

SMART ATM CARD FOR MULTIPLE BANK ACCOUNTS USING RFID

Submitted in partial fulfillment of the requirements for the award of Bachelor
of Engineering Degree in Electronics and Communication Engineering

by

C.MADHUMITHA (38130121)

C.SELVALAKSHMI (38130200)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with 'A' grade by NAAC | 12B status by UGC | Approved by AICTE
Jeppiar Nagar, Rajiv Gandhi salai, Chennai-600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **MADHUMITHA C (38130121)** and **SELVALAKSHMI C (38130200)** who carried out project entitled” **SMART ATM CARD FOR MULTIPLE BANK ACCOUNTS**” under our supervision from October 2021 to April 2022.

Internal Guide

Dr. R. JOANY, M.E, Ph.D.,

Head of the Department

Dr. T. RAVI, M.E., Ph.D.,

Submitted for Viva voce Examination held on

Internal Examiner

External Examiner

DECLARATION

We **MADHUMITHA C (38130121)** and **SELVALAKSHMI C (38130200)** hereby declare that the Project Report entitled “**SMART ATM CARD FOR MULTIPLE BANK ACCOUNTS – USING RFID**” done by us under the guidance of **Dr. R. JOANY, M.E, Ph.D.**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

1)

2)

DATE:

PLACE:

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey our thanks to **Dr. N.M. NANDHITHA, M.E, Ph.D., Dean, School of Electronics and Communication Engineering** and **Dr. T. RAVI, M.E, Ph.D., Head of the Department, Department of Electronics and Communication Engineering** for providing me necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. R.M. Joany, M.E, Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

We wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and communication Engineering** who were helpful in many ways for the completion of the project.

We express our gratitude to our parents for their continuous encouragement and support for the completion of the project.

ABSTRACT

ATM is an abbreviation of Automated Teller Machine. It is introduced in the year 1959 for encouraging self-service in retail banking. This makes people to deposit, withdraw and transfer amount without the help of banking personnel's and it can be done at anytime and anywhere. At first, the ATM was made to transact for the particular bank customers but later on the ATM are connected to inter-bank network, so that it enables people to deposit, withdraw and transfer amount from the ATM machines not belonging to that particular bank (i.e.) any one can access any banks ATM machine to carry out their transactions. ATM's rely on authorization of a financial transaction by the card issuer or other authorizing institution via the communication network. This is often performed through an ISO 8583 messaging system. Many bank charges ATM usage fees from the customers for the transactions. At present every customer has an individual ATM card for each and every bank in which he/she maintains account. So handling the cards, their passwords play a major role here. So to overcome these difficulties we embedded more than one bank account of the user in a single ATM smart card, so that the user can swipe the card and can select the bank from which he/she are interested to carry out transaction.

TABLE OF CONTENT

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF FIGURES	VIII
	LIST OF TABLES	X
	LIST OF ABBREVIATIONS	XI

CHAPTER NO	TITLE	PAGE NO
1	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 SCOPE OF THE PROJECT	1
	1.3 EXISTING SYSTEM	1
	1.4 EXISTINGSYSTEM DISADVANTAGES	
2	LITERATURE SURVEY	2
	2.1 LITERATURE SURVEY	2
	2.2 LITERATURE SUMMARY	5
3	AIM AND SCOPE OF THE PRESENT INVESTIGATION	6
	3.1 PROPOSED SYSTEM	6
	3.1.1 PROPOSED SYSTEM ADVANTAGES	6
	3.2 BLOCK DIAGRAM	6
	3.3 MODULES NAME	7
	3.4 MODULE DESCRIPTION	7
	3.4.1 RFID IDENTIFICATION	7
	3.4.2 BIOMETRIC AUTHENTICATION	8
	3.4.3 TRANSACTION PROCESS	8
	3.4.4 ALERT MODULE	9
	3.5 BLOCK DIAGRAM DESCRIPTION	9

4	MATERIALS AND METHODS USED	
4.1	HARDWARE DESCRIPTION	10
4.1.1	ARDUINO	10
4.1.1.1	ARDUINO MEGA	12
4.1.2	GSM	25
4.1.2.1	FUNCTIONAL DESCRIPTION	28
4.1.3	POWER SUPPLY	29
4.1.4	RFID READER AND TAG	37
4.1.5	KEYPAD	39
4.1.6	LCD	42
4.1.7	FINGERPRINT SENSOR	46
4.1.7.1	OPERATION PRINCIPLE	47
4.1.7.2	FEATURES	47
4.1.8	BUZZER	53
4.1.8.1	BUZZER FEATURES & SPECIFICATIONS	53
4.1.9	SERVOMOTOR	54
4.1.9.1	FEATURES AND APPLICATIONS	56
4.1.9.2	SERVO MOTOR WORKING PRINCIPLE	57
4.2	SOFTWARE DESCRIPTION	59
4.2.1	EMBEDDED C	59
4.2.2	ARDUINO SOFTWARE IDE	59
5	RESULT AND DISCUSSION	
5.1	OUTPUT	69
6	CONCLUSION AND FUTURE WORKS	
6.1	CONCLUSION	81
6.2	ADVANTAGES	81
6.3	APPLICATIONS	82
6.4	FUTURE ENHANCEMENTS	82
	REFERENCE	83

LIST OF FIGURES

CHAPTER NO.	TITLE	PAGE NO
3.1	Block diagram	6
3.2	RFID identification	7
3.3	Biometric authentication	8
3.4	Transaction process	8
3.5	Alert module	9
4.1	Arduino mega board	12
4.2	Arduino mega	13
4.3	Ultrasonic sensor code	16
4.4	Arduino Mega 2560 PIN diagram	18
4.5	Power supply diagram	25
4.6	GSM module	26
4.7	System block diagram	27
4.8	Keypad connection SIM	32
4.9	SIM connection (no power control)	33
4.10	MIC input (differential input)	36
4.11	MIC input (single-ended input)	36
4.12	RFID reader	38
4.13	RFID card	38
4.14	Application circuit	39
4.15	Keypad	40
4.16	LCD	42
4.17	Data register	45
4.18	Finger print sensor	46
4.19	Hardware interface	48
4.20	Servo motor	58
4.21	block diagram representation of embedded system	59
4.22	block diagram of Embedded C Programming development	60
5.1	Overall circuit Connections	69
5.2	Using RFID Tag	69

5.3	Using Fingerprint sensor	70
5.4	Registration of fingerprint	70
5.5	Finger print sensor	70
5.6	Mobile no Registration process	71
5.7	Keypad	71
5.8	Mobile number Registration process	72
5.9	Mobile Number Access	72
5.10	Registering our Mobile Number	72
5.11	Otp is generated	73
5.12	Received otp	73
5.13	Enter the otp	74
5.14	Verification of otp	74
5.15	Deposit	74
5.16	Biometric process	75
5.17	Deposit	75
5.18	Selecting SBI bank(1)	75
5.19	Widraw	76
5.20	Select the Bank	77
5.21	Enter the Amount	77
5.22	Withdrawal amount and Balance	77
5.23	Checking BAL	78
5.24	view Balance	78
5.25	Place Finger	79
5.26	Fraud Access	79
5.27	Mismatched finger print	79
5.28	Un authorized	80
5.29	messages to the registered mobile number	80

LIST OF TABLES

CHAPTER NO	TITLE	PAGE NO
4.1	Technical Specifications	13
4.2	Arduino Mega 2560 PIN mapping table	18
4.3	Product Description	27
4.4	VBAT Pin Description	29
4.5	GND Pin Description	29
4.6	Power consumption in OFF mode	30
4.7	Power consumption in EGSM 900MHZ	30
4.8	Power consumption in DCS/PCS 1800MHZ	30
4.9	VCC Pin Description	31
4.10	SPI Pin Description	31
4.11	SIM Interface	32
4.12	Audio Interface	34
4.13	Microphone Electrical Characteristics	34
4.14	Speaker Electrical Characteristics	35
4.15	Pin detail of LCD display	43
4.16	Important Command Codes for LCD	45
4.17	Serial communication pin details	49
4.18	USB Communication	49
4.19	Buzzer Pin Configuration	53

LIST OF ABBREVIATION

1. **ATM** - Automatic Teller Machine
2. **RFID** - Radio Frequency Identification
3. **GSM** - Global System for Mobile Communication
4. **LCD** - Liquid Crystal Display
5. **IOT** - Internet of Things
6. **CLK** - Clock
7. **SIM** - Subscriber Identity Module Card
8. **CPU** - Central Processing Unit
9. **PIN** - Personal Identification Number
10. **LED** - Light Emitting Diode
11. **IDE** - Integrated Development Environment
12. **USB** - Universal Server Bus
13. **RAM** - Random Access Memory
14. **ROM** - Read Only Memory
15. **ADC** - Analog to Digital Convertor
16. **DAC** - Digital to Analog Convertor
17. **AVR** - Automatic Voltage Regulation
18. **TMS** - Transportation Management System
19. **SCL** - System Check List
20. **SPI** - Schedule Performance Index
21. **PCB** - Printed Circuit Board
22. **GPIO** - General Purpose Input/Output
23. **TXD** - Serial Data Transmit Pin
24. **RXD** - Serial Data Receive Pin
25. **SCI** - Serial Communication Interface
26. **UART** - Universal Asynchronous Receiver- Transmitter

CHAPTER 1

INTRODUCTION

1.1 GENERAL

An automated teller machine (ATM) is an electronic banking outlet that allows customers to complete basic transactions without the aid of a branch representative or teller. Anyone with a credit card or debit card can access cash at most ATMs.

ATMs are convenient, allowing consumers to perform quick self-service transactions such as deposits, cash withdrawals, bill payments, and transfers between accounts. Fees are commonly charged for cash withdrawals by the bank where the account is located, by the operator of the ATM, or by both. Some or all of these fees can be avoided by using an ATM operated directly by the bank that holds the account.

1.2 SCOPE OF THE PROJECT

The main aim of the project is to eliminate several ATM cards and to replace it with a single card which can be interfaced with multiple banks. This reduces the cards if multiple accounts are present.

1.3 EXISTING SYSTEM

In the existing system, when the user enters the ATM system the user has to insert the respective card of the user's choice into the ATM system's card slot. The next immediate step the user has to do is enter the pin for that particular card. There are multiple cards if the user has bank accounts and there is a possible chance of entering PIN's of other cards which may affect the transaction process.

1.3.1 EXISTING SYSTEM DISADVANTAGES

- Multiple Cards
- Security of ATM card is less

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

Title 1: ATM Security using GSM technology.

Authors: Sudhakar Hallur, Manjunath Bajantri, Sagar.

Year: 2018

Description

Automated Teller Machine (ATM)'s now a days are extensively used all over the world for the withdrawal of cash. A unique card is issued for each user along with the unique code provided to him so as to the person may do all his transactions personally without anyone getting known. Since transactions are extensively secure there is no much more security required but in countries like India it's very necessary to have a physical security to the machine. A provision to give physical security to the machine is being discussed over here in this paper presented over here.

Title 2: Implementation of Bank Security System using GSM and Internet of Things

Authors: Moturi Phalgun Satish, Bala Kishore. G

Year: 2017

Description

The Internet of Things (IoT) is one of the hottest topics in the technology sector, and with good reason. It influences the interaction of technological, economic, social, societal, and individual changes. Internet of things has been governing the electronics era with cloud services dominating the ever increasing electronics product segment. Security and safety has always become a basic necessity for urban population. The Implementation of the Bank Security System by Using GSM and IoT is developed into the security application. The main objective of this system is to develop an embedded system, which is used for ATM security applications. The embedded ATM authentication system is based on sensors & communication technology. In this paper

when any disturbance takes place for the ATM machine then information is send through IoT and door is automatically closed then it will send the machine, by alerting the surrounding area using buzzer, at the same time total data will be uploaded in web page using IoT and puts alert message to consult person. In the same way if any fire occurred controller puts a message to consult person and uploads same data in web page using IoT. It also counts how many persons are entering into the ATM centre. This concept is not only a single ATM center; we can consider this ATM center as node1 (like node1, node2, node3..... Etc) are connected to web page through IoT module (ESP 8266). So that if there is any disturbance or any fire accident in any node we can get the area information through IoT to the web page along with buzzer.

Title 3: Advanced Security Management System for ATM's using GSM and MEMS

Authors: Venka Reddy Maram, Mirza Sajid Ali Baig, Narsappa Reddy.

Year: 2015

Description

Security in ATM networks is necessary because ATM is widespread and many areas such as financial- or medical-applications, network-administration, etc. require very sensitive handling of the transmitted data. If we look at other fields of interest (only multimedia technologies and cable television shall be mentioned among others) we see that ATM channels might be used for billing. Misuse of the ATM network, manipulation of transmitted data, spoofing, or repudiation would be fatal in billing-/accounting-system. Robberies of ATM"s, misuse of credit cards and all. Our project is going to concentrate on the ATM security system. Whenever a thief enters and tries to touch an ATM forcefully. The movement will be observed by the MEMS sensor. While MEMS observe the movement it sends a request to the microcontroller. Microcontroller will automatically lock the door which is represented with the help of DC motor and send a message to the authority through GSM. It will produce sound with the help of buzzer to alert the security. And this door will be unlocked with the switch which is present outside the room.

Title 4: ATM machine security system using GSM and MEMS sensor

Authors: Prof.Shinde S.P, Chingale R.R., Dhane D.C.,Vader P.B.

Year: 2017

Description

The Idea of Designing and of Security Based ATM machine security system project is born with the observation in our real life examples happening around us. This project overcomes the drawback of older technology used in our society. Project deals with the security of ATM machine. Whenever robbery occurs, Vibration sensor and MEMS sensor is used here which senses vibration produced from ATM machine and the movement of ATM machine. GSM is used to send the message to the police station and authority. In this project we uses PIC controller which is based on embedded system process real time data collected using the MEMS sensor and vibration sensor. When the movement of machine and the vibration is sensed the beep sound will occur from the buzzer. For closing the door of ATM DC Motor is used. Smoke detector is used here to sense the gaseous or smoke near ATM machine. The use of camera is always in processing and sending video continuous to the Pc and it will be saved. Here LCD display board shows the status of our project. It will prevent the robbery occur. Here, MPLAB tools are used to implement the idea and results are obtained. MPLAB tools are used for run the DC motor for automatic door lock.

Title 5: Advance Security System for ATM

Authors: Aman Kumar.

Year: 2015

Description

In present time ATM (Automatic Teller Machine) robbery is the common thing, because we have not strong security and rule or regulation to withdraw the cash from ATM with help of ATM card. In this paper I have provide the solution of robbery of ATM to use various technology, rules and regulation. This is common weakness in ATM security facility, remove this problem we used following technology to improve security system .This technology work in main three steps or phases, in first phase

user need to swap the ATM and used palm for scanning after successfully swap machine will generate voice you can enter for 2nd phase and palm will save in that transaction database. When you go for 2nd phase then need to scan the retina and then go for phase 3rd. there will final ATM. In performing the transaction gate will closed. The wall of phases will of strong fiber that cannot cut by any Gas. The final phase has Gas that can power for unconscious. And all CCTV cameras monitored by near police station. When any thief does robbery in ATM then alarm generate in police station and Gas will leak in near ATM. Then in this situation all information like, robbery time, date, CCTV data, voice data will stored in server. The server monitored by state level. This paper shows only how three phases will work and generate alarm in near police station & send police in robbery situations.

2.2 LITERATURE SUMMARY

As we have mentioned in the above literature surveys in the past years ,There is gradual development in the accessibility of ATM card and necessity of improving the user friendly and Secure domain in the ATM cards and It evolving method of using it for multiple bank accounts So, We have initiated the Fingerprint Security for out Project in order to differentiate and stand alone from the previous projects The fingerprint technology is the most widely accepted and mature biometric method and is the easiest to deploy and for a higher level of security at your fingertips. It is simple to install and also it takes little time and effort to acquire one's fingerprint with a fingerprint identification device .

Thus, finger print recognition is considered among the least intrusive of all biometric verification techniques. The finger print features are different for each human being so the user can be identified uniquely. Instead of using multiple ATM card fingerprint based ATM is safer and secure. There is no worry of losing ATM card and no need to carry ATM card in your wallet. We just have to use our fingerprint in order to do any banking transaction.

CHAPTER 3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 PROPOSED SYSTEM

The proposed scheme of MAASC (Multiple Account Access using Single ATM Card) provides the individual, the comfort of accessing users multiple accounts of different banks using a single card. Also, it provides the user one level higher convenience than the existing system. The transactions are approved by biometric authentication of the user which is replaced instead of PIN. As the bank server are different so the linking of single card to multiple server won't be a tough process.

3.1.1 PROPOSED SYSTEM ADVANTAGES

- Single ATM card provides more convince of using multiple bank transactions
- Higher security based on the biometric module
- More secure transaction using biometric of the user.

3.2 BLOCK DIAGRAM

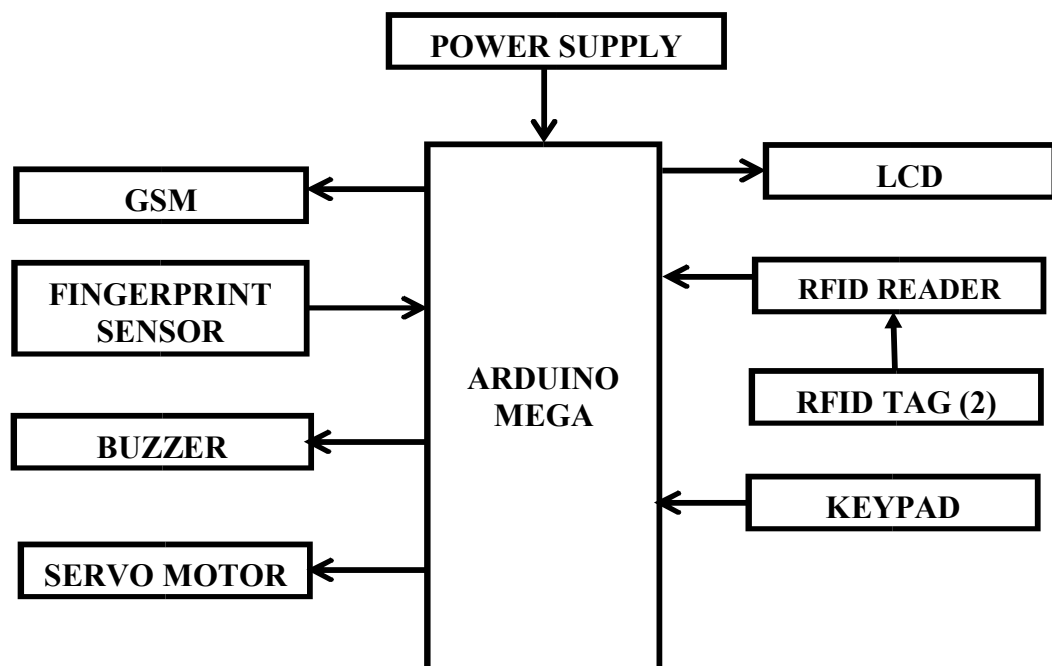


Fig 3.1: Block diagram

3.3 MODULES NAME

- ❖ RFID Identification
- ❖ Biometric Authentication

- ❖ Transaction Process
- ❖ Alert Module

3.4 MODULE DESCRIPTION

In this project four modules are there.

They are:

- RFID identification
- Biometric module
- Transaction process
- Alert module

3.4.1 RFID IDENTIFICATION

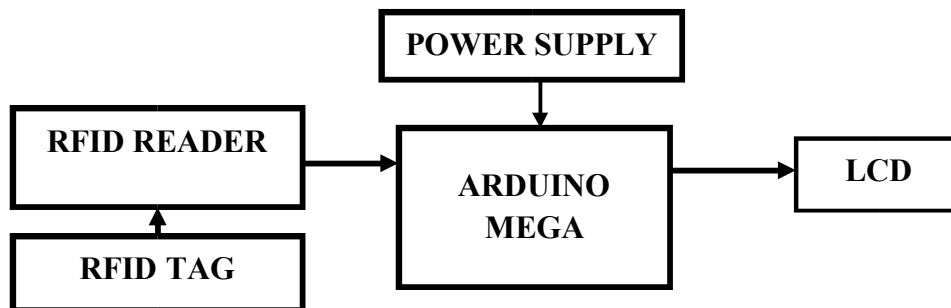


Fig 3.2: RFID IDENTIFICATION

This module which is used to identify the user of the ATM through RFID tag and reader. The user of the ATM is identified using this instead of magnetic strips that is being used in the existing method. This is displayed in the LCD. The Microcontroller controls the overall process of this system.

3.4.2 BIOMETRIC AUTHENTICATION

The biometric authentication is one of the feature where the security is enhanced. After The user of the ATM card ensures their identity by RFID followed by their biometric to continue the transaction process. If the biometric and the RFID matches then the user interface of the prototype directs the user to select the bank to which they have been registered.

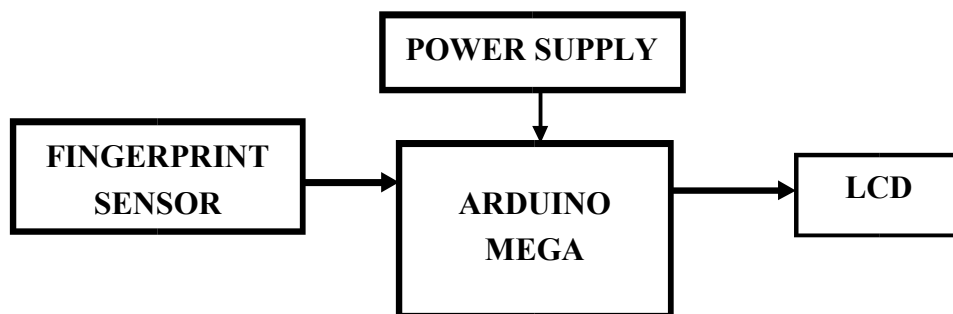


Fig 3.3: BIOMETRIC AUTHENTICATION

3.4.3 TRANSACTION PROCESS

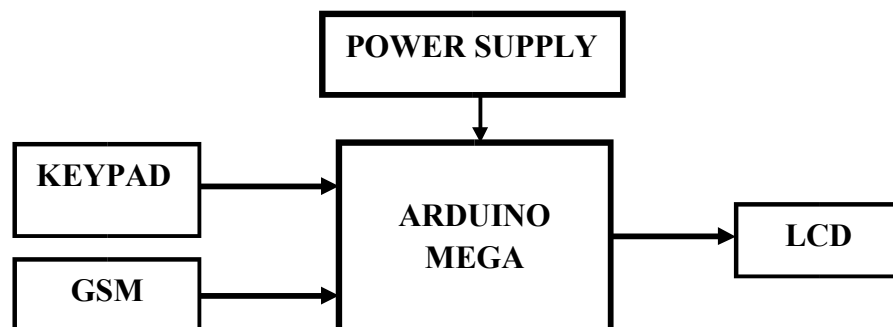


Fig 3.4: TRANSACTION PROCESS

After the authentication process the user is directed to select the bank that the user have already registered. The user can use the keypad to select the bank according to the banks displayed in the LCD. The transaction process is the same as the normal ATM after the selection of the bank. The balance, debit and credit status is displayed in the LCD so that user can select it to this convenience.

3.4.4 ALERT MODULE

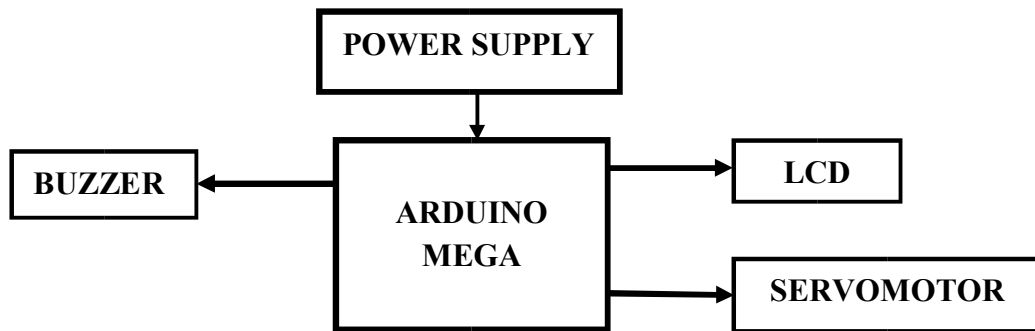


Fig 3.5: ALERT MODULE

If there is any mismatch in the rfid or the biometric then the alarm is triggered through the microcontroller and a warning message is displayed in the LCD and same is sent as a SMS to the user of the card. If the authentication process is successful, then after the transaction process a servomotor is triggered to show the transaction is successful and a SMS is also sent to the user about the transaction and balance.

3.5 BLOCK DIAGRAM DESCRIPTION AND WORKING

In the proposed method, the magnetic strip based ATM card is replaced with RFID based card which have a unique number. The Arduino MEGA microcontroller is used to process the data from the sensor. The fingerprint module is used to authenticate the user and the user needs to enter the One Time Password (OTP) which is generated by the controller and it is sent to the registered user mobile phone. The user can register the bank details and also withdraw the amount from the registered bank details. Hence this system provides more secure and multiple bank account using single ATM card.

CHAPTER 4

MATERIALS AND MATHOD USED

4.1 HARDWARE DESCRIPTION:

4.1.1 ARDUINO

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

WHY ARDUINO?

Thanks to its simple and accessible user experience, Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to

prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduino community.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than \$50.
- **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from

Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.

➤ **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

4.1.1.1 ARDUINO MEGA

The MEGA 2560 is designed for more complex projects. With 54 digital I/O pins, 16 analog inputs and a larger space for your sketch it is the recommended board for 3D printers and robotics projects. This gives your projects plenty of room and opportunities.



Fig: 4.1 Arduino mega board

The **Arduino Mega 2560** is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, 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. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.

TABLE 4.1 :TECHNICAL SPECIFICATIONS:

Microcontroller	<u>ATmega2560</u>
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by boot loader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weight	37 g

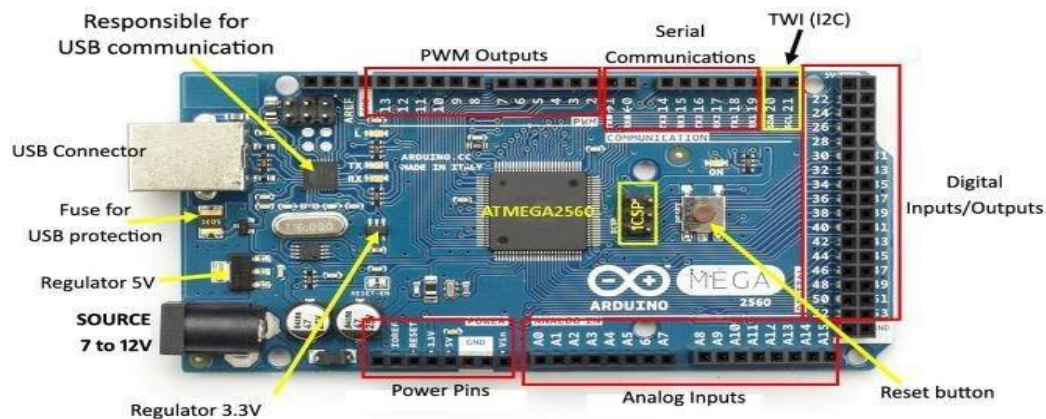


Fig 4.2: Arduino mega

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available.

Although the hardware and software designs are freely available under copyleft licenses, the developers have requested the name Arduino to be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product. Several Arduino-compatible products commercially released have avoided the project name by using various names ending in -duino. An early Arduino board with an RS-232 serial interface (upper left) and an Atmel ATmega8 microcontroller chip (black, lower right); the 14 digital I/O pins are at the top, the 6 analog input pins at the lower right, and the power connector at the lower left.

Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an I²C serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the LilyPad, run at 8 MHz and dispense with the onboard voltage regulator due to specific formfactor restrictions.

Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default bootloader of the Arduino UNO is the optiboot bootloader. Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between RS-232 logic levels and transistor–transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE,

standard AVR in-system programming (ISP) programming is used. An official Arduino Uno R2 with descriptions of the I/O locations

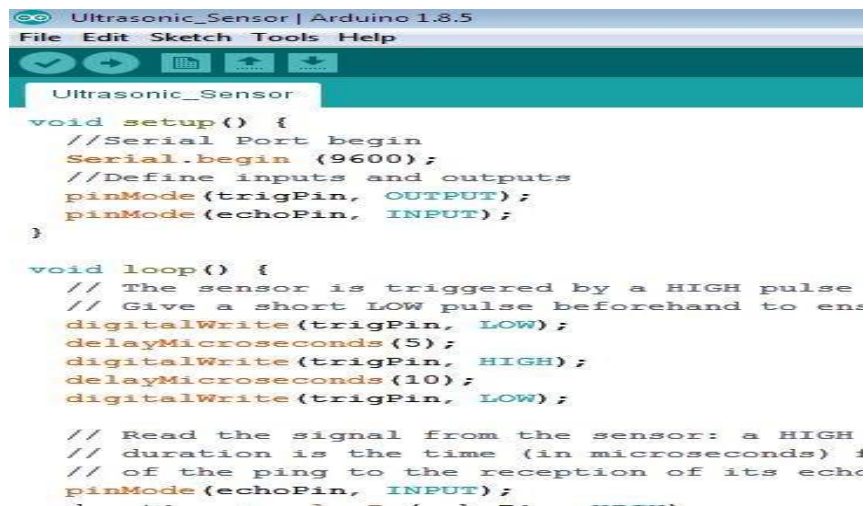
The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.

The Mega 2560 board can be programmed with the Arduino Software (IDE). For details, see the reference and tutorials.

The ATmega2560 on the Mega 2560 comes preprogrammed with a boot loader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar; see these instructions for details.



```

Ultrasonic_Sensor | Arduino 1.8.5
File Edit Sketch Tools Help

Ultrasonic_Sensor

void setup() {
  //Serial Port begin
  Serial.begin (9600);
  //Define inputs and outputs
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop() {
  // The sensor is triggered by a HIGH pulse
  // Give a short LOW pulse beforehand to ens
  digitalWrite(trigPin, LOW);
  delayMicroseconds(5);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Read the signal from the sensor: a HIGH
  // duration is the time (in microseconds) d
  // of the ping to the reception of its echo
  pinMode(echoPin, INPUT);
}

```

Fig 4.3 Ultrasonic sensor code

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the Arduino repository. 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. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU boot loader). See this user-contributed tutorial for more information.
- **POWER:** The Mega 2560 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 (wallwart) or battery. The adapter can be connected by plugging a 2.1mm centerpositive 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

become 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 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 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 ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the EEPROM library).
- **INPUT AND OUTPUT:** ATmega2560-Arduino Pin Mapping.

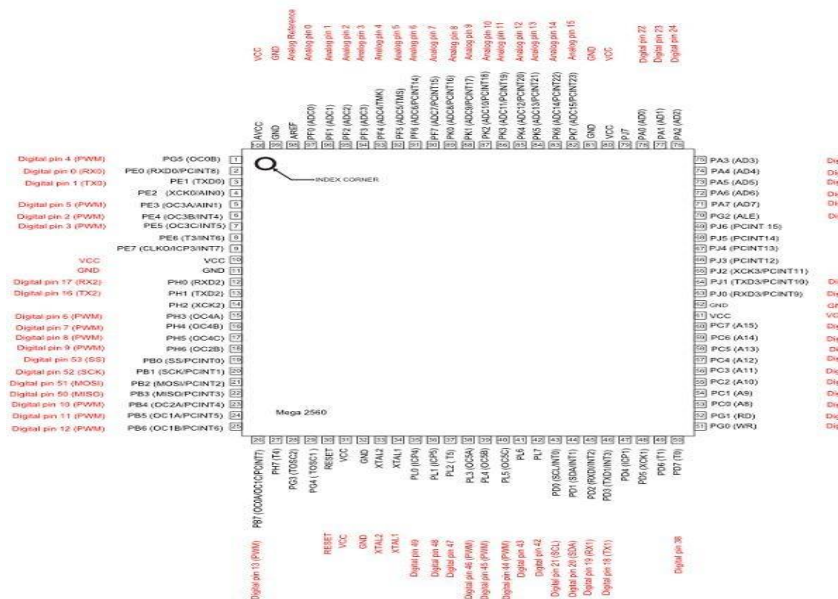


Fig 4.4: Arduino Mega 2560 PIN diagram

Above is the pin mapping for the Atmega2560. The chip used in Arduino 2560. There are pin mappings to Atmega8 and Atmega 168/328 as well.

TABLE 4.2 Arduino Mega 2560 PIN mapping table

Pin Number	Pin Name	Mapped Pin Name
1	PG5 (OC0B)	Digital pin 4 (PWM)
2	PE0 (RXD0/PCINT8)	Digital pin 0 (RX0)
3	PE1 (TXD0)	Digital pin 1 (TX0)
4	PE2 (XCK0/AIN0)	
5	PE3 (OC3A/AIN1)	Digital pin 5 (PWM)
6	PE4 (OC3B/INT4)	Digital pin 2 (PWM)
7	PE5 (OC3C/INT5)	Digital pin 3 (PWM)
8	PE6 (T3/INT6)	
9	PE7 (CLKO/ICP3/INT7)	
10	VCC	VCC
11	GND	GND
12	PH0 (RXD2)	Digital pin 17 (RX2)
13	PH1 (TXD2)	Digital pin 16 (TX2)

14	PH2 (XCK2)	
15	PH3 (OC4A)	Digital pin 6 (PWM)
16	PH4 (OC4B)	Digital pin 7 (PWM)
17	PH5 (OC4C)	Digital pin 8 (PWM)
18	PH6 (OC2B)	Digital pin 9 (PWM)
19	PB0 (SS/PCINT0)	Digital pin 53 (SS)
20	PB1 (SCK/PCINT1)	Digital pin 52 (SCK)
21	PB2 (MOSI/PCINT2)	Digital pin 51 (MOSI)
22	PB3 (MISO/PCINT3)	Digital pin 50 (MISO)
23	PB4 (OC2A/PCINT4)	Digital pin 10 (PWM)
24	PB5 (OC1A/PCINT5)	Digital pin 11 (PWM)
25	PB6 (OC1B/PCINT6)	Digital pin 12 (PWM)
26	PB7 (OC0A/OC1C/PCINT7)	Digital pin 13 (PWM)
27	PH7 (T4)	
28	PG3 (TOSC2)	
29	PG4 (TOSC1)	
30	RESET	RESET
31	VCC	VCC
32	GND	GND
33	XTAL2	XTAL2
34	XTAL1	XTAL1
35	PL0 (ICP4)	Digital pin 49

36	PL1 (ICP5)	Digital pin 48
37	PL2 (T5)	Digital pin 47
38	PL3 (OC5A)	Digital pin 46 (PWM)
39	PL4 (OC5B)	Digital pin 45 (PWM)
40	PL5 (OC5C)	Digital pin 44 (PWM)
41	PL6	Digital pin 43
42	PL7	Digital pin 42
43	PD0 (SCL/INT0)	Digital pin 21 (SCL)

44	PD1 (SDA/INT1)	Digital pin 20 (SDA)
45	PD2 (RXDI/INT2)	Digital pin 19 (RX1)
46	PD3 (TXD1/INT3)	Digital pin 18 (TX1)
47	PD4 (ICP1)	
48	PD5 (XCK1)	
49	PD6 (T1)	
50	PD7 (T0)	Digital pin 38
51	PG0 (WR)	Digital pin 41
52	PG1 (RD)	Digital pin 40
53	PC0 (A8)	Digital pin 37
54	PC1 (A9)	Digital pin 36
55	PC2 (A10)	Digital pin 35
56	PC3 (A11)	Digital pin 34
57	PC4 (A12)	Digital pin 33
58	PC5 (A13)	Digital pin 32
59	PC6 (A14)	Digital pin 31
60	PC7 (A15)	Digital pin 30
61	VCC	VCC
62	GND	GND
63	PJ0 (RXD3/PCINT9)	Digital pin 15 (RX3)
64	PJ1 (TXD3/PCINT10)	Digital pin 14 (TX3)
65	PJ2 (XCK3/PCINT11)	
66	PJ3 (PCINT12)	
67	PJ4 (PCINT13)	
68	PJ5 (PCINT14)	

69	PJ6 (PCINT 15)	
70	PG2 (ALE)	Digital pin 39
71	PA7 (AD7)	Digital pin 29
72	PA6 (AD6)	Digital pin 28
73	PA5 (AD5)	Digital pin 27

74	PA4 (AD4)	Digital pin 26
75	PA3 (AD3)	Digital pin 25
76	PA2 (AD2)	Digital pin 24
77	PA1 (AD1)	Digital pin 23
78	PA0 (AD0)	Digital pin 22
79	PJ7	
80	VCC	VCC
81	GND	GND
82	PK7 (ADC15/PCINT23)	Analog pin 15
83	PK6 (ADC14/PCINT22)	Analog pin 14
84	PK5 (ADC13/PCINT21)	Analog pin 13
85	PK4 (ADC12/PCINT20)	Analog pin 12
86	PK3 (ADC11/PCINT19)	Analog pin 11
87	PK2 (ADC10/PCINT18)	Analog pin 10
88	PK1 (ADC9/PCINT17)	Analog pin 9
89	PK0 (ADC8/PCINT16)	Analog pin 8
90	PF7 (ADC7)	Analog pin 7
91	PF6 (ADC6)	Analog pin 6
92	PF5 (ADC5/TMS)	Analog pin 5
93	PF4 (ADC4/TMK)	Analog pin 4
94	PF3 (ADC3)	Analog pin 3
95	PF2 (ADC2)	Analog pin 2
96	PF1 (ADC1)	Analog pin 1
97	PF0 (ADC0)	Analog pin 0
98	AREF	Analog Reference
99	GND	GND
100	AVCC	VCC

Each of the 54 digital pins on the Mega 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 20 mA as recommended operating condition and has

an internal pull-up resistor (disconnected by default) of 20-50 k ohm. A maximum of 40mA is the value that must not be exceeded to avoid permanent damage to the microcontroller.

In addition, some pins have specialized functions:

- **Serial:** 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega16U2 USB-to-TTL Serial chip.
- **External Interrupts:** 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low level, a rising or falling edge, or a change in level. See the `attachInterrupt()` function for details.
- **PWM:** 2 to 13 and 44 to 46. Provide 8-bit PWM output with the `analogWrite()` function.
- **SPI:** 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication using the SPI library. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Arduino /Genuino Uno and the old Duemilanove and Diecimila Arduino boards.
- **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.
- **TWI:** 20 (SDA) and 21 (SCL). Support TWI communication using the Wire library. Note that these pins are not in the same location as the TWI pins on the old Duemilanove or Diecimila Arduino boards.

See also the mapping Arduino Mega 2560 PIN diagram.

The Mega 2560 has 16 analog inputs, 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.

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 Mega 2560 board has a number of facilities for communicating with a computer, another board, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega16U2 (ATmega 8U2 on the revision 1 and revision 2 boards) on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the

ATmega8U2/ATmega16U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1.

A `SoftwareSerial` library allows for serial communication on any of the Mega 2560's digital pins.

The Mega 2560 also supports TWI and SPI communication. The Arduino Software (IDE) includes a `Wire` library to simplify use of the TWI bus; see the documentation for details. For SPI communication, use the `SPI` library.

- **AUTOMATIC (SOFTWARE) RESET:** Rather than requiring a physical press of the reset button before an upload, the Mega 2560 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 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted

(taken low), the reset line drops long enough to reset the chip. The Arduino Software (IDE) 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 wellcoordinated with the start of the upload.

This setup has other implications. When the Mega 2560 board 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 bootloader is running on the ATmega2560. 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 Mega 2560 board 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; see this forum thread for details.

- **REVISIONS:** The Mega 2560 does not use the FTDI USB-to-serial driver chip used in past designs. Instead, it features the ATmega16U2 (ATmega8U2 in the revision 1 and revision 2 Arduino boards) programmed as a USB-to-serial converter. Revision 2 of the Mega 2560 board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

Revision 3 of the Arduino board and the current Genuino Mega 2560 have the following improved features:

- **1.0 pinout:** SDA and SCL pins - near to the AREF pin - and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the board that uses ATSAM3X8E, that operate with 3.3V. The second one is a not connected pin that is reserved for future purposes.
- Stronger RESET circuit.

- Atmega 16U2 replace the 8U2.

APPLICATIONS:

- Arduboy, a handheld game console based on Arduino.
- Arduinome, a MIDI controller device that mimics the Monome.
- Ardupilot, drone software and hardware.
- ArduSat, a cubesat based on Arduino.
- C-STEM Studio, a platform for hands-on integrated learning of computing, science, technology, engineering, and mathematics (C-STEM) with robotics.
- Data loggers for scientific research.
- OBDuino, a trip computer that uses the on-board diagnostics interface found in most modern cars.

4.1.2 POWER SUPPLY

This section describes how to generate +5V DC power supply

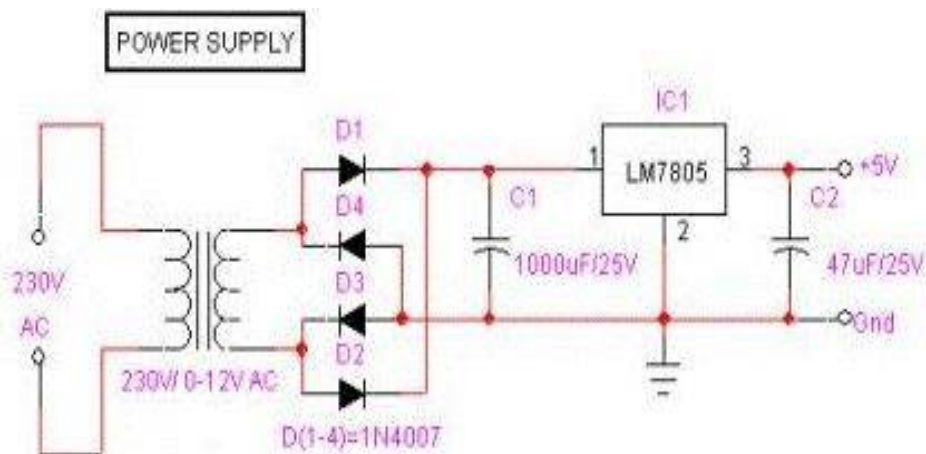


Fig 4.5: Power supply diagram

The power supply section is the important one. It should deliver constant output regulated power supply for successful working of the project. A 0-12V/1 mA transformer is used for this purpose. The primary of this transformer is connected in to main supply through on/off switch& fuse for protecting from overload and short circuit protection. The secondary is connected to the diodes to convert 12V AC to 12V

DC voltage. And filtered by the capacitors, which is further regulated to +5v, by using IC 7805.

4.1.2 GSM

A GSM modem is a wireless modem that works with a GSM wireless

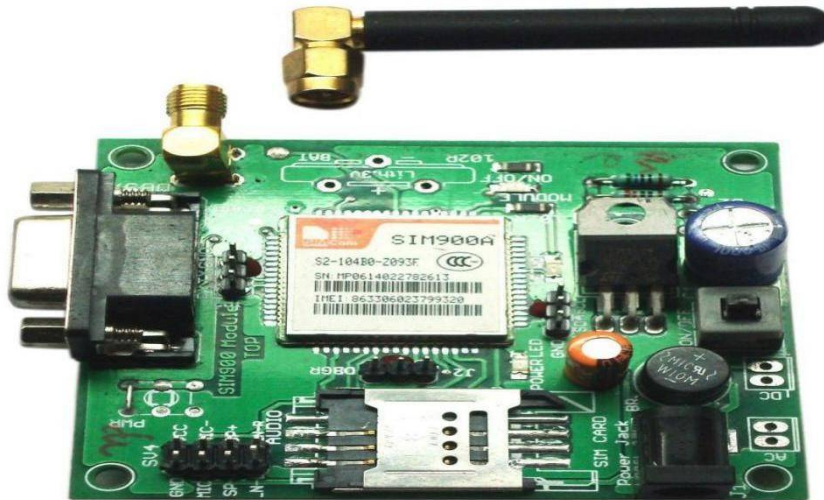


Fig 4.6: GSM module

network. A wireless modem behaves like a dial-up modem. The main difference between them is that a dial-up modem sends and receives data through a fixed telephone line while a wireless modem sends and receives data through radio waves. The working of GSM modem is based on commands, the commands always start with AT (which means ATtention) and finish with a <CR> character. For example, the dialing command is ATD<number>; ATD3314629080; here the dialing command ends with semicolon.

The AT commands are given to the GSM modem with the help of PC or controller. The GSM modem is serially interfaced with the controller with the help of MAX 232. Here max 232 acts as driver which converts TTL levels to the RS 232 levels. For serial interface GSM modem requires the signal based on RS 232 levels. The T1_OUT and R1_IN pin of MAX 232 is connected to the TX and RX pin of GSM modem

- **OBJECT OF THE DOCUMENT:** This document gives an overview of the LinkSprite GSM/GPRS module: a miniature, single-side board, quad-band GSM

850/EGSM 900/DCS 1800/PCS 1900 module, ready for integration in various kinds of Fix wireless phones and other wireless devices.

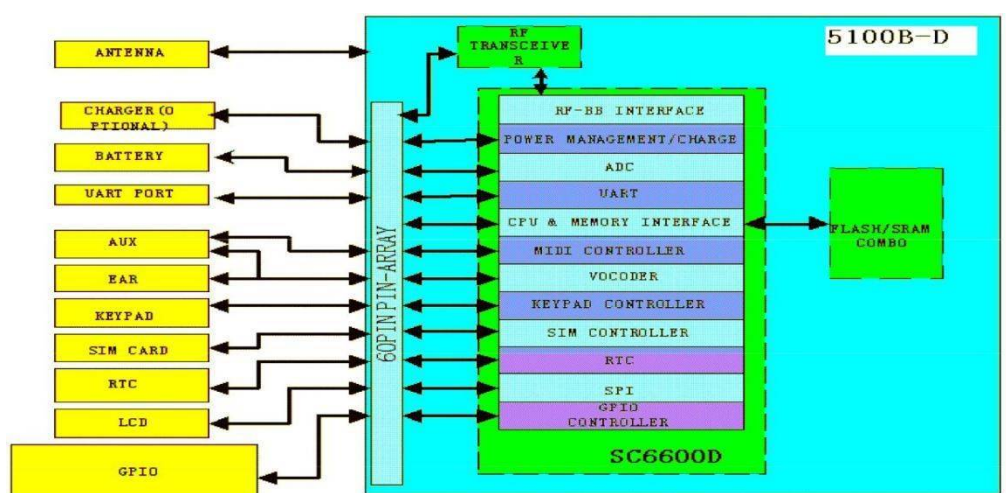


Fig 4.7: System block diagram

TABLE 4.3: PRODUCT DESCRIPTION

Temperature range	Normal range: -10°C to +55°C (Full compliant) Storage: -40°C to +85°C
Weight	< 9g
Physical dimensions	35.0X39.0X2.9 mm (typical)
Connection	60 pins
Power supply	VBAT: 3.3V to 4.2V range, 3.6V typical.
Power consumption	Off mode: <100uA Sleep mode: <2.0mA Idle mode: <7.0mA (average) Communication mode: 350 mA (average GSM) Communication mode: 2000mA (Typical peak during TX slot GSM)
Li-ion Battery charging management and interface (OPTION)	Li-ion Battery charging management is included. The charger interface is provided on 60-pin connector. (only for 3.7V Li-ion Battery)
Frequency bands	EGSM900 + GSM850 + DCS1800 + PCS1900
Transmit power	Class 4 (2W) for EGSM900/GSM850 Class 1 (1W) for DCS1800/PCS1900
Supported SIM card	3V/1.8V SIM card. (auto recognise)
Keyboard interface	4x6 keyboard interface is provided Up to 460 kbps
UART0 interface with flow control	Full hardware flow control signals (+3.0V) are provided on 60 pins.
UART1 interface without flow control	2-Wire UART interface Up to 460 kbps.
LCD interface	Support standard SPI interface

4.1.2.1 FUNCTIONAL DESCRIPTION

- **RF Functionalities:** The RF part of this module converts RF signals to baseband for receiver chain and translates base band signals into RF frequency spectrum.

The operating frequencies are:

Rx (EGSM 850): 869 to 894MHz

Tx (EGSM 850): 824 to 849MHz

Rx (EGSM 900): 925 to 960MHz

Tx (EGSM 900): 880 to 915MHz

Rx (DCS 1800): 1805 to 1880MHz

Tx (DCS 1800): 1710 to 1785MHz

Rx (PCS 1900): 1930 to 1990MHz

Tx (PCS 1900): 1850 to 1910MHz

- **Baseband Functionalities:** The baseband part of SM5210 is composed of a SPREADTRUM's SC6600D chip. This chipset is using 0.18μm mixed signal CMOS technology which allows massive integration as well as low power consumption. SC6600D provides single-chip solution to wireless Quad-band telephone handsets and data modems confirming to the EGSM 900, GSM 850, DCS 1800 and PCS 1900.
- **Memory Functionalities :** The memory used in this module is a combination of 32 Megabit (2M x 16-Bit) CMOS 3.0V Volt-only, simultaneous operation flash memory and 4 Megabit (256K x 16-Bit) static RAM.

4.1.3 POWER SUPPLY

Power Supply Description

The power supply is one of the key issues in the designing GSM terminals. Due to the 577us radio burst emission in GSM every 4.615ms, power supply must be able to deliver high current peaks in a short time. During these peaks, ripples and drops on the supply voltage must not exceed a certain limit.

Table 4.4: VBAT Pin Description

	Pin number	VMIN	VNOM	VMAX	Ripple max
VBAT (V)	1,3,5,7	3.3V (*)	3.6V	4.2V (**)	50mVPP for freq<200KHZ 2mVPP for freq>200KHZ

(*): This value has to be guaranteed during the burst (with 2.0A Peak in GSM or GPRS mode).

(**): max operation Voltage Stationary Wave Ratio.

When supplying the module with a battery, the total impedance (battery + protections + PCBS) should be less than 150mOhms.

Table : 4.5 GND Pin Description

	Pin number	Description
GND	9,11,13,15,17 Shielding can install pin(*)	Ground

Table 4.6: Power consumption in OFF mode

Mode	Condition	I(Typical)	I(Max)
Off mode	off	50 uA	100uA

Table 4.7 : Power consumption in EGSM 900MHZ

Mode	Condition	I(Typical)	I(Max)
GSM850/900mode	During TX bursts @PCL5*	1.6 A	2 A
GSM850/900mode	During RX bursts	100 mA	130 mA
GSM850/900mode	Average @PCL5*	350 mA	400 mA
GSM850/900mode	Average @PCL8*	250 mA	300 mA
GSM850/900mode	Average idle mode	13 mA	20 mA

(*): PCL: Power control level

PCL0: 1W emission requested (max power)

PCL3: 0.25W emission requested

Table 4.8 : Power consumption in DCS/PCS 1800MHZ

Mode	Condition	I(Typical)	I(Max)
DCS/PCS mode	During TX bursts @PCL0*	1.4 A	1.6 A
DCS/PCS mode	During RX bursts	100 mA	130 mA
DCS/PCS mode	Average @PCL0*	300 mA	350 mA
DCS/PCS mode	Average @PCL3*	200 mA	250 mA
DCS/PCS mode	Average idle mode	13 mA	20 mA

- **VCC OUTPUT** : This output can be used to power some external functions, such as LCD.

Table 4.9 : VCC Pin Description

Module Pin No.	Pin name	I/O type	Description
47 , 60	VCC	Supply	2.8V Power supply

Table 4.8 : VCC Electrical Characteristics

Parameter	Unit	Min.	Typ.	Max.
Output Voltage	V	2.5	2.8	3.0
Output Current	mA	Supply		50

- **SPI INTERFACE** : The SPI bus includes a CLK signal, an IO signal, a RS signal and a CS signal complying with the standard SPI bus. These pins can also be used as GPIO.

Table 4.10 : SPI Pin Description

Module Pin No.	pin name	I/O type	description
33	GPIO42/LCD_DATA/NB OT	O	SPI Data
38	GPIO6/LCD_CL K	O	SPI Serial Clock
39	GPIO1/LCD_R	O	SPI Reset

	ST		
41	GPIO9LCD_RS	O	Command/D ata Mode Select
54	GPIO8/LCD_C S	O	SPI Enable

- **KEYPAD INTERFACE** : This interface provides 10 connections: 4 rows (KBR0 to KBR3) and 6 columns (KBC0 to KBC5).

No external pull-up resistors are needed when the internal pull-up at the input pads are enabled.

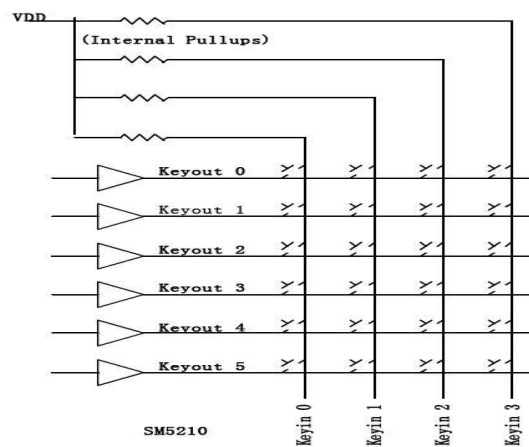


Fig 4.8: Keypad connection SIM

- **General Description:** The SIM interface is compatible with the ISO 7816-3 IC card standard required by the GSM 11.11 Phase 2+ standard. The module also supports Release 99 of the SIM Toolkit recommendation and supports a Fixed Dialling Number directory.

TABLE 4.11: SIM Interface

Pin number	Type	Description
29	I/O	SIM Serial Data
27	O	SIM Clock
21	O	SIM Reset
51	P	SIM Power Supply
Parameter	Max	Remarks
VL (V)	0.6/0.8	
VH (V)	-	

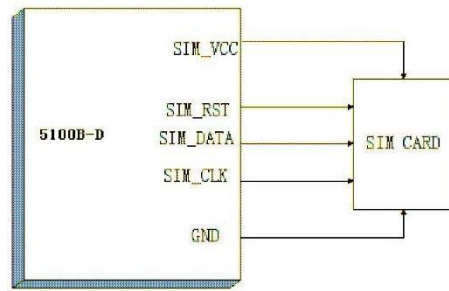


Fig 4.9: SIM connection (no power control)

Figure 3 shows system design for 3.0V and 1.8V SIM card interface. The SIM level shift is built in the SC6600D chip.

Note:

- 1) SIM_VCC need add three capacitor: 4.7uF, 0.1uF, 27pF.
- 2) SIM_RST, SIM_DATA, SIM_CLK lines also need add 27pF capacitor for each line.
- 3) SIM_DATA need to Pull-up with 2K resistor to SIM_VCC.

NOTES: Does not support 5V SIM card.

AUDIO

The module supports the following voice codec:

- Full-Rate
- Enhanced Full Rate

Two different microphone inputs and two different speaker outputs are supported.

Table 4.12 : Audio Interface

Pin number	Pin name	Type	Description
18	EAR_SPKP	AO	Earphone Differential Positive Output

16	EAR_SPKN	AO	Earphone Differential Negative Output
14	EAR_MICP	AI	Microphone Differential Positive Input
12	EAR_MICN	AI	Microphone Differential Negative Input
44	AUX_SPKP	AO	Auxiliary Speaker Differential Positive Output
58	AUX_SPKN	AO	Auxiliary Speaker Differential Negative Output
42	AUX_MICP	AI	Auxiliary Microphone Differential Positive Input
50	AUX_MICN	AI	Auxiliary Microphone Differential Negative Input

Their frequency response curve should be within the mask given by GSM specification.

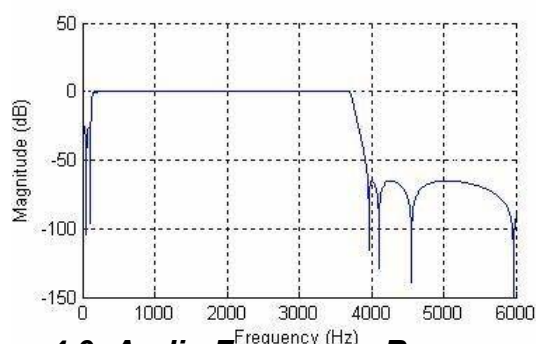


Figure 4.9: Audio Frequency Response Curve

Typical Microphone Inputs Characteristics

The gain of MIC inputs is internally adjusted and can be turned from 9dB to 42dB.

Table 4.13: Microphone Electrical Characteristics

Parameter	Conditions/Comments	Min.	Typ.	Units
Input common mode range	EAR_SPKP	0.6		V
PGA gain range		9		dB
Maximum analog input range after PGA	Differential			V
Digital audio output sample rate			8	kHz
ADC resolution		13		bits
ADC signal to (noise + distortion) ratio				dB
Digital output word			16	bits

- **Typical Speaker Outputs Characteristics:** The gain of speaker outputs is internally adjusted and can be turned from -21dB to 12dB.

Table 4.14 : Speaker Electrical Characteristics

Parameter	Conditions/Comments	Min.	Typ.	Max.	Units
Maximum differential output voltage			±1.4		V
Maximum single-ended output voltage			1.4		V
Output common mode voltage			1.1		V
Minimum load resistance		16			Ω
gain range		-21		12	dB

- **Typical Microphone Connection:** Two MIC inputs already include the biasing for an electret microphone allowing easy connection to the headset.

The connection can be either differential or single ended; but using a differential connection in order to reject common mode noise TDMA noise is strongly recommended.

- Recommended characteristics for the microphone Type: Electret
- Impedance: 2.2Kohm
- Sensitivity: -42dB(typ)
- SNR > 50dB min
- Frequency response compatible with the GSM specifications

- **Differential Connection:** The recommended characteristic of capacitors:

C1=22pF to 47pF

C2=C3=C4=47pF to 100pF

L1=L2=100nH

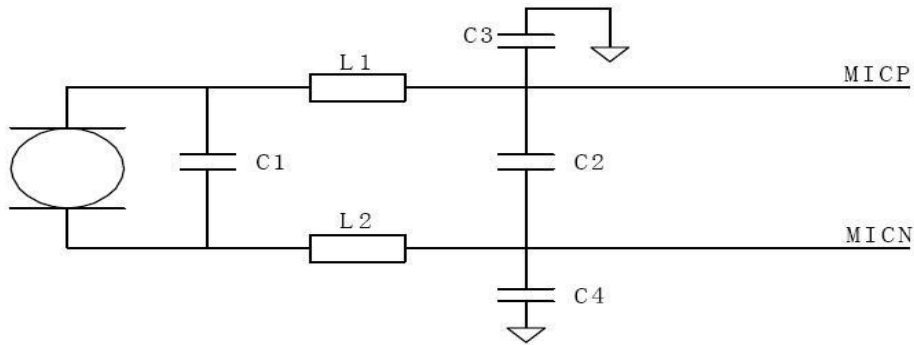


Fig 4.10: MIC input (differential input)

Microphone manufacturer can provide a microphone soldered C1 directly on it, so that C1 can be the nearest to the Microphone.

Single-ended Connection

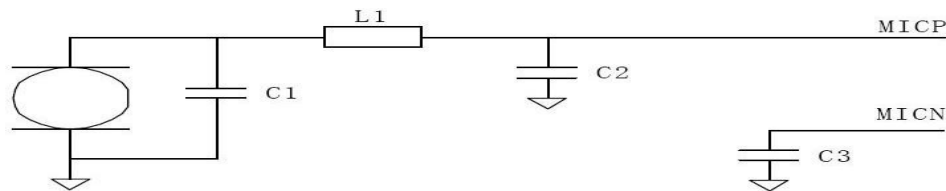


Fig 4.11: MIC input (single-ended input)

Do not recommend this connection.

When using a single-ended connection, be sure to have a very good ground plane, a very good filtering as well as shielding in order to avoid any disturbance on the audio path.

The recommended characteristic:

$C1 = 22\text{pF}$ to 47pF

$C2 = C3 = 47\text{pF}$ to 100pF

$L1 = 100\text{nH}$

Microphone manufacturer can provide a microphone soldered C1 directly on it, so that C1 can be the nearest to the Microphone.

➤ **Typical Speaker Connection :**

- Recommended characteristics for the speaker
- Type: 10 to 100mW, electro-magnetic
- Impedance: 16 to 150 Ohm
- Sensitivity: 110dB SPL min. (0dB = 120uPa)
- Frequency response compatible with the GSM specifications.

4.1.4 RFID READER AND TAG

An RFID reader is a device that is used to interrogate an RFID tag. The reader has an antenna that emits radio waves; the tag responds by sending back its data.

An RFID tag is a microchip combined with an antenna in a compact package; the packaging is structured to allow the RFID tag to be attached to an object to be tracked. "RFID" stands for Radio Frequency Identification. The tag's antenna picks up signals from an RFID reader or scanner and then returns the signal, usually with some additional data (like a unique serial number or other customized information).

A passive tag is an RFID tag that does not contain a battery; the power is supplied by the reader. When radio waves from the reader are encountered by a passive rfid tag, the coiled antenna within the tag forms a magnetic field. The tag draws power from it, energizing the circuits in the tag. The tag then sends the information encoded in the tag's memory.

The RX and TX pins of RFID reader connected to Tx and Rx pins of 8051 Microcontroller respectively. Then the reader senses the data from the Tag and transmits the sensed data to microcontroller via serial port.



Fig 4.12: RFID reader



Fig4.13: RFID card

The EM-18 RFID Reader module operating at 125kHz is an inexpensive solution for your RFID based application. The Reader module comes with an on-chip antenna and can be powered up with a 5V power supply. Power-up the module and connect the transmit pin of the module to receive pin of your microcontroller. Show your card within the reading distance and the card number is thrown at the output. Optionally the module can be configured for also a weigand output.

Typical application

E-Payment

- E-Toll Road Pricing
- E-Ticketing for Events
- E-Ticketing for Public Transport
- Access Control
- PC Access
- Authentication
- Printer / Production Equipment

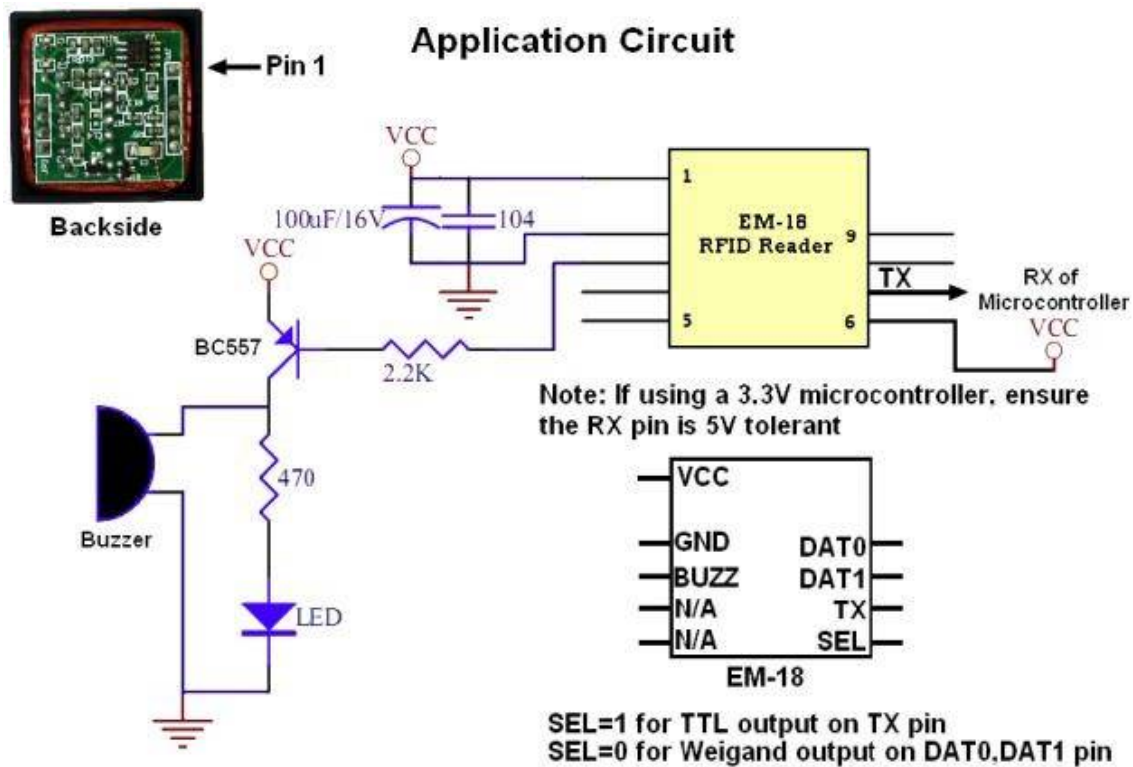


Fig 4.14: Application circuit

4.1.5 KEYPAD

At the lowest level, keyboards are organized in a matrix of rows and columns. The CPU accesses both rows and column through ports; therefore, with two 8-bit ports, an 8*8 matrix of keys can be connected to a microprocessor. When a key pressed, a row and column make a connect otherwise, there is no connection between row and column. In IBM PC keyboards, a single microcontroller (consisting of microprocessor, RAM and EPROM, and several ports all on a single chip) takes care of software and hardware interfacing of keyboard. In such systems it is the function of programs



Fig 4.15: Keypad

stored in the EPROM of microcontroller to scan the keys continuously, identify which one has been activated, and present it to the motherboard.

Figure shows a 4*4 matrix connected to two ports. The rows are connected to an output port and the columns are connected to an input port. If no key has been pressed, reading the input port will yield 1s for all columns since they are all connected to high (Vcc) If all the rows are grounded and a key is pressed, one of the columns will have 0 since the key pressed provides the path to ground. It is the function of the microcontroller to scan the keyboard continuously to detect and identify the key pressed.

Grounding rows and reading columns

To detect a pressed key, the microcontroller grounds all rows by providing 0 to the output latch, and then it reads the columns. If the data read from the columns is D3D0=1111, no key has been pressed and the process continues until a key press is detected. However, if one of the column bits has a zero, this means that a key press has occurred. For example, if D3-D0=1101, this means that a key in the D1 column has been pressed. After a key press is detected, the microcontroller will go through the process of identifying the key. Starting with the top row, the microcontroller grounds it by providing a low to row D0 only; then it reads the columns. If the data

read is all 1s, no key in that row is activated and the process is moved to the next row. It grounds the next row, reads the columns, and checks for any zero. This process continues until the row is identified. After identification of the row in which the key has been pressed, the next task is to find out which column the pressed key belongs to. This should be easy since the microcontroller knows at any time which row and column are being accessed.

In the program, it is assumed that P1 and P2 are initialized as output and input, respectively. Program13.1 goes through the following four major stages:

1. To make sure that the preceding key has been released, 0s are output to all rows at once, and the columns are read and checked repeatedly until all the columns are high. When all columns are found to be high, the program waits for a short amount of time before it goes to the next stage of waiting for a key to be pressed.
2. To see if any key is pressed, the columns are scanned over and over in an infinite loop until one of them has a 0 on it. Remember that the output latches connected to rows still have their initial zeros (provided in stage 1), making them grounded. After the key press detection, it waits 20ms for the bounce and then scans the columns again. This serves two functions: (a) it ensures that the first key press detection was not an erroneous one due to spike noise, and (b) the 20ms delay prevents the same key press from being interpreted as a multiple key press. If after the 20-ms delay the key is still pressed, it goes to the next stage to detect which row it belongs to; otherwise, it goes back into the loop to detect a real key press.
3. To detect which row the key press belongs to, it grounds one row at a time, reading the columns each time. If it finds that all columns are high, this means that the key press cannot belong to that row; therefore, it grounds the next row and continues until it finds the row the key press belongs to. Upon finding the row that the key press belongs to, it sets up the starting address for the look-up table holding the scan codes (or the ASCII value) for that row and goes to the next stage to identify the key.
4. To identify the key press, it rotates the column bits, one bit at a time, into the carry flag and checks to see if it is low. Upon finding the zero, it pulls out the ASCII

code for that key from the look-up table; Otherwise, it increments the pointer to point to the next element of the look-up table.

While the key press detection is standard for all keyboards, the process for determining which key is pressed varies. The look-up table method shown in program can be modified to work with any matrix up to 8*8.

4.1.6 LCD

LCD screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

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.

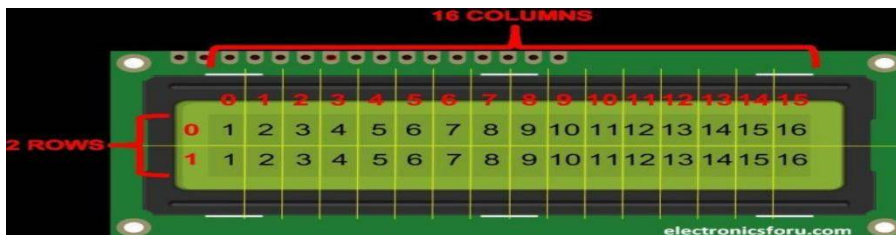


Fig 4.16: LCD

It come across LCD displays everywhere around them. Computers, calculators, television sets, mobile phones, digital watches use some kind of display to display the time. An LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16x2 LCD display is a very basic module commonly

used in projects. The 16×2 translates to a display 16 characters per line in 2 such lines. In this LCD each character is displayed in a 5×7 pixel matrix.

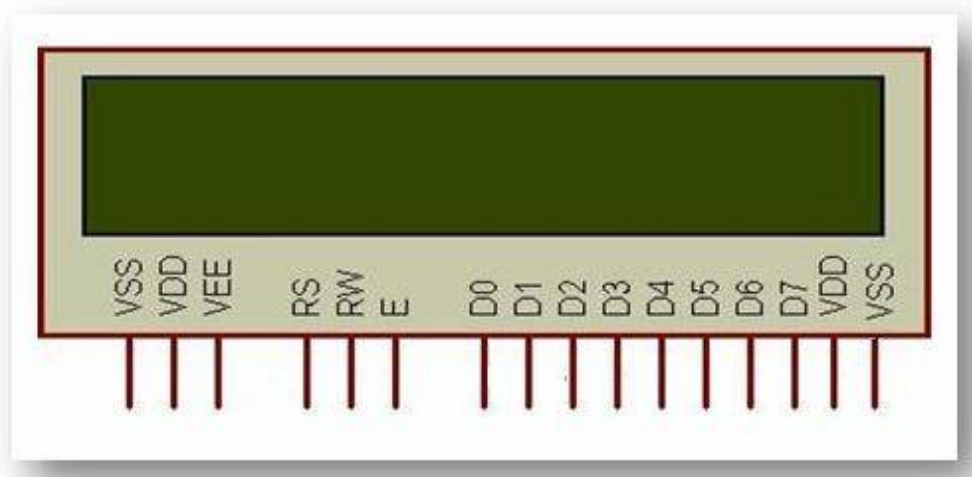


Table 4.15: Pin detail of LCD display

PIN NO.	FUNCTION	NAME
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	VCC
3	Contrast adjustment; the best way is to use a variable resistor such as a potentiometer. The output of the potentiometer is connected to this pin. Rotate the potentiometer knob forward and backwards to adjust the LCD contrast.	Vo / VEE
4	Selects command register when low, and data register when high	RS (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; Extra voltage push is required to execute the instruction and EN(enable) signal is used for this purpose. Usually, we make it en=0 and when we want to execute the instruction we make it high en=1 for some milliseconds. After this we again make it ground that is, en=0.	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2

10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15	Backlight VCC (5V)	Led+
16	Backlight Ground (0V)	Led-

- **RS (REGISTER SELECT)** : A 16X2 LCD has two registers, namely, command and data. The register select is used to switch from one register to other. RS=0 for command register, whereas RS=1 for data register.
- **COMMAND REGISTER** : 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. Processing for commands happens in the command register.
- **DATA REGISTER** : 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. When we send data to LCD it goes to the data register and is processed there. When RS=1, data register is selected.

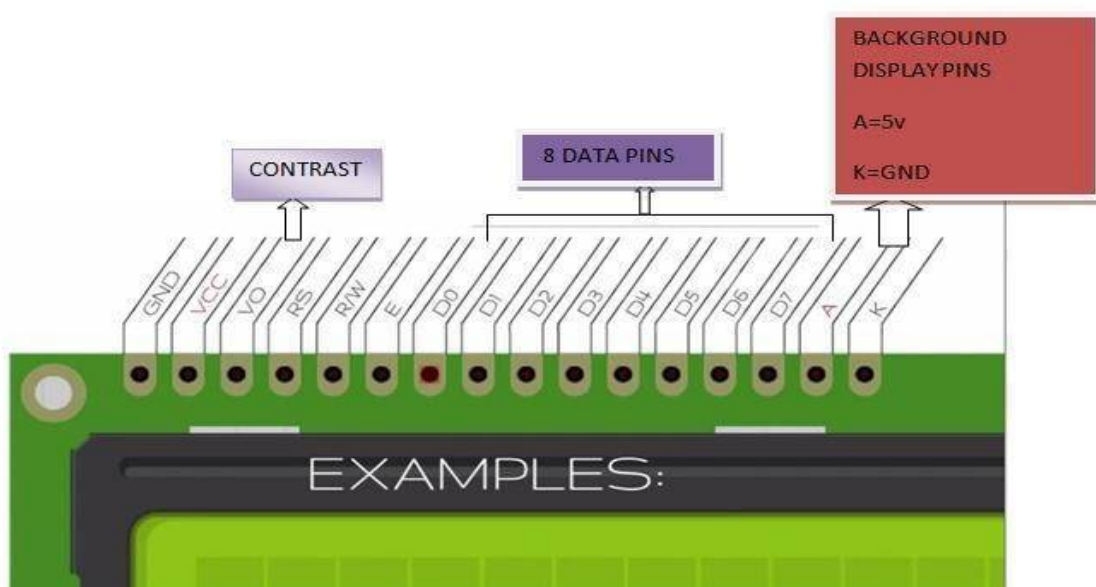


Fig 4.17: Data register

TABLE 4.16: IMPORTANT COMMAND CODES FOR LCD

SR.NO.	HEX CODE	COMMAND TO LCD INSTRUCTION REGISTER
1	01	Clear display screen
2	02	Return home
3	04	Decrement cursor (shift cursor to left)
4	06	Increment cursor (shift cursor to right)
5	05	Shift display right
6	07	Shift display left
7	08	Display off, cursor off
8	0A	Display off, cursor on
9	0C	Display on, cursor off
10	0E	Display on, cursor blinking
11	0F	Display on, cursor blinking
12	10	Shift cursor position to left
13	14	Shift cursor position to right
14	18	Shift the entire display to the left
15	1C	Shift the entire display to the right
16	80	Force cursor to beginning (1st line)
17	C0	Force cursor to beginning (2nd line)
18	38	2 lines and 5×7 matrix

➤ **DISPLAYING CUSTOM CHARACTERS ON 16X2 LCD:**

Generating custom characters on LCD is not very hard. It requires the knowledge about custom generated random access memory (CG-RAM) of LCD and the LCD chip controller. Most LCDs contain Hitachi HD4478 controller. CG-RAM is the main component in making custom characters. It stores the custom characters once declared in the code. CG-RAM size is 64 byte providing the option of creating eight characters at a time. Each character is eight byte in size.

CG-RAM address starts from 0x40 (Hexadecimal) or 64 in decimal. We can generate custom characters at these addresses. Once we generate our characters at these addresses, now we can print them on the LCD at any time by just sending

simple commands to the LCD. Character addresses and printing commands are below.

In the table above you can see starting addresses for each character with their printing commands. The first character is generated at address 0x40 to 0x47 and is printed on LCD by just sending simple command 0 to the LCD. The second character is generated at address 0x48 to 0x55 and is printed by sending 1 to LCD.

4.1.7 FINGERPRINT SENSOR

R305 Fingerprint Module consists of optical fingerprint sensor, high-speed DSP processor, high-performance fingerprint alignment algorithm, high-capacity FLASH chips and other hardware and software composition, stable performance, simple structure, with fingerprint entry, image processing, fingerprint matching, search and template storage and other functions.



Fig 4.18: Finger print sensor

➤ Fingerprint Module R305

R305 is a finger print sensor module with TTL UART interface. The user can store the finger print data in the module and can configure it in 1:1 or 1: N mode for identifying the person. The FP module can directly interface with 3v Microcontroller. A level converter (like MAX232) is required for interfacing with PC.

The R305 fingerprint module has two interface TTL UART and USB2.0, USB2.0 interface can be connected to the computer; RS232 interface is a TTL level, the default baud rate is 57600 , can be changed, refer to a communication protocol ; can And microcontroller, such as ARM, DSP and other serial devices with a

connection, 3.3V 5V microcontroller can be connected directly. Needs to connect the computer level conversion, level conversion note, embodiments such as a MAX232 circuit.

4.1.7.1 Operation Principle

Fingerprint processing includes two parts: fingerprint enrollment 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. For 1:1 matching, system will compare the live finger with specific template designated in the Module; for 1:N matching, or searching, system will search the whole finger library for the matching finger. In both circumstances, system will return the matching result, success or failure. USB Cable Connections are 5V/D+/D-/GND (Optional)

4.1.7.2 Features

- Supply voltage: DC 4.2 ~ 6.0V.
- Supply current: Working current: 50mA (typical) Peak current: 80mA.
- Fingerprint image input time: <0.3 seconds.
- Window area: 14x18 mm.
- Matching method: Comparison method (1: 1).
- Search method (1: N).
- Characteristic file: 256 bytes.
- Template file: 512 bytes.
- Storage capacity: 1000 pieces.
- Security Level: Five (from low to high: 1, 2, 3, 4, and 5).
- Fake rate (FAR): <0.001%.
- Refusal rate (FRR): <1.0%.
- Search time: <1.0 seconds (1: 1000 hours, mean value).
- Host interface: UART \ USB1.1.
- Communication baud rate (UART): (9600xN) bps Where N = 1 ~ 12 (default N = 6, ie 57600bps).

- Working environment: Temperature: -20 °C - +40 °C Relative humidity: 40% RH-85% RH (no condensation).

➤ **Package Includes:-**

- 1 - Finger Print Sensor (R305) -TTL UART
- 1 - 6 pin RMC Connector with wire

➤ **HARDWARE INTERFACE**

Exterior View

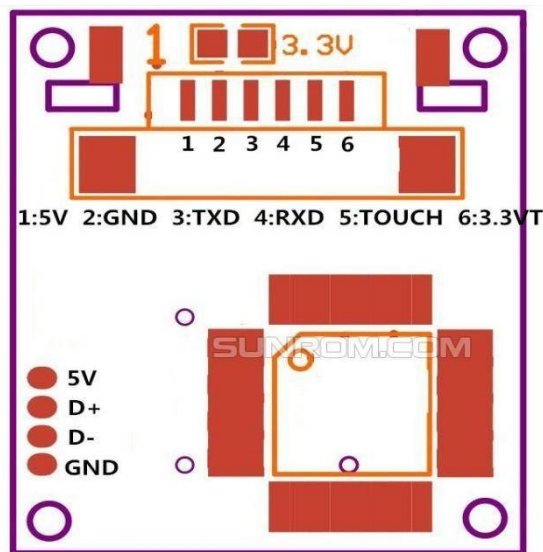


Fig 4.19 : Hardware interface

- **Serial Communication** : When the FP module communicates with user device, definition of J1 is as follows:

Table 4.17: Serial communication pin details

Pin number	Name	Type	Function Description
1	5v	In	Power input (DC 4.2V- 6V)
2	GND	-	Signal ground .connected to power ground
3	TXD	Out	Data output. TTL logical level
4	RXD	In	Data input. TTL logical level

5	Touch	OUT	Finger detection signal (maximum output current: 50mA)
6	3.3V	In	Finger detection power (DC3.3V - 5V、 about 5uA)

- **Hardware connection** : Via serial interface, the Module may communicate with MCU of 3.3V or 5V power: TXD (pin 3 of P1) connects with RXD (receiving pin of MCU), RXD (pin 4 of P1) connects with TXD (transferring pin of MCU). Should the upper computer (PC) be in RS-232 mode, please add level converting circuit, like MAX232, between the Module and PC.

Table 4.18 : USB Communication

Pin number	Name	Type	Function description
7	5V	In	Power Input
8	D+	Out	USB data output.
9	D-	In	USB data input.
10	GND	-	Signal ground.

➤ **Serial Communication**

The mode is semi duplex asynchronism serial communication. And the default baud rate is 57600bps. User may set the baud rate in 9600 ~ 115200bps 。
Transferring frame format is 10 bit: the low-level starting bit, 8-bit data with the LSB first, and an ending bit. There is no check bit.

➤ **Reset time**

At power on, it takes about 200ms for initialization. During this period, the Module can't accept commands for upper computer. To address demands of different customer, Module system provides abundant resources at user's use.

➤ **Buffer**

There are an image buffer and two 512-byte-character-file buffer within the RAM space of the module. Users can read & write any of the buffers by instructions. Note: Contents of the above buffers will be lost at Power-off.

➤ **Image buffer**

Image Buffer serves for image storage and the image format is 256*288 pixels, form is BMP. When transferring through UART, to quicken speed, only the upper 4 bits of the pixel is transferred (that is 16 grey degrees). And two adjacent pixels of the same row will form a byte before the transferring. When uploaded to PC, the 16-grey-degree image will be extended to 256-grey-degree format. That's 8-bit BMP format. When transferring through USB, the image is 8-bit pixel, that's 256 grey degrees.

➤ **Character file buffer**

Character file buffer, CharBuffer1, CharBuffer2, can be used to store both character file and template file.

➤ **Fingerprint Library**

System sets aside a certain space within Flash for fingerprint template storage, that's fingerprint library. Contents of the library remain at power off. Capacity of the library changes with the capacity of Flash, system will recognize the latter automatically. Fingerprint template's storage in Flash is in sequential order. Assume the fingerprint capacity N, then the serial number of template in library is 0, 1, 2, 3 ... N. User can only access library by template number.

Note: Commands shall only be sent from upper computer to the Module, and the Module acknowledges the commands.

Upon receipt of commands, Module will report the commands execution status and results to upper computer through acknowledge packet. Acknowledge packet has parameters and may also have following data packet. Upper computer can't ascertain Module's package receiving status or command execution results unless through acknowledge packet sent from Module. Acknowledge packet includes 1 byte confirmation code and maybe also the returned parameter.

Confirmation code's definition is:

00h: command execution complete;
01h: error when receiving data package;
02h: no finger on the sensor;
03h: fail to enroll the finger;
04h: fail to generate character file due to the over-disorderly fingerprint image;
05h: fail to generate character file due to the over-wet fingerprint image;
06h: fail to generate character file due to the over-disorderly fingerprint image;
07h: fail to generate character file due to lackness of character point or oversmallness of fingerprint image
08h: finger doesn't match;
09h: fail to find the matching finger;
0Ah: fail to combine the character files;
0Bh: addressing PageID is beyond the finger library;
0Ch: error when reading template from library or the template is invalid;
0Dh: error when uploading template;
0Eh: Module can't receive the following data packages.
0Fh: error when uploading image;
10h: fail to delete the template;
11h: fail to clear finger library;
13h: wrong password!
15h: fail to generate the image for the lackness of valid primary image;
18h: error when writing flash;
19h: No definition error;
1Ah: invalid register number;
1Bh: incorrect configuration of register;
1Ch: wrong notepad page number;

1Dh: fail to operate the communication port; others: system reserved;
 41h: No finger on sensor when add fingerprint on second time.
 42h: fail to enroll the finger for second fingerprint add.
 43h: fail to generate character file due to lackness of character point or oversmallness of fingerprint image for second fingerprint add
 44h: fail to generate character file due to the over-disorderly fingerprint image for second fingerprint add; 45h: Duplicate fingerprint.

Applications:

- Secure logins via keyboard modules
- User identification at kiosks
- Biometric door locks
- Credit card security
- Weapon activation
- Theft protection

4.1.8 BUZZER



Fig :Active Passive Buzzer

Active Passive Buzzer Pinout

TABLE 4.19 :Buzzer Pin Configuration

Pin Number	Pin Name	Description
------------	----------	-------------

1	Positive	Identified by (+) symbol or longer terminal lead. Can be powered by 6V DC
2	Negative	Identified by short terminal lead. Typically connected to the ground of the circuit

4.1.8.1 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

- **Equivalents for Passive Buzzer:** Piezo Electric buzzer, Speaker, Active Passive Buzzer with Module

How to use a Buzzer

A **buzzer** is a small yet efficient component to add sound features to our project/system. It is very small and compact 2-pin structure hence can be easily used on breadboard, Perf Board and even on PCBs which makes this a widely used component in most electronic applications.

There are two types are buzzers that are commonly available. The one shown here is a simple buzzer which when powered will make a Continuous Beeeeeeppp.... sound, the other type is called a readymade buzzer which will look bulkier than this and will produce a Beep. Beep. Beep. Sound due to the internal oscillating circuit present inside it. But, the one shown here is most widely used because it can be customised with help of other circuits to fit easily in our application.

This buzzer can be used by simply powering it using a DC power supply ranging from 4V to 9V. A simple 9V battery can also be used, but it is recommended to use a regulated +5V or +6V DC supply. The buzzer is normally associated with a switching circuit to turn ON or turn OFF the buzzer at required time and require interval.

Applications of Buzzer

- Alarming Circuits, where the user has to be alarmed about something
- Communication equipments
- Automobile electronics
- Portable equipments, due to its compact size

4.1.9 SERVO MOTOR

A **servo motor** is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a **servo mechanism**. If motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor. For this tutorial, we will be discussing only about the **DC servo motor working**. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy car, RC helicopters and planes, Robotics, etc.

Servo motors are rated in kg/cm (kilogram per centimeter) most hobby servo motors are rated at 3kg/cm or 6kg/cm or 12kg/cm. This kg/cm tells you how much weight your servo motor can lift at a particular distance. For example: A 6kg/cm Servo motor should be able to lift 6kg if the load is suspended 1cm away from the motors shaft, the greater the distance the lesser the weight carrying capacity. The position

of a servo motor is decided by electrical pulse and its circuitry is placed beside the motor.

Servomotors are not a specific class of motor although the term servomotor is often used to refer to a motor suitable for use in a closed-loop control system. The motor is paired with some type of encoder to provide position and speed feedback. In the simplest case, only the position is measured. The measured position of the output is compared to the command position, the external input to the controller.

If the output position differs from that required, an error signal is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position. As the positions approach, the error signal reduces to zero and the motor stops.

A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors.

A servomotor is closed-loop servomechanism that uses position feedback to control its motion and final position. The input to its control is some signal, either analog or digital, representing the position commanded for the output shaft.

4.1.9.1 FEATURES AND APPLICATIONS

□ Supply voltage: 5VDC □ Control
signal: Analog or Digital □ High-
precision positioning.

APPLICATIONS

- Conveyors
- Solar Tracking System
- Antenna Positioning
- Camera Auto Focus
- It is used to measure the speed of the output shaft.

- Servo Motor Working Mechanism
- It consists of three parts:
- Controlled device
- Output sensor
- Feedback system

It is a closed-loop system where it uses a positive feedback system to control motion and the final position of the shaft. Here the device is controlled by a feedback signal generated by comparing output signal and reference input signal.

Here reference input signal is compared to the reference output signal and the third signal is produced by the feedback system. And this third signal acts as an input signal to the control the device. This signal is present as long as the feedback signal is generated or there is a difference between the reference input signal and reference output signal. So the main task of servomechanism is to maintain the output of a system at the desired value at presence of noises.

4.1.9.2 SERVO MOTOR WORKING PRINCIPLE

A servo consists of a Motor (DC or AC), a potentiometer, gear assembly, and a controlling circuit. First of all, we use gear assembly to reduce RPM and to increase torque of the motor. Say at initial position of servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer. Now an electrical signal is given to another input terminal of the error detector amplifier. Now the difference between these two signals, one comes from the potentiometer and another comes from other sources, will be processed in a feedback mechanism and output will be provided in terms of error signal. This error signal acts as the input for motor and motor starts rotating. Now motor shaft is connected with the potentiometer and as the motor rotates so the potentiometer and it will generate a signal. So as the potentiometer's angular position changes, its output feedback signal changes. After sometime the position of potentiometer reaches at a position that the output of potentiometer is same as external signal provided. At this condition, there will be no output signal from the amplifier to the motor input as there is no difference between external applied signal and the signal generated at potentiometer, and in this situation motor stops rotating.

➤ INTERFACING SERVO MOTORS WITH MICROCONTROLLERS:

Interfacing hobby Servo motors like s90 servo motor with MCU is very easy. **Servos have three wires coming out of them.** Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU. An **MG995 Metal Gear Servo Motor** which is most commonly used for RC cars humanoid bots etc. The picture of MG995 is shown below:

All servo motors work directly with your +5V supply rails but we have to be careful on the amount of current the motor would consume if you are planning to use more than two servo motors a proper servo shield should be designed.

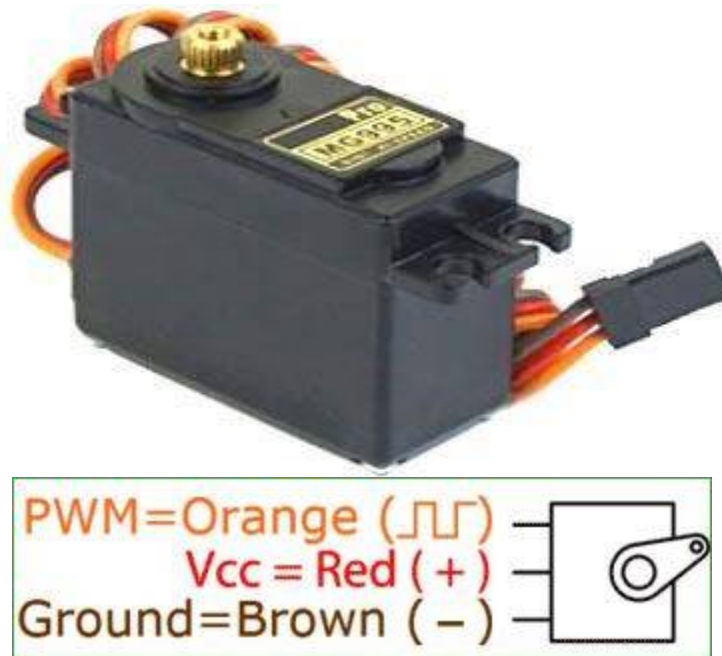


Fig 4.20 : Servo motor

The color coding of your servo motor might differ hence check for your respective datasheet.

➤ CONTROLLING SERVO MOTOR:

All motors have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU. Servo motor is controlled by PWM (Pulse with Modulation) which is provided by the control wires. There is a minimum pulse, a maximum pulse and a repetition rate. Servo motor can turn 90 degree from either direction from its neutral

position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns. For example, a 1.5ms pulse will make the motor turn to the 90° position, such as if pulse is shorter than 1.5ms shaft moves to 0° and if it is longer than 1.5ms than it will turn the servo to 180°.

Servo motor works on **PWM (Pulse width modulation)** principle, means its angle of rotation is controlled by the duration of applied pulse to its Control PIN. Basically servo motor is made up of **DC motor which is controlled by a variable resistor (potentiometer) and some gears**. High speed force of DC motor is converted into torque by Gears. We know that $WORK = FORCE \times DISTANCE$, in DC motor Force is less and distance (speed) is high and in Servo, force is High and distance is less. The potentiometer is connected to the output shaft of the Servo, to calculate the angle and stop the DC motor on the required angle.

Servo motor can be rotated from 0 to 180 degrees, but it can go up to 210 degrees, depending on the manufacturing. This degree of rotation can be controlled by applying the **Electrical Pulse** of proper width, to its Control pin. Servo checks the pulse in every 20 milliseconds. The pulse of 1 ms (1 millisecond) width can rotate the servo to 0 degrees, 1.5ms can rotate to 90 degrees (neutral position) and 2 ms pulse can rotate it to 180 degree.

All servo motors work directly with your +5V supply rails but we have to be careful about the amount of current the motor would consume if you are planning to use more than two servo motors a proper servo shield should be designed.

4.2 SOFTWARE REQUIREMENTS

4.2.1 EMBEDDED C

Embedded C is most popular programming language in software field for developing electronic gadgets. Each processor used in electronic system is associated with embedded software.

Embedded C programming plays a key role in performing specific function by the processor. In day-to-day life we used many electronic devices such as mobile

phone, washing machine, digital camera, etc. These all device working is based on microcontroller that are programmed by embedded C.

Fig 4.22 : block diagram representation of embedded system programming

The Embedded C code written in above block diagram is used for blinking the LED connected with Port0 of microcontroller.

In embedded system programming C code is preferred over other language. Due to the following reasons:

- Easy to understand
- High Reliability
- Portability
- Scalability

➤ **EMBEDDED SYSTEM PROGRAMMING:**

Basic Declaration

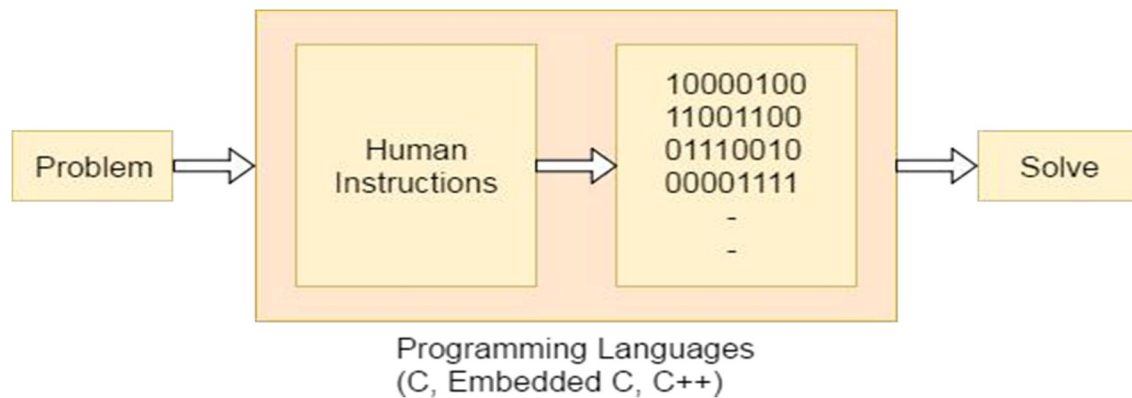


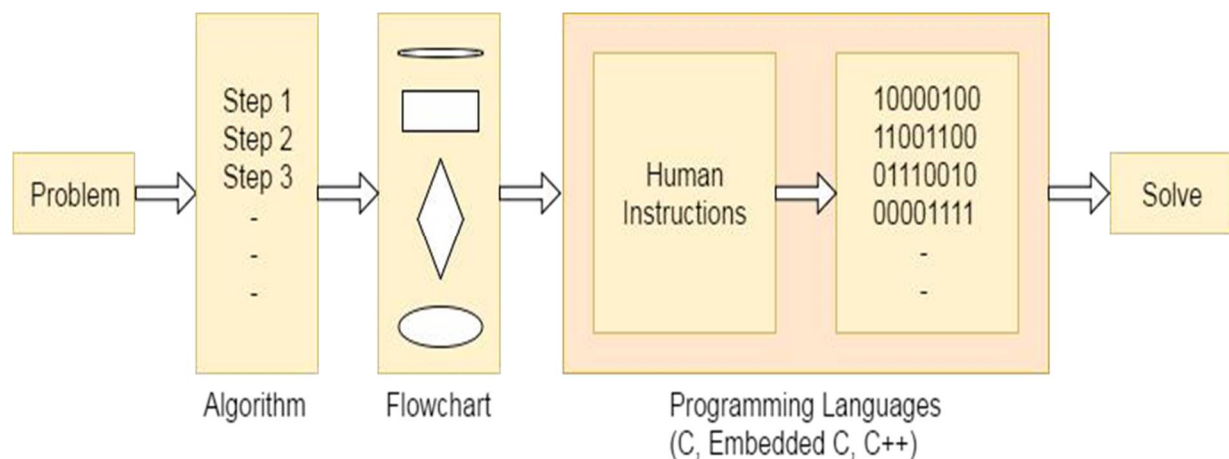
Fig 4.22: block diagram of Embedded C Programming development

Function is a collection of statements that is used for performing a specific task and a collection of one or more functions is called a programming language. Every language is consisting of basic elements and grammatical rules. The C language programming is designed for function with variables, character set, data types, keywords, expression and so on are used for writing a C program.

The extension in C language is known as embedded C programming language. As compared to above the embedded programming in C is also have some additional features like data types, keywords and header file etc is represented by `#include<microcontroller name.h>`.

➤ **BASIC EMBEDDED C PROGRAMMING STEPS:**

Let's see the block diagram representation of Embedded C Programming Steps:



The microcontroller programming is different for each type of operating system. Even though there are many operating system are exist such as Windows, Linux, RTOS, etc but RTOS has several advantage for embedded system development.

- **EMBEDDED SYSTEMS** : Embedded System is a system composed of hardware, application software and real time operating system. It can be small independent system or large combinational system.

Embedded System tutorial includes all topics of Embedded System such as characteristics, designing, processors, microcontrollers, tools, addressing modes, assembly language, interrupts, embedded c programming, led blinking, serial communication, lcd programming, keyboard programming, project implementation etc.

- **SYSTEM** : System is a way of working, organizing or performing one or many tasks according to a fixed set of rules, program or plan.

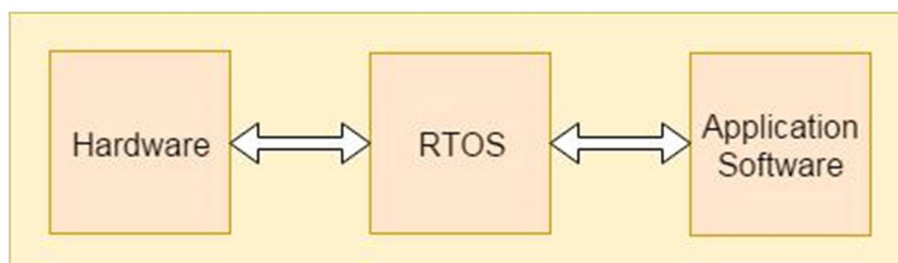
It is an arrangement in which all the unit combined to perform a work together by following certain set of rules in real time computation. It can also be defined as a way of working, organizing or doing one or many tasks according to a fixed plan.

An Embedded System is a system that has software embedded into computer-hardware, which makes a system dedicated for a variety of application or specific part of an application or product or part of a larger system.

An embedded system can be a small independent system or a large combinational system. It is a microcontroller-based control system used to perform a specific task of operation.

An embedded system is a combination of three major components:

- **Hardware:** Hardware is physically used component that is physically connected with an embedded system. It comprises of microcontroller based integrated circuit, power supply, LCD display etc.
- **Application software:** Application software allows the user to perform varieties of application to be run on an embedded system by changing the code installed in an embedded system.
- **Real Time Operating system (RTOS):** RTOS supervises the way an embedded system work. It act as an interface between hardware and application software which supervises the application software and provide mechanism to let the processor run on the basis of scheduling for controlling the effect of latencies.



➤ **CHARACTERISTICS OF EMBEDDED SYSTEM:**

- An embedded system is software embedded into computer hardware that makes a system dedicated to be used for variety of application.

- Embedded system generally used for do specific task that provide realtime output on the basis of various characteristics of an embedded system.
- Embedded system may contain a smaller part within a larger device that used for serving the more specific application to perform variety of task using hardware-software intermixing configuration.
- It provides high reliability and real-time computation ability.

Advantages:

- Same hardware can be used in variety of application.
- Lesser power requirement
- Lower operational cost of system
- Provide high performance and efficiency

Disadvantages:

- Developing a system required more time. Due to functional complexity.
- Skilled engineers required because one mistake may result in destroying of complete project.

Designing of an embedded system

➤ **BASIC STRUCTURE OF AN EMBEDDED SYSTEM:**

Let's see the block diagram shows the basic structure of an embedded system.

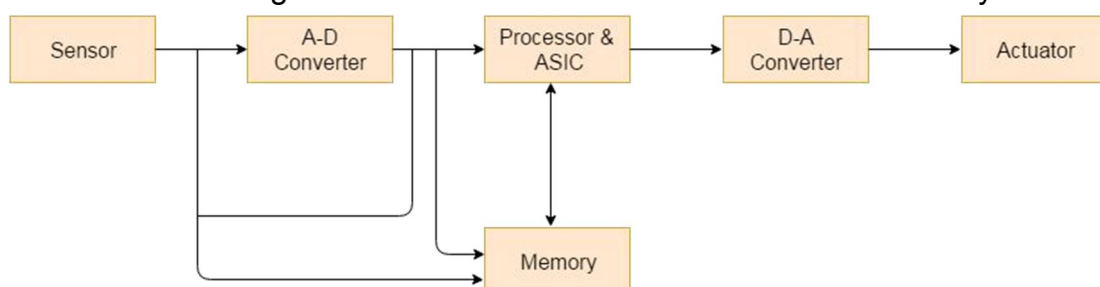


Fig : basic structure of an embedded system.

- **Sensor:** Sensor used for sensing the change in environment condition and it generate the electric signal on the basis of change in environment

condition. Therefore it is also called as transducers for providing electric input signal on the basis of change in environment condition.

- **A-D Converter:** An analog-to-digital converter is a device that converts analog electric input signal into its equivalent digital signal for further processing in an embedded system.
- **Processor & ASICs:** Processor used for processing the signal and data to execute desired set of instructions with high-speed of operation. Application specific integrated circuit (ASIC) is an integrated circuit designed to perform task specific operation inside an embedded system.
- **D-A Converter:** A digital-to-analog converter is a device that converts digital electric input signal into its equivalent analog signal for further processing in an embedded system.
- **Actuators:** Actuators is a comparator used for comparing the analog input signal level to desired output signal level for providing the error free output from the system.

➤ **DESIGN STEPS REQUIRED FOR THE DEVELOPMENT OF EMBEDDED SYSTEM:**

Designing steps required for embedded system are different from the design process of another electronic system.

➤ **EMBEDDED SYSTEM TOOLS AND PERIPHERALS:**

- **Compiler:** Compiler is used for converting the source code from a high-level programming language to a low-level programming language. It converts the code written in high level programming language into assembly or machine code. The main reason for conversion is to develop an executable program.

Let's see the operations performed by compiler are:

- Code generation
- Code optimization
- Parsing
- Syntax direct translation

- Preprocessing

➤ **Cross-Compiler:**

If a program compiled is run on a computer having different operating system and hardware configuration than the computer system on which a compiler compiled the program, that compiler is known as cross-compiler.

➤ **Decompiler:**

A tool used for translating a program from a low-level language to high-level language is called a decompiler. It is used for conversion of assembly or machine code to high-level programming language.

➤ **Assembler:**

Assembler is embedded system tool used for translating a computer instruction written in assembly language into a pattern of bits which is used by the computer processor for performing its basic operations. Assembler creates an object code by translating assembly language instruction into set of mnemonics for representing each low-level machine operation.

Let's see the different debugging tools used in embedded system are:

- **Simulators:**

Simulator is a tool used for simulation of an embedded system. Code tested for microcontroller unit by simulating code on the host computer. Simulator is used for model the behavior of the complete microcontroller in software.

Functions of simulators:

Let's see the functions performed by simulator are:

- It defines the processing or processor device family with various version of target system.

- It monitors the detailed information of a source code and symbolic arguments as the execution goes for each single step of operation.
- It simulates the ports of target system for each single step of execution.
- It provides the working status of RAM.
- It monitors the response of system and determines the throughput.
- It provides the complete meaning of the present command.
- It monitors the detailed information of the simulator commands entered from the keyboard or selected from the menu.
- It facilitates synchronization of internal peripherals and delays.

➤ **MICROCONTROLLER STARTER KIT:**

For developing an embedded system based project a complete microcontroller starter kit is required. The major advantage of this kit over simulator is that they work in real-time operating condition. Therefore it allows the easy input/output functional verification. Consider a microcontroller starter kit consists of:-

- Hardware Printed Circuit Board (PCB)
- In-System Programmer (ISP)
- Some embedded system tools like compiler, assembler, linker, etc
- Sometimes, there is a requirement of an Integrated Development Environment (IDE)

The above component available in microcontroller starter kit is completely enough and the cheapest option available for developing simple microcontroller projects.

- **Emulators:** An emulator is a software program or a hardware kit which emulates the functions of one computer system into another computer system. Emulators have an ability to support closer connection to an authenticity of the digital object.
- It can also be defined as the ability of a computer program in electronic device to emulate another program or device. It focusing on recreating the original computer environment and helps a user to work on any type of application or operating system.

➤ PERIPHERAL DEVICES IN EMBEDDED SYSTEMS:

Communication of an embedded system with an outside environment is done by using different peripheral devices as a combination with microcontroller.

Let's see the different peripheral devices in embedded system are:-

- Universal Serial Bus (USB)
- Networks like Ethernet, Local Area Network(LAN) etc
- Multi Media Cards (SD Cards, Flash memory, etc)
- Serial Communication Interface (SCI) like RS-232, RS-485, RS-422, etc
- Synchronous Serial Communication Interface like SPI, SSC and ESSI
- Digital to Analog/ Analog to Digital (DAC/ADC)
- General Purpose Input/Output (GPIO)
- Debugging like In System Programming (ISP), In Circuit Serial Programming (ICSP), BDM Port, etc

➤ CRITERIA FOR CHOOSING MICROCONTROLLER:

Choosing a microcontroller is essential process in designing of embedded system. While selecting a microcontroller, make sure that it meets the system need and it must be cost effective. We need to decide whether an 8-bit, 16-bit or 32-bit microcontroller is best suitable for the computing needs of a task.

In addition to above, the following points need to be kept in mind while selecting a microcontroller:-

- **Speed:** The operational speed of the microcontroller or the highest speed microcontroller can support.
- **Packaging:** Packaging is important for improving the assembling, space and prototyping of an end-product.
- **RAM and ROM:** On the basis of operation of embedded system and memory need for storage data and programs the type of microcontroller required for designing system is decided.

- **Count of I/O pins:** The number of input and output devices connected with the system plays an essential role in choosing the type of microcontroller. ➤ **Cost per unit:** It is important in terms of final cost of the product in which the microcontroller is to be used.
- **Power consumption:** Power consumption plays an important role for maintaining the efficiency of an embedded system

Additional commands are found within the five menus: **File**, **Edit**, **Sketch**, **Tools**, and help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

CHAPTER 5

RESULT AND DISCUSSION

5.1 OUTPUT

CONNECTIONS:

The below figure shows the Arduino Mega, RFID tag, RFID reader, GSM module, 16x2 LCD display, servo motor, keypad, buzzer and finger print sensor which was shown as a complete connection of the hardware components.

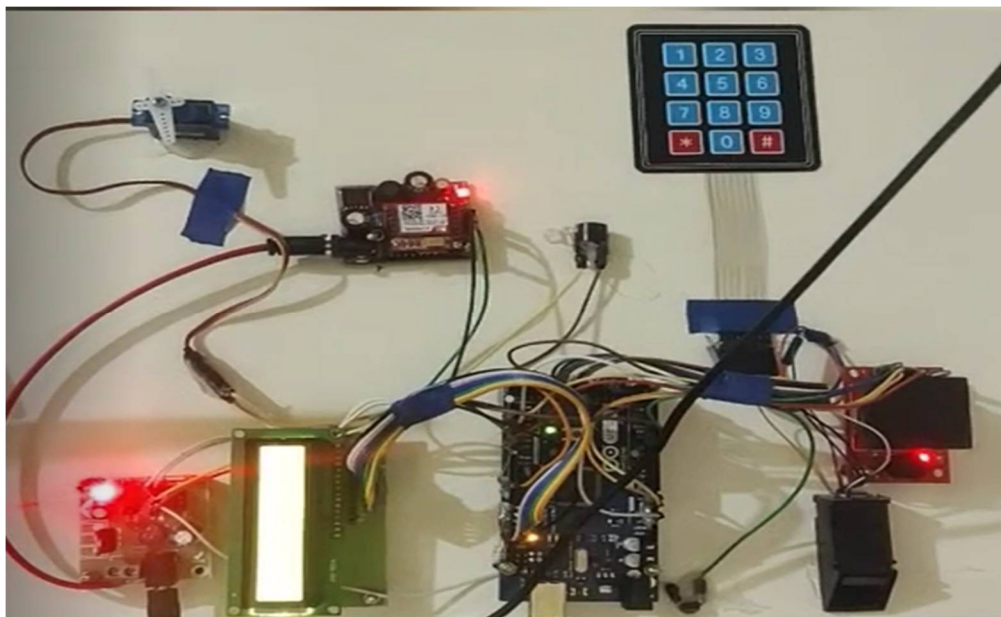


Fig 5.1: Overall circuit Connections

STEP:1 REGISTER



fifig 5.2: Using RFID Tag

The RFID Card instead of our ATM card to start our Registration process.



Fig 5.3: Using Fingerprint sensor

Here, the Biometric method using fingerprint sensor to access our RDID Card.



Fig 5.4: Registration of fingerprint



Fig 5.5 : finger print sensor

Once, our fingerprint is registered then we can easily access all our multiple accounts using our fingerprint without the hectic of having multiple card and multiple ATM pins.



Fig 5.6: mobile no Registration process

The registred mobile number to our respected fingerprint and RDIF card using “REG”.



Fig 5.7: Keypad

Keypad is used to operate our hardware components in a feasible way.



Fig 5.8: Mobile number Registration process

By using keypad here we are going to press "6" in order to access "PH" process to register our mobile number.

STEP:2 ENTER THE MOBILE NUMBER

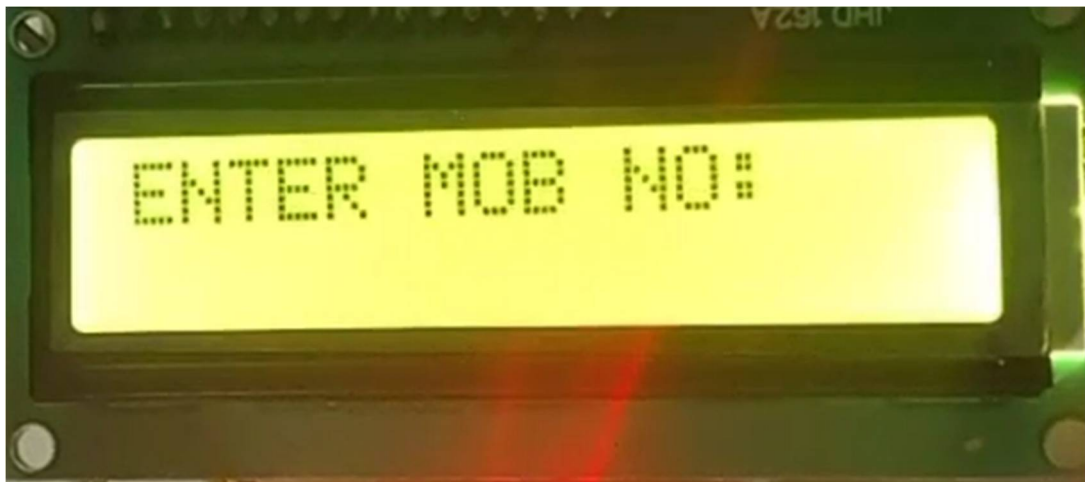


Fig 5.9: Mobile Number Access



Fig 5.10: Registering our Mobile Number

By registering our mobile number we can access our mobile number to the respected fingerprint to get otp(One Time Password).

Waiting for OTP



Fig 5.11: otp is generated

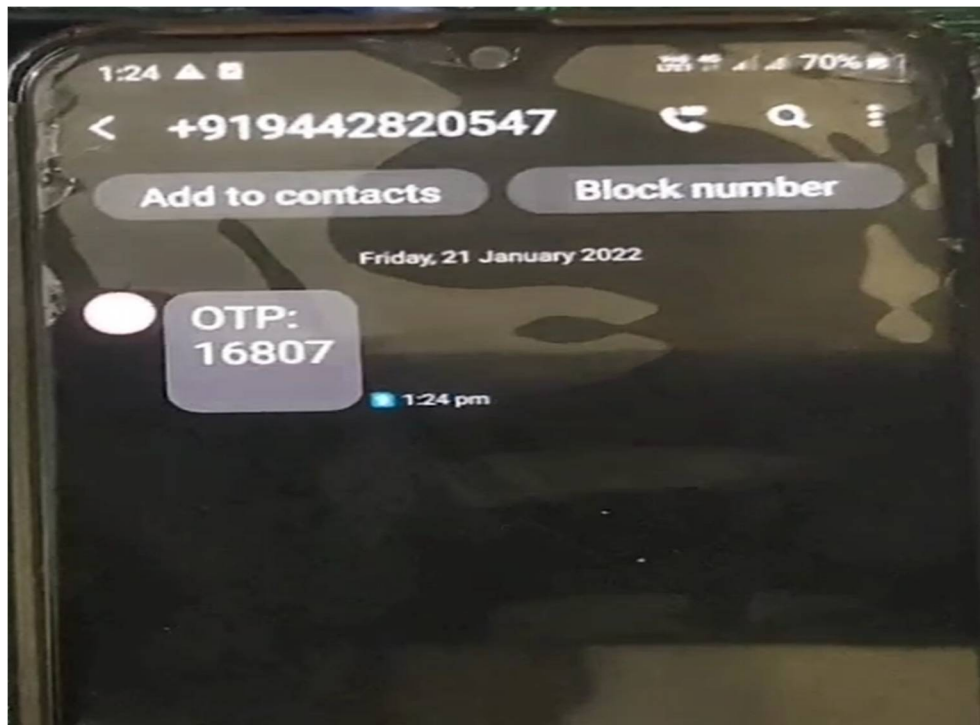


Fig 5.12: received otp

Thus received our otp to access our bank account to the resigtered Mobile Number.

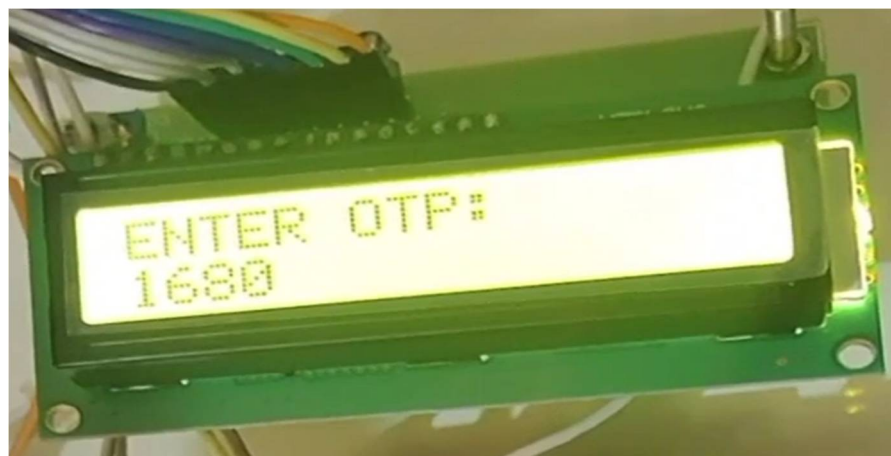


Fig 5.13: Enter the otp

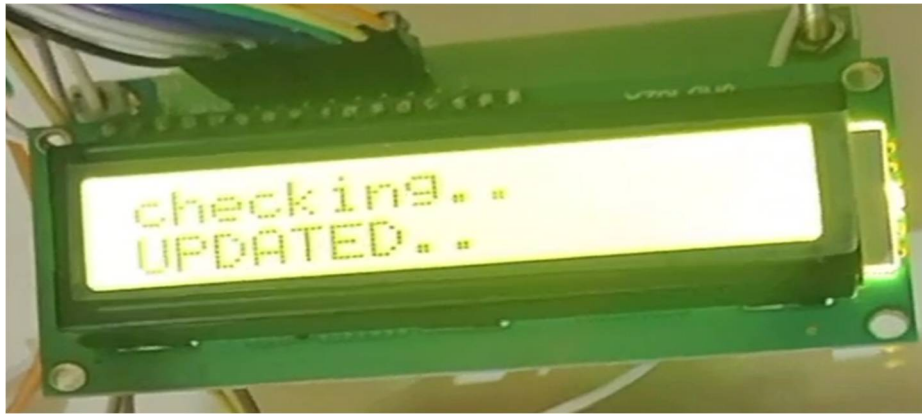


Fig 5.14: verification of otp

By entering the one time password we can access all our accounts that we want to use in our future now we can do all our transaction process using our smart ATM.

STEP:3 DEPOSIT

- **PROCESS OF DEPOSIT**



Fig: 5.15: Deposit

Using “(1) DEP” we can start to deposit in our account.

- **ACCESS THE BIOMETRIC**

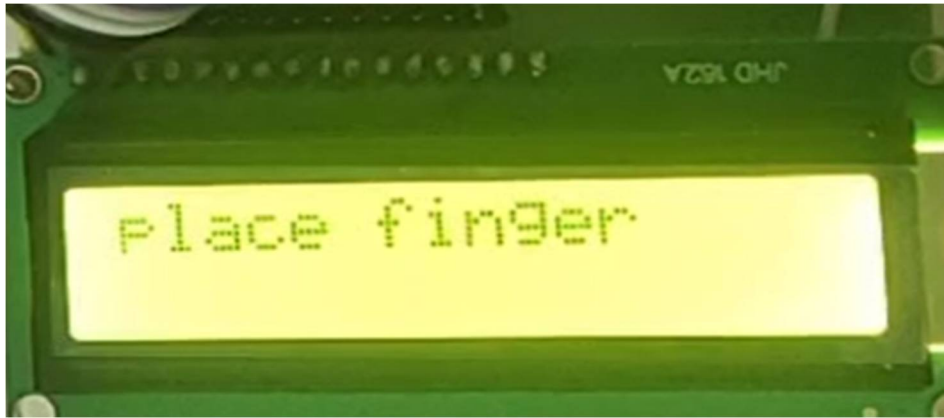


Fig 5.16: biometric process



Fig 5.17: Usage Biometric Security

- **SELECT THE BANK FOR DEPOSIT**



Fig: 5.18: Deposit

By pressing “(1) DEP” using our keypad, we can start our process of Depositing to our bank account.



Fig 5.19: selecting SBI bank(1)

By pressing “(1) SBI” in our keypad, we can deposit money in our SBI account.



Fig 5.19: Deposit Rs:5000

STEP:4 WITHDRAWAL

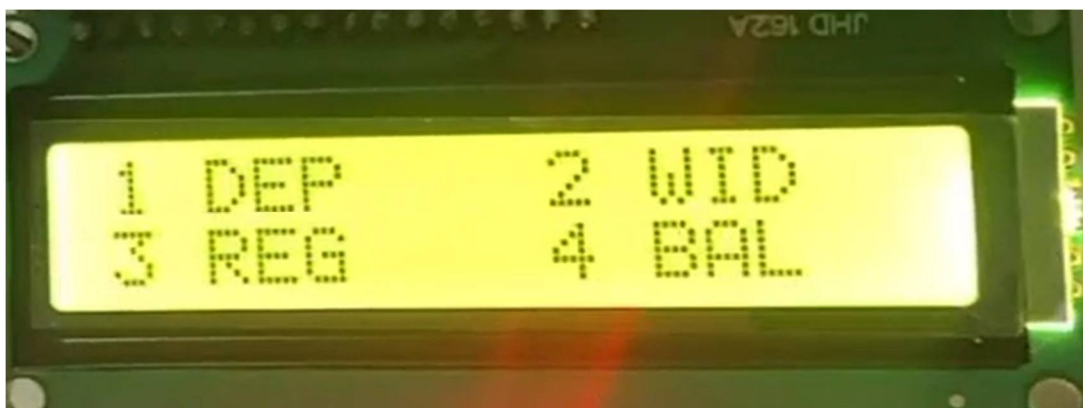


Fig 5.19: WID

Select “(2) WID” to start the process of width drawl in Smart ATM.



Fig 5.20: Select the Bank

Select “(1) SBI” to width drawl money from the selected bank account.



Fig 5.21: Enter the Amount

Enter the Amount which you want to width drawl from the bank account.



Fig 5.22: Withdrawal amount and Balance

Now the Withdrawal amount and Balance is visible in LCD Screen and also all these activities are sent as message to the registered mobile number.

STEP :5 CHECK THE BALANCE



Fig 5.23: Checking BAL

- Select “(4) **BAL**” from the keypad to check the balance from the registered and deposited bank accounts.
- Select “(1) **SBI**” using keypad to check the balance from the bank using the Smart ATM.



Fig 5.24: view Balance

Now our bank Balance is visible in LED Screen for our convenience.

STEP 6: UNAUTHORIZED

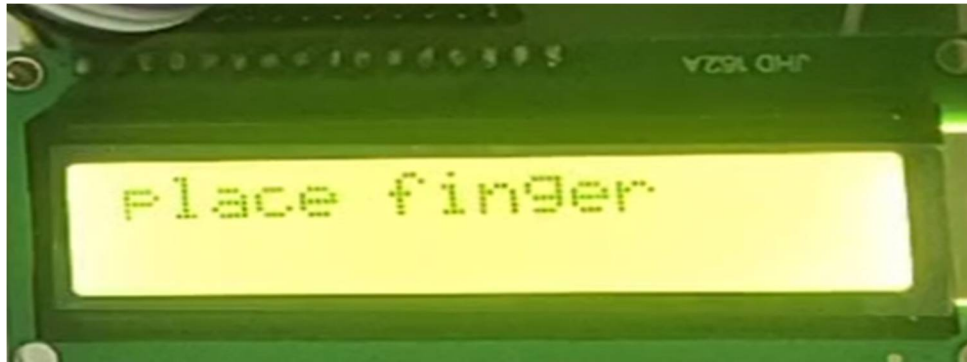


Fig 5.25: Place Finger



Fig 5.26: Fraud Access

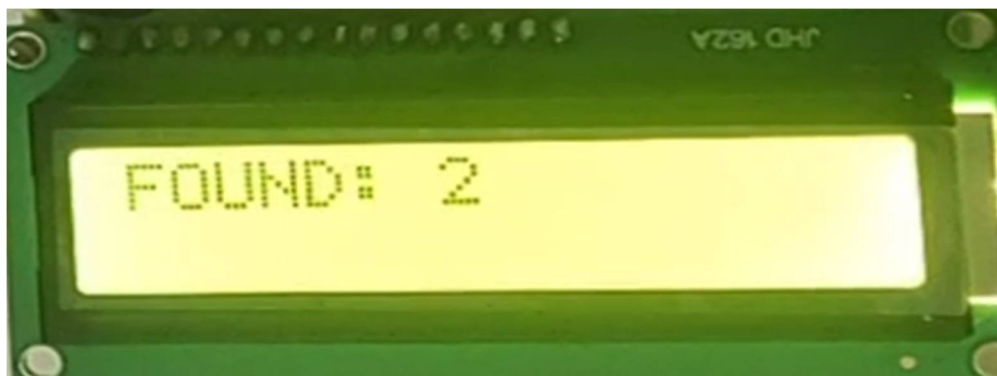


Fig 5.27: Mismatched finger print

In this case using biometric mismatched fingerprint will be easily dedected by Smart ATM.

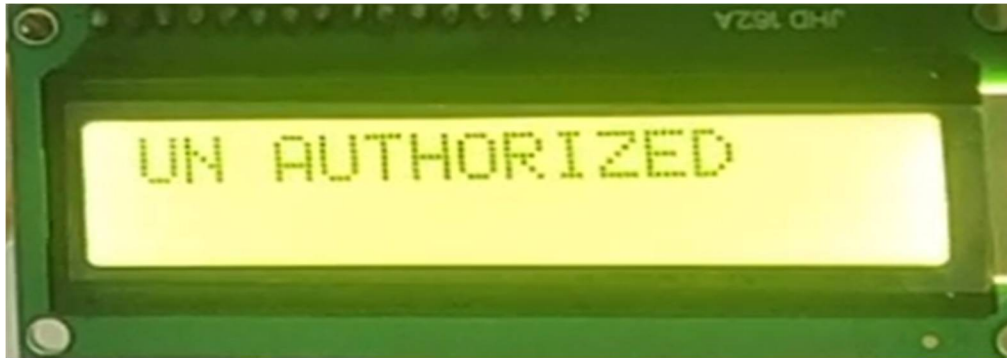


Fig 5.28: Un authorized

The LED Screen will display “**UN AUTHORIZED**” and the buzzer sound will be audible to everyone to catch the un authorized user.

STEP 7: MESSAGES TO THE REGIATERED MOBILE NUMBER



Fig 5.29: messages to the registered mobile number

From the above figure we can notice that all our Transactions, Registrations, Width drawl, Checking balances, Addition of bank and Removal of bank accounts will be sent as a message to the registered mobile number from the respected bank for the safety of the user.

CHAPTER: 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION:

There is gradual development in the accessibility of ATM card and necessity of improving the user friendly and Secure domain in the ATM cards and It evolving method of using it for multiple bank accounts So, the initiated the Fingerprint Security for the Project in order to differentiate and stand alone from the previous projects The fingerprint technology is the most widely accepted and mature biometric method and is the easiest to deploy and for a higher level of security at the fingertips. It is simple to install and also it takes little time and effort to acquire one's fingerprint with a fingerprint identification device . Thus, finger print recognition is considered among the least intrusive of all biometric verification techniques. The finger print features are different for each human being so the user can be identified uniquely. Instead of using multiple ATM card fingerprint based ATM is safer and secure. There is no worry of losing ATM card and no need to carry ATM card in your wallet. We just have to use our fingerprint in order to do any banking transaction. Several cards with different PINS to them is a difficult process to remember it and sometimes the user can mistyped the pin with can lead to blockage of ATM card. So this system is proposed to eliminate such issues and this system can make the user convenient with one card. And instead of PIN the user use their biometric which make it more secure. The transactions are same as the logic of a normal ATM. There will be an alert system so the user can identify their misuse of cards. This reduces the card blockage and other issues like mismatch of PINs.

6.2 ADVANTAGES

- Single ATM card provides more convince of using multiple bank transactions.
- Higher security based on the biometric module.

6.3 APPLICATIONS

- Single card can be used for multiple bank transaction.
- Transaction

6.4 FUTURE ENHANCEMENT

- Face identification and biometric can be used in the future for bank transactions.
- Several other security measures can be implemented in the future.

REFERENCES

- [1] Ahmad, A.W., Jan, N., Iqbal, S. and Lee, C. (2011) Implementation of ZigBee—GSM Based Home Security Monitoring and Remote Control System. IEEE 54th International Midwest Symposium on Circuits and Systems, Seoul, 7-10 August 2011, 1-4.
- [2] Archana et al., "Enhance the Security in the ATM System with Multimodal Biometrics and Two-Tier Security", International Journal of Advanced Research in Computer Science and Software Engineering 3(10), pp. 261-266, October – 2013.
- [3] Biswas S., Bardhan Roy A., Ghosh K. And Dey N., "A Biometric Authentication Based Secured ATM Banking System", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 4, April 2012
- [4] El-Medany, W.M. and El-Sabry, M.R. (2008) GSM-Based Remote Sensing and Control System using FPGA. Proceedings of International Conference on Computer and Communication Engineering, Kuala Lumpur, 13-15 May 2008, 1093-1097.
- [5] Golzar, M.G. and Tajozzakerin, H.R. (2010) A New Intelligent Remote Control System for Home Automation and Reduce Energy Consumption. 4th Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation, Kota Kinabalu, 26-28.
- [6] "Gsmd – Openmoko". Wiki.openmoko.org. 8 February 2010. Retrieved 22 April 2010.
- [7] "GSM UMTS 3GPP Numbering Cross Reference". ETSI. Retrieved 30 December 2009.
- [8] G. Vidhya Krishnan, R.Nagarajan, T. Durka, M.Kalaiselvi, M.Pushpa and S. Shanmugapriya, "Vehicle Communication System Using Li-Fi Technology," International Journal of Engineering And Computer Journal of Engineering Sciences Vol 8, Issue 2, 2017
- [9] J.Chandra mohan, R.Nagarajan, K. Satheesh kumar, N.Ajith kumar, P.A.Gopinath and S.Ranjithkumar, "Intelligent Smart Home Automation and Security System Using Arduino and Wi-fi,"

- [10] Karki, James (September 2000). "Signal Conditioning Piezoelectric Sensors" (PDF). Texas Instruments. Retrieved 2007-12-02.
- [11] Kim, Bo-Ra, "Domestic ATM status and meanings", Payment and Settlement, and IT, Vol. 44, pp. 76, 2011.
- [12] Ms. C. Hemalatha, Mr. R. Nagarajan, P. Suresh, G. Ganesh Shankar and A. Vijay, "Brushless DC Motor Controlled by using Internet of Things," IJSTE - International Journal of Science Technology & Engineering, Volume -3.Issue-09, pp. 373-377, March- 2017.
- [13] Rana, G.M.S.M., Khan, A.A.M., Hoque, M.N. and Mitul, A.F. (2016) Design and Implementation of a GSM Based Remote Home Security and Appliance Control System. Proceedings of the 2nd International Conference on Advances in Electrical Engineering, Dhaka, 19-21 December 2016, 291-295.
- [14] S.Suresh, R.Nagarajan, L.Sakthivel, V.Logesh, C.Mohandass and G.Tamilselvan, "Transmission Line Fault Monitoring and Identification System by Using Internet of Things," International Journal of Advanced Engineering Research and Science (IJAERS), Vol4.Issue-4, pp. 9-14, Apr- 2017.
- [15] Sakr, Sharif. "ARM co-founder John Biggs". Engadget. Retrieved December 23, 2011. "[...] the ARM7- TDML was licensed by Texas Instruments and designed into the Nokia 6110, which was the first ARM powered GSM phone."
- [16] S.T. Bho sale and Dr. B.S.Sawant "Security in E-Banking via Card Less Biometric ATMs", International Journal of Advanced Technology & Engineering Research, Volume 2, Issue 4, July 2012
- [17] Sunil Lohiya "Biometric identification and verification techniques -A future of ATM Banking System", Indian Streams Research Journal, Volume 2, Issue. 7, Aug 2012
- [18] Yuksekkaya, B., Kayalar, A.A., Tosun, M.B., Ozcan, M.K. and Alkar, A.Z. (2006) A GSM, Internet and Speech Controlled Wireless Interactive Home Automation System. IEEE Transactions on Consumer Electronics, 52, 837.

