Design and Implementation Home Security Lock System Using Fingerprint

A final project report
presented to
the Faculty of Engineering

By

Muhammad Keanu Chaniago

002201400009

In partial fulfillment

Of the requirements of the degree

Bachelor of science in Electrical Engineering

President University

Jababeka Education Park, Jl. Ki Hajar Dewantara, Kota Jababeka, Cikarang Baru,

Bekasi 17550 – Indonesia

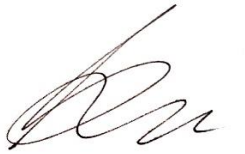
Phone (021) 8910 9762-6, Fax (021) 8910 9768

Email: enrollment@president.ac.id, http://www.president.ac.id

DECLARATION OF ORIGINALITY

I declare that this final project report, entitled "**Design and Implementation Home Security Lock System Using Fingerprint**" is my own original piece of work and, to the best of my knowledge and belief, has not been submitted, either in whole or in part, to another university to obtain a degree. All sources that are quoted are referred to are truly declared.

Cikarang, Indonesia, January 2020



Muhammad Keanu Chaniago

APPROVAL PAGE

Design and Implementation Home Security Lock System Using Fingerprint

BY

Muhammad Keanu Chaniago

002201400009

Approved by

Antonius Suhartomo, Ph.D
Final Project Supervisor

Antonius Suhartomo, Ph.D
Head of Study Program
Electrical Engineering

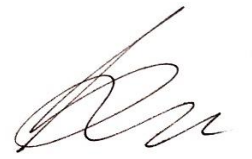
ACKNOWLEDMENTS

Foremost, I thank Allah SWT for all His grateful to my entire life to struggle and sacrifice for joy and sorrow of whole of my life until this day. I thank god for having such a good experience to study in President University from the beginning until finally I get my bachelor degree.
During my thesis, I would like to give my gratitude to those who surrounding me with infinite inclusion:

1. I would like to express my sincere gratitude to my final project supervisor Mr. Antonius Suhartomo, Ph.D., for the continuous support of my final project work, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this final project report.
2. My lovely family. My father M. Tamrin Caniago and mother Susi Agustianti, thank you for all of your supports and always be there for me no matter what. My little sisters, the most annoying sisters, my life wouldn't be complete without you guys.
3. Besides, my supervisor, I would like to thank all Electrical Engineering Lecturers: Dr. -Ing Erwin Sitompul, Joni Welman Simatupang, Ph.D, Ir. Adhi Wijaya, M.Sc., Arthur Silitonga, M.Sc., Carolus Kaswandi, Mia Galina, M.T., and of my university life in President University for their encouragement and extensive knowledge in Electrical Engineering.
4. My grateful thanks to all my Electrical Engineers, Kristi, Damanika, Taufan, Husna, Anastasia, Khoeruddin, Michael, Fauzi, Riano, Richard, Toby, William, and Alm. Yesua Andrew, who completed and helped my entire university life also involving your story to my life.
5. My beloved friends, Eliza Vania, Adi Asgaf, Rizky Rakhmadani, thank you for giving a colour to my university life and giving a wonderful journey that I never had before.
6. To my beloved girl, Kintan Umari, thank you for all of your supports that you gave to me, and always be there for me.
7. My skateboard group PAPANKERS, thank you for all of your supports and giving an enjoyable experience.

8. To my kostan kowandi friends, thank you for all of your supports and helps that you gave to me.

Cikarang, Indonesia, January 2020



Muhammad Keanu Chaniago

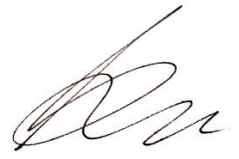
APPROVAL FOR SCIENTIFIC PUBLICATION

I hereby, for the purpose and development of science and technology, certify and approve to give President University a non-exclusive royalty-free right upon my final project report with the title.

HOME SECURITY SYSTEM WITH FINGERPRINT SCANNER PROJECT

Along with the related software or hardware prototype. With this non-exclusive royalty-free right, President University is entitled to conserve, to convert, to manage in database, to maintain, and to publish my final project report. These are be done with the obligation to mention my name as the copyright owner of my final project report.

Cikarang, Indonesia, January 2020



Muhammad Keanu Chaniago

ABSTRACT

Nowadays, fingerprint scanners usually used in mobile phones and access control absent machine in a work place. Fingerprint scanner it will work if the fingerprint is already registered in the memory. The fingerprint scanners will maintain our fingertip even though there's no power. It can only be accessed when a fingerprint user is registered. A home security with fingerprint sensors is solution that changing the regular based security systems into irregular systems. In this report will focuses on the use of home security system with fingerprint scanners to changing regular based security systems. The goal of home security lock system using fingerprint scanners is to protect what is inside.

Cikarang, Indonesia, January 2020

Muhammad Keanu Chaniago

TABLE OF CONTENT

LIST OF FIGURES AND TABLES

CHAPTER 1
INTRODUCTION

1.1 Background

Security systems with fingerprint sensor based also used in any platform such as, mobile phones, access control absent machine, etc. Fingerprints are the oldest and the most frequently used form for identification biometric based [1]. The utilization of fingerprint for distinguishing proof has been utilized in law implementation for about a decades [2]. Home security systems with fingerprint based is a procedure of checking the fingerprint pictures to unlock and lock.

This undertaking features the advancement of fingerprint identification. Verification is finished by looking at the information of approved fingerprint image with approaching unique fingerprint picture. At that point the data of incoming fingerprint image will experience the examination procedure to compare and approved fingerprint picture.

In this opportunity author would like to make a home security system for door access with fingerprint sensor based using Arduino Uno as the brain of the systems and using internet connection to let the systems make an alert notification through Telegram if there's someone want to hijacked it.

1.2 Problem Statement

Recently, there's a deadly incident in Pulomas, Jakarta. That took 6 lives [3]. Which is quiet terrifying if it happens around us. In order to prevent such an incident like that, home security system door lock with more secure and efficient is using fingerprint sensor based. Because, a complete data is needed from the owner of itself. It will prevent the possibility from the outside people to be able to access the door.

1.3 Objectives

The objectives of this final project are:

1. Design a home security system door lock with fingerprint sensor based in the purpose to prevent a crime.
2. Control the door access for authorized and unauthorized person.
3. The systems will give an alert notification through Telegram if there's authorized and unauthorized fingerprints.

1.4 Scopes and Limitations

Scopes and limitations on a portion of the issues used to maintain a strategic distance from deviations, for example, enlarging the principle issue with the goal that this examination is increasingly coordinated and encourages wetting so the exploration objectives will be obtained.

This final project is conducted under the following scopes:

1. The focus of this project is home security systems for door lock with fingerprint-based sensor.
2. This final project is aimed at helping people to make their home more safety.
3. This final project uses an Arduino Uno as the main controller.

In conducting this research, there are limitations to be considered:

1. The focus of this final project is only to detecting and recognizing the fingerprint for the purpose of door security system.
2. The information if the person is authorized or unauthorized be sent via telegram.

1.5 Outlines

The finals project consists of five chapters and is outlined as follows:

Chapter 1 – *Introduction*

This section will be a review of the general topic and the point of undertaking. It consists of: Final Project Background; Problem Statement; Final Project Objectives; Final Project Scopes and Limitations; and Final Project Outline.

Chapter 2 – *Literature Review & Research*

This section will talking about the theoretical viewpoints prompting the execution of the final project and the details required by the specific equipment of programming in order to be able to have the capacity to satisfy the prescribed requirements and objectives yet calculations of required component values

Chapter 3 – *Design Development and Implementations*

This section consists of model and the detailed description of the utilized techniques and the implementation of the described design, for example, schematic outline, block diagram, circuit outline, and programming codes.

Chapter 4 – *Results and Analysis*

This section explains how the device functions. It additionally introduces information gathered from a few trials, results of the project and discussion. Simulation results are examined to finally conclude the strengths and weakness of the proposed system. This chapter will be discussing further the results of Chapter III and discuss what actions taken from the analysis.

Chapter 5 – *Conclusions and Recommendations*

This section consists of conclusions obtained throughout the project implementation and suggestions for future improvement.

CHAPTER 2

LITERATURE REVIEW AND SPECIFICATION

2.1 Literature Review

There are so many technologies that provide Home security with Fingerprint based. To make an improvement and uniqueness of this project, the author takes 3 articles or projects that have the same concept for comparison.

2.1.1 First Project (Arduino Fingerprint Sensor)

The first project the home security systems using Arduino, fingerprint and OLED LCD display. This systems for only enroll and showing the name of the maker into OLED LCD display. [4]

2.1.2 Second Project (Fingerprint Based Security System)

In this second project is about home security systems using a microcontroller of 8051 families. In this project DC motor, Keypad, LCD Display and Buzzer. [5]

2.1.3 Third Project (Guide to Fingerprint Sensor Module with Arduino Uno)

In this project only wants to demonstrate the fingerprint sensors using Arduino Uno and using OLED display. [6]

So, in this project the author wants to develop this project more secured and efficient to access. Table 2.1 will show the comparison between author journal and other journals.

Table 2.1 Comparison Between Journals to Author Project.

	BOARD	ACTUATOR	LCD	BUZZER	INTERNET	KEYPAD
1ST PROJECT	ARDUINO UNO	NO	YES	NO	NO	NO
2ND PROJECT	8051 FAMILIES	DC MOTOR	YES	YES	NO	YES
3RD PROJECT	ARDUINO UNO	NO	YES	NO	NO	NO
THIS PROJECT	ARDUINO UNO AND NODEMCU	SOLENOID	YES	YES	YES	YES

2.2 SENSOR

2.2.1 Fingerprint Sensor

The Fingerprint sensor (Optical Scanners) that used in this project is FPM10A as shown in figure 2.1. This type of scanner is the oldest type because FPM10A relies on capturing an optical image. Nowadays, this type of fingerprint usually used attendance machine and for E-KTP registration.

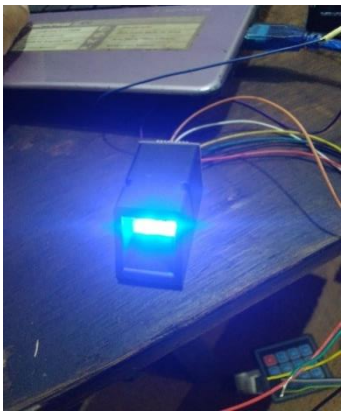


Figure 2.1 Fingerprint Sensor

The top layer is the place to put the fingertip or scan area. in the bottom of the area scan, there is a blitz that is utilized to enlighten the outside of the fingertip. Since the fingerprint is presented to light, it will create the reflections from the fingertips which are then caught by the recipient. After all of that process the data stored into memory.

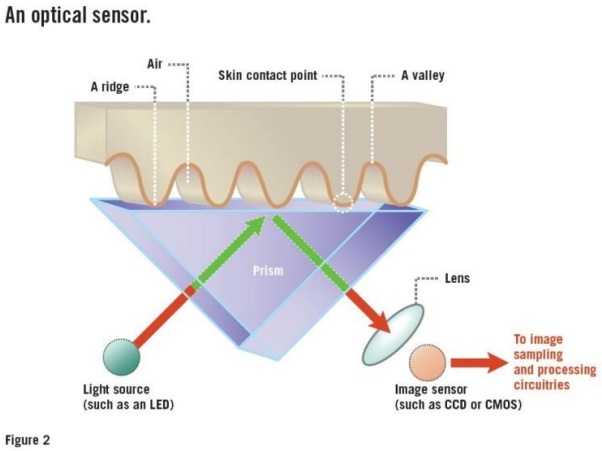


Figure 2.2 Fingerprint Progress
source: www.androidauthority.com

2.3 Microcontroller

2.3.1 Arduino Uno

Arduino Uno as shown in figure 2.3 is a board with an open source electronics platform. Arduino Uno provide 14 digital input/output pins, 6 can be used for PWM (Pulse width Modulation) as outputs), 6 analog input pins, a power barrel jack, ICSP header, reset button, and the last but not least USB connection for communicate to PC. This type of board is quite easy to find in store with affordable price.

This type of Arduino Uno has based datasheet from ATmega328P. On the Arduino AVR based board has EFPROM: memory whose data are kept when the board is turned off. This library empowers you to peruse and compose those bytes. The ATmega328P on the board have mount of EFPROM 1024 bytes.

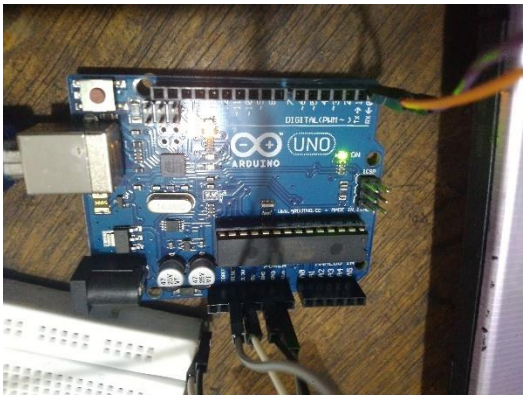


Figure 2.3 Arduino Uno

2.3.2 Arduino Uno Pins Configuration

There are so many varieties types of Arduino Uno, but most Arduino Uno have the same components in common.

a. Power (USB/Barrel Jack)

The Arduino Uno can be fueled from a USB link from your PC or power supply that is ended in a barrel jack but do not use a power supply that greater than 20 volts, it will destroy the board [7].

b. PINS (5V, 3.3V, GND, Analog, Digital, PWM, AREF)

- 5V & 3.3V: 5V pin will supplies 5 volts power, and 3.3V will supplies 3.3 volts of power [7].
- GND (Ground): There are a few GND sticks on the Arduino, any of which can be utilized to ground your circuit [7].

- Analog: The zone of pins under ‘Analog In’ mark (A0 until A5 on the board) are Analog In pins. These pins can read the sign from an Analog In for example ‘Temperature Sensor’ and convert into digital value that we can read [7].
- Digital/PWM: Opposite the Analog In there’s Digital pins (0 until 13 on the board). These pins can be utilized for Digital input and Digital output for example controlling a LED. And there’s a sign like (~) beside to some Digital pins (3, 5, 6, 9, 10, and 11, on the board). These pins go about as would as normal Digital pins, however can likewise be utilized for something called Pulse Width modulation (PWM) [7].
- AREF (Analog Reference): This type of pin we can leave it. It is sometimes used for an external reference voltage between 0 and 5 volts as far as possible for the Analog input pins [7].

c. Reset button

The Arduino Uno has a reset button if the code doesn’t work properly. When pushing the reset button, it will connect to the ground and reboot any code that is already loaded in the board [7].

d. LED Indicator

This LED should light up whatever point we plug Arduino Uno into a power source. This should be a sign that the board work properly or not, if not the LED wont light up or blinking [7].

e. TX (Transmit) & RX (Receive) LED

These markings show up a lot in gadgets to demonstrate the pins in a charge of sequential correspondence. For this situation, there are two places on the board where TX and RX show up, once by Digital pins at 0 & 1, and a second time alongside with TX and RX indicator LEDs. These LEDs will give some pleasant visual sign when Arduino Uno is accepting or transmitting a data [7].

f. Main IC (ATMEL)

The main IC on the Arduino Uno is marginally unique in relation from a board type to board type, yet is more often than not from the AT mega line of IC’s from ATMEL organization. This can be important, as you may need to realize the IC type (alongside to board type) before stacking up another program into the board. This data can be found at the top of IC itself [7].

g. Voltage Regulator

The voltage regulator does exist to control the measure of voltage that is let into the board. Consider it as a gatekeeper, it will dismiss an additional voltage that may damage the circuit. Obviously, it has breaking the points [7].

2.3.3 Arduino IDE

Arduino IDE (Integrated Development Environment) 1.8.9 as shown in figure 2.4is the software that author use for this final project.

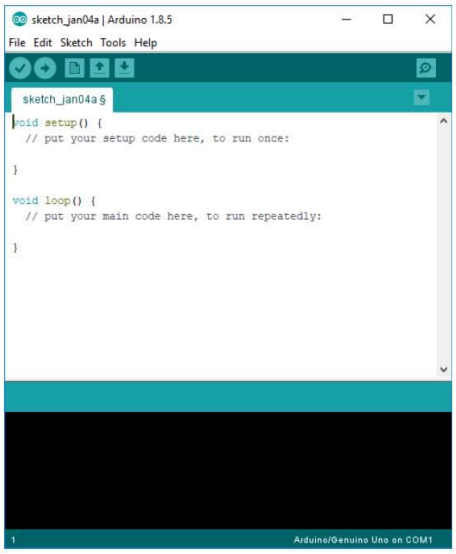


Figure 2.4 Arduino Uno IDE

Because the Arduino IDE is a cross-platform, this means it will run in every operating system. The Arduino IDE itself is a software that using a C-language program. This software can be used in any Arduino controller boards.

2.4 Modules

2.4.1 Buzzer

The Buzzer also known as Piezo Buzzer that shown in figure 2.5. This type of module will produce a sound based on the piezoelectric effect. These buzzers can be utilized caution a client on occasion comparing to an exchanging activity, counter sign or sensor input. They are additionally utilized as an alert circuit.

Figure 2.5 Buzzer

2.4.2 Solenoid Door Lock

The solenoid lock as shown in figure 2.6 can be used for electrical locking and unlocking. In most cases, the actual locking mechanism of a solenoid door lock will be identical to a standard locking system. The difference is the inclusion of a low-voltage solenoid in the mechanism, it will pull the latch back into the door when a controller is activated. These type of door locks usually used in automatic doors and remoted for security access.

Figure 2.6 Solenoid Door Lock 12V

The solenoid door lock, makes use of a static wire wound coil and a moving armature or plunger to perform the work of the key and linkages in conventional locks. The coil is wired into a circuit, which will include a low-voltage, direct current (DC) power source, and at least one control input.

2.4.3 LCD with I2C

The LCD stands for (Liquid Crystal Display) as shown in figure 2.7. The one of electronic made with logic rationale innovation that works by not delivering light but rather mirroring the light around it against from-lit or transmitting light from illuminated. LCD works as an information viewer in the form of characters, letters, numbers or graphic.

Figure 2.7 LCD

The I2C module as shown in figure 2.8 used in this project is mounted on the back of the LCD. This type of module has a potentiometer to adjust the LCD backlight and the I2C chip (PCF8574) itself. The advantages the LCD using this I2C module is that the wiring is simple. because the standards pin of the LCD is 16, with this I2C module the author only need 4 pins. That will cut all the pins that available on the LCD.

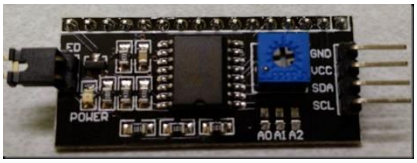


Figure 2.8 I2C

source: www.alselectro.wordpress.com

2.4.4 Telegram Application

Telegram as shown in figure 2.9 is a messaging application like the others applications that focus on speed, security. Can be used in any devices such as Android, iOS, Windows Phone, Windows MT, macOS, and Linux.

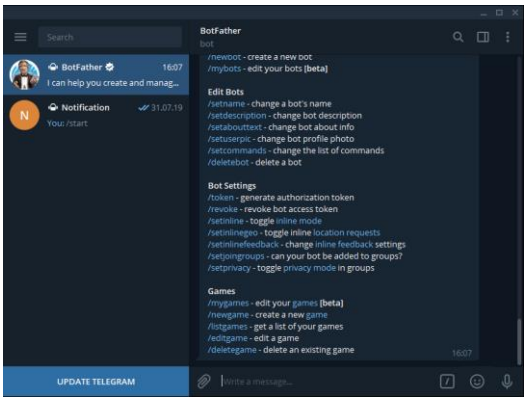


Figure 2.9 Telegram Application

2.4.5 Relay

Relay as shown in figure 2.10 is a switch which is worked electrically and mechanically, which comprises of two fundamental parts to be specific electromagnetic and mechanical. Rule of electromagnetic relays use to move the contact switch with the goal that the electric flow is small can deliver a higher voltage electricity. For an example, with relay that using an electromagnetic 5V and 50mA that able to move the Armature Relay (which functions as a switch) that can conduct 220V and 2A electricity [8].

Figure 2.10 Relay

2.4.6 NodeMCU (ESP82669)

NodeMCU as shown in figure 2.11 is a module that built-in ESP8266 or we can call it WI-FI module. The nodeMCU also has a featured like Arduino Uno.

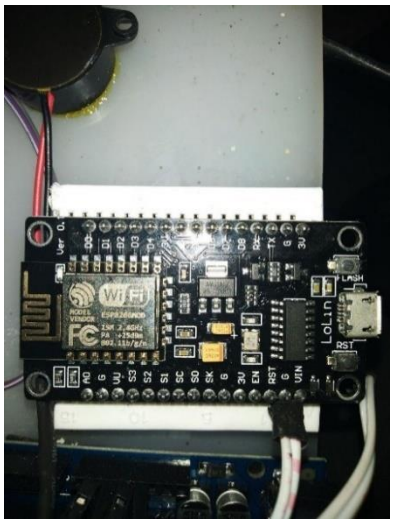


Figure 2.11 nodeMCU (ESP8266)

All the main pins and the function are the same as the Arduino Uno the differences between the Arduino is the nodeMCU has a cable type B/USB is the same as the smartphone that usually used and has the Wi-Fi module.

2.4.7 Step Down LM2596 Module

LM2596 stepdown as shown in figure 2.12 is a module having IC LM2596 as its main component. IC LM2596 is an integrated circuit/integrated circuit which serves as the Step-Down DC converters with 3A current rating. There are several variants of this series IC which can be grouped into two groups; the adjustable version of the output voltage can be adjusted. For an example, the input of electricity is from power supply 12V can be reduced into 5V.

Figure 2.12 Step Down
source: www.electronics-lab.com

CHAPTER 3

DESIGN DEVELOPMENT AND IMPLEMENTATION

3.1 Introductory Remarks

In this final project there are several requirements such as hardware implementation, software implementation, design of prototype, and wiring diagram. However, other required factors which are block diagram. Block diagram contain about the flow of the systems to describe the steps in.

So, in this block diagram figure 3.1 Arduino Uno as the brains source that control the systems such as, Fingerprint Module, LCD, Keypad, Solenoid Door Lock, nodeMCU, and Telegram App.

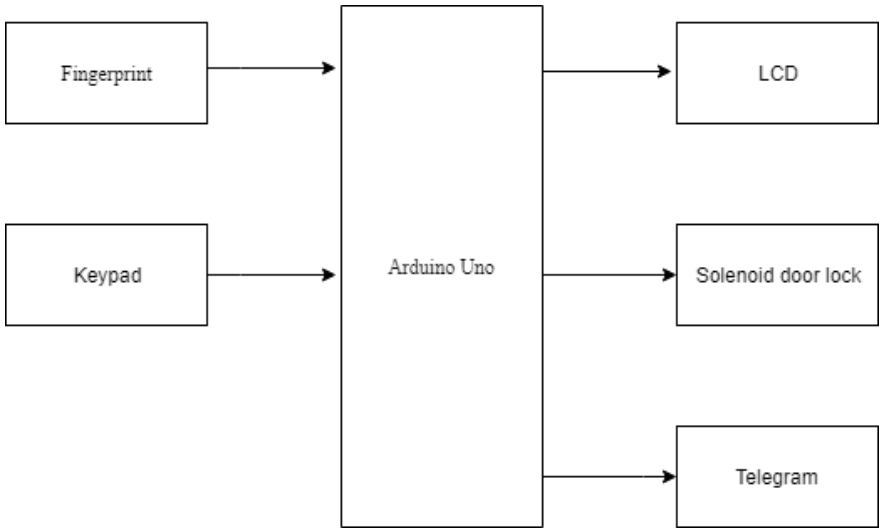


Figure 3.1 Block Diagram.

3.2 Hardware Implementation

3.2.1 Fingerprint Sensor Module

In this project the Fingerprint Sensor Module is used for detecting our fingertip that connected to Arduino Uno to recognize to unlock the door lock that using Solenoid. There are 4 pins that will connected to Arduino Uno and the pins shown in the table 3.1.

Table 3.1 Fingerprint Sensor Module Configuration.

Device	Pins			
Arduino Uno	GND	RX	TX	3.3V
FPM	GND	TX	RX	VCC

3.2.2 LCD

LCD is used for showing the access after tapping the fingertip into Fingerprint Sensor, the sensor will give the access or not. There are 6 pins that connected into the board shown in the table 3.2.

Table 3.2 LCD Configuration.

Device	Pins			
Arduino Uno	GND	5V	AREF	AREF
LCD i2c	GND	VCC	SDA	SCL

3.2.3 Keypad

Keypad is used for double security systems, after tapping the fingertip we have to insert the code through the keypad or it can be an alternative if the fingerprint is broken or it can be as double security check. There are 7 pins that connected into the board shown in the table 3.3.

Table 3.3 Keypad Configuration.

Device	Pins						
Arduino Uno	4	~5	~6	7	8	~9	~10
Keypad	C3	C2	C1	R3	R2	R1	R0

3.2.4 Solenoid Door Lock 12v

Solenoid door lock is used to unlocking the door and got the supply power from 12V DC. After fingerprint recognize the fingertip and input the password through the keypad. After all those steps, the solenoid will open.

3.2.5 NodeMCU (ESP8266)

In this project the NodeMCU (esp8266) is connected serially to Arduino Uno board. In this case, the author wants to utilize the Wi-Fi module that already built-in. There are 3 pins that connected into the board shown in the table 3.4.

Table 3.4 NodeMCU Configuration.

Device	Pins		
Arduino Uno	RX/0	TX/1	GND
NodeMCU	TX	RX	GND

3.3 Design of the Prototype

3.3.1 Circuit Diagram

In this part, shows about the connection of fingerprint sensor module. Each of modules are connected to the 5V and GND except the fingerprint sensor module, keypad. The fingerprint sensor module is connected to the 3.3V pin at the Arduino Uno. The keypad is connected to the analog pin number 4 – 10 of Arduino Uno.

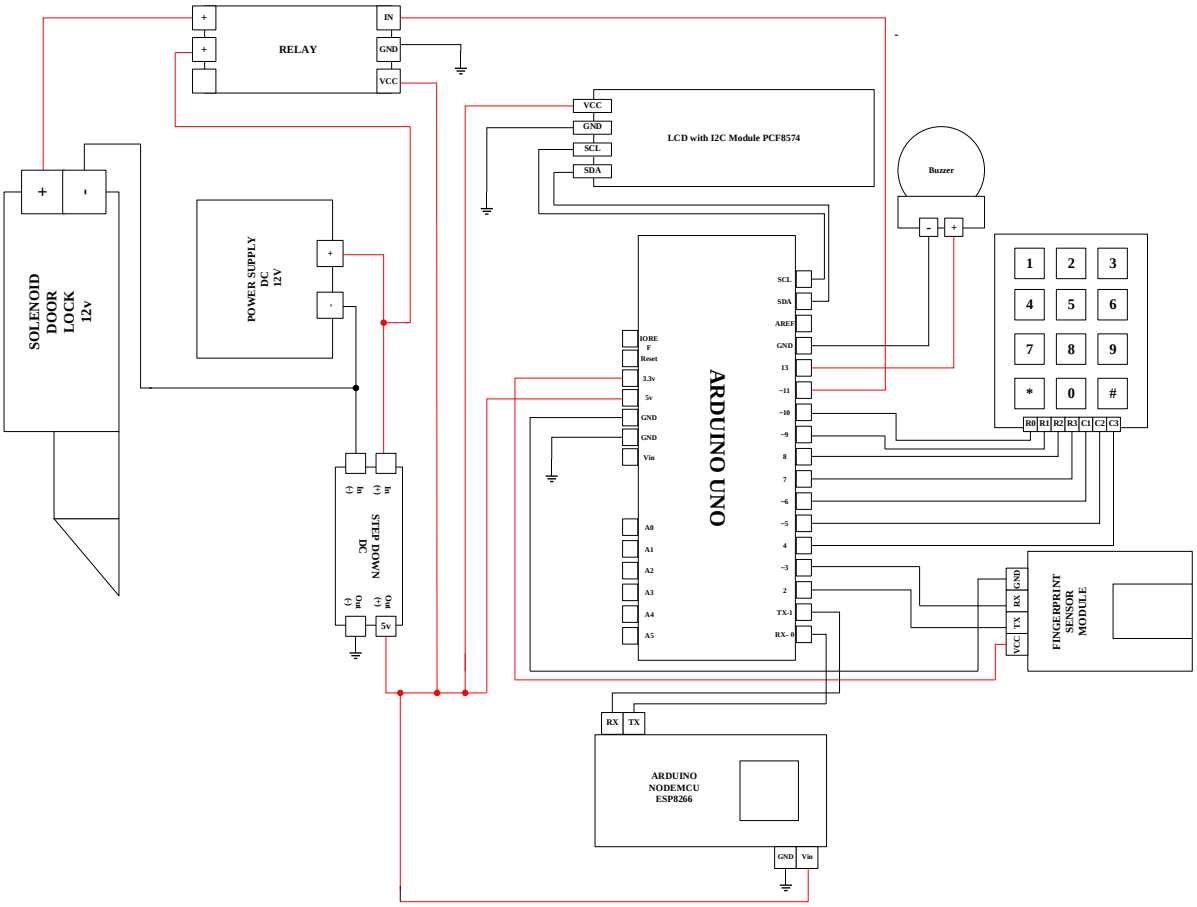


Figure 3.2 Design Circuit Diagram

The prototype the author makes for final project using cardboard or plastic box, and for the object to be scanned is fingertip. The prototype has fingerprint module, LCD, solenoid door lock and also an Arduino Uno. For a better understanding, the picture of the final project will be shown in figure 3.3.

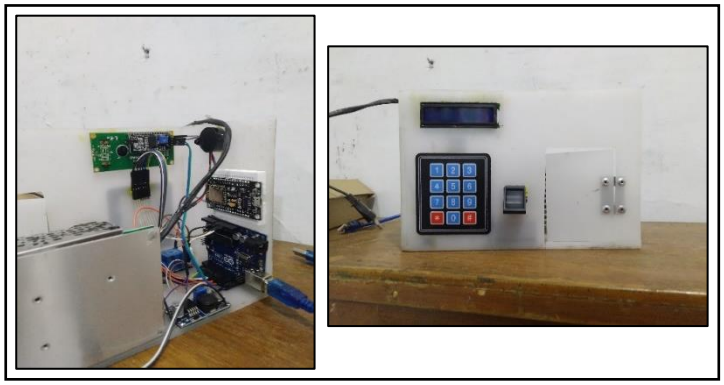


Figure 3.3 Design Prototype

3.4 Software Implementation

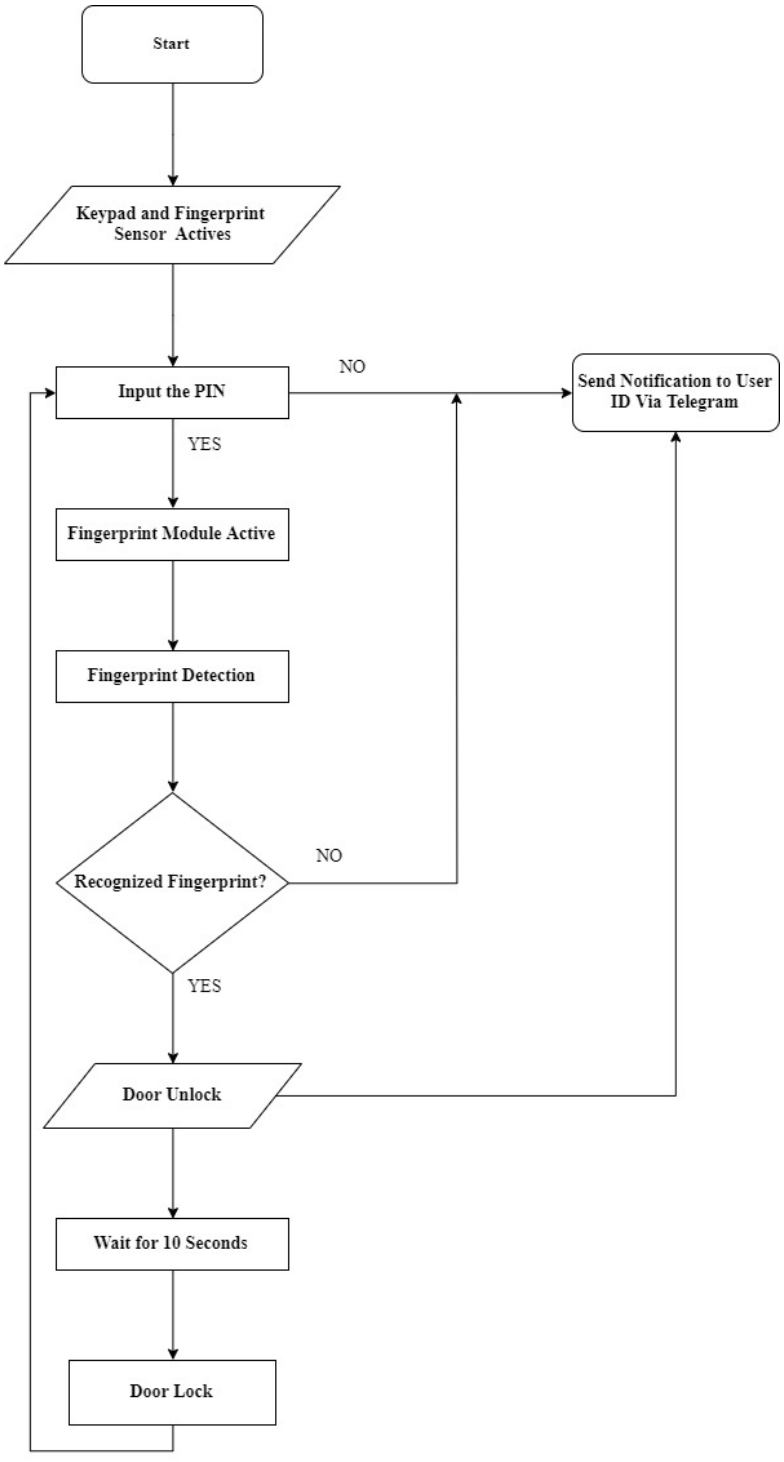


Figure 3.4 Flow Chart of the Home Security System for Door Access Using Fingerprint Scanners Project

This section explains the software of Arduino IDE 1.8.9 and Telegram app. Every program has 2 main parts. The first is the *setup ()* function, which describe the functions that will be executed once the first systems is running. The second is the *loop ()* function, which describe the function that will run the systems continuously.

3.4.1 Record the Fingerprint

First of all, the author has to enroll the fingerprint into fingerprint sensor module using this application.

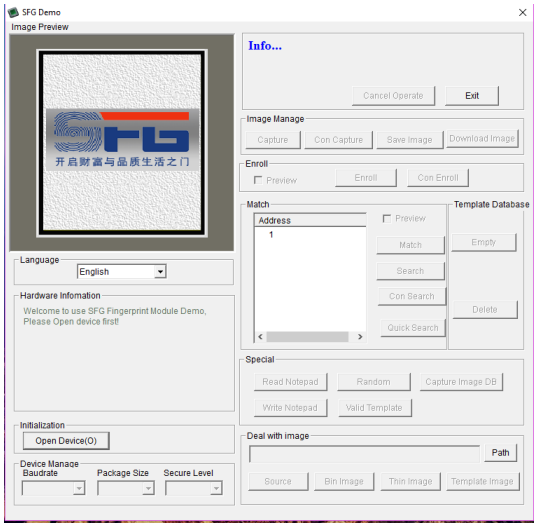


Figure 3.5 Application to Enroll the Fingerprint

If the fingerprint already enrolled into the fingerprint sensor module, make a code in the Arduino IDE.

Figure 3.6 Arduino IDE

And the final steps are check the fingerprint through the Arduino IDE serial monitor.

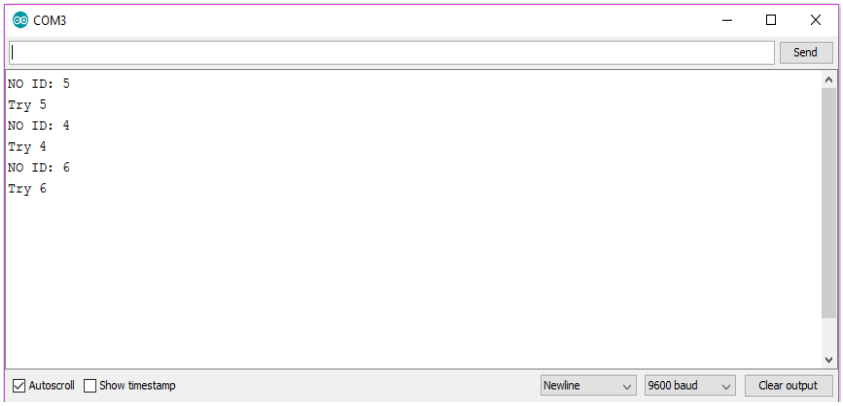


Figure 3.7 Fingerprint Checking using Arduino IDE

3.4.2 Integrated to Telegram.

First of all, the author has to make personal bot that already provided from the Telegram.

1. Search for the "botfather" telegram bot, this bot will assist the author to creating the personal bot.

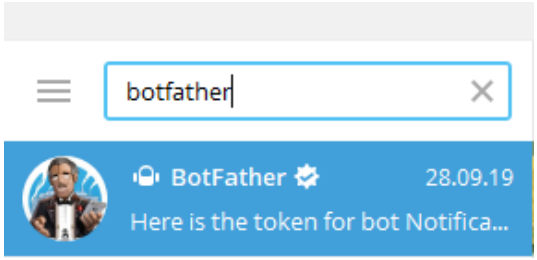


Figure 3.8 Searching botfather

This botfather will gave the author access for making a new personal bot for this project.

2. Then, type */help* or */start* to see all the commands that botfather provide.

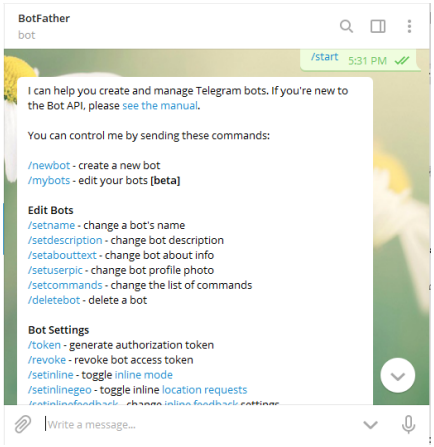


Figure 3.9 Get Started

3. Click on the */newbot* or type */newbot* to create the bot.

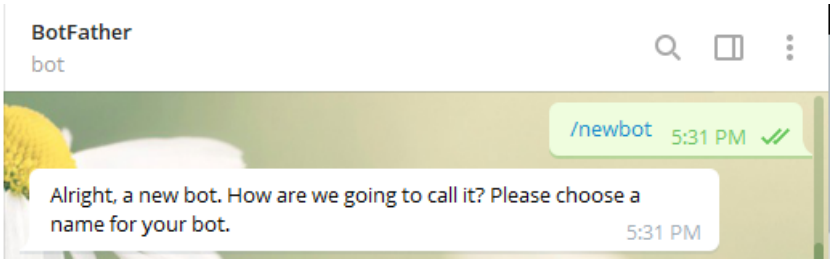


Figure 3.10 Get Started (cont.)

In this step, the author has to type the */newbot* to get the personal bot for his project.

4. After that, make a new name for the bot.

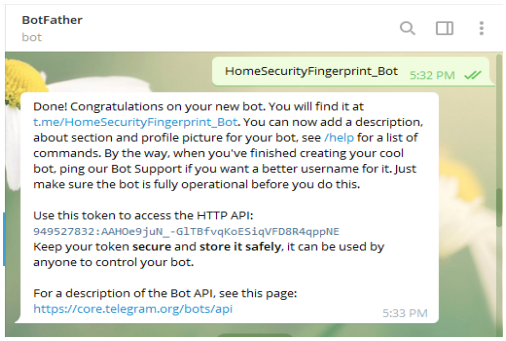


Figure 3.11 New Name for the BOT

So, the bot is created and named "**HomeSecurityFingerprint.Bot**". Then, the author should see the API token. This is the author API token that the application gave. **"949527832: AAHOe9juN_GlTBfvqKoESiqVFD8R4qpNE"**. This API token it will included later in the software Arduino IDE

5. Search IDBot then type */getid* to get the chat id from the Telegram.

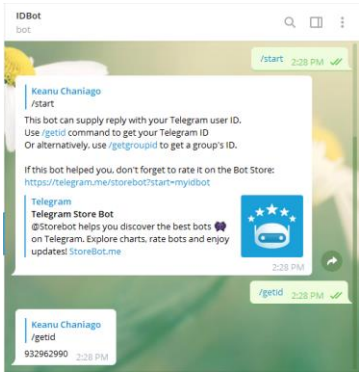


Figure 3.12 Get the ID number

The bot will give the author the personal chat ID "**932962990**"

6. After all those steps finished, input the bot API token and the Chat ID into the Arduino Uno code. Then upload it to the NodeMCU module.

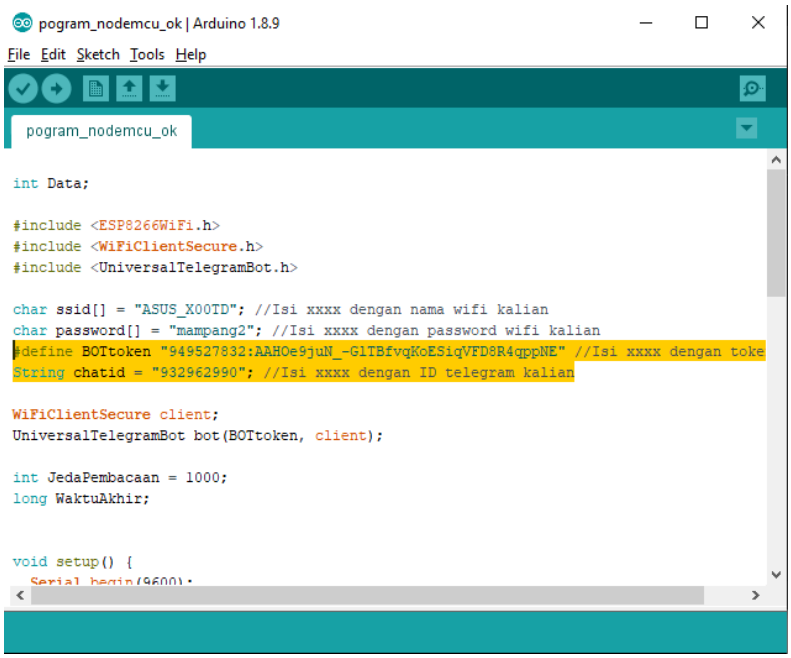


Figure 3.13 Input the data through the Arduino IDE

7. After input the personal author API token and the chat ID into the Arduino IDE, the Telegram will give the notification if the software and the machine are connected. Like the figure shown below.

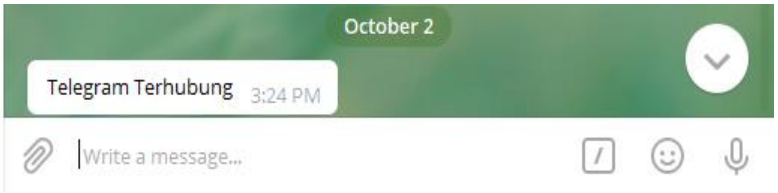


Figure 3.14 Telegram Connected to the Machine

CHAPTER 4
RESULTS AND DISCUSSIONS

4.1 Result and Discussions

This prototype is designed to lock and unlocking the doors. As the author stated in chapter 3, the fingerprint sensor will process the fingertip to access the door after input the password, and the solenoid will give the access if the password and fingertip is matched.

4.1.1 Fingerprint Sensor Testing

For the Fingerprint sensor testing, the author wants to make the fingerprint sensor can sensed fingertip. So, in this figure 4.1 it will tell us if the fingertip that we already enroll is matched into the Arduino IDE.

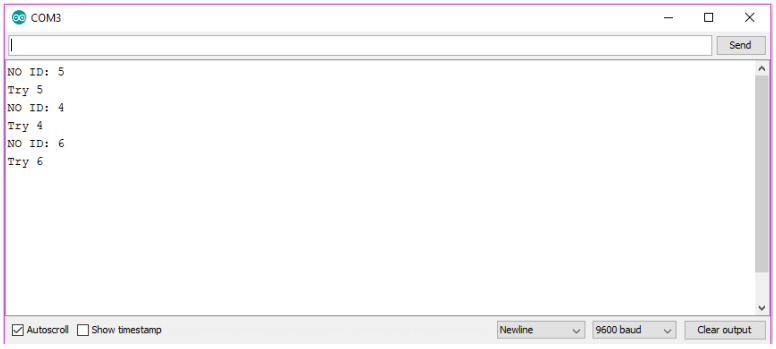


Figure 4.1 Arduino IDE Test

With this result, the author can sense the fingertip and matched into the ID number that author used.

4.1.2 Prototype Test

In this part, used all of the component which is already installed.

First of all, connect the machine to the laptop and upload the code that already made.

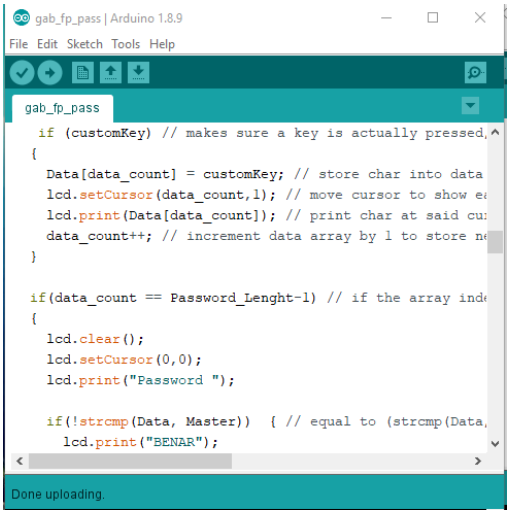


Figure 4.2 connected to Arduino IDE

Second, enter the password.

Figure 4.3 Prototype Testing Enter the Code

If the password is correct than you need to put your finger into the fingerprint sensor. The figure will be shown in figure 4.4.

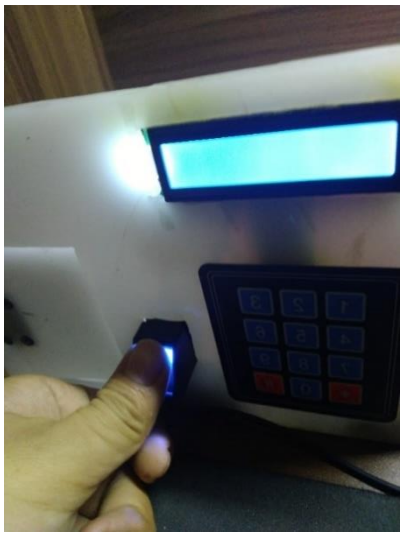


Figure 4.4 Prototype Testing Fingerprint Sensor.

After all those steps done, the door will give the access to open shown in the figure 4.5.

Figure 4.5 Prototype Testing Solenoid Door Lock.

4.1.3 Fingerprint Sensor System

So, in this project the author wants to performing experiments what sort of placement should make the fingerprint module can read it and which don't and will be shown below Table 4.1.

Table 4.1 Trial Results of Fingerprint Sensor system

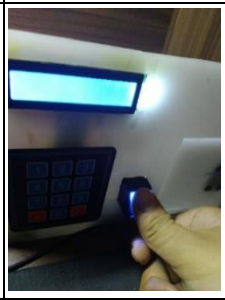
No	Real-time Processes	Database Image	Remarks	Percentage of successful (%)	Length of Time (s)
1		Hello, Thumb finger is here :).	Matched	10/10 = 100%	1.54s
2		Unknown.	Not Matched	0/10 = 0%	8s
3		Hello, Thumb finger is here :).	Matched	9/10 = 90%	1.93s

Table 4.1 Trial Result of Fingerprint Sensor Systems (cont.).

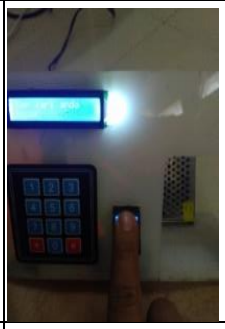
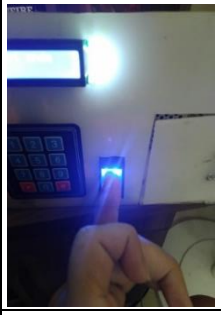
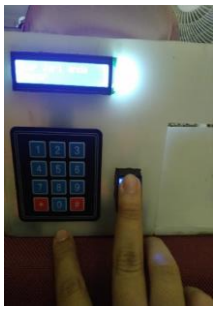
No	Real-time Processes	Database Image	Remarks	Percentage of successful (%)	Length of Time (s)
4		Hello, Index Finger is here :).	Matched	8/10 = 80%	3.66s
5		Hello, Index Finger is here :).	Matched	10/10 = 100%	1.70s
6		Unknown.	Not Matched	0/10= 0%	8s

Table 4.1 Trial Result of Fingerprint Sensor System (cont.).

No	Real-time Processes	Database Image	Remarks	Percentage of successful (%)	Length of Time (s)
7		Unknown.	Not Matched	0/10 = 0%	8s
8		Hello, Middle Finger is here :).	Matched	9/10 = 90%	1.49s
9		Hello, Middle Finger is here :).	Matched	10/10 = 100%	2.02s

4.2 Strengths and Weaknesses

4.3.1 The Strengths

In this final project, the strength that have been obtained are:

1. Matching method is successfully work.
2. The fingerprint sensor can read the fingertip when upside down.
3. Notification message is successfully sent to user Telegram.
4. Solenoid system is successfully work based on the result of fingerprint sensor.

4.3.2 The Weaknesses

The weakness found on this project are:

1. The machine needs to be connected to internet to get the notification to our personal ID Telegram.
2. The machine didn’t tell the name or ID through the LCD when someone that who’s try to get in.
3. Sometimes an error happens when connecting to the Telegram.

CHAPTER 5
CONCLUSIONS AND FUTURE DEVELOPMENTS

5.1 CONCLUSIONS

From the results and discussions of the entire system of this final project, there are several conclusions:

1. The system has successfully used to detecting and recognizing a fingerprint in a real-time.
2. Template matching method has successfully with accuracy around 98% applied to process the real-time image with database image.
3. The machine has successfully integrated to Telegram to monitor real-time condition of who try to access the door.
4. Door access has successfully controlled based on the fingerprint system results which set the solenoid to be open or close. So, only the authorized or enrolled person that can access the door.

5.2 RECOMMENDATIONS

There is no end point of learning and end point of improvement. There are several possibilities that can be improved from the results of this final project to obtain a better performance:

1. Use higher specification for the fingerprint sensor.
2. Use another higher specification for the machine such as the keypad, LCD, buzzer, etc., and combining another method like the face recognition.
3. Use a battery for the emergency power.

REFERENCES

[1] R. Triggs, “How fingerprint scanners work: optical, capacitive, and ultrasonic variants explained,” 2019. [Online]. Available: https://www.androidauthority.com/how-fingerprint-scanners-work-670934/. [Accessed: 16-Jul-2019].

[2] S. Goodner, “Finger Scanners: What They Are And Why They Are Gaining in Popularity,” 2019. [Online]. Available: https://www.lifewire.com/understanding-finger-scanners-4150464. [Accessed: 23-Nov-2019].

[3] et al Muhammad Taufiqqurrahman, “Geger Pembunuhan Sadis Sekeluarga: Dari Pulomas hingga Bekasi,” 2018. [Online]. Available: https://news.detik.com/berita/d-4300778/geger-pembunuhan-sadis-sekeluarga-dari-pulomas-hingga-bekasi?_ga=2.73151504.263987904.1582213251-1489426734.1557428272. [Accessed: 16-Nov-2019].

[4] N. Koumaris, “Arduino Fingerprint Sensor,” 2017. [Online]. Available: https://create.arduino.cc/projecthub/nickthegreek82/arduino-fingerprint-sensor-tutorial-103bb4?ref=tag&ref_id=security&offset=23. [Accessed: 10-Aug-2019].

[5] M. Technologies, “Fingerprint Based Security System.” [Online]. Available: https://www.projectsof8051.com/fingerprint-based-security-system/. [Accessed: 15-Sep-2019].

[6] S. Santos, “Guide to Fingerprint Sensor Module with Arduino (FPM10A),” 2018. [Online]. Available: https://randomnerdtutorials.com/fingerprint-sensor-module-with-arduino/. [Accessed: 15-Sep-2019].

[7] Farnell, “Arduino Uno Datasheet,” *Datasheets*, pp. 1–4, 2013.

[8] D. Kho, “Pengertian Relay dan Fungsinya,” 2018. [Online]. Available: https://teknikelektronika.com/pengertian-relay-fungsi-relay/. [Accessed: 24-Sep-2019].

APPENDIX

Code for the Arduino Uno

```
#include <Adafruit_Fingerprint.h>
 #include <SoftwareSerial.h>
 int getFingerprintIDez();
 SoftwareSerial mySerial(2, 3);
 Adafruit_Fingerprint finger =
Adafruit_Fingerprint(&mySerial);
#include <Keypad.h>
 #include <Wire.h>
 #include <LiquidCrystal_I2C.h>
 LiquidCrystal_I2C lcd(0x27,16,2);
 #define Password_Lenght 7                  // Give enough room for six chars + NULL char
 char Data[Password_Lenght];                // 6 is the number of chars it can hold + the null
                                            char = 7
 char Master[Password_Lenght] = "456123";
 byte data_count = 0, master_count = 0;
 bool Pass_is_good;
 char customKey;
 const byte ROWS = 4;
 const byte COLS = 3;
 char keys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}
 };
 byte rowPins[ROWS] = { 4,5,6,7};          //connect to the row pinouts of the keypad
 byte colPins[COLS] = { 8,9,10};           //connect to the column pinouts of the keypad
 Keypad customKeypad( makeKeymap(keys),
rowPins, colPins, ROWS, COLS);             //initialize an instance of class NewKeypad
 const int buzer = 13;
 const int kunci = 11;
 int mode = 0;
 int waktu = 0;
 void setup()   {
  Serial.begin(9600);
  mySerial.begin(9600);
  pinMode(buzer,OUTPUT);
  pinMode(kunci,OUTPUT);
  digitalWrite(kunci,HIGH);
  lcd.begin ();
  lcd.backlight();
   }
void loop () {
```

```
  if(mode==0) {     //pertama
    key_pad (); }
  if(mode==1) {     //ke dua
    finger_print();
    }
  }
void key_pad ()    {
  lcd.setCursor(0,0);
  lcd.print("enter Password");
  customKey = customKeypad.getKey();          // makes sure a key is actually pressed, equal to
   if (customKey)                             (customKey != NO_KEY)
  {
    Data[data_count] = customKey;             // store char into data array
    lcd.setCursor(data_count,1);              // move cursor to show each new char
    lcd.print(Data[data_count]);              // print char at said cursor
    data_count++;                             // increment data array by 1 to store new char, also
  }                                           keep track of the number of chars entered

  if(data_count == Password_Lenght-1)         // if the array index is equal to the number of
  {                                           expected chars, compare data to master
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Password ");

    if(!strcmp(Data, Master))  {              // equal to (strcmp(Data, Master) == 0)
      lcd.print("BENAR");
      alarm ();
      delay(2000);
      mode = 1;
   }
    else {
      lcd.print("SALAH");
      alarm ();
      alarm ();
      alarm ();
      delay(2000);                            // added 1 second delay to make sure the password
                                              is completely shown on screen before it gets
                                              cleared.
      Serial.println("z");
      mode = 0;
    }
      lcd.clear();
      clearData();
}
}
void clearData()
{
  while(data_count !=0)                       // This can be used for any array size
  {
    Data[data_count--] = 0;                   //clear array for new data
  }
```

```
  return;
}
int finger_print() {
    waktu = waktu + 1;                          //  Serial.println(waktu);
    if(waktu == 10) {
      alarm ();
      alarm ();
      alarm ();
      Serial.println("z");
      mode = 0;
      waktu = 0;}
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("tap jari anda ");
  uint8_t p = finger.getImage();
  p = finger.image2Tz();
  if (p != FINGERPRINT_OK)  return -1;
  p = finger.fingerFastSearch();
  if (p != FINGERPRINT_OK)  return -1;

if((finger.fingerID)==1) {                      //Serial.print("NO ID: ");
    lcd.setCursor(0,1);                          //Serial.println(finger.fingerID);
    lcd.print("benar ");
    Serial.println("a");
    alarm () ;
    buka_kunci ();
     mode = 0;
    lcd.clear();
 }
if((finger.fingerID)==2) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("b");
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
if((finger.fingerID)==3) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("c");
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
if((finger.fingerID)==4) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("d");
```

```
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
if((finger.fingerID)==5) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("e");
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
if((finger.fingerID)==6) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("f");
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
if((finger.fingerID)==7) {
    lcd.setCursor(0,1);
    lcd.print("benar ");
    Serial.println("g");
     alarm () ;
     buka_kunci ();
     mode = 0;
     lcd.clear();
 }
}
void alarm () {
  digitalWrite(buzer,HIGH);
  delay(200);
  digitalWrite(buzer,LOW);
  delay(200);
}
void buka_kunci () {
    digitalWrite(kunci,LOW);
    delay(3000);
     digitalWrite(kunci,HIGH);
    delay(1000);
}
```

Code for the esp8266 (NODEMCU)

```
int Data;

#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

char ssid[] = "ASUS_X00TD";                    //Isi xxxx dengan nama wifi kalian
char password[] = "mampang2";                  //Isi xxxx dengan password wifi kalian
#define BOTtoken "949527832:AAHOe9juN_-
GlTBfvqKoESiqVFD8R4qppNE"                      //Isi xxxx dengan token bot
String chatid = "932962990";                   //Isi xxxx dengan ID telegram kalian
WiFiClientSecure client;
UniversalTelegramBot bot(BOTtoken, client);

int JedaPembacaan = 1000;
long WaktuAkhir;

void setup() {
  Serial.begin(9600);

  WifiStatus();
  bot.sendMessage(chatid,"Telegram
Terhubung");
  Serial.println("Telegram Terhubung");
}
void loop() {
if (Serial.available()){
     Data=Serial.read();
     Serial.write(Data);

 if(Data=='z'){bot.sendMessage(chatid,"tidak
dikenal"); }
 if(Data=='a'){bot.sendMessage(chatid,"data 1");
}
 if(Data=='b'){bot.sendMessage(chatid,"data 2");
}
 if(Data=='c'){bot.sendMessage(chatid,"data 3");
}
 if(Data=='d'){bot.sendMessage(chatid,"Hello,
Thumb Finger is here :)"); }
 if(Data=='e'){bot.sendMessage(chatid,"Hello,
Index Finger is here :)"); }
 if(Data=='f'){bot.sendMessage(chatid,"Hello,
Middle Finger is here :)"); }
 if(Data=='g'){bot.sendMessage(chatid,"Hello,
Ring Finger is here :)"); }
    }
```

```
}
void WifiStatus() {
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(100);
  Serial.print("Connecting Wifi: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
}
```