THE NATIONAL COLLEGE [AUTONOMOUS]
BASAVANGUDI, BANGALORE-04

॥ ಶ್ರದ್ಧಾಹಿ ಪರಮಾ ಗತಿಃ ॥

A Project Report On

"SMART DIGITAL LOCKER"

Submitted in Partial fulfilment of the Requirements for the award of marks in

VI SEMESTER BACHELOR OF SCIENCE

In

ELECTRONICS

Project Associates:

BHARATH A	20NCBS3002
DARSHAN GOWDA N	20NCBS3003
NAGESH N	20NCBS3005
MAYUR P SHEETY	20NCBS3012

Under the Guidance of:

Mrs. Poornima Hegde., M.Sc., M.Phil.

HOD of Electronics

AND

Mrs. Ramya M., M.Sc.

Asst. Prof. in Electronics

DEPARTMENT OF ELECTRONICS

NATIONAL COLLEGE BASAVANGUDI, BANGALORE-560004

2022 -2023

THE NATIONAL COLLEGE BASAVANGUDI, BANGALORE

AUTONOMOUS

॥ श्रद्धाहि परमा गतिः ॥

DEPARTMENT OF ELECTRONICS

CERTIFICATE

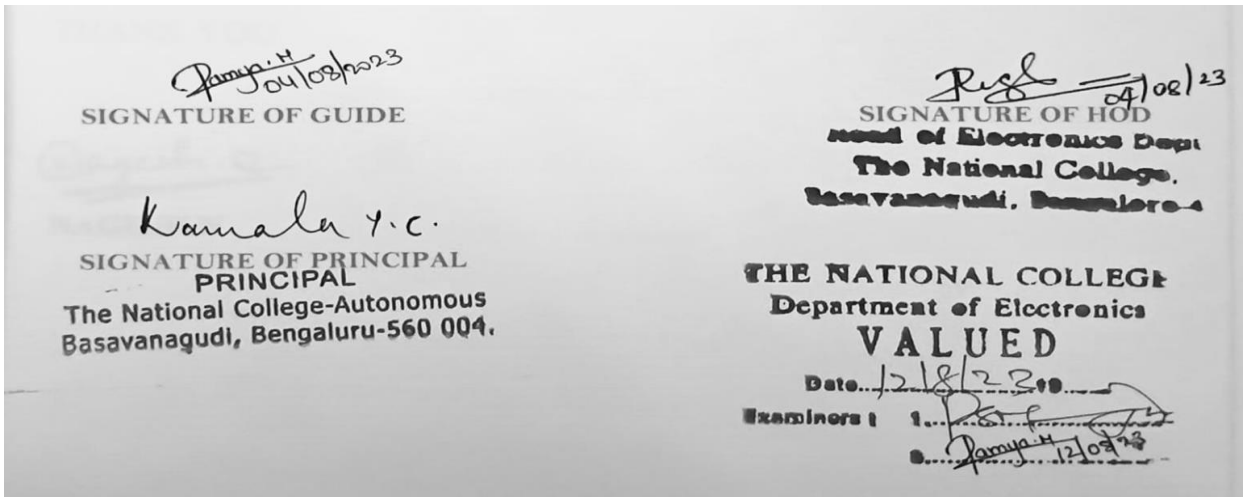
This is to certify that **Bharath A, Darshan Gowda N, Nagesh N and Mayur P Shetty of 6th Semester B.Sc.,**

has successfully completed the final year project work entitled:

"SMART DIGITAL LOCKER"

In partial fulfilment of the requirement for the award of marks in Bachelor of Science

in Electronics by the department during the academic year 2022 -2023.:



ACKNOWLEDGEMENT

Hard work and determination are the two key factors which lead to success. When these factors are backed by motivation under proper guidance, the success is ensuring with destiny.

The purpose of conducting projects will enhance our skill sets in applied science. This project as displayed as the moving message that "Life in science experiment is a mystery that creates history so we need to experience every moment “, that's the quality of experience we gained in this project.

We would like to express deep sense of gratitude to Dr . Y.C. Kamala, Principal, The National College Basavanagudi. Mrs. Poornima Hedge, Assoc. Professor & HOD of Electronics, Mrs. Ramya.M, Asst. Professor in Electronics, for their kind co-operation, assistance and encouragement in carrying out this project successfully.

We thank all the teaching and non-teaching staffs of the department for their kind assistance.

THANK YOU.

BHARATH.A **NAGESH.N**

DARSHAN GOWDA.N **MAYUR P SHEETY**

ABSTRACT:

Our Project aims to emphasize the security measures involved in the traditional lockers by implementing modern technologies which allows the users to conveniently access these lockers.

With the rapidly evolving technology our project provides a way to access the Locker using modern devices such as Smart phone.

The Project's methodology involves providing security through various mechanisms which allows the user to have an assured authorization.

Throughout the Project, we encountered certain challenges, such as loss sensitive Electronic Components due to our inadequate knowledge. The lessons learned from

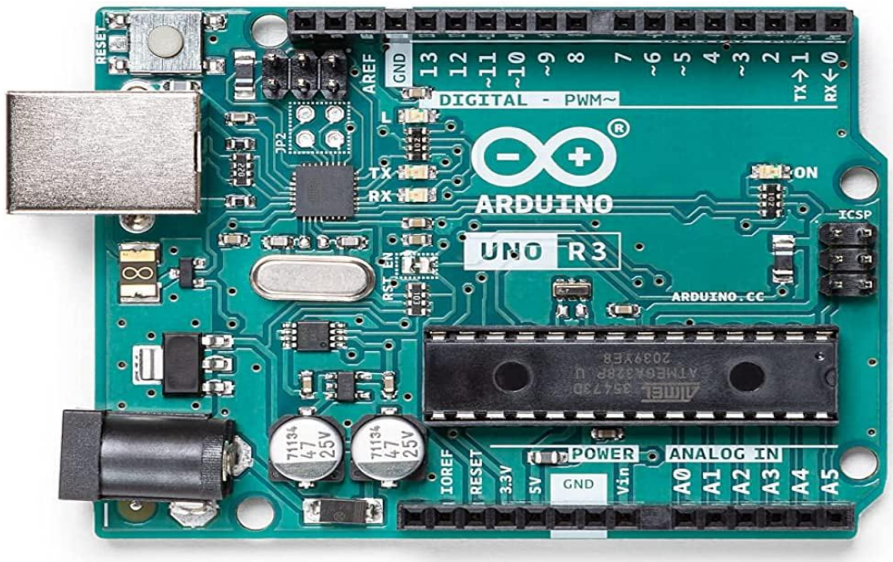
These challenges have further enriched our understanding of the subject and improved the projects overall quality

In conclusion, our project tries to imply the modern methodologies to enhance the security of the Locker making it a Smart Digital Locker.

ARDUINO UNO R3:

Arduino is an open-source prototyping platform in electronics based on easy-to-use hardware and software. Subtly speaking, Arduino is a microcontroller based prototyping board which can be used in developing digital devices that can read inputs like finger on a button, touch on a screen, light on a sensor etc. and turning it in to output like switching on an LED, rotating a motor, playing songs through a speaker etc.

In this Project, we will be utilizing the Arduino UNO board. Arduino UNO is a basic and inexpensive Arduino board and is the most popular of all the Arduino boards with a market share of over 50%. Arduino UNO is considered to be the best prototyping board for beginners in electronics and coding.



CONTENTS:

INTRODUCTION:

1.1 Introduction:

Lockers have been existing from Ancient Civilizations and have evolved along with us through time. With the Rise of urbanization and industrialization and the incorporation of valuable goods by the people the need to secure these valuables was an essential part, thus Lockers were invented to promise the security of these valuables. This project explores the possible ways to provide the security to the present conventional Lockers by utilizing the modern technical methods, Hence the name Digital Locker.

1.2 Objective:

To enhance the security of conventional Lockers using modern technological methods such as Arduino uno and Sim900A GSM Module. Thereby provide users a convenient way to access these lockers using their Smart phones.

COMPONENTS AND THEIR ROLES:

1. ARDUINO:

INTRODUCTION:

Arduino is an open-source prototyping platform in electronics based on easy-to-use hardware and software. Subtly speaking, Arduino is a microcontroller based prototyping board which can be used in developing digital devices that can read inputs like finger on a button, touch on a screen, light on a sensor etc. and turning it in to output like switching on an LED, rotating a motor, playing songs through a speaker etc.

The Arduino board can be programmed to do anything by simply programming the microcontroller on board using a set of instructions for which, the Arduino board consists of a USB plug to communicate with your computer and a bunch of connection sockets that can be wired to external devices like motors, LEDs etc.

Arduino is based on open source electronics project i.e., all the design specifications, schematics, software are available openly to all the users. Hence, Arduino boards can buy from

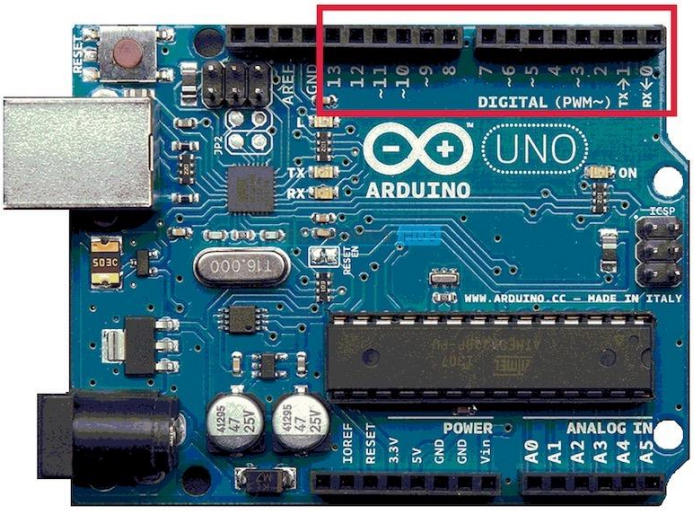
vendors as they are commercially available or else you can make your own board by if you wish i.e., you can download the schematic from Arduino's official website, buy all the components as per the design specification, assemble all the components, and make your own board

Hardware and Software:

Arduino boards are generally based on microcontrollers from Atmel Corporation like 8-, 16- or 32-bit AVR architecture-based microcontrollers. The important feature of the Arduino boards is the standard connectors. Using these connectors, we can connect the Arduino board to other devices like LEDs or add-on modules called Shields. The Arduino boards also consists of on-board voltage regulator and crystal oscillator. They also consist of USB to serial adapter using which the Arduino board can be programmed using USB connection. In order to program the Arduino board, we need to use IDE provided by Arduino. The Arduino IDE is based on Processing programming language and supports C and C++.

Arduino UNO:

In this Project, we will be utilizing the Arduino UNO board. Arduino UNO is a basic and inexpensive Arduino board and is the most popular of all the Arduino boards with a market share of over 50%. Arduino UNO is considered to be the best prototyping board for beginners in electronics and coding. UNO is based on ATmega328P microcontroller. There are two variants of the Arduino UNO: one which consists of through – hole microcontroller connection and other with surface mount type. Through-hole model will be beneficial as we can take the chip out in case of any problem and swap in with a new one. Arduino UNO comes with different features and capabilities. As mentioned earlier, the microcontroller used in UNO is ATmega328P, which is an 8-bit microcontroller based on the AVR architecture. UNO has 14 digital input – output (I/O) pins which can be used as either input or output by connecting them with different external devices and components. Out of these 14 pins, 6 pins are capable of producing PWM signal. All the digital pins operate at 5V and can output a current of 20mA.

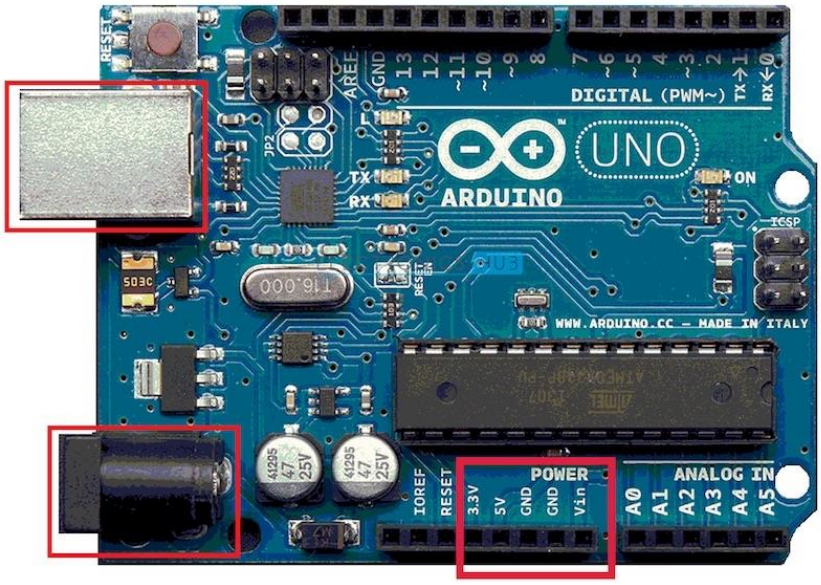


Some of the digital I/O pins have special functions which are describe below.

- Pins 0 and 1 are used for serial communication. They are used to receive and transmit serial data which can be used in several ways like programming the Arduino board and communicating with the user through serial monitor.
- Pins 2 and 3 are used for external interrupts. An external event can be triggered using these pins by detecting low value, change in value or falling or rising edge on a signal.
- As mentioned earlier, 6 of the 14 digital I/O Pins i.e., 3, 5, 6, 9, 10, and 11 can provide 8-bit PWM output.
- Pins 10, 11, 12 and 13 (SS, MOSI, MISO AND SCK respectively) are used for SPI communication.
- Pin 13 has a built-in LED connected to it. When the pin is HIGH, the LED is turned on and when the pin is LOW, it is turned off.

Arduino Uno has 6 analog input pins which can provide 10 bits of resolution i.e., 1024 different values. The analog pins on the Arduino UNO are labelled A0 to A5.

By default, all the analog pins can measure from ground to 5V. Arduino UNO has a feature, where it is possible to change the upper end of the range by using the AREF pin but the value should be less than 5V. Additionally, some analog pins have specialized functionality. Pins A4 and A5 are used for communication. There are different ways in which we can power the Arduino UNO board. The USB cable, which is used to program the microcontroller, can be used as a source of power.



There is a power jack, using which an external regulated power supply in the range of 7V – 12V can be supplied. Additionally, the power can also be supplied from a battery through the VIN pin. The UNO board has on-board voltage regulators for 5V and 3.3V, which can be used as power supply for small external devices like LEDs.This is a brief introduction to Arduino and Arduino UNO board. In the next tutorial, we'll see the installation and setup of the Arduino IDE.

ROLE OF ARDUINO UNO:

Arduino is the brain of our project which is capable of controlling and coordinating all other electronic components, the programming part of the Arduino Uno R3 is accomplished using Arduino integrated Development Environment (IDE).

2. SIM900A:

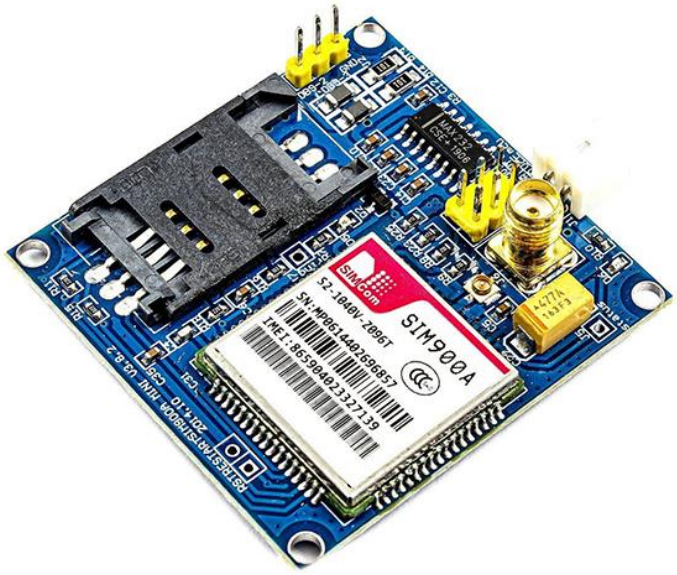
SIM900A Modem can work with any GSM network operator SIM card just like a mobile phone with its own unique phone number.

SIM900A GSM/GPRS modem is plug and play modem with RS232 serial communication supported. Hence Advantage of using this modem will be that its RS232 port can be used to communicate and develop embedded applications.

Applications like SMS Control, data transfer, remote control and logging can be developed. SIM900 modem supports features like voice call, SMS, Data/Fax, GPRS etc.

SIM900A modem uses AT commands to work with supported features.

Note that to be connected to a cellular network, the modem requires a SIM card provided by a network provider.



Power Requirement:

This board requires external power supply of ~12V and can draw up to ~2A of current at its peak.

Indicators:

It has two LED indicators as, ON: It shows that the Modem is getting powered and is switched on.

NET: This network LED blinks when the modem is communicating with the radio network.

Network LED:

When modem is powered up, network LED blink every second and after network registration it will start to blink after every 3 seconds. This shows that the modem is registered with the network.

To test modem, connect board serially to PC and send "ATE0" or "AT" through serial terminal. If "OK" response is received from the modem, then it means all is well

SIM900A is a GSM module that function like phone. It can send a message, call a phone number and use GPRS to send data. Here's the simple feature taken from its datasheet

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- Dual-Band 900/ 1900 MHz
- GPRS multi-slot class 10/8GPRS mobile station class B
- Compliant to GSM phase 2/2+Class 4 (2 W @850/ 900 MHz)
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- Low power consumption: 1.5mA (sleep mode)
- Operation temperature: -40°C to +85 °C

We just need connect 4 wires to SIM900A module, that is power connection (VCC and GND). And Serial communication (RX-TX). Because we use Arduino UNO which is use 5V operating voltage and has 5V logic level (TTL). So, we need to connect Arduino to 5RX and 5TX like in pinout picture below. Plug in sim card into sim card cartridge.

SIM900A and Arduino wiring connection

Wire SIM900A module to Arduino UNO like this:

Arduino -- SIM900A 5V -- VCC GND -- GND 10 -- TX 11 -- RX4

If you already connect the module with Arduino like in picture above, then this is the time to test communication between module and Arduino, to make sure that Arduino can give command to SIM900A module. Don't forget to double check the wiring connection.

In this test, we will use AT Command to communicate with SIM900A module. If we send command "AT" the module should reply "OK". And if that happen, this means the connection is successful

ROLE OF SIM900A:

The SIM900A gsm module is connected to the Arduino through I2C communication protocol which generates a onetime password for authorisation through SMS to the user registered mobile number.

3. LCD and KEYPAD:

The allows you to control LCD displays that are compatible with the Hitachi HD44780 driver. There are many of them out there, and you can usually tell them by the 16-pin interface.

Output of the sketch on a 16x2 LCD

The LCDs have a parallel interface, meaning that the microcontroller has to manipulate several interface pins at once to control the display. The interface consists of the following pins:

- A **register select (RS) pin** that controls where in the LCD's memory you're writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.
- A **Read/Write (R/W) pin** that selects reading mode or writing mode
- An **Enable pin** that enables writing to the registers
- 8 **data pins (D0 -D7)**. The states of these pins (high or low) are the bits that you're writing to a register when you write, or the values you're reading when you read.

There's also a **display contrast pin (Vo)**, **power supply pins (+5V and GND)** and **LED Backlight (Bklt+ and BKlt-)** pins that you can use to power the LCD, control the display contrast, and turn on and off the LED backlight, respectively.

The process of controlling the display involves putting the data that form the image of what you want to display into the data registers, then putting instructions in the instruction register. The Liquid Crystal Library simplifies this for you so you don't need to know the low-level instructions.

The Hitachi-compatible LCDs can be controlled in two modes: 4-bit or 8-bit. The 4-bit mode requires seven I/O pins from the Arduino, while the 8-bit mode requires 11 pins. For displaying text on the screen, you can do most everything in 4-bit mode, so example shows how to control a 16x2 LCD in 4-bit mode.

Hardware Required:

- Arduino Board
- LCD Screen (compatible with Hitachi HD44780 driver)
- pin headers to solder to the LCD display pins
- 10k ohm potentiometer
- 220-ohm resistor
- hook-up wires
- breadboard

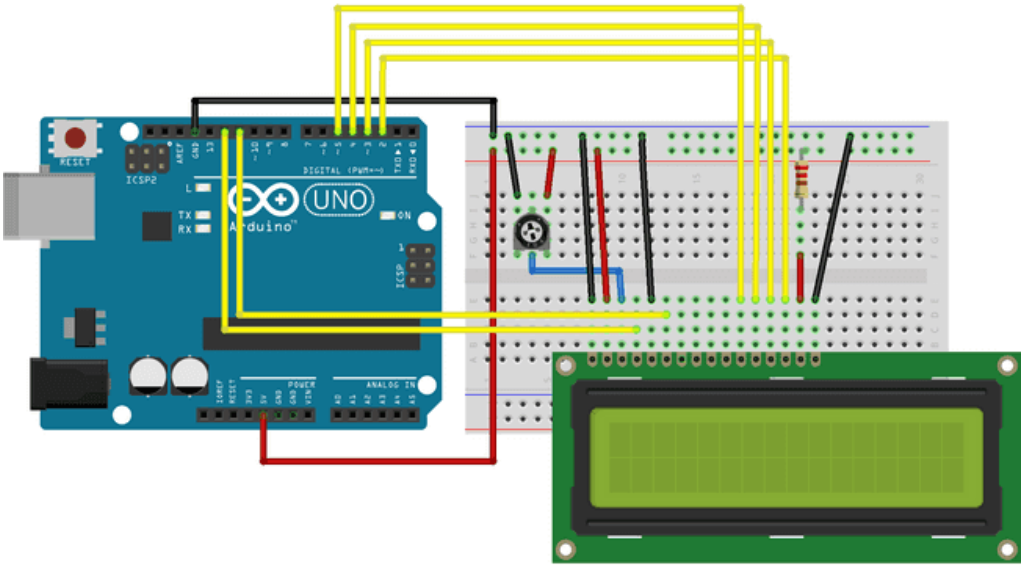
Circuit:

Note that this circuit was originally designed for the Arduino UNO. As the Arduino is communicating with the display using SPI, pin 11 & 12 will change depending on what board you are using. For example, on a MKR WIFI 1010, the SPI bus is attached to pin 8 & 11.
Before wiring the LCD screen to your Arduino board we suggest to solder a pin header strip to the 14 (or 16) pin count connector of the LCD screen, as you can see in the image further up.

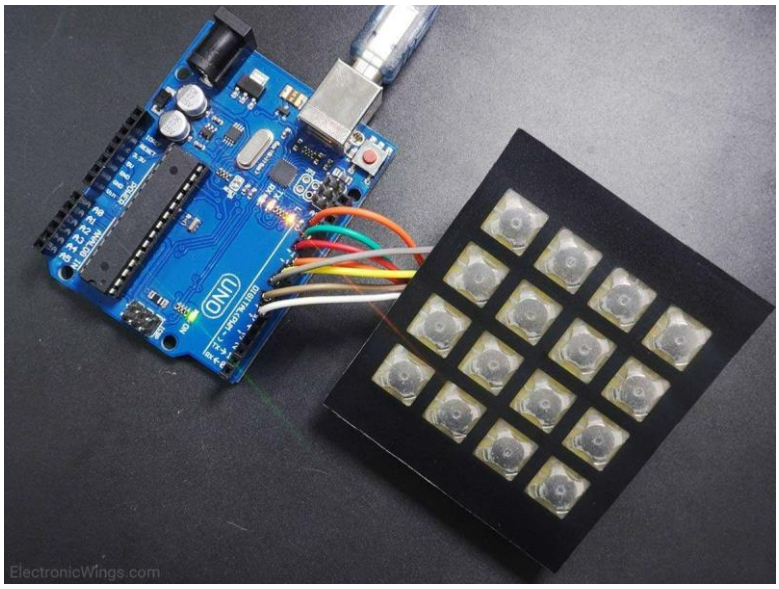
To wire your LCD screen to your board, connect the following pins:

- LCD RS pin to digital pin 12
- LCD Enable pin to digital pin 11
- LCD D4 pin to digital pin 5
- LCD D5 pin to digital pin 4
- LCD D6 pin to digital pin 3
- LCD D7 pin to digital pin 2
- LCD R/W pin to GND
- LCD VSS pin to GND
- LCD VCC pin to 5V
- LCD LED+ to 5V through a 220-ohm resistor
- LCD LED- to GND

Additionally, wire a 10k potentiometer to +5V and GND, with its wiper (output) to LCD screens VO pin (pin3).



4x4 Keypad Interfacing with Arduino UNO



Overview of 4x4 Matrix Keypad

4x4 Keypad

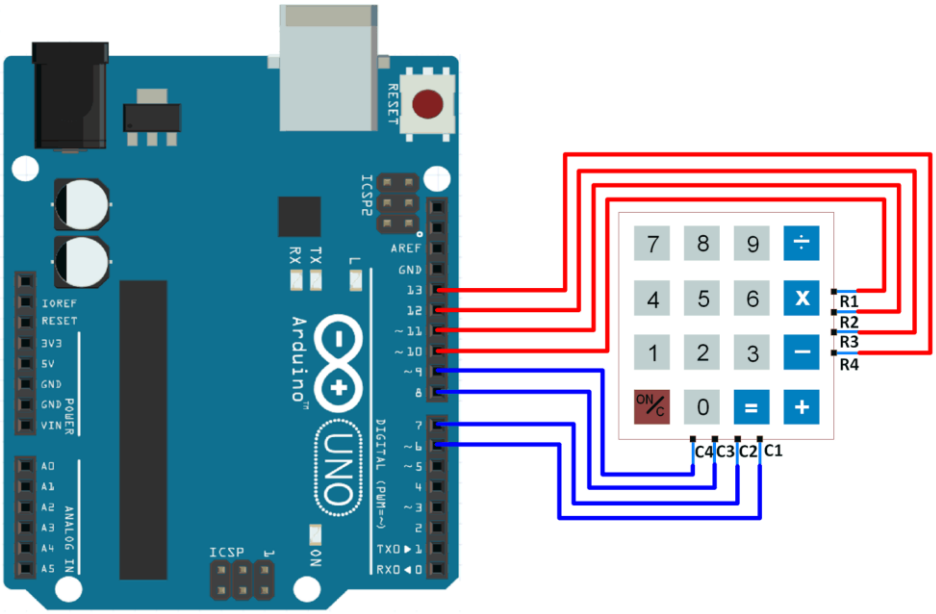
Keypad is used as an input device to read the key pressed by the user and to process it.

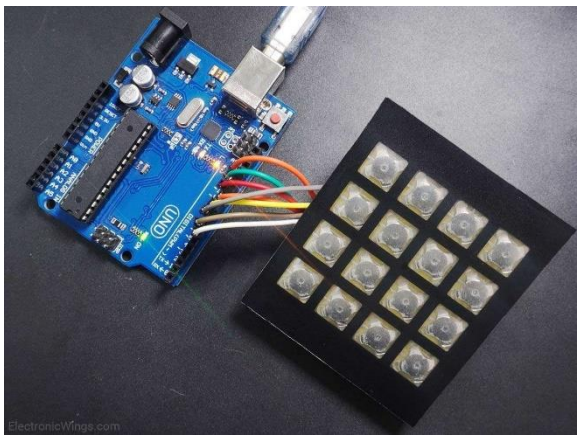
4x4 keypad consists of 4 rows and 4 columns. Switches are placed between the rows and columns.

A key press establishes a connection between the corresponding row and column, between which the switch is placed.

For more information about keypad and how to use it, refer the topic **4x4 Keypad** in the sensors and modules section.

UNO Connection Diagram of 4x4 Keypad with Arduino Interfacing 4x4 Keypad with Arduino UNO





Read the 4x4 Keypad using Arduino Uno

Reading the key pressed on the 4x4 keypad and displaying it on the serial terminal of Arduino.

Word of Caution:

In the example sketch Custom Keypad included in the Keypad library, digital pins 0 and 1 are used for connecting rows. This will result in those rows not working for some people.

The Arduino UNO board has digital pins 0 and 1 connected to the Tx and Rx pins which are used for serial communication. Since the sketch uses serial communication to display the key pressed on serial terminal, this will cause erroneous behaviour.

To avoid this, use pins other than pins 0 and 1. You can refer the sketch given below to get an idea about how to do this.

Note:

byte row Pins [ROWS] = {R1, R2, R3, R4}; /* connect to the row pinouts of the keypad */

byte colPins [COLS] = {C1, C2, C3, C4}; /* connect to the column pinouts of the keypad */If you do not connect the pins according to this function, the key press identification will not give results according to the keypad you have defined.

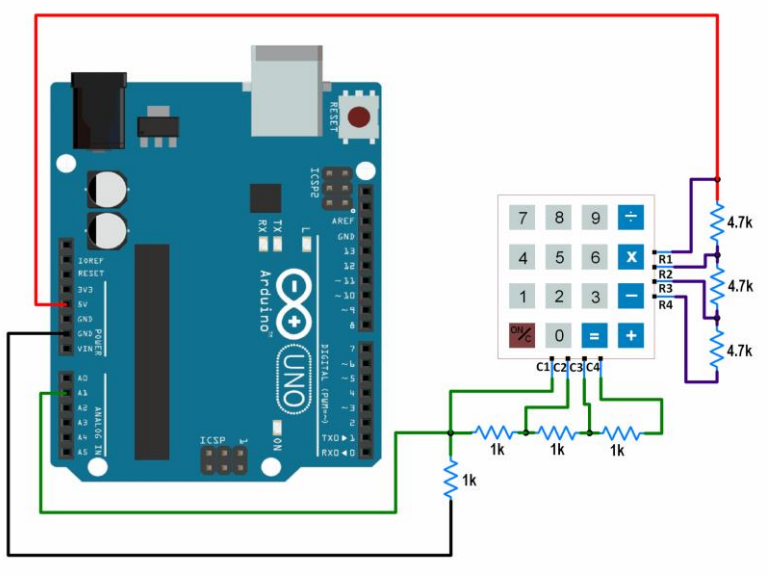
1-Wire interfacing of Keypad With Arduino

The above shown interfacing consumes 8 GPIO pins of the Arduino board. By using the 1-wire interfacing method for a keypad, we can achieve the same result as above using just 1 GPIO pin.

This method makes use of simple voltage divider concept to generate different voltages for each key pressed.

Let’s see how to interface keypad to Arduino using a single GPIO pin.

Connection Diagram of 1-Wire Keypad Interface with Arduino



1-Wire Interfacing of Keypad with Arduino

ROLE OF LCD and KEYPAD:

Lcd and keypad is used as a user interface to verify the Password and to accept the One Time Password generated by the SIM900 gsm module.

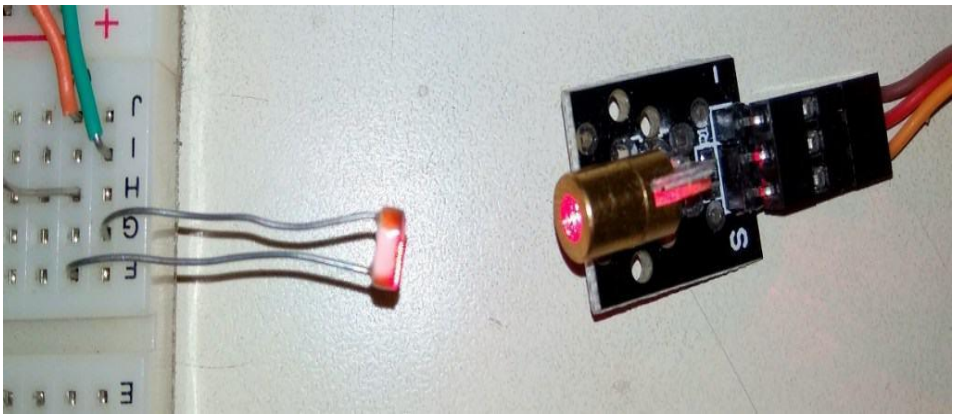
4.LASER and LDR:

Laser Diode Module KY-008:

Laser Transmitter module KY-008 for Arduino emits a dot-shaped, red laser beam. The KY-008 Laser transmitter module consists of a 650nm red laser diode head and a resistor. Handle with caution, do not look directly into the laser head.

Working of the Laser Light Security System Using Arduino:

The project basically works on the principle of interruption. If by any means the LASER light is interrupted the alarm will start unless it is reset with push-button. The laser is a concentrated light source that puts out a straight beam of light of a single colour.



The LDR is sensitive to light and puts out a voltage when the laser light hits it. When the laser beam is interrupted and can’t reach LDR, its voltage output changes, and eventually the alarm will ring.

ROLE OF LASER AND LDR:

Laser system remains active until user authorisation is successful which is sensed by the LD. The LDR senses the change in the intensity of the Laser light which has a fixed resistance value as its output, however during the change of intensity of Laser light the resistance value varies, this variation causes the siren buzzer and the Led’s to activate.

5.LED AND PIEZO BUZZER:

LEDs are small, powerful lights that are used in many different applications. To start, we will work on blinking an LED, the Hello World of microcontrollers. It is as simple as turning a light on and off. Establishing this important baseline will give you a solid foundation as we work towards experiments that are more complex.

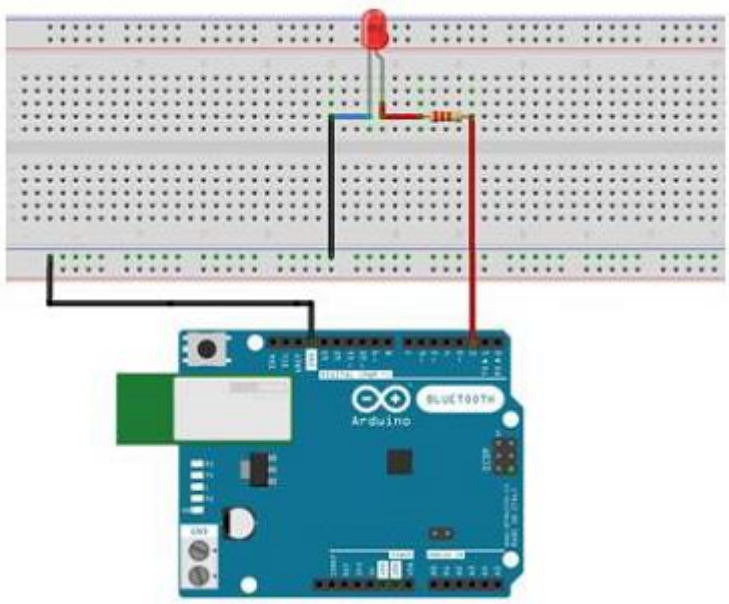
Components Required:

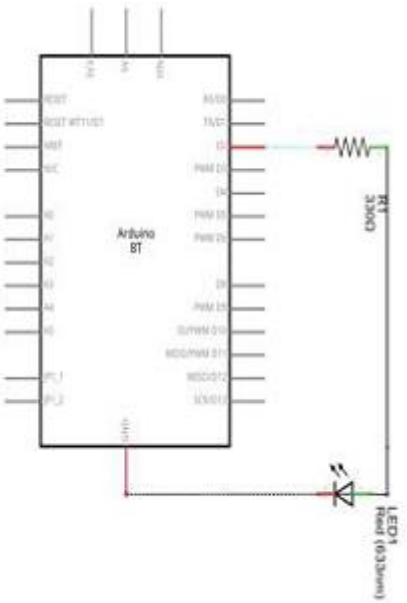
You will need the following components:

- 1 × Breadboard
- 1 × Arduino Uno R3
- 1 × LED
- 1 × 330Ω Resistor
- 2 × Jumper

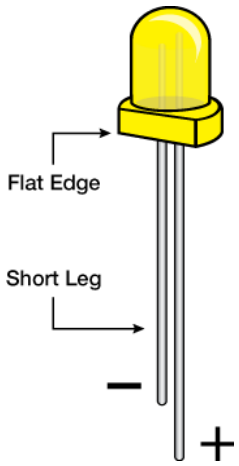
Procedure:

Follow the circuit diagram and hook up the components on the breadboard as shown in the image given below.





Note − To find out the polarity of an LED, look at it closely. The shorter of the two legs, towards the flat edge of the bulb indicates the negative terminal.



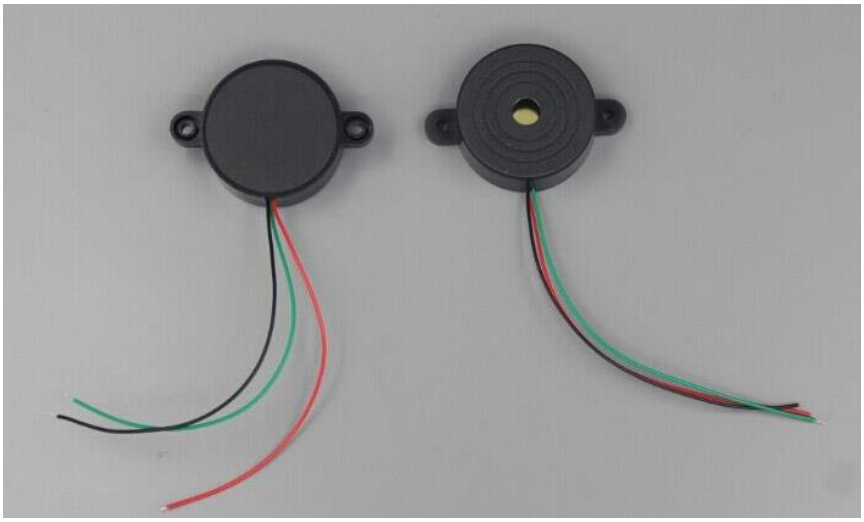
Components like resistors need to have their terminals bent into 90° angles in order to fit the breadboard sockets properly. You can also cut the terminals shorter.

6.THE BUZZER:

The module can be connected to digital outputs to emit a tone when the output is digital and a variety of tones and effects using an analog pulse-width modulation output.

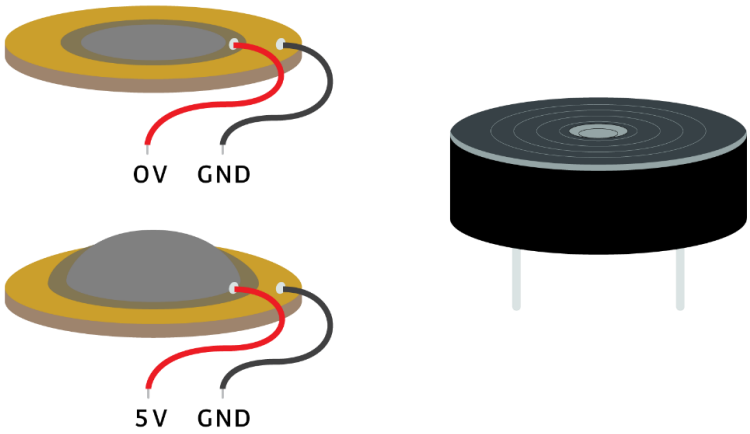
Introduction:

The piezo, also known as the buzzer, is a component that is used for generating sound. It is a digital component that can be connected to digital outputs, and emits a tone when the output is HIGH. Alternatively, it can be connected to an analog pulse-width modulation output to generate various tones and effects. The **Grove Buzzer** operates at both 3.3V and 5V with a sound output of 85 decibels. This module can be used to provide sound feedback to your application just like the click sound of a button on a digital watch.



Working principle of the buzzer:

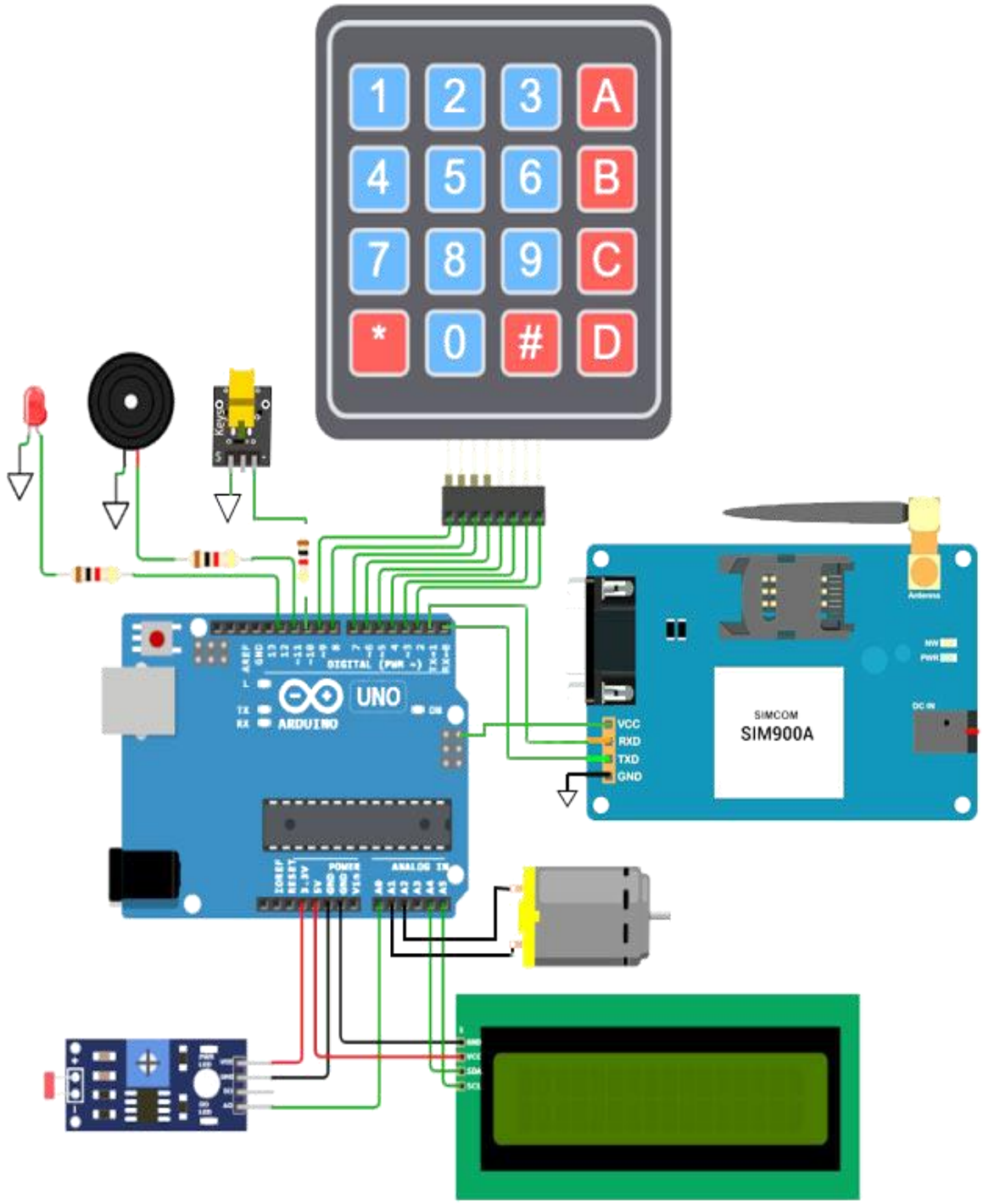
The buzzer is a sounding device that can convert audio signals into sound signals. It is usually powered by DC voltage. It is widely used in alarms, computers, printers and other electronic products as sound devices. It is mainly divided into piezoelectric buzzer and electromagnetic buzzer, represented by the letter "H" or "HA" in the circuit. According to different designs and uses, the buzzer can emit various sounds such as music, siren, buzzer, alarm, and electric bell.



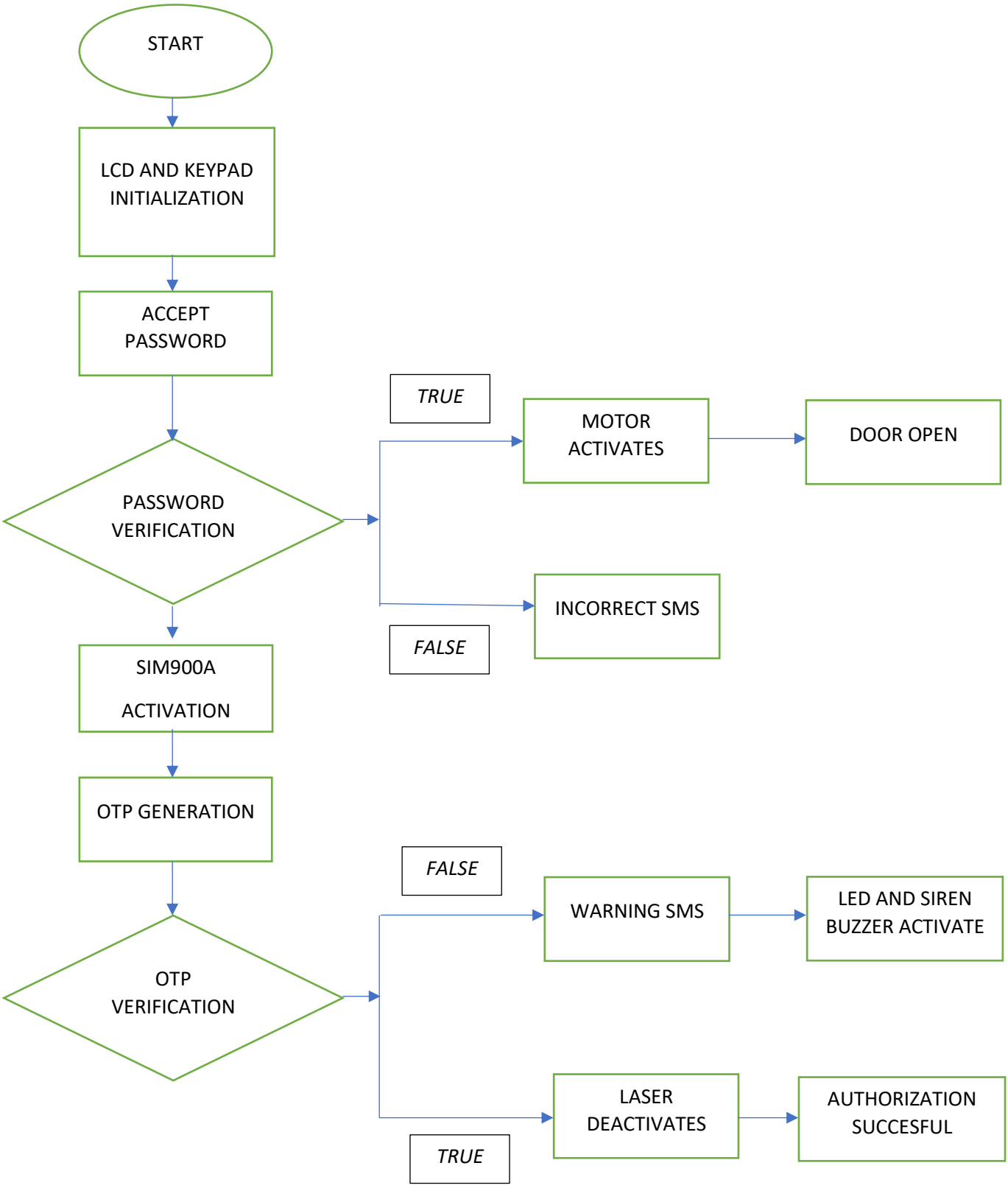
ROLE OF LED AND PIEZO BUZZER:

The Led and Piezo Buzzer activates upon interference of hand or any other object to the Laser and LDR system.

CIRCUIT DESIGN:



FLOW CHART:


START
LCD AND KEYPAD INITIALIZATION
ACCEPT PASSWORD
PASSWORD VERIFICATION
TRUE
MOTOR ACTIVATES
DOOR OPEN
FALSE
INCORRECT SMS
SIM900A ACTIVATION
OTP GENERATION
OTP VERIFICATION
FALSE
WARNING SMS
LED AND SIREN BUZZER ACTIVATE
TRUE
LASER DEACTIVATES
AUTHORIZATION SUCCESFUL

WORKING:

- The Entire Smart Digital Locker is safeguarded using both Active and Passive electronic components as shown in the Circuit Diagram above.
- The Working of all the devices is controlled and coordinated by the Arduino Uno R3.
- Sim900A is used to communicate with the register's user.
- LCD and Keypad are used as User Interface for Authorization.
- Lasers and LDR assure the Security of the Valuables inside the Locker.
- The Locker is locked by a Door with a Latch. To unlock the Door, Latch the User has to Enter the Password for Verification.
- The User enters the Password through LCD and Keypad which unlocks the Latch using DC motor and mechanical gears.
- Once the verification is successful and the door is unlatched. An OTP will be sent to the user for next step of Authorization.
- If the user fails to Enter the OTP and tries to steal the valuable thing inside the Locker the Siren Beep and Lights turn on automatically and a warning message will be sent to the user.
- If the OTP is Entered correctly the Laser deactivates and the user is free to access the object or valuable thing inside the Locker.

SOFTWARE DESCRIPTION:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <SoftwareSerial.h>
SoftwareSerial SIM900A(0,1);  // 0=TX of SIM900A,1=RX of SIM900A



#define Password_Length 5
int OTP_array[5];
int OTP;
char OTP_array1[5];
char Data [Password Length];
char Master[Password_Length] = "B#57";
int Data1[Password Length];
byte data count = 0, master count = 0;
bool Pass_is_good;
char customKey;
int intensity=150; // Value reduces when Laser is Interrupted

const byte ROWS = 4;
const byte COLS = 4;

char hexaKeys [ROWS][COLS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
```

```
byte row Pins [ROWS] = {9,8,7,6};
byte colPins [COLS] = {5,4,3,2};

Keypad customKeypad = Keypad(makeKeymap(hexaKeys), row Pins, colPins, ROWS, COLS);

LiquidCrystal_I2C lcd(0x27,16,2);

void setup ()
{

  SIM900A.begin(9600);   // Setting the baud rate of GSM Module
  Serial.begin(9600);    // Setting the baud rate of Serial Monitor
  lcd. clear();
  lcd. backlight ();
  pin Mode (A1, OUTPUT);
  pin Mode (A2, OUTPUT); // Motor
  pin Mode (11, OUTPUT); // Buzzer
  pin Mode (12, OUTPUT); // Led
  digital Write (10, HIGH); // Laser

 }

void loop () {

  int a=analogRead(A0); // LDR Analog value
  int b=map(a,0,1023,0,255);
  Serial.println(b);

   if(b<intensity)
  {
```

```
 Serial.println(b);
 SendMessage1();
}
else
{
  }
lcd.setCursor(1,0);
lcd.print ("Enter Password");
customKey = customKeypad.getKey();
if(customKey){
  Serial.println(customKey);
  Data[data_count] = customKey;
  Data1[data_count]=customKey;
  lcd.setCursor(data_count+5,1);
  lcd.print ("*******");
  data_count++;
  }

if(data_count == Password_Length-1)
{
  lcd.clear();
  if(!strcmp(Data, Master))
  {
    SendMessage2();
    lcd.print("Correct");
    digitalWrite(A1,HIGH);
    digitalWrite(A2,LOW);
    delay(5000);
    digitalWrite(A1,LOW);
    digitalWrite(A2,LOW);//MOTOR activates
  }
```

```
    else if(!strcmp(Data,OTP_array1)) //OTP Verification
 {
  Serial.println("Verification Succesfull");
  lcd.print("Verification Succesfull");
  delay(5000);
  digitalWrite(10,LOW);//Lasers is off
 }
      else {
     lcd.setCursor(1,6);
     lcd.print("Incorrect");
     SendMessage3();// Warning Message is Sent
     delay(5000);
     }
     lcd.clear();
   clearData();
  }
}
void clearData(){
  while(data_count !=0){
    Data[data_count--] = 0;
  }
  return;
}

void SendMessage1() //Laser interruption Warning!!!
{
  SIM900A.println("AT+CMGF=1");  //Text Mode initialisation
  delay(1000);
  SIM900A.println("AT+CMGS=\"+916363407487\"\r"); // Receiver's Mobile Number
  delay(1000);
  SIM900A.println("Warning Laser Interrupted");  // Messsage content
```

```
  delay(1000);
  SIM900A.println((char)26);  //   delay(1000);
  Serial.println ("Message sent succesfully");
 }
void SendMessage2() //OTP sending...
{
  OTP=rnd();
  intToArray(OTP,OTP_array,5);
  char OTP_array1[sizeof(OTP_array)];
  for (int i = 0; i < sizeof(OTP_array); i++)
  {
  OTP_array1[i] = char(OTP_array[i]);
  }
  Serial.println(OTP);
  SIM900A.println("AT+CMGF=1");  // Text Mode initialisation
  delay(1000);
  SIM900A.println("AT+CMGS=\"+916363407487\"\r"); // Receiver's Mobile Number
  delay(1000);
  SIM900A.println(OTP);// Messsage content
  delay(1000);
  SIM900A.println((char)26); // delay(1000);
  Serial.println ("Message sent succesfully");
 }
void SendMessage3()
{
  SIM900A.println("AT+CMGF=1");    //Text Mode initialisation to Warn the User
  delay(1000);
  SIM900A.println("AT+CMGS=\"+916363407487\"\r"); // Receiver's Mobile Number
  delay(1000);
  SIM900A.println("Incorrect Password Entered"); // Messsage content
  delay(1000);
```

```
  SIM900A.println((char)26); // delay(1000);
  Serial.println ("Message sent succesfully");
}
int rnd()
 {
  int b;
  b=random(1000,9999);
  return b;
 } //OTP generator Function

void intToArray(int number, int array[], int arraySize)
{
  for (int i = arraySize - 1; i >= 0; i--) {
    array[i] = number % 10;
    number /= 10;
  }} //Converting int to Array
```

CONCLUSION:

In conclusion, this project has been an exciting and insightful journey. Through meticulous research, thoughtful analysis, and creative problem-solving, we have achieved the objectives set at the beginning of the project.

Our findings have provided valuable information about advanced electronic systems, shedding light on Arduino and SIM900A gsm module. These discoveries have significant implications for advanced security and they open up new avenues for further exploration and research.

Throughout the project, we encountered challenges that tested our abilities to adapt and innovate. However, these challenges also provided valuable learning opportunities that helped us grow both personally and professionally.

In retrospect, we recognize that effective collaboration and communication played a vital role in the success of this project. The teamwork and support among the project members were integral to overcoming obstacles and achieving our goals.

While the project's outcomes are valuable, we also acknowledge that there are limitations to our work. Factors such as time constraints and resource availability may have impacted the scope and depth of our investigations. However, these limitations can serve as a foundation for future researchers to build upon and address in more comprehensive studies.

Overall, this project has been a gratifying experience, and we believe that the knowledge gained from this endeavour will contribute positively to the society assuring high security. We hope our findings will inspire others to delve further into this area of research and continue exploring the fascinating world of electronics and programming.

Lastly, we express our gratitude to all those who supported and guided us throughout this project, including our mentors, peers, and the resources provided by THE NATIONAL COLLEGE BASAVANGUDDI. Their contributions have been invaluable.

As we conclude this project, we are filled with a sense of accomplishment and anticipation for what lies ahead. We are confident that the skills honed and lessons learned during this undertaking will serve us well in future endeavours.

Thank you.

REFERENCE:

- LASTMINTUEENGINEERS.COM
- ELECTRONICWINGS.COMS
- ARDUINO.COM
- GOOGLE.COM
- CHATGPT
- QUISURE.COM

REFFERED BOOKS:

- ARDUINO WORKSHOP - AUTHOR: JOHN BOXALL
- SAMS TEACH YOURSELF

  ARDUINO PROGRAMMING IN 24 HOURS - AUTHOR: RICHARD BLUM

DATASHEETS:

1) ARDUINO
2) SIM900A
3) LCD
4) KEYPAD