DEVELOPMENT OF ELECTRONIC VOTING MACHINE WITH INCLUSION OF NEAR FIELD COMMUNICATION AND BIOMETRIC FINGERPRINT IDENTIFIER

A PROJECT REPORT

Submitted by

ARAVINDAN.S	**17106013**
ASIR SAM.R	**17106018**
DHINESH.M	**17106027**
DINAGAR.P	**17106028**

in partial fulfilment for award of degree of

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATION ENGINEERING
HINDUSTHAN COLLEGE OF ENGINEERING AND TECHNOLOGY
Approved by AICTE, New Delhi and Accredited with 'A' Grade by NAAC
(An Autonomous Institution, Affiliated to Anna University, Chennai)
Othakalmandapam Post, Coimbatore-641032

APRIL-2021

Hindusthan College of Engineering and Technology

Approved By AICTE, New Delhi And Accredited With 'A' Grade By NACC
(An Autonomous Institution, Affiliated to Anna University, Chennai)

Othakalmandapam Post, Coimbatore-641032

BONAFIDE CERTIFICATE

Certified that this project titled **"DEVELOPMENT OF ELECTRONIC VOTING MACHINE WITH INCLUSION OF NEAR FIELD COMMUNICATION AND BIOMETRIC FINGERPRINT IDENTIFIER"** is the Bonafide work of **ARAVINDAN.S, ASIR SAM.R, DHINESH.M and DINAGAR.P** who carried out the project under my supervision.

SIGNATURE	**SIGNATURE**
Dr. P. RAJESWARI M.E., Ph.D.,	**Dr.J.SATHEESH KUMAR M.E, Ph.D.**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor and Head,	**Assistant Professor,**
Department of Electronics and Communication Engineering,	Department of Electronics and Communication Engineering,
Hindusthan College of Engineering and Technology,	Hindusthan College of Engineering and Technology,
Coimbatore - 641032.	Coimbatore - 641032.

Submitted for Autonomous Project Viva-Voce examination Held on_________.

INTERNAL EXAMINAR **EXTERNAL EXAMINAR**

ACKNOWLEDGEMENT

We would like to thank the management and trustee for providing us with all necessary infrastructures to successfully carry out this project.

We extend our thanks to our honorable Chairman, **Mr.T.S.R.KHANNAIYANN** Secretary, **Mrs.Smt.SARASUWATHI KHANNAIYANN** of Hindusthan Educational and Charitable Trust for providing the facilities within the college.

We extend our thanks to our Chief Executive Officer **Dr.K.KARUNAKARAN**,Ph.D., for his constant support and motivation.

We wish to express our sincere thanks and gratitude to our Head of the Department **Dr.P.RAJESWARI** M.E, Ph.D., for her encouragement, guidance and support to complete the project.

We extend our sincere and heartiest thanks to our Project coordinator **Mr.D.BASKAR** M.E and **Mrs.T.NIVETHITHA** M.E, for their valuable suggestions and technical support.

We owe a whole hearted sense of reverence and gratitude to our project guide **Dr.J.SATHEESH KUMAR** M.E,Ph.D., for her valuable guidance and also for helping us to complete the assigned task.

We are indented to all teaching and non-teaching staff members of the Department of Electronics and Communication Engineering for their kind cooperation and encouragement.

ABSTRACT

ABSTRACT

This project describes design of Biometric Voting Machine Using Fingerprint Scanner and Arduino for voting in institutes and organizations. Indian constitution empowers its citizen to exercise right to vote. Election decides the future of country, so that the system used for voting should be trustworthy. The conventional system for voting is ballot paper and Electronic voting machine too, has many flaws and trust issues. To eradicate malpractice and defrauding of the above methods of voting, we have designed an advanced system by using arduino and Fingerprint module. In this system, a person has to register a fingerprint ID with the system which will be centrally stored in arduino. In organizations, educational institutes, a co-operative bank, maximum number of votes elect head of organization that holds the office of public interest. For confirmation of voter, the name of the candidate will be displayed on LCD for whom the voter has cast a vote. It has simple hardware design and it is easily accessible. In case user wants to remove any of stored ID then the user need to press DEL key, after pressing DEL key, LED will ask to select ID that is to be deleted. After pressing OK key, the selected ID will be deleted and LCD will display that which ID has been deleted successfully. This system is flexible to use.

TABLE OF CONTENTS

CONTENTS

LIST OF FIGURES

LIST OF FIGURES

LIST OF ABBREVIATIONS

LIST OF ABBREVIATIONS

1	WSN	WIRELESS SENSOR NETWORK
2	UART	UNIVERSAL ASYNCHRONOUS RECEIVER/TRANSMITTER
3	MCU	MICRO CONTROLLER UNIT
4	TTL	TRANSISTOR-TRANSISTOR LOGIC
5	ADC	ANALOG TO DIGITAL CONVERTER
6	IDE	INTEGRATED DEVELOPMENT ENVIRONMENT
7	COMM PORT	COMMUNICATION PORT
8	ROM	READ ONLY MEMORY
9	VREF	VOLTAGE REFERENCE
10	IC	INTEGRATED CIRCUIT
11	AC	ALTERNATING CURRENT
12	DC	DIRECT CURRENT
13	PSP	PARALLEL SLAVE PORT
14	FIFO	FIRST IN FIRST OUT
15	SFR	SPECIAL FUNCTION REGISTERS
16	EEPROM	ELECTRICALLY ERASABLE PROGRAMMALE READ ONLY MEMORY
17	SPI	SERIAL PERIPHERAL INTERFACE

LIST OF TABLES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

CHAPTER 1

1.1 INTRODUCTION

The objective of voting is to allow voters to exercise their right to express their choices regarding specific issues, pieces of legislation, citizen initiatives, constitutional amendments, recalls and/or to choose their government and political representatives. Technology is being used more and more as a tool to assist voters to cast their votes. To allow the exercise of this right, almost all voting systems around the world include the following steps:

- voter identification and authentication
- voting and recording of votes cast
- vote counting
- publication of election results

Voter identification is required during two phases of the electoral process: first for voter registration in order to establish the right to vote and afterwards, at voting time, to allow a citizen to exercise their right to vote by verifying if the person satisfies all the requirements needed to vote (authentication).

Ancient archeological artifacts and historical items have been discovered to still retain a large number of fingerprints on them. Since this was a discovered significant stride in fingerprinting and identification have been made. In 1788 a detailed description of anatomical formations of fingerprints was made. Then in1823 fingerprints began to be classified into nine categories, (Handbook) and by the 19th century Sir Francis Galton had developed analytical methods for fingerprint matching. As the criminal justice system evolved, there arose the need for criminals to be uniquely identified by some physically identifiable trait. Richard Edward Henry of Scotland Yard began using fingerprinting in 1901 and its success eventually lead to its increased use in the law enforcement field

The field of biometrics was formed and has since expanded on to many types of physical identification. Still, the human fingerprint remains a very common identifier and the biometric method of choice among law enforcement. These concepts of human identification have lead to the development of fingerprint scanners that serve to quickly identify individuals and assign access privileges. The basic point of these devices is also to examine the fingerprint data of an individual and compare it to a database of other fingerprints.

Nearly everyone in the world is born with a fingerprint that is unique; a separate and comprehensively identifying attribute that sets us apart from the other 6.5 billon people that inhabit this world. It is because of this fact that the fingerprint has proven such a useful part of biometric security. The very reason that fingerprint scanners are useful can be found in this fact as well. However, this is far from the only reason they are used.

Another important reason fingerprint scanners are used is, they provide a quick, easy, efficient, and secure measure through which, an individual with the proper access privileges can authenticate. The fingerprint of an employee for example, is stored in a database that the scanner queries every time it is used. There are two basic Boolean conditions the scanner then goes through when an individual's print is scanned. First, the print is usually searched for in a database of fingerprints, once it is found it then looks at the print to see what access privileges are associated with the print and compares them to the access they are trying to gain. If everything checks out the subject is allowed access and they are not otherwise. In any case, a log of the event is usually stored for security purposes the size of these devices is another reason they have become so mainstream recently. Fingerprint scanners can be deployed directly near a door for access or as a peripheral to a computer for logging in. Modern day scanners have even been embedded on computer keyboards, mice, and USB devices because engineers have been able to reduce their size. Fingerprint scanners are also very versatile in the function that they can serve. The most common use may be for access restriction; however, they have served as time clocks, personal data retrievers, and even to cut down on truancy in some schools. Since they have experienced so much success in these areas, businesses are expanding upon their use and they are getting more public exposure

Finger printing recognition, the electronic methods of recording and recognizing an individual finger print, advanced substantially during the last decade of the 21th century. Today, identification can be achieved in a few seconds with reasonable accuracy. As a result, the use of automated fingerprint identification systems (AFIS) that record, store, search, match and identify finger prints is rapidly expanding. AFIS can be integrated with a microcontroller and other peripherals to form an embedded system which is a comprehensive electronic voting machine with fingerprint print identification system.

CHAPTER 2

LITERATURE SURVEY

CHAPTER 2

2.1 LITERATURE SURVEY

LITERATURE 1:

G.Kalaiyarasi, K. Balaj, T.Narmadha, V.Naveen propose in the paper **"E-Voting System In Smart Phone Using Mobile Application", 2020.** The development in the web technologies given growth to the new application that will make the voting process very easy and proficient. The E-voting helps in providing convenient, capture and count the votes in an election. This project provides the description about e-voting using an Android platform. The proposed e-voting system helps the user to cast the vote without visiting the polling booth. The application provides authentication measures in order to avoid fraud voters using the OTP. Once the voting process is finished the results will be available within a fraction of seconds. All the casted vote count is encrypted using AES256 algorithm and stored in the database in order to avoid any outbreaks and revelation of results by third person other than the administrator. The percentage of people those who cast votes get increased since this E voting application is available in the playstore so that they no need to travel for the purpose of casting the votes which is registered in their native. The transportation charge will be decreased for carrying the electronic voting machine to pooling booths. The human resources for conducting the election and counting the votes will be reduced. The burden of government employees and police protection will be reduced by using E voting application. The problem arising conflicts between the candidate and the election parties can be ignored by this E –voting mobile application.

LITERATURE 2:

Ch.Jaya Lakshmi ,S.Kalpana describe in the paper **"SECURED AND TRANSPARENT VOTING SYSTEM USING BIOMETRICS", 2018.** Every citizen or voter of India is allowed to exercise their right to express their choices regarding specific issues, pieces of legislation, citizen initiatives, constitutional amendments, recalls and/or to choose their government and political representative's through casting their votes. To allow the exercise of this right, almost all voting systems include the following steps: voter identification and authentication, voting and recording of votes cast, vote counting, publication of election results. Voter identification is required during electoral process. Security is a heart of e-voting process. Therefore the necessity of designing a secure e-voting process is very important. A secured electronic voting machine using unique identification number i.e. AADHAR number has been developed. To provide additional security along with the AADHAR number biometric identification is used. At the time of voting in the elections, the voter authentication can be done through biometric pattern. If the biometric information of the voter matches the database of the AADHAR then the person is allowed to cast their vote. Transparency is additional advantage for the above system. The local databases will retrieve only the data that is pertaining to the voting process and exclude all other irrelevant information. These databases will be used for generating statistics and results of the electoral process. These databases make it possible to allow voting from anywhere provided that the voter is within electoral circuits.

LITERATURE 3:

Segundo Moisés Toapanta ,Toapant,Luis Enrique Mafla Gallegos propose a concept **"Model of Shared Secret Applied to a Voting System for the National Electoral Council of Ecuador", 2019.** The vote is one of the pillars of Democracy to make the decision to choose different dignities, ICTs have an important role in governance to optimize resources. The problem is to improve the information security in the electoral process. The objective is to define a trust model that applies Secret Sharing to a voting system, to protect electoral information in the stages of electronic voting. Elections are the common goal of a Democracy where citizens execute their right to vote in order to choose between different candidates; framed on constitution of a nation, with guarantees that must be backed by the internal and external control organisms; the security and integrity of this right when uses physical ballot or electronic systems are the relevant characteristics of the process; Web 2.0 leads to electronic voting systems that improve cost-benefit and democratic quality, especially for social groups with difficult access to electoral sites. The deductive method and the exploration were applied to examine the information of the articles cited. It resulted a Conceptual model of voting system, General design of voting system, SS Algorithm applied to voting system expressed in flow chart, Application prototype and Prognosis of electronic voters. It was concluded that to improve the information integrity in a voting system, the application of a cryptographic algorithm is necessary; Ecuadorian population has 66.36% of internet access that should be exploited to maximize benefits and minimize errors.

LITERATURE 4:

Md. Mahiuddin develop a concept **"Design a Secure Voting System Using Smart Card and Iris Recognition", 2019.** Security is the main concern of existing voting systems. Sometimes an unauthorized person gives vote. Some politicians try to follow illegal method to win the election. In paper ballot and EVM systems needed more manpower. These existing systems are much more time consuming and also slow. In proposed system, we use Irish pattern and smart card. Hence the proposed voting system is more secure than existing system. Smart card is a card in which a microprocessor and a memory chip are attached used for processing and storing information respectively. Secure exchange between the reader and the card is performed in the card more easily. Smart card has the capacity of store and access data. It also provides an immediate exchange of necessary information. We can store a person's iris data and personal information in smart card. With the increasing the population day by day, the improvement of voting system is necessary. Undoubtedly the proposed voting system is techniques are especially good. We have used iris recognition and smart card for improving this system. Many bio metric methods are available but iris recognition has high accuracy rate. Using the smart card, it is likely to poll from any polling booth rather than the particular polling booth. The iris pattern of the person is obviously unique. It reduces the polling time which is most important. It totally rules out the chance of invalid vote. Bangladesh is one of the countries in which introduced voting system in parliamentary and assembly polls. But in every election, the election commission is facing a lot of troubles and various types of problems throughout the election. The most familiar issue faced by the election commission is improper confirmation with respect to the arrangement of casting the votes, duplication or illegal casting of votes. In this paper, a secure and new voting system is developed to improve the existing voting system using smartcard and iris recognition. Iris is one of the most secure biometric of person identification. The main goal of this article is to avoid the duplication of casting votes.

LITERATURE 5:

Yingming Zhao, Yue Pan, Sanchao Wang, and Junxing Zhang proposed a paper **"An Anonymous Voting System Based on Homomorphic Encryption"** 2014. t an electronic voting system based on Homomorphic encryption to ensure anonymity, privacy, and reliability in the voting. Homomorphic encryption is a subset of privacy homomorphism. It is capable of computing the encrypted data directly and also encrypting the result of the operation automatically. For this reason it has a wide range of applications including secure multi-party computation, database encryption, electronic voting, etc. In this paper, we make use of the homomorphic encryption mechanism to design and implement an electronic voting system that supports the separation of privileges among voters, tellers, and announcers. Our experimental results show the system not only ensures anonymity in voting but also presents cheating during the counting process. Homomorphic encryption is a form of encryption. It allows users to perform specific algebraic operations on ciphertext and still get ciphertext as the result, which is as if the same operations are carried out on the corresponding cleartext and then the result is encrypted, since the two results would be the same. The definition of Homomorphic encryption can be described as: Let E denotes the encryption operation, m be the plaintext, and e be the corresponding ciphertext, i.e. e = E (m), then m = E-1(e). Given there is an operation f for plaintext, if we can construct a corresponding function F for E such that F (e) = E (f (m)), then E is a homomorphic encryption algorithm over f.

LITERATURE 6:

Irham Mulkan Rodiana, Budi Rahardjo, Aciek Ida conclude in a concept **"Design of a Public Key Infrastructure-based Single Ballot E-Voting System"** 2018. Electronic voting system (e-voting) has grown rapidly and potentially replaced the conventional polling system that uses paper. This e-voting system is more desirable because of it is convenient to distribute the supporting tools and the ease of data collection. However, several important factors that must be considered in evoting are anonymity and verifiability. Both factors will be refined and combined in the proposed system. The voter will always not be seen directly in his / her identity and may also verify the result of his choice at every stage of e-voting. This paper proposes the design of such system based on public key infrastructure and hash function. In addition, this paper will also present key management on the e-voting system. The design of this system is considered for e-voting applications that are done in a dispersed country such as in Indonesia. proposed e-voting design is done using Public Key Infrastructure (PKI) and Hash Function. This design considering the ease of deployment later in the real world but does not eliminate the anonymity and verifiability aspects of the e-voting system. The proposed design is suitable for wide and scattered countries such as Indonesia. To improve the safety, this design is still a semi online system that depends on the terminal committee. The next development of this design is deployment and evaluation after it is implemented.

LITERATURE 7:

Tao Li, Hao Yang, Yilei Wang, Qiuliang Xu "The Electronic Voting in the Presence of Rational Voters" 2015. —The most distinct character of electronic voting is that voters need not to vote at a certain ballot box. With the development of Internet, electronic voting is becoming an important field in electronic commerce. The basic security requirements for electronic voting are anonymity of the voters, privacy and fairness of the votes. In fact, electronic voting can be regarded as a multi-party computation, where distributed parties wish to securely compute the votes in electronic voting systems. In this paper, we redefined the types of parties in electronic voting by using definitions in rational multi-party computation. More specifically, voters are regarded as rational other than honest or malicious, where voting is considered as a social choice. Rational voters care about their utilities when they decide to vote. We first present a rational secret sharing scheme (RSSS) and then construct an electronic voting protocol based on this RSSS.

LITERATURE 8:

Segundo Moisés Toapanta Toapanta, Luis Antonio Palomino Romero, Felix Gustavo Mendoza Quimi "Model of Shared Secret Applied to a Voting System for the National Electoral Council of Ecuador" 2019. The vote is one of the pillars of Democracy to make the decision to choose different dignities, ICTs have an important role in governance to optimize resources. The problem is to improve the information security in the electoral process. The objective is to define a trust model that applies Secret Sharing to a voting system, to protect electoral information in the stages of electronic voting. The deductive method and the exploration were applied to examine the information of the articles cited. It

resulted a Conceptual model of voting system, General design of voting system, SS Algorithm applied to voting system expressed in flow chart, Application prototype and Prognosis of electronic voters. It was concluded that to improve the information integrity in a voting system, the application of a cryptographic algorithm is necessary; Ecuadorian population has 66.36% of internet access that should be exploited to maximize benefits and minimize errors.

CHAPTER 3

PROJECT DESCRIPTION

CHAPTER 3

PROJECT DESCRIPTION

3.1 OBJECTIVE

In the current scenario, most of the countries of the world hold their elections using Electronic Voting Machines, where your vote gets registered electronically with the help of an Electronic Machine without using and wasting ballot paper to vote for elections. As security is a major concern nowadays, ensuring that no person exercises the right to vote twice is the main aspect. We can resolve this issue by introducing Finger Print Based Voting, where a person can be authorized based on his Finger Print. This will put an end to fake voting. The domain of the project is the Wireless data transfer where we are building Fingerprint Based Biometric Voting Machine using Arduino. We know that Biometric is the system of interrelated computing devices, mechanical and digital machines, objects, and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. Thus our Fingerprint online module is an application where the user is recognized by his finger pattern. As we know that the minutiae features on a finger of each human being are different, the voter can be easily authenticated. The online system allows the voter to vote through his fingerprint. The fingerprint of the voter is used to uniquely and distinctively identify him/her using the fingerprint module. Also, the system promises the right to vote a candidate only once, thus not allowing the voter to vote for the second time. An admin is assigned to add all the candidates standing for the election. Only the admin has the right to add a candidate name and photo who are nominated. The Admin will also register the voter's name by verifying voter’s details. Admin will authenticate the voter by verifying the voter’s identity and then admin will register the voter. Once the user has got the voters id and password from the admin the user can log in and vote for the candidates who were nominated. The system will allow the user to vote for only one candidate. In the end, the election result is published by using the election id. Even users can view the election result.

3.2 PROBLEM STATEMENT

In recent years many studies contributing to online voting, its advantages and disadvantages have been developed. In some cases, Online Voting has been tagged as the ultimate solution and in other cases, dangerous. The use of the insecure Internet, well-documented cases of incorrect implementations and the resulting security breaches have been reported recently. These problems and challenges have to be overcome in order to create public trust in online voting. An Online Voting System should be provided where an online registration form is developed for the voters to cast their vote securely using fingerprint authentication. The system should be developed keeping in mind high security and a user-friendly interface. The era before 2004 used Paper Ballot System. Voters had to go to polling booths and cast their votes by marking on the seal in front of the symbol of a candidate for which they wanted to cast their vote on the ballot paper. It was a very time-consuming process [4]. The advantages of the Paper ballot system include no chance of hacking. Also, a paper ballot is still used internationally. The disadvantages include Paper Ballot is wastage of paper, time-consuming manual ballot counting, booth capturing by means of muscle power, methods of vote manipulation in ballot paper

3.3 SCOPE OF THE PROJECT

The proposed system is based on electronic voting machine. The system is able to identify each voter by getting their fingerprint. Whenever the system will receive a fingerprint, it will match the fingerprint from the database. According to the information given by the database, the system will decide if the person is registered or not. System is also able to distinguish second vote. If a particular voter is not registered voter or tries to cast more than one vote, system will identify him and will restrict from voting. However, if neither case is applicable for a voter, it will allow the voter to cast the vote. The system is designed in such a way, if vote is given to a candidate mistakenly, the voter has the ability to change their decision but only once. Furthermore, just like any other electronic voting machines, the device will count votes for each candidate. It is also able to show the result, after a certain period of time when the voting is over. This is an advantage that it will not require too much time to publish who has won the election. It has very high accuracy rate in case of both identifying voter and counting votes. Another advantage of the system is, it is completely offline system. For this reason, the data cannot be hacked.

3.4 EXISTING METHODOLOGY

- As soon as the last voter has voted, the Polling Officer in-charge of the Control Unit will press the 'Close' Button.
- Thereafter, the EVM will not accept any votes. Further, after the close of poll, the Balloting Unit is disconnected from the Control Unit and kept separately.
- Votes can be recorded only through the Balloting Unit.
- Again the Presiding officer, at the close of the poll, will hand over to each polling agent present an account of votes recorded.
- At the time of counting of votes, the total will be tallied with this account and if there is any discrepancy, this will be pointed out by the Counting Agents.
- During the counting of votes, the results are displayed by pressing the 'Result' button.

3.4.1 DISADVANTAGES

- Now the election seems to be a great messy proceeding. On or before election days transport system totally ceases and maximum surface transport vehicles are taken off the road for election purpose.
- Moreover official works in a majority of public sectors are suspended during election months. Officers and staffs from public sectors are appointed on election duties.
- On a particular election day, the election booths become heavily crowded. People have to stand in the scorching sunlight for hours just to cast “a vote”. Aged people and senior citizens have to face the same problems.

3.5 PROPOSED METHODOLOGY

- Here we planned to interface biometric finger print sensor with Arduino microcontroller, also interface GSM.

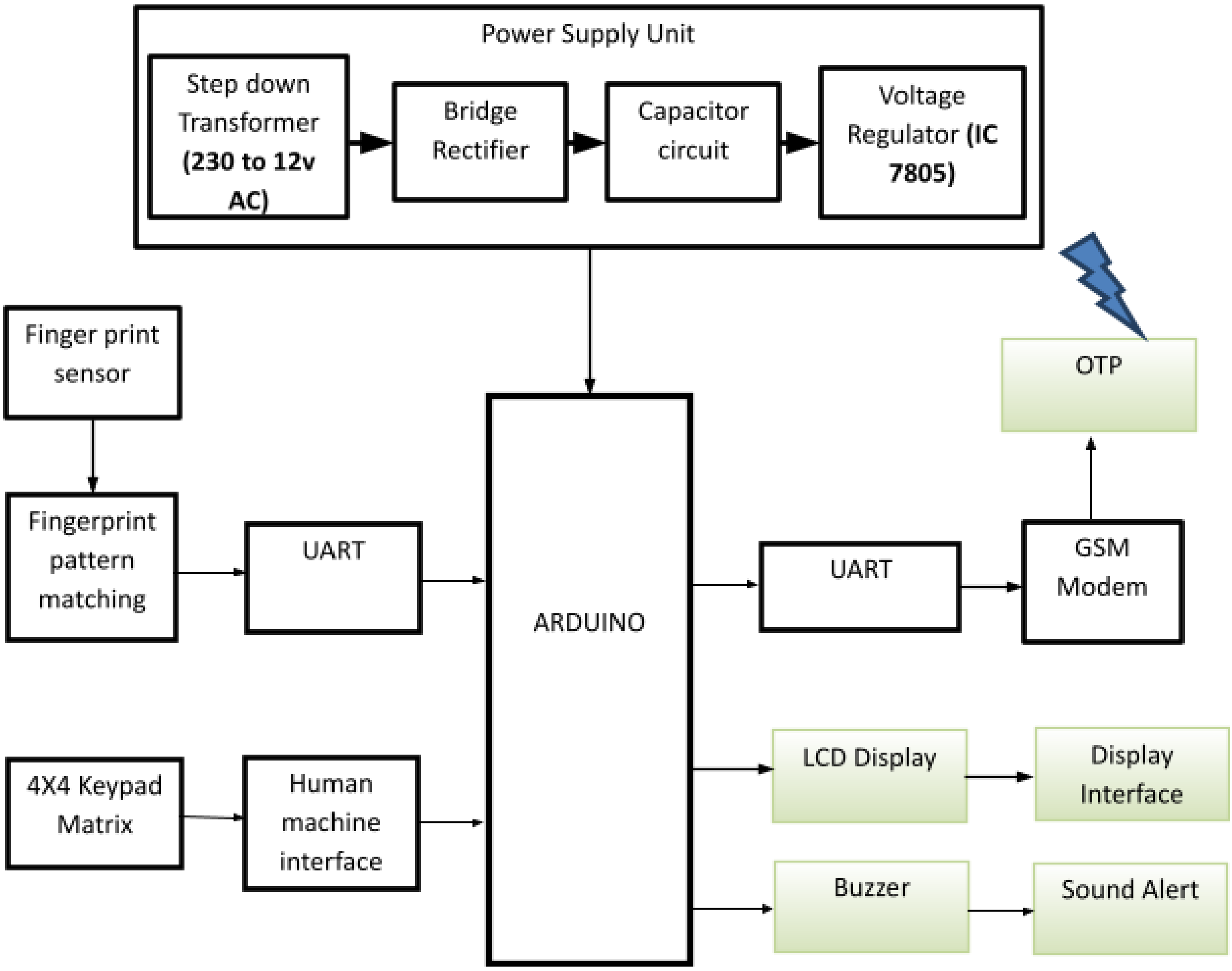
- The fingerprint sensor helps to detect the biometric fingerprint of each and every person, here we choose 3-person fingerprint for demo.
- Once fingerprint sensor provides proper value to controller. Then the control will send the OTP (One Time Password) to registered mobile number. We need this OTP to proceed next.
- Once the OTP is entered properly, the system will provide opportunity to put Vote for any party. Once a person registers his vote, the system will not allow to next chance. Here the LCD display will help to display the details of voting stages.
- The keypad device will help to communicate to the system with human. Buzzer alert will get if any abnormal activities like second time vote attempt.

3.5.1 ADVANTAGES

- The person can register vote properly.
- The system will not allow for multiple votes also unverified vote.
- Simple logic for register people votes.
- To make the voting system was faster and more secure

3.6 WORKING

BLOCK DIAGRAM



CHAPTER 4
PROJECT REQUIREMENTS

CHAPTER 4

PROJECT REQUIREMENTS

4.1 HARDWARE DESCRIPTION

4.1.1 ARDUINO

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz 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 a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter. "Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions.

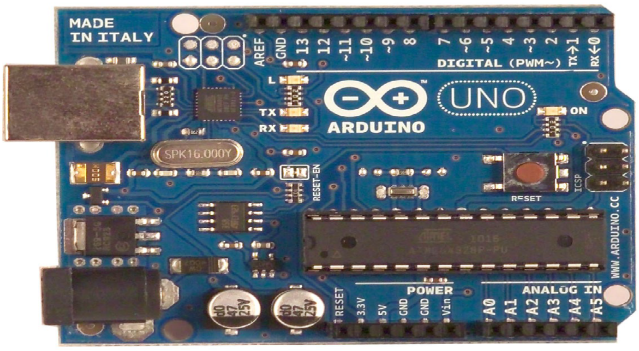


Fig 4.1.1 Arduino

POWER:

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

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

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

THE POWER PINS ARE AS FOLLOWS:

· **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to5 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.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

· **3V3.** A 3.3volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

· **GND.** Ground pins.

INPUT & OUTPUT:

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

· **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data.

4.1.2 FINGER PRINT SENSOR

This all-in-one optical fingerprint sensor will make adding fingerprint detection and verification super simple. These modules are typically used in safes - there's a high powered DSP chip AS601 that does the image rendering, calculation, feature-finding and searching. Connect to any microcontroller or system with TTL serial, and send packets of data to take photos, detect prints, hash and search. You can also enroll new fingers directly - up to 12 finger prints can be stored in the onboard FLASH memory. As the usage, the fingerprint is really easy to use with the serial UART.

Features

- Communication: UART(TTL)
- Fingerprint number: 120 on default
- Can set the security level and baud rate flexibility
- Working Current@Voltage: <120mA@DC3.6~6V
- Temprature: -20 - +50 degrees

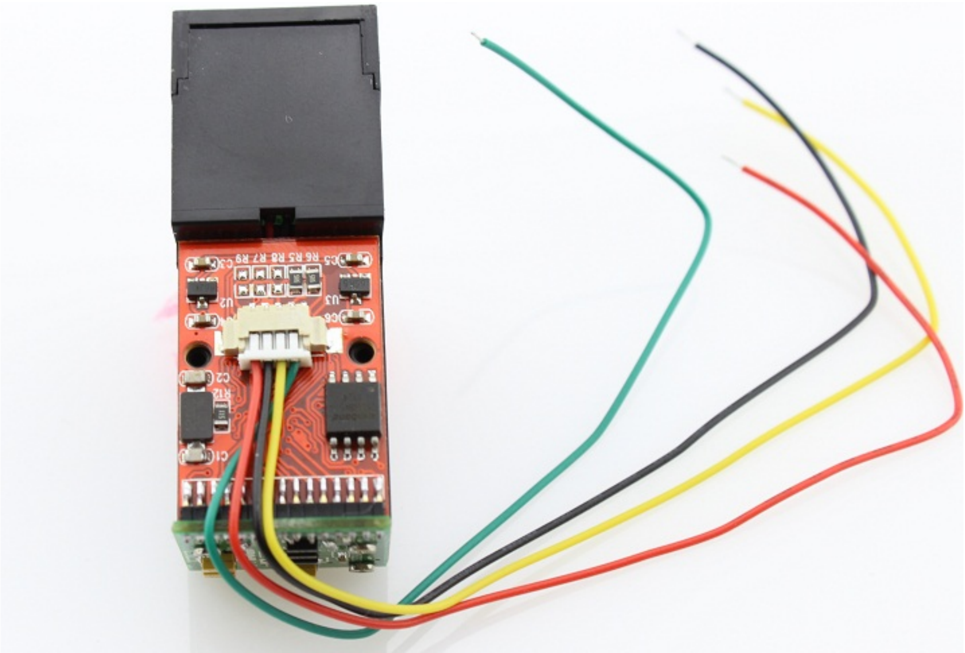


Fig 4.1.2 Finger print sensor PCB

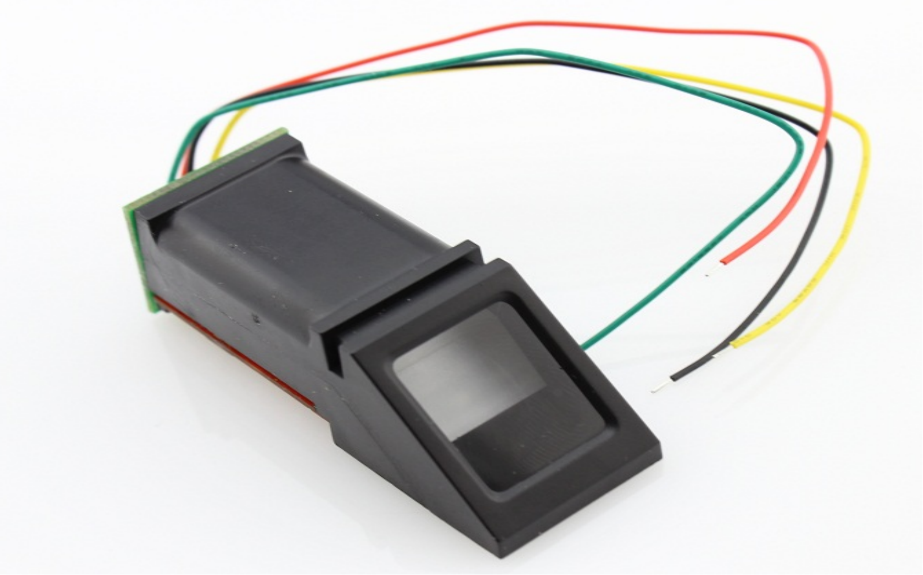


Fig 4.1.2.1 Finger print Scanner

4.1.3 KEY PAD

KEYPAD INTERFACING WITH THE MICROCONTROLLERS

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

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. How it is done is explained next.

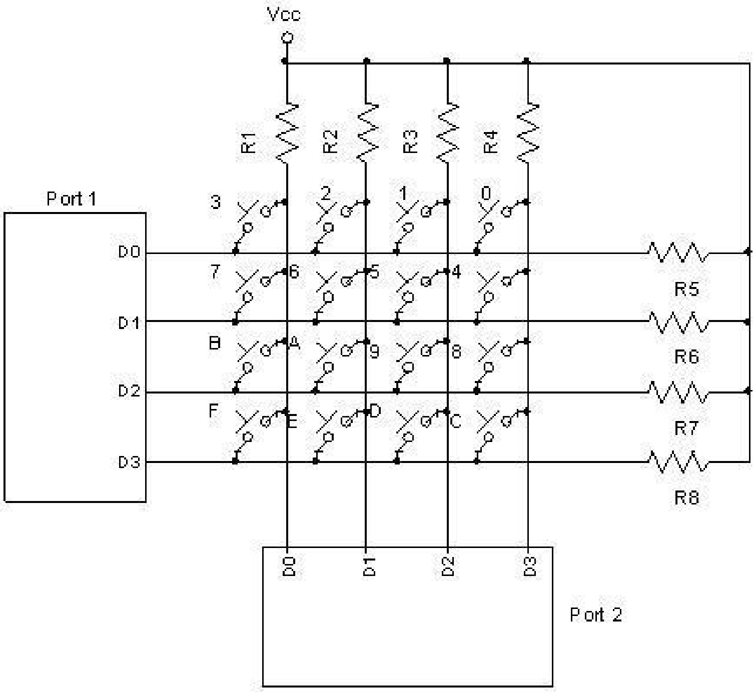


Fig 4.1.3 Matrix keypad circuit

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 D3-D0=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 all1s, 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.

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.

4.1.4 ADC

The ADC0808, ADC0809 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8-single-ended analog signals. The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE outputs.

The design of the ADC0808, ADC0809 has been optimized by incorporating the most desirable aspects of several A/D conversion techniques. The ADC0808, ADC0809 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make this device ideally suited to applications from process and machine control to consumer and automotive applications. For 16-channel multiplexer with common output (sample/hold port) see ADC0816 data sheet. (See AN-247 for more information.)

FEATURES

■ Easy interface to all microprocessors

■ Operates ratiometrically or with 5 VDC or analog span
adjusted voltage reference

■ No zero or full-scale adjust required

■ 8-channel multiplexer with address logic

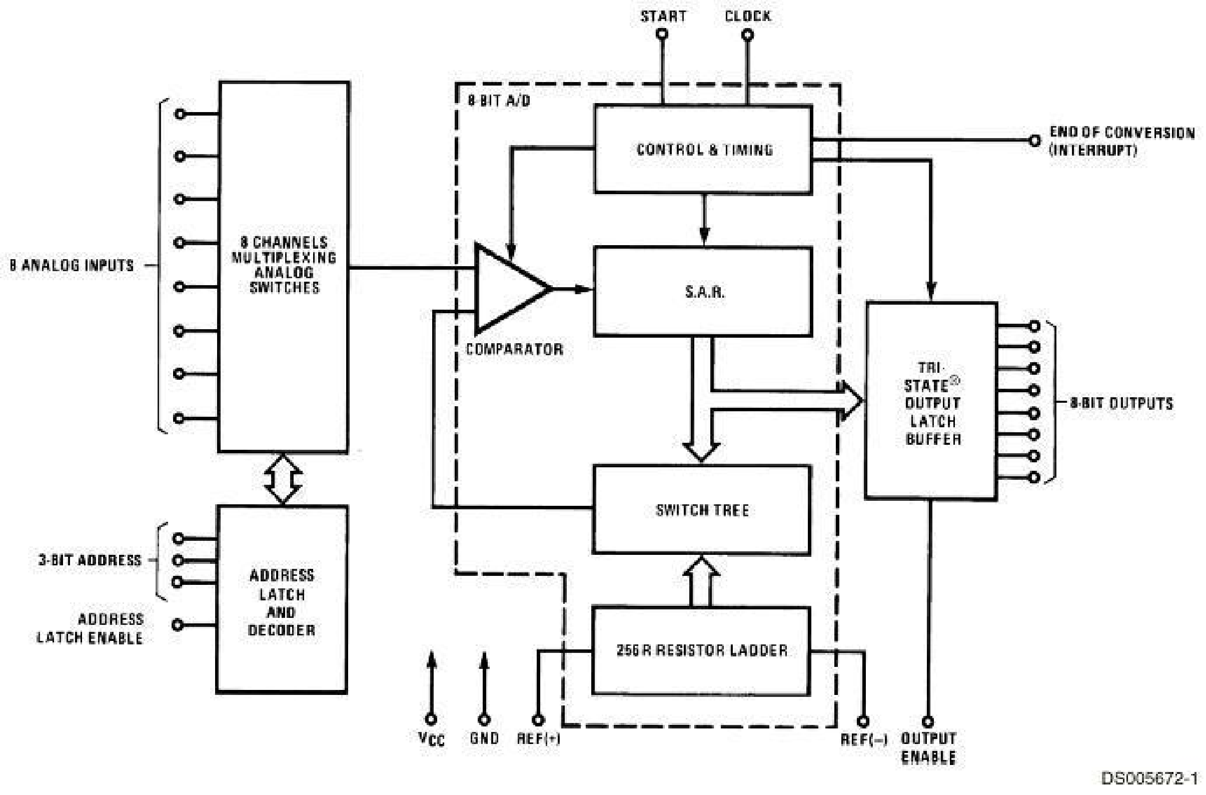


Fig 4.1.4 ADC Channel Diagram

ADC TYPES

These are the most common ways of implementing an electronic ADC:

- A **direct-conversion ADC** or **flash ADC**
- A **successive-approximation ADC**
- A **ramp-compare ADC**
- The **Wilkinson ADC**
- An **integrating ADC** (also **dual-slope** or **multi-slope** ADC)
- A **delta-encoded ADC** or **counter-ramp**

4.1.5 UART

A **universal asynchronous receiver/transmitter**, abbreviated **UART** /ˈjuːɑrt/, is a computer hardware device that translates data between parallel and serial forms. UARTs are commonly used in conjunction with communication standards such as TIA (formerly EIA)RS-232, RS-422 or RS-485. The universal designation indicates that the data format and transmission speeds are configurable. The electric signalling levels and methods (such as differential signalling etc.) are handled by a driver circuit external to the UART.

A UART is usually an individual (or part of an) integrated circuit (IC) used for serial communications over a computer or peripheral device serial port. UARTs are now commonly included in microcontrollers. A dual UART, or DUART, combines two UARTs into a single chip. An octal UART or OCTART combines eight UARTs into one package, an example being the NXP SCC2698. Many modern ICs now come with a UART that can also communicate synchronously; these devices are called USARTs (universal synchronous/asynchronous receiver/transmitter).

TRANSMITTING AND RECEIVING SERIAL DATA

The universal asynchronous receiver/transmitter (UART) takes bytes of data and transmits the individual bits in a sequential fashion.[1] At the destination, a second UART re-assembles the bits into complete bytes. Each UART contains a shift register, which is the fundamental method of conversion between serial and parallel forms. Serial transmission of digital information (bits) through a single wire or other medium is less costly than parallel transmission through multiple wires.

The UART usually does not directly generate or receive the external signals used between different items of equipment. Separate interface devices are used to convert the logic level signals of the UART to and from the external signalling levels. External signals may be of many different forms. Examples of standards for voltage signaling are RS-232, RS-422 and RS-485 from the EIA. Historically, current (in current loops) was used in telegraph circuits. Some signaling schemes do not use electrical wires. Examples of such areoptical fiber, IrDA (infrared), and (wireless) Bluetooth in its Serial Port Profile (SPP). Some signaling schemes use modulation of a carrier signal (with or without wires). Examples are modulation of audio signals with phone line modems, RF modulation with data radios, and the DC-LIN for power line communication.

Communication may be *simplex* (in one direction only, with no provision for the receiving device to send information back to the transmitting device), *full duplex* (both devices send and receive at the same time) or *half duplex* (devices take turns transmitting and receiving).

DATA FRAMING

Bit number	1	2	3	4	5	6	7	8	9	10	11
	Start bit	**5–8 data bits**								**Stop bit(s)**	
	Start	Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Stop	

TABLE 4.1.5 Data Framing

The idle, no data state is high-voltage, or powered. This is a historic legacy from telegraphy, in which the line is held high to show that the line and transmitter are not damaged. Each character is sent as a logic low start bit, a configurable number of data bits (usually 8, but users can choose 5 to 8 or 9 bits depending on which UART is in use), an optional parity bit if the number of bits per character chosen is not 9 bits, and one or more logic high stop bits. In most applications the least significant data bit (the one on the left in this diagram) is transmitted first, but there are exceptions (such as the IBM 2741 printing terminal).

4.1.5.1 RECEIVER

All operations of the UART hardware are controlled by a clock signal which runs at a multiple of the data rate, typically 8 times the bit rate. The receiver tests the state of the incoming signal on each clock pulse, looking for the beginning of the start bit. If the apparent start bit lasts at least one-half of the bit time, it is valid and signals the start of a new character. If not, it is considered a spurious pulse and is ignored. After waiting a further bit time, the state of the line is again sampled and the resulting level clocked into a shift register. After the required number of bit periods for the character length (5 to 8 bits, typically) have elapsed, the contents of the shift register are made available (in parallel fashion) to the receiving system. The UART will set a flag indicating new data is available, and may also generate a processor interrupt to request that the host processor transfers the received data.

Communicating UARTs usually have no shared timing system apart from the communication signal. Typically, UARTs resynchronize their internal clocks on each change of the data line that is not considered a spurious pulse. Obtaining timing information in this manner, they reliably receive when the transmitter is sending at a slightly different speed than it should. Simplistic UARTs do not do this, instead they resynchronize on the falling edge of the start bit only, and then read the center of each expected data bit, and this system works if the broadcast data rate is accurate enough to allow the stop bits to be sampled reliably.

4.1.5.2 TRANSMITTER

Transmission operation is simpler as the timing does not have to be determined from the line state, nor is it bound to any fixed timing intervals. As soon as the sending system deposits a character in the shift register (after completion of the previous character), the UART generates a start bit, shifts the required number of data bits out to the line, generates and sends the parity bit (if used), and sends the stop bits. Since transmission of a single character may take a long time relative to CPU speeds, the UART maintains a flag showing busy status so that the host system does not deposit a new character for transmission until the previous one has been completed; "ready for next character" may also be signaled with an interrupt. Since full-duplex operation requires characters to be sent and received at the same time, UARTs use two different shift registers for transmitted and received characters.

4.1.5.3 APPLICATION

Transmitting and receiving UARTs must be set for the same bit speed, character length, parity, and stop bits for proper operation. The receiving UART may detect some mismatched settings and set a "framing error" flag bit for the host system; in exceptional cases the receiving UART will produce an erratic stream of mutilated characters and transfer them to the host system.

Typical serial ports used with personal computers connected to modems use eight data bits, no parity, and one stop bit; for this configuration the number of ASCII characters per second equals the bit rate divided by 10.

Some very low-cost home computers or embedded systems dispense with a UART and use the CPU to sample the state of an input port or directly manipulate an output port for data transmission. While very CPU-intensive (since the CPU timing is critical), the UART chip can thus be omitted, saving money and space. The technique is known as bit-banging.

STRUCTURE

A UART usually contains the following components:

- a clock generator, usually a multiple of the bit rate to allow sampling in the middle of a bit period.
- input and output shift registers
- transmit/receive control
- read/write control logic
- transmit/receive buffers (optional)
- parallel data bus buffer (optional)
- First-in, first-out (FIFO) buffer memory (optional)

4.1.6 PROTOCOL

In telecommunications, a communications protocol is a system of rules that allow two or more entities of a communications system to transmit information via any kind of variation of a physical quantity. These are the rules or standard that defines the syntax, semantics and synchronization of communication and possible error recovery methods. Protocols may be implemented by hardware, software, or a combination of both.

Communicating systems use well-defined formats (protocol) for exchanging messages. Each message has an exact meaning intended to elicit a response from a range of possible responses pre-determined for that particular situation. The specified behaviour is typically independent of how it is to be implemented. Communications protocols have to be agreed upon by the parties involved. To reach agreement, a protocol may be developed into a technical standard. A programming language describes the same for computations, so there is a close analogy between protocols and programming languages: protocols are to communications as programming languages are to computations.

Communicating systems

The information exchanged between devices—through a network, or other media—is governed by rules and conventions that can be set out in technical specifications called communications protocol standards. The nature of a communication, the actual data exchanged and any state-dependent behaviours, is defined by its specification.

In digital computing systems, the rules can be expressed by algorithms and data structures. Expressing the algorithms in a portable programming language makes the protocol software operating system independent.Operating systems usually contain of a set of cooperating

processes that manipulate shared data to communicate with each other. This communication is governed by well-understood protocols, which can be embedded in the process code itself.
In contrast, because there is no common memory, communicating systems have to communicate with each other using a shared transmission medium. Transmission is not necessarily reliable, and individual systems may use different hardware and/or operating systems.

Data formats for data exchange. Digital message bitstrings are exchanged. The bitstrings are divided in fields and each field carries information relevant to the protocol. Conceptually the bitstring is divided into two parts called the header area and the data area. The actual message is stored in the data area, so the header area contains the fields with more relevance to the protocol. Bitstrings longer than the maximum transmission unit (MTU) are divided in pieces of appropriate size.

Address formats for data exchange. Addresses are used to identify both the sender and the intended receiver(s). The addresses are stored in the header area of the bitstrings, allowing the receivers to determine whether the bitstrings are intended for themselves and should be processed or should be ignored. A connection between a sender and a receiver can be identified using an address pair (sender address, receiver address). Usually, some address values have special meanings. An all-1s address could be taken to mean an addressing of all stations on the network, so sending to this address would result in a broadcast on the local network. The rules describing the meanings of the address value are collectively called an addressing scheme.

Address mapping. Sometimes protocols need to map addresses of one scheme on addresses of another scheme. For instance, to translate a logical IP address specified by the application to an Ethernet hardware address. This is referred to as address mapping.[15]
Routing. When systems are not directly connected, intermediary systems along the route to the intended receiver(s) need to forward messages on behalf of the sender. On the Internet, the networks are connected using routers. This way of connecting networks is called internetworking.

Protocol design

Communicating systems operate in parallel. The programming tools and techniques for dealing with parallel processes are collectively called *concurrent programming*. Concurrent programming only deals with the synchronization of communication. The syntax and semantics of the communication governed by a low-level protocol usually have modest complexity, so they can be coded with relative ease. High-level protocols with relatively large complexity could however merit the implementation of language interpreters. An example of the latter case is the HTML language.

Concurrent programming has traditionally been a topic in operating systems theory texts. Formal verification seems indispensable, because concurrent programs are notorious for the hidden and sophisticated bugs they contain. A mathematical approach to the study of concurrency and communication is referred to as *Communicating Sequential Processes* (CSP). Concurrency can also be modelled using finite state machines like Mealy and Moore machines. Mealy and Moore machines are in use as design tools in digital electronics systems, which we encounter in the form of hardware used in telecommunications or electronic devices in general.

This kind of design can be a bit of a challenge to say the least, so it is important to keep things simple. For the Internet protocols, in particular and in retrospect, this meant a basis for protocol design was needed to allow decomposition of protocols into much simpler, cooperating protocols.

4.1.6.1 LAYERING

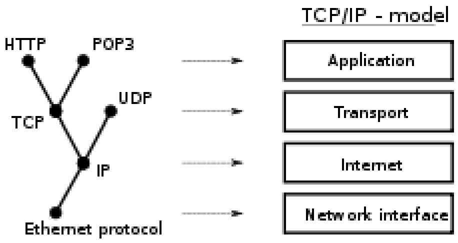


Fig 4.1.6.1 Layering Structure

The TCP/IP model or Internet layering scheme and its relation to some common protocols. The communications protocols in use on the Internet are designed to function in very diverse and complex settings. To ease design, communications protocols are structured using a layering scheme as a basis. Instead of using a single universal protocol to handle all transmission tasks, a set of cooperating protocols fitting the layering scheme is used. The layering scheme in use on the Internet is called the TCP/IP model. The actual protocols are collectively called the Internet protocol suite. The group responsible for this design is called the Internet Engineering Task Force (*IETF*).

4.1.6.2PROTOCOL LAYERING

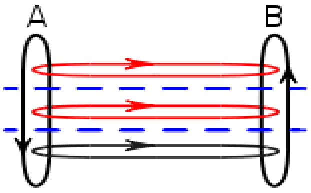


Fig 4.1.6.2 Protocol Layering

Message flows using a protocol suite. Black loops show the actual messaging loops, red loops are the effective communications between layers enabled by the lower layers.

Protocol layering now forms the basis of protocol design. It allows the decomposition of single, complex protocols into simpler, cooperating protocols, but it is also a functional decomposition, because each protocol belongs to a functional class, called a *protocol layer*. The protocol layers each solve a distinct class of communication problems. The Internet protocol suite consists of the following layers: application-, transport-, internet- and network interface-functions. Together, the layers make up a *layering scheme* or *model*.

4.1.7 GSM

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

CIRCUIT

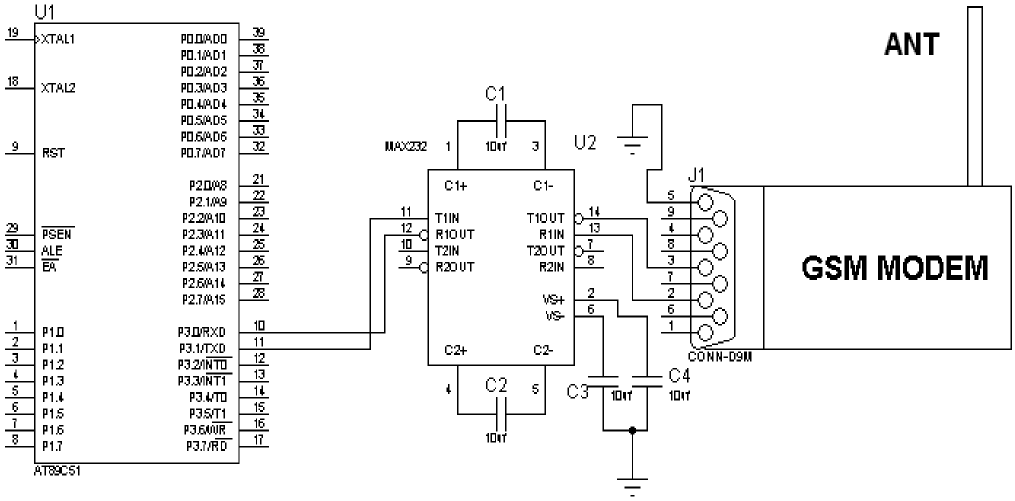


Fig 4.1.7 GSM Circuit

DEFINITION

Global system for mobile communication (GSM) is a globally accepted standard for digital cellular communication. GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard that would formulate specifications for a pan-European mobile cellular radio system operating at 900 MHz.

4.1.7.1 FREQUENCIES

Originally it had been intended that GSM would operate on frequencies in the 900 MHz cellular band. In September 1993, the British operator Mercury One-to-One launched a network. Termed DCS 1800 it operated at frequencies in a new 1800 MHz band. By adopting new frequencies new operators and further competition was introduced into the market apart from allowing additional spectrum to be used and further increasing the overall capacity. This trend was followed in many countries, and soon the term DCS 1800 was dropped in favour of calling it GSM as it was purely the same cellular technology but operating on a different frequency band. In view of the higher frequency used the distances the signals travelled was slightly shorter but this was compensated for by additional base stations.

Introduction to GSM Wireless Modems

What is a GSM Modem?

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

A GSM modem can be an external device or a PC Card / PCMCIA Card. Typically, an external GSM modem is connected to a computer through a serial cable or a USB cable. A GSM modem in the form of a PC Card / PCMCIA Card is designed for use with a laptop computer. It should be inserted into one of the PC Card / PCMCIA Card slots of a laptop computer. Like a GSM mobile phone, a GSM modem requires a SIM card from a wireless carrier in order to operate. As mentioned in earlier sections of this SMS tutorial, computers use AT commands to control modems. Both GSM modems and dial-up modems support a common set of standard AT commands. You can use a GSM modem just like a dial-up modem. In addition to the standard AT commands, GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards. With the extended AT commands, you can do things like:

- Reading, writing and deleting SMS messages.
- Sending SMS messages.
- Monitoring the signal strength.
- Monitoring the charging status and charge level of the battery.
- Reading, writing and searching phone book entries.

The number of SMS messages that can be processed by a GSM modem per minute is very low -- only about six to ten SMS messages per minute.

4.1.8 LCD

INTRODUCTION:

Liquid crystal cell displays (LCDs) are used in similar applications where LEDs are used. These applications are display of display of numeric and alphanumeric characters in dot matrix and segmental displays.

LCDS ARE OF TWO TYPES:

I. Dynamic scattering type
II. Field effect type

THE CONSTRUCTION OF A DYNAMIC SCATTERING LIQUID CRYSTAL CELL:

The liquid crystal material may be one of the several components, which exhibit optical properties of a crystal though they remain in liquid form. Liquid crystal is layered between glass sheets with transparent electrodes deposited on the inside faces. When a potential is applied across the cell, charge carriers flowing through the liquid disrupt the molecular alignment and produce turbulence. When the liquid is not activated, it is transparent. When the liquid is activated the molecular turbulence causes light to be scattered in all directions and the cell appear to be bright.

LCD consists of two glass panels, with the liquid crystal materials sandwiched in between them. The inner surface of the glass plates is coated with transparent electrodes which define in between the electrodes and the crystal, which makes the liquid crystal molecules to maintain a defined orientation angle. When a potential is applied across the cell, charge carriers flowing through the liquid will disrupt the molecular alignment and produce turbulence.

4.1.8.1 WORKING:

When sufficient voltage is applied to the electrodes the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarizer, which would result in activating/highlighting the desired characters. The power supply should be of +5v, with maximum allowable transients of 10mv. To achieve a better/suitable contrast for the display the voltage (VL) at pin 3 should be adjusted properly. A module should not be removed from a live circuit.

The ground terminal of the power supply must be isolated properly so that voltage is induced in it. The module should be isolated properly so that stray voltages are not induced, which could cause a flicking display. LCD is lightweight with only a few, millimeters thickness since the LCD consumes less power, they are compatible with low power electronic circuits, and can be powered for long durations. LCD does not generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. LCDs have long life and a wide operating temperature range. Before LCD is used for displaying proper initialization should be done.

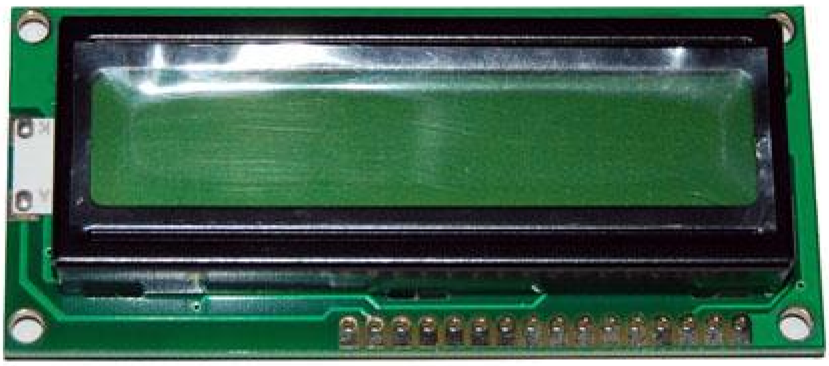


Fig 4.1.8 LCD Display

PIN DESCRIPTION FOR LCD:

PIN NO	SYMBOL	FUNCTION
1	Vss	Ground terminal of Module
2	Vdd	Supply terminal of Module, + 5v
3	Vo	Power supply for liquid crystal drive

4	RS	Register select
		RS=0…Instruction register
		RS=1…Data register

5	R/W	Read/Write R/W=1…Read R/W=0…Write

6	EN	Enable
7-14	DB0-DB7	Bi-directional Data Bus. Data Transfer is performed once ,thru DB0-DB7,incase of interface data length is 8-bits;and twice, thru DB4-DB7 in the case of interface data length is 4-bits.Upper four bits first then lower four bits.
15	LAMP-(L-)	LED or EL lamp power supply terminals
16	LAMP+(L+) (E2)	Enable

Table 4.1.8 LCD Pin Description

LCD PIN DESCRIPTIONS:

The function of each pins of LCD is described below **VCC, VSS and VEE** while v and v provide +5v and ground, respectively, v is used for controlling LCD contrast.

RS, register select

There are two very important registers inside the LCD. The RS pin is used for their selection as follows. If RS=0, the instruction code register is selected, allowing the user to send a command such as clear display, cursor at home,etc.if RS=1 the data register is selected, allowing the user to send data to be displayed on the LCD.

R/W, read/write

R/W input allows the user to write information to the LCD or read information from it. R/W=1 when reading; R/W=0 when writing.

E, enable

The enable pin is used by the LCD to latch information presented on its data pins. When data is supplied to data pins, a high to low pulse must be applied to this pin in order for the LCD to latch in the data present at the data pins.

D0 - D7

The 8-bit data pins, D0 – D7, are used to send information to the LCD or read contents of the LCD'S internal registers. There are also instruction codes that can be sent to the LCD to clear the display or force the cursor to the home position or blink the cursor. RS=0 is used to check the busy flag bit to see if the LCD is ready to receive information. The busy flag is D7 and can be read when R/W=1 and RS=0, as follows: if R/W=1, RS=0.when D7=1, the LCD is busy taking care of internal operation and will not accept any new information, when D7=0, the LCD is ready to receive new information.

APPLICATIONS:

1. Watches
2. Fax & Copy machines & Calculators.

4.1.9 BUZZER

A **buzzer** or **beeper** is a signaling device, The word "buzzer" comes from the rasping noise that buzzers made when they were electromechanical devices, operated from stepped-down AC line voltage at 50 or 60 cycles. Other sounds commonly used to indicate that a button has been pressed are a ring or a beep

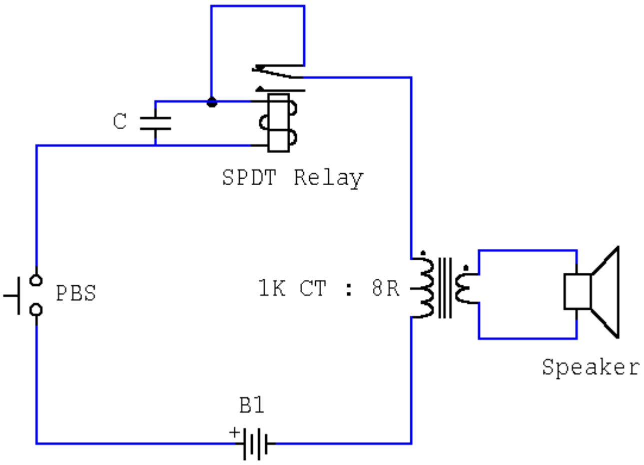


Fig 4.1.9 Buzzer Circuit

This novel buzzer circuit uses a relay in series with a small audio transformer and speaker. When the switch is pressed, the relay will operate via the transformer primary and closed relay contact. As soon as the relay operates the normally closed contact will open, removing power from the relay, the contacts close and the sequence repeats, all very quickly...so fast that the pulse of current causes fluctuations in the transformer primary, and hence secondary.

The speaker tone is thus proportional to relay operating frequency. The capacitor C can be used to "tune" the note. The nominal value is 0.001uF, increasing capacitance lowers the buzzers tone.

4.1.10 POWER SUPPLY

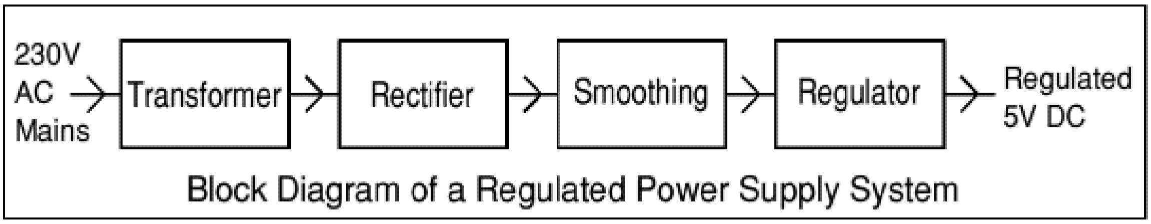
Power supply is a reference to a source of electrical pow. A device or system that supplies electrical or other types of energy to an output load or group of loads is called a power supply unit or PSU. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

Power supplies for electronic devices can be broadly divided into linear and switching power supplies. The linear supply is a relatively simple design that becomes increasingly bulky and heavy for high current devices; voltage regulation in a linear supply can result in low efficiency. A switched-mode supply of the same rating as a linear supply will be smaller, is usually more efficient, but will be more complex.

4.1.10 LINEAR POWER SUPPLY

An AC powered linear power supply usually uses a transformer to convert the voltage from the wall outlet (mains) to a different, usually a lower voltage. If it is used to produce DC, a rectifier is used. A capacitor is used to smooth the pulsating current from the rectifier. Some small periodic deviations from smooth direct current will remain, which is known as ripple. These pulsations occur at a frequency related to the AC power frequency (for example, a multiple of 50 or 60 Hz).

The voltage produced by an unregulated power supply will vary depending on the load and on variations in the AC supply voltage. For critical electronics applications a linear regulator will be used to stabilize and adjust the voltage. This regulator will also greatly reduce the ripple and noise in the output direct current. Linear regulators often provide current limiting, protecting the power supply and attached circuit from over current.



Block Diagram of a Regulated Power Supply System

4.1.10.2 TRANSFORMER

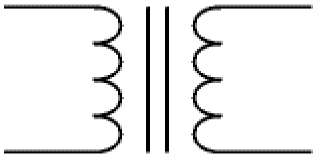


Fig 4.1.10.2 Transformer Circuit

Transformers convert AC electricity from one voltage to another with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC.

Step-up transformers increase voltage; step-down transformers reduce voltage. Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (230V in UK) to a safer low voltage.

The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the core.

Turns ratio=Vp/Vs=Nn/Ns and Power out=Power in

Vs*Is=Vp * Ip

Vp = primary (input) voltage   Vs = secondary (output) voltage
Np = number of turns on primary   Ns = number of turns on secondary
Ip = primary (input) current   Is = secondary (output) current

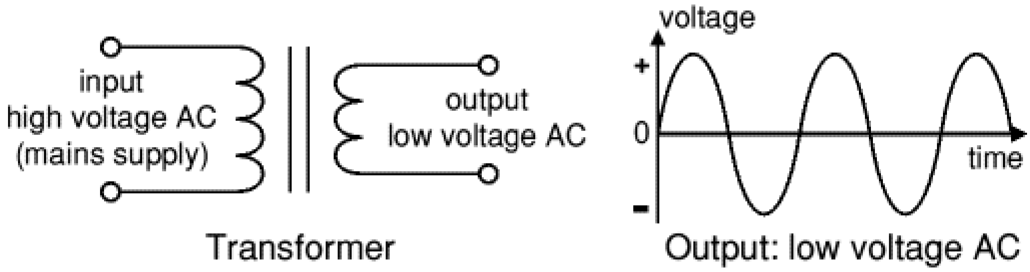


Fig 4.1.10.3 Transformer model

The low voltage AC output is suitable for lamps, heaters and special AC motors. It is not suitable for electronic circuits unless they include a rectifier and a smoothing capacitor.

4.1.10.3 RECTIFIER:

There are several ways of connecting diodes to make a rectifier to convert AC to DC. The bridge rectifier is the most important and it produces full-wave varying DC. A full-wave rectifier can also be made from just two diodes if a center-tap transformer is used, but this method is rarely used now that diodes are cheaper. A single diode can be used as a rectifier but it only uses the positive (+) parts of the AC wave to produce half-wave varying DC.

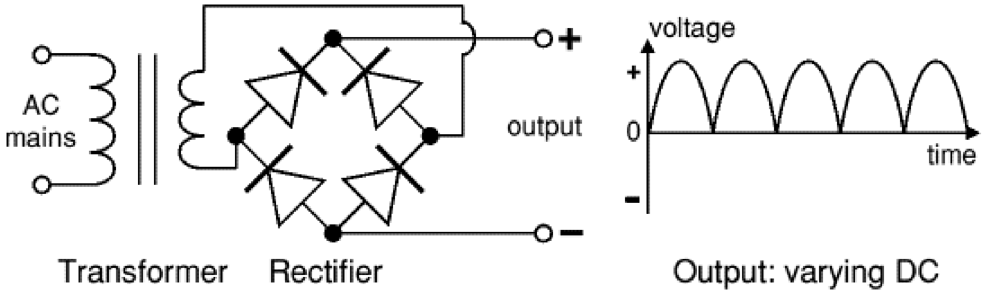


Fig 4.1.10.4 Rectifier circuit

SOFTWARE IMPLEMENTATION

4.2 SOFTWARE DESCRIPTION

4.2.1 ARDUINO IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

4.2.1.1 WRITING SKETCHES

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension **.ino**. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

4.2.1.2 SKETCHBOOK

The Arduino Software (IDE) uses the concept of a sketchbook: a standard place to store your programs (or sketches). The sketches in your sketchbook can be opened from the File > Sketchbook menu or from the Open button on the toolbar. The first time you run the Arduino software, it will automatically create a directory for your sketchbook. You can view or change the location of the sketchbook location from with the Preferences dialog. Beginning with version 1.0, files are saved with a .ino file extension. Previous versions use the. pde extension. You may still open .pde named files in version 1.0 and later, the software will automatically rename the extension to .ino.

Tabs, Multiple Files, and Compilation

Allows you to manage sketches with more than one file (each of which appears in its own tab). These can be normal Arduino code files (no visible extension), C files (.c extension), C++ files (.cpp), or header files (.h).

Uploading

Before uploading your sketch, you need to select the correct items from the Tools > Board and Tools > Port menus. Theboards are described below. On the Mac, the serial port is probably something like /dev/tty.usbmodem241 (for an Uno or Mega2560 or Leonardo) or /dev/tty.usbserial-1B1 (for a Duemilanove or earlier USB board), or/dev/tty.USA19QW1b1P1.1 (for a serial board connected with a Keyspan USB-to-Serial adapter). On Windows, it's probably COM1 or COM2 (for a serial board) or COM4, COM5, COM7, or higher (for a USB board) - to find out, you look for USB serial device in the ports section of the Windows Device Manager. On Linux, it should be /dev/ttyACMx ,/dev/ttyUSBx or similar. Once you've selected the correct serial port and board, press the upload button in the toolbar or select the Upload item from the File menu. Current Arduino boards will reset automatically and begin the upload. With older boards (pre-Diecimila) that lack auto-reset, you'll need to press the reset button on the board just before starting the upload. On most boards, you'll see the RX and TX LEDs blink as the sketch is uploaded. The Arduino Software (IDE) will display a message when the upload is complete, or show an error.

When you upload a sketch, you're using the Arduino bootloader, a small program that has been loaded on to the microcontroller on your board. It allows you to upload code without using any additional hardware. The bootloader is active for a few seconds when the board resets; then it starts whichever sketch was most recently uploaded to the microcontroller. The bootloader will blink the on-board (pin 13) LED when it starts (i.e. when the board resets).

Libraries

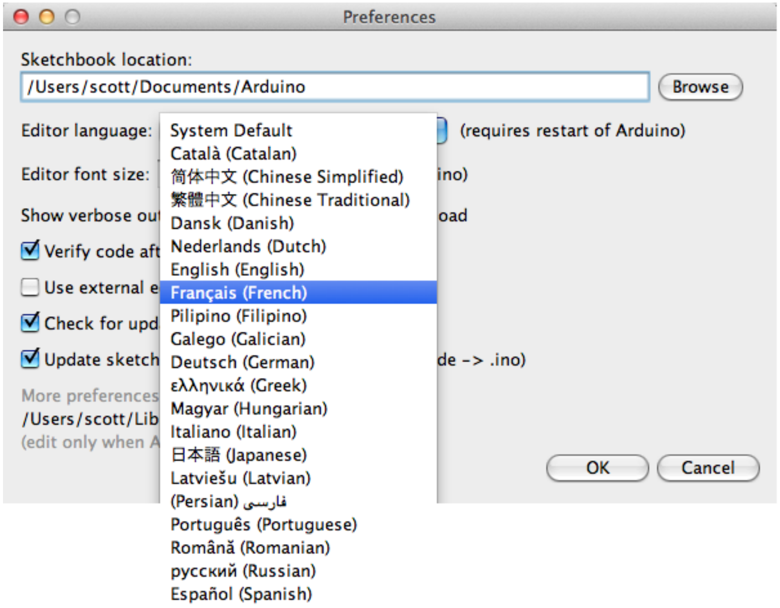
Libraries provide extra functionality for use in sketches, e.g. working with hardware or manipulating data. To use a library in a sketch, select it from the Sketch > Import Library menu. This will insert one or more #include statements at the top of the sketch and compile the library with your sketch. Because libraries are uploaded to the board with your sketch, they increase the amount of space it takes up. If a sketch no longer needs a library, simply delete its #includestatements from the top of your code.

Serial Monitor

Displays serial data being sent from the Arduino or Genuino board (USB or serial board). To send data to the board, enter text and click on the "send" button or press enter. Choose the baud rate from the drop-down that matches the rate passed to Serial.begin in your sketch. Note that on Windows, Mac or Linux, the Arduino or Genuino board will reset (rerun your sketch execution to the beginning) when you connect with the serial monitor.

You can also talk to the board from Processing, Flash, MaxMSP, etc (see the interfacing page for details).

Language Support



Preferences

Sketchbook location: /Users/scott/Documents/Arduino

Editor language: Français (French) (requires restart of Arduino)

Fig 4.2.1.3 Sketch Diagram

Since version 1.0.1 , the Arduino Software (IDE) has been translated into 30+ different languages. By default, the IDE loads in the language selected by your operating system. (Note: on Windows and possibly Linux, this is determined by the locale setting which controls currency and date formats, not by the language the operating system is displayed in.)

If you would like to change the language manually, start the Arduino Software (IDE) and open the Preferences window. Next to the Editor Language there is a dropdown menu of currently supported languages. Select your preferred language from the menu, and restart the software to use the selected language. If your operating system language is not supported, the Arduino Software (IDE) will default to English.

4.2.2 EMBEDDED C

Embedded C is a set of language extensions for the C Programming language by the C Standards committee to address commonality issues that exist between C extensions for different embedded systems. Historically, embedded C programming requires nonstandard extensions to the C language in order to support exotic features such as fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations.

In 2008, the C Standards Committee extended the C language to address these issues by providing a common standard for all implementations to adhere to. It includes a number of features not available in normal C, such as, fixed-point arithmetic, named address spaces, and basic I/O hardware addressing. Embedded C uses most of the syntax and semantics of standard C, e.g., main() function, variable definition, datatype declaration, conditional statements (if, switch, case), loops (while, for), functions, arrays and strings, structures and union, bit operations, macros, etc.

4.2.2.1 NECESSITY

During infancy years of microprocessor-based systems, programs were developed using assemblers and fused into the EPROMs. There used to be no mechanism to find what the program was doing. LEDs, switches, etc. were used to check for correct execution of the program. Some 'very fortunate' developers had In-circuit Simulators (ICEs), but they were too costly and were not quite reliable as well. As time progressed, use of microprocessor-specific assembly-only as the programming language reduced and embedded systems moved onto C as the embedded programming language of choice. C is the most widely used programming language for embedded processors/controllers. Assembly is also used but mainly to implement those portions of the code where very high timing accuracy, code size efficiency, etc. are prime requirements.

As assembly language programs are specific to a processor, assembly language didn't offer portability across systems. To overcome this disadvantage, several high level languages, including C, came up. Some other languages like PLM, Modula-2, Pascal, etc. also came but couldn't find wide acceptance. Amongst those, C got wide acceptance for not only embedded systems, but also for desktop applications. Even though C might have lost its sheen as mainstream language for general purpose applications, it still is having a strong-hold in embedded programming.

ADVANTAGES

- It is small and simpler to learn, understand, program and debug.
- Compared to assembly language, C code written is more reliable and scalable, more portable between different platforms.
- C compilers are available for almost all embedded devices in use today, and there is a large pool of experienced C programmers.
- Unlike assembly, C has advantage of processor-independence and is not specific to any particular microprocessor/microcontroller or any system. This makes it convenient for a user to develop programs that can run on most of the systems.

EMBEDDED SYSTEMS PROGRAMMING

Embedded systems programming is different from developing applications on a desktop computer. Key characteristics of an embedded system, when compared to PCs, are as follows: Embedded devices have resource constraints (limited ROM, limited RAM, limited stack space, less processing power) Components used in embedded system and PCs are different; embedded systems typically use smaller, less power consuming components. · Embedded systems are more tied to the hardware.

Two salient **features of Embedded Programming** are code speed and code size. Code speed is governed by the processing power, timing constraints, whereas code size is governed by available program memory and use of programming language. Goal of embedded system programming is to get maximum features in minimum space and minimum time.Embedded systems are programmed using different type of languages:

- Machine Code
- Low level language, i.e., assembly
- High level language like C, C++, Java, Ada, etc.
- Application level language like Visual Basic, scripts, Access, etc.

Assembly language maps mnemonic words with the binary machine codes that the processor uses to code the instructions. Assembly language seems to be an obvious choice for programming embedded devices. However, use of assembly language is restricted to developing efficient codes in terms of size and speed. Also, assembly codes lead to higher software development costs and code portability is not there. Developing small codes are not much of a problem, but large programs/projects become increasingly difficult to manage in assembly language. Finding good assembly programmers has also become difficult nowadays. Hence high level languages are preferred for embedded systems programming.

DIFFERENCE BETWEEN C AND EMBEDDED C

Though **C and embedded C** appear different and are used in different contexts, they have more similarities than the differences. Most of the constructs are same; the difference lies in their applications.

C is used for desktop computers, while **embedded C** is for microcontroller based applications. Accordingly, C has the luxury to use resources of a desktop PC like memory, OS, etc. While programming on desktop systems, we need not bother about memory. However, embedded C has to use with the limited resources (RAM, ROM, I/Os) on an embedded processor. Thus, program code must fit into the available program memory. If code exceeds the limit, the system is likely to crash.

Compilers for C (ANSI C) typically generate OS dependant executables. **Embedded C** requires compilers to create files to be downloaded to the microcontrollers/microprocessors where it needs to run. Embedded compilers give access to all resources which is not provided in compilers for desktop computer applications.

Embedded systems often have the real-time constraints, which is usually not there with desktop computer applications. Embedded systems often do not have a console, which is available in case of desktop applications. So, what basically is different while programming with **embedded C** is the mindset; for embedded applications, we need to optimally use the resources, make the program code efficient, and satisfy real time constraints, if any. All this is done using the basic constructs, syntaxes, and function libraries of 'C'.

4.2.3 PROTEUS SOFTWARE

Proteus (**PRO**cessor for **TE**xt **E**asy to **US**e) is a fully functional, procedural programming language created in 1998 by Simone Zanella. Proteus incorporates many functions derived from several other languages: C, BASIC, Assembly, Clipper/dBase; it is especially versatile in dealing with strings, having hundreds of dedicated functions; this makes it one of the richest languages for text manipulation.

Proteus owes its name to a Greek god of the sea (Proteus), who took care of Neptune's crowd and gave responses; he was renowned for being able to transform himself, assuming different shapes. Transforming data from one form to another is the main usage of this language.

INTRODUCTION

Proteus was initially created as a multiplatform (DOS, Windows, Unix) system utility, to manipulate text and binary files and to create CGI scripts. The language was later focused on Windows, by adding hundreds of specialized functions for: network and serial communication, database interrogation, system service creation, console applications, keyboard emulation, ISAPI scripting (for IIS). Most of these additional functions are only available in the Windows flavor of the interpreter, even though a Linux version is still available.

Proteus was designed to be practical (easy to use, efficient, complete), readable and consistent.

- Its strongest points are:
- powerful string manipulation;
- comprehensibility of Proteus scripts;
- availability of advanced data structures: arrays, queues (single or double), stacks, bit maps, sets, AVL trees.
- The language can be extended by adding user functions written in Proteus or DLLs created in C/C++.

4.2.3.1 LANGUAGE FEATURES

At first sight, Proteus may appear similar to Basic because of its straight syntax, but similarities are limited to the surface:

- Proteus has a fully functional, procedural approach;
- variables are untyped, do not need to be declared, can be local or public and can be passed by value or by reference;
- all the typical control structures are available (if-then-else; for-next; while-loop; repeat-until; switch-case);
- new functions can be defined and used as native functions.

SYNOPSIS AND LICENSING

The main features of this language are:

- fully functional, procedural language;
- multi-language support: Proteus is available in several languages (keywords and messages);
- no data types: all variables can be used as integer numbers, floating point numbers or strings; variables are interpreted according to the functions being applied – Proteus keeps different representations of their values between calls, to decrease execution time in case of frequent conversions between one type and the other;
- no pre-allocated structures: all data used by Proteus are dynamically allocated at execution time; there are no limits on: recursion, maximum data size, number of variables, etc.;
- no operators: Proteus is a completely functional language – there are no operators; thus, there is no ambiguity when evaluating expressions and parenthesis are not needed;
- large library of predefined functions: Proteus is not a toy-language, it comes with hundreds of library functions ready to be used for working on strings, dates, numbers, for sorting, searching and so on;
- advanced data access (DAO), pipes, Windows sockets, serial ports: in the Windows version, Proteus includes hundreds of system calls which are operating system-specific;
- clear and comprehensible syntax: the names of the library functions resamble those of corresponding functions in C, Clipper/Flagship and Assembly; by using medium-length keywords, Proteus programs are very easy to understand;
- native support for high-level data structures: arrays, queues (single or double), stacks, bit maps, sets, AVL trees are already available in Proteus and do not require additional code or libraries to be used;

EXAMPLE PROGRAMS

Hello World

The following example prints out "Hello world!".

```
CONSOLELN "Hello World!"
```

Extract two fields

The following example reads the standard input (CSV format, separator ";") and prints out the first two fields separated by "|":

```
CONSOLELN TOKEN(L, 1, ";") "|" TOKEN(L, 2, ";")
```

Proteus scripts by default work on an input file and write to an output file; the predefined identifier L gets the value of every line in input. The function TOKEN returns the requested item of the string; the third parameter represents the delimiter. String concatenation is implicit.

CHAPTER 5
CONCLUSION AND FUTURE SCOPE

CHAPTER 5

5.1 CONCLUSION

Our fingerprint-based secured voting mechanism, it safe to say that, this system has managed to overcome most of the problems faced during the voting period by EVM system. The efficiency of the system depends on the User Interface design and the flexibility that it provides as well as the usability for it. This ensures a safer voting method which is totally required for the healthy growth of a developing nation. In this paper, the proposed online voting system using biometrics that is the fingerprint scanner is better and faster than the previous system. The online voting system using a fingerprint scanner will provide a chance to avoid invalid votes. In this system, only an authenticated and registered person will be able to vote. As a challenging field in the area of biometrics, fingerprint analysis is one of the emerging techniques used for verification and identification of an individual. Automatic minutiae extraction is an awfully decisive process. The fingerprint verification stage works by comparing two fingerprints and recognizes if they belong to the same person. Though several approaches have been proposed in the literature so far, each has its own strengths and weaknesses. This project focuses on dealing with most of them and provides a better solution.

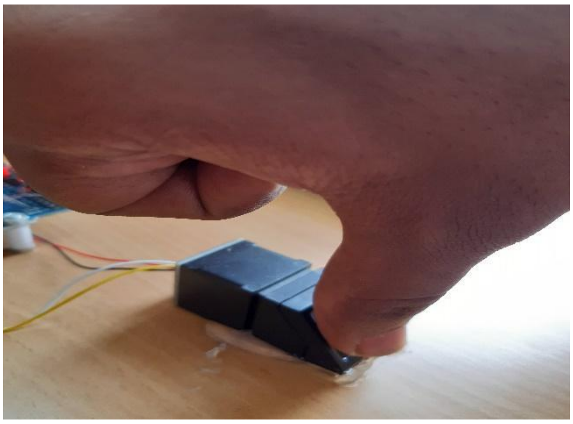
5.2 FUTURE SCOPE

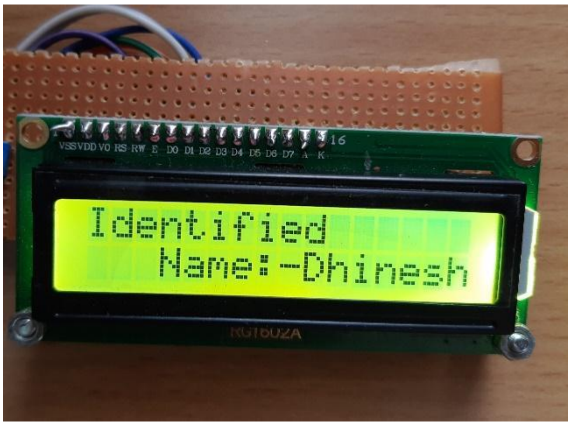
Based on the work presented in this thesis, there are several there are various advanced features that we can add to the system. Feature combinations in multimodal biometric can be worked upon by various combinations of features. The proposed method can be extended further using other biometrics such as irises, DNA and gait. We could work on devising an algorithm which can predict the minutiae points, gender and the precise age of the person to whom the fingerprint belongs. When a fingerprint is given as input, it should display the minutiae points, the gender and the precise age in the output GUI. The proposed algorithm for the minutiae identification, gender classification and age classification could be tried on noisy fingerprint images. The algorithms which are used in this thesis could be tried out on the fingerprints collected from the crime scene. Thus it can help in catching the criminals fast and easily. The relationship between fingerprints and gaits can be examined in the near future. Fuzzy Logic and Genetic are some of the renowned soft computing techniques which can be used here. Programming can be incorporated with Neural Network and SVM to obtain more precise output. To increase the performance, this fingerprint biometrics can be combined with other biometric techniques near the future.

APPENDIX

USER ACCESS OUTPUT:

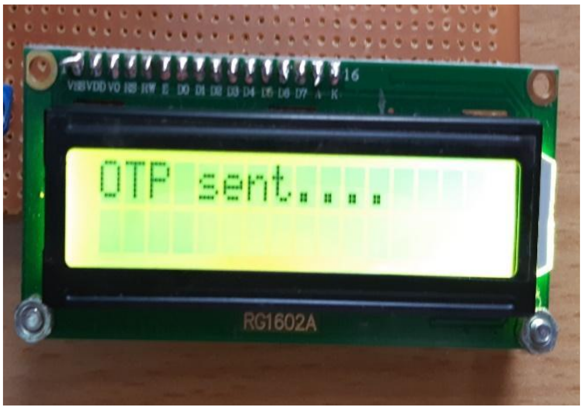
a) Initial start **b) place finger**

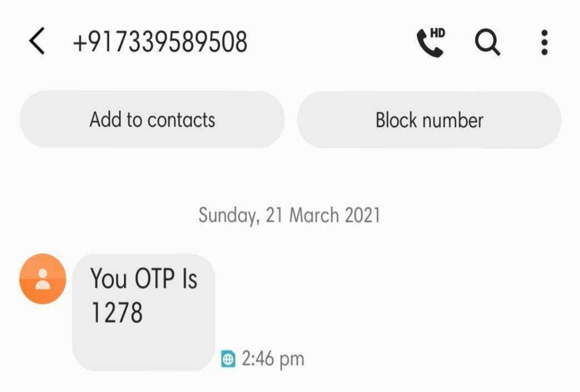




c) Finger placed

d) Person Identified





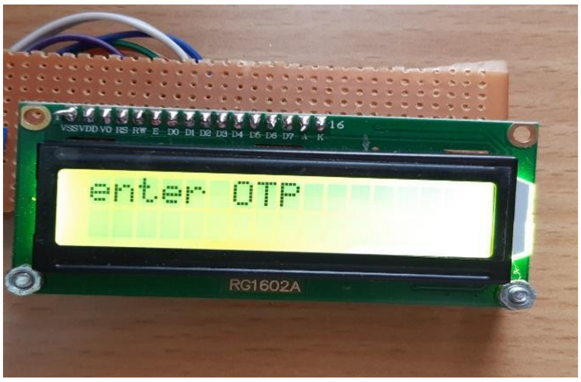
Sunday, 21 March 2021

You OTP Is
1278

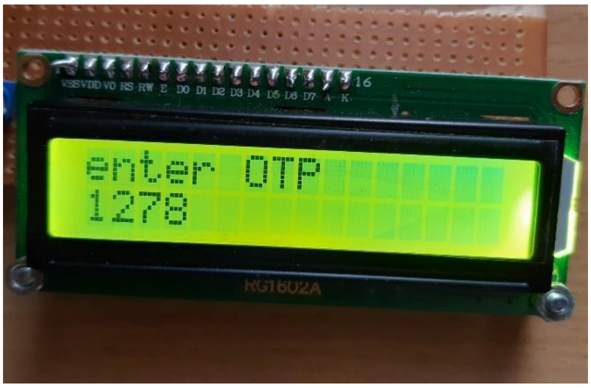
2:46 pm

e) OTP Sent

f) OTP Received



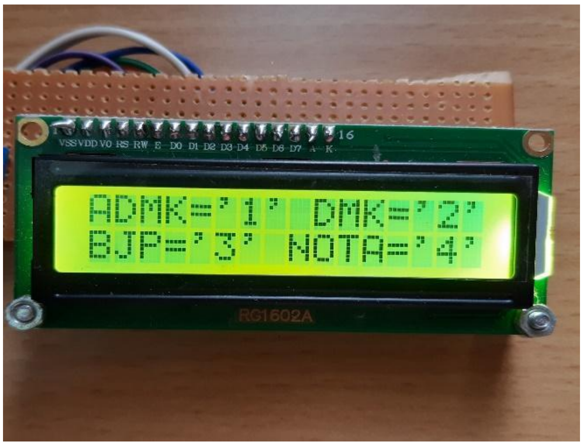
```
enter OTP
```



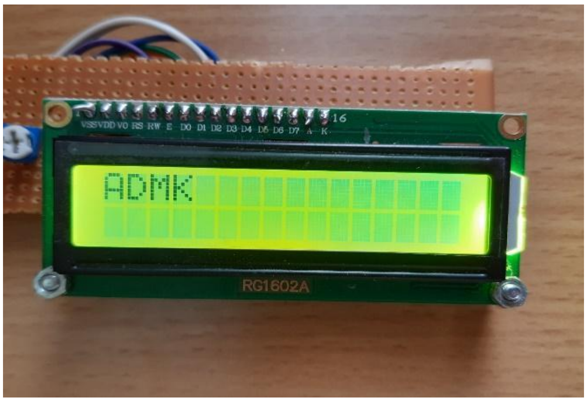
```
enter OTP
1278
```

g) Enter OTP

h) OTP entered



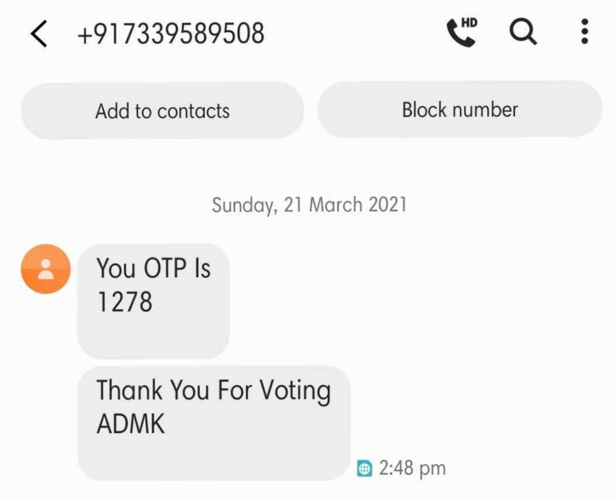
```
ADMK='1' DMK='2'
BJP='3' NOTA='4'
```



```
ADMK
```

i) select party to vote

j) Voted



+917339589508

Sunday, 21 March 2021

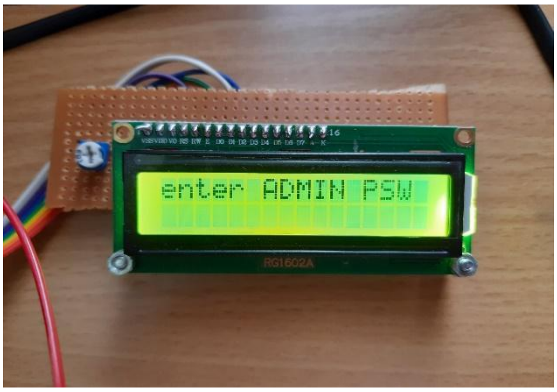
You OTP Is 1278

Thank You For Voting ADMK

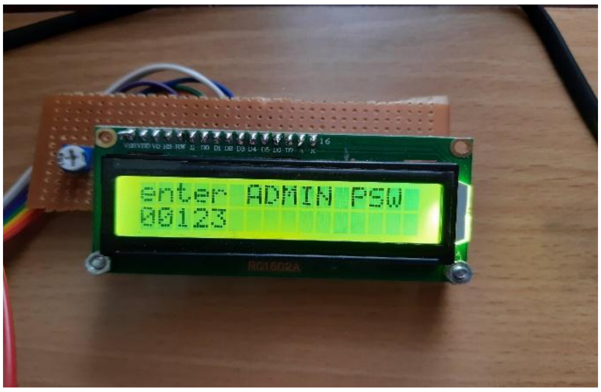
2:48 pm

k) Message Received

ADMIN ACCESS OUTPUT:



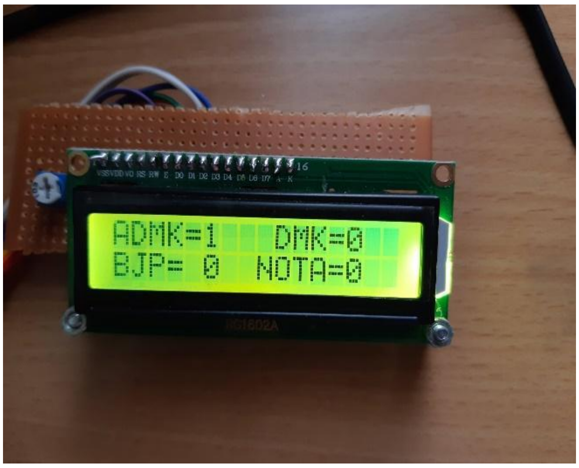
a) Enter Admin Password



b) Password entered

```
Voting Updation
 Loading....
```

c) Updation



```
ADMK=1   DMK=0
BJP= 0  NOTA=0
```

d) Vote Counted

USER ACCESS DESCRIPTION:

a) LCD display shows the welcome message of the project, 16*2 LCD was used here with 4 data pin connection to arduino.

b) After displaying the welcome message, our system will enter into the next loop of the program which waiting for the fingerprint input.

c) Once the fingerprint sensor will Detects the authorized finger matched with the pre saved dataset.

d) Matched finger will authenticate the user to vote.

e) OTP sent to the authorized person which fingerprint matched.

f) Message received to mobile number which registered in the voter ID or Adhar card.

g) received OTP need to enter in the device with use of keypad

h) if entered OTP matched with the received OTP in mobile, our system will grant access to vote, else buzzer will turn ON and vote denied.

i) Number of election parties are displayed in the LCD.

j) with the use keypad we can select the candidate by our choice and voted.

k) Message received to mobile for confirmation.

USER ACCESS DESCRIPTION:

a) System will ask for admin password to approve the access with fingerprint biometric.

b) password needs to confirm the admin user for vote counting access.

c) Takes some time for counting process.

e) Finally counted votes will be shown in the display for counting.

CODING

```
#include<LiquidCrystal.h>
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);
#define c1 A0
#define c2 A1
#define c3 A2
#define r1 A3
#define r2 7
#define r3 6
#define r4 5
```

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);
SoftwareSerial mySerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
int fingerprintID = 0;
int buzzer()
{
  digitalWrite(4, 1);
  delay(200);
  digitalWrite(4, 0);
  delay(200);
  digitalWrite(4, 1);
  delay(200);
  digitalWrite(4, 0);
  delay(200);
  digitalWrite(4, 1);
  delay(200);
  digitalWrite(4, 0);
  delay(200);
}
char keypad()
{
  while (1)
  {
    digitalWrite(c1, 1);
    digitalWrite(c2, 1);
    digitalWrite(c3, 1);
    digitalWrite(r1, 1);
    digitalWrite(r2, 1);
    digitalWrite(r3, 1);
    digitalWrite(r4, 1);
    digitalWrite(r1, 0);
    if (digitalRead(c1) == 0)
    {
      while (digitalRead(c1) == 0);
      lcd.print('1');
      return ('1');
    }
    if (digitalRead(c2) == 0)
    {
      while (digitalRead(c2) == 0);
      lcd.print('2');
      return ('2');
    }

    if (digitalRead(c3) == 0)
    {
      while (digitalRead(c3) == 0);
      lcd.print('#');
      return ('#');
    }
void setup() {
  lcd.begin(16, 2);
  pinMode(c1, INPUT);
  pinMode(c2, INPUT);
```

```
 pinMode(c3, INPUT);
 pinMode(r1, OUTPUT);
 pinMode(r2, OUTPUT);
 pinMode(r3, OUTPUT);
 pinMode(r4, OUTPUT);
 pinMode(4, OUTPUT);
 Serial.begin(9600);
 finger.begin(57600);
 Wire.begin();
 display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
}
 void loop() {
 digitalWrite(4, 0);
 int otp;
 lcd.clear();
 lcd.setCursor(3, 0);
 lcd.print("welcome to");
 lcd.setCursor(0, 1);
 lcd.print("Voting machine");
 delay(3000);
 int check1 = 0;
 int check2 = 0;
 int check3 = 0;  // sms sendding
 int i;
 while (1)
 {
       lcd.setCursor(3, 1);
       lcd.print("Name:-Dinagar");
       delay(2000);
```

```
    p1 = 10;    //////////////
    finger.fingerID = 50;
  }
}
if (p2 == 0)
{
  if (finger.fingerID == 60)                              // person 2
  {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Identified");
    lcd.setCursor(3, 1);
    lcd.print("Name:-Dhinesh");
    delay(2000);
    finger.fingerID = 50;
  }
}
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("enter ADMIN PSW");
  char p0[10] = {"00123"}, g0[10];
    lcd.print("Loading....");
    delay(3000);
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("ADMK=    DMK=   ");
    lcd.setCursor(0, 1);
    lcd.print("BJP=    NOTA=   ");
    lcd.setCursor(5, 0);
```

```
   lcd.print(ADMK);
  else
  {
   lcd.setCursor(0, 0);
   lcd.print("try again");
   delay(2500);
   digitalWrite(4, 0);
   delay(500);
   goto here;
  }
 }
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("OTP sent....");
  delay(3000);
  while (1)
  {
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("enter OTP");
   char p0[10] = {"2255"}, g0[10];
   int c0;
   pas(g0);
   c0 = strcmp(p0, g0);
   if (c0 == 0)
   {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("update your vote");
```

```
delay(2000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("ADMK='1' DMK='2'");
lcd.setCursor(0, 1);
lcd.print("BJP='3' NOTA='4'");
delay(3000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Select and-Tap #");
lcd.setCursor(0, 1);
p1 = 1111;
delay(10);
c3 = strcmp(p3, g);
  delay(100);
  Serial.println((char)26);
  delay(100);
  delay(1000);
  buzzer();
  goto top;
}
else if (c1 == 0)
{
  Serial.println("AT+CMGF=1");
  delay(100);
  Serial.println("AT+CMGS=\"+917708284135\"\r");
  delay(100);
  Serial.println("Thank You For voting");
  delay(100);
```

```
      Serial.println("BJP");
      delay(100);
      Serial.println((char)26);
    else if (c3 == 0)
    {
      lcd.clear();
      lcd.print("NOTA");
      NOTA = NOTA + 1;
      delay(1000);
      Serial.println("AT");
      delay(100);
      Serial.println("AT+CMGF=1");
      delay(100);
      Serial.println("AT+CMGS=\"+917708284135\"\r");
      delay(100);
      Serial.println("Thank You For voting");
      delay(100);
      Serial.println("NOTA");
      delay(100);
      Serial.println((char)26);
      delay(100);
      buzzer();
      goto top;
    }
else if (p2 == 11)                    //// for second
{
  Serial.println("AT");
  delay(100);
  Serial.println("AT+CMGF=1");
```

```
delay(100);
Serial.println("AT+CMGS=\"+917010018221\"\r");
delay(100);
Serial.println("You OTP Is");
delay(100);
Serial.println("1278");
//otp = 2255;
delay(100);
Serial.println((char)26);
delay(100);


lcd.clear();
lcd.setCursor(0, 0);
lcd.print("OTP sent....");
delay(3000);
   else if (c1 == 0)
   {
     lcd.clear();
     lcd.print("DMK");
     DMK = DMK + 1;
     Serial.println("AT");
     delay(100);
     Serial.println("AT+CMGF=1");
     delay(100);
     Serial.println("AT+CMGS=\"+917010018221\"\r");
     delay(100);
     Serial.println("Thank You For Voting");
     delay(100);
     buzzer();
```

```
        delay(100);
        delay(1000);
        buzzer();
        goto top;
      }
    }
    else
    {
      digitalWrite(4, 1);
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("wrong OTP");
      lcd.setCursor(0, 1);
      lcd.print("try agin");
      delay(2000);
      digitalWrite(4, 0);
    }
else if (p3 == 12)                    //// third
{
  Serial.println("AT");
  delay(100);
  Serial.println("AT+CMGF=1");
  delay(100);
  Serial.println("AT+CMGS=\"+916380310171\"\r");
  delay(100);
  Serial.println("You OTP Is");
  delay(100);
  Serial.println("4589");
  //otp = 2255;
  delay(100);
  Serial.println((char)26);
  delay(100);
     if (c == 0)
     {
       lcd.clear();
       lcd.print("ADMK");
       ADMK = ADMK + 1;
       Serial.println("AT");
       delay(100);
       Serial.println("AT+CMGF=1");
       delay(100);
       Serial.println("AT+CMGS=\"+916380310171\"\r");
       delay(100);
       Serial.println("Thank You For Voting");
       delay(100);
       Serial.println("ADMK");
       delay(100);
       Serial.println((char)26);
       delay(100);
       delay(1000);
       buzzer();
       goto top;
     }
else if (p4 == 13)                        //// for 4th
{
  Serial.println("AT");
  delay(100);
  Serial.println("AT+CMGF=1");
```

```
        delay(100);
        Serial.println("AT+CMGS=\"+919500922588\"\r");
        delay(100);
        Serial.println("You OTP Is");
        delay(100);
        Serial.println("2356");
          char p[10] = {"1"}, g[10];
          char p1[10] = {"2"};
          char p2[10] = {"3"};
          char p3[10] = {"4"}; // all value mentioned
          int c, c1, c2, c3;
          pas(g);
          else if (c2 == 0)
          {
           lcd.clear();
           lcd.print("BJP");
           BJP = BJP + 1;
           Serial.println("AT");
           delay(100);
           Serial.println("AT+CMGF=1");
           delay(100);
           Serial.println("AT+CMGS=\"+919500922588\"\r");
           delay(100);
           Serial.println("Thank You For Voting");
           delay(100);
{
 uint8_t p = finger.getImage();
 if (p != FINGERPRINT_OK)  return -1;
 p = finger.image2Tz();
```

```
  if (p != FINGERPRINT_OK)  return -1;
  p = finger.fingerFastSearch();
  if (p != FINGERPRINT_OK)  return -1;
  return finger.fingerID;
}
```

REFERENCE

[1] V. Kiruthika Priya , V. Vimaladevi , B. Pandimeenal , T. Dhivya, “Arduino based smart electronic voting machine”, 2017 International Conference on Trends in Electronics and Informatics (ICEI) Year: 2017, conference Paper, Publisher: IEEE.

[2] Rahil Rezwan, Huzaifa Ahmed, M. R. N. Biplo, S. M. Shuvo, Md. Abdur Rahman, “Biometrically secured electronic voting machine”, 2017 IEEE Region 10 Humanitarian Technology Conference (R10- HTC).

[3] Prof. Sunita Patil, Amish Bansal, Utkarsha Raina, Vaibhavi Pujari, Raushan Kumar, “E-Smart Voting Machine with Secure Data Identification Using Cryptography”, 2018 Publisher: IEEE

[4] Annalisa Franco, “Fingerprint: Technologies and Algorithms for Biometrics Applications”, Year: 2011 , Course , Publisher: IEEE.

[5] A. Piratheepan, S. Sasikaran, P. Thanushkanth, S. Tharsika, M. Nathiya, C. Sivakaran, N. Thiruchchelvan and K. Thiruthanigesan, “Fingerprint Voting System Using Arduino”, College of Technology Jaffna, Sri Lanka University College of Anuradhapura, University of Vocational Technology, Sri Lanka.

[6] Rohan Patel, Vaibhav Ghorpade, Vinay Jain and Mansi Kambli, “Fingerprint Based e-Voting System using Aadhar Database”, 2015.

[7] S Wolchok, E Wustrow, JA Halderman. “Security analysis of India's electronic voting machines” 2010.

[8] Qijun Zhao, Lei Zhang, David Zhang and Nan Luo, “Adaptive Pore Model for Fingerprint Pore Extraction”, IEEE, 978-1-4244-2175 - 6/08.

[9] Md. Mahboob Karim, Nabila Shahnaz Khan, Ashratuz Zavin, Shusmoy Kundu, Asibul Islam, Brazab Nayak, “A proposed framework for biometric electronic voting system”, IEEE International conference on 2017

[10] Soumyajit Chakraborty, Siddhartha Mukherjee, Bhaswati Sadhukhan, Kazi Tanvi Yasmin,“Biometric Voting System using Adhar Card in India” 2016

[11] https://learn.adafruit.com/adafruit-all-about-arduino-libraries-installuse/arduino-libraries.

[12]A. K.Agarwala, D. T. Shahani, and P. V. Indiresan. Report of the expert committee for evaluation of the upgraded electronic voting machine (EVM). Sept. 2006.

[13] Khasawneh, M., Malkawi, M., & Al-Jarrah, O. (2008). A Biometric-Secure e-Voting System for Election Process. Proceeding of the 5th International Symposium on Mechatronics and its Applications (ISMA08). Amman, Jordan.

[14] Prasad, H. K., Halderman, A. J., & Gonggrijp, R. (Oct. 2010). Security Analysis of India’s Electronic Voting