of embedded systems in temperature scanning automation, door opening, etc.

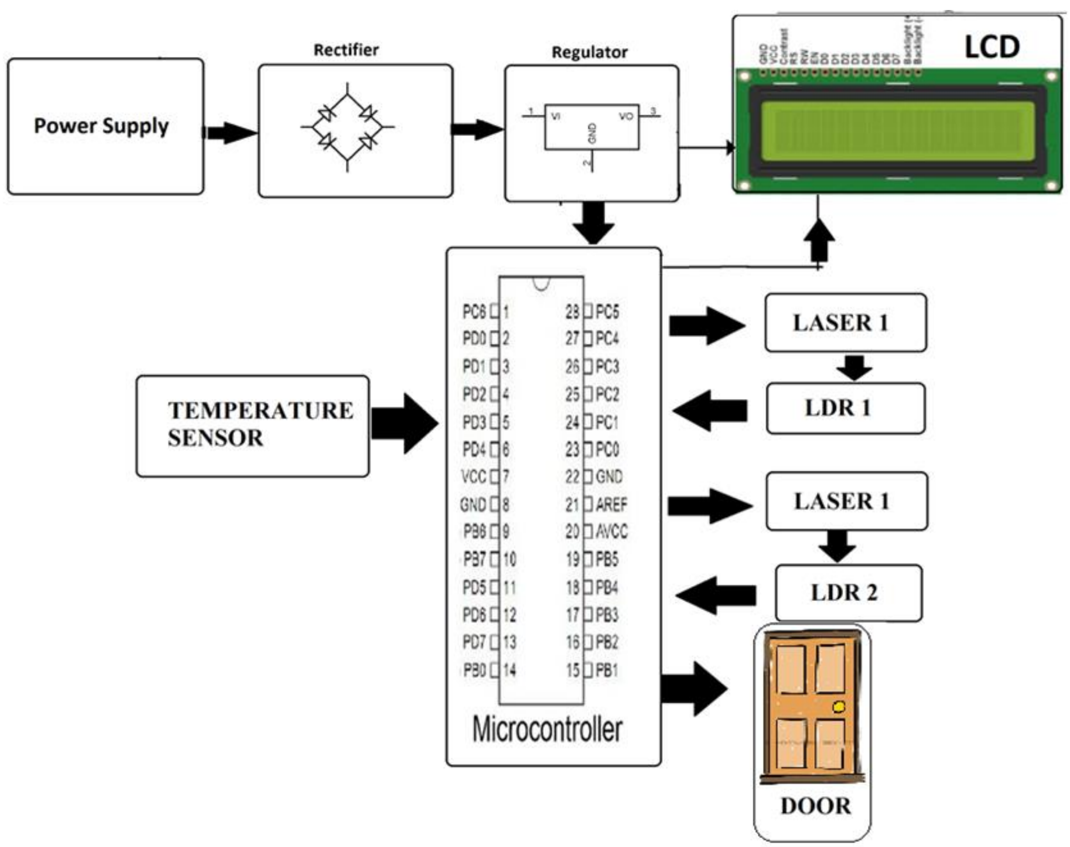
People are now anticipating the automation of all necessary simple procedures, since the last three years every human being knows about the pandemic "COVID-19" (1) that has threatened the world and has had a significant impact on practically every element of human life worldwide. Fever, exhaustion, sore throat, loss of (scent/taste), and congestion in the nose are typical signs of coronavirus infection. Most of the time, it is transmitted directly (individually) via respiratory droplets, but also indirectly via surfaces consequently, using face masks and Sanitizers has demonstrated efficacy in preventing illness spread limiting, these procedures are now carried out manually so that staff can interact with one another while sanitizing and checking the temperature may not be correct.

A very accurate body temperature monitor enables the early identification of possible infected individuals preventing the spread of the pandemic. The project introduces a method and body temperature detection primarily with the capability of measuring temperature without contact. We try to energize a space when such safeguards are used. the usage of a PIR sensor to identify the human entry. When the detector detects entrance, it will check the temperature of the person using a contactless temperature sensor. If the temperature is within the predetermined temperature limit, admission is permitted; otherwise, entry is prohibited. The normal body temperature ranges from 36.5°C to 37 5°C. The status of health below is stated as hypothermia and the status above is referred to as fever and hyperthermia conditions. Using a noncontact temperature measurement device will help to reduce the risk of spreading Covid-19 infection. So that many persons can be assessed separately at points of entry, this non-contact device can quickly measure and show temperature readings. Without having their temperature scanned, no one will be permitted to enter the room. This is especially helpful for identifying fevers, which indicate potential COVID-19 infections and other illnesses.

RELATED RESEARCH AND PROBLEM IDENTIFICATION

METHADOLOGY

System Block Diagram



Hardware Design*:*

1. Arduino Board
2. LDR Sensor 12MM
3. Laser
4. Bluetooth HC-05
5. 16x2 LCD Display
6. Motor-45 RPM
7. Temperature Sensor
8. A Tmega 328p Microcontroller
9. Buzzer
10. Adapter 12V 1AMP
11. Breadboard

: Materials of the Device

Category	**Quantity**	**Value**
Modules	1	2 pin relimate
Capacitors	1 2 3	470uf 10uf 22pf
Resistors	1 3 2	330 10k 3k3
Integrated Circuits	1 1 1 1	Atmega328 7805 LM1117 L293D
Diodes	1 1	Led R-LED
Miscellaneous	1 1 1 1 1 1	Female strip Bridge Buzzer Bluetooth DC POWER JACK Serial

	1	LDR_1
	1	LDR_2
	1	Lazer_1
	1	Lazer_2
	1	Infrared
	1	Push button
	1	10k pot
	1	16mhz

HC-05 Pinout Configuration

Pin Number	Pin Name	Description
1	Enable / Key	This pin is used to toggle between Data Mode (set low) and AT command mode (set high). By default, it is in Data mode
2	Vcc	Powers the module. Connect to +5V Supply voltage
3	Ground	Ground pin of module, connect to system ground.
4	TX – Transmitter	Transmits Serial Data. Everything received via Bluetooth will be given out by this pin as serial data.
5	RX – Receiver	Receive Serial Data. Every serial data given to this pin will be broadcasted via Bluetooth

6	State	The state pin is connected to on board LED, it can be used as feedback to check if Bluetooth is working properly.
7	LED	Indicates the status of Module • Blink once in 2 sec: Module has entered Command Mode • Repeated Blinking: Waiting for connection in Data Mode • Blink twice in 1 sec: Connection successful in Data Mode
8	Button	Used to control the Key/Enable pin to toggle between Data and command Mode

Software Design (Programming Code)

```
#include <SoftwareSerial.h>
#include <Wire.h>
#include <Adafruit_MLX90614.h>
#include <LiquidCrystal.h>
Adafruit_MLX90614 mlx =
Adafruit_MLX90614();
//String c = "";
#define led 5
#define buzzer 4
#define LDR_1 A0
#define LDR_2 A1
#define MAX_NO 10
char temp[10];
char cont[10];
float temperature = 0;
int counter = 0;
int counter_limit = 9;
int prev_counter = 0;
float default_temperature = 100.5;
const int motor_1 = A2;
const int motor_2 = A3;
11); ,10 ,9 ,8 ,7 ,LiquidCrystal lcd(6
tx ,3); //Bl RX ,SoftwareSerial bluetooth(2
const int bluetooth_state_pin = 0;
void setup()
{
bluetooth.begin(9600);
2); ,lcd.begin(16
Serial.begin(9600);
mlx.begin();
// mlx.writeEmissivityReg(64650);
// Serial.println(mlx.readEmissivity());
 delay(1000);
INPUT); ,pinMode(bluetooth_state_pin
INPUT); ,pinMode(LDR_1
INPUT); ,pinMode(LDR_2
OUTPUT); ,pinMode(led
OUTPUT); ,pinMode(buzzer
OUTPUT); ,pinMode(motor_1
OUTPUT); ,pinMode(motor_2
lcd.clear();
0); ,lcd.setCursor(3
lcd.print(F("COVID Safe"));
1); ,lcd.setCursor(0
lcd.print(F(" ROOM"));
delay(2000);
```

```
HIGH); ,digitalWrite(buzzer
while (!digitalRead(bluetooth_state_pin))
 {
 lcd.clear();
 lcd.print(" Pair");
1); , lcd.setCursor(0
 lcd.print(" Bluetooth");
 delay(1000);
 }
lcd.clear();
lcd.print(" Bluetooth");
1); ,lcd.setCursor(0
lcd.print(" Paired");
delay(2000);
Serial.print("Hello"); //
}
void loop()
{
val1 = analogRead(LDR_1); int
val2 = analogRead(LDR_2); int
temperature = mlx.readAmbientTempF(); //
lcd.clear();
0); ,lcd.setCursor(0
;)" :lcd.print("Person
lcd.print(counter);
1); ,lcd.setCursor(0
;)" :lcd.print("Temp
1); ,lcd.setCursor(7 //
lcd.print(temperature); //
lcd.print(" F"); //
if (counter != prev_counter)
 {
 prev_counter = counter;
 + "* String data_send = "3
;"#" + )String(prev_counter
 bluetooth.print(data_send);
 bluetooth.flush();
 Serial.println("Send data to bluetooth");
 }
delay(500);
if (val1 < 800)
 {
LOW); , digitalWrite(buzzer
 delay(500);
HIGH); , digitalWrite(buzzer
 temperature = mlx.readObjectTempF();
1); , lcd.setCursor(7
 lcd.print(temperature);
 lcd.print(" F");
```

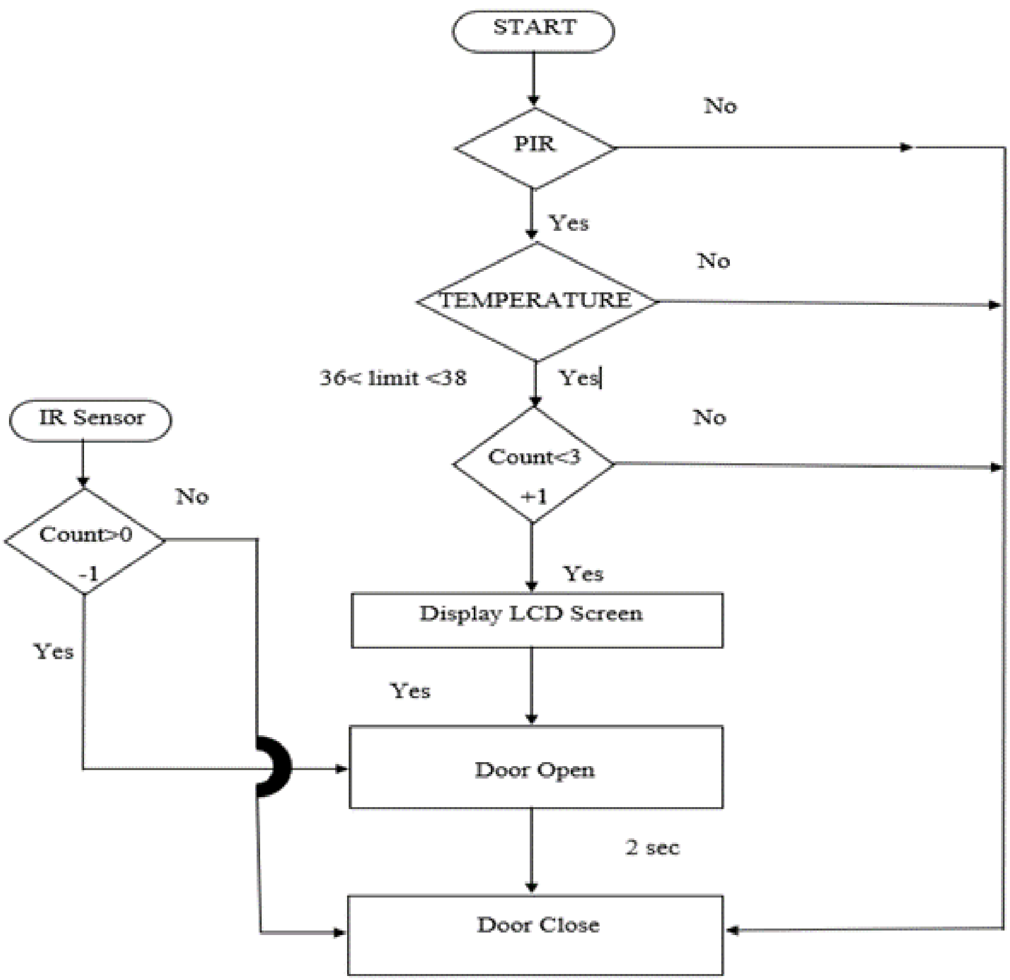
```
if (temperature < default_temperature)
 {
 if (counter >= counter_limit)
 {
 counter = counter_limit;
 Serial.print("Too many people inside");
LOW); , digitalWrite(buzzer
 lcd.clear();
 lcd.print("ROOM FULL ..");
 delay(2000);
HIGH); , digitalWrite(buzzer
 }
 else
 {
 door_open();
HIGH); , digitalWrite(led
 while (val2 > 800)
 {
 val2 = analogRead(LDR_2);
 }
1; + counter = counter
 Serial.println("Door Open");
;)" : Serial.print(" Counter
 Serial.println(counter);
 delay(800);
 while (analogRead(LDR_2) < 800);
 door_close();
LOW); , digitalWrite(led
 }
 }
 else
 {
 Serial.println("Temperature is High");
 lcd.clear();
 lcd.print(" HIGH");
1); , lcd.setCursor(0
TEMPERATURE"); "(lcd.print
LOW); , digitalWrite(buzzer
HIGH); , digitalWrite(led
 delay(2000);
 while (analogRead(LDR_1) < 800);
LOW); , digitalWrite(led
HIGH); , digitalWrite(buzzer
 }
 }
else if (val2 < 800)
 {
LOW); , digitalWrite(buzzer
 delay(500);
```

```
HIGH); , digitalWrite(buzzer
 if (counter == 0)
 {
 counter = 0;
 Serial.print("No member inside");
 delay(500);
 }
 else
 {
 door_open();
HIGH); , digitalWrite(led
 while (val1 > 800)
 {
 val1 = analogRead(LDR_1);
 }
 Serial.println("Door Open");
1; - counter = counter
;)" : Serial.print("counter
 Serial.println(counter);
;)" : Serial.print("val2
 Serial.println(val2);
 delay(800);
 }
 while (analogRead(LDR_1) < 800);
 door_close();
LOW); , digitalWrite(led
 }
if (bluetooth.available() > 0)
 {
 String data = bluetooth.readString();
;)" : Serial.print("Data
 Serial.println(data);
 if (data.indexOf('1') >= 0)
 {
 Serial.println("Temperature");
 char data_arr[25];
 + )(data.length , data.toCharArray(data_arr
1);
+ " : Serial.println("data_char
String(data_arr));
100#*1 //
temp);& ,"#]#^[%*1" , sscanf(data_arr
String(temp)); + " : Serial.println("Temp
 default_temperature = atof(temp);
 lcd.clear();
 lcd.print("Temperature ");
1); , lcd.setCursor(0
+ " : lcd.print("Limit
String(default_temperature));
 delay(2000);
```

```
data = "";
 bluetooth.flush();
 Serial.flush();
 }
 else if (data.indexOf('2') >= 0)
 {
 Serial.println("count");
 char data_arr[4];
*/)(data.length , data.toCharArray(data_arr
;)/*1 +
+ " : Serial.println("data_char
String(data_arr));
cont);& ,"#]#^[%*2" , sscanf(data_arr
 // counter_limit = atoi(cont);
 if (atoi(cont) > counter)
 {
 counter_limit = atoi(cont);
+ ": Serial.println("cont
String(counter_limit));
 lcd.clear();
 lcd.print("Person ");
1); , lcd.setCursor(0
String(counter_limit)); + " : lcd.print("Limit
 delay(2000);
 data = "";
 bluetooth.flush();
 Serial.flush();
 }
 else
 {
 lcd.clear();
 lcd.print("INVALID COUNT ..");
 delay(2000);
 }
 }
 else
 {
 bluetooth.flush();
 }
 }
}
void door_open()
{
HIGH); ,digitalWrite(motor_1
LOW); ,digitalWrite(motor_2
delay(550);
LOW); ,digitalWrite(motor_1
LOW); ,digitalWrite(motor_2
delay(900); //
```

```
}
void door_close()
{
LOW); ,digitalWrite(motor_1
HIGH); ,digitalWrite(motor_2
delay(550);
LOW); ,digitalWrite(motor_1
LOW); ,digitalWrite(motor_2
}
```

Figure a1 shows the controller program flow chart



Fig, a1 program flow chart for the system controller

System Collecting and Testing

After collecting and connecting all the parts that talked about in this research paper with programming the system in a certain way it has led to the making of a smaller sample of the project in the figure a below

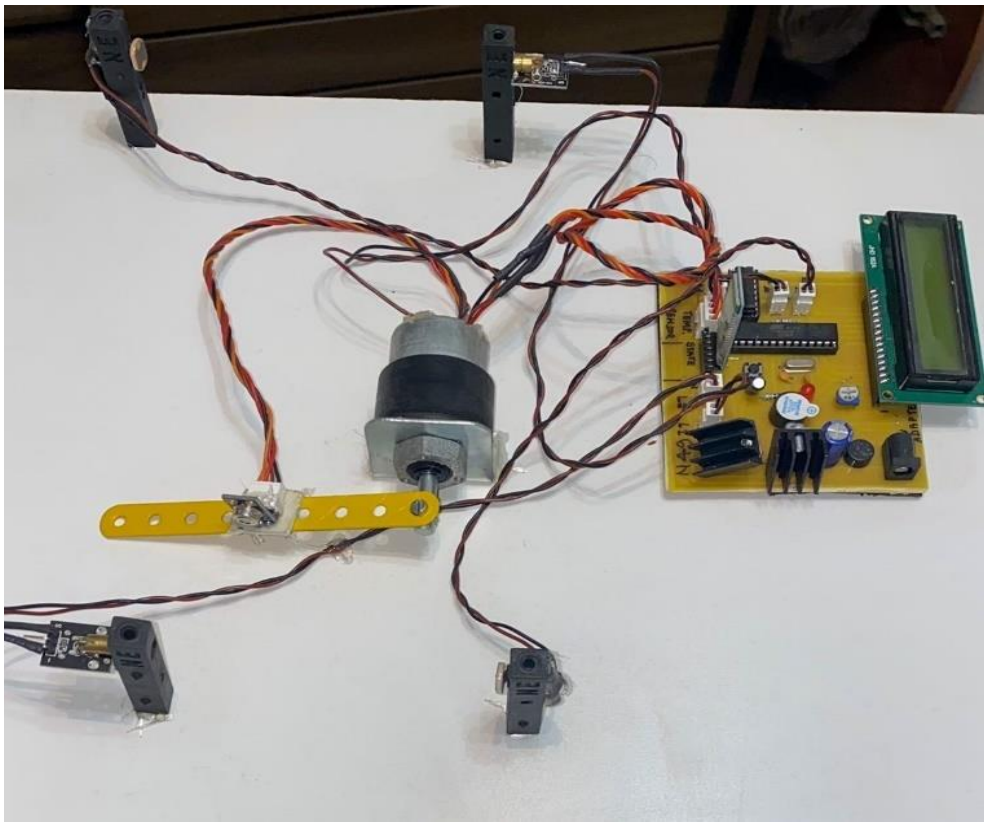


Figure a2 - Version of Automatic Temperature Device

System Testing

Phase 1: The device has been plugged in and turned on.

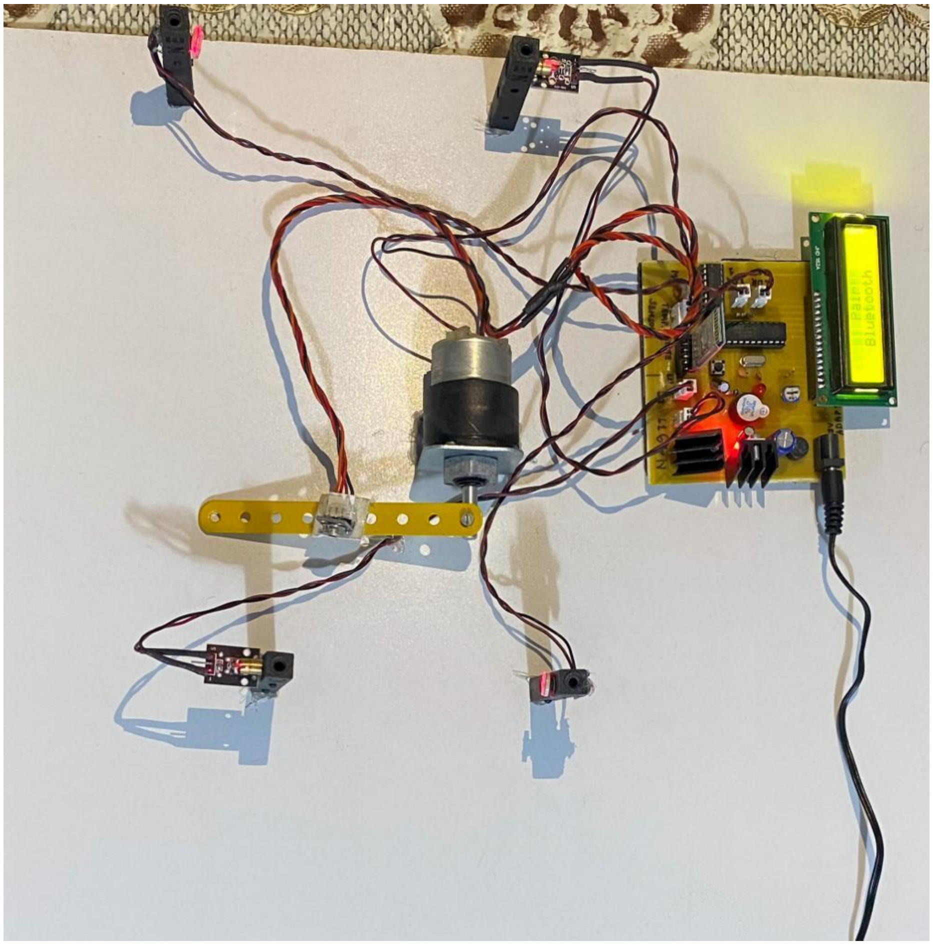


Figure b - Plugging and turning on the Device

Phase 2: Connecting the device to the smart phone via Bluetooth using the Monitor Temperature-persons app.

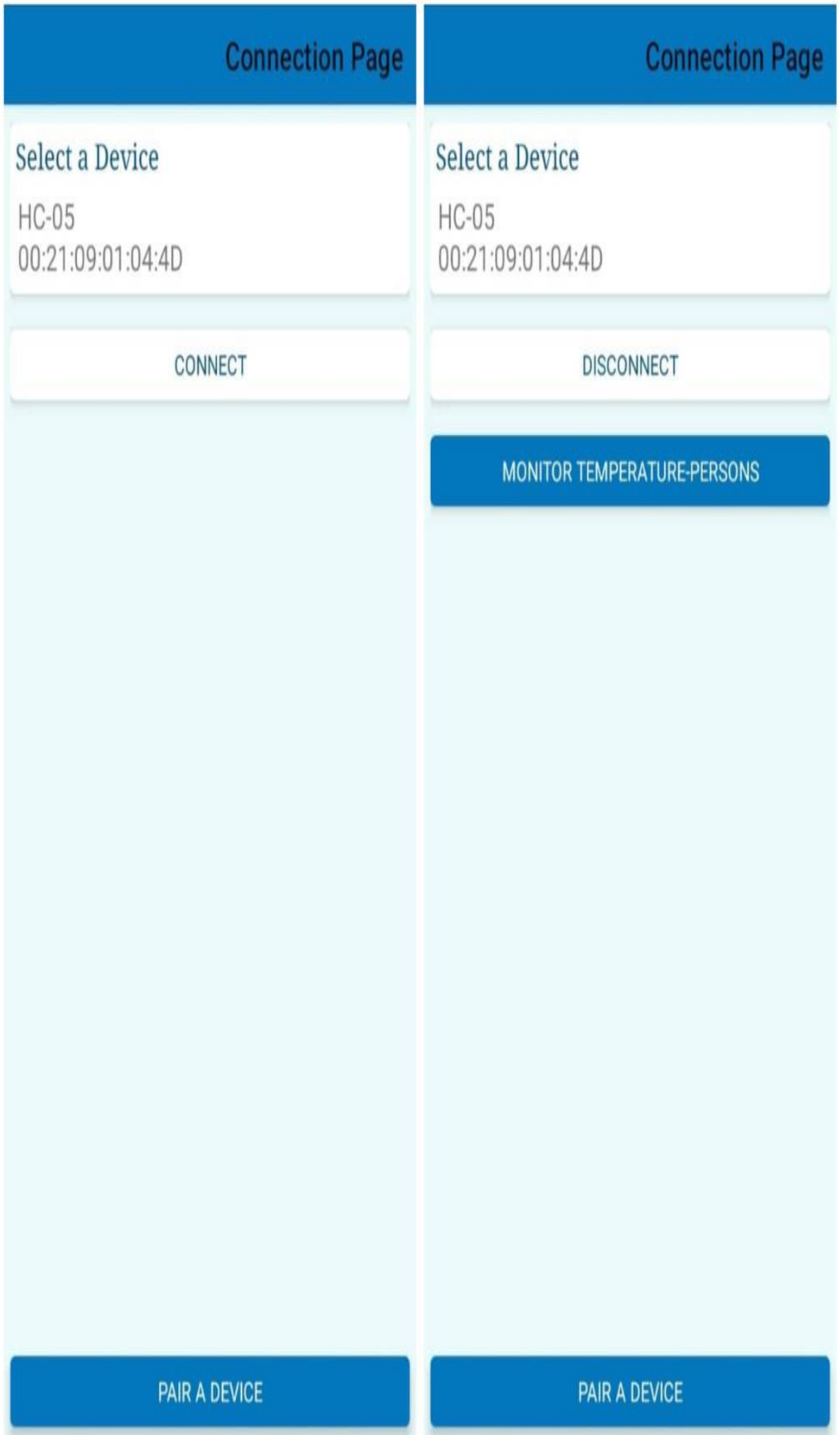


Figure c- Connecting the device by using Monitor Temperature-person's app

Phase 3: After connecting, set the temperature and the number of people who are permitted to enter.

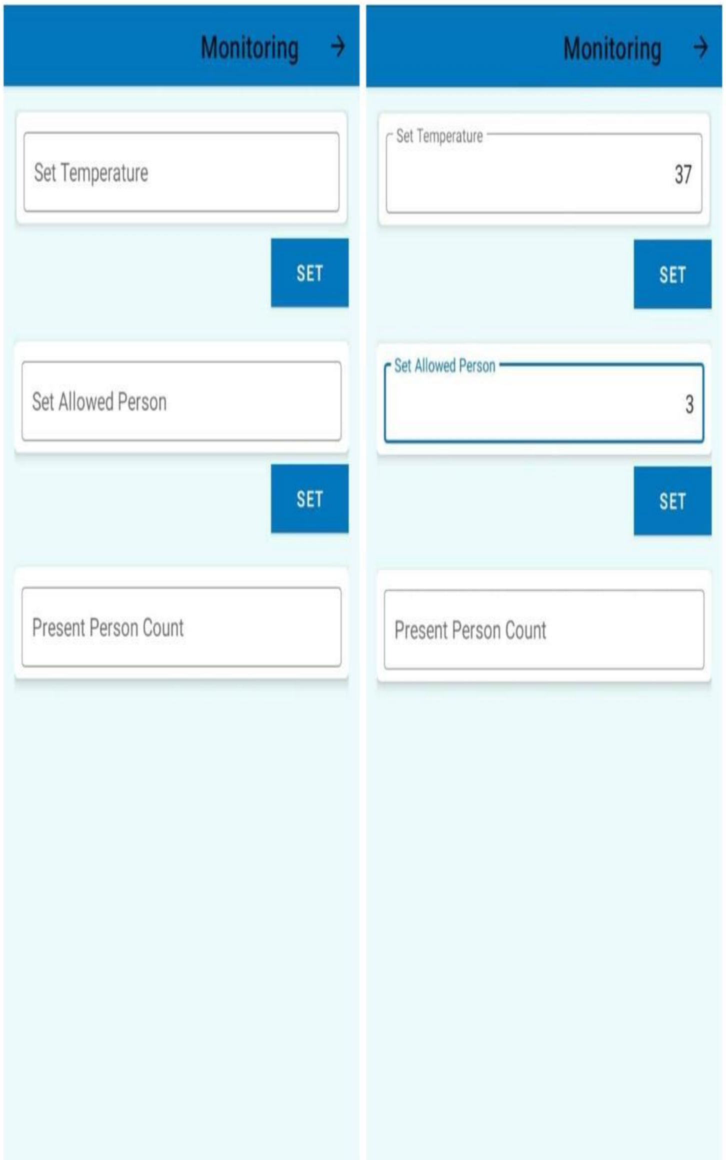


Figure d - setting the temperature and the number of people

Phase 4: To test the accuracy of sensors the first person is represented by using a hand.

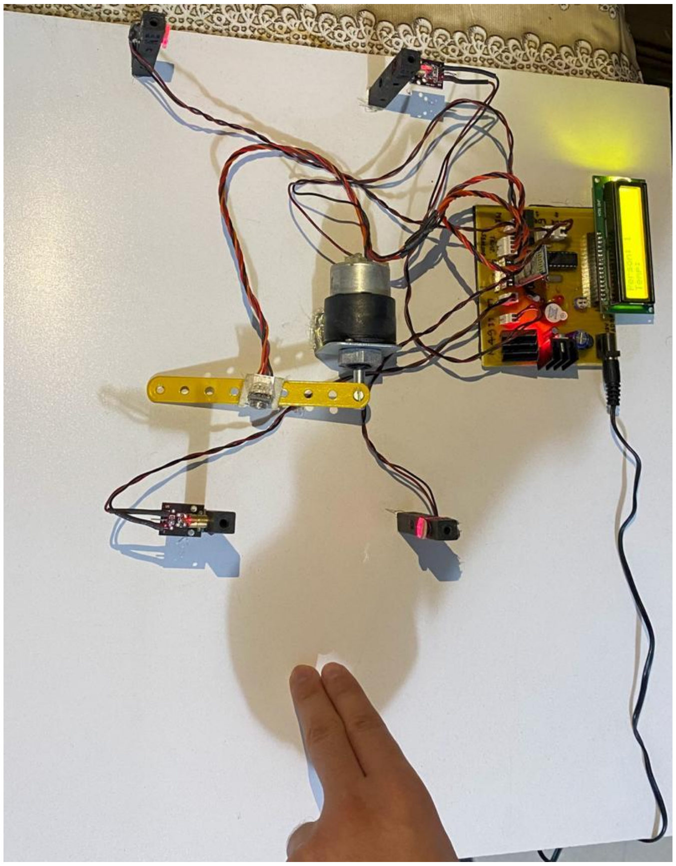


Figure e – Using the hand as a first person

Phase 5: Putting one's hand in front of the LDR2 and the temperature sensor.

Phase 6: The results will be displayed on the screen; if the temperature is lower than the set temperature, the door will open; if the temperature is higher than the set temperature, an alert sound will be activated, with the message “HIGH TEMPERATURE” displayed on the screen as shown in the figure g.

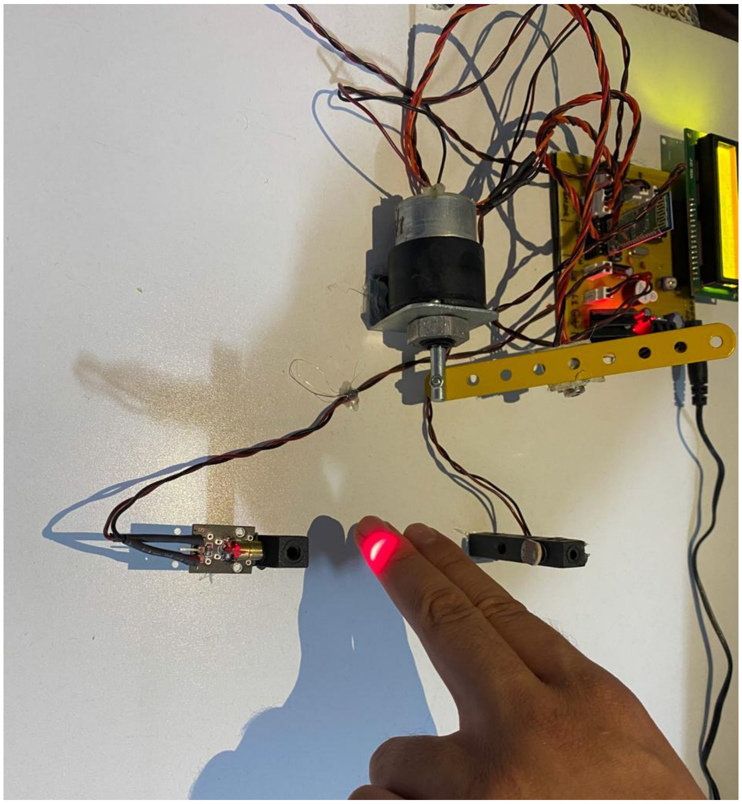


Figure f - placing the hand in front of LDR2 and Temperature senso

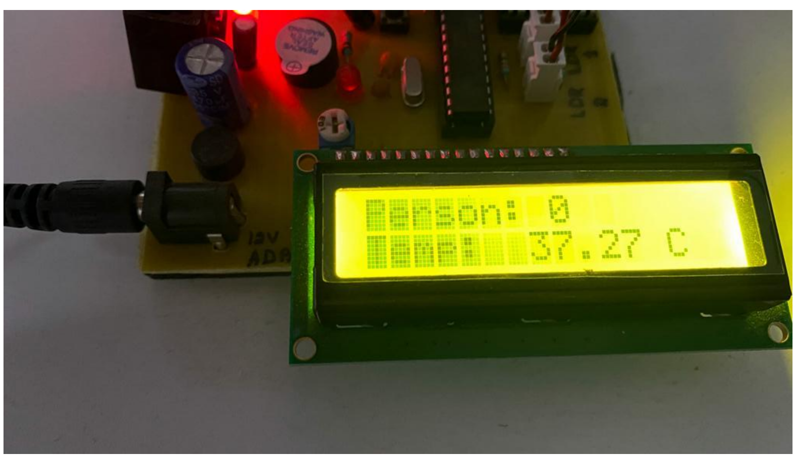


Figure g - first-person temperature measurement

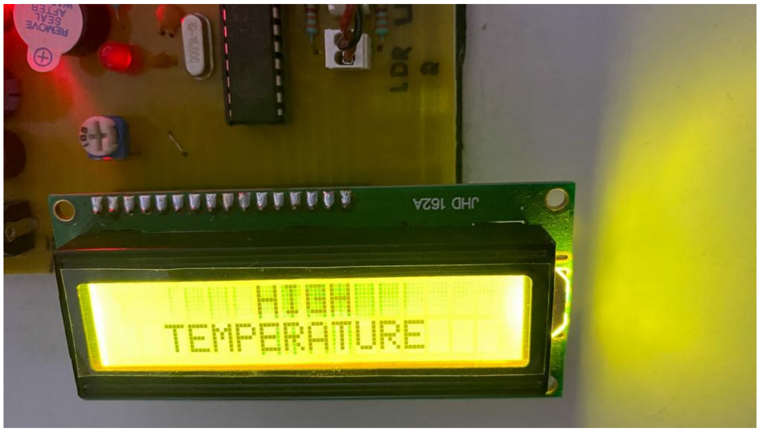


Figure h - Results of detecting a High Temperature person on the screen

Phase 7: After opening the door, move your hand in front of the LDR1 to detect the entrance, the door will then be closed, and the number of people on the screen will change as shown in figure k

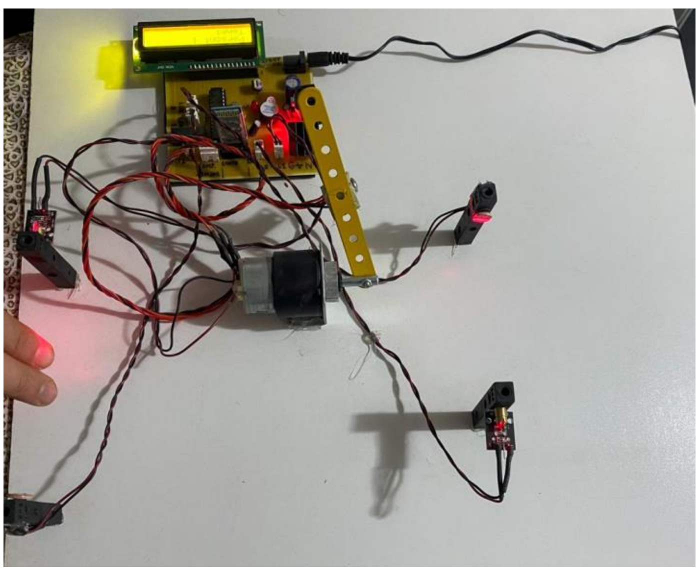


Fig. K moving the hand in front of the LDR 1

Fig.l The firist personhas been detectedby the motion sensoe

Phase 8: After a certain number of people have entered the room, the next person will be denied entry and the screen will display “ROOM FULL” as shown in the figure (m).

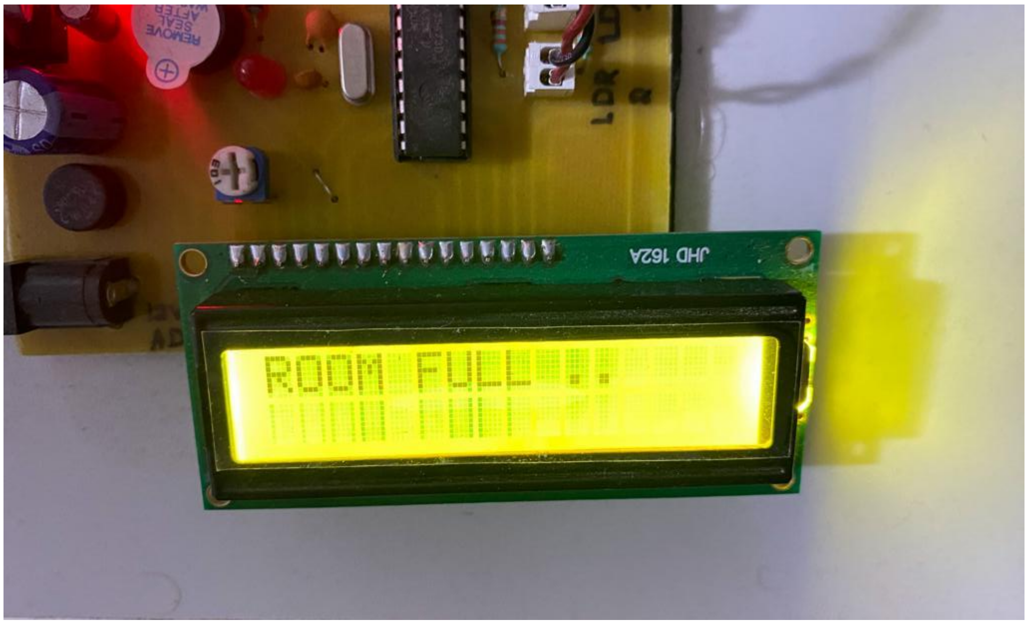


Fig. m - showing a message about a full room on the screen

Phase 9: When the specified number of people has been reached, the only way to open the door is to reduce the number of people by exiting the room.

Fig. n - Exit sign

Discussion

This paper presents the development of an automatic temperature device based on an infrared digital temperature sensor coupled with Arduino UNO. The automatic temperature device successfully recorded accurate measurement results. The automatic temperature device is permissible and feasible to be employed by pharmacies and shops. This application could be very well extended to bigger fields including hospitals, factories, schools, banks, and others.

CONCLUSION

After design, implementation, and testing, it can be concluded that the automatic temperature device can limit the spread of viruses among the people in a closed space by measuring the temperature and controlling the number of people who are permitted to enter, which will lead to the detection of fevers and possible Covid-19 infections. Moreover, this device works with efficiency and accuracy when compared to a variety of other tools or devices in the same field.

REFRENCES

1) Singhal. (2020) *A Review of Coronavirus Disease-2019 (COVID-19).* Indian Journal of Pediatrics volume.
2) Badamasi. (2014) *The working principle of an Arduino*. IEEE.
3) Bayle. (2013) *C Programming for Arduino*. Packt Publishing Ltd.
4) Dell'Isola, Cosentini, Canale, Ficco, Dell'Isola. (2021) *Noncontact Body Temperature Measurement: Uncertainty Evaluation and Screening Decision Rule to Prevent the Spread of COVID-19.* MDPI.
5) Abuzairia, Sumantri, Irfana, Mohamad. (2020) *Infrared thermometer on the wall (iThermowall): An open source and 3-D print infrared thermometer for fever screening.* Elsevier Ltd.
6) Rahimoon, Abdullah, Taib. (2020) *Design of a contactless body temperature measurement system using Arduino*. Indonesian Journal of Electrical Engineering and Computer Science.

7) Temperature and Face Mask Detection with Contactless Sanitizer Dispenser Authors: Nihaal Manjunath, Mohammed Shafi, Nagendra Prasad HM , Mohammed Omar, Dr. Geethashree DOI Link: https://doi.org/10.22214/ijraset.2022.45449

8) https://www.microcontrollerslab.com

9) https://www.components101.com

10) https://www.wikipedia.com

11) https://www.watElectronics.com

12) https://techzeero.com

13) https://www.WHO.int

14) https://www.javatpoint.com

15) https://www.Alibaba.com

16) https://www.Farnell.com

17) https://www.circuito.io