ADVANCE LOCKER SYSTEM

A PROJECT REPORT

Submitted by

TARBADA VIVEK B
BUTANI UMANG R
SOLANI HARSH B

In fulfillment for the submission of Project I

Of

BACHELOR OF ENGINEERING

In

ELECTRONICS & COMMUNICATION

ATMIYA INSTITUTE OF TECHNOLOGY AND SCIENCE
RAJKOT-360005

GUJARAT TECHNOLOGICAL UNIVERSITY

APRIL 2018

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2017-2018

Date of certificate generation : 11 April 2018 (03:33:52)

This is to certify that, ***Butani Umangkumar Rameshbhai*** (Enrolment Number - 140030111005) working on project entitled with ***Advance Locker System*** from ***Electronics & Communication Engineering*** department of ***ATMIYA INSTITUTE OF TECHNOLOGY & SCIENCE, RAJKOT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Butani Umangkumar Rameshbhai

Name of Guide : Mr.Mehta Tushar J

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2017-2018

Date of certificate generation : 11 April 2018 (03:38:28)

This is to certify that, ***Tarbada Vivekkumar Bhagubhai*** (Enrolment Number - 140030111024) working on project entitled with ***Advance Locker System*** from ***Electronics & Communication Engineering*** department of ***ATMIYA INSTITUTE OF TECHNOLOGY & SCIENCE, RAJKOT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Tarbada Vivekkumar Bhagubhai

Name of Guide : Mr.Mehta Tushar J

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2017-2018

Date of certificate generation : 08 May 2018 (06:22:37)

This is to certify that, ***Solani Harsh Bakulbhai*** (Enrolment Number - 140030111023) working on project entitled with ***Advance Locker System*** from ***Electronics & Communication Engineering*** department of ***ATMIYA INSTITUTE OF TECHNOLOGY & SCIENCE, RAJKOT*** had submitted following details at online project portal.

Periodic Progress Reports (PPR)	**Completed**
Business Model Canvas (Image)	**Completed**
Business Model Canvas (Report)	**Completed**
Patent Drafting Exercise (PDE)	**Completed**
Final Plagiarism Report	**Completed**
Final Project Report	**Completed**

Name of Student : Solani Harsh Bakulbhai Name of Guide : Mr.Mehta Tushar J

Signature of Student : _____ *Signature of Guide : _____

Disclaimer :
This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.

ATMIYA INSTITUTE OF TECHNOLOGY AND SCIENCE

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
2017-18

CERTIFICATE

Date:

This is to certify that the dissertation entitled "ADVANCE LOCKER SYSTEM**" has been carried out by** BUTANI UMANGKUMAR R.(140030111005)**under my guidance in fulfillment of the degree of Bachelor of Engineering in Electronics & Communication (8th semester) during the academic year 2017-18.**

Guide: Asst. Prof. Tushar J Mahta

Head of the Department
Dr. Ashish M Kothari

ATMIYA INSTITUTE OF TECHNOLOGY AND SCIENCE

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
2017-18

CERTIFICATE

Date:

This is to certify that the dissertation entitled "ADVANCE LOCKER SYSTEM**" has been carried out by** TARBADA VIVEKKUMAR B.(140030111024) **under my guidance in fulfillment of the degree of Bachelor of Engineering in Electronics & Communication (8th semester) during the academic year 2017-18.**

Guide: Asst. Prof. Tushar J Mahta

Head of the Department
Dr. Ashish M Kothari

ATMIYA INSTITUTE OF TECHNOLOGY AND SCIENCE

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING
2017-18

CERTIFICATE

Date:

This is to certify that the dissertation entitled "ADVANCE LOCKER SYSTEM**" has been carried out by** SOLANI HARSH(140030111023) **under my guidance in fulfillment of the degree of Bachelor of Engineering in Electronics & Communication (8th semester) during the academic year 2017-18.**

Guide: Asst. Prof. Tushar J Mahta

Head of the Department
Dr. Ashish M Kothari

ACKNOWLEDGEMENT

Any accomplishment requires the effort of many people and this work is not different. And it is our prime duty to acknowledge the person who directly or indirectly helped us during completion of this report. So we take opportunity to heartily thank our project in charge respected **Asst. Prof. TUSHAR J MEHTA** for his valuable guidance and touch of inspiration and motivation throughout the project work without whose help the work would not have been in the shape what it is.

We heartily thank our Head of Department **Dr. Ashish M Kothari** without whose motivation our project work would not have been in the shape what it is.

We are also thankful toothier relevant staff of department for their co-operation, help and guidance. We are also thankful to our co-colleagues for building good team work to make the success of project.

ABSTRACT

Locker is a place where we can put your thing or documents for a security purpose. Locker comes with their different configurations like simple combination lock, password protected also with finger print locker. For user of locker system we design a new system that can help user to make their locker more secure and also with different security features. In this project we design a locker with high security system based on fingerprint, GSM technology and OTP (One time password) which can be organized in banks, protected offices and homes. In this system will collect the biometric data of person for the lockers only authentic person can be recovered things or documents from the locker. We have implemented a locker security system based on fingerprint, GSM technology and OTP. Fingerprints are one of many forms of biometrics, used to identify persons and verify their identity and it is more secure form of data. This cannot easily copy. so we use that and after that locker send OTP to mobile which already in that system that 4 digit code come in mobile and use of 4x4 keypad OR alternate system is that send OTP via mobile in the system and if OTP is matched than system allow to take things or documents form locker.

Appendix 1

LIST OF FIGURES

Appendix 2

LIST OF TABLES

Appendix 3 LIST OF SYMBOLS, ABBREVIATIONS AND NOMENCLATURE

Symbol Name	Abbreviations
R	RESISTER
LCD	LIQUID CRYSTAL DISPLAY
V	VOLTAGE
GND	GROUND
R/S	READ/SET
W	WRITE
E	ENABLE
Rx	RECEIVER
Tx	TRANSMITTER
D0-D13	DIGITAL I/O
A0-A5	ANALOG I/Z

TABLE OF CONTENTS

TABLE OF CONTENTS

1. INTRODUCTION

1.1: Objective

AIM: An **ADVANCE LOCKER SYSTEM** is ideal for all sorts of environments which include home, office and business to provide a secure locker to the customer from market and make sure that its cost is efficient for all the people who can buy it and secure their precious thing in safe.

The objectives of the present invention are:

(1) AADHAAR card has unique identity for every person.

(2) To protect the digital password from hack.

(3) To use OTP password that change every attempt of open so no need to remember.

1.2: Motivation

In life of human we all have special things which love most and we need it to keep safe. So that Desire takes us to invention of locker. Locker is place where we can keep our thing as a safe place so in doing of that first come simple locker take place in order to make them safe use key to open and close it. So only authorized person can only open it or close it. But after that come that another one takes footprint of your key and they can handle it. So it's time to move from analog locker to digital locker because it can provide more security than analog system we simply can add here a code which we want to add here. So to batter security we make more dependency on digital locker. Digital lockers are safe come with simply 4 digit digital number code which only access by who knows code. That is very good technology and its use mostly. but in locker we put our most valuable things and to secure it we must do whatever it take place so inventions of advance locker never get stop and more and more advanced locker come in to picture so we can secure our thing very well. In addition of that technology research take one step ahead and put biometric security at that place. Which allow you to open locker using your fingerprint scan or by applying your eye retina scan person can access their locker.

In today's hi-tech world everyone has access to state of the art security systems. One of the latest trends in security systems is Digital safes locker. These digital safe lockers are Electronic safe locking systems which operate as per the signals received through the input key boards.

With the increase of crimes in the society, Safety and security have become a primary concern for all. It is advisable to have the cash, ornaments and other valuables under safe custody because burglars these days are very tech savvy and they have a lot of modern equipments with them. Burglars are now equipped with instruments and they can destroy most of the conventional safety locker systems. The enhanced security features of the digital safe lockers have made it very difficult for the thieves to operate a digital safe locker. The need for safe locker systems in homes, Offices, shops business establishments, banks, financial institutions, Petrol stations, Brokers, Hotels and Hospitals is increasingly felt in these days due to the increase in the security concerns.

They are tamper proof and provide many security options for the user. Digital Electronic keypad entry lock offers several advantages also as it is not only more reliable and strong but also easy to use. It works on keypad number entry and is based on the combination of locks. The Digital lockers automatically go into locking mode if left idle for more than a pre-programmed time period.

That all advanced technology are very complex and costly to do so we must need to make sure biometric work properly at time of open it. So we decide to make system that is very easy to use and also to make that happen we also take care of cost that is very high is become at normal place.

Instead of using a simple biometric system we develop our advanced system that give more secure your locker and you feel free to put your things in locker. We here introduce ADVANCED LOCKER SYSTEM that work on system of addhar card and also in based on your mobile system that mean it provide you more security that any other locker in market give you and also easy to use for any common man which has their own addhar card.

1.3: Application

- Bank
- Home
- Office
- Collage
- School

2. PROTOTYPE OVERVIWE

2.1: Block diagram

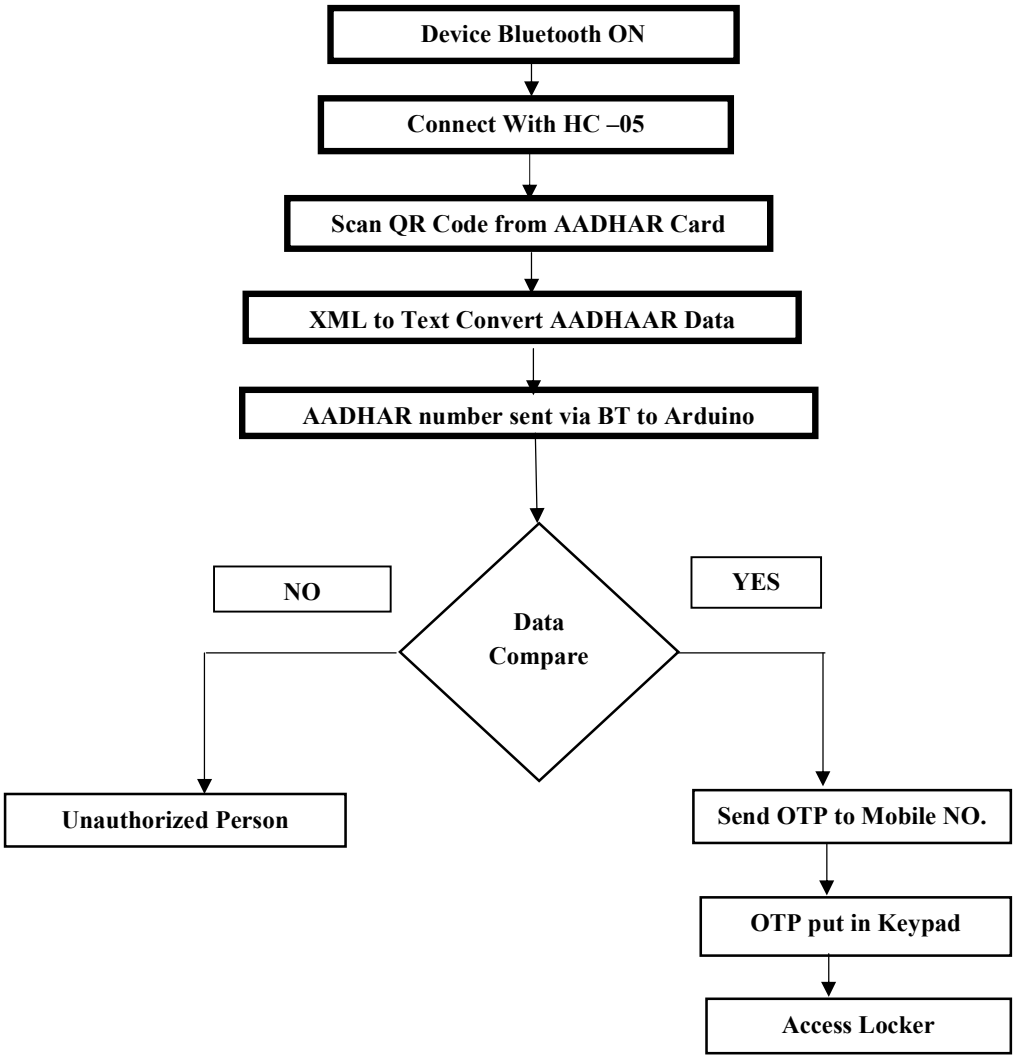

Fig2.1: Flow diagram of the invention.

3. Implementation of Prototype

3.1: Circuit Diagram

MAKING OF PRODUCT

Advance locker system is system that provides the advance security in locker as in form of hardware as some electronics components and also in form of software. As here as electronics we use controller as arduino and software as simple arduino IDE which provide simple C coding that use as in assembly C.

Coding in Arduino IDE is very easy because of their easy user interface and also nice GUI. Arduino IDE provide inbuilt library for different hardware that make it easy to make circuit configuration.

3.1.1 Interfacing 16x2 LCD and 4x4 matrix to Arduino.

Interfacing 16x2 LCD to Arduino.

To establish a good communication between human world and machine world, display units play an important role. And so they are an important part of embedded systems. Display units - big or small, work on the same basic principle. Besides complex display units like graphic displays and 3D displays, one must know working with simple displays like 16x1 and 16x2 units. The 16x1 display unit will have 16 characters and are in one line. The 16x2 LCD will have 32 characters in total 16in 1st line and another 16 in 2nd line. Here one must understand that in each character there are 5x10=50 pixels so to display one character all 50 pixels must work together. But we need not to worry about that because there is another controller (HD44780) in the display unit which does the job of controlling the pixels. (you can see it in LCD unit, it is the black eye at the back).

So, we use simple 16x2 LCD Display and arduino in that arduino has library for LCD displays so we can use that direct and can configure the display as we want.

RS pin of the LCD module is connected to digital pin A0 of the arduino. R/W pin of the LCD is grounded. Enable pin of the LCD module is connected to digital pin A1 of the arduino. In this project, the **LCD module and arduino are interfaced in the 4-bit mode**. This means only four of the digital input lines (DB4 to DB7) of the LCD are used. This method is very simple, requires less connections and you can almost utilize the full potential

of the LCD module. Digital lines DB4, DB5, DB6 and DB7 are interfaced to digital pins A2, A3, A4 and A5 of the Arduino. The 10K potentiometer is used for adjusting the contrast of the display. 560 ohm resistor R1 limits the current through the back light LED. The arduino can be powered through the external power jack provided on the board. +5V required in some other parts of the circuit can be tapped from the 5V source on the arduino board. The arduino can be also powered from the PC through the USB port.

To interface a LCD to the ARDUINO UNO, we need to know a few things.

1. #include <LiquidCrystal.h>
2. lcd.begin(16, 2);
3. LiquidCrystal lcd(A0,A1,A2,A3,A4,A5);
4. lcd.print ("hello, world!");

As by the above we only need to look at these four lines for establishing a communication between an ARDUINO and LCD.

First we need to enable the header file ('#include <LiquidCrystal.h>'), this header file has instructions written in it, which enables the user to interface an LCD to UNO in 4 bit mode without any fuzz. With this header file we need not have to send data to LCD bit by bit, this will all be taken care of and we don't have to write a program for sending data or a command to LCD bit by bit.

Second we need to tell the board which type of LCD we are using here. Here we are going to interface a 16x2 LCD to the UNO so we get 'lcd.begin(16, 2);'In this instruction we are going to tell the board where we connected the pins.

The pins which are connected need to be represented in order as "RS, En, D4, D5, D6, and D7". These pins are to be represented correctly. Since we have connected RS to PIN0 and so on as show in the circuit diagram, we represent the pin number to board as "LiquidCrystal lcd(0, 1, 8, 9, 10, 11);".

The data which needs to be displayed in LCD should be written as "lcd.print ("hello, world!");" With this command the LCD displays 'hello, world!'

Interfacing 4x4 matrix keypad To Arduino.

Now, we configure arduino and 4x4 matrixes that are also as simple LCD so it has 4x4 matrix library which is help to configure with it. That is basically works on the principal of PUSS to ON switch.

Pins 8, 7, 6, 5 on the keypad should be connected to **digital pins 0,1,2,3 on the Arduino** respectively.

Pins 4, 3, 2, 1 on the keypad should be connected to **digital pins 4,5,6,7 on the Arduino** respectively.

After that all configuration we have now two different system that is one is 16x2 LCD Display and second one is 4x4 matrix that two complete system make a GUI for user to operate a locker. In doing of that we make a complex code in arduino IDE and compile in Arduino controller.

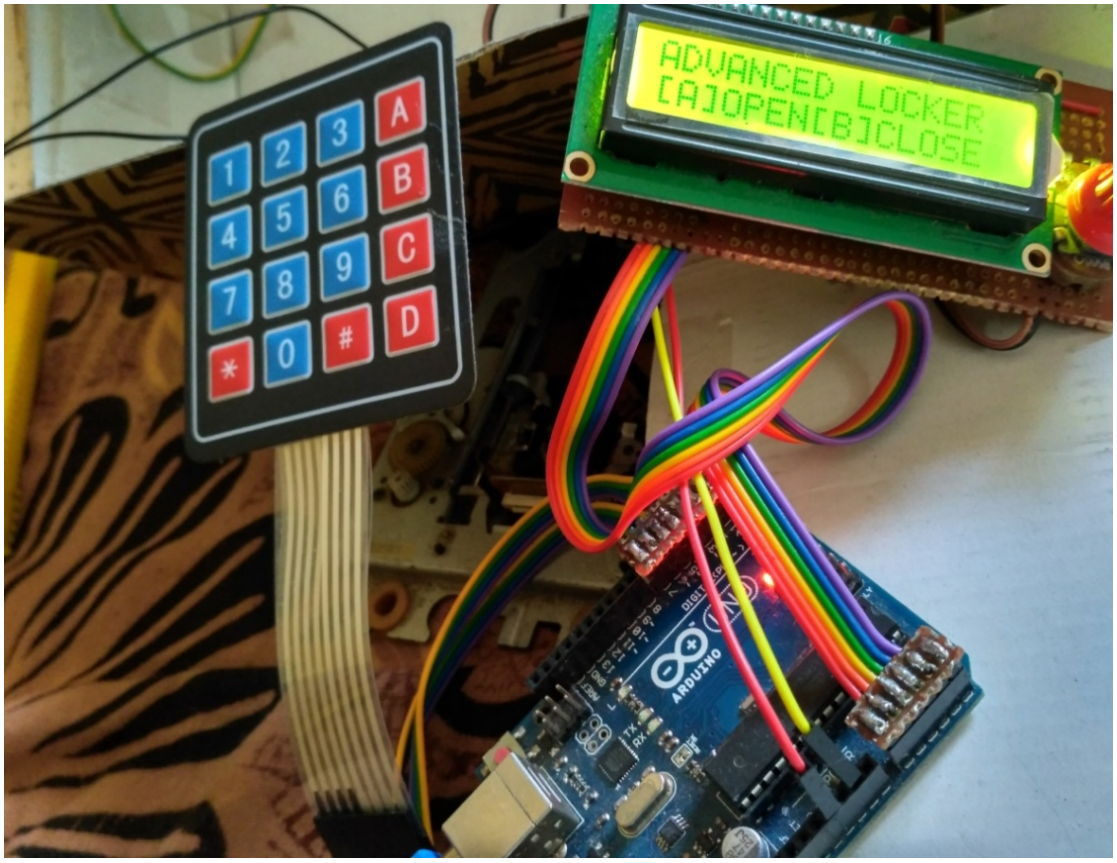

Fig.3.1: Connection of Arduino with LCD and Keypad.

3.1.2 Interfacing GSM to Arduino.

Connecting GSM modem with arduino is very simple just connect RX Line of Arduino to TX Line of GSM Modem and vice versa TX of arduino to Rx of GSM modem. **Make sure use TTL RX, TX lines of GSM modem.** Give 12V 2Amp power supply to GSM modem, Use of less current power supply can cause reset problem in GSM modem, give sufficient current to GSM modem.

3.1.3 Interfacing L293D to Arduino.

Now after combining that system we develop a mechanism that use for open and close door. In doing of that we come to conclusion that motor door is best for door mechanism and also reliable for it.

So to operate a motor we here use a simple motor driver IC that is L293D.In that we can drive 2 motor in continuous by giving pulses form arduino.

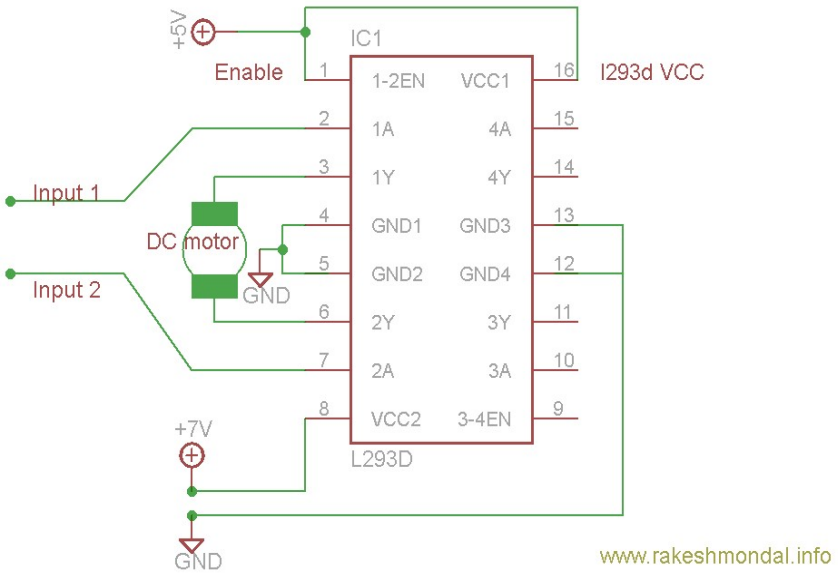

Fig.3.2: L293D IC Pin Diagram For input/output.

L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current amplifiers since they take a low-current control signal and provide a higher-current signal. This higher current signal is used to drive the motors.

L293D contains two inbuilt H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse

direction. The motor operations of two motors can be controlled by input logic at pins 2 & 7 and 10 & 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively.

Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are off and in the high-impedance state.

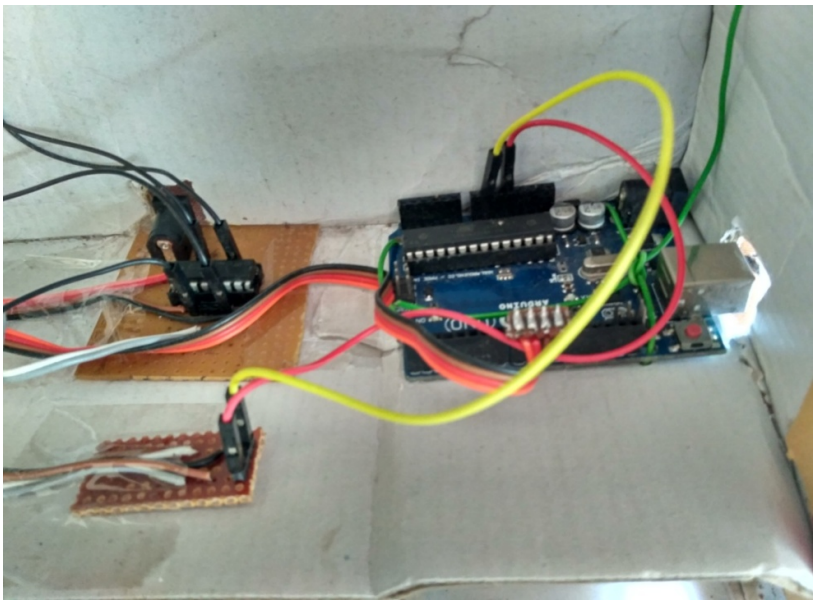

Fig.3.3: Connection of Arduino with L293D.

Now we can operate motor from keypad code just simply change in the code of arduino. So it is very easy that when we put code which taken by user run through keypad that open or do some process as written code.

3.2: PCB layout

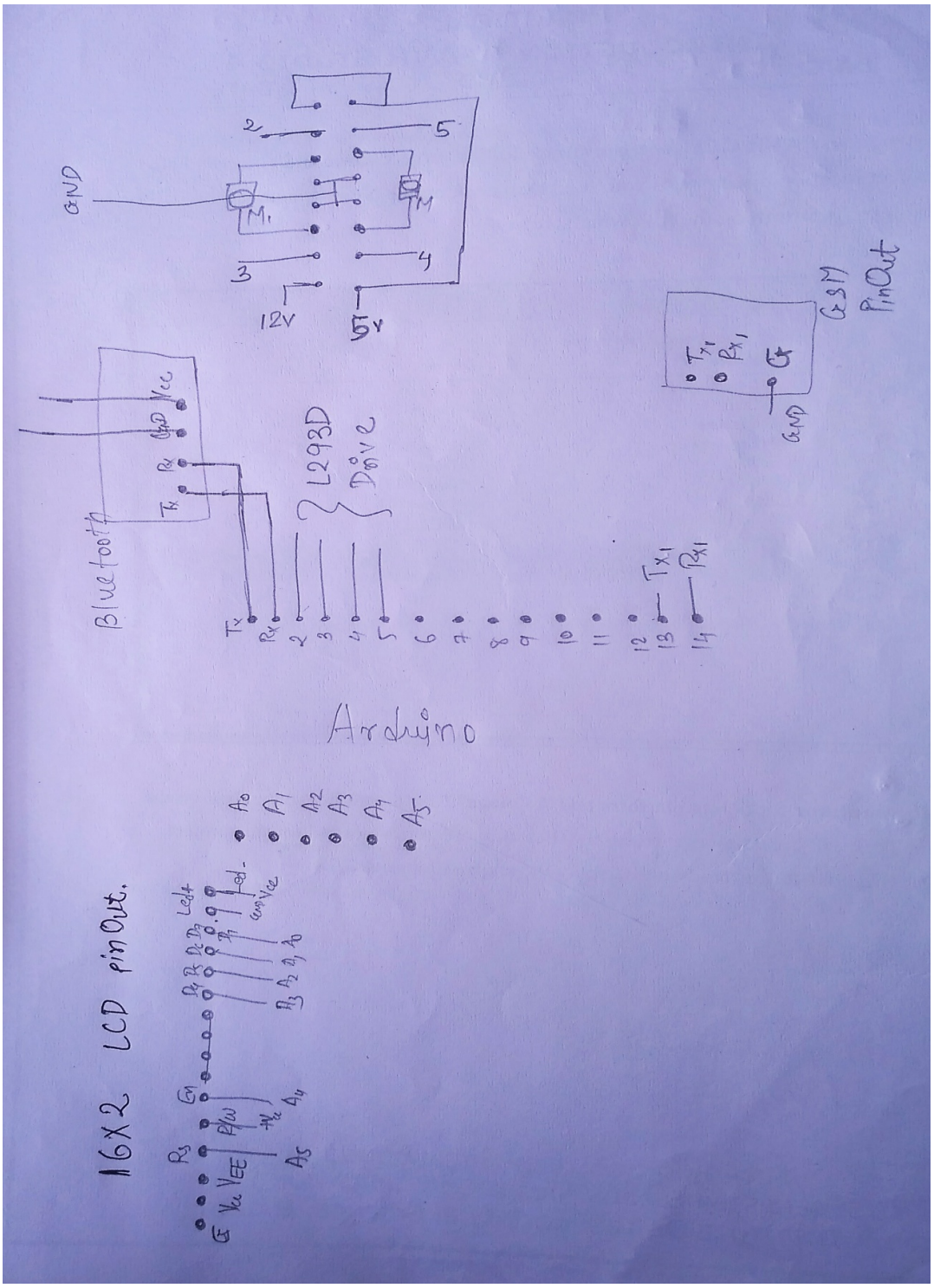

Fig.3.4: Connection of Arduino with other Devices

3.3: Component require

- **Arduino**
- **16x2 LCD display**
- **GSM sim900A**
- **L293D motor Driver**
- **4x4 Keypad**
- **Bluetooth HC-05**

3.4: Component Description

3.4.1: ARDUINO.

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards.

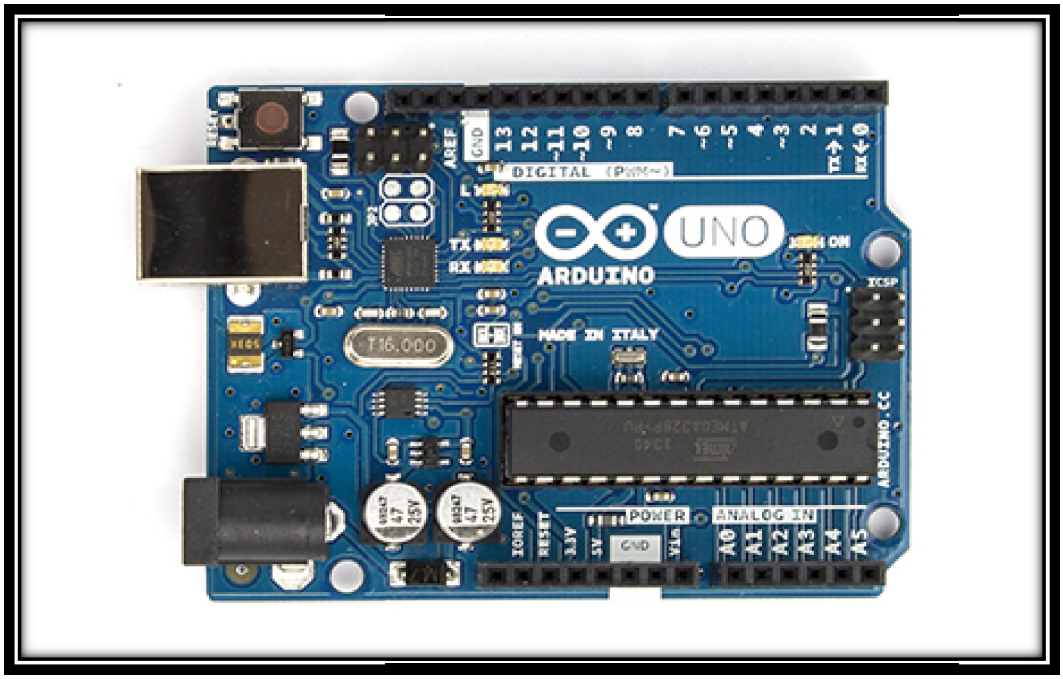

Fig.3.5: ARDUINO UNO.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.**This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.
- **IOREF.** This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM **library.**

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using **pinMode(),digitalWrite()** and **digitalRead()**functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the **attach Interrupt()** function.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with **analog Write ()**function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the **SPI library.**
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and **analogReference()** function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the **Wire Library.**

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with **analogReference()**
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, **on windows,a.inf file is required** .The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A software Serial library allows for serial communication on any of the Uno's digital pins.

Programming

The Arduino Uno can be programmed with the Arduino software. Select "Arduino Uno from the **Tools > Board** menu (according to the microcontroller on your board).The ATmega328 on the Arduino Uno comes pre-burned with a **bootloader**that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nano-farad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the boot loader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line.

USB Over-current Protection

The Arduino Uno has a resettable polypus that protects your computer's USB ports from shorts and over-current. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between

digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

Table 3.1: ARDUINO PIN DESCRIPTION.

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 Ma
DC Current for 3.3V Pin	50 Ma
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by boot loader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

3.4.2:16x2 LCD Display.

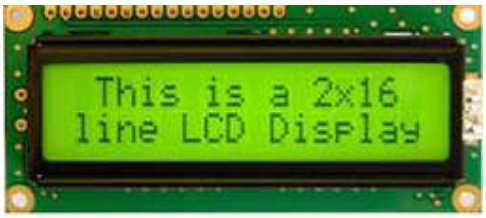

Fig.3.6: A 16x2 LCD

A **16x2 LCD** means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD. Click to learn more about internal structure of a LCD.

Pin Diagram:

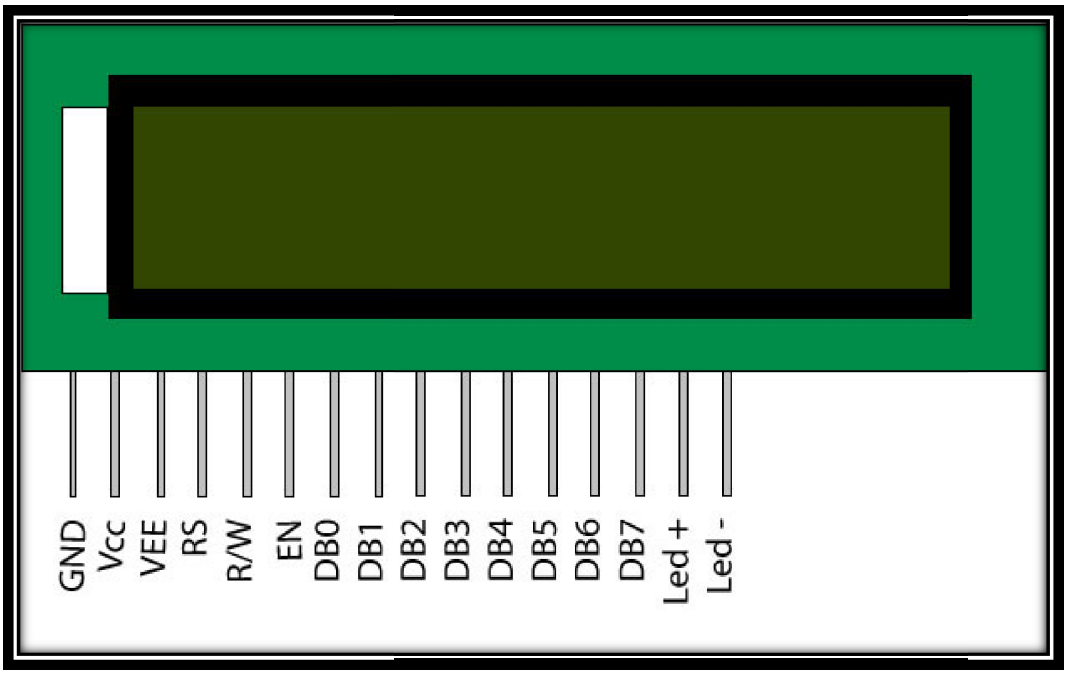

Fig.3.7: 16x2 LCD Pin Diagram.

Table 3.2: Pin Description of 16x2 LCD:

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight V_{CC} (5V)	Led+
16	Backlight Ground (0V)	Led-

3.4.3: Bluetooth Module HC-05.

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Blue core 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature).

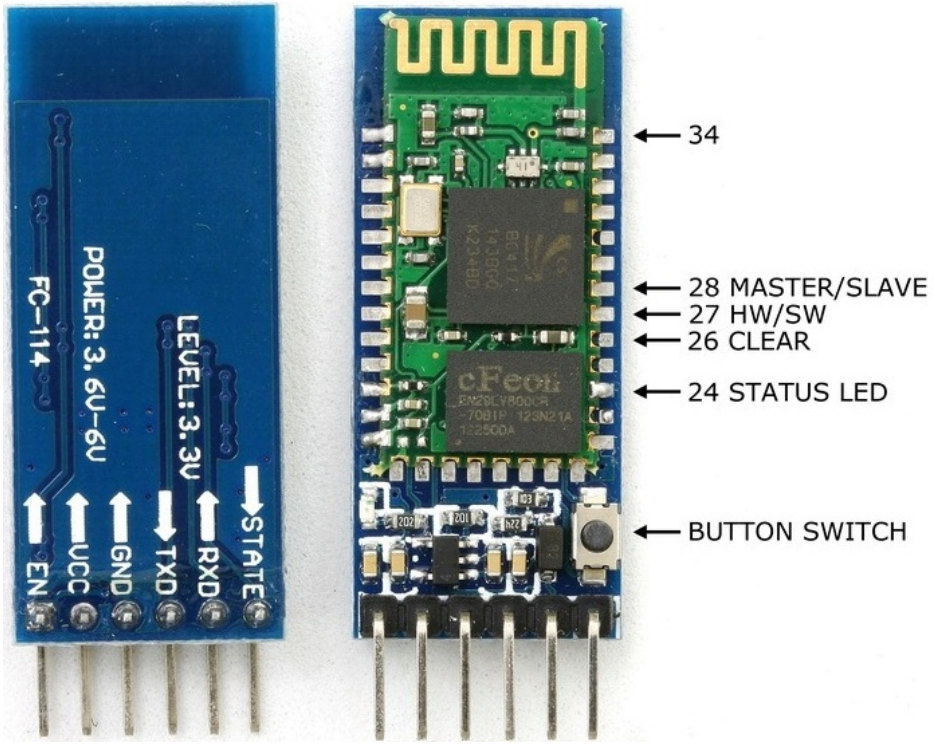

Fig. 3.8: Bluetooth Module HC-05

Hardware Features

- Typical -80dBm sensitivity.
- Up to +4dBm RF transmit power.
- 3.3 to 5 V I/O.
- PIO (Programmable Input/output) control.
- UART interface with programmable baud rate.

- With integrated antenna.
- With edge connector.

Pin Description

The HC-05 Bluetooth Module has 6pins. They are as follows:

- ENABLE:
  When enable is pulled LOW, the module is disabled which means the module will not turn on and it fails to communicate. When enable is left open or connected to 3.3V, the module is enabled i.e. the module remains on and communication also takes place .
- Vcc:
  Supply Voltage 3.3V to 5V

- GND:
  Ground pin

- TXD & RXD:
  These two pins acts as an UART interface for communication

- STATE:
  It acts as a status indicator. When the module is not connected to / paired with any other Bluetooth device, signal goes Low. At this low state, the led flashes continuously which denotes that the module is not paired with other device. When this module is connected to/paired with any other Bluetooth device, the signal goes high. At this high state, the led blinks with a constant delay say for example 2s delay which indicates that the module is paired.

- BUTTON SWITCH:
  This is used to switch the module into AT command mode. To enable AT command mode, press the button switch for a second. With the help of AT commands, the user can change the parameters of this module but only when the module is not paired with any other BT device. If the module is connected to any other Bluetooth device, it starts to communicate with that device and fails to work in AT command mode

3.4.4: 4x4 matrix keypad.

Matrix keypads use a combination of four rows and four columns to provide button states to the host device, typically a microcontroller. Underneath each key is a pushbutton, with one end connected to one row, and the other end connected to one column. These connections are shown in Figure below.

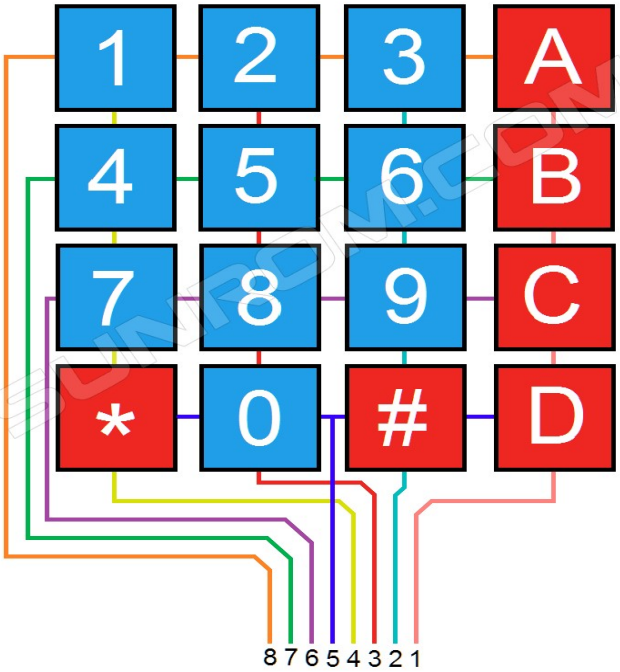

Fig.3.9: 4x4 matrix keypad

In order for the microcontroller to determine which button is pressed, it first needs to pull each of the four columns (pins 1-4) either low or high one at a time, and then poll the states of the four rows (pins 5-8). Depending on the states of the columns, the microcontroller can tell which button is pressed. For example, say your program pulls all four columns low and then pulls the first row high. It then reads the input states of each column, and reads pin 1 high. This means that a contact has been made between column 4 and row 1, so button 'A' has been pressed.

3.4.5: L293D motor driver IC.

The L293 and L293D devices are quadruple high current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current/high-voltage loads in positive supply applications.

Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo-Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN.

The L293 and L293D are characterized for operation from 0°C to 70°C.

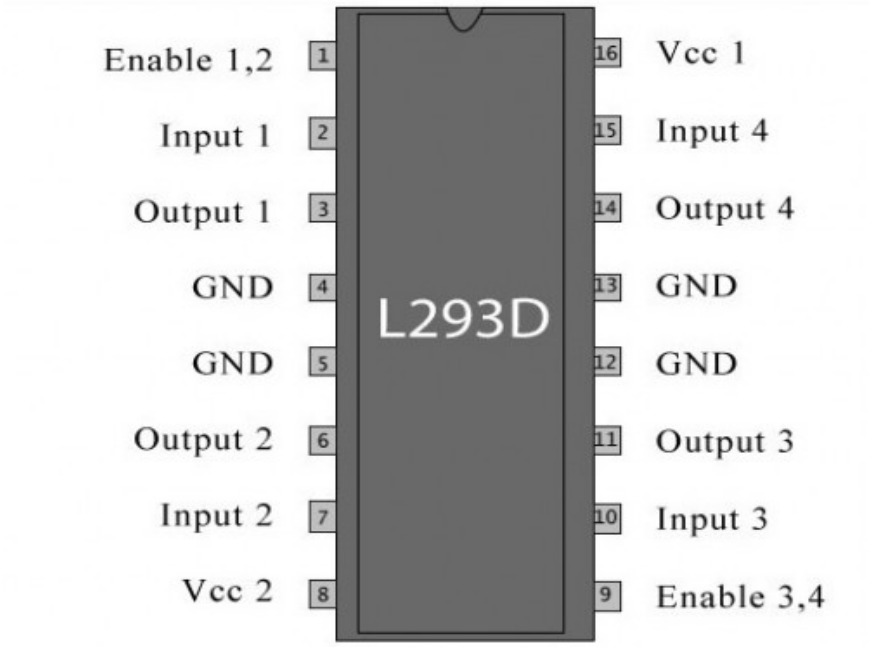

Fig.3.10: L293D IC PIN Diagram

Work concept:

It works on the concept of H-bridge. H-bridge is a circuit which allows the voltage to be flown in either direction. As you know voltage need to change its direction for being able to rotate the motor in clockwise or anticlockwise direction, Hence H-bridge IC are ideal for driving a DC motor In a single L293D chip there are two h-Bridge circuit inside the IC which can rotate two dc motor independently. Due its size it is very much used in robotic

application for controlling DC motors. Given below is the pin diagram of a L293D motor controller.

There are two Enable pins on l293d. Pin 1 and pin 9, for being able to drive the motor, the pin 1 and 9 need to be high. For driving the motor with left H-bridge you need to enable pin 1 to high. And for right H-Bridge you need to make the pin 9 to high. If anyone of the either pin1 or pin9 goes low then the motor in the corresponding section will suspend working. It's like a switch.

3.4.6: GSM module

GSM/GPRS Modem-RS232 is built with Dual Band GSM/GPRS engine- SIM900A, works on frequencies 900/ 1800 MHz The Modem is coming with RS232 interface, which allows you connect PC as well as microcontroller with RS232 Chip(MAX232). The baud rate is configurable from 9600-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface. The onboard Regulated Power supply allows you to connect wide range unregulated power supply . Using this modem, you can make audio calls, SMS, Read SMS, attend the incoming calls and internet ect through simple AT commands.

\Features

Dual band GSM/GPRS 900/1800MHz.

Configurable baud rate.

SIM card holder.

Built in network status LED.

Inbuilt powerful TCP/IP protocol stack for internet data transfer over GPRS.

Applications

Access control devices.

Supply chain management.

Specifications

Parameter	Value
Operating voltage	+12v DC
Weight	<140g

Pin Specification

	Pin Name	Details
1	GND	Power supply ground
2	tx	transmitter
3	rx	receiver
4	Line_r & Line_l	Line input
5	Spk_p & spk_n	Speaker positive & negative
6	Mic_p & mic_n	Mic positive & negative
7	DTR	Data terminal ready
8	CTS	Clear to send
9	RTS	Request to send

Working

Unlike mobile phones, a GSM modem doesn't have a keypad and display to interact with. It just accepts certain commands through a serial interface and acknowledges for those. These commands are called as AT commands. There are a list of AT commands to instruct the modem to perform its functions. Every command starts with "AT". That's why they are called as AT commands. AT stands for attention.

4. SOFTWARE MODELING

4.1: Flow Chart

Stage 1 Application Flow Chart

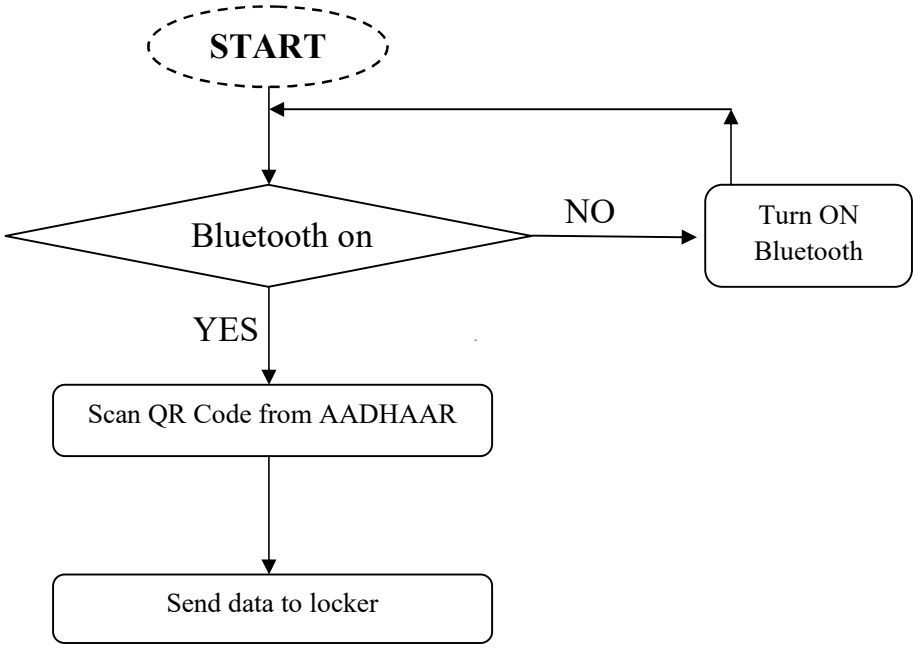

Fig 4.1: Flow chart of Stage 1 Application

Stage2. Application Flow Chart

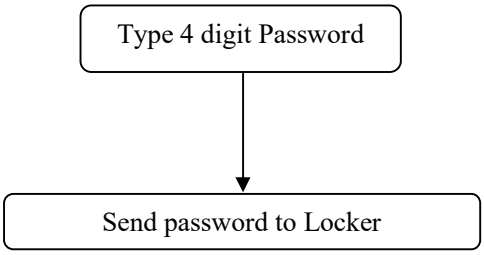

Fig 4.2: Flow chart of stage 2 Application

4.2: Software Description

4.2.1: Mobile Application

In this project we require one mobile application to perform different task. In that task first system run on Bluetooth so mobile and application require to pair with each other so we provide one Bluetooth button that connects application and system.

Now second is to scan QR code from AADHAAR card. Our application is capable to identify any QR code and give Output below that box provided in application and any one can only see what is information of that QR code but we are not provide any edit or copy operation from that output.

As information appears on box application have button that send that information to system.

Now systems check that information if AADHAAR number match with user AADHAAR than system send OTP to user registered number with system.

In next phase Application send data in that Data there is OTP which is received previously and OTP match system allow user to access their document

4.2.2: Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages *Processing* and *Wiring*. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple *one-click* mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.[56]

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

5. Conclusion

Our project entitled advance locker system is successfully completed and the results obtained are satisfactory. it will be easier for the people who are going to take the project for the further modification. The common men also use this project with their Aadhaar card and register mobile number. This project is more suitable for a common man as it is having much more advantages i.e. double layer security, 24/7 banking locker system, no need to remember password. This can be easily handled. This system is having facility of two layer security system that can be providing by Aadhaar card number matching and OTP (one time password) send by the register mobile number.

In this project we made an android application “Aadhaar base advance locker system”. This application scan QR code of Aadhaar card and send the data of Aadhaar card to the arduino through Bluetooth module this is first page of an android application. Second page of an android application is the received OTP entered and verify it.

REFERENCES

Ariponnammal, S. and Natarajan, S. (1994) 'Transport Phonomena of Sm Sel – X Asx', Pramana – Journal of Physics Vol.42, No.1, pp.421-425.

Shin, K.G. and Mckay, N.D. (1984) 'Open Loop Minimum Time Control of Mechanical Manipulations and its Applications', Proc.Amer.Contr.Conf., San Diego, CA, pp. 1231-1236.

http://www.instructables.com/id/Connecting-a-4-x-4-Membrane-Keypad-to-an-Arduino

http://www.circuitstoday.com/interfacing-lcd-to-arduino

A paper on secure fuzzy vault based fingerprint verification system

http://ieeexplore.ieee.org/abstract/document/1399199/

Datasheet of ATmega328P

http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

Arduino Guide

https://www.arduino.cc/en/Guide/Introduction