CSE461 Project Report

Title: Anti theft and fire alarm system

Group Number: 4

Section: 4

Group Members:

1. Anika Ahmed, ID: 21101029
2. Nafis Chowdhury, ID: 21101034
3. Moinul Haque, ID: 21101186
4. Yeamin Hossain Shihab, ID: 21101174
5. Mirza Fahad Bin Kamal, ID: 20101399

Submitted to:
Shayekh Bin Islam
Shakir Rouf

Submitted on:
27-4-2024

Table of Contents

Topic	Page
Introduction	2
Objective	2-3
Planning and Design	3
Components	3-4
Algorithm	4-8
Experimental Setup	8
Algorithm Explanation	9-14
Feature	14
Schematic diagram	15
Circuit Setup	16
Details of Hardware used	17-21
User Manual	22-28
Challenges	28
Conclusion	29

Introduction:

In our project, we aim to provide efficient security and also a protective system to the user's home. Our project is divided into two parts: one part deals with the security of the house and the other part deals with the fire alarm system. In the anti-theft system, it provides security inside the user's home. Whenever any intruder, or thief enters the user's house, there are various sensors installed within the system that detect the presence of the intruder and report it to the user. In the anti-theft system, we installed a force sensor on the floor so that whenever any intruder enters the house, the buzzer and LEDs will turn ON informing the user of an intruder alert. After a little time, a message will be sent to the user's phone via Bluetooth alerting the user. If the user feels like it is a false alarm then the user can turn off the system via his phone. This will turn off the alarm system like the buzzer and LED however the sensors will still keep on sensing the information and it will again alert the user if there is any intruder. To permanently turn off the system, there is a kill switch that the user can press to turn off the anti-theft system. The user can press the switch again to turn on the anti-theft system. To further increase the security of the system, we also installed PIR sensors. PIR sensors detect changes in infrared radiation caused by the object within its range. So when any intruder moves within the range, the PIR sensor detects it and activates the alarms. The same message will be sent to the user's phone and the user can turn it off via phone or completely deactivate the system via the kill switch. In the second part of our project, we have a fire alarm system. In the fire alarm system, there are sensors like flame sensor and DHT22. The flame sensor detects the presence of fire and DHT22 continuously measures the temperature. If the temperature is above a certain threshold of temperature, then the motor will start to sprinkle water. As the temperature increases, the speed at which the motor sprinkles water will also increase. Both LED and buzzer will also turn ON indicating the presence of fire. Subsequently, the user will receive a notification on their mobile device, alerting them to the presence of a fire.

Objective:

In this project, we aim to provide a smart and autonomous safety and protection system. We designed an efficient intruder detection system that will continuously monitor the environment and report the user via Bluetooth to the user's phone about any sort of home invasion. In this way, the users are continuously informed about the situation of the home. Based on the information, the user can take action in order to save their assets and protect their home from intruders and thieves. Our project also comes with a fire alarm system. The system will continuously monitor for the presence of fire and will send a message to the user via Bluetooth if there is any presence of fire. The system will try to extinguish the fire with its automated water spraying. Based on the message the user can call the fire service and the people can evacuate the area and save their lives. One such incident where our system could be useful is the incident of the 2024 Dhaka Bailey Road fire where several innocents lost their lives. Through these systems, they could have been alerted of the fire and several innocent lives could have been saved. So our objective is to improve the user's quality of life and safety.

Planning and Design

Scope of the Project

- Can be used in home environments or even in offices, schools and even inside cars for better safety.
- The total cost of all components required is quite low compared to commercially available systems, making this project a viable option for home security.
- The power required to drive the system is also quite low compared to other options available as the whole system can be run using its own power source like 5V battery.
- The system will also be easy to use and accessible for everyone as it will have a bluetooth notification system to notify the owner of possible threats.
- Can be further integrated into the main security system, allowing for a better and more accurate system in general.

Components:

- Arduino Uno
- Male to Male jumper wire
- Male-to-female jumper wire
- Female to Female jumper wire
- Slide switch
- Flame sensor
- DHT22
- PIR sensor
- Force sensor
- HC-06 Bluetooth module
- Relay module
- LED

- Buzzer
- Breadboard
- 10K Resistor
- 2.2K Resistor
- DC Motor
- Water pipe

Algorithm:

```
int pirPin = 3; //the digital pin connected to the PIR sensor's output
bool force = false;
bool motion = false;
int output = 4;
int count=0;
char data =0;
bool flag=true;
#include <Wire.h>
#include "DHT.h"
#include<SoftwareSerial.h>
SoftwareSerial BTSerial(10, 11); // RX, TX for HC-05 Bluetooth module

#define DHTPIN 8
#define DHTTYPE DHT22
int flamePin = 7;
int buzzer = 12
;
int red = 12;
int motor = 9;
int swt = 4;
int count1=0;
DHT dht(DHTPIN, DHTTYPE);

void setup(void) {
  Serial.begin(9600);
  pinMode(pirPin, INPUT);
//  digitalWrite(pirPin, LOW);
  pinMode(5, OUTPUT);
  //Bluetooth.begin(9600);
  pinMode(flamePin, INPUT);
  pinMode(swt, INPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(red, OUTPUT);
  Serial.begin(9600);
  pinMode(motor, OUTPUT);
```

```
 // pinMode(10, INPUT);
 // pinMode(11, OUTPUT);
  // Serial.println(F("DHTxx test!"));
  dht.begin();
  }

void loop(void) {
    int sensorValue = analogRead(A0);
    int pir = digitalRead(pirPin);
    int switc = digitalRead(2);
    data=Serial.read();
    // Serial.print(data);
    if (data=='1'){
      flag=false;
    }
    else{
      flag=true;
    }

    if (sensorValue > 50 && force == false){
    force = true;

    }
    if (motion == false && pir == 1){
      motion = true;
    }
    if(switc == 0 and flag){
      Serial.print('entry');
      if(force == true || motion == true){
        if (count>2){
          // Serial.println("Invasion");
          sendNotification("Burglar alert. Press 1 to turn it off");
        }
        count=count+1;
        digitalWrite(5,1);
        delay(1000);
  //      Bluetooth.print("Invasion!");
  //      Bluetooth.print(";");

      }

    }
    else{
        force = false;
```

```
      motion = false;
      count=0;
      digitalWrite(5,0);
      delay( 1000);
//      Bluetooth.print("All good!");
//  Bluetooth.print(";");
    }


float t = dht.readTemperature();
int flame = digitalRead(flamePin);
int swi = digitalRead(swt);
// Serial.println("Flame: ");
// Serial.println(flame);
// Serial.println("Temp: ");
// Serial.println(t);
// Serial.println("Count: ");
// Serial.println(count1);

if (swi == 0 and flag){
  // Serial.print('fire entry');
if (flame == 0){
  if (count1>2){
    // Serial.println("Fire");
    sendNotification("Fire alert. Press 1 to turn it off");
  }
  //float t = dht.readTemperature();
  if (t>26.50){
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(1500);


  }

  else if (t>26){
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(1000);


  }else{
```

```
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(500);
  }
//  digitalWrite(motor, LOW);
//  delay(1000);
  digitalWrite(motor, HIGH);

  delay(1000);
  count1=count1+1;
}
else{
  analogWrite(buzzer, LOW);
  digitalWrite(red, LOW);
  digitalWrite(motor, HIGH);
  delay(1000);
  count1=0;


}
}
else{
  analogWrite(buzzer, LOW);
  digitalWrite(red, LOW);
  digitalWrite(motor, HIGH);
  delay(1000);
  count1=0;

}
  // Serial.println("Force Sensor: ");
  // Serial.println(sensorValue);
  // Serial.println("Force flag: ");
  // Serial.println(force);
  // Serial.println("-----------");
  // Serial.println("PIR: ");
  // Serial.println(pir);
  // Serial.println("PIR flag: ");
  // Serial.println(motion);
  // Serial.println("-----------");
  // Serial.println("Burglar switch: ");
  // Serial.println(switc);
  delay(1000);
```

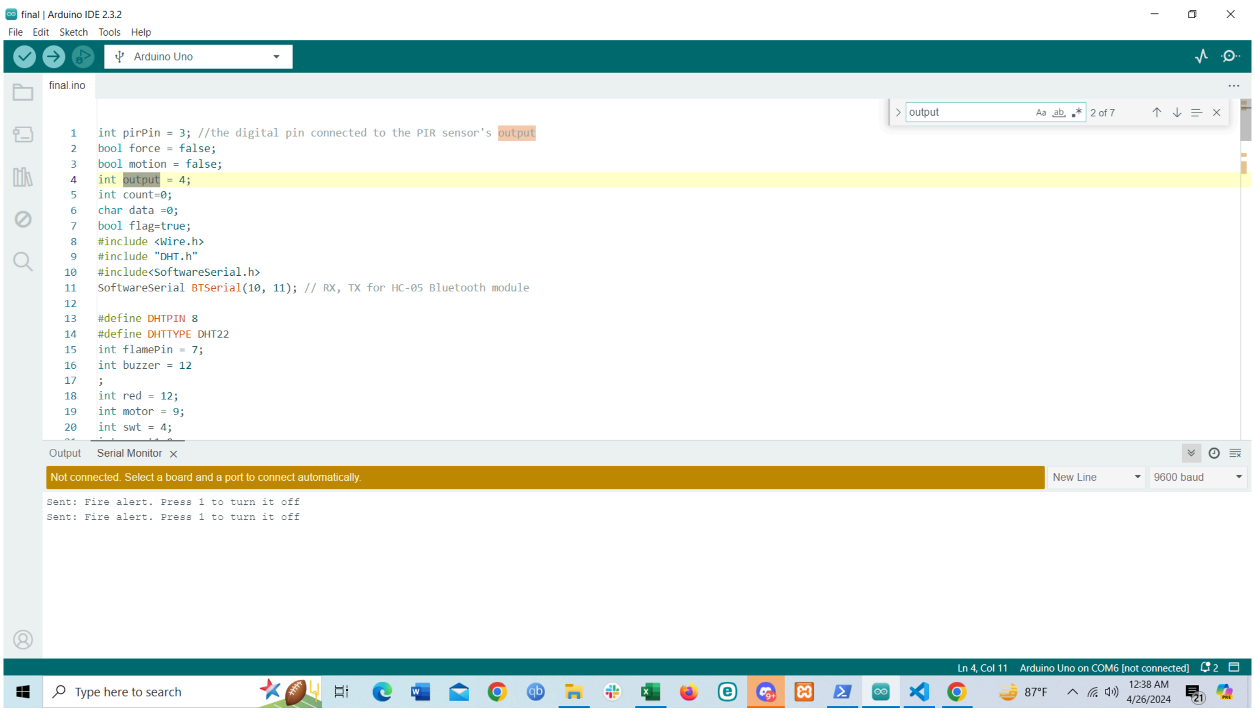
```
}

void sendNotification(String message) {
  BTSerial.println(message); // Send message via Bluetooth
  Serial.println("Sent: " + message);
  delay(1000); // Delay to avoid flooding the Bluetooth serial
}
```

Experimental Setup:

The PIR sensor has GND, VCC, and output. The 5V pin from the Arduino is connected to the positive terminal of the breadboard and the GND pin of the Arduino is connected to the negative terminal of the breadboard. The output pin of the PIR sensor is connected to the 3rd slot of the digital pin. In this project, we used two switches. One of the switches is connected to the 4th slot of the digital pin and the other switch is connected to the 2nd slot of the digital pin. DHT22 consists of GND, VCC, and an output pin. The output pin from DHT22 is connected to the 8th slot of the digital pin in the Arduino board, the VCC is connected to the positive terminal of the breadboard and the GND pin is connected to the negative terminal of the breadboard. Similarly flame sensor also has GND, VCC, and an output pin. The output pin from flame sensor is connected to the 7th slot of the digital pin in the Arduino board, the VCC is connected to the positive terminal of the breadboard and the GND pin is connected to the negative terminal of the breadboard. Furthermore, two buzzers are used in this project. One of the buzzer is connected to the 12th slot of the arduino board and the other buzzer is connected to the 5th slot of the Arduino board. The LED is connected to the 12th slot of the arduino board in series with the 2.2K ohm resistor so that the LED does not catch fire. Force sensor is an analog sensor so the pin to detect pressure is connected to the analog pin A0 of the arduino board in series with a 10K ohm resistor in series which is then connected to the ground and the VCC of the force sensor is connected to the positive terminal of the breadboard. The HC-06 bluetooth module has VCC pin which is connected to the positive terminal of breadboard and GND pin which is connected to the negative terminal of breadboard. The receive pin, RX of the HC-06 is connected to the transmit pin TX of Arduino board. The receive pin, RX of the Arduino board is connected to the transmit pin TX of HC-06. Relay has got Vcc and GND pin connected to positive and negative terminal respectively, and the relay pin is connected to Arduino pin no 9. Relay's another point has got another positive terminal connected to it for amplifying the motor voltage, and another point is connec ted to motor's positive terminal and motor's negative terminal is connected to ground.

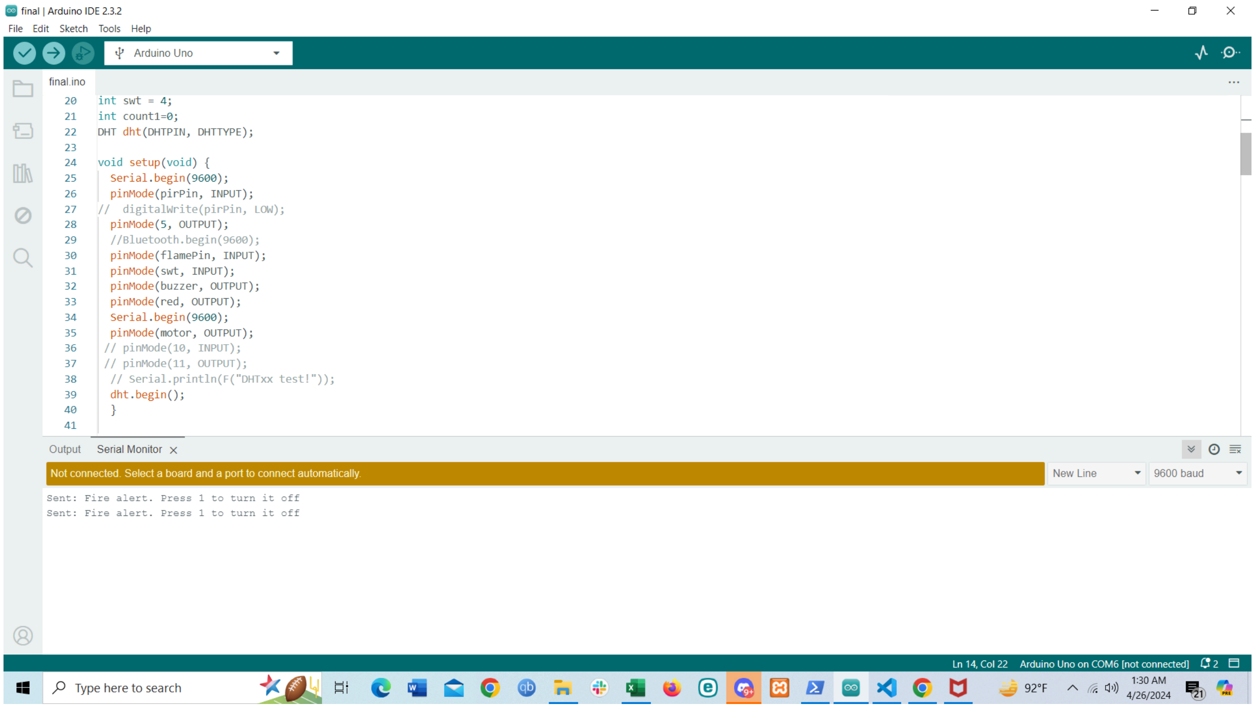
Algorithm Explanation:

1. Variable declaration and initializing pin number:



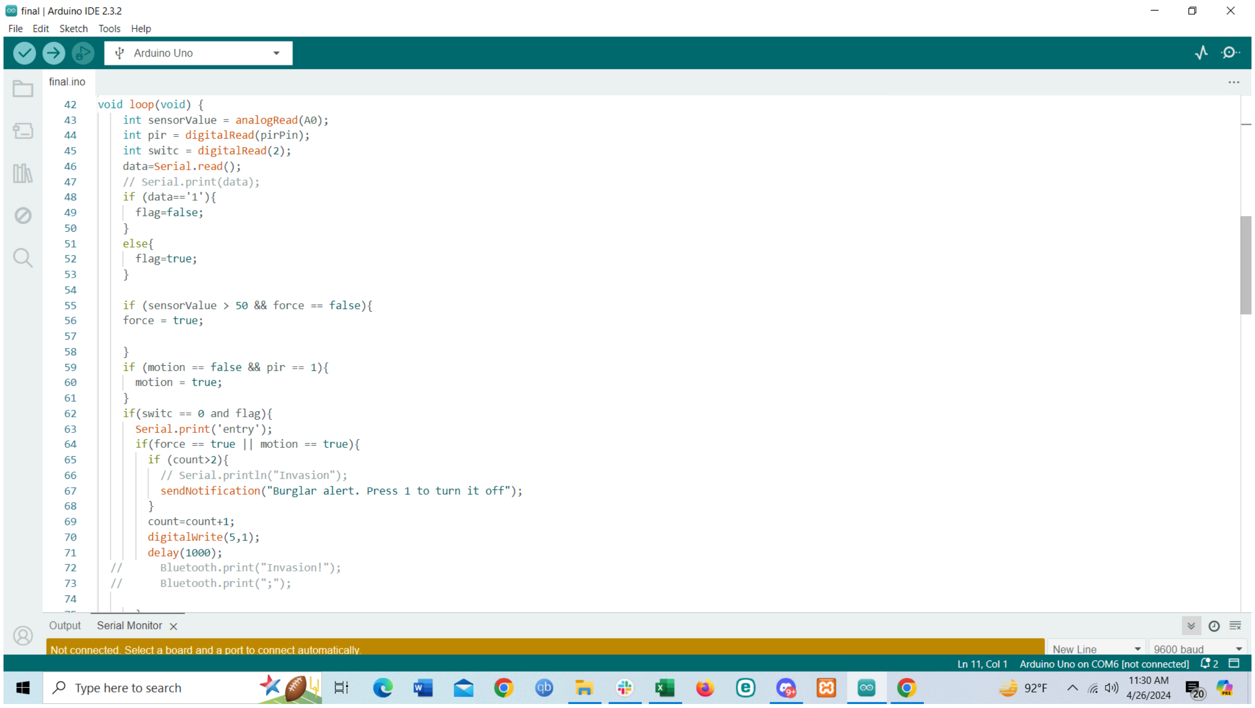
In the above screenshot, we have assigned pin numbers to some variables. We have used three boolean variables here force,motion and flag. The force variable changes to True when the pressure changes in the force sensor and the motion variable changes to True if there is a difference in infrared radiation. In order to use DHT22, at first we had to download the library for DHT22. By “`#include "DHT.h"` we are including the library for DHT22. In this lines ‘#define DHTPIN 8’ and ‘#define DHTTYPE DHT22’ we are defining the digital pin number and the type of DHT we are using. ‘DHT dht(DHTPIN, DHTTYPE);’ in this line in the dht parameter we are sending both the DHTPIN and DHTTYPE. In order to access the HC-06 bluetooth module, we have to give ‘#include<SoftwareSerial.h>’. We defined our bluetooth module as ‘SoftwareSerial BTSerial’.

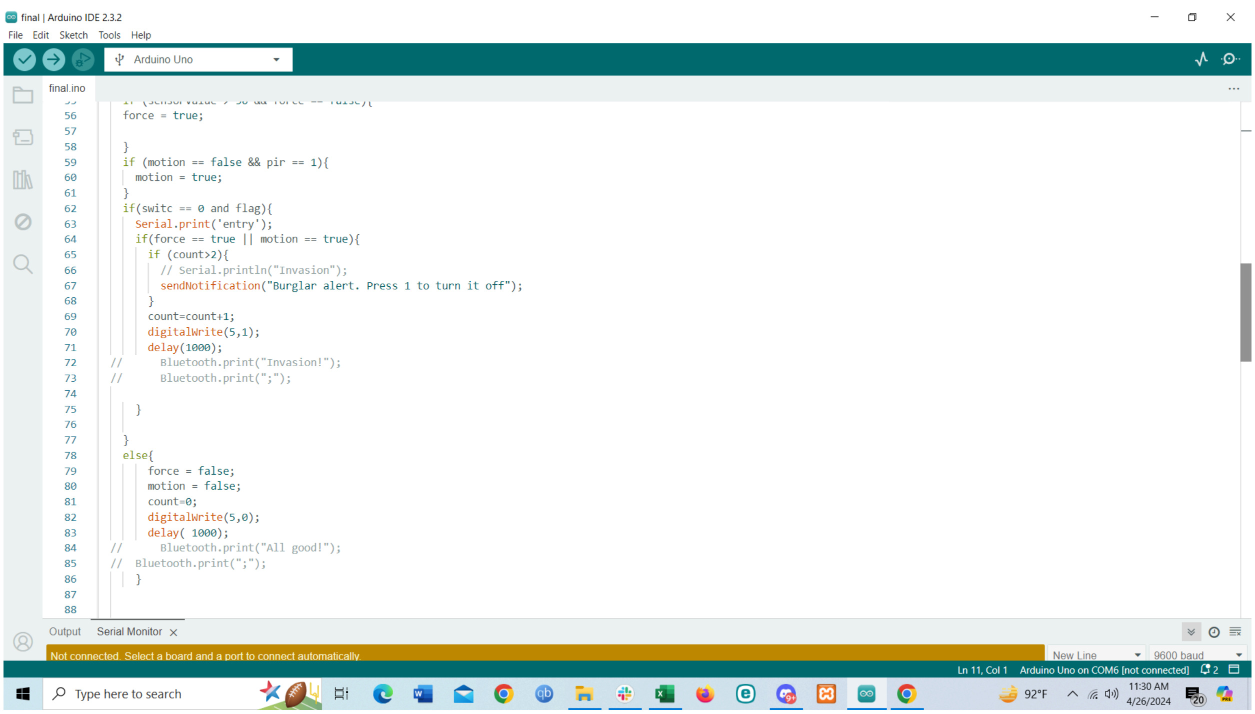
2. Void Setup



In the void setup, we beagan the serial monitor and set the baud rate to 9600. We defined the pinMode of PIR sensor to input, motor to output, flamepin to input, switch to input, buzzer to output and led to output. To activate DHT22 we have to write ‘dht.begin()’.

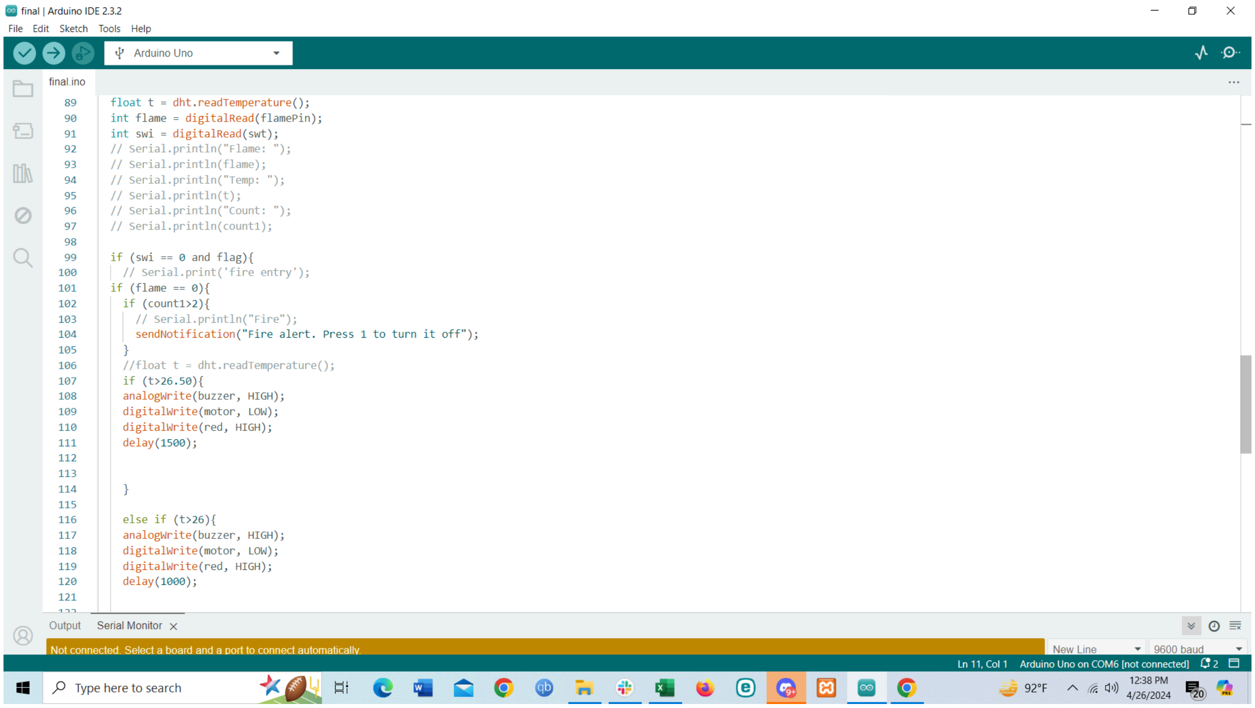
3. Anti-theft system:



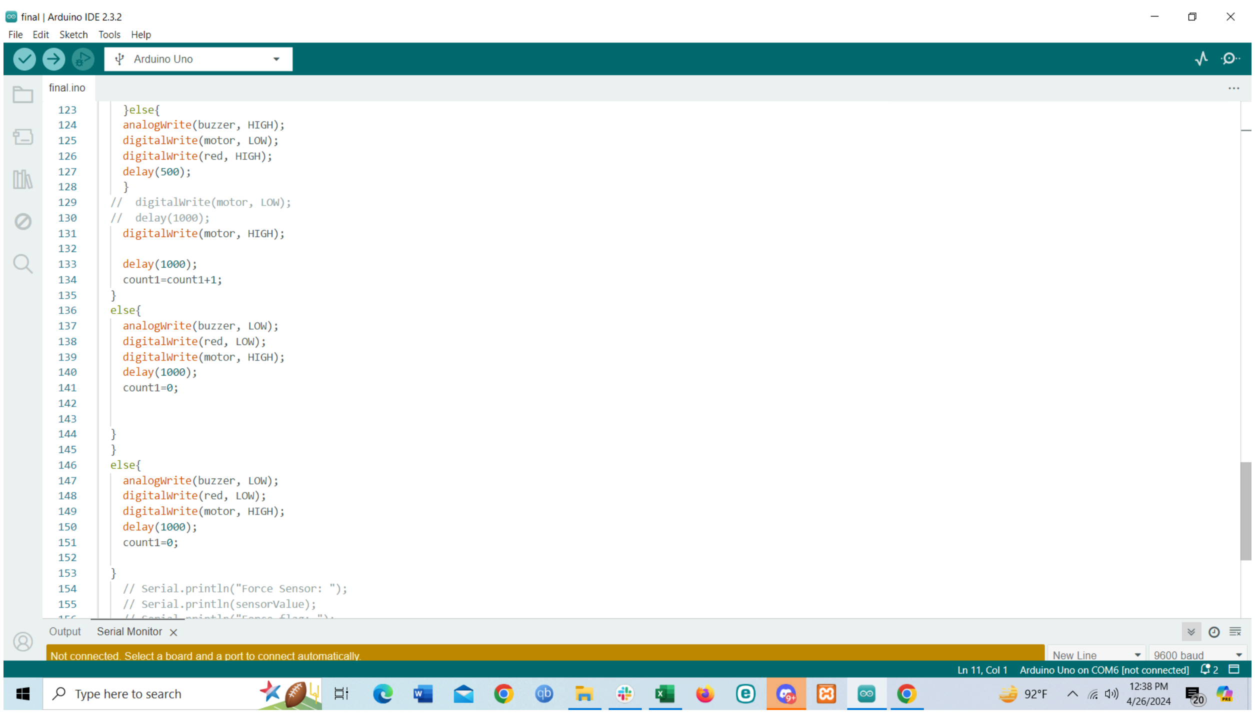


The data from the force sensor is continuously read from A0 pin and stored in sensorValue. The data from the PIR sensor is digital and stored in the pir variable. When there is no movement the value from the PIR sensor is 0, but when there is movement the value from the PIR sensor is 1. Similarly the data form the switch is stored in switc variable. If the switch is not pressed then the value is 1, and when the switch is pressed the value is 0. Serial.read() continuously reads any data that the user sent through his phone. If the user send 1 via his phone, flag variable is set to False else it will always remain True. If the force from the force sensor is greater than 50 while the force variable was False then the force variable will be changed to True. Similarly if the pir variable from the PIR sensor reads 1 when there is movement and while the motion variable remains False then the motion variable is changed to True. If the switch is equal to 0 and the flag variable remains True then it refers that the system is active. Within the system, it checks whether the force or motion variable is True: If the condition does not satisfy then both force and motion variable is set to False, buzzer is set to low and there is a count variable for time interval since invasion which is also set to zero. However, if the condition satisfies buzzer is set to High and count variable is incremented by 1. If the condition continues to satisfy and if count variable is greater than 2 then a message will be sent to user's phone via a function known as sendNotification which takes the message as a parameter. If the user sends 1 via his phone, then the Flag will be set to False and for the system to run both switch needs to be 0 and flag variable needs to be True. The user can also press the switch to turn off the system.

4. **Fire alarm:**



```
float t = dht.readTemperature();
int flame = digitalRead(flamePin);
int swi = digitalRead(swt);
// Serial.println("Flame: ");
// Serial.println(flame);
// Serial.println("Temp: ");
// Serial.println(t);
// Serial.println("Count: ");
// Serial.println(count1);

if (swi == 0 and flag){
  // Serial.print('fire entry');
if (flame == 0){
  if (count1>2){
    // Serial.println("Fire");
    sendNotification("Fire alert. Press 1 to turn it off");
  }
  //float t = dht.readTemperature();
  if (t>26.50){
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(1500);


  }

  else if (t>26){
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(1000);

```



```
  }else{
  analogWrite(buzzer, HIGH);
  digitalWrite(motor, LOW);
  digitalWrite(red, HIGH);
  delay(500);
  }
//  digitalWrite(motor, LOW);
//  delay(1000);
  digitalWrite(motor, HIGH);

  delay(1000);
  count1=count1+1;
}
else{
  analogWrite(buzzer, LOW);
  digitalWrite(red, LOW);
  digitalWrite(motor, HIGH);
  delay(1000);
  count1=0;


}
}
else{
  analogWrite(buzzer, LOW);
  digitalWrite(red, LOW);
  digitalWrite(motor, HIGH);
  delay(1000);
  count1=0;

}
  // Serial.println("Force Sensor: ");
  // Serial.println(sensorValue);
```

”

The temperature from DHT22 is stored in variable t. The data from the flame pin is stored in flame variable. If there is no flame then the flame pin reads 1 and if there is flame then it read 0. The swi variable reads the value from the switch. Similar to Anti-theft, if the switch value is equal to 0 and fag variable is True then the fire alarm system will start. Inside the system, if the flameis detected then another check is made to check if the temperature is greater than 26.50

then buzzer will be set to high, motor is turned ON by setting it to low and the LED is set to high and the delay is set to 1.5s. If the temperature is greater than 26 and less than 26.50 then buzzer will be set to high, motor is turned ON by setting it to low and the LED is set to high and the delay is set to 1s. Furthermore if the temperature is below 26, then buzzer will be set to high, motor is turned ON by setting it to low and the LED is set to high and the delay is set to 0.5s. Also count1 variable is also incremented by 1 and if the count1 variable is greater than 2 then a message will be sent to user's phone via a function known as sendNotification which takes the message as a parameter. If the user sends 1 via his phone, then the Flag will be set to False and for the system to run both switch needs to be 0 and flag variable needs to be True. The user can also press the switch to turn off the system.

5. sendNotification function:

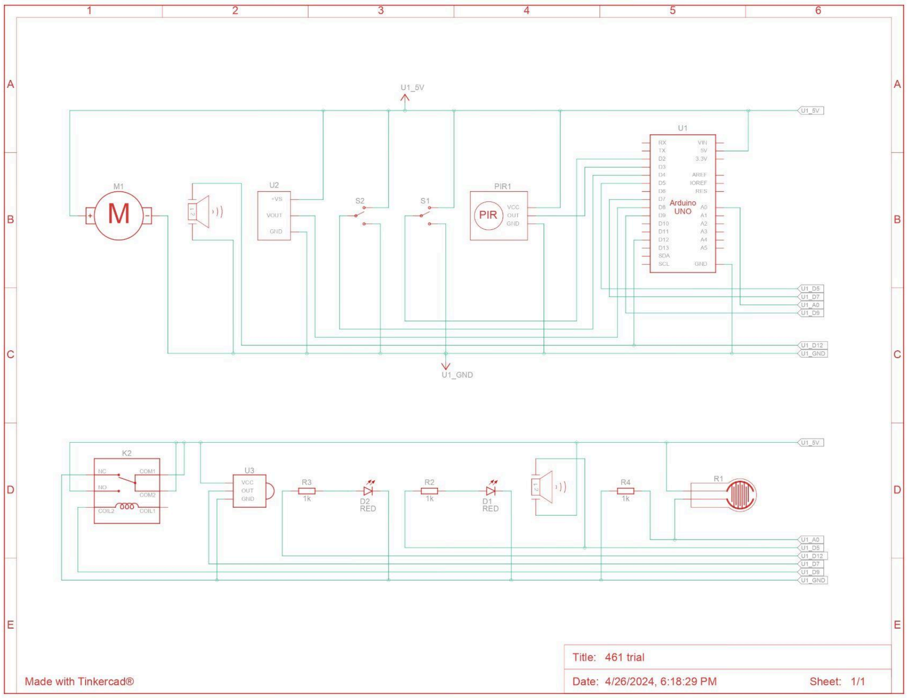
```
void sendNotification(String message) {
  BTSerial.println(message); // Send message via Bluetooth
  Serial.println("Sent: " + message);
  delay(1000); // Delay to avoid flooding the Bluetooth serial
}
```

In this function, the message is sent to user's phone. The message is sent as a parameter and defined as message in the function. "BTSerial.println(message)" then sents the message to user's phone.

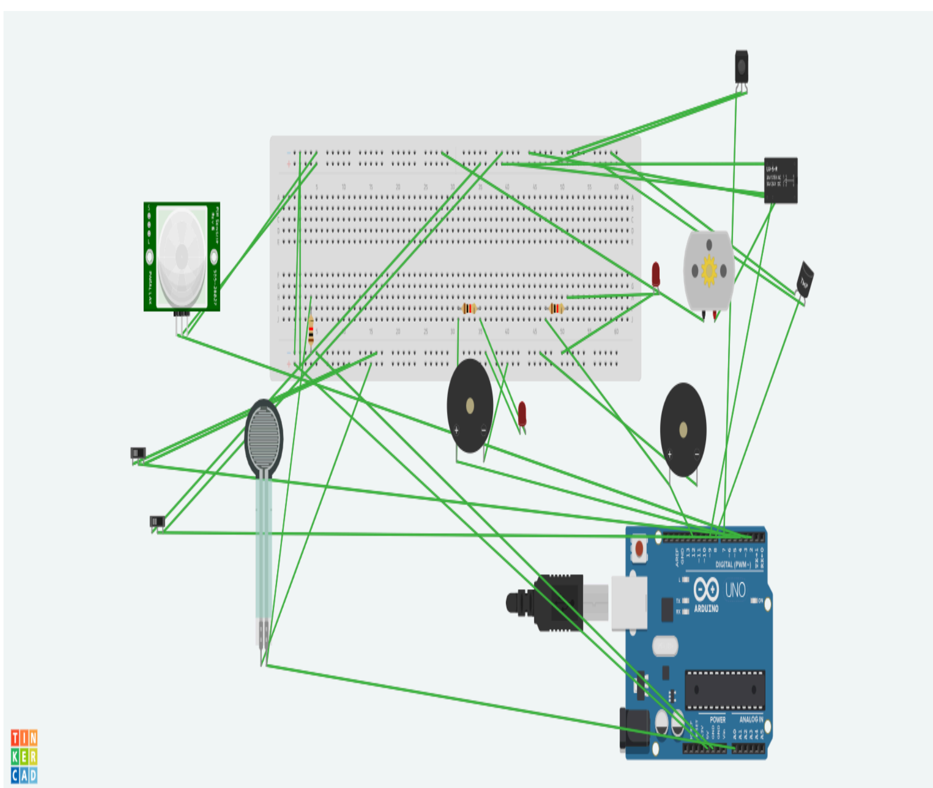
Features:

1. Detect fire inside the house.
2. Spray water based on the temperature level based on the fire area.
3. Detect any attempt made to steal or tamper with objects within the house.
4. Activate the buzzer and LED to notify the residents of danger.
5. Also notify the owner through Bluetooth.
6. Owner can deactivate the system by pressing 1.
7. Have a kill switch that deactivates the system in case of false alarm.

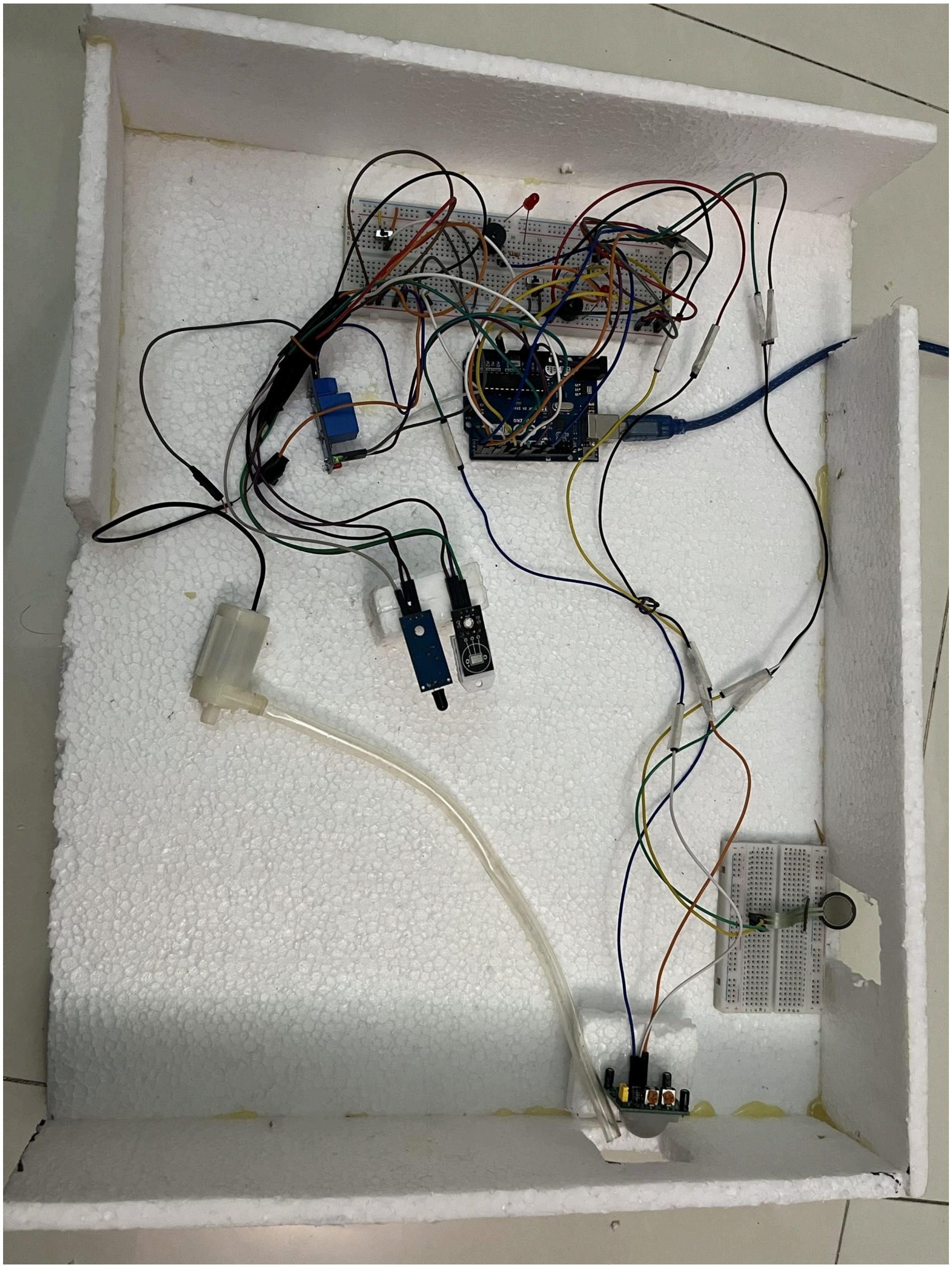
Schematic Diagram:



‘



Circuit Setup:



Video Link:
https://youtu.be/xwbY7XyuG4k?si=_wSH6p63rUs-yxvN

Details of hardware used:

In this segment, we divide details of the hardware into whether they read analog or digital value, pin modulation, types of data they read, how data is transmitted to system and how they work for every hardware.

1. **Switch:**

    Analog or Digital: Switches are typically digital devices, providing a binary output signal (on/off) indicating the state of the switch.

    Pin Modulation: Switches are simple devices that don't involve pin modulation. They have two states: open (off) and closed (on).

    Types of Data: Switches read the state of a circuit, indicating whether it's open or closed.

    Data Transfer to System: The state of the switch is typically read by a microcontroller or other digital input device, usually connected to the switch through a GPIO (General Purpose Input/Output) pin.

    How They Work: Switches work by physically opening or closing a circuit. When the switch is pressed or toggled, it changes the electrical connection between its terminals, signaling a change in state to the connected system.

2. **PIR sensor**

    Analog or Digital: PIR sensors are typically digital sensors, providing a digital output signal indicating the presence or absence of motion.

    Pin Modulation: PIR sensors often have digital output pins that toggle between high and low states to indicate motion detection.

    Types of Data: PIR sensors primarily read changes in infrared radiation levels caused by motion within their detection range.

    Data Transfer to System: The digital output signal from the PIR sensor is transferred to a microcontroller or other digital input device, usually through a GPIO (General Purpose Input/Output) pin.

How They Work: PIR sensors detect changes in infrared radiation emitted by objects within their field of view. When motion is detected, the sensor's electronics generate a digital output signal, triggering an alert or other action in the connected system.

3. **Force sensitive resistor:**

   Analog or Digital: Force-sensitive resistors (FSRs) are typically analog sensors. They change their resistance in response to applied force, resulting in a continuous analog output signal.

   Pin Modulation: FSRs do not involve pin modulation. They change resistance based on the applied force, affecting the voltage or current passing through the sensor.

   Types of Data: FSRs measure the force applied to them, typically providing an analog resistance value that varies with the applied force.

   Data Transfer to System: The resistance change of the FSR is typically read by an analog-to-digital converter (ADC) connected to a microcontroller or other digital input device. The ADC converts the analog resistance value into a digital signal that can be processed by the system.

   How They Work: FSRs consist of a conductive polymer material sandwiched between two conductive layers. When force is applied to the FSR, the conductive particles in the polymer material move closer together, reducing the resistance of the sensor. This change in resistance is proportional to the applied force, allowing the sensor to measure force variations.

4. **Infrared Flame Detection Sensor Module:**

   Analog or Digital: Infrared flame detection sensor modules can be either analog or digital, depending on the specific model.

   Pin Modulation: Infrared flame detection sensor modules typically output an analog voltage or current signal (for analog modules) or have digital output pins (for digital modules) that toggle between high and low states to indicate the presence or absence of a flame.

   Types of Data: Infrared flame detection sensor modules detect the presence of a flame based on the infrared radiation emitted by the flame. They provide output signals indicating the presence or absence of a flame.

   Data Transfer to System: For analog modules, the analog output signal is typically read by an analog-to-digital converter (ADC) connected to a microcontroller or other digital input device. For digital modules, the digital output signal is directly read by the microcontroller or digital input device.

How They Work: Infrared flame detection sensor modules contain infrared-sensitive components that detect the infrared radiation emitted by a flame. When a flame is detected, the sensor generates an output signal indicating the presence of a flame. The sensitivity and detection range of the sensor can vary depending on the specific model and design.

5. **DHT22:**

   Analog or Digital: The DHT22 sensor is a digital sensor. It provides a digital signal output containing temperature and humidity data.

   Pin Modulation: The DHT22 sensor communicates using a single-wire digital protocol. It has a data pin through which it sends digital signals containing temperature and humidity readings.

   Types of Data: The DHT22 sensor measures both temperature and humidity. It provides digital output signals containing these measurements.

   Data Transfer to System: The DHT22 sensor communicates with a microcontroller or other digital input device using a single-wire digital protocol. The digital output signal containing temperature and humidity data is read by the microcontroller or digital input device.
   How They Work: The DHT22 sensor contains a temperature and humidity sensor element, along with a signal processing chip. When prompted by the microcontroller, the sensor takes temperature and humidity measurements and sends digital signals containing these measurements to the microcontroller. The microcontroller then processes the data and performs any necessary actions based on the readings.

6. **Relay module:**

   Analog or Digital: Relay modules are digital devices. They have a digital input signal that controls the switching of the relay contacts between open and closed states.

   Pin Modulation: Relay modules typically have digital input pins that toggle between high and low states to control the relay. They may also have additional pins for power supply and ground connections.

   Types of Data: Relay modules do not read data; instead, they are used to control the flow of electrical current in a circuit. They have output contacts that can be used to switch high-power loads such as motors, lights, or heaters.

Data Transfer to System: The digital input signal to the relay module is usually provided by a microcontroller or other digital output device. When the input signal changes state (e.g., from low to high), the relay module switches its contacts accordingly.

How They Work: Relay modules consist of an electromechanical relay and associated control circuitry. When the input signal to the relay module changes state, the control circuitry energizes or de-energizes the relay coil, causing the relay contacts to switch between open and closed states. This allows the relay module to control the flow of electrical current in a separate circuit connected to its output contacts.

7. **HC-06 bluetooth module:**

Analog or Digital: The HC-06 Bluetooth module is a digital device. It communicates wirelessly using Bluetooth technology and provides a digital serial interface for data exchange.

Pin Modulation: The HC-06 module typically has several pins, including VCC (power supply), GND (ground), TX (transmit), and RX (receive).

Types of Data: The HC-06 module is used for wireless data communication between devices. It can transmit and receive various types of data, such as text, numbers, or sensor readings.
Data Transfer to System: Data is transferred to and from the HC-06 module via its TX and RX pins. It can be connected to a microcontroller or other digital device that supports serial communication (e.g., Arduino) to exchange data wirelessly.

How They Work: The HC-06 module uses Bluetooth technology to establish a wireless connection with other Bluetooth-enabled devices, such as smartphones, tablets, or other modules. Once paired, data can be transmitted and received between devices using the serial communication protocol. The HC-06 module acts as a bridge between the wired and wireless communication interfaces, enabling wireless data exchange between devices.

8. **Buzzer:**

Analog or Digital: Buzzer modules are typically digital output devices. They have a digital input signal that controls the generation of sound.

Pin Modulation: Buzzer modules usually have two pins: one for power (VCC) and one for the digital input signal.

Types of Data: Buzzer modules do not read data; they are used to generate audible alerts, tones, or alarms.

Data Transfer to System: The digital input signal to the buzzer module is usually provided by a microcontroller or other digital output device. When the input signal changes state (e.g., from low to high), the buzzer generates sound.

How They Work: Buzzer modules typically contain a piezoelectric element or an electromagnetic coil. When an electrical signal is applied to the buzzer, it causes the element to vibrate, producing sound waves. The frequency and duration of the sound can be controlled by varying the characteristics of the electrical signal applied to the buzzer.

9. **DC motor:**

Analog or Digital: DC motors are typically analog devices.

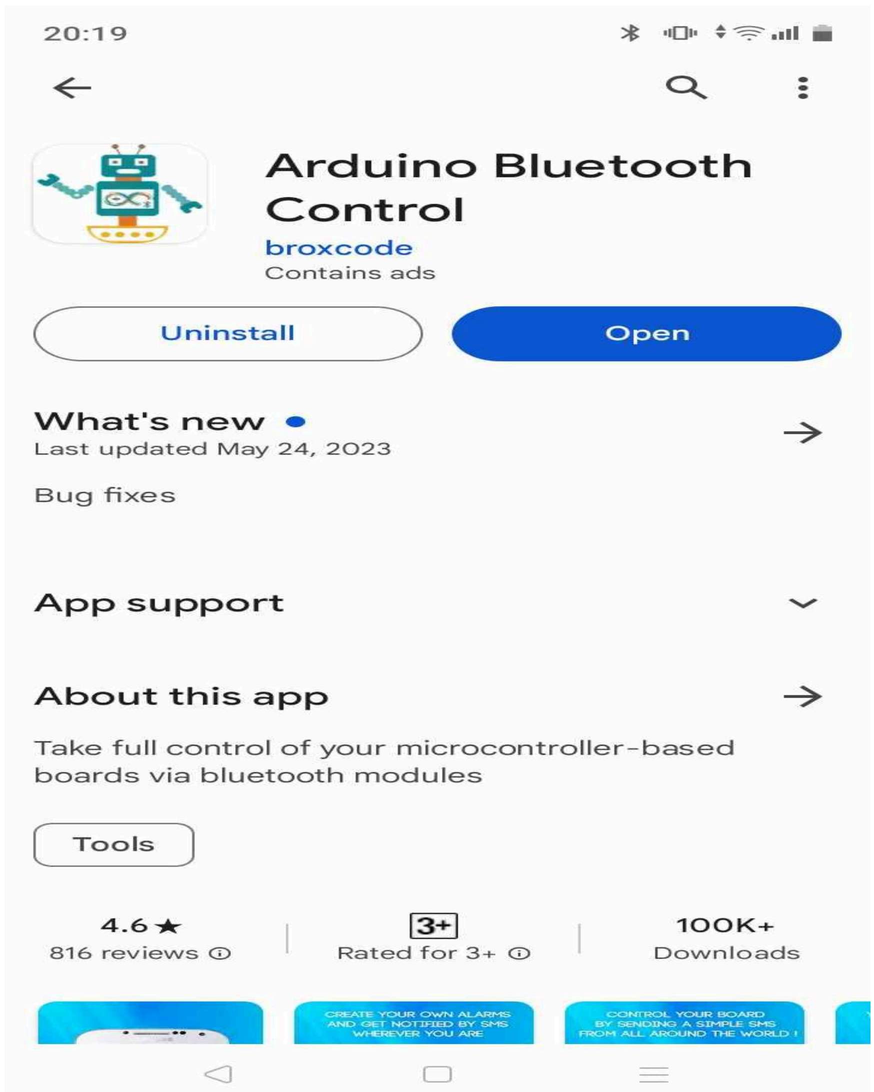
Pin Modulation: DC motors typically have two terminals for connecting to a power supply. The speed is controlled by varying the voltage applied to the motor.

Types of Data: DC motors do not read data; they are used to convert electrical energy into mechanical motion.

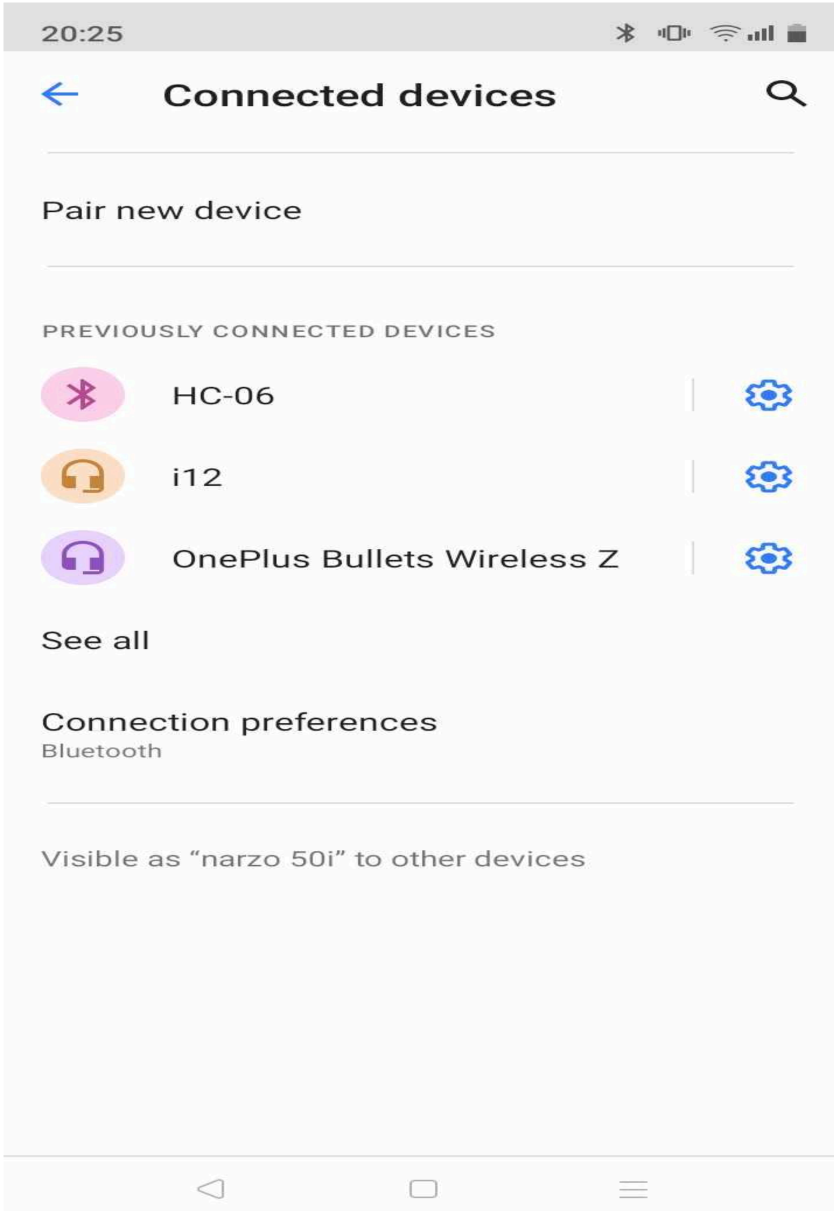
Data Transfer to System: The power supply to the DC motor is usually provided by a motor driver or H-bridge circuit, which is controlled by a microcontroller or other digital output device. The microcontroller generates PWM (Pulse Width Modulation) signals to control the speed.
How They Work: DC motors operate based on the principle of electromagnetic induction. When electrical current flows through the motor's coil (armature), it creates a magnetic field that interacts with the magnetic field produced by the motor's permanent magnets. This interaction causes the armature to rotate, producing mechanical motion. By controlling the voltage and polarity of the power supply, the speed of the motor can be controlled.

User Manual:

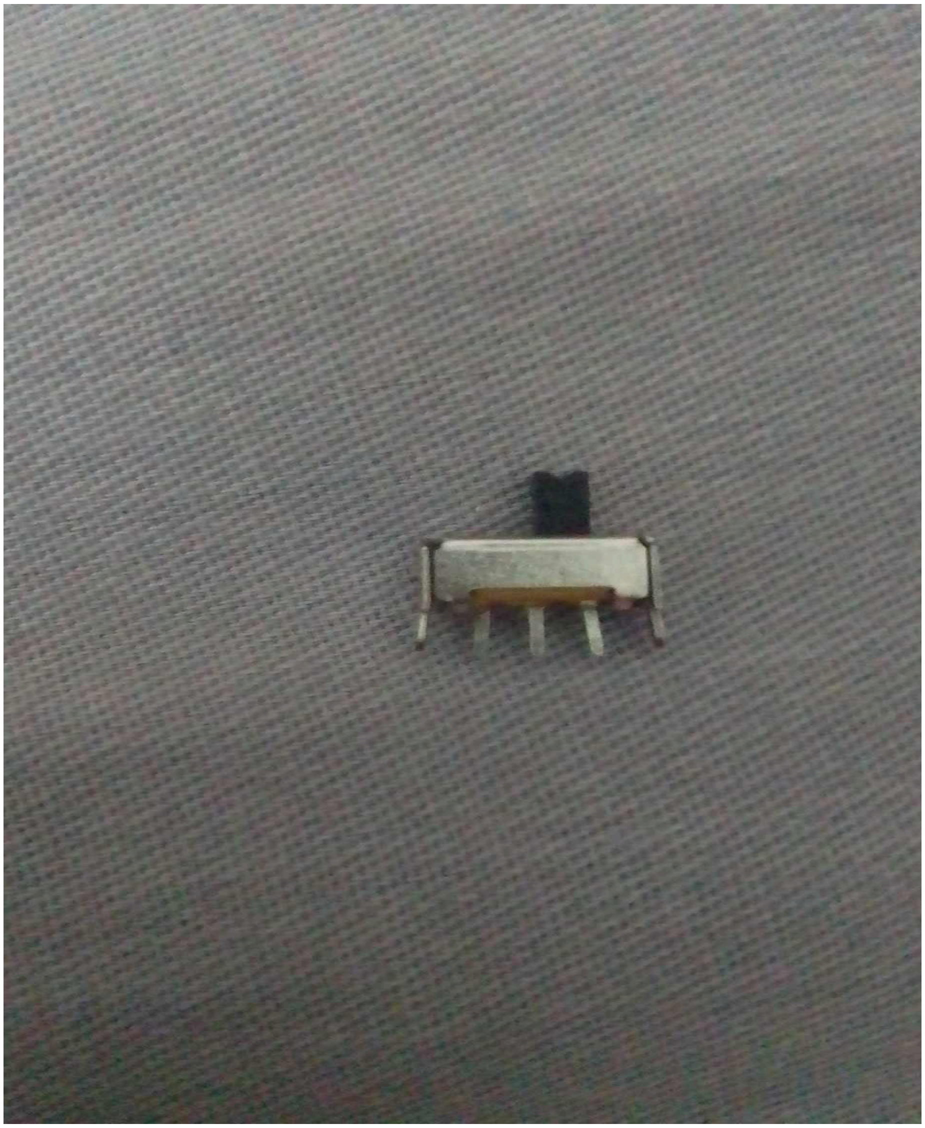
1. The user need to have ‘arduino bluetooth controller’ app installed on their phone.



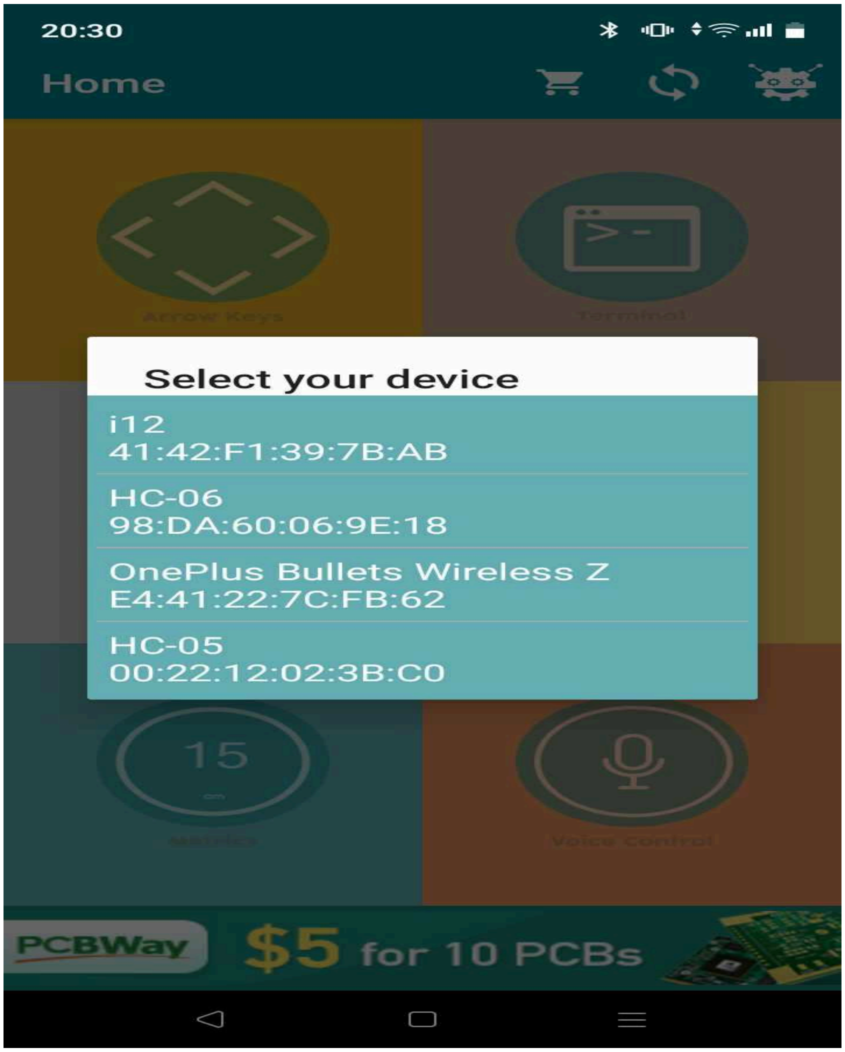
2. The user needs to turn their bluetooth ON and connect to HC-06 module.

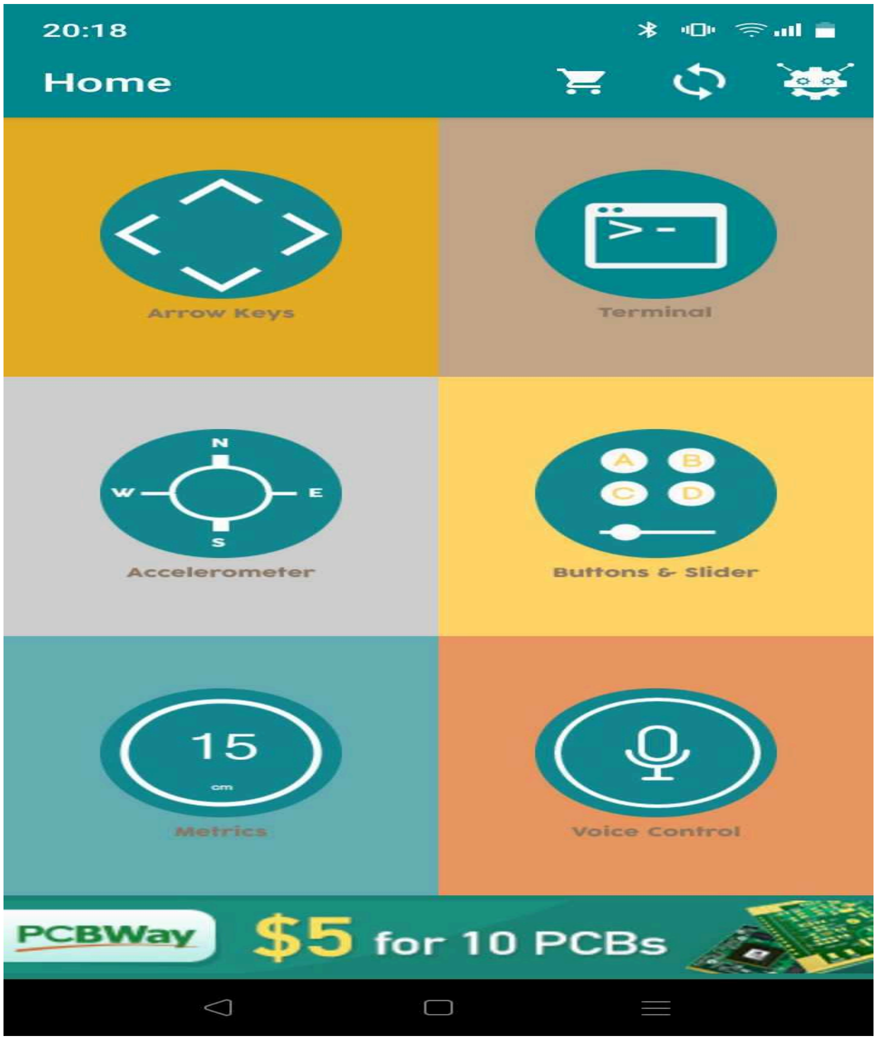


3. The user needs to turn ON the switch to activate the system.

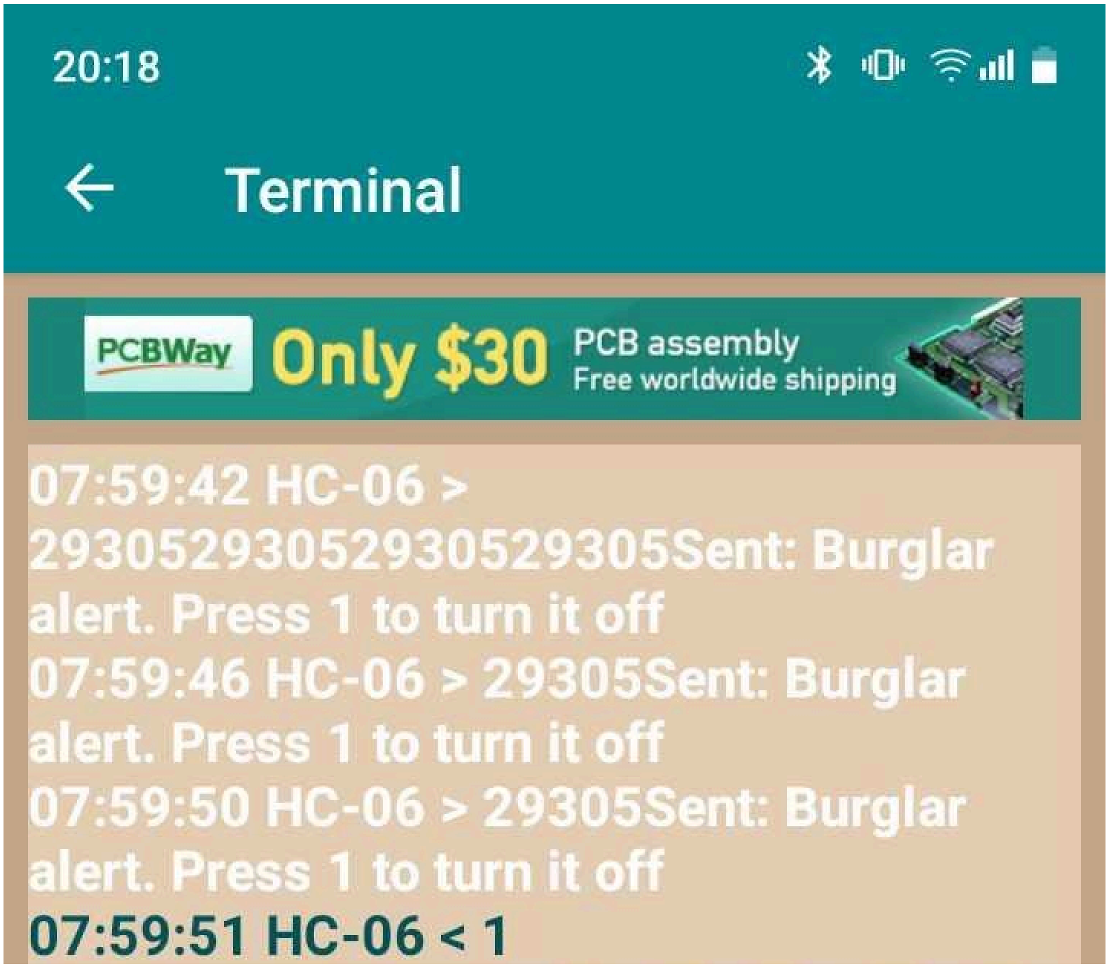


4. The user need to connect the HC-06 module with the app and then click on the terminal.

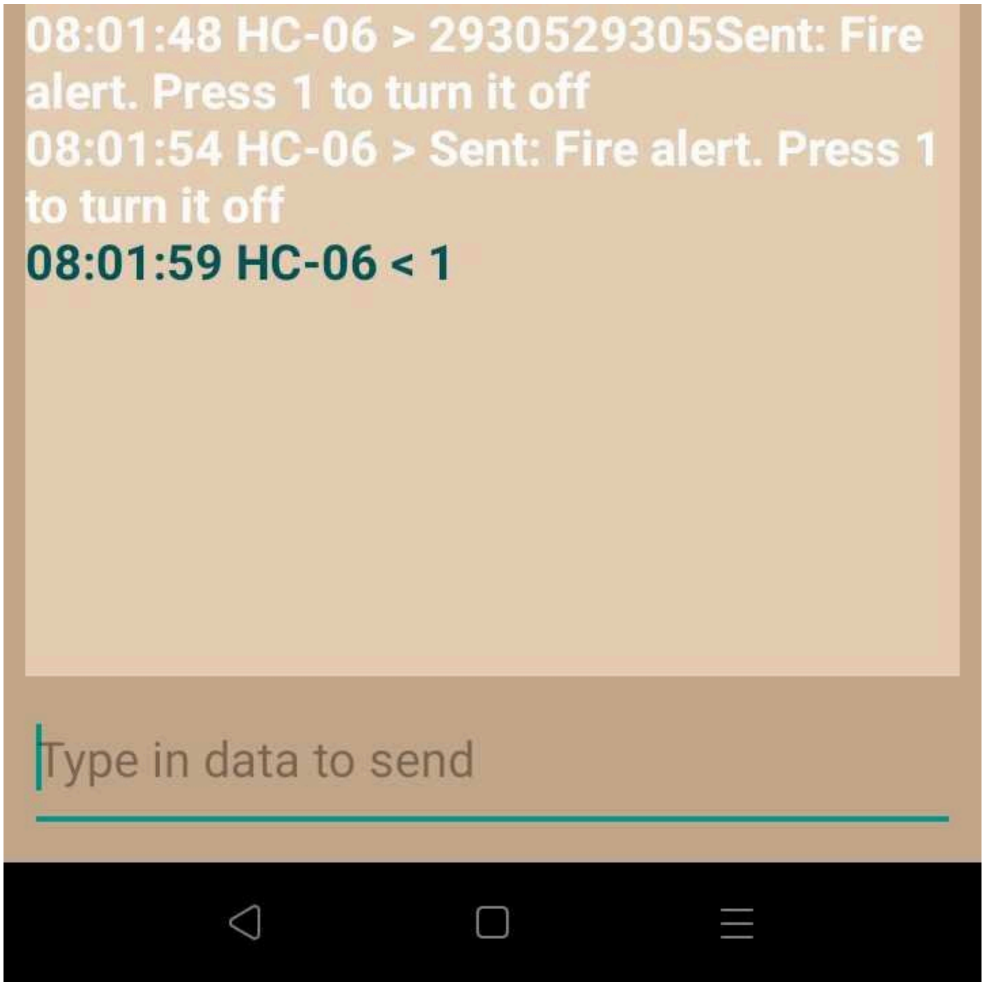

20:30
Home
Select your device
i12
41:42:F1:39:7B:AB
HC-06
98:DA:60:06:9E:18
OnePlus Bullets Wireless Z
E4:41:22:7C:FB:62
HC-05
00:22:12:02:3B:C0
15
PCBWay
$5 for 10 PCBs



5. The user can view message and press 1 to deactivate the ant-theft alarm system



6. The user can view message and press 1 to deactivate the fire alarm system



Challenges:

- Maintenance: This system will have many functionalities, sensors and sensitive connections which might be difficult for everyone to maintain and manage.
- False Alarm: The system can get falsely activated due to something unintentional triggering the sensors or due to excessive heat. A killswitch has been installed for this scenario.
- Cost: While the system itself is quite inexpensive and easy to build, the cost related to maintenance and management of the system might be high for some.
- Integration: The system might be hard to integrate in environments which already have such anti-theft systems installed which are problematic to integrate with.
- Bluetooth System and Range: The user has to keep the bluetooth module on their phone always active which can be detrimental for the battery life. Also, the range of bluetooth transmitters are not really ideal for large environments and might cause delay.

Conclusion:

To sum up, the installation of fire and anti-theft alarm systems is essential for protecting both property and public safety. These devices help to prevent theft or tampering, notify residents, detect fires early, and speed up emergency response times. Even while these systems have a lot to offer, there are drawbacks as well, like making sure that maintenance and management are done properly, dealing with annoying false alarms, and controlling expenses. It is imperative that users prioritize routine maintenance, user education, and the thoughtful placement of alerts and protection measures in order to optimize the effectiveness of these systems. Users may improve the dependability and effectiveness of their fire safety and security systems by proactively resolving these issues, which will ultimately help to create a safer environment for all inhabitants.