POLITEKNIK SULTAN SALAHUDDIN ABDUL AZIZ SHAH

BIOMETRIC FINGERPRINT SENSOR TO LOCK AND UNLOCK DOOR

NAME	REGISTRATION NO
SHANKARI NAIR A/P MURALI	**08DEU20F2004**

JABATAN KEJURUTERAAN ELEKTRIK

SESI 2 2022/2023

POLITEKNIK

SULTAN SALAHUDDIN ABDUL AZIZ SHAH

BIOMETRIC FINGERPRINT SENSOR TO LOCK AND UNLOCK DOOR

NAME	REGISTRATION NO
SHANKARI NAIR A/P MURALI	08DEU20F2004

This report submitted to the Electrical Engineering Department in fulfillment of the requirement for a Diploma in Electrical Engineering

JABATAN KEJURUTERAAN ELEKTRIK

SESI 2 2022/2023

CONFIRMATION OF THE PROJECT

The project report titled "Biometric Fingerprint Sensor to Lock and Unlock Door" has been submitted, reviewed and verified as a fulfills the conditions and requirements of the Project Writing as stipulated

Checked by:

Supervisor's name : PUAN NAAGAJOOTHI A/P ADIN NARAINA

Supervisor's signature:

Date :

Verified by:

Project Coordinator name :

Signature of Coordinator :

Date :

"I acknowledge this work is my own work except the excerpts I have already explained to our source"

1. Signature :

Name : **SHANKARI NAIR A/P MURALI**

Registration Number : **08DEU20F2004**

Date :

DECLARATION OF ORIGINALITY AND OWNERSHIP

TITLE : BIOMETRIC FINGERPRINT SENSOR TO LOCK AND UNLOCK DOOR

SESSION: SESI 2 2022/2023

1. I, **1. SHANKARI NAIR A/P MURALI (08DEU20F2004)**

   is a final year student of **Diploma in Electrical Engineering (MEDICAL), Department of Electrical, Politeknik Sultan Salahuddin Abdul Aziz Shah**, which is located at **Persiaran Usahawan,40150 Shah Alam Selangor Darul Ehsan**. (Hereinafter referred to as 'the Polytechnic').

2. I acknowledge that 'The Project above' and the intellectual property therein is the result of our original creation /creations without taking or impersonating any intellectual property from the other parties.

3. I agree to release the 'Project' intellectual property to 'The Polytechnics' to meet the requirements for awarding the **Diploma in Electrical Engineering** to me.

Made and in truth that is recognized by;

a) **SHANKARI NAIR A/P MURALI**) . ....…….
(Identification card No: - 020902011744)) **SHANKARI NAIR A/P MURALI**

In front of me, **PUAN NAAGAJOOTHI A/P ADIN NARAINA**) .....……
As a project supervisor, on the date:) **NAAGAJOOTHI A/P ADIN NARAINA**

ACKNOWLEDGEMENTS

I put effort on this project. However, without the kind support and assistance of many people and organizations, it would not have been feasible. I want to express my sincere gratitude to each one of them. I owe a great deal to MADAM NAAGAJOOTHI A/P ADIN NARAINA for their leadership, continual oversight, and provision of the information required for the Project as well as their assistance in seeing it through to completion.

I want to thank my parents and the members of the electrical engineering department for their kind encouragement and cooperation in helping me finish my project. I want to offer my sincere appreciation and thanks to those in the profession who took the time to pay me such close attention.

My gratitude and appreciations also go out to my colleague who helped me developing the Project and those who volunteered their skills to assist me.

ABSTRACT

This project is to design and develop a “THE BIOMETRIC FINGERPRINT SENSOR TO LOCK AND UNLOCK THE DOOR”. In our everyday lives, security is a top priority, and digital locks have emerged as a key component of these security systems. One of the safest doors locking systems is fingerprint sensor-based because it can accurately identify and distinguish everyone without error. The fingerprint-based security system can be employed in a variety of settings, including workplaces, educational institutions, and even our homes. Additionally, having a set of keys has become a hassle because modern homeowners should try to live a minimum, key less lifestyle. Fingerprint door locks use tried-and true technology. The most developed and tried form of biometric technology is fingerprint reader scanning. Recent biometric research has demonstrated that the fingerprint method is more precise and economical than the hand method. This fingerprint-based door lock system's circuit is made up of an Arduino Mega2560, which manages every aspect of the project, as well as a push button, buzzer, and LCD. The entire process is managed by an Arduino Mega2560. The push button is connected directly to the Arduino Mega2560's pins A9 (ENROL), A10 (OK/DEL), A11 (UP), A12 (DOWN), and A8 (CLOSED) with respect to ground. A 10-ohm resistor is used to connect the red and green LEDs to the Arduino Mega2560's digital pin D4 and D3, respectively, with respect to ground. Rx and Tx of the Fingerprint Module are directly linked to Arduino Mega2560's Software Serial or Digital pins D11 and D10. The fingerprint module is powered by a 5-volt supply and attached to an Arduino Mega pin. A servo motor is also connected to Arduino Mega2560's PWM pin D5. A 16x2 LCD is set up in 4-bit mode, and the RS, EN, D4, D5, D6, and D7 of the Arduino Mega2560 are directly linked to those pins. The buzzer is linked to the Arduino Mega2560's Digital pin D14 with regard to the ground. DHT is attached to the Arduino Mega2560's Digital Pin D2 and to ground and Vcc. LCD is connected to the center pin's potentiometer (Vo).

ABSTRAK

Projek ini adalah untuk mereka bentuk dan membangunkan "PENDERIA CAP JARI BIOMETRI UNTUK MENGUNCI DAN MEMBUKA PINTU". Dalam kehidupan seharian kita, keselamatan adalah keutamaan utama dan kunci digital telah muncul sebagai komponen utama sistem keselamatan ini. Salah satu sistem penguncian pintu paling selamat ialah berasaskan penderia cap jari kerana ia boleh mengenal pasti dan membezakan semua orang dengan tepat tanpa kesilapan. Sistem keselamatan berasaskan cap jari boleh digunakan dalam pelbagai tetapan, termasuk tempat kerja, institusi pendidikan, dan juga rumah kita. Selain itu, mempunyai satu set kunci telah menjadi kerumitan kerana pemilik rumah moden harus cuba menjalani gaya hidup minimum tanpa kunci. Kunci pintu cap jari menggunakan teknologi yang telah dicuba dan benar. Bentuk teknologi biometrik yang paling maju dan dicuba ialah pengimbasan pembaca cap jari. Penyelidikan biometrik terkini telah menunjukkan bahawa kaedah cap jari adalah lebih tepat dan menjimatkan daripada kaedah tangan. Litar sistem kunci pintu berasaskan cap jari ini terdiri daripada Arduino Mega2560, yang menguruskan setiap aspek projek, serta butang tekan, buzzer dan LCD. Keseluruhan proses diuruskan oleh Arduino Mega2560. Butang tekan disambungkan terus ke pin Arduino Mega2560 A9 (ENROL), A10 (OK/DEL), A11 (ATAS), A12 (BAWAH), dan A8 (TUTUP) berkenaan dengan tanah. Perintang 10-ohm digunakan untuk menyambungkan LED merah dan hijau ke pin digital Arduino Mega2560 D4 dan D3, masing-masing, berkenaan dengan tanah. Rx dan Tx Modul Cap Jari dipautkan terus ke Siri Perisian Arduino Mega2560 atau pin Digital D11 dan D10. Modul cap jari dikuasakan oleh bekalan 5 volt dan dipasang pada pin Arduino Mega. Motor servo juga disambungkan ke pin PWM D5 Arduino Mega2560. LCD 16x2 disediakan dalam mod 4-bit, dan RS, EN, D4, D5, D6, dan D7 Arduino Mega2560 dipautkan terus ke pin tersebut. Buzzer dipautkan ke pin Digital Arduino Mega2560 D14 berkenaan dengan tanah. DHT dipasang pada Pin Digital Arduino Mega2560 D2 dan ke tanah dan Vcc. LCD disambungkan ke potensiometer pin tengah (Vo).

TABLE OF CONTENTS

CHAPTER 1

1 INTRODUCTION

1.1 Introduction

The suggested described project focuses on a door lock security system based on the biometric fingerprint that may be installed as a smart door lock in any home. First of all, carrying a key for a door lock has gotten heavier with time. Additionally, lock breaking and the theft of expensive items from within a home are now very common at night and when the owner is away from home. Even today, most homes only have a lock and key security system because there are several opportunities for a key to be lost or duplicate keys to the same lock to be made. As a result, this system will safeguard people's lives and the priceless possessions they keep in their homes from thieves and alert the owner via email if the smart door lock is being tampered with. Without keys and key duplication problems, this system will make life easier for users by preventing them from having to carry around a set of keys while still guaranteeing the security of their homes.

1.2 Background Research

The biometric performance evaluation enables the developer to quickly understand the state of the available technology and expedite the commercialization process. Additionally, this creates a large market and shows users how reliable the product is. Because fingerprint recognition is the most distinctive biometric technology in the sense that many types of fingerprint sensors are being marketed, algorithm developers as well as users require objective and quantitative evaluation techniques. The goal of this study is to suggest a performance evaluation procedure, evaluation metrics, and result presentation for fingerprint sensor modules. By contrasting the quantitative measure of fingerprint image quality impacted by environmental elements (temperature and humidity) and user characteristics, four commercial fingerprint sensor modules have been tested and appraised (skin humidity and pressure). The outcome of this study is anticipated to be used to improve algorithms that adapt to various fingerprint sensor modules.

1.3 Problem Statement

Due to the new technology that is being created every day to elevate people's standards of living, there is a major need for solutions that can make daily life for users simpler and improve home security. To be more specific, there are a number of drawbacks to the current manual door lock that bother a substantial section of the populace who haven't yet chosen to implement a secure system by using fingerprints to lock and unlock doors for their home. To make their daily jobs easier, people commonly rely on electrical and technological devices. The majority of door locks have drawbacks and have begun to irritate users. Start with the first major present issue, which is that the house lock pads frequently have structural issues. Mechanical lock pad failure, internal component damage, or even lock pads that corrode with time are examples of significant defects. In addition, a lot of people lose their keys, making it impossible for them to access their own places. This problem is made worse when people want services from others in respected settings, which ultimately results in absolute frustration for everyone.

1.4 Research Objectives

The main objectives are:

- To design the current fingerprint censoring method more dependable and accessible to people of all ages and demographics.
- To implement the outcomes in order to determine the accuracy and effectiveness of the fingerprint system.
- To develop a censoring system for the entrance in order to develop a system with superior security.

1.5 Scope of Research

The aim of this project is to create a basic, key less lifestyle and improve home security. It also includes the system act as a manual.

1.6 Contribution of the project

As the world is moving forward with Industrial Revolution 4.0, it is a necessary to have technology in innovation and invention to boost economic growth.It is important to have a fingerprint sensor since it provides high security to the current door lock system. The importance of this biometric fingerprint door lock system is it only consumes less power, and it can storage up to 200 fingerprints. This fingerprint security system can also be used in ATM, and also fingerprint operated vehicles.

1.7 Chapter Summary

An introduction to the project, which uses a biometric fingerprint sensor to lock and unlock, can be found in this chapter. I then create the problem statement, project objective, and project scope for this project after conducting background research.

CHAPTER 2

2 LITERATURE REVIEW

2.1 Introduction

Engineering students and professionals today routinely research the project. This is as a result of their increased awareness and interest in the system. For them, the success of this project is essential to the development of technology and the future easement of human life.

2.2 Table of 5 Literature Review

NO	TITLE/AUTHOR	OBJECTIVE	METHOD	RESULT
1.	➢ **Smart door lock using fingerprint sensor** ➢Piash paul, Md.Abdullah Al Achib, Hazrat Sauda Hossain, Md. Kaiviul Hossain	To provide adjustable security level setting account for variations.	This project includes Atmel Atmega328p, a fingerprint sensor, gsm module, motor driver, a motor, microcontroller and some other hardware devices.	The system will also go into a secure state where it will continue to buzz the buzzer to alert the neighbor that something is wrong. The system will be reset once a known print will be entered.
2.	➢**Smart door unlock system using fingerprint** ➢ K.Rajesh#1,ASST. PROFESSER,B. Venkata Rao#2,P.AV.S.K.Chaitanya #3,A.ruchitha Reddy#4	To create an advance system that will allow the user to save and delete a fingerprint into the system.	Fingerprint sensor can be interfaced with a microcontroller. Result is displayed in a LCD display	Through keypad we can add new user and delete the existing user, also identify the user by selecting corresponding option through

			whether the user is authorized or not. LCD also helps to make troubleshooting easier. Alarming option is provided to warn about an unauthorized usage. Microcontroller used is PIC16F877.	keypad. In this project we use a fingerprint sensor to read one identity to automatically operate the door of the car. For this, we use a microcontroller to enable the door opening or closing if the matching between scanned data and the already existing data is correct.
3.	➢ **Smart door lock** ➢ Aleksander IBRO, Augusto WONG, Maria ZYLA	To provide students basic knowledge on how the fingerprint reader used in the industry.	The project has three main components: a Raspberry Pi, a cloud backend, and a mobile application. The Raspberry Pi is attached to the door and is responsible for controlling a servo motor, a camera, and an actuator.	Users can open the door by either tapping a button on the mobile app, or just by approaching the door. When a user is within a certain radius, a PIR sensor detects them, and activates the camera. The Raspberry Pi then sends an image of the user at the door to the backend server.
4.	➢ **DOOR ACCESS SYSTEM ARDUINO BASED** ➢ Puan Shuhada Natasha binti Mohd Zainor, 2012	- To create a door access system using Arduino. - To test the accuracy of this	Arduino is a type of microcontroller that uses its own	Interfacing all inputs and outputs with Arduino produced

		device.	programming language. It has its own electronic prototyping platform to be used during experiment. This project has two features as inputs, which are keypad and fingerprint scanner. While the outputs are magnetic switch, indicator, LCD display and a siren.	accurate results in accessing the door based on the accuracy results obtained by performing a test on five individuals with four different fingerprints each person. The entered password using keypad and minutiae obtained by scanning fingerprint are clarified as inputs, while the output is the result displayed on LCD display.
5.	➢ **FINGERPRINT BASED DOOR LOCK USING ARDUINO** ➢ Suyash Gaikwad *1, Sanjiv Jamadar*2, Rupali Shelke*3	To create a door access system using Arduino approach.	There are primarily four components which are used for the designing of our Fingerprint door lock using Arduino, and they are Arduino Nano, Fingerprint Sensor, Relay Module and Solenoid Lock.	This system is based for improving the security which will register the owner's fingerprint into the Arduino using the fingerprint sensor, and this system we have given 5v power supply to Arduino through the code uploading wire. When you put your thumb on fingerprint sensor after registering yourself the lock

				will be unlocked and you repeat this process again then the solenoid lock will be got locked. The process of locking and unlocking requires less than 1 second so this is why the Solenoid lock is used inside this project.

2.3 Chapter Summary

This chapter discusses the literature reviews of five journals that I discovered to be relevant to this study. One Atmel Atmega328p, one microcontroller, and the other Arduino were used from this five journals.

CHAPTER 3

3 RESEARCH METHODOLOGY

3.1 Introduction

A very thorough plan is being implemented in order to actualize this Project as a product that is ready to use with safety characteristics. To ensure that the Project is finished on time, a step-by-step process is followed. This involves gathering information from a sample fingerprint censoring locking system.

3.2 Project Design and Overview.

The circuit for this capacitive biometric fingerprint lock and unlock system is relatively straightforward and consists of an Arduino Mega2560, which manages every aspect of the project, as well as a push button, buzzer, and LCD. The entire process is managed by an Arduino Mega2560.

The push button is connected directly to the Arduino Mega2560's pins A9 (ENROL), A10 (OK/DEL), A11 (UP), A12 (DOWN), and A8 (CLOSED) with respect to ground. A 10ohm resistor is used to connect the red and green LEDs to the Arduino Mega2560's digital pin D4 and D3, respectively, with respect to ground. Rx and Tx of the Fingerprint Module are directly linked to Arduino Mega2560's Software Serial or Digital pins D11 and D10.

The finger print module is powered by a 5volt supply and attached to an Arduino Mega pin. A servo motor is also connected to Arduino Mega2560's PWM pin D5. A 16x2 LCD is set up in 4-bit mode, and the RS, EN, D4, D5, D6, and D7 of the Arduino Mega2560 are directly linked to those pins. The buzzer is connected to the Arduino Mega2560's Digital pin D14 with respect to the ground. DHT is attached to the Arduino Mega2560's Digital Pin D2 and to ground and Vcc. LCD is connected to the center pin's potentiometer (Vo).

In order to enroll the authenticating user's finger print, the "Enroll" button is first clicked. By clicking the "OK/Del" button, a fingerprint is saved. The "Close" button on the module is used to close the gate. When a user attempts to open the gate, the module verifies the user's identity by comparing the user's fingerprint with the database; if the fingerprint matches the one in the database, Arduino sends the signal to run the motor, which opens the gate and displays the messages "WELCOME" and "DOOR OPENED" on the LCD.

3.2.1 Block Diagram of the Project

Figure 3.1 shows the block diagram of the whole system. It is shown that

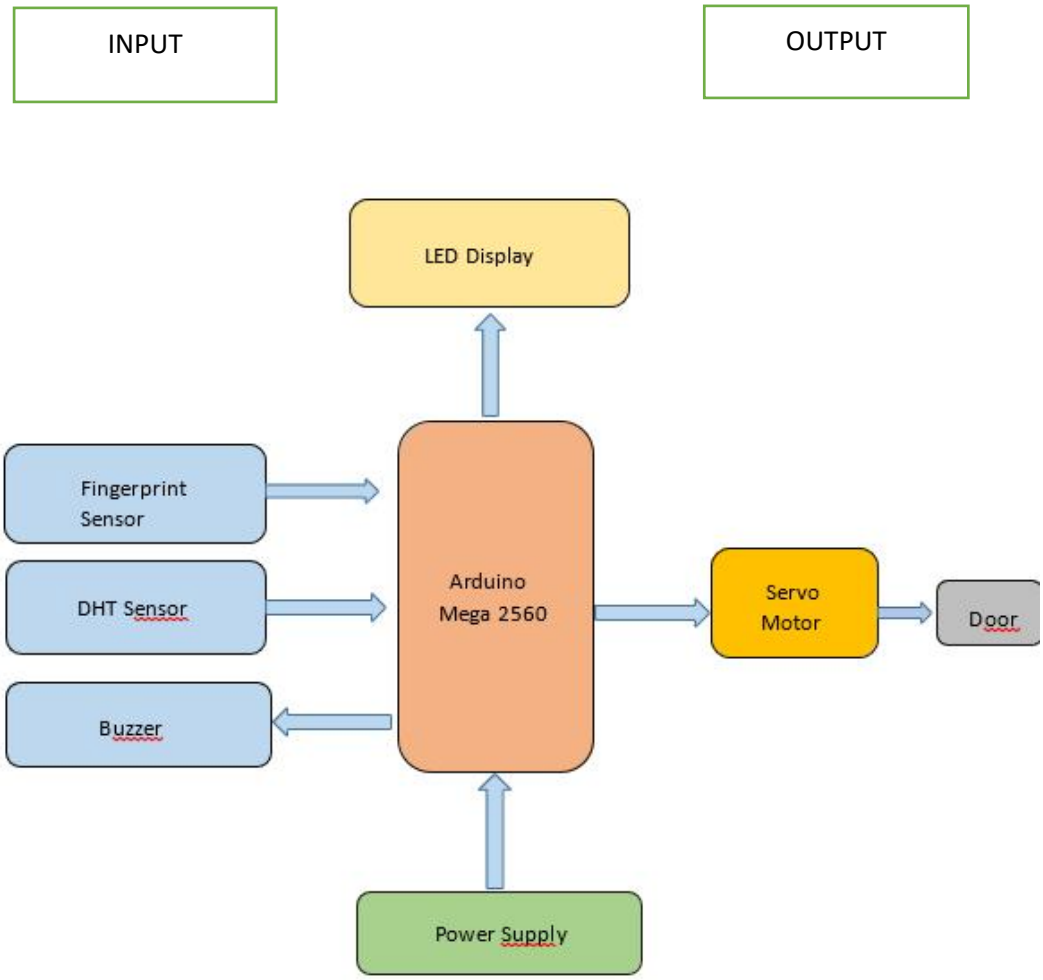
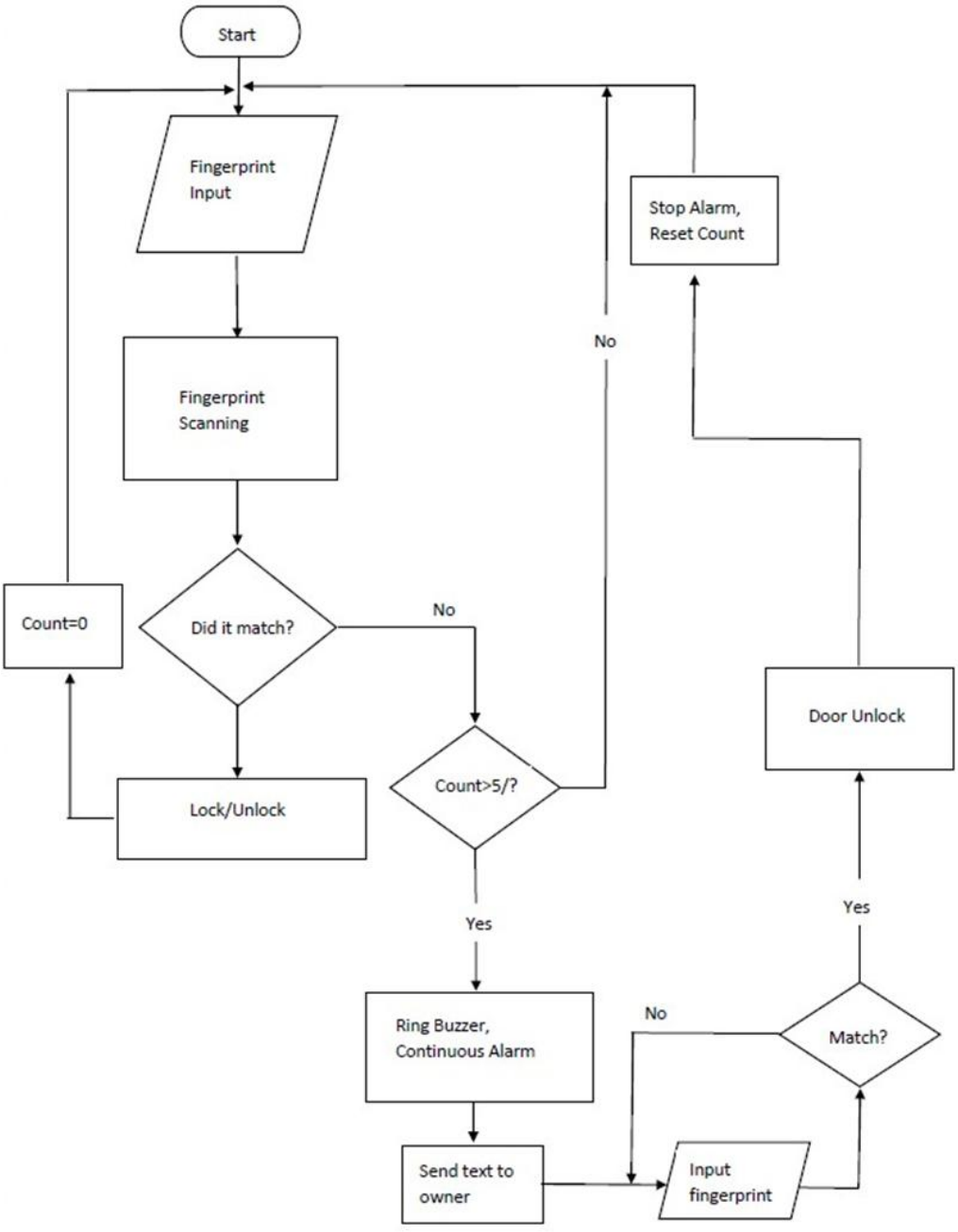


Figure 3.1: Block diagram of operation of the system

3.2.2 Flowchart of the Project 2

Figure 3.1 shows the circuit diagram of the whole system. It is show that



3.2.3 Project Description

The purpose of this system is to lock and unlock the door by using biometric fingerprint using capacitive sensor. The same person will be recognised using an Arduino Mega 2560 in a manner similar to how a fingerprint scanner function. It is an electronic gadget that takes digital pictures. A live scan is the name given to the acquired image. It merely recognises the fingerprint and reports the outcome. Then, fingers are examined when they are put against a smooth surface by fingerprint sensor devices. The ridges and valleys on the finger are scanned, and a collection of unique locations where the ridges and valleys meet or part are referred to as minutiae. These small details are what the fingerprint recognition system compares. Servo motors are used for locking and unlocking the door. LCD displays work by illuminating your finger with an LED and then using a sensor to record the 2D image data. An optical fingerprint sensor can use such data to assess whether there is a match by comparing it to subsequent picture data after it has been acquired. To address concerns about security and secrecy, a fingerprint-based door access control system is suggested.

3.3 Project Hardware

As mentioned in the chapter above, the controller was designed using an Arduino Mega 2560. This microcontroller recognises a fingerprint, reports the results, and manages push buttons, buzzers, LCDs, and the entire project's workflow. The entire process is managed by an Arduino Mega2560.

3.3.1 Schematic Circuit

Figure 3.2 shows the overall circuit diagram of this Project:

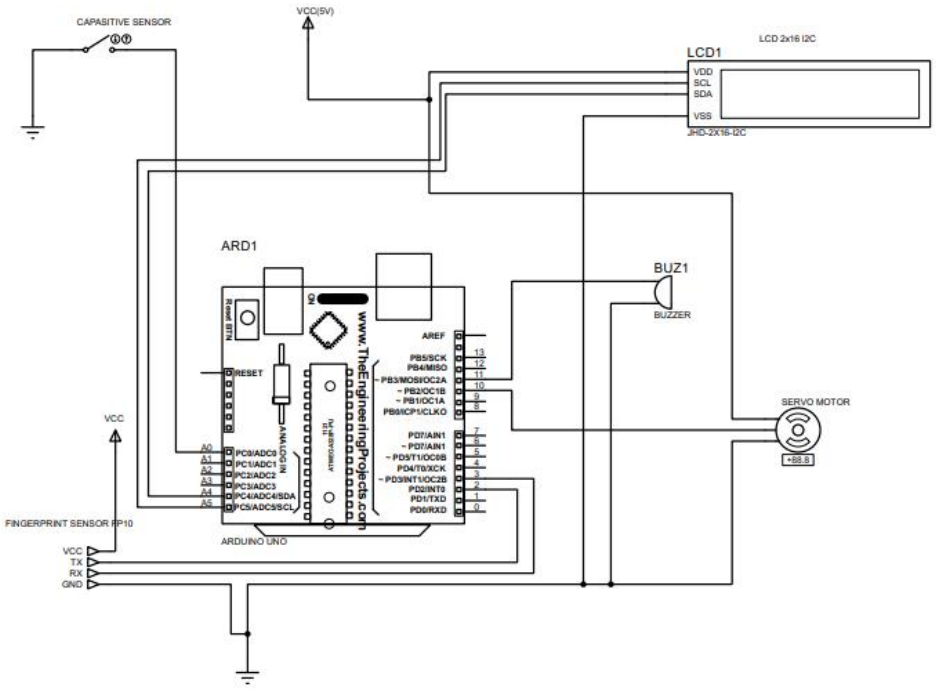


Figure 3.2: Circuit Diagram

3.3.2 Description of Main Component

ARDUINO MEGA 2560

A microcontroller board based on the ATmega2560 is called the Arduino Mega 2560. It contains everything required to support the microcontroller; just connect it to a computer with USB cable or power it with most shields designed for the Arduino Duemilanove or Diecimili. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analogue inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP HEADER, AND A RESWT BUTTON. Either an external power source or the USB connection can be used to power the Arduino Mega. The power source is automatically chosen.

3.3.2.1 Component 1

(SERVO MOTOR)

The angular or linear position, velocity, and acceleration can be precisely controlled with a servomotor, which is a rotary actuator or linear actuator. It consists of an appropriate motor connected to a position feedback sensor. It also needs a rather sophisticated controller, frequently a special module created just for use with servomotors. Although the word "servomotor" is frequently used to describe a motor appropriate for use in a closed-loop control system, servomotors are not a particular sort of motor.

3.3.2.2 Component 2

(FINGERPRINT SENSOR)

The friction ridges on a human finger leave an impression known as a fingerprint. Obtaining partial fingerprints from a crime scene is a crucial forensic science technique. Fingerprints left on metal or glass surfaces by moisture and oil on the finger. By carefully applying ink or other substances from the peaks of friction ridges on the skin to a flat surface, such as paper, one can create impressions of entire fingerprints. Although fingerprint cards also often record sections of the lower joint areas of the fingers, fingerprint records typically contain impressions from the pad on the last joint of fingers and thumbs.

Human fingerprints are suitable as permanent markers of human identification because they are intricate, almost one-of-a-kind, challenging to change, and resilient throughout the course of an individual's lifetime. They may be used by the police or other authorities to identify people who want to hide their identity or to identify those who are unable to identify themselves due to illness or death, as can occur after a natural disaster.

3.3.2.3 Component 3

(BATTERY)

For the first transistor radios, the nine-volt battery, also known as the 9-volt battery, was developed. It has a top polarised snap connector and a rectangular prism form with rounded sides. Clocks, smoke alarms, and walkie-talkies all frequently employ this type. In primary carbon-zinc and alkaline chemistry, primary lithium iron disulfide, nickel-cadmium, nickel-metal hydride, and lithium-ion rechargeable forms, nine-volt batteries are frequently used. This type of mercury-oxide battery, which was once widely used, hasn't been produced in a long time due to its mercury concentration. This format has the designations NEDA 1604, IEC 6F22 (for zinccarbon), or MN 1604 6LR61 (for alkaline). No of the chemistry, the size is typically referred to as PP3; this name was formerly only used for carbon-zinc or, in some countries, E or E-block.

3.3.2.4 Component 4

(LED)

A semiconductor light source called a light-emitting diode (LED) produces light when current passes through it. Recombining electrons and electron holes in the

semiconductor results in the release of energy in the form of photons. Electroluminescence is the name of this phenomenon. The energy needed for electrons to pass the semiconductor's band gap determines the color of the light, which corresponds to the energy of the photons. White light is produced by stacking semiconductors or coating the semiconductor with a layer of light-emitting phosphor.

3.3.2.5 Component 5

(LCD DISPLAY)

A flat-panel display or other electronically manipulated optical device that makes use of liquid crystals' ability to modulate light is known as a liquid-crystal display (LCD). Liquid crystals don't directly emit light; instead, they create images in either colour or monochrome via a backlight or reflector. There are LCDs that can show arbitrary graphics (as on a general-purpose computer display) or fixed images with little information that may be seen or hidden, such text, numbers, and seven-segment displays like those used in digital clocks. They both make use of the same fundamental technology, however different displays have larger elements whereas random images are made up of a lot of tiny pixels. Depending on the polarizer configuration, LCDs can be switched between being normally on (positive) and off (negative). A character negative LCD will have a black backdrop with letters that are the same colour as the backlight, while a character positive LCD will have black writing on a background that is the opposite of the colour of the illumination. Blue LCDs have optical filters applied to the white to give them their distinctive appearance.

3.3.2.6 Component 6

(BUZZER)

A buzzer or beeper is an electronic signalling device that is frequently found in cars, home appliances like microwaves, or game shows. It typically consists of a number of switches or sensors connected to a control unit that determines whether and which button was pressed or if a predetermined amount of time has passed. Upon detection, the control unit typically illuminates the corresponding button or control panel with a light and emits an intermittent or continuous buzzing or beeping sound. The electromechanical technology on which this instrument was first built was essentially an electric bell minus the metal gong (which makes the ringing noise). These units frequently used a wall or ceiling as a sounding board and were fastened to one of those surfaces. A circuit that converts AC electricity into sound loud enough to operate a loudspeaker was created with certain AC-connected gadgets, and this circuit was connected to an inexpensive 8-ohm speaker. Nowadays, using a highpitched Son alert or other ceramic-based piezoelectric sounder is more common. These were typically connected to "driver" circuits that changed the sound's pitch or pulsed it on and off.

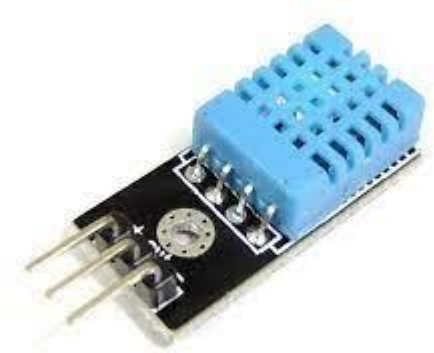
3.3.2.7 Component 7

(JUMPER WIRE)

An electrical wire, or collection of them in a cable, having a connector or pin at each end, is referred to as a jump wire. It is typically used to connect the parts of a breadboard, or other prototype or test circuit, internally or with other machinery or components, without soldering.

3.3.2.8 Component 8

(DHT SENSOR)



The low cost DHT temperature and humidity sensors . Although very sluggish and simple, these sensors are excellent for hobbyists who want to do some simple data logging. The thermistor and capacitive humidity sensor are the two components that make up the DHT sensors. A very simple chip that converts analogue data to digital data and outputs a digital signal with the temperature and humidity is also included within. Any microcontroller can easily read the digital signal.

3.3.2.9 Component 9

(POT METER)

A three-terminal resistor with a sliding or revolving contact that creates a variable voltage divider is called a potentiometer. It functions as a variable resistor or rheostat if only the wiper and one end of the connector are used. The component, hence its name, is an implementation of the same idea as the measuring device known as a potentiometer, which is effectively a voltage divider used to measure electric potential (voltage). Electrical devices, such as the volume controls on audio equipment, are frequently controlled by potentiometers. For instance, with a

joystick, potentiometers controlled by a mechanism can serve as position transducers**.**

3.3.2.10 Component 10

(PUSH BUTTON)

A push-button, often known as a push button or just a button, is a straightforward switch mechanism used to regulate a machine's or process's operation. Usually constructed of metal or plastic, buttons are made of strong materials. The surface is often flat or contoured to fit a human finger or hand, making it simple to push or depress. The majority of the time, buttons function as biassed switches, while many un-biased buttons still need a spring to return to their unpressed state due to their physical nature.

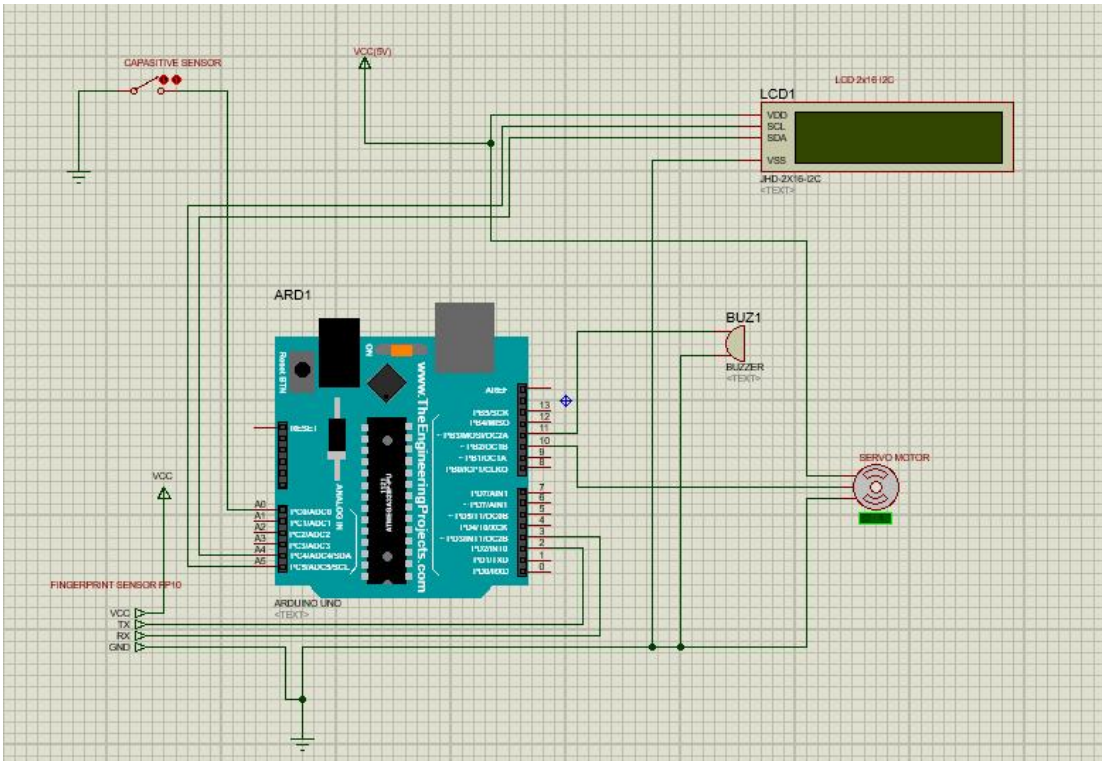
3.3.2.11 Component 11

(BATTERY HOLDER)

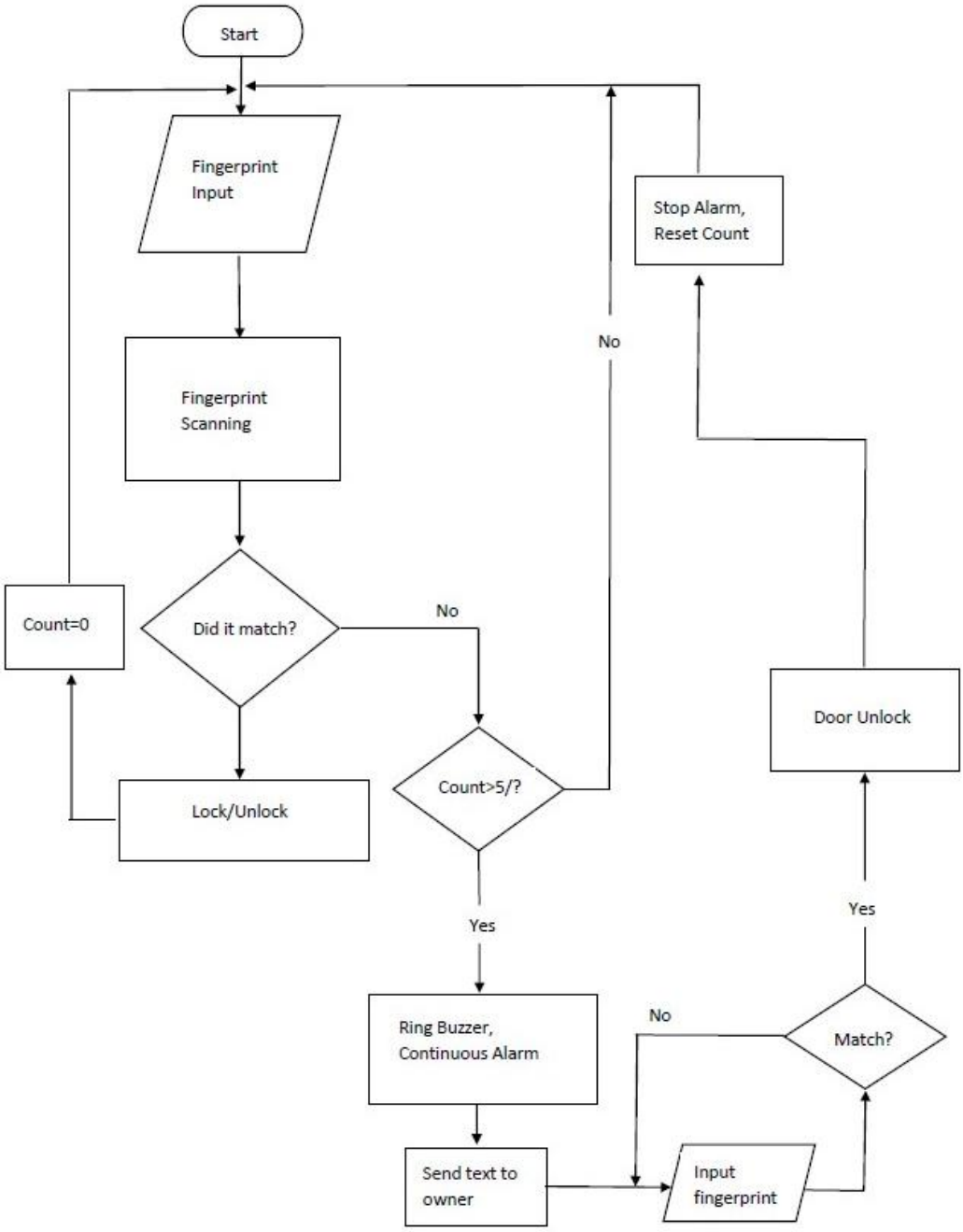
A battery holder is a chamber or compartment that can accommodate one or more batteries. The holder must also make electrical contact with the battery terminals when using dry cells. Cables are frequently attached to the battery terminals for wet cells, as is the case with emergency lighting equipment or autos. A battery holder can be a separate plastic holder that is mounted using screws, eyelets, adhesive, doublesided tape, or another method, or it can be a plastic case with the

shape of the housing moulded as a compartment or compartments that receive a battery or batteries. Battery holders may have a lid to keep the batteries within and preserve them, or they may be sealed to stop battery leakage from harming the circuitry and other components. The two most popular ways to connect electrically within a holder are coils of spring wire or flat tabs that press against the battery terminals. On battery holders, external connections are typically established using pin contacts, surface mount feet, solder lugs, or wire leads.

3.3.3 Circuit Operation



3.4 Flowchart of the System



3.4.1 Description of Flowchart

I've shown the procedure for this project in the flowchart above. First, after placing the finger, the fingerprint sensor will scan the fingerprint of the owner. Then, if it matches with the record the door would lock or unlock. In case, the fingerprint does not match the through the buzzer an alarm would ring and would send a text to the owner to show that the fingerprint does not match. Then, the owner should place the fingerprint again to see whether it matches or not if yes then the door would unlock and the alarm would stop ringing plus it would reset the count but if no then the whole process would be repeated.

3.5 Sustainability Element in The Design Concept

Utilizing fingerprint technology in these fields promotes sustainable development from an environmental standpoint. It can be seen using the manufacturer Bio link as an example. The manufacture of fingerprint FTIR scanners complies with regulations governing the restriction of hazardous substances. Although infrared lights are used to scan fingerprints, they are safe since no one can see the rays while they are being scanned. Various metals and plastics are used by the manufacturer in the production of their products. The scanning method uses organic glass. The energy consumption of a terminal scanner is 1 Ampere/h, compared to 150 milliampere/h for a USB scanner. This technology is resource-efficient, uses less resources than usual, and uses relatively little energy. Its main goal is to satisfy consumer expectations by facilitating quicker system verification (operational efficiency) and offering physical and information security for those involved in a specific area. For homes or businesses, fingerprint door locks are a great investment. It offers excellent security by restricting unauthorized access. With the addition of the authorized person's distinctive biological traits, this device raises the bar for security. Fingerprint-based door lock systems are the greatest option for anyone who wants to increase the security of their houses.

3.6 Chapter Summary

The project's research methodology is covered in this chapter. I've included the project's block diagram, project description, hardware, circuit diagram, component description, circuit operation, project flowchart, and flowchart description in this study technique.

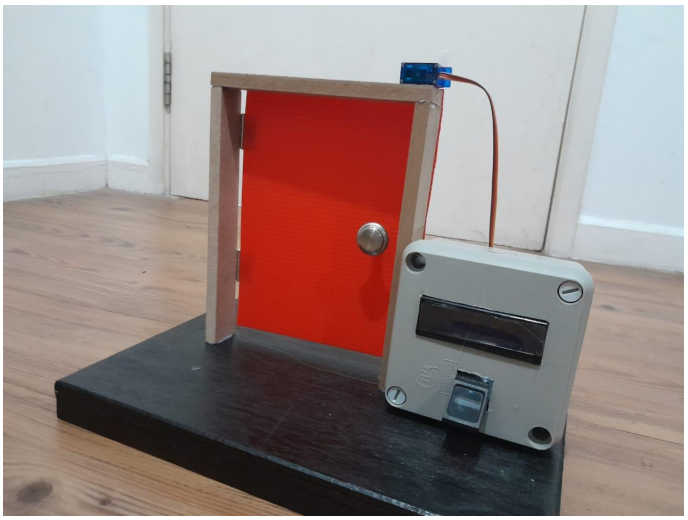
CHAPTER 4

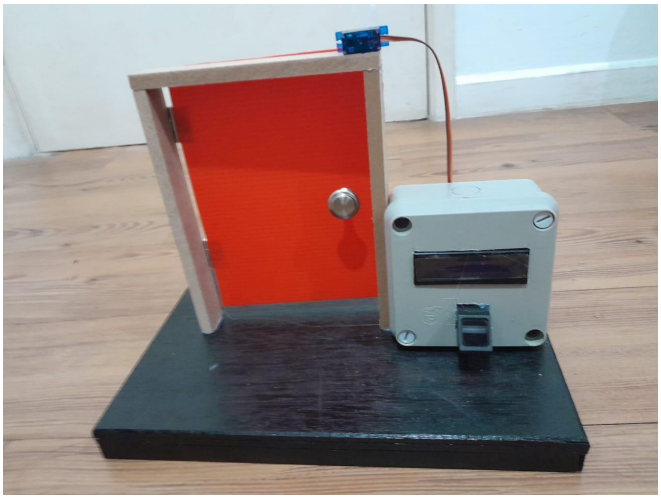
4 RESULTS AND DISCUSSION

4.1 Introduction

The components of the system that we built and tested are examined in this chapter. Along with the qualitative information, the development of the questionnaire, and the outcomes, the quantitative findings of the project's research were also analyzed. The findings are also reviewed in the context of earlier research findings and the literature in order to pinpoint parallels, variable changes in the project, and contrasts between this study and prior studies and literature. To successfully complete this study, test the hypothesis, and address the research issues, data analysis is crucial.

4.2 Results and Analysis





(Diagram above shows the final result of my project)

The results portion of the research paper should seek to summarize the findings without attempting to analyse or evaluate them. It ought to serve as a road map for the discussion portion as well. The analysis will aid the public when the results are revealed. The author explains the methodology used to use the data from the analysis section. It is clear that the prototype we built meets our requirements and operates wonderfully. People can utilize the product risk-free, the method can be improved, and there is less chance of human error.When a user tries to open the gate, the module verifies the user's identity by comparing the user's fingerprint with the database; if the fingerprint matches the one in the database, Arduino sends the signal to run the motor, which opens the gate and displays the user's name on the LCD.

4.3 Discussion

The scope of work frequently discusses project requirements and describes how particular project tasks will contribute to the achievement of the project's goals. The work that was done on the project will typically be summarized in this section. I created a schedule so that I could quickly and efficiently split the work that needs to be done.

4.4 Chapter Summary

A simple explanation sequence has been devised to condense this chapter portion. First and foremost, as soon as I finished, the system began to function under the direction of the fingerprint sensor. The usefulness or defects of a project must be tested in order to be discovered. After the test was complete, the project's following phase involved producing a result, an analysis, and a discussion to satisfy its characteristic. It is feasible to assess the general state of the project using the test results as a starting point. The overall project's operational traits and work ethic can then be succinctly articulated in the discussion that follows.

CHAPTER 5

5 CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

Chapter 5 concentrates on Project 2's conclusion. This chapter will describe our achievements and objectives for creating a "Biometric Fingerprint Sensor to Lock and Unlock Door." With a specific goal in mind, the "Biometric Fingerprint Sensor to Lock and Unlock Door" enables users to save time and energy without moving a single cell in their bodies, as my product will take care of all the work. We can see how much the work that is done has changed with the introduction of biometric fingerprint sensors to lock and unlock doors to reduce human mistake.

5.2 Conclusion

The project can be improved to make it more interesting, appealing to other customers, and useful going forward. If we introduce this product to the market, it will simplify everyone's work. As a result, this biometric fingerprint sensor that locks and unlocks doors offers customers greater security and is more user-friendly because it can store up to 200 fingerprints. As a result, the usefulness, resilience, and sustainability of our project are given more weight in terms of efficiency. The project is not only simple for the user to operate and offers good security, but it also lasts for a very long time without putting the user or others in danger.

5.3 Suggestion for Future Work

Biometric identification management technology is advancing at an extremely rapid rate. In the future, we will create advanced systems with several functionalities, such as sms alerts when an authorized person attempts to lock the door.Based on a password system and an image recognition process. Additionally, biometric technology makes people convenient in real life by using their eyes and retina as a password that aids authorised individuals in admission authentication. Thus, the fingerprint-based door lock system need further development.

5.4 Chapter Summary

A final conclusion that evaluates the project's overall progress must be produced once all of the project's work has been completed. The conclusion of a project may also show whether or not its planned purpose or scope was achieved. A project performance assessment is needed to find any potential weak points after the conclusion phase. If there is, a stronger

call to stop it will be heard. As a result, future initiatives will be enhanced, which is advantageous. Since they will encourage uniqueness and creativity while also promoting society as a whole, these conclusions and recommendations are essential to us and our job. My enthusiasm and effort have gone into this undertaking, thus it is very easy to identify our hypothesis or objectives. If we want to strengthen the legitimacy of the country in the future, we need to understand why we are taking this action and encourage education and creativity.

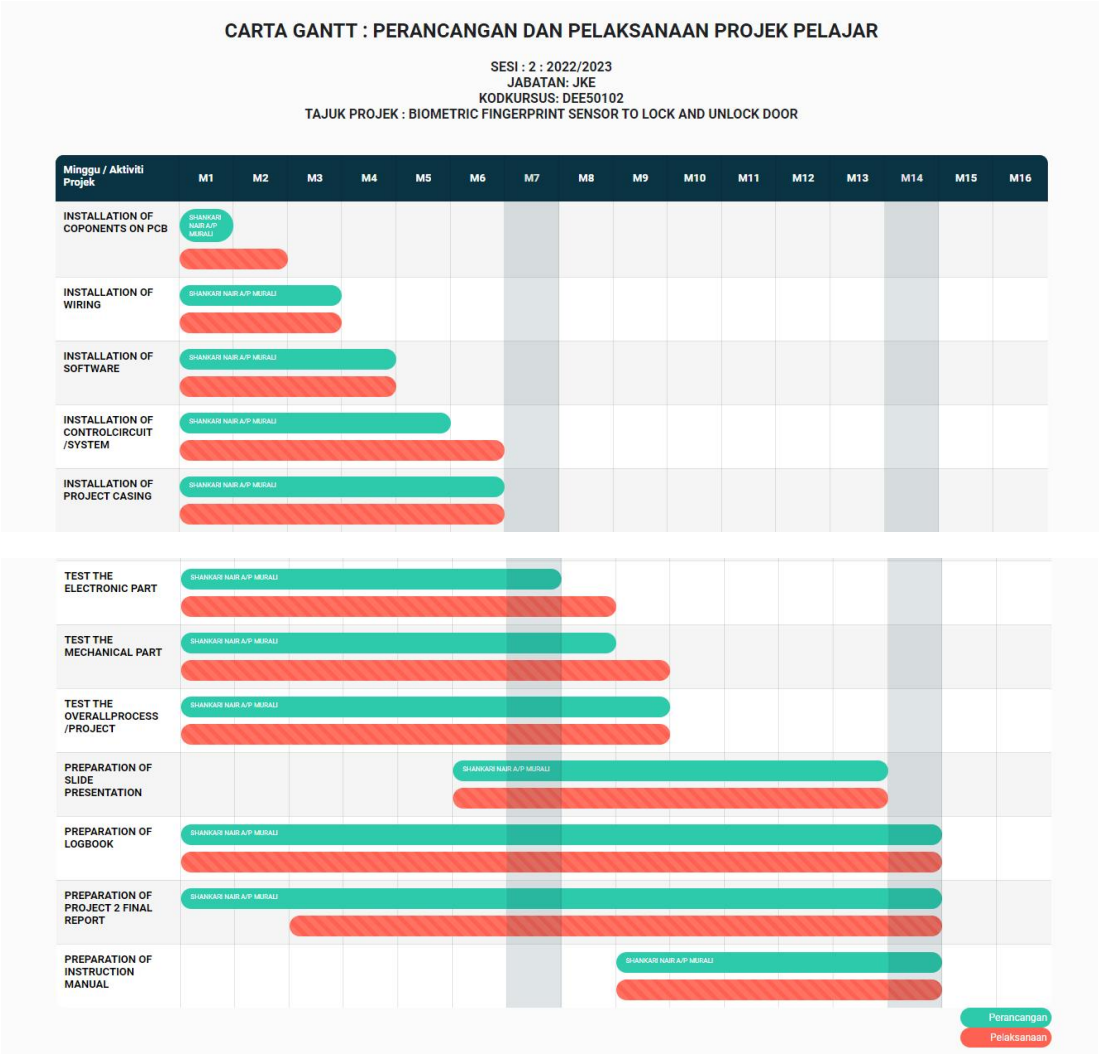
CHAPTER 6

6 PROJECT MANAGEMENT AND COSTING

6.1 Introduction

Because we don't have a sponsor, we had to use our own money to buy the majority of the essential components and supplies for the project. The estimated price is RM 793.45. Compared to the other projects, its cost is much lower than the budget. The development costs are still manageable for the following six months. The research indicates that it is doable and attainable.

6.2 Gant Chart and Activities of the Project



6.3 Cost and Budgeting

This project involves the cost of purchasing components and materials throughout its implementation components involving hardware Arduino Mega, fingerprint sensor, servo motor, battery, led, LCD display, buzzer, connecting wires (jumper wires), DHT11, Pot sensor, push button and battery holder are components that have a cost. To make things simpler and save money, all of these components are obtained through online methods.

The overall gross budget estimate in the implementation of this project is RM 543.45 and other expenses is at RM 250.00. According to this budget cost, this project is can be considered as a less costly project compared to other projects that can cost over a thousand ringgit. The project's cost is also consistent with one of the essential qualities of a competent project developer, which is having a cheap cost yet higher quality project.

No.	Component and materials	The unit price	Quantity	Total
1	Arduino Mega	RM 212.84	1	RM 212.84
2	Finger print Sensor	RM 55.90	1	RM 55.90
3	Servo Motor	RM 9.80	1	RM 9.80
4	Battery	RM 27.75	1	RM 27.75
5	led	RM 0.066	some	RM 6.60
6	LCD display	RM 199.00	1	RM 199.00
7	Buzzer	RM 8.45	1	RM 8.45
8	Connecting wires (jumper wires)	RM 0.09	some	RM 3.60
9	DHT11	RM 4.80	1	RM 4.80
10	POT sensor	RM 4.70	1	RM 4.70
11	Push button	RM 7.20	1	RM 7.20
12	Battery holder	RM 2.81	1	RM 2.81
			Total :	**RM 543.45**
	List of other costing			
1	Transportation			
2	Postage			
3	Craft Work			
4	Internet			
5	Application			
			Total :	**RM250.00**
			Overall total	**RM793.45**

6.4 Chapter Summary

Costing and project management have both been covered in this chapter. A gantt chart and the project's operations are given. The project's cost and budget list, complete with component quantities and the amount, is also included.

REFERENCES

- Piash paul, Md.Abdullah Al Achib, Hazrat Sauda Hossain, Md. Kaiviul Hossain . (2015). Smart door lock using fingerprint sensor.

- Suyash Gaikwad *1, Sanjiv Jamadar*2, Rupali Shelke*3. (2021). FINGERPRINT BASED DOOR LOCK USING ARDUINO. *1,2Student,Department of Computer Engineering, MM Polytechnic, Pune, Maharashtra India. *3Co-Ordinator, Department of Computer Engineering, MM Polytechnic, Pune, Maharashtra, India.

- SHUHADA NATASHA BINT MOHD ZAINOR. (2012). DOOR ACCESS SYSTEM – ARDUINO
BASED. Department of Electrical & Electronic Engineering in Partial Fulfilment of the Requirements for the Degree Bachelor of Engineering (Hons) (Electrical & Electronic Engineering)

- K.Rajesh#1,ASST.PROFESSER,B.Venkata Rao#2,P.AV.S.K.Chaitanya#3,A.ruchitha Reddy#4. (2019). Smart door unlock system using fingerprint. #Department of ECE, Chalapathi Institute of Engineering and Technology, Lam, Guntur, AP, India -522034.

Aleksander IBRO, Augusto WONG, Maria ZYLA.(2019).Smart Door Lock. the Faculty of WORCESTER POLYTECHNIC INSTITUTE in partial fulfillment of the requirements for the Degree in Bachelor of Science.

7 APPENDICES

APPENDIX A- PROGRAMMING

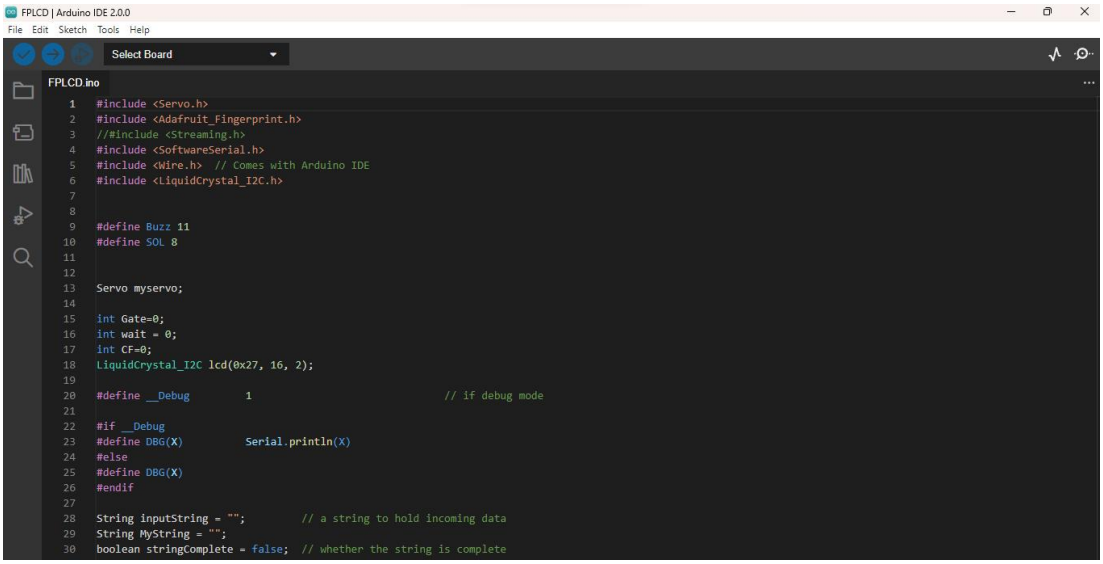
APPENDIX B- PROJECT MANUAL/PRODUCT CATALOGUE

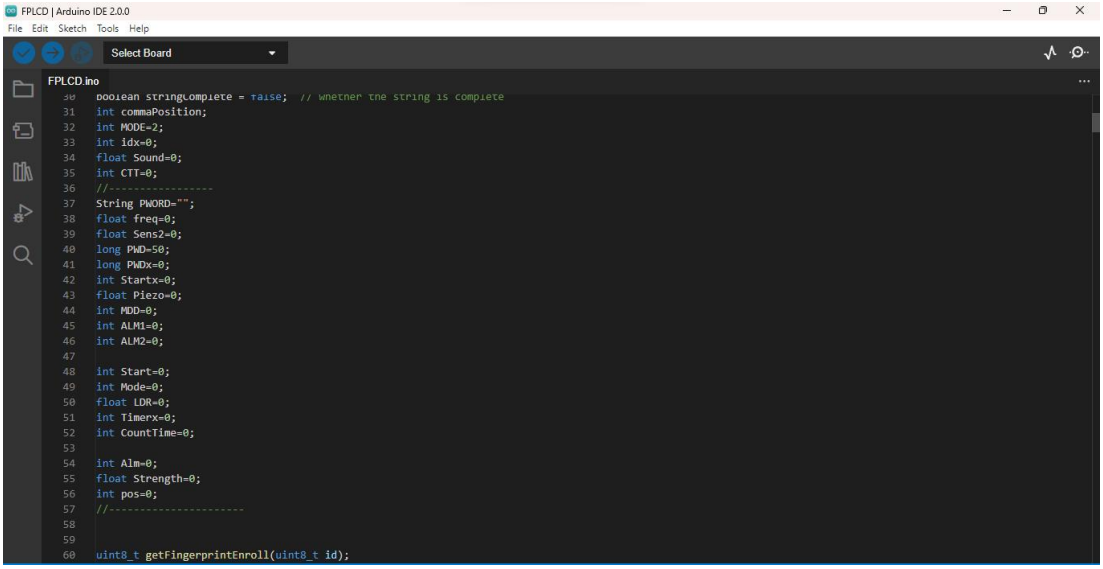
APPENDIX A- PROGRAMMING

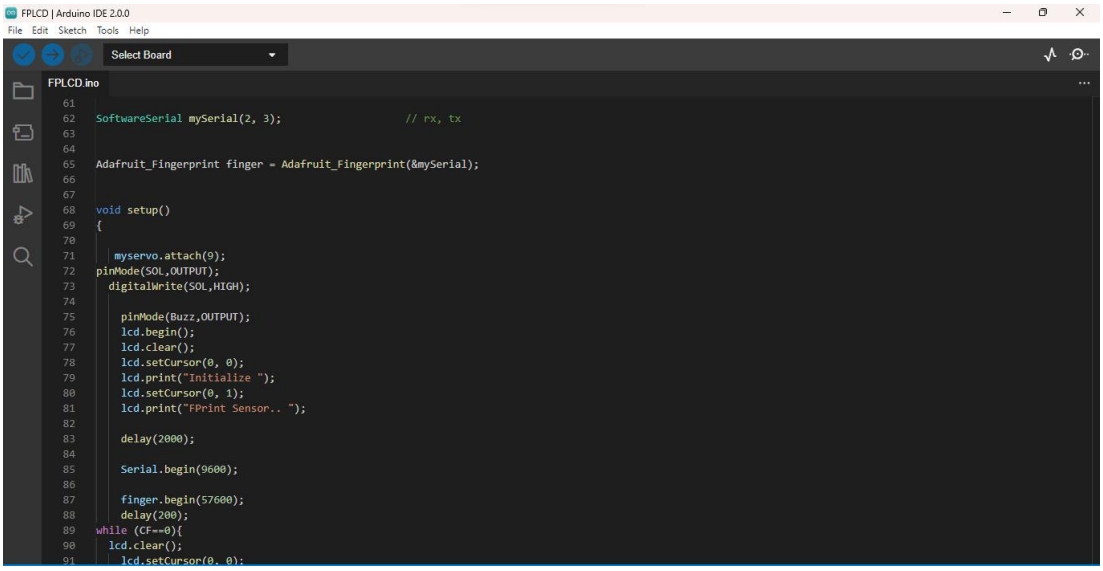
Coding is the process of converting codes between different languages. It can also be regarded as a subset of programming because it executes the initial stages of programming. It necessitates writing programmes as directed in a range of languages. The machine can only understand machine code, also referred to as binary language, and cannot communicate with people. The primary responsibility of a coder is to translate requirements into language that computers can understand. Coders must be fluent in the working language of the project. They do, however, generally write their code in compliance with the requirements and guidelines of the project. The process of creating a software product starts here. Programming is the process of creating a machine-level executable programme that can be executed without mistake. To keep human inputs and machine outputs in sync, formal coding is the practise that is used.

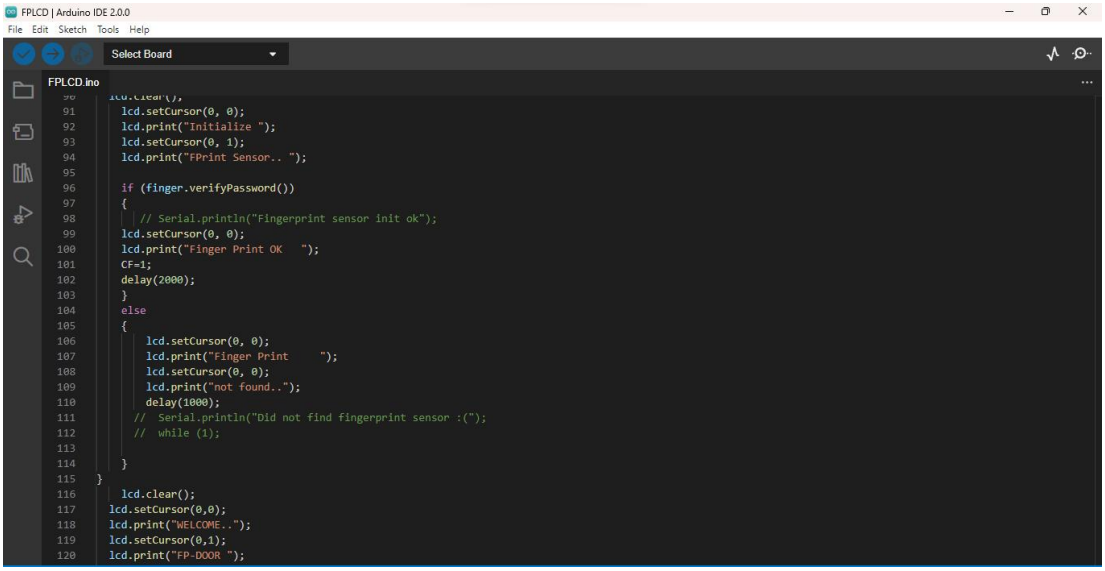
Writing code is the initial step, which is then examined and put into practise to produce the desired machine-level result. Additionally, it takes into account all of the significant factors, including debugging, compilation, testing, and implementation. To create the required machine outputs, programmers analyse and comprehend the numerous communication components. The Arduino Integrated Development Environment (IDE) is used to develop Arduino programming. You may create sketches (Arduino jargon for programmes) for several Arduino boards using the Arduino IDE, an application that runs on your computer. Processing, a very basic hardware programming language akin to C, is the foundation of the Arduino programming language. The sketch must be uploaded to the Arduino board for execution after being created in the Arduino IDE.

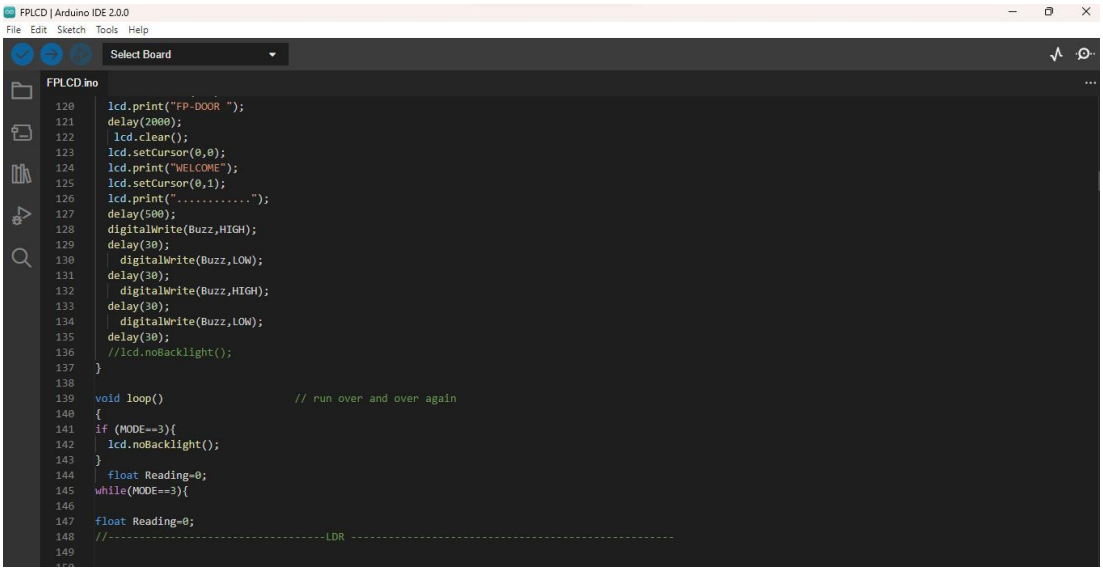
CODING IN ARDUINO IDE:

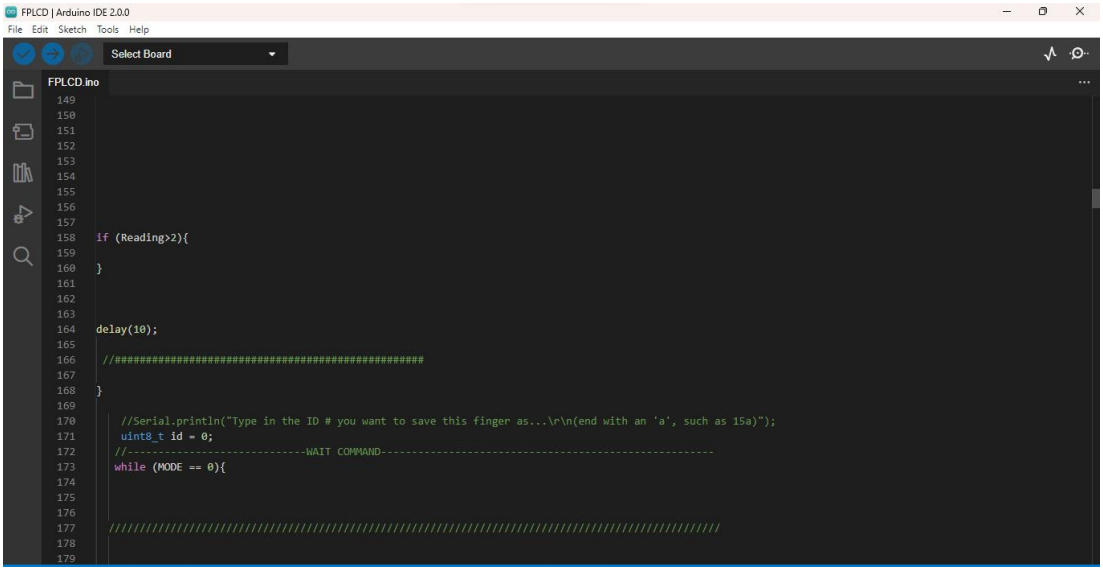


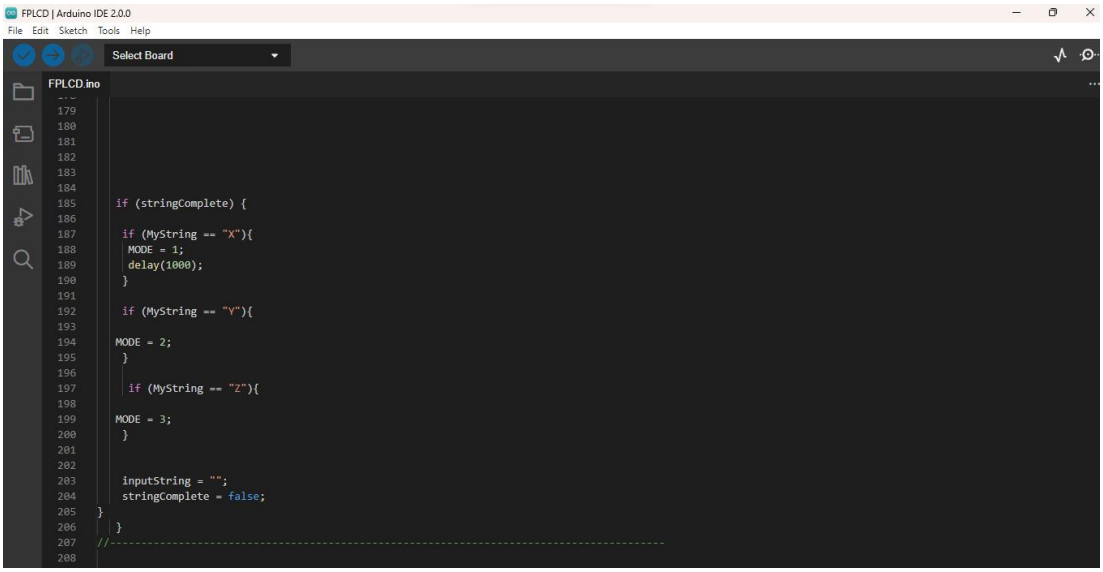


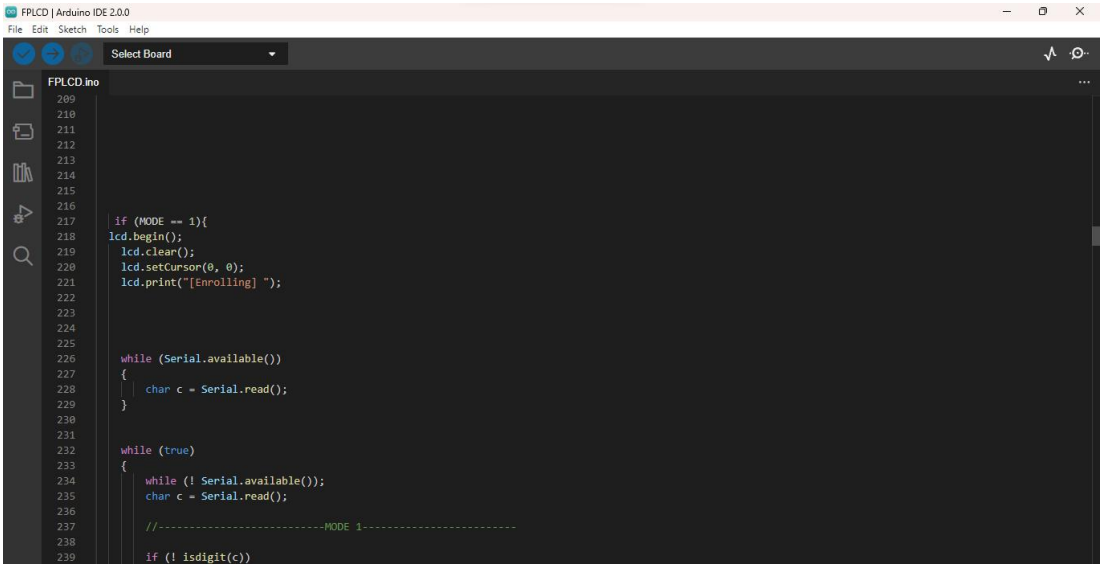

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
SoftwareSerial mySerial(2, 3); // rx, tx
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
void setup()
{
myservo.attach(9);
pinMode(SOL,OUTPUT);
digitalWrite(SOL,HIGH);
pinMode(Buzz,OUTPUT);
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Initialize ");
lcd.setCursor(0, 1);
lcd.print("FPrint Sensor.. ");
delay(2000);
Serial.begin(9600);
finger.begin(57600);
delay(200);
while (CF==0){
lcd.clear();

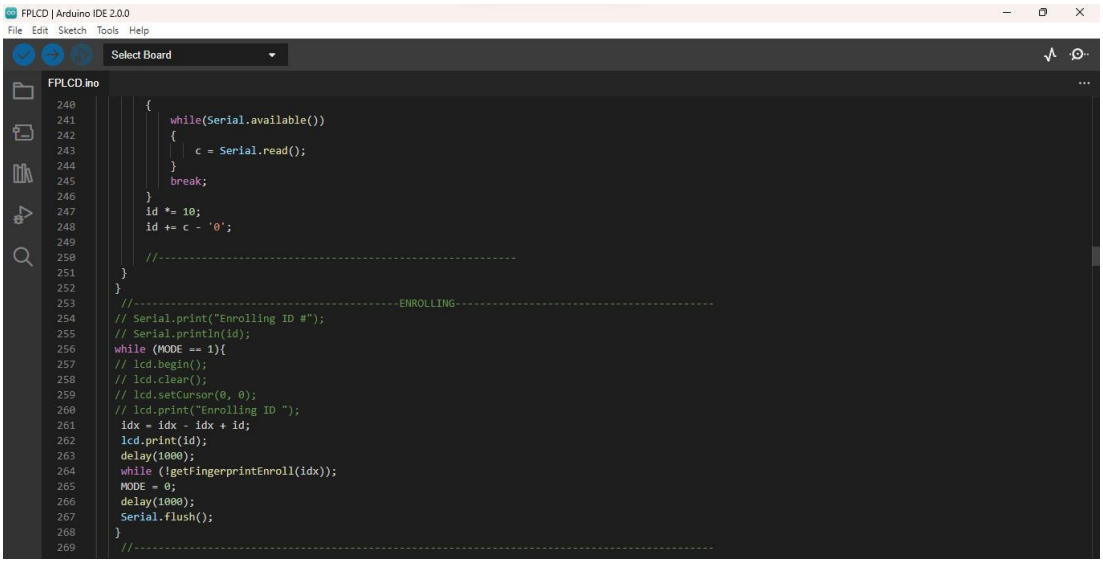

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
lcd.setCursor(0, 0);
lcd.print("Initialize ");
lcd.setCursor(0, 1);
lcd.print("FPrint Sensor.. ");
if (finger.verifyPassword())
{
// Serial.println("Fingerprint sensor init ok");
lcd.setCursor(0, 0);
lcd.print("Finger Print OK ");
CF=1;
delay(2000);
}
else
{
lcd.setCursor(0, 0);
lcd.print("Finger Print ");
lcd.setCursor(0, 0);
lcd.print("not found..");
delay(1000);
// Serial.println("Did not find fingerprint sensor :(");
// while (1);
}
}
lcd.clear();
lcd.setCursor(0,0);
lcd.print("WELCOME..");
lcd.setCursor(0,1);
lcd.print("FP-DOOR ");

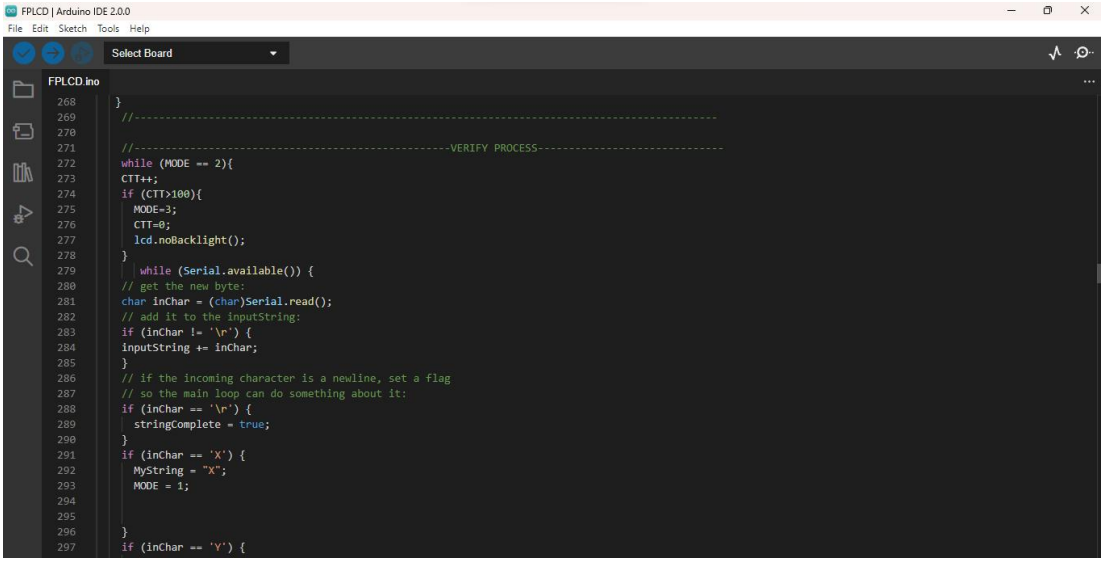

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
lcd.print("FP-DOOR ");
delay(2000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("WELCOME");
lcd.setCursor(0,1);
lcd.print(".....");
delay(500);
digitalWrite(Buzz,HIGH);
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
digitalWrite(Buzz,HIGH);
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
//lcd.noBacklight();
}
void loop() // run over and over again
{
if (MODE==3){
lcd.noBacklight();
}
float Reading=0;
while(MODE==3){
float Reading=0;
//-----------------------------------LDR --

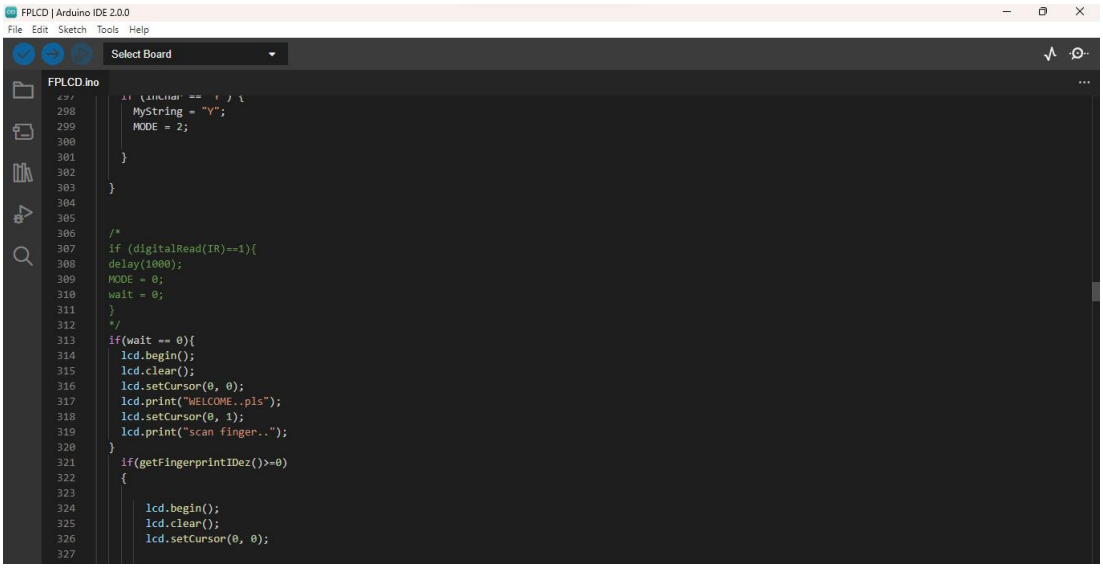

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
if (Reading>2){
}
delay(10);
//##
}
//Serial.println("Type in the ID # you want to save this finger as...\r\n(end with an 'a', such as 15a)");
uint8_t id = 0;
//----------------------------WAIT COMMAND--
while (MODE == 0){
//

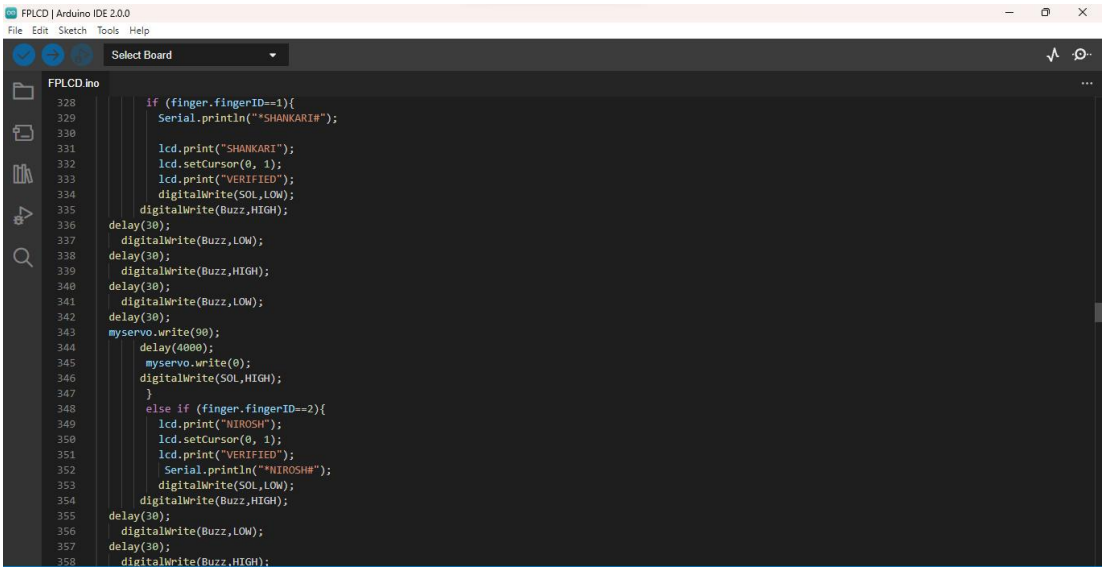

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
if (stringComplete) {
if (MyString == "X"){
MODE = 1;
delay(1000);
}
if (MyString == "Y"){
MODE = 2;
}
if (MyString == "Z"){
MODE = 3;
}
inputString = "";
stringComplete = false;
}
}
//----------

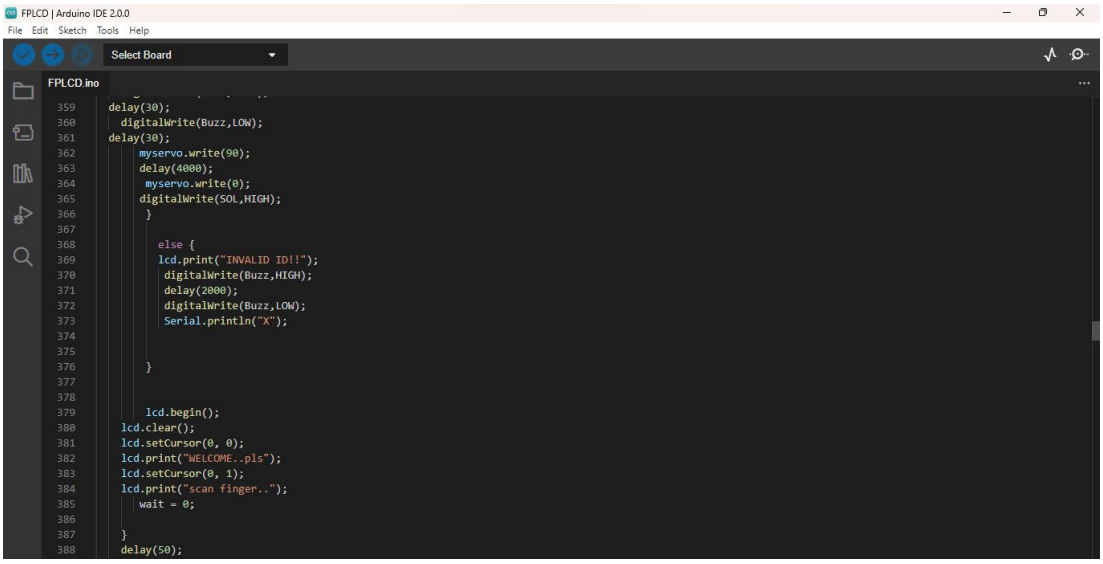

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
if (MODE == 1){
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("[Enrolling] ");
while (Serial.available())
{
char c = Serial.read();
}
while (true)
{
while (! Serial.available());
char c = Serial.read();
//----------MODE 1----------
if (! isdigit(c))

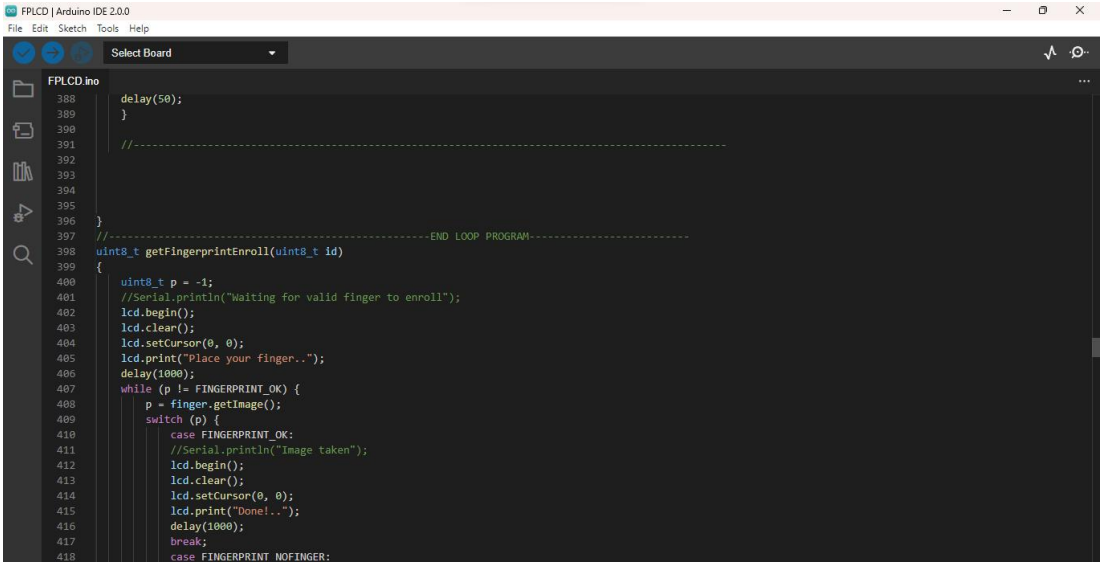

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
{
while(Serial.available())
{
c = Serial.read();
}
break;
}
id *= 10;
id += c - '0';
//---
}
}
//---ENROLLING---
// Serial.print("Enrolling ID #");
// Serial.println(id);
while (MODE == 1){
// lcd.begin();
// lcd.clear();
// lcd.setCursor(0, 0);
// lcd.print("Enrolling ID ");
idx = idx - idx + id;
lcd.print(id);
delay(1000);
while (!getFingerprintEnroll(idx));
MODE = 0;
delay(1000);
Serial.flush();
}
//---

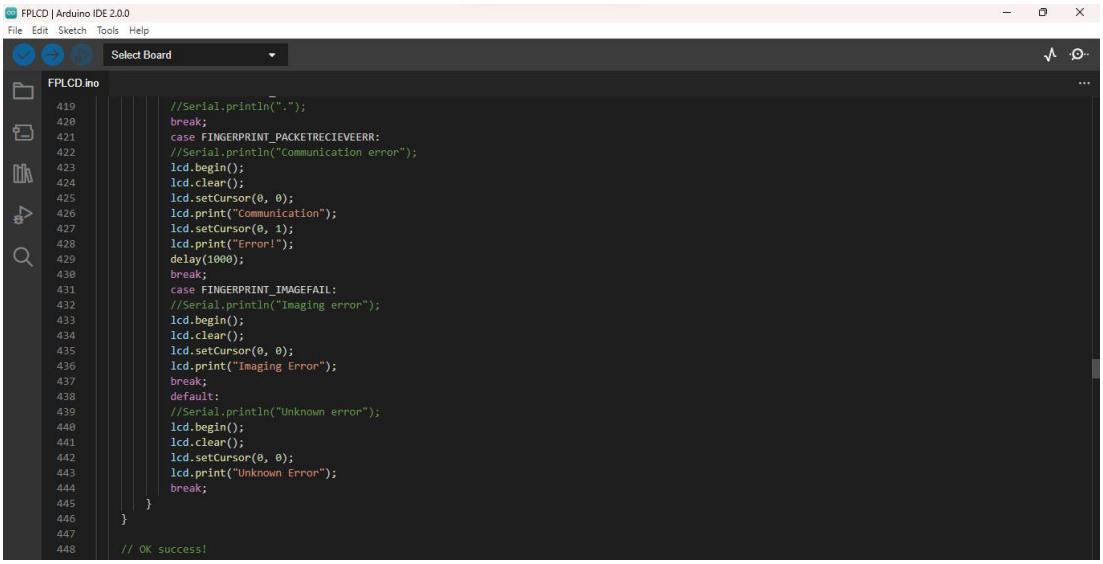

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
}
//---
//---VERIFY PROCESS-----------------------------
while (MODE == 2){
CTT++;
if (CTT>100){
MODE=3;
CTT=0;
lcd.noBacklight();
}
while (Serial.available()) {
// get the new byte:
char inChar = (char)Serial.read();
// add it to the inputString:
if (inChar != '\r') {
inputString += inChar;
}
// if the incoming character is a newline, set a flag
// so the main loop can do something about it:
if (inChar == '\r') {
stringComplete = true;
}
if (inChar == 'X') {
MyString = "X";
MODE = 1;
}
if (inChar == 'Y') {

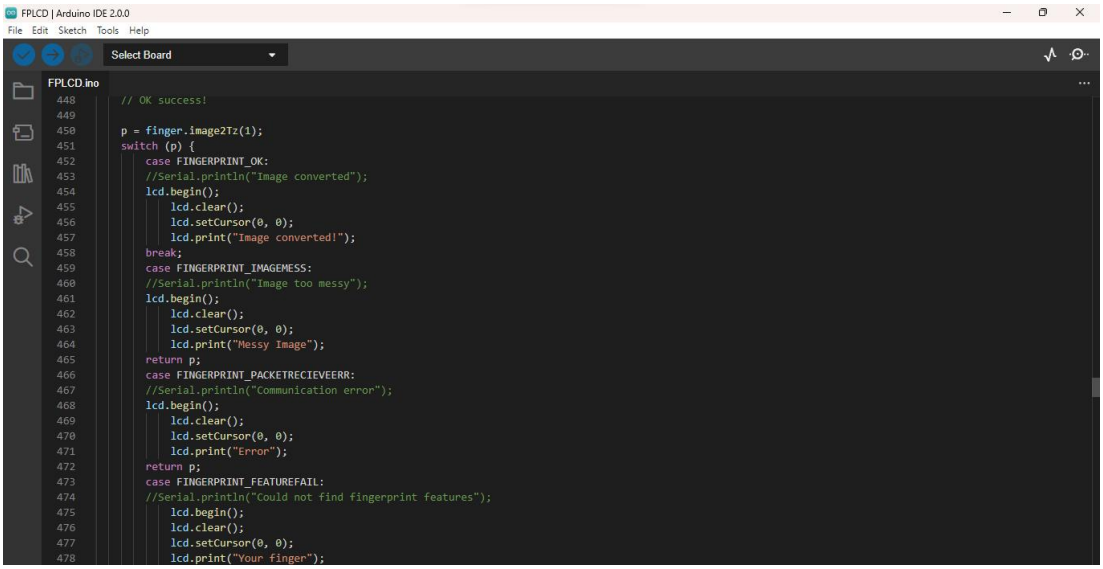

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
MyString = "Y";
MODE = 2;
}
}
/*
if (digitalRead(IR)==1){
delay(1000);
MODE = 0;
wait = 0;
}
*/
if(wait == 0){
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("WELCOME..pls");
lcd.setCursor(0, 1);
lcd.print("scan finger..");
}
if(getFingerprintIDez()>=0)
{
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);

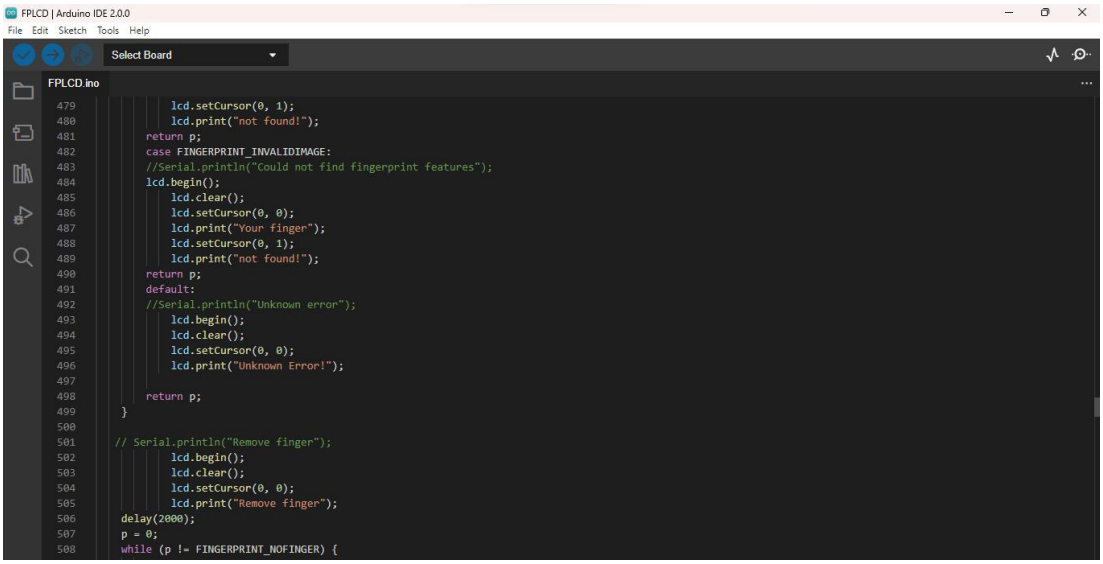

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
if (finger.fingerID==1){
Serial.println("*SHANKARI#");
lcd.print("SHANKARI");
lcd.setCursor(0, 1);
lcd.print("VERIFIED");
digitalWrite(SOL,LOW);
digitalWrite(Buzz,HIGH);
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
digitalWrite(Buzz,HIGH);
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
myservo.write(90);
delay(4000);
myservo.write(0);
digitalWrite(SOL,HIGH);
}
else if (finger.fingerID==2){
lcd.print("NIROSH");
lcd.setCursor(0, 1);
lcd.print("VERIFIED");
Serial.println("*NIROSH#");
digitalWrite(SOL,LOW);
digitalWrite(Buzz,HIGH);
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
digitalWrite(Buzz,HIGH);

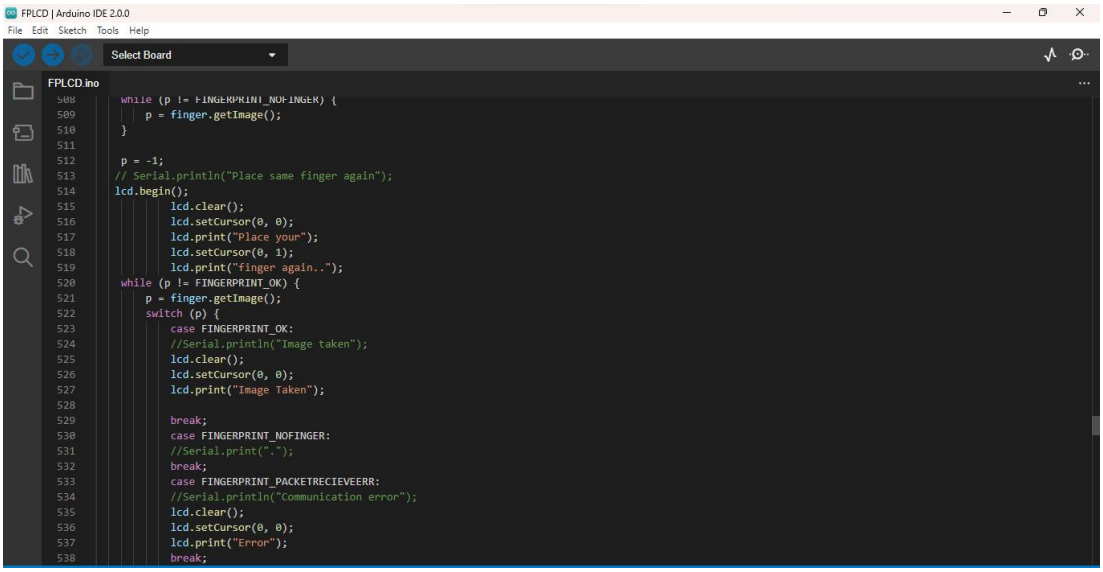

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
delay(30);
digitalWrite(Buzz,LOW);
delay(30);
myservo.write(90);
delay(4000);
myservo.write(0);
digitalWrite(SOL,HIGH);
}
else {
lcd.print("INVALID ID!!");
digitalWrite(Buzz,HIGH);
delay(2000);
digitalWrite(Buzz,LOW);
Serial.println("X");
}
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("WELCOME..pls");
lcd.setCursor(0, 1);
lcd.print("scan finger..");
wait = 0;
}
delay(50);

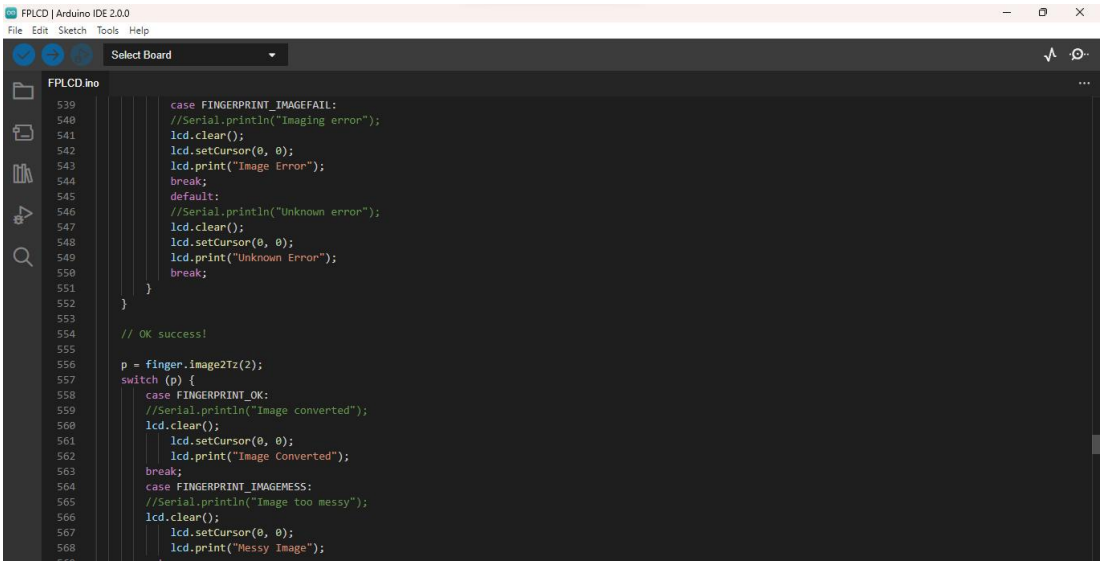

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
delay(50);
}
//----------
}
//----------END LOOP PROGRAM----------
uint8_t getFingerprintEnroll(uint8_t id)
{
uint8_t p = -1;
//Serial.println("Waiting for valid finger to enroll");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Place your finger..");
delay(1000);
while (p != FINGERPRINT_OK) {
p = finger.getImage();
switch (p) {
case FINGERPRINT_OK:
//Serial.println("Image taken");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Done!..");
delay(1000);
break;
case FINGERPRINT_NOFINGER:

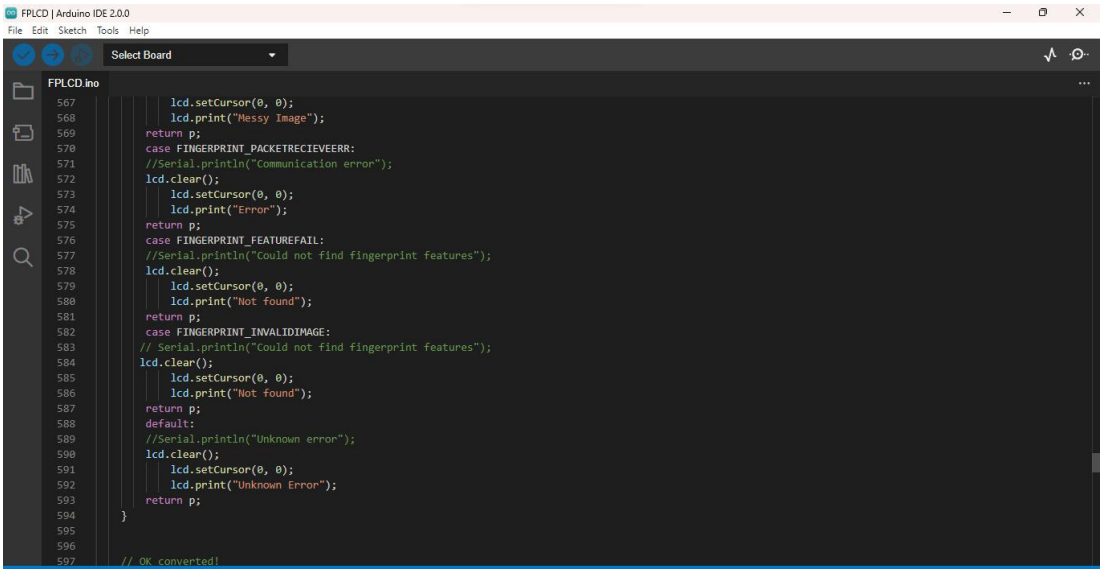

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
//Serial.println(".");
break;
case FINGERPRINT_PACKETRECIEVEERR:
//Serial.println("Communication error");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Communication");
lcd.setCursor(0, 1);
lcd.print("Error!");
delay(1000);
break;
case FINGERPRINT_IMAGEFAIL:
//Serial.println("Imaging error");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Imaging Error");
break;
default:
//Serial.println("Unknown error");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Unknown Error");
break;
}
}
// OK success!

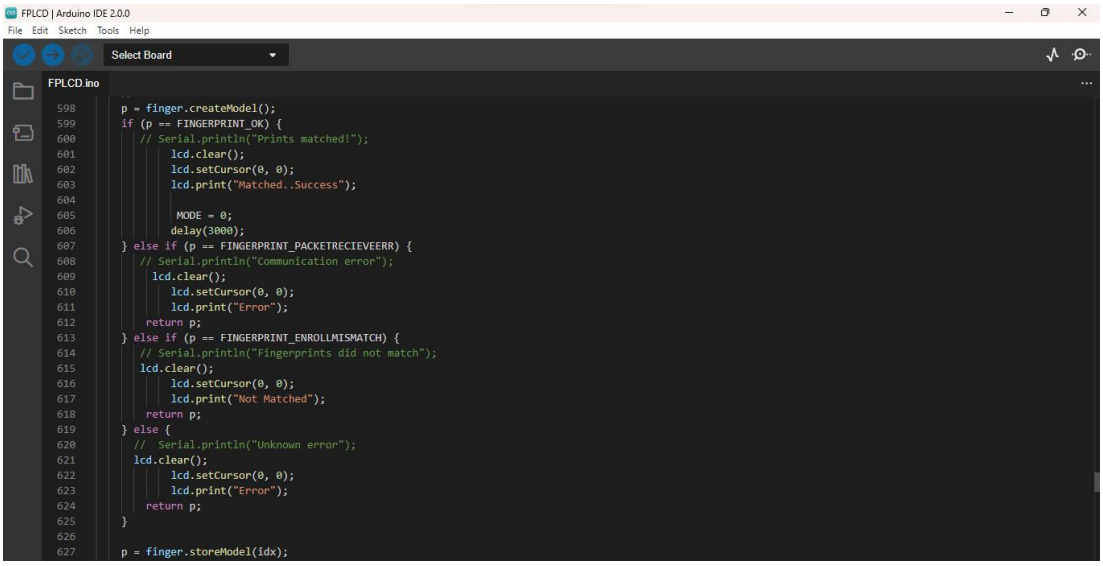

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
// OK success!
p = finger.image2Tz(1);
switch (p) {
case FINGERPRINT_OK:
//Serial.println("Image converted");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Image converted!");
break;
case FINGERPRINT_IMAGEMESS:
//Serial.println("Image too messy");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Messy Image");
return p;
case FINGERPRINT_PACKETRECIEVEERR:
//Serial.println("Communication error");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
case FINGERPRINT_FEATUREFAIL:
//Serial.println("Could not find fingerprint features");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Your finger");

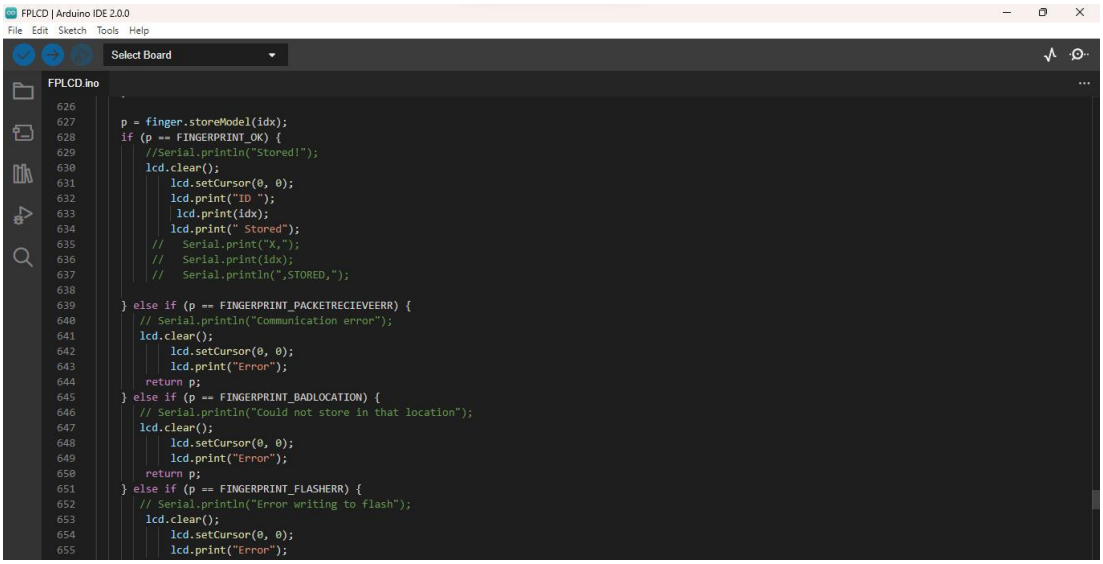

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
lcd.setCursor(0, 1);
lcd.print("not found!");
return p;
case FINGERPRINT_INVALIDIMAGE:
//Serial.println("Could not find fingerprint features");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Your finger");
lcd.setCursor(0, 1);
lcd.print("not found!");
return p;
default:
//Serial.println("Unknown error");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Unknown Error!");
return p;
}
// Serial.println("Remove finger");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Remove finger");
delay(2000);
p = 0;
while (p != FINGERPRINT_NOFINGER) {

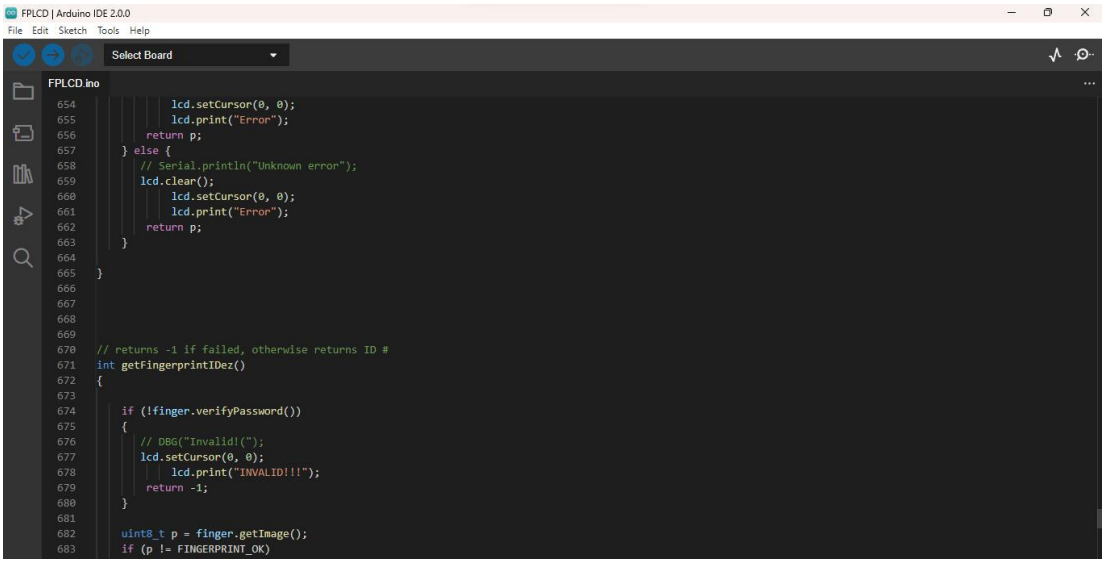

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
while (p != FINGERPRINT_NOFINGER) {
p = finger.getImage();
}
p = -1;
// Serial.println("Place same finger again");
lcd.begin();
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Place your");
lcd.setCursor(0, 1);
lcd.print("finger again..");
while (p != FINGERPRINT_OK) {
p = finger.getImage();
switch (p) {
case FINGERPRINT_OK:
//Serial.println("Image taken");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Image Taken");
break;
case FINGERPRINT_NOFINGER:
//Serial.print(".");
break;
case FINGERPRINT_PACKETRECIEVEERR:
//Serial.println("Communication error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
break;

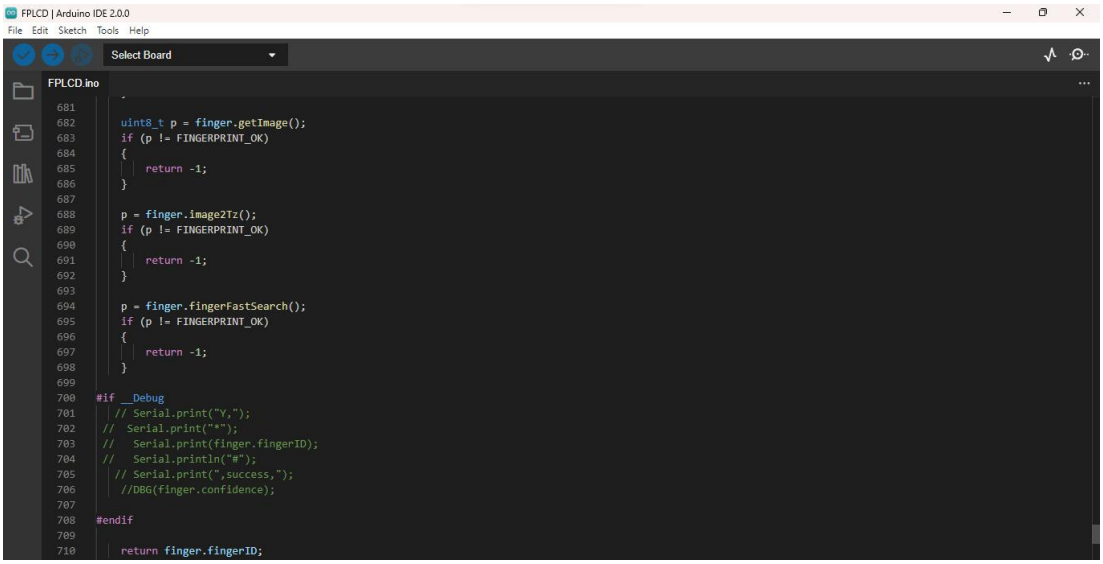

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
case FINGERPRINT_IMAGEFAIL:
//Serial.println("Imaging error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Image Error");
break;
default:
//Serial.println("Unknown error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Unknown Error");
break;
}
}
// OK success!
p = finger.image2Tz(2);
switch (p) {
case FINGERPRINT_OK:
//Serial.println("Image converted");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Image Converted");
break;
case FINGERPRINT_IMAGEMESS:
//Serial.println("Image too messy");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Messy Image");

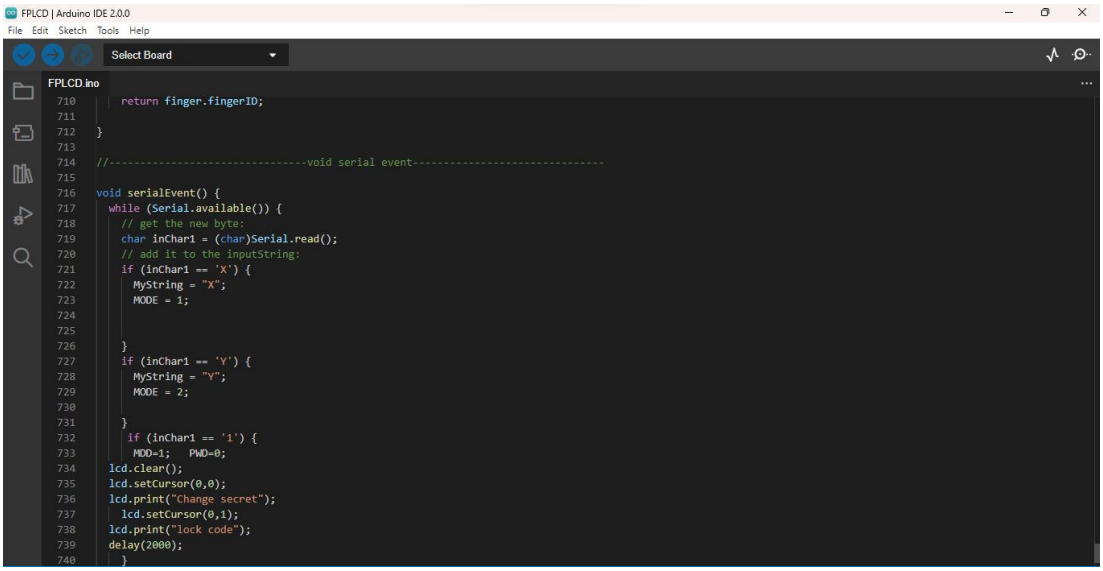

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
lcd.setCursor(0, 0);
lcd.print("Messy Image");
return p;
case FINGERPRINT_PACKETRECIEVEERR:
//Serial.println("Communication error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
case FINGERPRINT_FEATUREFAIL:
//Serial.println("Could not find fingerprint features");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Not found");
return p;
case FINGERPRINT_INVALIDIMAGE:
// Serial.println("Could not find fingerprint features");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Not found");
return p;
default:
//Serial.println("Unknown error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Unknown Error");
return p;
}
// OK converted!

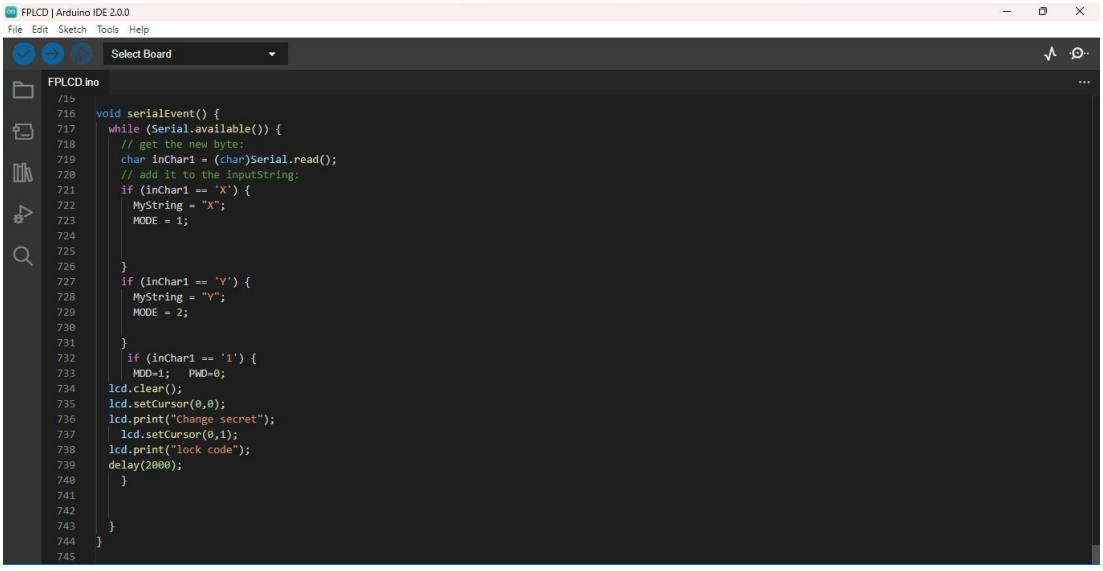

FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
p = finger.createModel();
if (p == FINGERPRINT_OK) {
// Serial.println("Prints matched!");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Matched..Success");
MODE = 0;
delay(3000);
} else if (p == FINGERPRINT_PACKETRECIEVEERR) {
// Serial.println("Communication error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
} else if (p == FINGERPRINT_ENROLLMISMATCH) {
// Serial.println("Fingerprints did not match");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Not Matched");
return p;
} else {
// Serial.println("Unknown error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
}
p = finger.storeModel(idx);


FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
p = finger.storeModel(idx);
if (p == FINGERPRINT_OK) {
//Serial.println("Stored!");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("ID ");
lcd.print(idx);
lcd.print(" Stored");
// Serial.print("X,");
// Serial.print(idx);
// Serial.println(",STORED,");
} else if (p == FINGERPRINT_PACKETRECIEVEERR) {
// Serial.println("Communication error");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
} else if (p == FINGERPRINT_BADLOCATION) {
// Serial.println("Could not store in that location");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");
return p;
} else if (p == FINGERPRINT_FLASHERR) {
// Serial.println("Error writing to flash");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Error");


FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
654 lcd.setCursor(0, 0);
655 lcd.print("Error");
656 return p;
657 } else {
658 // Serial.println("Unknown error");
659 lcd.clear();
660 lcd.setCursor(0, 0);
661 lcd.print("Error");
662 return p;
663 }
664
665 }
666
667
668
669
670 // returns -1 if failed, otherwise returns ID #
671 int getFingerprintIDez()
672 {
673
674 if (!finger.verifyPassword())
675 {
676 // DBG("Invalid!(");
677 lcd.setCursor(0, 0);
678 lcd.print("INVALID!!!");
679 return -1;
680 }
681
682 uint8_t p = finger.getImage();
683 if (p != FINGERPRINT_OK)


FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
681
682 uint8_t p = finger.getImage();
683 if (p != FINGERPRINT_OK)
684 {
685 return -1;
686 }
687
688 p = finger.image2Tz();
689 if (p != FINGERPRINT_OK)
690 {
691 return -1;
692 }
693
694 p = finger.fingerFastSearch();
695 if (p != FINGERPRINT_OK)
696 {
697 return -1;
698 }
699
700 #if __Debug
701 // Serial.print("Y,");
702 // Serial.print("*");
703 // Serial.print(finger.fingerID);
704 // Serial.println("#");
705 // Serial.print(",success,");
706 //DBG(finger.confidence);
707
708 #endif
709
710 return finger.fingerID;


FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
710 return finger.fingerID;
711
712 }
713
714 //------------------------------void serial event-----------------------------
715
716 void serialEvent() {
717 while (Serial.available()) {
718 // get the new byte:
719 char inChar1 = (char)Serial.read();
720 // add it to the inputString:
721 if (inChar1 == 'X') {
722 MyString = "X";
723 MODE = 1;
724
725
726 }
727 if (inChar1 == 'Y') {
728 MyString = "Y";
729 MODE = 2;
730
731 }
732 if (inChar1 == '1') {
733 MDD=1; PWD=0;
734 lcd.clear();
735 lcd.setCursor(0,0);
736 lcd.print("Change secret");
737 lcd.setCursor(0,1);
738 lcd.print("lock code");
739 delay(2000);
740 }


FPLCD | Arduino IDE 2.0.0
File Edit Sketch Tools Help
Select Board
FPLCD.ino
716 void serialEvent() {
717 while (Serial.available()) {
718 // get the new byte:
719 char inChar1 = (char)Serial.read();
720 // add it to the inputString:
721 if (inChar1 == 'X') {
722 MyString = "X";
723 MODE = 1;
724
725
726 }
727 if (inChar1 == 'Y') {
728 MyString = "Y";
729 MODE = 2;
730
731 }
732 if (inChar1 == '1') {
733 MDD=1; PWD=0;
734 lcd.clear();
735 lcd.setCursor(0,0);
736 lcd.print("Change secret");
737 lcd.setCursor(0,1);
738 lcd.print("lock code");
739 delay(2000);
740 }
741
742
743 }
744 }
745

CODING:

```
#include <Servo.h>
#include <Adafruit_Fingerprint.h>
//#include <Streaming.h>
#include <SoftwareSerial.h>
#include <Wire.h>  // Comes with Arduino IDE
#include <LiquidCrystal_I2C.h>


#define Buzz 11
#define SOL 8
```

```
Servo myservo;
```

```
int Gate=0;
int wait = 0;
int CF=0;
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
#define __Debug         1                               // if debug
mode
```

```
#if __Debug
#define DBG(X)          Serial.println(X)
#else
#define DBG(X)
#endif
```

```
String inputString = "";         // a string to hold incoming data
String MyString = "";
boolean stringComplete = false;  // whether the string is complete
int commaPosition;
int MODE=2;
int idx=0;
float Sound=0;
int CTT=0;
//-----------------
String PWORD="";
float freq=0;
float Sens2=0;
long PWD=50;
long PWDx=0;
int Startx=0;
float Piezo=0;
int MDD=0;
int ALM1=0;
int ALM2=0;
```

```
int Start=0;
```

```
int Mode=0;
float LDR=0;
int Timerx=0;
int CountTime=0;
```

```
int Alm=0;
float Strength=0;
int pos=0;
//----------------------
```

```
uint8_t getFingerprintEnroll(uint8_t id);
```

```
SoftwareSerial mySerial(2, 3);                    // rx, tx
```

```
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
```

```
void setup()
{
```

```
   myservo.attach(9);
pinMode(SOL,OUTPUT);
  digitalWrite(SOL,HIGH);
```

```
    pinMode(Buzz,OUTPUT);
    lcd.begin();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Initialize ");
    lcd.setCursor(0, 1);
    lcd.print("FPrint Sensor.. ");

    delay(2000);
```

```
    Serial.begin(9600);

    finger.begin(57600);
    delay(200);
while (CF==0){
  lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Initialize ");
    lcd.setCursor(0, 1);
    lcd.print("FPrint Sensor.. ");

    if (finger.verifyPassword())
    {
       // Serial.println("Fingerprint sensor init ok");
    lcd.setCursor(0, 0);
```

```
    lcd.print("Finger Print OK   ");
    CF=1;
    delay(2000);
    }
    else
    {
        lcd.setCursor(0, 0);
        lcd.print("Finger Print     ");
        lcd.setCursor(0, 0);
        lcd.print("not found..");
        delay(1000);
      //  Serial.println("Did not find fingerprint sensor :(");
      //  while (1);

    }
}
    lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("WELCOME..");
  lcd.setCursor(0,1);
  lcd.print("FP-DOOR ");
  delay(2000);
   lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("WELCOME");
  lcd.setCursor(0,1);
  lcd.print("............");
  delay(500);
  digitalWrite(Buzz,HIGH);
  delay(30);
    digitalWrite(Buzz,LOW);
  delay(30);
    digitalWrite(Buzz,HIGH);
  delay(30);
    digitalWrite(Buzz,LOW);
  delay(30);
  //lcd.noBacklight();
}
```

```
void loop()                     // run over and over again
{
if (MODE==3){
  lcd.noBacklight();
}
  float Reading=0;
while(MODE==3){

float Reading=0;
```

```
//-----------------------------------LDR ------------------------------
----------------------
```

```

```

```
if (Reading>2){

}
```

```
delay(10);
```

```
 //#####################################################
```

```
}

    //Serial.println("Type in the ID # you want to save this finger
as...\r\n(end with an 'a', such as 15a)");
    uint8_t id = 0;
   //-----------------------------WAIT COMMAND-------------------------
-----------------------------
   while (MODE == 0){
```

```
  ////////////////////////////////////////////////////////////////////
////////////////////////////
```

```

```

```
   if (stringComplete) {

    if (MyString == "X"){
     MODE = 1;
     delay(1000);
    }

    if (MyString == "Y"){

   MODE = 2;
    }
```

```
    if (MyString == "Z"){

   MODE = 3;
    }


    inputString = "";
    stringComplete = false;
}
   }
//--------------------------------------------------------------------------
--------------------
```

```
  if (MODE == 1){
  lcd.begin();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("[Enrolling] ");



    while (Serial.available())
    {
        char c = Serial.read();
    }


    while (true)
    {
        while (! Serial.available());
        char c = Serial.read();

        //----------------------------MODE 1-------------------------

        if (! isdigit(c))
        {
            while(Serial.available())
            {
                c = Serial.read();
```

```
            }
            break;
        }
        id *= 10;
        id += c - '0';

        //-------------------------------------------------------
    }
   }
    //-----------------------------------------ENROLLING-----------
-----------------------------
   // Serial.print("Enrolling ID #");
   // Serial.println(id);
   while (MODE == 1){
   // lcd.begin();
   // lcd.clear();
   // lcd.setCursor(0, 0);
   // lcd.print("Enrolling ID ");
    idx = idx - idx + id;
    lcd.print(id);
    delay(1000);
    while (!getFingerprintEnroll(idx));
    MODE = 0;
    delay(1000);
    Serial.flush();
   }
    //-------------------------------------------------------------
-----------------------------

    //-----------------------------------------------------VERIFY
PROCESS------------------------------
    while (MODE == 2){
    CTT++;
    if (CTT>100){
      MODE=3;
      CTT=0;
      lcd.noBacklight();
    }
       while (Serial.available()) {
    // get the new byte:
    char inChar = (char)Serial.read();
    // add it to the inputString:
    if (inChar != '\r') {
    inputString += inChar;
    }
    // if the incoming character is a newline, set a flag
    // so the main loop can do something about it:
    if (inChar == '\r') {
      stringComplete = true;
```

```
  }
  if (inChar == 'X') {
    MyString = "X";
    MODE = 1;
```

```
  }
  if (inChar == 'Y') {
    MyString = "Y";
    MODE = 2;

  }

}


/*
if (digitalRead(IR)==1){
delay(1000);
MODE = 0;
wait = 0;
}
*/
if(wait == 0){
  lcd.begin();
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("WELCOME..pls");
  lcd.setCursor(0, 1);
  lcd.print("scan finger..");
}
  if(getFingerprintIDez()>=0)
  {

      lcd.begin();
      lcd.clear();
      lcd.setCursor(0, 0);
```

```
      if (finger.fingerID==1){
        Serial.println("*SHANKARI#");

        lcd.print("SHANKARI");
        lcd.setCursor(0, 1);
        lcd.print("VERIFIED");
        digitalWrite(SOL,LOW);
     digitalWrite(Buzz,HIGH);
delay(30);
  digitalWrite(Buzz,LOW);
delay(30);
```

```
  digitalWrite(Buzz,HIGH);
delay(30);
  digitalWrite(Buzz,LOW);
delay(30);
myservo.write(90);
     delay(4000);
      myservo.write(0);
     digitalWrite(SOL,HIGH);
      }
      else if (finger.fingerID==2){
        lcd.print("NIROSH");
        lcd.setCursor(0, 1);
        lcd.print("VERIFIED");
         Serial.println("*NIROSH#");
        digitalWrite(SOL,LOW);
     digitalWrite(Buzz,HIGH);
delay(30);
  digitalWrite(Buzz,LOW);
delay(30);
  digitalWrite(Buzz,HIGH);
delay(30);
  digitalWrite(Buzz,LOW);
delay(30);
     myservo.write(90);
     delay(4000);
      myservo.write(0);
     digitalWrite(SOL,HIGH);
      }

        else {
        lcd.print("INVALID ID!!");
         digitalWrite(Buzz,HIGH);
         delay(2000);
         digitalWrite(Buzz,LOW);
         Serial.println("X");


      }
```

```
      lcd.begin();
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("WELCOME..pls");
  lcd.setCursor(0, 1);
  lcd.print("scan finger..");
     wait = 0;

  }
```

```
    delay(50);
    }

    //-----------------------------------------------------------------
-------------------------------




}
//-----------------------------------------------------END LOOP PROGRAM-
-------------------------
uint8_t getFingerprintEnroll(uint8_t id)
{
    uint8_t p = -1;
    //Serial.println("Waiting for valid finger to enroll");
    lcd.begin();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Place your finger..");
    delay(1000);
    while (p != FINGERPRINT_OK) {
        p = finger.getImage();
        switch (p) {
            case FINGERPRINT_OK:
            //Serial.println("Image taken");
            lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Done!..");
            delay(1000);
            break;
            case FINGERPRINT_NOFINGER:
            //Serial.println(".");
            break;
            case FINGERPRINT_PACKETRECIEVEERR:
            //Serial.println("Communication error");
            lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Communication");
            lcd.setCursor(0, 1);
            lcd.print("Error!");
            delay(1000);
            break;
            case FINGERPRINT_IMAGEFAIL:
            //Serial.println("Imaging error");
            lcd.begin();
            lcd.clear();
```

```
            lcd.setCursor(0, 0);
            lcd.print("Imaging Error");
            break;
            default:
            //Serial.println("Unknown error");
            lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Unknown Error");
            break;
        }
    }
```

```
    // OK success!
```

```
    p = finger.image2Tz(1);
    switch (p) {
        case FINGERPRINT_OK:
        //Serial.println("Image converted");
        lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Image converted!");
        break;
        case FINGERPRINT_IMAGEMESS:
        //Serial.println("Image too messy");
        lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Messy Image");
        return p;
        case FINGERPRINT_PACKETRECIEVEERR:
        //Serial.println("Communication error");
        lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
        case FINGERPRINT_FEATUREFAIL:
        //Serial.println("Could not find fingerprint features");
            lcd.begin();
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Your finger");
            lcd.setCursor(0, 1);
            lcd.print("not found!");
        return p;
        case FINGERPRINT_INVALIDIMAGE:
        //Serial.println("Could not find fingerprint features");
```

```
      lcd.begin();
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Your finger");
          lcd.setCursor(0, 1);
          lcd.print("not found!");
      return p;
      default:
      //Serial.println("Unknown error");
          lcd.begin();
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Unknown Error!");

      return p;
  }
```

```
 // Serial.println("Remove finger");
          lcd.begin();
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Remove finger");
  delay(2000);
  p = 0;
  while (p != FINGERPRINT_NOFINGER) {
      p = finger.getImage();
  }
```

```
  p = -1;
 // Serial.println("Place same finger again");
 lcd.begin();
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Place your");
          lcd.setCursor(0, 1);
          lcd.print("finger again..");
  while (p != FINGERPRINT_OK) {
      p = finger.getImage();
      switch (p) {
          case FINGERPRINT_OK:
          //Serial.println("Image taken");
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Image Taken");

          break;
          case FINGERPRINT_NOFINGER:
          //Serial.print(".");
          break;
```

```
            case FINGERPRINT_PACKETRECIEVEERR:
            //Serial.println("Communication error");
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
            break;
            case FINGERPRINT_IMAGEFAIL:
            //Serial.println("Imaging error");
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Image Error");
            break;
            default:
            //Serial.println("Unknown error");
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Unknown Error");
            break;
        }
    }
```

```
    // OK success!
```

```
    p = finger.image2Tz(2);
    switch (p) {
        case FINGERPRINT_OK:
        //Serial.println("Image converted");
        lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Image Converted");
        break;
        case FINGERPRINT_IMAGEMESS:
        //Serial.println("Image too messy");
        lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Messy Image");
        return p;
        case FINGERPRINT_PACKETRECIEVEERR:
        //Serial.println("Communication error");
        lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
        case FINGERPRINT_FEATUREFAIL:
        //Serial.println("Could not find fingerprint features");
        lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Not found");
        return p;
```

```
      case FINGERPRINT_INVALIDIMAGE:
     // Serial.println("Could not find fingerprint features");
     lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Not found");
      return p;
      default:
      //Serial.println("Unknown error");
      lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Unknown Error");
      return p;
  }
```

```
  // OK converted!
  p = finger.createModel();
  if (p == FINGERPRINT_OK) {
     // Serial.println("Prints matched!");
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Matched..Success");

           MODE = 0;
          delay(3000);
  } else if (p == FINGERPRINT_PACKETRECIEVEERR) {
     // Serial.println("Communication error");
       lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Error");
      return p;
  } else if (p == FINGERPRINT_ENROLLMISMATCH) {
     // Serial.println("Fingerprints did not match");
     lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Not Matched");
      return p;
  } else {
    //  Serial.println("Unknown error");
    lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Error");
      return p;
  }
```

```
  p = finger.storeModel(idx);
  if (p == FINGERPRINT_OK) {
      //Serial.println("Stored!");
      lcd.clear();
```

```
            lcd.setCursor(0, 0);
            lcd.print("ID ");
             lcd.print(idx);
            lcd.print(" Stored");
         //   Serial.print("X,");
         //   Serial.print(idx);
         //   Serial.println(",STORED,");

    } else if (p == FINGERPRINT_PACKETRECIEVEERR) {
       // Serial.println("Communication error");
       lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
    } else if (p == FINGERPRINT_BADLOCATION) {
       // Serial.println("Could not store in that location");
       lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
    } else if (p == FINGERPRINT_FLASHERR) {
       // Serial.println("Error writing to flash");
        lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
    } else {
       // Serial.println("Unknown error");
       lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Error");
        return p;
    }

}
```

```
// returns -1 if failed, otherwise returns ID #
int getFingerprintIDez()
{
```

```
    if (!finger.verifyPassword())
    {
       // DBG("Invalid!(");
       lcd.setCursor(0, 0);
            lcd.print("INVALID!!!");
        return -1;
    }
```

```
    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;
    }
```

```
#if __Debug
   // Serial.print("Y,");
 //  Serial.print("*");
 //    Serial.print(finger.fingerID);
 //    Serial.println("#");
   // Serial.print(",success,");
    //DBG(finger.confidence);

#endif
```

```
    return finger.fingerID;

}
```

```
//---------------------------------void serial event--------------------
-----------
```

```
void serialEvent() {
  while (Serial.available()) {
    // get the new byte:
    char inChar1 = (char)Serial.read();
    // add it to the inputString:
    if (inChar1 == 'X') {
      MyString = "X";
      MODE = 1;
```

```
    }
    if (inChar1 == 'Y') {
      MyString = "Y";
      MODE = 2;
```

```
    }
     if (inChar1 == '1') {
      MDD=1;   PWD=0;
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Change secret");
    lcd.setCursor(0,1);
  lcd.print("lock code");
  delay(2000);
    }


  }
}
```

APPENDIX B: PROJECT MANUAL/PRODUCT CATALOGUE

POLITEKNIK MALAYSIA
SULTAN SALAHUDDIN ABDUL AZIZ SHAH

BIOMETRIC FINGERPRINT SENSOR TO LOCK AND UNLOCK DOOR

Nama Pelajar : SHANKARI NAIR A/P MURALI

No. Pendaftaran : 08DEU20F2004

Nama Penyelia : PUAN NAAGAJOOTHI A/P ADIN NARAINA

BACKGROUND RESEARCH

The biometrics performance evaluation enables the developer to quickly understand the state of the available technology and expedite the commercialization process. Additionally, this creates a large market and shows users how reliable the product is. Because fingerprint recognition is the most distinctive biometrics technology in the sense that many types of fingerprint sensors are being marketed, algorithm developers as well as users require objective and quantitative evaluation techniques. The goal of this study is to suggest a performance evaluation procedure, evaluation metrics, and result presentation for fingerprint sensor modules. By contrasting the quantitative measure of fingerprint image quality impacted by environmental elements (temperature and humidity) and user characteristics, four commercial fingerprint sensor modules have been tested and appraised (skin humidity and pressure). The outcome of this study is anticipated to be used to improve algorithms that adapt to various fingerprint sensor modules.

PROBLEM STATEMENT

Due to the new technology that is being created every day to elevate people's standards of living, there is a major need for solutions that can make daily life for users simpler and improve home security. To be more specific, there are a number of drawbacks to the current manual door lock that bother a substantial section of the populace who haven't yet chosen to implement a secure system by using fingerprints to lock and unlock doors for their home. To make their daily jobs easier, people commonly rely on electrical and technological devices. Start with the first major present issue, which is that the house lock pads frequently have structural issues. Mechanical lock pad failure, internal component damage, or even lock pads that corrode with time are examples of significant defects. In addition, a lot of people lose their keys, making it impossible for them to access their own places.

IMPORTANCE PROJECT

OBJECTIVES OF PROJECT

- To improve the current fingerprint sensoring method more dependable and accessible to people of all ages and demographics.
- To analyze the outcomes in order to determine the accuracy and effectiveness of the fingerprint system.
- To create a sensoring system for the entrance in order to develop a system with superior security.

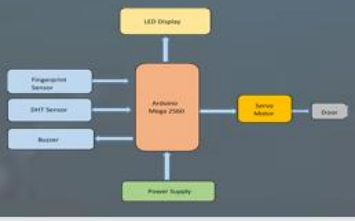
PROJECT'S PICTURE

people's standards of living, there is a major need for solutions that can make daily life for users simpler and improve home security. To be more specific, there are a number of drawbacks to the current manual door lock that bother a substantial section of the populace who haven't yet chosen to implement a secure system by using fingerprints to lock and unlock doors for their home. To make their daily jobs easier, people commonly rely on electrical and technological devices. Start with the first major present issue, which is that the house lock pads frequently have structural issues. Mechanical lock pad failure, internal component damage, or even lock pads that corrode with time are examples of significant defects. In addition, a lot of people lose their keys, making it impossible for them to access their own places.

IMPORTANCE PROJECT

The aim of this project is to create a basic, keyless lifestyle and improve home security. It also includes the system act as a manual.

BLOCK DIAGRAM



to people of all ages and demographics.
- To analyze the outcomes in order to determine the accuracy and effectiveness of the fingerprint system.
- To create a sensoring system for the entrance in order to develop a system with superior security.

PROJECT'S PICTURE