SHRI VAISHNAV VIDHYAPEETH VISHWAVIDYALAYA, INDORE (M.P.)

Session:2019-20

A Major Project Report
on

BANK LOCKER SECURITY SYSTEM USING FINGERPRINT BASED ON GSM

Submitted in Partial Fulfillment for the award of the Degree of

BACHELOR OF TECHNOLOGY
in
ELECTRONICS & COMMUNICATION ENGINEERING

Submitted by-
NITIN VERMA (16010BTEC00583)

Under the guidance of

MR. PRITESH KUMAR JAIN
Assistant Professor

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING
SHRI VAISHNAV INSTITUTE OF TECHNOLOGY& SCIENCE, INDORE

SHRI VAISHNAV VIDYAPEETH
VISHWAVIDYALAYA,
INDORE (M.P)

RECOMMENDATION

The project report entitled, "**BANK LOCKER SECURITY SYSTEM USING FINGERPRINT BASED ON GSM "**, submitted to Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore, MP by **NITIN VERMA (16010BTEC00583)** during the academic year 2019-20, as a partial fulfillment for the award of degree of the **Bachelor of Technology in Electronics & Communication**, is a record of their own work carried out by them under our direct supervision, in the **Department of Electrical & Electronics Engineering, SVITS, SVVV, Indore**. The work contained in the report is a satisfactory account of their project work and is recommended for the award of the degree.

Mr. Pritesh Kumar Jain	**Dr. Namit Gupta**
Assistant Professor	**Head of Department**

SHRI VAISHNAV VIDYAPEETH
VISHWAVIDYALAYA,
INDORE (M.P)

CERTIFICATE

This is to certify that the report entitled "**BANK LOCKER SECURITY SYSTEM USING FINGERPRINT BASED ON GSM"** is a bonafide record of the major project done by **NITIN VERMA (16010BTEC00583)** under my guidance in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Electronics & Communication** from Shri Vaishnav Institute of Technology and Science of Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore for the academic year 2019-20.

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

We would like to take this opportunity to express our extreme gratitude towards our project guide **Mr. PRITESH KUMAR JAIN,** Assistant Professor, Department of Electrical & Electronics Engineering for his invaluable guidance, advice and support throughout the project. His motivation and help have been a source of great inspiration to us.

We are also grateful to our Project Coordinator **Mr. PREET JAIN,** Associate Professor in our department providing us adequate facilities because of which my project has been successful.

We are very indebted to **Dr. NAMIT GUPTA, Director SVITS** who motivated us to improve the quality of our project and extended all supports required at institution level.

Last but not the least we are also thankful to all faculty & staff members of our department for corporation extended in completion of our project.

NITIN VERMA (16010BTEC00583)

ABSTRACT

When human beings were on earth, need of various things emerged. As years passed and with tremendous development people started earning money, property, jewelry and many more precious things. With huge development people felt need to secure their earnings.

In today's modern world, security plays an important role. Every person has precious accessories like gold, documents or cash. The main goal of this project is to design and implement a bank locker security system based on fingerprint and GSM technology. It reduces wastage of time for both banker as well as customer and provides advanced security.

In this system, only authentic persons can recover money or accessories from bank locker. In this system the user's name, fingerprint and mobile number are enrolled. If the fingerprint matches, then four digit code will be entered by the person using keypad. If it matches with the correct password, then locker will be open , otherwise it will be in locked position and gives an alarm when any mismatch occurs. The sensors will be active during night times to provide security against thefts.

CONTENTS

List of Abbreviations

MPOP – Maximum Power Operating Point

MPPT – Maximum Power Point Tracking

PAO – Perturbation And Observation

Inc. Cond. – Incremental Conductance

PV – Photovoltaic

List of Symbols

V_{pp} – Voltage corresponding to peak power

I_o = Output current in A

V_o = Output voltage in V

I_{ph} = Photon current in A

I_{sat} = Dark saturation current = 10^{-5} to 10^{-15} A

q = charge of an electron = 1.6 X 10^{-19} C

A = Diode ideality factor = 1 to 5

K = Boltzmann constant =1.38 X 10^{-23} J/K

T = absolute temperature in K

R_{se} = Series resistance in ohms

R_{sh} = Shunt resistance in ohms

List of Figures

Chapter 1 : INTRODUCTION

1.1 INTRODUCTION

In the real world, peoples are more concerned about their safety for their valuable things like jewellery, money, important documents etc. So the bank lockers are the safest place to store them. The arrival of fast growing technologies makes users to have high security systems with electronic identification options. These identification technologies include Bank Lockers and ATM as well as other intelligent cards, user IDs and password based systems, and so on. But, unfortunately these are not protected due to hacker attacks, thefts, and forgotten passwords. In spite of all these faults or failure and malfunctions or crash these systems are still existing; however, the biometric or fingerprint authentication based identification is the most efficient and reliable solution for stringent security. Biometrics measure individual's unique physical or the characteristics to recognize or authenticate their identity The physical characteristics are fingerprint hand, face, iris etc and the characteristics are signature, voice keystroke patterns etc. Biometric system operates in verification mode or identification mode. In the verification mode system validates person's identity by comparing the captured biometric template which is pre-stored in the system data base. In the identification mode the system recognize an individual by searching entire template data base for match. And the system performs one to many comparisons to establish the individual identity or fails if the subject is not enrolled in the system data base. So in our project we are using fingerprint security system.

Global system for mobile communication (GSM) is mainly used for sending or receiving data such as voice and message. In our security system GSM plays important role. Through the use of GSM the user will get the message if an unauthorized person will try to open the lock. We are implementing this bank locker security system using fingerprint, password and GSM Technology based security system which provide most efficient and reliable security system than the traditional system.

1.2 Problem Definition

To design and implement a Bank Locker using fingerprint which is based on GSM.

Chapter 2 : LITRETURE SURVEY

These are some of the existing Smart Security designs that have been implemented -

(a) GSM Based Security System -

PIR sensor detects motion by sensing the difference in infrared or radiant heat levels emitted by surrounding objects. The output of the PIR sensor goes high when it detects any motion. The range of a typical PIR sensor is around 6 meters or about 30 feet. When the PIR sensor detects any motion, the output of the sensor is high. This is detected by the Arduino. Then it communicates with the GSM module via serial communication to make a call to the preprogrammed mobile number. An important point to be noted about PIR sensors is that the output will be high when it detects motion.

(b) IR based security alarm system -

IR based security alarm circuit can detect any movement and trigger the alarm. This circuit is very useful in homes, banks, shops, restricted areas where an alert alarm is needed on any movement. This circuit is based on IR sensor where an IR beam is continuously falling on a photodiode, and whenever this Infrared beam breaks, by any kind of movement, alarm is triggered. In this IR based security alarm circuit, we have placed IR LED in front of photodiode, so that IR light can directly falls on photodiode. Whenever someone moves through this beam, IR rays stops falling on photodiode and Buzzer start beeping.

Internet of things has been governing the electronics with cloud services influencing the ever increasing electronics product segment. Security and safety has always become a basic necessity for urban population. The paper proposes a security system based on Open source cloud server “things speak .com” and a low cost esp8266 Wi-Fi module. The project includes a PIR module which constantly monitoring the Home or Work space to be monitored .When the PIR module detects a intruder it sends a signal to the Atmega 328p microcontroller and the controller is connected to a Esp8266 Wifi module and also to a alarm system. The System transmits an alert signal to the Open source cloud which provides a alert signal on the users mobile phone. The system employs a second esp8266 module which is programmed to act as a web server, which allows the user to activate or deactivate the security system by means of any device with internet. The system also employs a thumb print reader rs305 which controls the opening and the closing of a safety locker door. Thus the system uses

esp8266 WiFi module and atmega328p to control the security system from the user's mobile phone by means of any device with a potential internet connection

Chapter 3 : Problem Identification

- In the conventional system, the security is not so high as people can steal the precious things by breaking their locker and as we know only security system which is unbreakable which is fingerprint. And all we know that each individual has a unique fingerprint which is not matched by anyone. So to secure precious things like incident things, jewellery etc ,we use this for securing purpose. Another best security system is password based security system. But sometimes it might be hacked by hackers .

  So to overcome this problem, we proposed a system that requires fingerprint as well as password based which is not easily break by the hackers.

Chapter 4. Proposed Methodology

4.1 Hardware Requirements :

S.No	Components	Quantity
1.	Fingerprint Sensor	1
2.	GSM	1
3.	Microcontroller	1
4.	LCD	1
5.	Keypad(4X4)	1
6.	D.C. Motor	1
7.	L293D Driver	1
8.	Relay	1
9.	Buzzer	1
10.	LED	1
11.	Resistors	11
12.	Capacitors	2

1. Microcontroller(AT89S51)

8051 Microcontroller

The 8051 Microcontroller was designed in 1980's by Intel. Its foundation was on Harvard Architecture and was developed principally for bringing into play in Embedded Systems. At first it was created by means of NMOS technology but as NMOS technology needs more power to function therefore Intel re-intended Microcontroller 8051 employing CMOS technology and a new edition came into existence with a letter 'C' in the title name, for illustration: 80C51. These most modern Microcontrollers need fewer amount of power to function in comparison to their forerunners.

There are two buses in 8051 Microcontroller one for program and other for data. As a result, it has two storage rooms for both program and data of 64K by 8 size. The microcontroller comprise of 8 bit accumulator & 8 bit processing unit. It also consists of 8 bit B register as majorly functioning blocks and 8051 microcontroller programming is done with embedded C language using Keil software. It also has a number of other 8 bit and 16 bit registers.

For internal functioning & processing Microcontroller 8051 comes with integrated built-in RAM. This is prime memory and is employed for storing temporary data. It is unpredictable memory i.e. its data can get be lost when the power supply to the Microcontroller switched OFF.

2. Fingerprint Sensor

Here in this proposed model, Fingerprint module is used for authorized person to unlock the bank locker. Once a Finger print is scanned then you have enter the password. If it matches, the locker may unlock. In case of any unauthorized person try to unlock than he/she has two chances. If he/she fails to unlock , then an alert message is send to mobile via GSM to monitor the bank locker security details.

Figure 1. Finger Print sensor

The sensor is a solid-state fingerprint sensor that reliably captures fingerprint information. It is designed to integrate into devices for improved security and convenience. The sensor provides a reliable, quick and user-friendly alternative to passwords, PIN's and other forms of user authentication.

Finger print processing includes two parts ,fingerprint enrolment and fingerprint matching (the matching can be 1:1 or 1: N). When enrolling, user needs to enter the finger two times. The system will process the two time finger images, generate a template of the finger based on processing results and store the template. When matching, user enters the finger through optical sensor and system will generate a template of the finger and compare it with templates of the finger library.

Features :

- Power DC : 3.6 V~6.0 V
- Interface : UART (TTL logical level) / USB 1.1
- Working Current : 100 mA
- Peak Current : 150mA
- Matching Mode : 1:1 and 1:N
- Character file size: 256 bytes
- Image acquiring time : <0.5sec
- Template size : 512 bytes

- Storage capacity: 120

3. GSM

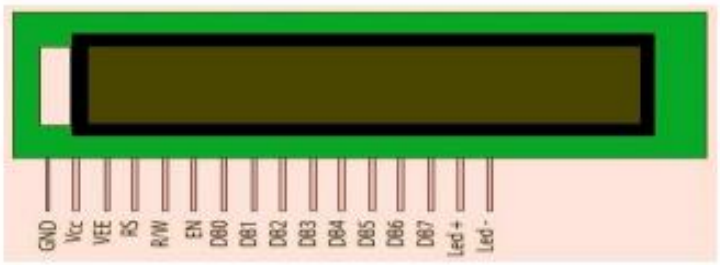
Fig.(1): GSM

GSM (Global System for Mobile communications) is a cellular network, which means that mobile phones connect to it by searching for cells in the immediate vicinity. GSM networks operate in four different frequency ranges. Most GSM networks operate in the 900 MHz or 1800 MHz bands. Some countries in the American continents use the 850 MHz and 1900 MHz bands because the 900 and 1800 MHz frequency bands were already allocated. The rarer 400 and 450 MHz frequency bands are assigned in some countries, where these frequencies were previously used for first-generation systems. 2.3.1 GSM Advantages GSM pioneered a low-cost, to the network carrier, alternative to voice calls, the Short message service which is now supported on other mobile standards as well.

Features :

- Single supply voltage 3.2V-4.5V
- Typical power consumption in SLEEP Mode:2.5mA,SIM300 tri-band
- MT,MO,CB, text and PDU mode, SMS storage: SIM card
- Supported SIM Card :1.8V,3V

4. Liquid Crystal Display (LCD)



A liquid crystal display (commonly abbreviated LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. It is often utilized in battery powered electronic devices because it uses very small amounts of electric power. In this project LCD Display is used for monitoring purpose.

LCDs are made with either a passive matrix or an active matrix display grid. The active matrix LCD is also known as a thin film transistor (TFT) display. The passive matrix LCD has a grid of conductors with pixels located at each intersection in the grid. A current is sent across two conductors on the grid to control the light for any pixel. An active matrix has a transistor located at each pixel intersection, requiring less current to control the luminance of a pixel. For this reason, the current in an active matrix display can be switched on and off more frequently, improving the screen refresh time. Some passive matrix LCD's have dual scanning, meaning that they scan the grid twice with current in the same time that it took for one scan in the original technology. However, active matrix is still a superior technology out of the two.

5. Buzzer

A buzzer is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke.

Buzzer Features and Specifications

- Rated Voltage: 6V DC
- Operating Voltage: 4-8V DC
- Rated current: <30mA
- Sound Type: Continuous Beep
- Resonant Frequency: ~2300 Hz
- Small and neat sealed package
- Breadboard and Perf board friendly

6. D.C. Motor

A DC motor is an electric motor that runs on direct current power. In any electric motor, operation is dependent upon simple electromagnetism. A current carrying conductor generates a magnetic field, when this is then placed in an external magnetic field, it will encounter a force proportional to the current in the conductor and to the strength of the external magnetic field. It is a device which converts electrical energy to mechanical energy. It works on the fact that a current carrying conductor placed in a magnetic field experiences a force which causes it to rotate with respect to its original position.

Practical DC Motor consists of field windings to provide the magnetic flux and armature which acts as the conductor.

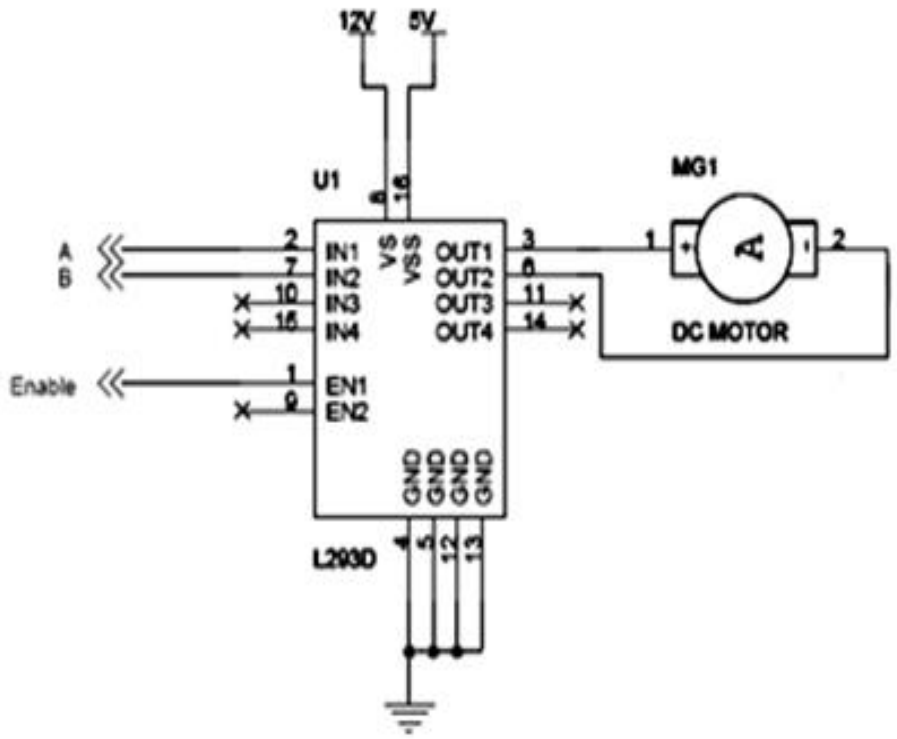
Advantages of DC Motor:

1. Provide excellent speed control for acceleration and deceleration
2. Easy to understand design
3. Simple, cheap drive design

Connecting DC Motor with Microcontroller

Microcontrollers can't drive the motors directly. So we need some kind of drivers to control the speed and direction of motors. The motor drivers will acts as interfacing devices between microcontrollers and motors. Motor drivers will act as current amplifiers since they take a low current control signal and provide a high current signal. This high current signal is used to drive the motors. Using L293D chip is the easy way for controlling the motor using microcontroller. It contains two H-bridge driver circuits internally.

Here is an example of DC motor which is interfaced with L293D microcontroller.



DC motor interfaced with L293D microcontroller

L293D has two set of arrangements where one set has input 1, input 2, output 1 and output 2 and other set has input 3, input 4, output 3 and output 4, according to above diagram.

- If pin no 2 and 7 are high then pin no 3 and 6 are also high. If enable 1 and pin number 2 are high leaving pin number 7 as low then the motor rotates in forward direction.

- If enable 1 and pin number 7 are high leaving pin number 2 as low then the motor rotates in reverse direction.

4.2 Software Requirements

- Proteus Design Suite
- Diptrace
- Keil µVision

1. Proteus Design Suite

The **Proteus Design Suite** is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

It is a software suite containing **schematic**, **simulation** as well as **PCB designing**.

- **ISIS** is the software used to draw schematics and simulate the circuits in real time. The simulation allows human access during run time,thus providing real time simulation.
- **ARES** is used for PCB designing. It has the feature of viewing output in 3D view of the designed PCB along with components.
- The designer can also develop 2D drawings for the product.

Features

ISIS has wide range of components in its library. It has sources, signal generators, measurement and analysis tools like **Oscilloscope**, voltmeter, ammeter etc., probes for real time monitoring of the parameters of the circuit, **switches**, **displays**, loads like motors and lamps, discrete components like resistors, capacitors, inductors,

transformers, digital and analog Integrated circuits, semi-conductor switches, relays, microcontrollers, processors, sensors etc.

ARES offers PCB designing up to 14 inner layers, with surface mount and through hole packages. It is embedded with the foot prints of different category of components like ICs, transistors, headers, connectors and other discrete components. It offers Auto routing and manual routing options to the PCB Designer. The schematic drawn in the ISIS can be directly transferred ARES.

Proteus Modules

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation. All PCB Design products include an auto router and basic mixed mode SPICE simulation capabilities.

Schematic Capture

Schematic capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

Microcontroller Simulation

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool. Support is available for co-simulation of:

- Microchip Technologies PIC10, PIC12, PIC16,PIC18,PIC24,dsPIC33 Microcontrollers.
- Atmel AVR (and Arduino), 8051 and ARM Cortex-M3 Microcontrollers
- NXP 8051, ARM7, ARM Cortex-M0 and ARM Cortex-M3 Microcontrollers.
- Texas Instruments MSP430, PICCOLO DSP and ARM Cortex-M3 Microcontrollers.

PCB Design

The PCB Layout module is automatically given connectivity information in the form of a netlist from the schematic capture module. It applies this information, together with the user specified design rules and various design automation tools, to assist with error free board design. PCB's of up to 16 copper layers can be produced with design size limited by product configuration.

3D Verification

The 3D Viewer module allows the board under development to be viewed in 3D together with a semi-transparent height plane that represents the boards enclosure. STEP output can then be used to transfer to mechanical CAD software such as Solidworks or Autodesk for accurate mounting and positioning of the board.

2. Diptrace

DipTrace is an EDA/CAD software for creating schematic diagrams and printed circuit boards. The developers provide a multi-lingual interface and tutorials (currently available in English and 21 other languages). DipTrace has 4 modules: schematic capture editor, PCB layout editor with built-in shape-based autorouter and 3D-preview & export, component editor, and pattern editor.

Basic Features

- Simple user interface
- Multi-sheet and hierarchical schematics
- High-speed and differential signal routing
- Smart manual routing modes
- Wide import/export capabilities

Schematic Capture

Advanced circuit design tool with support of multi-sheet and multi-level hierarchical schematics that delivers a number of features for visual and logical pin connections. Cross-module management ensures that principal circuits can be easily converted into a PCB, back-annotated, or imported/exported from/to other EDA software, CAD formats and net-lists. DipTrace Schematic has ERC verification and Spice export for external simulation.

PCB Layout

Engineering tool for board design with smart manual routing, differential pairs, length-matching tools, shape-based autorouter, advanced verification, layer stackup manager, and wide import/export capabilities. Design requirements are defined by net classes, class-to-class rules, and detailed settings by object types for each class or layer. When routing with real-time DRC, the program reports errors on the fly before actually making them. DRC also checks length and phase tolerances for differential pairs and controls signal synchronization for nets and buses (including layer stackup and bonding wire induced signal delays). The board can be previewed in 3D and exported to STEP format for mechanical CAD modeling. Design rule check with in-depth detailing and net connectivity verification procedures are available.

3-D Preview and Export

This module includes real-time 3D preview & export feature. It shows the model of the manufactured printed circuit board with all components installed. Rotate board in three axes, zoom in and out in real time, change colors of the board, copper areas, solder mask, silkscreen, and background. 3D preview works on all stages of the design. Board can be exported to STEP or VRML 2.0 formats for mechanical CAD modeling. More than 7500 3D models of PCB packages are supplied for free. Externally designed 3D models in *.wrl, *.step, *.iges, and *.3ds formats can be uploaded and attached to patterns in Pattern Editor or PCB Layout.

Component Editor

Manage component libraries and create single- or multi-part components by selecting a template and its dimensions, defining visual and electrical pin parameters, setting up a Spice model, and attaching pattern with a 3D model to finalize component creation. BSDL import, bulk pin naming, and pin manager tools for pins and buses. Importing libraries from different EDA formats. More than 140000 components in standard libraries.

Pattern Editor

Draw patterns with various types of shapes, pads, holes, and dimensions. Circle, lines (headers, DIP), square (QFP), matrix (BGA), rectangle (RQFP), and zig-zag standard templates. Creation of pattern is basically selecting a template, entering a couple of vital parameters, drawing the silkscreen, and launching automatic pad renumbering. Custom templates can be created for non-standard patterns. DXF import makes creating complex layouts easier.

3. Keil µVision

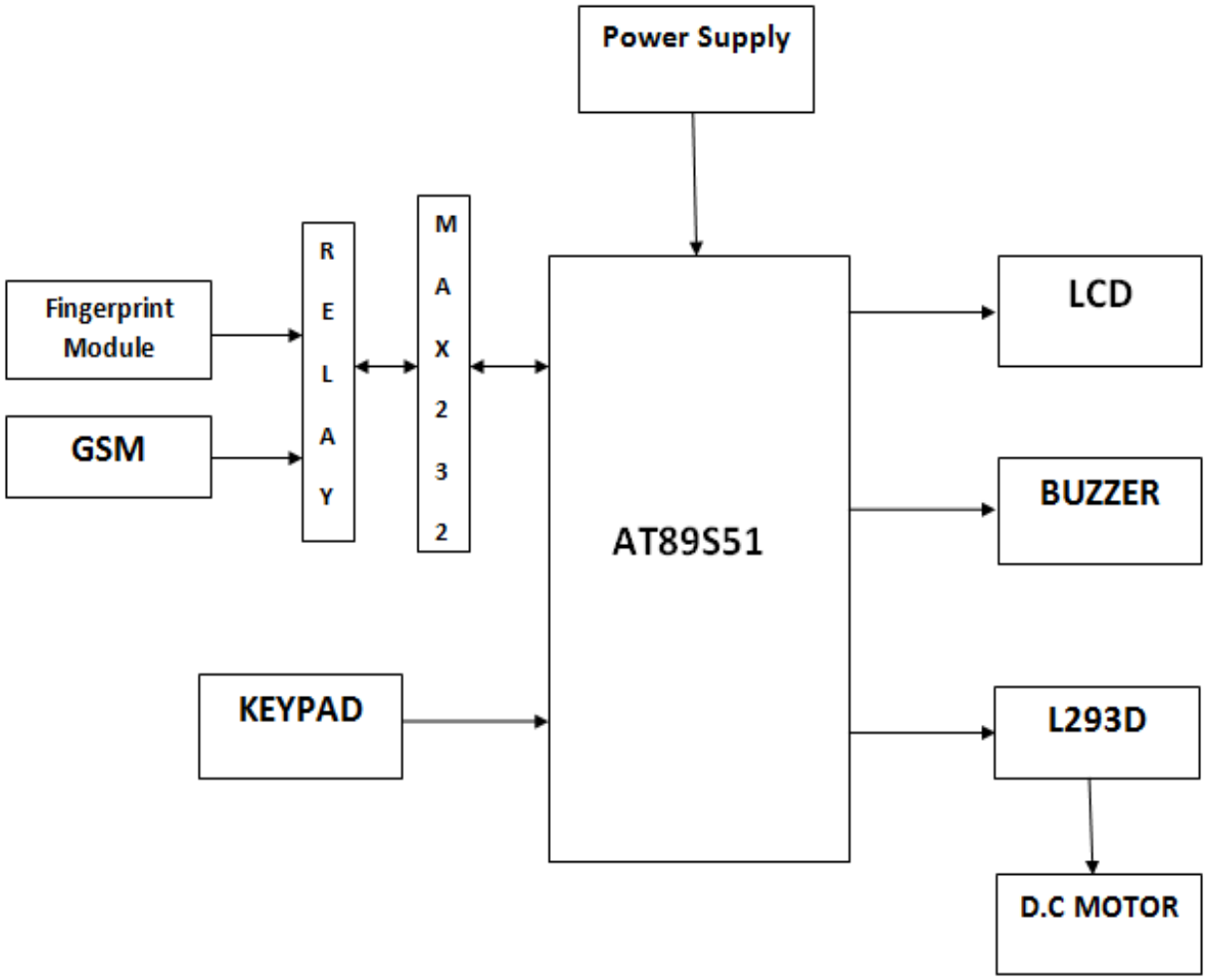
Keil MicroVision is a free software which solves many of the pain points for an embedded program developer. This software is an integrated development environment (IDE), which integrated a text editor to write programs, a compiler and it will convert your source code to hex files too.

Here is simple guide to start working with Keil uVision which can be used for

- Writing programs in C/C++ or Assembly language
- Compiling and Assembling Programs
- Debugging program
- Creating Hex and Axf file
- Testing your program without Available real Hardware (Simulator Mode)

Chapter 5 : Block Diagram

5.1 BLOCK DIAGRAM



5.2 BLOCK DIAGRAM DESCRIPTION

1. Microcontroller

The AT89S51 is a low-power, high-performance CMOS 8-bit micro controller. It provides a highly flexible and cost-effective solution to many embedded control applications. It has 256 bytes of internal RAM and 32 Programmable I/O Lines.

2. Finger Print Module

In this technology one's finger is the key i.e., one's fingerprints are used as the "PASSWORD" for identification and verification. Fingerprint technology was developed by Fujitsu to help combat the increasing incidence of financial fraud and forgery. Among these biometric traits, fingerprint proves to be one of the best traits providing good mismatch ratio, highly accurate in terms of security and also reliable.

3. GSM

GSM (Global System for Mobile communications) is a cellular network, which means that mobile phones connect to it by searching for cells in the immediate vicinity. GSM networks operate in four different frequency ranges. Most GSM networks operate in the 900 MHz or 1800 MHz bands. Some countries in the American continents use the 850 MHz and 1900 MHz bands because the 900 and 1800 MHz frequency bands were already allocated. The rarer 400 and 450 MHz frequency bands are assigned in some countries, where these frequencies were previously used for first-generation systems. 2.3.1 GSM Advantages GSM pioneered a low-cost, to the network carrier, alternative to voice calls, the Short message service which is now supported on other mobile standards as well.

4.Keypad

In our project we are using 4x4 keypad. It is used for entering the password.

5. LCD and Buzzer

It is used as an output for indications. Here LCD is used for display and buzzer is used for alarming purpose.

6. L293D

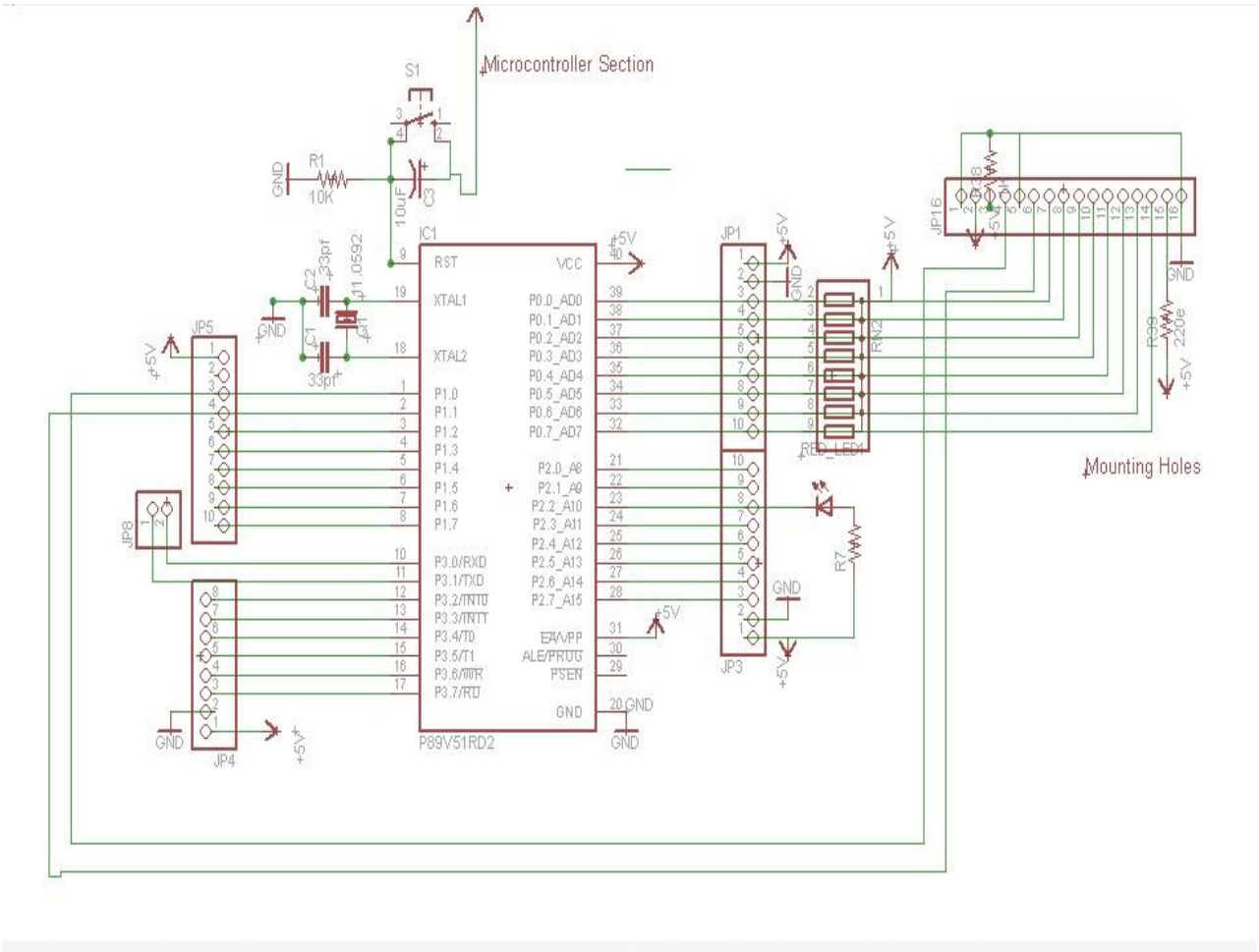
L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. **L293D** is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. It means that you can control two DC motor with a single **L293D** IC. Dual H-bridge Motor Driver integrated circuit (I.C.)

7. D.C. Motor

It is used as a locker for opening and closing the door in our project.

Chapter 6 : Circuit Diagram

6.1 Circuit Diagram



6.2 Circuit Diagram Description

Finger print processing includes two parts ,fingerprint enrolment and fingerprint matching (the matching can be 1:1 or 1: N). When enrolling, user needs to enter the finger two times. The system will process the two time finger images, generate a template of the finger based on processing results and store the template. When matching, user enters the finger through optical sensor and system will generate a template of the finger and compare it with templates of the finger library.

The power required to the microcontroller (5V) is given through the power supply. MAX232 interfaces the microcontroller with fingerprint and GSM. Fingerprint and GSM modules are connected to MAX232 using relay. MAX232 converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits. This reduces the complexity of the power supply design. The above diagram shows that various components like keypad, sensors, LCD, buzzer are connected to microcontroller. The L293D driver provides the required power to the motor to open the locker door.

Chapter 7 : Results and Discussion

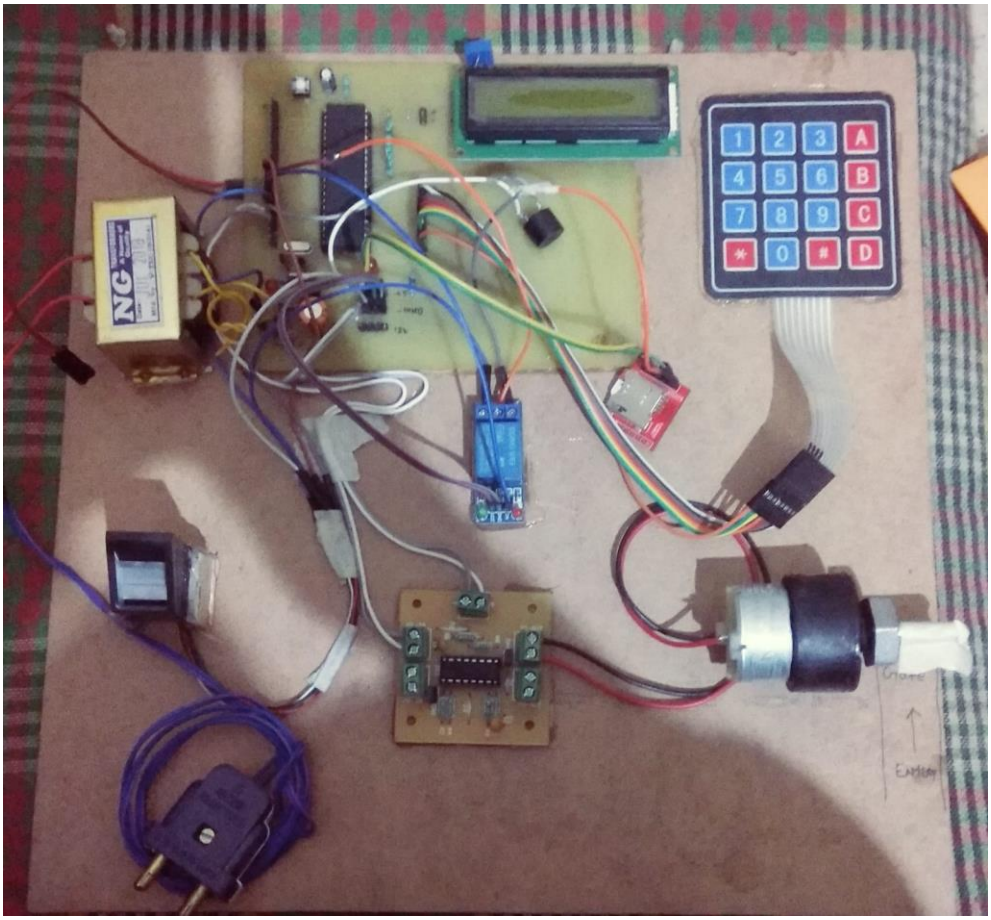


Fig.1 : General arrangement

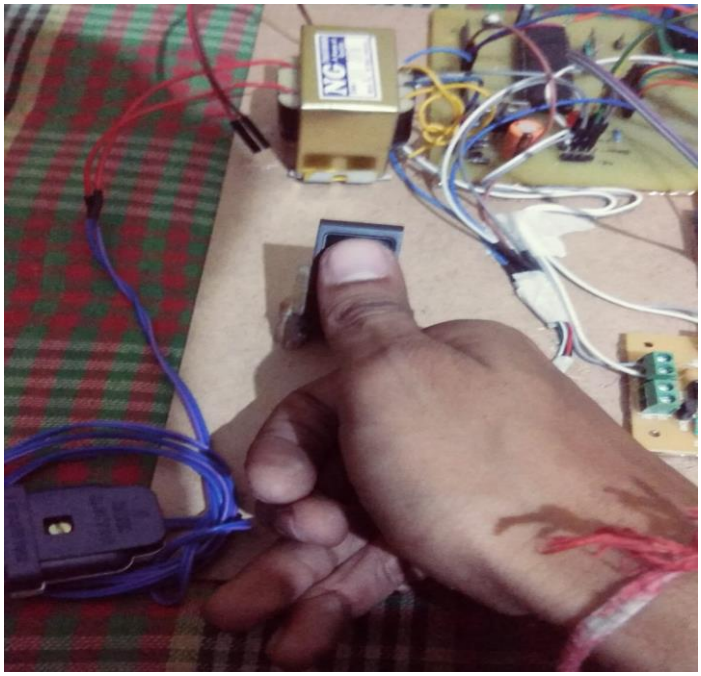


Fig.2 : Scan the finger

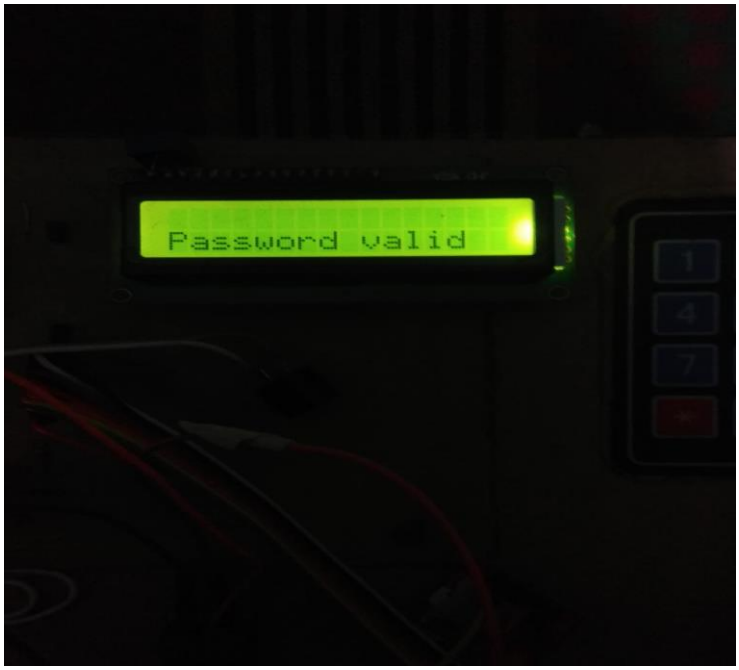


Fig.3 : When password is correct

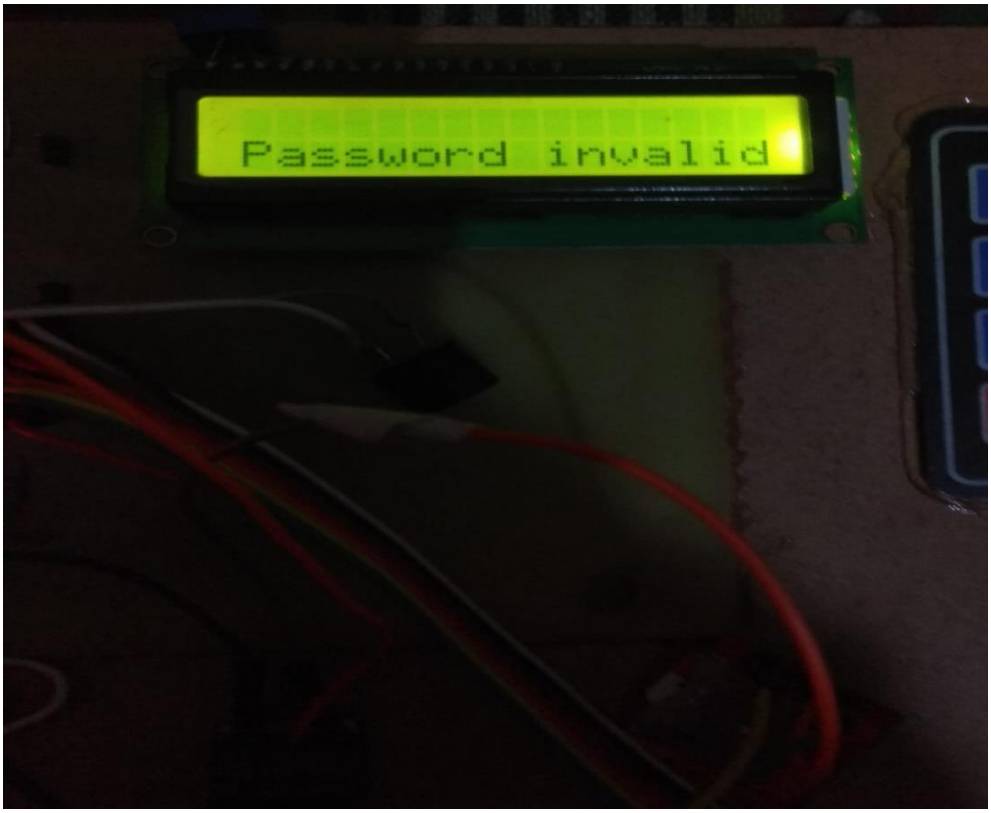


Fig.4 : When password is incorrect

Chapter 8 : Conclusion

8.1 Conclusion

The main goal of this project is to design and implement a bank locker security system based on Finger print. This can be organized in bank, offices and homes. In this system only the authenticate person recover the documents or money from the lockers. In this security system fingerprint and GSM is used. The 8051 Microcontroller is used as heart of the project. IC embedded C program is written using Keil. The 8051 Microcontroller is reprogrammable. In the future we can enhance it for more security issues for related to Jewellery Shops, RBI, Aerospace, Defense , Navy, Hospital, etc. 8051 Microcontroller is re-programmable, so can enhance it more number of applications in future.

Finger print and GSM security system will provide higher security than existing system. The design system which when implemented would surely give a very good protection of the lockers curbing theft and making the lockers more reliable. The assurance it will give to the bank customers will force them to use it and hence protect their valuables from theft or any kind of robbery.

8.2 Future Scope of Work

- In addition to this the future scope of this project is to develop smart bank Locker security system based on “FACE”, “IRIS and Retina” Scanning for visual identification of the person.

- We can also add Voice recognition feature for passwords for better experience.

REFERENCES

[1] A.Aditya Shankar, P.R.K.Sastry, A.L.Vishnu ram.A.Vamsidhar Fingerprint Based Door Locking System International Journal of Engineering and Computer Sciences ISSN:2319-7242, Volume 4 Issue 3 March 2015.

[2] Kanak Chopra, garvit Jain Door Opening System Based On Fingerprint Scanning International Journal of Engineering Research Management Technology, March 2015, Volume 2,Issue-2.

[3] Pavithra.B.C, Myna.B.C, Kavyashree.M Fingerprint Based Bank Locker System Using Microcontroller Proceedings of IRF International Conference, 5 April-2014, Pondicherry, India, ISBN: 978-93-82702-71-9.

[4] Tutorial on microcontroller: www.8051pro jects.net/microcontroller_tutorials

[5] Tutorial on LCD: wwws.8051projects.net/ lcd-interfacing/

APPENDIX

```
#include<reg51.h>                       //Header file inclusion for 8051
#include<stdio.h>
sfr LCDdata = 0x80;                     //Connections of LCD data lines// port 0
sbit LCDrs = P1^0;                      //The Register select Pin
sbit LCDen = P1^1;                      //The Enable Pin

sbit r1=P2^0;
sbit r2=P2^1;
sbit r3=P2^2;
sbit r4=P2^3;
sbit c1=P2^4;
sbit c2=P2^5;
sbit c3=P2^6;
sbit c4=P2^7;
sbit led=P1^7;
sbit relay=P3^3;
sbit scan=P3^4;
sbit buzzer=P3^5;
sbit m1=P3^6;
sbit m2=P3^7;
unsigned char rec1[12],rec2[12],rec3[12],rec4[17];
unsigned char abc[16];
unsigned int i;
unsigned char Search[12] = {0XEF, 0X01, 0XFF, 0XFF, 0XFF, 0XFF, 0X01, 0X00, 0X03, 0X01,
0X00, 0X05 };
char p,q,r,s;
unsigned int j;

void delay(unsigned int rtime)
{
        unsigned int r,s;
        for(r=0;r<rtime;r++)
        for(s=0;s<1275;s++);
}
//This function makes one column to GND one by one and check the rows for key press
char keypress()
{
 while(1)
 {
 c1=0;c2=1;c3=1;c4=1;
 if(r1==0){return '1';}
 if(r2==0){return '4';}
 if(r3==0){return '7';}
 if(r4==0){return '*';}

 c1=1;c2=0;c3=1;c4=1;
 if(r1==0){return '2';}
 if(r2==0){return '5';}
 if(r3==0){return '8';}
 if(r4==0){return '0';}

 c1=1;c2=1;c3=0;c4=1;
 if(r1==0){return '3';}
 if(r2==0){return '6';}
 if(r3==0){return '9';}
 if(r4==0){return '#';}
 c1=1;c2=1;c3=1;c4=0;
 if(r1==0){return 'A';}
```

```
 if(r2==0){return 'B';}
 if(r3==0){return 'C';}
 if(r4==0){return 'D';}
 }
}

void lcdcmd (unsigned char DATA)
        {
                LCDrs=0;
                LCDen=1;
                LCDdata=DATA;
                LCDrs=0;
                LCDen=0;
        }

void initialize (void)
        {
                lcdcmd (0x30);
                delay(30);
                lcdcmd (0x38);
                delay(30);
                lcdcmd (0x0C);
                delay(30);
                lcdcmd (0x01);
                delay(30);
                lcdcmd (0x06);
                delay(30);
        }
void lcddat (unsigned int DATA)
        {
                LCDrs=1;
                LCDen=1;
                LCDdata=DATA;
                LCDrs=1;
                LCDen=0;
        }
void display_lcd (unsigned char location, unsigned char *d)
        {
                lcdcmd(0x80 | location);
                delay(1);
                while(*d)
                    {
                            lcddat(*d++);
                            //delay(1);
                        }
        }

void ser_send(char dat)
{
SBUF=dat;
while(TI==0);
TI=0;
}

char ser_rec(void)
{
   //char val;
   while(!RI);
   RI=0;
   //val=SBUF;
```

```
   return SBUF;
}
void main(void)
{
        scan=1,led=1;
        relay=0;
        initialize();
  SCON = 0x52; // 8-bit UART mode
        TMOD = 0x20; // timer 1 mode 2 auto reload
        TH1= 0xfd; // 9600 8-n-1
        TR1 = 1; // run timer1
        TI=1;
        printf("AT\n");
        delay(20);
        printf("AT+CMGF=1\n");
        delay(20);
        printf("AT+CMGD=1\n");
        delay(20);
        printf("AT+CMGS=\"8319877823\"\n");
        delay(10);
        printf("two step security system \n");
        delay(10);
        putchar(26);
        delay(1000);
          //sprintf(abc,"%s","super");
                //sprintf(abc,"%c",OpenDevice[i]);
                //display_lcd(0xc0,abc);


        while(1)
        {
                start2:
   start4:
        relay=1;// relay off postion
        SCON = 0x52; // 8-bit UART mode
        TMOD = 0x20; // timer 1 mode 2 auto reload
        TH1= 0xfd; // 9600 8-n-1
        TR1 = 1; // run timer1
        TI=1;
        led=1;
        display_lcd(0xc0,"Start          ");

        if(scan==0 )
        {
   display_lcd(0xc0,"Scan           ");
                ser_send(0xFF);
                ser_send(0x05);
                ser_send(0xFD);
                ser_send(0xFF);
                ser_send(0x4B);
                ser_send(0xFF);
                ser_send(0xEE);
                ser_send(0x82);
                ser_send(0x25);
                ser_send(0xF3);
                ser_send(0xF6);
                ser_send(0x04);
                ser_send(0x88);
                ser_send(0x08);
                ser_send(0x08);
```

```
                ser_send(0xFF);
                ser_send(0xEF);
                ser_send(0x01);
                ser_send(0xFF);
                ser_send(0xFF);
                ser_send(0xFF);
                ser_send(0xFF);
                ser_send(0x01);
                ser_send(0x00);
                ser_send(0x07);
                ser_send(0x13);
                ser_send(0x00);
                ser_send(0x00);
                ser_send(0x00);
                ser_send(0x00);
ser_send(0x00);
                ser_send(0x1B);

        for (i = 0; i < 12; i++)
                 {
 rec1[i] = ser_rec();
 }
                for(i=0;i<12;i++)  //Identify
                {
                ser_send(Search[i]); // scane need to send 0x05 last value
                }
        for (i = 0; i < 12; i++)
                {
 rec2[i] = ser_rec();
}
 if (rec2[11] == 0X0A)
{
                                ser_send(0XEF);// 13 value sent
                                ser_send(0X01);
                                ser_send(0XFF);
                                ser_send(0XFF);
                                ser_send(0XFF);
                                ser_send(0XFE);
                                ser_send(0X01);
                                ser_send(0X02);
                                ser_send(0X04);
                                ser_send(0X02);
                                ser_send(0X01);
                                ser_send(0X09);
                                ser_send(0X08);
                        for (i = 0; i <12; i++)
                        {
 rec3[i] = ser_rec();
 }
}
 if (rec3[11] == 0X0A)
{
                                ser_send(0XEF);// 17 value sent
                                ser_send(0X01);
                                ser_send(0XFF);
                                ser_send(0XFF);
                                ser_send(0XFF);
                                ser_send(0XFF);
                                ser_send(0X01);
                                ser_send(0X00);
```

```
                                        ser_send(0X08);
                                        ser_send(0X1b);
                                        ser_send(0XC1);
                                        ser_send(0X00);
                                        ser_send(0XH0);
                                        ser_send(0X03);
                                        ser_send(0XE9);
                                        ser_send(0X01);
                                        ser_send(0X11);

 rec4[0] = ser_rec();
                                rec4[1] = ser_rec();
                                rec4[2] = ser_rec();
                                rec4[3] = ser_rec();
                        rec4[4] = ser_rec();
              rec4[5] = ser_rec();
                                rec4[6] = ser_rec();
                                rec4[7] = ser_rec();
 rec4[8] = ser_rec();
                                rec4[9] = ser_rec();
                                rec4[10] = ser_rec();
                                rec4[11] = ser_rec();
}
 if(rec2[11]==0x0A )
          {
          display_lcd(0xc0,"   print found  ");
          delay(300);
          }
          if(rec4[11]==0x01)
          {
                    led=0;
          display_lcd(0xc0,"ID: 1           ");
          delay(200);
                    start1:
display_lcd(0x40,"Password:       ");
                                p = keypress();
                                sprintf(abc,"%c",p);
                                display_lcd(0x49,abc);
                                delay(50);
                                q = keypress();
                                sprintf(abc,"%c",q);
                                display_lcd(0x4A,abc);
                                delay(50);
                                r = keypress();
                                sprintf(abc,"%c",r);
                                display_lcd(0x4B,abc);
                                delay(50);
                                s = keypress();
                                sprintf(abc,"%c",s);
                                display_lcd(0x4C,abc);
                                delay(50);
                                if(p=='2'&& q=='5'&& r=='6'&& s=='3')
                                {
                                 display_lcd(0x40,"Password valid");
                                 delay(200);
                                          m1=1;
                                          m2=0;
                                          delay(725);
                                          m1=0;
                                          m2=0;
```

```
                        delay(800);
                        m1=0;
                        m2=1;
                        delay(725);
                        m1=0;
                        m2=0;
                        delay(100);
                    }
                    else
                    {

                     buzzer=0;
                     display_lcd(0x40,"Password invalid");
                     delay(50);
                     j++;
                     delay(50);
                     buzzer=1;
                     if(j<2)
                     {
                     display_lcd(0x40,"Try Again       ");
                     delay(100);
                     }

                     if(j==2)
                     {
                     j=0;
                     display_lcd(0x40,"Max attemp finsh");
                     delay(500);
                     SCON = 0x52; // 8-bit UART mode
            TMOD = 0x20; // timer 1 mode 2 auto reload
                     TH1= 0xfd; // 9600 8-n-1
                     TR1 = 1; // run timer1
                     TI=1;
 relay=0;
 delay(500);
                     printf("AT\n");
                     delay(20);
                     printf("AT+CMGF=1\n");
                     delay(20);
                     printf("AT+CMGD=1\n");
                     delay(20);
                     printf("AT+CMGS=\"8319877823\"\n");
                     delay(10);
                     printf(" wrong access \n");
                     delay(10);
                     putchar(26);
                     delay(1000);
                          goto start2;
                     }
                     goto start1;
                    }//password else

        }// FP if close
                else if(rec4[11]==0x02)
        {
                led=0;
        display_lcd(0xc0,"ID: 2          ");
        delay(200);
                start3:
display_lcd(0x40,"Password:      ");
```

```
                    p = keypress();
                    sprintf(abc,"%c",p);
                    display_lcd(0x49,abc);
                    delay(50);
                    q = keypress();
                    sprintf(abc,"%c",q);
                    display_lcd(0x4A,abc);
                    delay(50);
                    r = keypress();
                    sprintf(abc,"%c",r);
                    display_lcd(0x4B,abc);
                    delay(50);
                    s = keypress();
                    sprintf(abc,"%c",s);
                    display_lcd(0x4C,abc);
                    delay(50);
                    if(p=='2'&& q=='5'&& r=='6'&& s=='3')
                    {
                     display_lcd(0x40,"Password valid");
                     delay(200);
                              m1=1;
                              m2=0;
                              delay(725);
                              m1=0;
                              m2=0;
                              delay(800);
                              m1=0;
                              m2=1;
                              delay(725);
                              m1=0;
                              m2=0;
                              delay(100);
                    }
                    else
                    {

                     buzzer=0;
                     display_lcd(0x40,"Password invalid");
                     delay(50);
                     j++;
                     delay(10);
                     buzzer=1;
                     delay(50);
                     if(j<2)
                     {
                     display_lcd(0x40,"Try Again       ");
                     delay(100);
                     }

                     if(j==2)
                     {

                     j=0;
                     display_lcd(0x40,"Max attemp finsh");
                     SCON = 0x52; // 8-bit UART mode
          TMOD = 0x20; // timer 1 mode 2 auto reload
                     TH1= 0xfd; // 9600 8-n-1
                     TR1 = 1; // run timer1
                     TI=1;
relay=0;
```

```
    delay(500);
                                printf("AT\n");
                                delay(20);
                                printf("AT+CMGF=1\n");
                                delay(20);
                                printf("AT+CMGD=1\n");
                                delay(20);
                                printf("AT+CMGS=\"8319877823\"\n");
                                delay(10);
                                printf(" wrong access \n");
                                delay(10);
                                putchar(26);
                                delay(1000);
                                goto start4;
                                }
                                goto start3;
                               }

            }
  else
            {
            display_lcd(0xc0,"Access Denied! ");
            delay(200);
            }
            rec1[12]=0;rec2[12]=0;rec3[12]=0;rec4[17]=0;

//////////////////////////
}//if scane
}// while
}//void
```