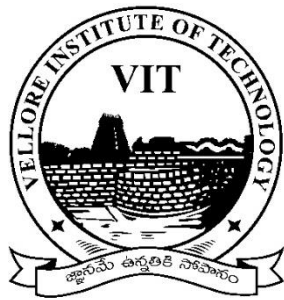


SECURITY SYSTEM USING ARDUINO



VIT[®]
AP

MEMBERS INVOLVED:

- | | |
|------------------------|-------------|
| 1. Gurmehar Singh Soni | [18BCD7095] |
| 2. Saswata Haldar | [18BCD7096] |
| 3. Rajat Kumar | [18BCD7079] |
| 4. Praveen Raj | [18BCE7088] |
| 5. Glen Paulson V | [18BCE7010] |
| 6. Viswanadha Vedvyas | [18BCE7190] |

UNDER THE GUIDANCE OF:

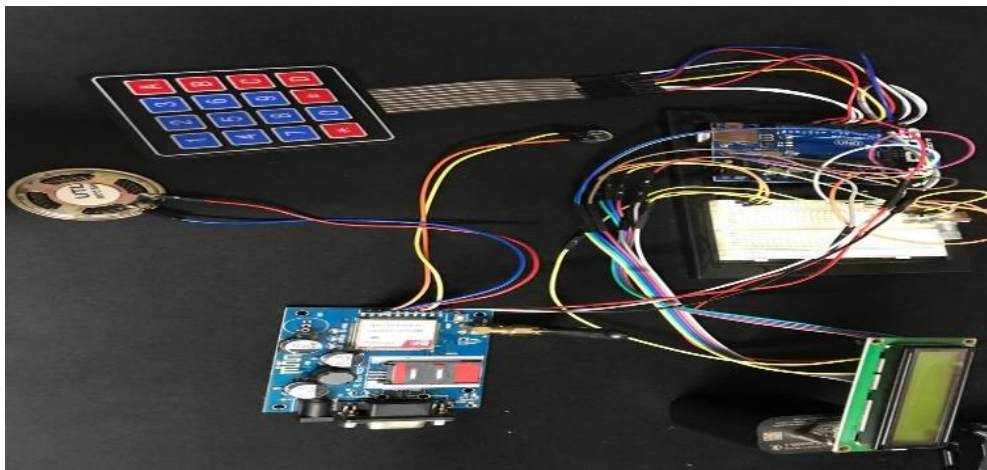
Prof. Anoop Kumar Mishra

Department of ECE

ABSTRACT

Security System Using Arduino is a smart alternative to hard and inefficient work. The main goal of the project is to reduce the difficulty of work for the security personnel who have to cope with challenging weather and unruly people. As a complimentary result we also achieve better security. Any person entering the premises **WILL HAVE TO GO THROUGH THE MODULE LOCATED AT THE GUARD BOOTH NEAR THE GATE**. As the system is Arduino based; after the prototype will go into mass production it will not only be very cost effective but also available easily, making it a go-to security system. As Arduino is a very simple platform the maintenance cost also decreases exponentially with very less human effort required. As the system is a stand-alone system, it not only avoids the hassle of multiple servers and systems but also makes it extremely safe and nearly unhackable. The stand-alone feature also allows for easy and effective surveillance.

All in all the system will provide a great advantage over the existing security models in a very consumer friendly and cost effective way.



CONTENTS

SR NUMBER	TITLE	PAGE NUMBER
1	Abstract	2
2	Introduction	5
3	Background	6
4	Problem definition	6
5	Objective	7
6	Methodology/procedure	7
7	Results and Discussion	12
8	Conclusion and Future Scope	13
9	References	14
10	Codes in Appendix	15

LIST OF FIGURES

SR NO	IMAGE	PAGE
1	Working prototype	2
2	Arduino and gsm module	9
3	Arduino and Keypad	10
4	Arduino and Lcd	10
5	Complete circuit diagram	11
6	Architectural diagram	12

INTRODUCTION

We live in a society where the need for security is important. And when it comes to dealing with personal or private security, there's no need for one to have second thoughts on whether security is required or not. For a long time, it has played a major role in the major part of our life. Through the years, many businesses and institutions all over the world have used security to protect their assets and prevent crime. As a result, all these organisations make sure that they employ the best security personnel and best security equipment to ensure the same.

Some of the identified core elements of the security field are: personal security, physical security, loss prevention, risk management, legal aspects, crisis and disaster management, violence in workplace along with many other factors. Security breach has become very important in today's world, as a result more methods need to be adopted to confront the attempts to bypass the security. So, speaking of security access control systems are a must. In the past we used the lock and key access systems. But with the advent of technology we have come a long way from that. With the present technology we have more advanced machinery and algorithms that helps to make sure that security is ensured to the employer.

Our device is one such device that employs and makes use of the modern technology to ensure the same. It's kind of a stand-alone device that's powered by an Arduino, a gsm module and a few more other components that helps in the smooth working of the system. Even though the project's intention is to enhance the access control system in the educational institutions where the security personnel has to follow a long procedure of verifying an outsider, the same project can be implemented in various other scenarios. As stated, our project helps the security personnel verify an outsider and in case if the outsider doesn't have a pin, he can even send a message to the security head, who will then verify the credentials of the outsider and give the person an access code if and only if the persons details checks out.

Though at present our project has integrated everything that can be done in a given short span of time, we believe it still has the potential to incorporate more details and facilities, which will take our project to the next level with several other addons making the procedure of access authorization far easier for the security personnel and ensure greater security for the host and the guest at the same time.

BACKGROUND

The reason behind the success of this project resides with the six-member team that toiled days behind this project, trying to resolve the unforeseen issues that aroused from the same. Even though the project might seem to be simple in concept, the implementation requires a lot of coding for each component to be integrated.

As stated in the introductory page, when the time came for the team to choose a topic for the project, the team picked this one because after all we believed that lack of security is something that can affect life adversely. And if there's something that we could do to ensure better security, it would definitely help to improve life in one way or another.

So, then the whole team put their hearts and will to it, started from scratch, improvising solutions to all the unforeseen problems and finally integrated all the components so as to deliver our promise.

We are also extremely grateful to and indebted to our project guide and all other friends who gave us a hand during the difficult times.

The detailed facts and technical details regarding the project will be discussed in the following pages of the report.

PROBLEM DEFINITION

It is quite cumbersome for someone to stand in the scorching heat of the Andhra sun at the gate and do their duty, no matter how much they are trained for it. Because they are human. Also, there is no guarantee if the visitor is in his right mind. He/She might just have a rough morning or an audience with a pesky traveler on the way there. There is no saying if they would be sympathetic and tolerant with the security personnel. Once again because they are human. Therefore, to reduce plausible errors and conflicts we propose to remove the human point of interaction with a more secure and error free alternative. The modus operandi of this alternative is a pin generated for all the members with access to the facility (in this case the campus of VIT-AP), which when entered at the module located near the gate, grants immediate access inside the premises. In case the person has some work within the premises but was not a direct part of the institution, they could call the security personnel in charge of the gate and verify their identity by means of PAN/Aadhar or any other such system. Once verified the person gets access, to enter. All without chances of quarrels, fraud or infiltration. There will be a guard booth stationed near the gate which would house this module. The guard will be situated somewhere from where he could reach the gate if need arose. Thus, eliminating also his hardships to be stationed in front of the gate in hard weather like hot summers and rains. Therefore, this shall solve the problem of human error and make the system more secure and efficient all the while reducing the hardships we put the guards through entrusting them with our security.

OBJECTIVE

Our objective is to make a stand alone security system using arduino which can send and receive call and sms with the help of GSM module. It can be used at big Industries, Institutions and public places which have multiple gate openings. There it can be used by guards to communicate with each other. This will reduce the stress on guards. This will lead the industry towards automation and this will be the cost effective solution.

PROCEDURE/METHODOLOGY

1. CIRCUIT BUILDING

- In the beginning gather all the components i.e. Arduino Uno, GSM Module SIM900, 16x2 LCD, 4x4 Keypad, Breadboard or PCB, Connecting jumper wire, Power supply, Speaker, MIC, SIM Card
- Then connect the LCD display to the analog pins of the Arduino board. Connect RS, E, D4, D5, D6, D7 to A0, A1, A2, A3, A4, A5 on the Arduino board respectively. Connect VDD, L+ pin to 5V supply and VEE and L- to the potentiometer. Ground the VSS, RW pin. One pin from the potentiometer is also connected to the 5v supply. Use 10K ohm potentiometer only. Potentiometer is used to control the brightness of the display
- Next connect the keypad to the Arduino. A, B, C, D pins on keypad are connected to the D11, D10, D9, D8 on the board respectively. 1, 2, 3, 4 pins are then connected to the D7, D6, D5, D4 pins on the board respectively.
- After this the gsm module is connected . Connect Rx, Tx pin on the module to the D3, D2 pin on the board respectively. Vcc pin is connected to the 5V supply and GND pin is connected to ground on the Arduino uno. Connect the speaker to SP+ and SP- on the gsm module and microphone to mic+ and mic-. Supply the gsm module with 12V supply. Insert a sim in the slot provide.

2. CODING

- Open Arduino IDE and write the code.
- Compile the code.
- Upload the code to the board by clicking on the upload button and connecting the board to one of the usb ports on the pc. Ensure that you have selected the correct port.
- Make sure that the gsm module is disconnected from the board while the code is getting uploaded.
- Once the code is uploaded, connect the gsm module to the board and the power supply.
- Allow the module to stabilize.

- The system should be running well and the display should be asking you to enter the pin.

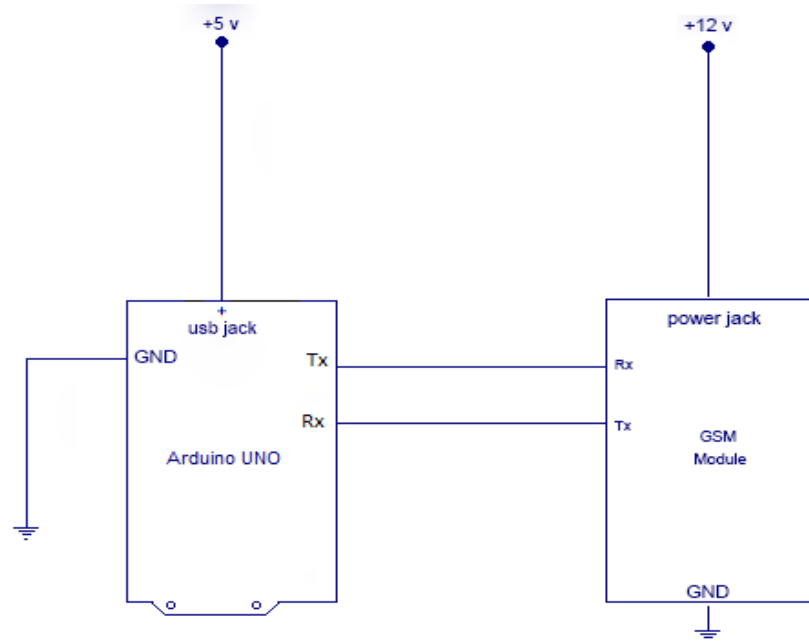
3. RUNNING THE SYSTEM

- First enter the pin using the keypad. If the pin is correct, the gate will automatically open.
- If the pin is incorrect, the system will automatically call the guard
- Tell your Name and your date of birth to the guard
- Disconnect the call
- Enter your Aadhar number by pressing B on the keypad. Send it by again pressing the B button.
- The guard enters the Aadhar number on the UIDAI website.
- If the Aadhar details available on UIDAI website matches with the details given to the guard on the call, the guard opens the gate by pressing a button.
- If the details do not match, the person is not allowed to enter.

4. COMPONENTS EXPLAINED

1. GSM Module

- The GSM module we used for our project is SIM900A. The 900 in the name signifies its working frequency that is 900 Hz or 1800 Hz. The power requirement for this module is 12V and 1A. The module has three led which help us in finding if the module is working correctly. The band rate can be varied between 9600 – 115200.
- The first led is power which glows red when the dc power is given to the module. The second led is the status led which will glow blue if the power provided is sufficient for the working of the module. The third pin is the network led will blink green with the blink rate of 1 blink per second which means the module is searching for network. When the module is connected to the network, we see the blink rate drops to 1 blink per 3 sec.
- The antenna used with the module is a SMA antenna which stands for standard male antenna. The module has 10 pins but for this project we only require 4 pins, namely Tx, Rx, GND, VCC. The Tx pin is connected to the D2 pin of the Arduino and the Rx pin to D3 while GND to ground and VCC to 5V supply of the Arduino.
- The work of the module in our project is to call and send SMS to the guard. The circuit diagram of only GSM module connected to Arduino is

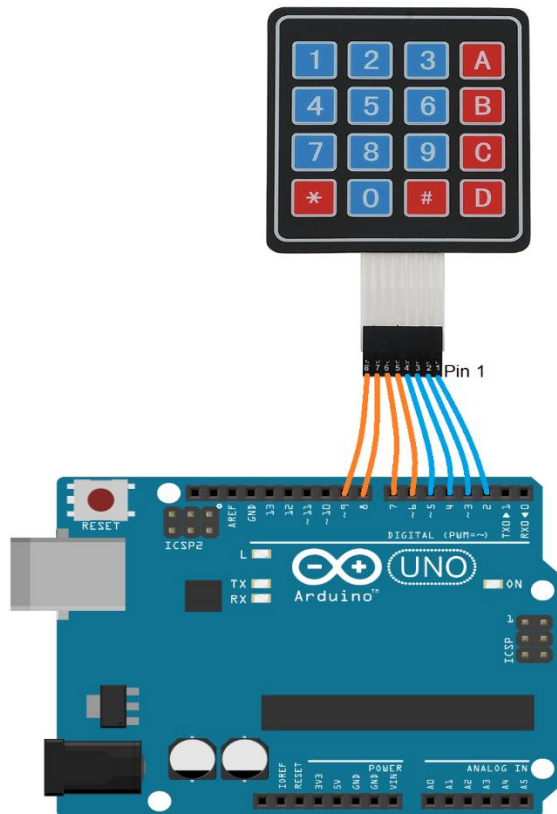


2. Arduino

- The Arduino uses ATmega328 microcontroller which requires a working voltage of 5V. But the input voltage can vary between 7V to 20V. There are 14 digital pins in Arduino and 6 Analog pins. There are two power ports, one of 5V other of 3.3V. It has 3 GND pins.
- The Arduino is capable of collecting data and do some calculations using that data and give output as per the result of the function and conditions written in the code. The programming for Arduino is done in Arduino IDE which is a free software.
- In our project the Arduino is used to control the GSM module, the LCD display and the keypad and it helps in sending and receiving a message or calling.

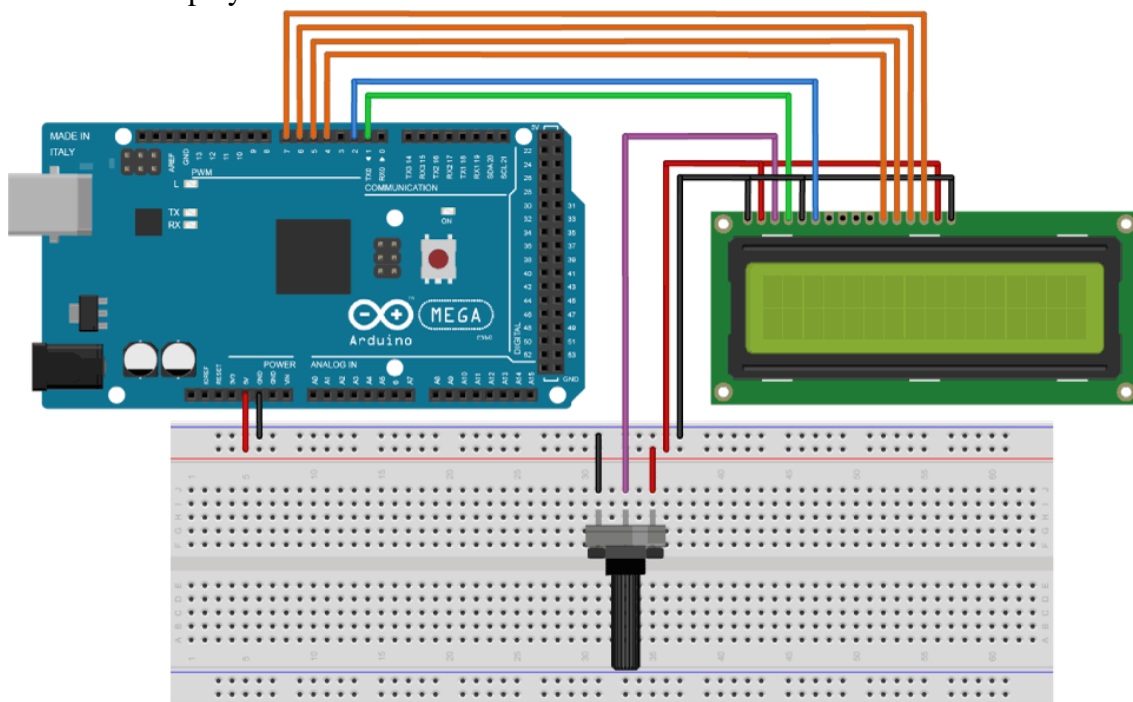
3. Keypad

- It is used for entering the data. It has 8 pins representing each row and column.
- In our project we have used it for entering the phone number, text message, calling and receiving calls and sending and receiving messages



4. LCD Display

- Lcd display is used to display the output. It has 16 pins.
- We have used a 16x2 LCD for our project.
- In our project we have used E,RS,D4,D5,D6,D7,RW,VSS,L-,L+,VDD,VEE pins of the display.



5. Libraries

5.1 Software serial library

- The Arduino hardware has built-in support for serial communication on pins 0 and 1. But two pins are not enough when we use many modules at once. To overcome this problem the SoftwareSerial library has been developed to allow serial communication on other digital pins of the Arduino, using software to replicate the functionality (hence the name "SoftwareSerial").
- It is possible to have multiple software serial ports with speeds up to 115200 bps. In our project we use a speed of 9600 bps because it is the defines band rate for GSM module.

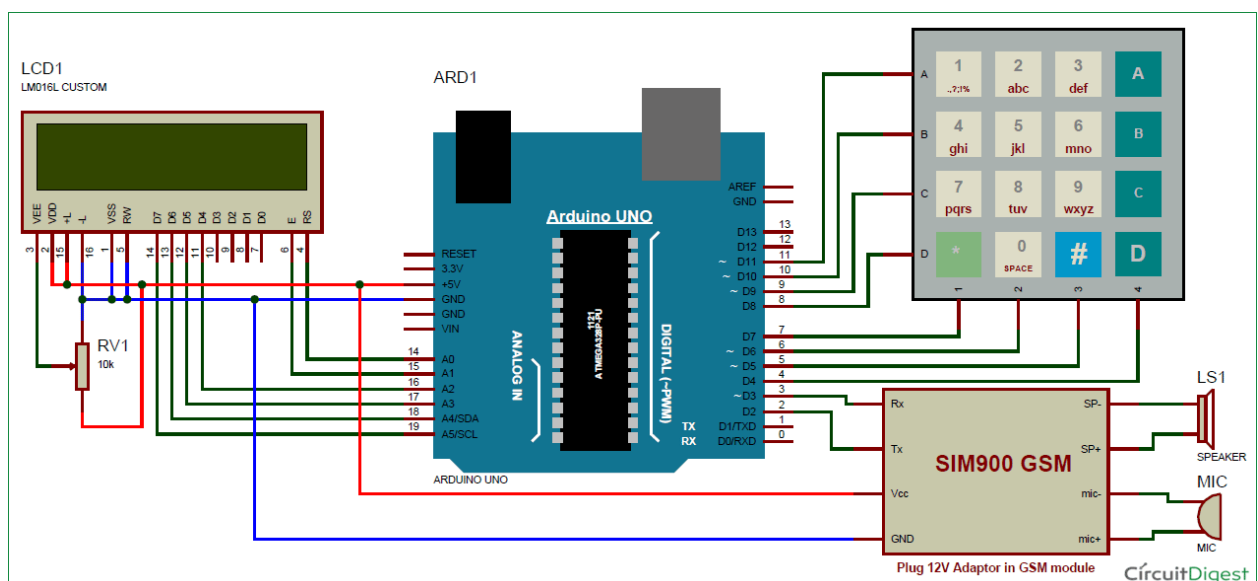
5.2 Liquid Crystal Library

- This library allows an Arduino board to control LiquidCrystal displays (LCDs) based on the Hitachi HD44780 (or a compatible) chipset, which is found on most text-based LCDs. The library works with in either 4- or 8-bit mode (i.e. using 4 or 8 data lines in addition to the rs, enable, and, optionally, the rw control lines).
- We have used this library to control the LCD

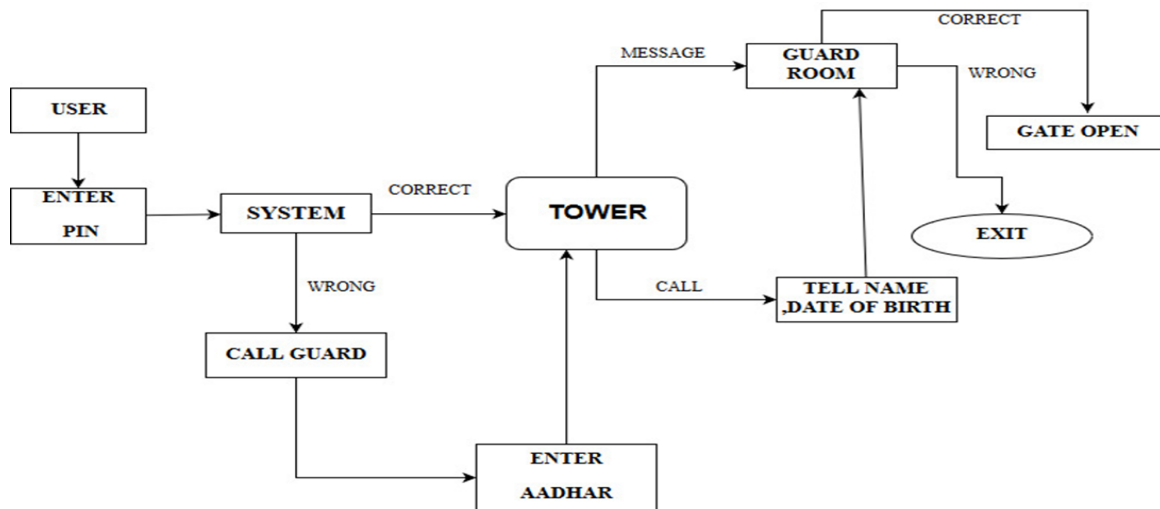
5.3 Keypad Library

- This Library is used for controlling the matrix style keypads with the arduino. It is not a built in library and has to be imported separately.
- This library supports multi-keypresses
- We have used this to control our keypad.

The final circuit diagram:



Architectural Diagram



RESULT AND DISCUSSIONS:

The results of this project include the working of the LCD, even though we faced a bit of difficulty at first, the GSM Module, the speakers, the mic in a unison. We were able to achieve the call and messaging functionality in our project.

Objectives Met:

1. Calling Function
2. Messaging Function
3. LCD Display

Objectives Not-Met:

1. Linking the whole module with Aadhar database.

We had to re-configure the connections of the LCD Display due to the error while soldering the LCD to the jumper wires.

We had to get a local GSM Subscriber Identification Module so as to remove the delay in calling and messaging.

We combined the LCD Module and the Alpha-Numerical keypad to make the user enter the mobile number and the message which he/she wants to send.

CONCLUSION AND FUTURE SCOPE

The present study has been made to suggest and develop some tools which will eventually be useful to the security firms to provide safety and security at reasonable cost and of a specified quality. This project is targeted towards financing, designing, implementing and operating security facilities and services that were traditionally expensive and complicated yet with many loopholes. This project aims to provide cost effective security solutions to everyone ranging from home security solutions to providing security to large MNC's. As this project is simple to operate anyone with some basic knowledge of computers can operate this. Also because one server has to be maintained over which the whole system will be based and operate for that particular client maintenance is largely reduced. As arduino boards and components are the main building blocks of this system, this product highly cost-effective and easy to mass produce.

That being said the current prototype can be highly improved in the coming years with more R&D.

- Can be a fully integrated multi modular security system. This will connect each security personal and device over a single secure security network. This will ensure that the system is a stand-alone system and is not hackable easily. This also will cut down maintenance of multiple systems.
- Additional layers of security to counter any attempt of security breach. If anyone who has not been identified tries to forcefully enter the premises, additional security measures such as tire puncturing nails will get activated and an alert will be sent to all the guards. All the gates will also get closed automatically.
- Automated gate opening and closing system on the trigger of a single switch. This will decrease dependence on manual labour and will also increase security as the system will ensure the closure of the gates when not in use or after they have been kept opened for a certain period of time.
- A system present in the main guard room to override the whole protocol in case of an emergency or in case the system fails. Overriding the system will switch to manual operation. This will enable the overall security in charge to take control of a situation if things go out of hand.
- Whole system to run on future proof non-conventional renewable sources of energy such as piezo electrics, solar etc. This will ensure that the system keeps on functioning even when it is cut-off from the main electric grid supply. The motor and electronics will be able to switch between two or more sources of energy to function continuously depending on which source is available that point of time.

REFERENCES

- 1) <https://www.arduino.cc/en/Guide/ArduinoMega2560>
- 2) <http://playground.arduino.cc>
- 3) <https://learn.adafruit.com/category/learn-arduino>
- 4) Sams Teach Yourself “ Arduino Programming” by Richard Blum ,Pearson Education 2015.
- 5) [https://circuitdigest.com/microcontroller-projects/call-and-message-using-arduino-sim900- gsm](https://circuitdigest.com/microcontroller-projects/call-and-message-using-arduino-sim900-gsm)
- 6) https://en.wikipedia.org/wiki/Arduino_Uno
- 7) <https://youtu.be/5esQqZIJ83g>
- 8) https://youtu.be/QO_JlzlqpDw

APPENDIX

CODE FOR RUNNING THE SYSTEM

```
#include <SoftwareSerial.h>
SoftwareSerial Serial1(2, 3); // RX, TX

#include<LiquidCrystal.h>
LiquidCrystal lcd(14,15,16,17,18,19);

byte back[8] =
{
    0b000000,
    0b000000,
    0b111111,
    0b10101,
    0b11011,
    0b11111,
    0b000000,
    0b000000
};

String number="";
String msg="";
String instr="";
String str_sms="";
String str1="";
int ring=0;
int i=0,temp=0;
int sms_flag=0;
char sms_num[3];
int rec_read=0;
int temp1=0;

#include <Keypad.h>
const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns
char hexaKeys[ROWS][COLS] =
{
    {'1','2','3','A'},
    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};

byte rowPins[ROWS] = {11, 10, 9, 8}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {7, 6, 5, 4}; //connect to the column pinouts of the keypad

//initialize an instance of class NewKeypad
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);
String ch="1,.?!@abc2def3ghi4jkl5mno6pqrs7tuv8wxyz90 ";

void setup()
{
    Serial1.begin(9600);
    lcd.begin(16,2);
    lcd.createChar(1, back);
    lcd.print("Simple Mobile ");
    lcd.setCursor(0,1);
    lcd.print("System Ready..");
    delay(1000);
    gsm_init();
}
```

```

    lcd.clear();
    lcd.print("System Ready");
    delay(2000);
}

void loop()
{
    serialEvent();
    if(sms_flag==1)
    {
        lcd.clear();
        lcd.print("New Message");
        int ind=instr.indexOf("+CMTI: \\\"SM\\\",");
        ind+=12;
        int k=0;
        lcd.setCursor(0,1);
        lcd.print(ind);
        while(1)
        {
            while(instr[ind]!= 0x0D)
            {
                sms_num[k++]=instr[ind++];
            }
            break;
        }
        ind=0;
        sms_flag=0;
        lcd.setCursor(0,1);
        lcd.print("Read SMS --> D");
        delay(4000);
        instr="";
        rec_read=1;
        temp1=1;
        i=0;
    }

    if(ring == 1)
    {
        number="";
        int loc=instr.indexOf("+CLIP: \\\"");
        if(loc > 0)
        {
            number+=instr.substring(loc+8,loc+13+8);
        }
        lcd.setCursor(0,0);
        lcd.print("Incomming... ");
        lcd.setCursor(0,1);
        lcd.print(number);
        instr="";
        i=0;
    }
    else
    {
        serialEvent();
        lcd.setCursor(0,0);
        lcd.print("Call --> C ");
        lcd.setCursor(0,1);
        lcd.print("SMS --> B ");
        if(rec_read==1)
        {

```



```

    lcd.write(1);
    lcd.print("  ");
}
else
lcd.print("  ");
}

char key=customKeypad.getKey();
if(key)
{
    if(key== 'A')
    {
        if(ring==1)
        {
            Serial1.println("ATA");
            delay(5000);
        }
    }
    else if(key=='C')
    {
        call();
    }
    else if(key=='B')
    {
        sms();
    }
    else if(key == 'D' && temp1==1)
    {
        rec_read=0;
        lcd.clear();
        lcd.print("Please wait...");
        Serial1.print("AT+CMGR=");
        Serial1.println(sms_num);
        int sms_read_flag=1;
        str_sms="";
        while(sms_read_flag)
        {
            while(Serial1.available()>0)
            {
                char ch=Serial1.read();
                str_sms+=ch;
                if(str_sms.indexOf("OK")>0)
                {
                    sms_read_flag=0;
                    //break;
                }
            }
        }
        int l1=str_sms.indexOf("\r\n");
        int l2=str_sms.indexOf("OK");
        String sms=str_sms.substring(l1+3,l2-4);
        lcd.clear();
        lcd.print(sms);
        delay(5000);
    }
    delay(1000);
}
}

```

```

void call()
{
    number="";
    lcd.clear();
    lcd.print("After Enter No.");
    lcd.setCursor(0,1);
    lcd.print("Press C to Call");
    delay(2000);
    lcd.clear();
    lcd.print("Enter Number:");
    lcd.setCursor(0,1);
    while(1)
    {
        serialEvent();
        char key=customKeypad.getKey();
        if(key)
        {
            if(key=='C')
            {
                lcd.clear();
                lcd.print("Calling.....");
                lcd.setCursor(0,1);
                lcd.print(number);
                Serial1.print("ATD");
                Serial1.print(number);
                Serial1.println(";");
                long stime=millis()+5000;
                int ans=1;
                while(ans==1)
                {
                    while(Serial1.available()>0)
                    {
                        if(Serial1.find("OK"))
                        {
                            lcd.clear();
                            lcd.print("Ringing....");
                            int l=0;
                            str1="";
                            while(ans==1)
                            {
                                while(Serial1.available()>0)
                                {
                                    char ch=Serial1.read();
                                    str1+=ch;
                                    if(str1.indexOf("NO CARRIER")>0)
                                    {
                                        lcd.clear();
                                        lcd.print("Call End");
                                        delay(2000);
                                        ans=0;
                                        return;
                                    }
                                }
                            }
                        }
                    }
                    char key=customKeypad.getKey();
                    if(key == 'D')
                    {
                        lcd.clear();
                        lcd.print("Call End");
                        delay(2000);
                        ans=0;
                    }
                }
            }
        }
    }
}

```

```

        return;
    }
    if(ans==0)
        break;
    }
}
}
}
else
{
    number+=key;
    lcd.print(key);
}
}
}
}
void sms()
{
    lcd.clear();
    lcd.print("Initilising SMS");
    Serial1.println("AT+CMGF=1");
    delay(400);
    lcd.clear();
    lcd.print("After Enter No.");
    lcd.setCursor(0,1);
    lcd.print("Press D ");
    delay(2000);
    lcd.clear();
    lcd.print("Enter Rcpt No.:");
    lcd.setCursor(0,1);
    Serial1.print("AT+CMGS=\"");
    while(1)
    {
        serialEvent();
        char key=customKeypad.getKey();
        if(key)
        {
            if(key=='D')
            {
                //number+="";
                Serial1.println("\");
                break;
            }
            else
            {
                //number+=key;
                Serial1.print(key);
                lcd.print(key);
            }
        }
    }
    lcd.clear();
    lcd.print("After Enter MSG ");
    lcd.setCursor(0,1);
    lcd.print("Press D to Send ");
    delay(2000);
    lcd.clear();
    lcd.print("Enter Your Msg");
    delay(1000);
}

```

```

lcd.clear();
lcd.setCursor(0,0);
alfakey();
}
void alfakey()
{
int x=0,y=0;
int num=0;
while(1)
{
  lcd.cursor();
  char key=customKeypad.getKey();
  if(key)
  {
    if(key=='1')
    {
      num=0;
      lcd.setCursor(x,y);
      lcd.print(ch[num]);
      for(int i=0;i<3000;i++)
      {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='1')
        {
          num++;
          if(num>5)
          num=0;
          lcd.setCursor(x,y);
          lcd.print(ch[num]);
          i=0;
          delay(200);
        }
      }
      x++;
      if(x>15)
      {
        x=0;
        y++;
        y%=2;
      }
      msg+=ch[num];
    }
    else if(key=='2')
    {
      num=6;
      lcd.setCursor(x,y);
      lcd.print(ch[num]);
      for(int i=0;i<3000;i++)
      {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='2')
        {
          num++;
          if(num>9)
          num=6;
          lcd.setCursor(x,y);
          lcd.print(ch[num]);

```

```

        i=0;
        delay(200);
    }
}
x++;
if(x>15)
{
    x=0;
    y++;
    y%=2;
}
msg+=ch[num];
}
else if(key=='3')
{
    num=10;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='3')
        {
            num++;
            if(num>13)
            num=10;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='4')
{
    num=14;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='4')
        {
            num++;
            if(num>17)
            num=14;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;

```

```

        delay(200);
    }
}
x++;
if(x>15)
{
    x=0;
    y++;
    y%=2;
}
msg+=ch[num];
}

    else if(key=='5')
{
    num=18;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='5')
        {
            num++;
            if(num>21)
            num=18;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}

    else if(key=='6')
{
    num=22;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='6')
        {
            num++;
            if(num>25)
            num=22;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}

```

```

    }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='7')
{
    num=26;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='7')
        {
            num++;
            if(num>30)
            num=26;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='8')
{
    num=31;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='8')
        {
            num++;
            if(num>34)
            num=31;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
}

```

```

    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='9')
{
    num=35;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='9')
        {
            num++;
            if(num>39)
            num=35;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='0')
{
    num=40;
    lcd.setCursor(x,y);
    lcd.print(ch[num]);
    for(int i=0;i<3000;i++)
    {
        lcd.noCursor();
        char key=customKeypad.getKey();
        if(key=='0')
        {
            num++;
            if(num>41)
            num=40;
            lcd.setCursor(x,y);
            lcd.print(ch[num]);
            i=0;
            delay(200);
        }
    }
}

```



```

    x++;
    if(x>15)
    {
        x=0;
        y++;
        y%=2;
    }
    msg+=ch[num];
}
else if(key=='D')
{
    lcd.clear();
    lcd.print("Sending SMS....");
    // Serial1.print("AT+CMGS=");
    // Serial1.print(number);
    // delay(2000);
    Serial1.print(msg);
    Serial1.write(26);
    delay(5000);
    lcd.clear();
    lcd.print("SMS Sent to");
    lcd.setCursor(0,1);
    lcd.print(number);
    delay(2000);
    number="";
    break;
}
}
}
}

void send_data(String message)
{
    Serial1.println(message);
    delay(200);
}

void send_sms()
{
    Serial1.write(26);
}

void lcd_status()
{
    lcd.setCursor(2,1);
    lcd.print("Message Sent");
    delay(2000);
    //lcd.setCursor()
    //lcd.print("")
    //return;
}

void back_button()
{
    //lcd.setCursor(0,15);
}

void ok_button()
{
    lcd.setCursor(0,4);
    lcd.print("OK");
}

```

```

void call_button()
{
  lcd.setCursor(0,4);
  lcd.print("CALL");
}

void sms_button()
{
  lcd.setCursor(0,13);
  lcd.print("SMS");
}

void gsm_init()
{
  lcd.clear();
  lcd.print("Finding Module..");
  boolean at_flag=1;
  while(at_flag)
  {
    Serial1.println("AT");
    while(Serial1.available()>0)
    {
      if(Serial1.find("OK"))
        at_flag=0;
    }

    delay(1000);
  }

  lcd.clear();
  lcd.print("Module Connected..");
  delay(1000);
  lcd.clear();
  lcd.print("Disabling ECHO");
  boolean echo_flag=1;
  while(echo_flag)
  {
    Serial1.println("ATE1");
    while(Serial1.available()>0)
    {
      if(Serial1.find("OK"))
        echo_flag=0;
    }
    delay(1000);
  }

  lcd.clear();
  lcd.print("Echo OFF");
  delay(1000);
  lcd.clear();
  lcd.print("Finding Network..");
  boolean net_flag=1;
  while(net_flag)
  {
    Serial1.println("AT+CPIN?");
    while(Serial1.available()>0)
    {
      if(Serial1.find("+CPIN: READY"))
        net_flag=0;
    }
    delay(1000);
  }
  lcd.clear();

```

```

lcd.print("Network Found..");
delay(1000);
lcd.clear();
}
void serialEvent()
{
  while(Serial1.available())
  {
    char ch=Serial1.read();
    instr+=ch;
    i++;
    if(instr[i-4] == 'R' && instr[i-3] == 'I' && instr[i-2] == 'N' && instr[i-1] == 'G' )
    {
      ring=1;
    }
    if(instr.indexOf("NO CARRIER")>=0)
    {
      ring=0;
      i=0;
    }
    if(instr.indexOf("+CMTI: \"SM\"")>=0)
    {
      sms_flag=1;
    }
  }
}

```