PROTOTYPE LAMP CONTROLLER WITH ARDUINO
BASED ANDROID VIA Wi-Fi AND MOTION SENSOR

A final project report
presented to
the Faculty of Engineering

By
Anastasia Salsabila
002201400001

in partial fulfillment
of the requirements of the degree
Bachelor of Science in Electrical Engineering

President University
February 2018
DECLARATION OF ORIGINALITY

I declare that this final project report, entitled “Prototype Lamp Controller with Arduino base Android via Wi-Fi and Motion Sensor” 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 or referred to are truly declared.

Cikarang, Indonesia, February 2018

Anastasia Salsabila
PROTOTYPE LAMP CONTROLLER WITH ARDUINO
BASED ANDROID VIA Wi-Fi AND MOTION SENSOR

By
Anastasia Salsabila
002201400001

Approved by

Antonius Suhartomo, Ph.D.
Final Project Supervisor

Joni Welman Simatupang, Ph.D.
Head of Study Program
Electrical Engineering
ACKNOWLEDGEMENTS

In the name of Allah the Most Gracious, the Most Merciful. Alhamdullilah because His love and blessing I could reach this phase of my university life and complete my final project on time in order the requirement for undergraduate program at President University.

This thesis is dedicated to all people who always give me a lot motivation and support, especially to my parent, Alm. Bambang Sukaton and Entis Trisarani who always given me support, love and sincerity. My brother Abiyyu Asmaradana thank you for always being my reason to be a best sister and also so many motivation for me.

I would like to thank to Mr. Dr.-Ing Erwin Sitompul,ST.,M.Sc as Dean of Engineering, Mr Joni Welman Simatupang,Ph.D as head of Electrical Engineering study program, and also my final project advisor, Mr. Antonius Suhartomo, Ph.D. for advice, guidance, patience and support to do the best for this final project. I wish to thank all Electrical Engineering lectures for their guidance since I begin my university life.

Thank you so much to all my beloved Electrical Engineering 2014, Damanika Faskayana, Fauzi Ramadhan, Hamim Maulana, Husna Amiliansyah, Keanu Chaniago, Khoerudin, Kristi Mahardi, Michael Andrian, Riano Purba, Richard Agustanto, Setiawan Ginting, Taufan Adhi, Timothy Toby and William Zhuang who always teach me, support me during the class, thank you all see you on top.

Last but not least, I would like to thank all my beloved Badaiers, Ayu Chairunisya, Delvia Anggita, Elsa Christien, Firasmuda, Naomi Wiyoshi, Tania Agneta, Thania Gracia, Vero Oktaviani, and Shely who always being my every part of university life and also the support every single part of university life, thanks guys for everything. And also for my beloved friends Vanessa Azilina and Hanny NRA who always support me too. I love you all.

Cikarang, February 2018

Anastasia Salsabila
APPROVAL FOR SCIENTIFIC PUBLICATION

I hereby, for the purpose of 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:

PROTOTYPE LAMP CONTROLLER WITH ARDUINO BASED ANDROID VIA WIFI AND MOTION SENSOR

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 a database, to maintain, and to publish my final project report. These are to be done with the obligation from President University to mention my name as the copyright owner of my final project report.

Cikarang, February 2018

Anastasia Salsabila
002201400001
ABSTRACT

Nowadays, technology in the form of smartphone and microcomputer is an integral part in our live especially in this modern era. In this final project, a prototype of lamp control and lamp intensity control is proposed. Lamp can now be turned on, turned off, or dimmed, by using smartphone and human motion. The smartphone, connected to Wi-Fi, can control a lamp from a long distance. The intensity of the lamp can be controlled by human motion which is detected by using PIR (Passive Infrared Receiver). The finished prototype is able to control both lamp by using motion detected by PIR and smartphone via Android application.

Keywords: Controller Lamp, Android Smartphone, Wi-Fi Module, PIR (Passive Infrared Receiver) motion sensor
# TABLE OF CONTENT

DECLARATION OF ORIGINALITY ........................................................................... ii  
APPROVAL PAGE ................................................................................................... iii  
ACKNOWLEDGEMENTS ....................................................................................... iv  
APPROVAL FOR SCIENTIFIC PUBLICATION ................................................... v  
ABSTRACT ............................................................................................................. vi  
TABLE OF CONTENT ......................................................................................... vii  
LIST OF FIGURES ............................................................................................... ix  
LIST OF TABLES ................................................................................................... x  

## CHAPTER 1 INTRODUCTION ............................................................................ 1  
1.1. Final Project Background ............................................................................ 1  
1.2. Problem Statement ...................................................................................... 2  
1.3. Final Project Objectives ............................................................................. 2  
1.4. Final Project Scopes and Limitations .......................................................... 2  
1.5. Final Project Outline ................................................................................... 3  
  1.5.1. Chapter I – Introduction ...................................................................... 3  
  1.5.2. Chapter II – Design Specifications ...................................................... 3  
  1.5.3. Chapter III – Design Implementations ............................................... 3  
  1.5.4. Chapter IV – Result and Discussion ...................................................... 3  
  1.5.5. Chapter V – Conclusions and Recommendations ............................... 3  
  1.5.6. References ........................................................................................... 3  
  1.5.7. Appendices ............................................................................................ 3  

## CHAPTER 2 DESIGN SPECIFICATIONS ......................................................... 4  
2.1. Android ...................................................................................................... 4  
  2.1.1. Application (Reconnet) ..................................................................... 4  
  2.1.2. Wi-Fi Android .................................................................................. 5  
2.2. Sensors ...................................................................................................... 6  
  2.2.1. Motion Sensor .................................................................................. 6
2.2.2. PIR Motion Sensor (Passive Infrared Receiver) ........................................ 6
2.3. ESP8266 Wi-Fi Module ............................................................................. 7
2.4. Microcontroller ......................................................................................... 8
2.5. Arduino UNO IDE .................................................................................. 11
2.6. Relay Module 2 Channels ....................................................................... 12
2.7. Fitting Lamp ........................................................................................... 13
2.8. Plug – in .................................................................................................. 13
2.9. Dimmer .................................................................................................. 14

CHAPTER 3 DESIGN IMPLEMENTATIONS ......................................................... 15
3.1. Introductory Remarks ............................................................................. 15
3.2. Hardware Implementation ..................................................................... 18
   3.2.1. PIR Motion Sensor Implementation ...................................................... 18
   3.2.2. ESP8266 Wi-Fi Module Implementation ........................................... 19
3.3. Arduino, Relay, Dimmer, and Lamp Implementation .............................. 20
3.4. Software Implementations ..................................................................... 21
3.5. Hardware Design ................................................................................... 21
3.6. Electrical Design (wiring diagram) ....................................................... 21

CHAPTER 4 RESULTS AND DISCUSSIONS ...................................................... 22
4.1. Results and Discussions ........................................................................ 22
4.2. Experimental Result .............................................................................. 23
4.3. Experimental Method ............................................................................ 27
4.4. Strength and Weakness ......................................................................... 28

CHAPTER 5 CONCLUSIONS AND FUTURE DEVELOPMENTS ...................... 29
5.1. Conclusions ........................................................................................... 29
5.2. Recommendations and Future Developments ........................................ 29

REFERENCES .......................................................................................... 30

APPENDIX A PROGRAM CODE .................................................................. 31
APPENDIX B WIRING DIAGRAM ................................................................. 38
LIST OF FIGURES

Figure 2.1 Reconnet application ................................................................. 4
Figure 2.2 Android application relay control interface ............................... 5
Figure 2.3 Wi-Fi android setting ................................................................. 5
Figure 2.4 PIR motion sensor [2] .................................................................. 7
Figure 2.5 How PIR works [2] ...................................................................... 7
Figure 2.6 Pin configuration of PIR sensor [2] .............................................. 7
Figure 2.7 ESP8266 module [3] ..................................................................... 8
Figure 2.8 Arduino UNO [4] ......................................................................... 9
Figure 2.9 Arduino UNO pin configuration [6] ............................................. 11
Figure 2.10 Arduino UNO IDE interface ....................................................... 12
Figure 2.11 2 Channels relay module [7] ....................................................... 12
Figure 2.12 Fitting lamp .............................................................................. 13
Figure 2.13 Plug-in ....................................................................................... 13
Figure 2.14 Dimmer ....................................................................................... 14
Figure 3.1 Block diagram of the installation system ..................................... 15
Figure 3.2 Overall flow chart of the installation .......................................... 17
Figure 3.3 PIR, Arduino uno, and relay implementation .............................. 19
Figure 3.4 ESP8266 implementation between Arduino and ESP8266 ............ 19
Figure 3.5 ESP8266 implementation between Arduino with 2x2 relay 5V 2 channels ..... 20
Figure 3.6 Arduino, relay, dimmer, and lamp implementation ....................... 20
Figure 3.7 Prototype lamp controller design .............................................. 21
Figure 4.1 Prototype lamp controller ........................................................... 22
Figure 4.2 Lamp with 25% intensity ............................................................. 24
Figure 4.3 Lamp with 50% intensity ............................................................. 24
Figure 4.4 Lamp with 100% intensity ........................................................... 25
Figure 4.5 Lamp off with PIR ....................................................................... 25
Figure 4.6 Lamp 2 on ................................................................................. 26
Figure 4.7 Lamp 2 off .................................................................................. 26
LIST OF TABLES

Table 2.1 Pin Function of PIR Sensor ................................................................. 7
Table 2.2 Specification of ESP8266 Wi-Fi Module.............................................. 8
Table 2.3 Arduino Uno Specification................................................................. 9
Table 3.1 List of Relay and Command............................................................... 18
Table 3.2 Pin Configuration of PIR Sensor ....................................................... 18
Table 3.3 Pin Configuration between Arduino and ESP8266............................. 19
Table 3.4 Pin Configuration between Arduino UNO and 2x2 Relay 5V 2 Channels .... 20
Table 3.5 Pin Configuration between Arduino UNO, Relay 5V with 2x2 2 Channels,
Dimmer and Lamp......................................................................................... 21
Table 4.1 Table of Experiment Result of Lamp Controller..................................... 23
CHAPTER 1
INTRODUCTION

1.1. Final Project Background

Technology is an integral part in our live. So many improvements very fast following the time and space. With technology, our life becomes very easy and useful. In this era of globalization we know that every people have gadget on their fingertips. So many gadget that we can choose to improve and help our life. Gadget is one of the technologies that happening until today, without gadget I think every people can’t life without it.

In this modern era, we know that there are so many type of gadget. Android is the one of example of gadget. Android is very familiar for every people, because android is very easy to use and adaptable for every level of generation. So many applications in android which can help our activity in the real life, like messaging, chatting, etc.

In this final project, the author wants to use the feature of the android itself, which is the Wi-Fi. The author will modify the android so that the android can control the lamp. With android the author hope that the costumer can control the lamp without switch anymore, we can control all with our phone. Either to turn on or turn off the lamp.

For some excuse such the phone is at low battery or when we are in the basement area and signal is not available, we can use the motion sensor. Motion sensor is different with the android controller. One of the problem of controlling the lamp is about the intensity of the lamp, sometimes people wants to control the lamp with intensity that they wants. With motion sensor the author hope that the user not only able to switch the lamp on or off , but also could control the brightness intensity with their motion as they wish. Both of them control by Arduino Uno.
1.2. Problem Statement

Based on the final project background, the problem statement to reach the result of this final project are:

- How the author could take an advantage of a nearby Wi-Fi connection to control a lamp with their smartphone?
- How to change intensity and control a lamp with our motion?

1.3. Final Project Objectives

The objectives of this project are:

1. To implement the application on android smartphone in order to control a lamp with ESP8266 module Wi-Fi connection.
2. To implement intensity controller of a lamp by PIR (Passive Infrared Receiver) motion sensor.

1.4. Final Project Scopes and Limitations

In doing this final project, there will be some exact scope and limitation. Because of the time and resource are limited, the final project will be conducted under the following scopes:

- The final project will discuss about controller lamp with Android and motion sensor.
- The motion sensor used for control the lamp is PIR (Passive Infrared Receiver).
- Concept of Motion Sensor PIR (Passive Infrared Receiver).
- The controller used is Arduino UNO.
- Serial monitor display on Android.

In conducting this project, there are several limitations to be considered:

- The PIR sensor is used only to detect the presence of motion.
- The ESP8266 is used to read the data from Wi-Fi that receive signal from Android for further process. If the data already can be read by ESP8266 the module that data will has been stored in Arduino, the lamp will be ON/OFF.
The range of PIR sensor for reader the motion only up to 5 meter, very effective for motion detector, and also very sensitive with any motion surrounding.

The ESP8266 only can read for higher frequency of the Wi-Fi, if the quota of internet strong and the signal strong the controller with Android will be strong enough.

1.5. Final Project Outline

The final project report consists of six chapters and is outlined as follow:

1.5.1. Chapter I – Introduction

Chapter 1: Introduction. This chapter consists of Final Project Background, Problem Statement, Final Project Objective, Final Project Scope, and Limitation, and Final Project Outline.

1.5.2. Chapter II – Design Specifications

Chapter 2: Design Specification. The supporting theory that be required for make the controller lamp with android via Wi-Fi, the other of motion sensor. This chapter will include the explanation about Wi-Fi Android, motion sensor, and Arduino.

1.5.3. Chapter III – Design Implementations

Chapter 3: Design Implementation. It is the description of Arduino itself, PIR Motion Sensor, Relay Module, Wi-Fi Module, and Reconnet Software Arduino.

1.5.4. Chapter IV – Result and Discussion

Chapter 4: Experiment Result and Discussion. It will be the discussion result of the experiment itself, and show the software monitor in the android.

1.5.5. Chapter V – Conclusions and Recommendations

Chapter 5: Conclusion and Recommendations. This chapter consists of conclusion obtained throughout the project implementation and recommendation for future development.

1.5.6. References

1.5.7. Appendices
CHAPTER 2
DESIGN SPECIFICATIONS

2.1. Android

Android is a mobile operating system developed by Google. Usually, we use Android to communicate with each other. Linux kernel and other open source software are designed primarily for touchscreen mobile devices such as smartphone and tablet. With android (smartphone) we can use it everything that we need, not only for telephone or sending message but with Android we can use for sending the data, email, or controlling everything with Android. We know that Android has so many applications available in PlayStore, with PlayStore we can download everything that we need. In this project we can use application that can compare with the module for my project. The application is “Reconnet”.

2.1.1. Application (Reconnet)

Reconnet is an application to control the relay with internet. With this application, we can control the relay via the internet (anywhere in the world). This is one application of Internet of Things. This application can control so many relay with switch allows you to control a relay which is connected to the Arduino and Wi-Fi module ESP8266 from anywhere (as long as it is connected to the internet).

Figure 2.1 Reconnet application
2.1.2. Wi-Fi Android

Wi-Fi is a technology for wireless local area networking with device based on the IEEE 802.11 standard. A Wi-Fi connection is established using wireless adapter to create hotspot. Area in the vicinity of wireless routers which is connected to the network and allow users to access internet service. Wi-Fi provides wireless connectivity to our device like Android (Smartphone) by emitting frequency between 2.4GHz-5GHz, based on the amount of data network. Every Wi-Fi has an IP address, IP address is a long numbers or hexadecimals (number and letter) used identify our business computer or phone when you’re online. In addition to our computer’s IP address, there is also a separates IP address assigned to our wireless router. [1]
2.2. Sensors

2.2.1. Motion Sensor
This project is not only used an Android as a controller but also it can be controlled by motion sensor. Motion sensor is a device which detects physical movement on a device or within the environment. It can detect and capture physical or kinetic movement in real time. In this project the author use the motion sensor as the second choices to control the lamp without the Android. The motion sensor can detect the motion of humans or some device. In this problem the motion sensor can detect the hand motion.

2.2.2. PIR Motion Sensor (Passive Infrared Receiver)
The sensor motion which is used in this project is the passive infrared receiver (PIR). A passive infrared sensor, as shown in Figure 2.3 below, is an electrical component which measures the level of infrared light radiating for all objects in its field view. A PIR based motion detector is used to sense movement of people, animals or other object. They are small, inexpensive, low-power, easy to use and don’t wear out. PIR’s are basically made of a pyroelectric sensor witch can detect levels of infrared radiation. The sensor in a motion detector is actually split in two halves. The PIR sensor itself has two slots in it, each slot is made of a special material that is sensitive to IR. The lens used here is not really doing much and so we see that the two slots can 'see' out past some distance (basically the sensitivity of the sensor). When the sensor is idle, both slots detect the same amount of IR, the ambient amount radiated from the room or walls or outdoors. When a warm body like a human or animal passes by, it first intercepts one half of the PIR sensor, which causes a positive differential change between the two halves.[2] When the warm body leaves the sensing area, the reverse happens, whereby the sensor generates a negative differential change. These change pulses are what is detected. The works of the PIR sensor can be seen in Figure 2.4 below. The pin configuration and function of PIR sensor pin can be seen in Figure 2.5 and Table 2 below.
2.3. ESP8266 Wi-Fi Module

ESP8266 is a Wi-Fi module that serves as a microcontroller enhancement such as Arduino to connect directly with Wi-Fi and help connect TCP / IP. The power of this module is about 3.3v and has three Wi-Fi modes: station, access point and both. In this module there is also a processor, memory and GPIO where the number of pins depends on the type of ESP8266. I still use the microcontroller Arduino Uno. In this project works well as a
transmitter and receiver. It will send signals from mobile device to Arduino Uno. These module has the specification can be seen in Table 2.2 below. Disadvantages of this module are there’s not the sensor in this module therefore the author use motion sensor for the second problem. Figure 2.6 below shows the ESP8266 module.

Figure 2.7 ESP8266 module [3]

Table 2.2 Specification of ESP8266 Wi-Fi Module

<table>
<thead>
<tr>
<th>Specification</th>
<th>ESP8266</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCU</td>
<td>Xtensa Single-Core 32-bit L106</td>
</tr>
<tr>
<td>802.11 b/g/n Wi-Fi</td>
<td>Yes, HT20</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>N/A</td>
</tr>
<tr>
<td>Typical Frequency</td>
<td>80 MHz</td>
</tr>
<tr>
<td>SRAM</td>
<td>160 kBytes</td>
</tr>
<tr>
<td>Flash</td>
<td>SPI Flash up to 16 Mbytes</td>
</tr>
<tr>
<td>GPIO</td>
<td>17</td>
</tr>
<tr>
<td>Hardware/Software PWM</td>
<td>None / 8 Channels</td>
</tr>
<tr>
<td>SPI/I2C/I2S/UART</td>
<td>2/1/2/2</td>
</tr>
<tr>
<td>ADC</td>
<td>10-bit</td>
</tr>
<tr>
<td>CAN</td>
<td>N/A</td>
</tr>
<tr>
<td>Ethernet MAC Interface</td>
<td>N/A</td>
</tr>
<tr>
<td>Touch Sensor</td>
<td>N/A</td>
</tr>
<tr>
<td>Temperature Sensor</td>
<td>N/A</td>
</tr>
<tr>
<td>Working Temperature</td>
<td>-40° C - 125° C</td>
</tr>
</tbody>
</table>

2.4. Microcontroller

2.4.1 Arduino UNO R3

This final project is using an Arduino UNO R3 microcontroller. The reason of using this microcontroller is because it has enough amounts of pins necessary for building this project, and this microcontroller is more familiar than other microcontroller, and also this microcontroller easy to find in market both online or offline. The Figure of Arduino UNO
R3 is shown in Figure 2.7 below. The table of specification of Arduino UNO R3 is shown in Table 2.3.[4]

![Arduino UNO](image)

**Figure 2.8 Arduino UNO [4]**

<table>
<thead>
<tr>
<th>Arduino Uno R3</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating voltage</td>
<td>5V</td>
</tr>
<tr>
<td>Input Voltage (recommendation)</td>
<td>7-12V</td>
</tr>
<tr>
<td>Input Voltage (limits)</td>
<td>6-20V</td>
</tr>
<tr>
<td>Digital I/O Pins</td>
<td>14 (Which 6 provide PWM output)</td>
</tr>
<tr>
<td>Analog Input Pins</td>
<td>6</td>
</tr>
<tr>
<td>DC Current per I/O Pin</td>
<td>40 mA</td>
</tr>
<tr>
<td>DC Current for 3.3 V Pin</td>
<td>50 mA</td>
</tr>
</tbody>
</table>

### 2.4.2 Arduino UNO R3 Pin Configuration

Arduino UNO has enough pins for this project that why I use Arduino UNO in this project. Arduino UNO has 14 pins input form digital output where 6 pin can be used as PWM (Pulse Width Modulator) output. Here are the explanation and the descriptions of the Arduino UNO pin configuration:[5]

#### a. Power Pin

Arduino UNO can be supplied via USB connection or with an external power supply. The external supply (non USB) can be obtained from an AC adapter to DC or battery. Arduino UNO R3 has 4 power pins as follows:

- **VIN (Voltage In) Pin**: This is the pin to supply power to the Arduino board the use an external power source (not a 5 volt USB connection or any other source).
- **5V Pin**: This pin is a 5V set from the regulator on the board. It is pin which gives 5V output voltage.
- **3.3 V Pin**: This pin is the pin that supplies 3.3 volts generated by the regulator on board. It is the pin which gives 3.3 volt output voltage, and the maximum current that can be passed is 50 mA.
• GND (Ground) Pin: There are 2 ground pins, function as negative voltage.

b. Analog Pin
The analogue pins on the Arduino can be used for digital input and output. An analog pin can convert incoming analog signals into digital values. This analog pin is connected to the converter on the microcontroller. This converter converts the analog value of a volatile signal into a digital form. In Arduino this converter has a 10 bit resolution, meaning that the value of the conversion result is from 0 to 1023 which equivalent from 0-5V in default or can be set to another value of voltage using AREF pin. In Arduino UNO, the analog pins are marked with labels A0 through A5.

c. Digital Pin
Arduino UNO R3 has 14 digital pins. It consists of:
• PWM (Pulse Width Modulation) pins
There are 6 pins of PWM which are pin 3, 5, 7,9,10 and 11. The PWM pin marked with symbol ~.

d. IOREF Pin
This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5 V or 3.3 V.

e. LED
Led in Arduino is on pin 13. At pin 13 it is made to connect LED pin to digital 13, when pin is connected LED is HIGH it will be ON, and if LED connect LOW it will be OFF. This explanation is for ON LED. Another LED in Arduino Uno board are, RX LED, TX LED and L LED. The RX LED, we know that RX for the receiver that RX LED is indicator if the Arduino receive signal from other Arduino part. For the example in the beginning of testing the Arduino , the Arduino connect to the computer and computer sending the data to Arduino , the RX LED will turn ON because Arduino receiving data. TX LED is indicator if the Arduino transmit signal to the computer, for example when using Serial Monitor command in Arduino IDE, the TX LED will turn ON because Arduino is transmitting data.
f. AREF Pin
This pin is the next of the power pin there are IOREF pin. AREF pin is the references voltage for the analog input.

g. Reset Button
This button use to reset the program that working in Arduino. Typically used to add a rest button to shields which block the one on the board.

h. RX and TX pin
0 (RX) and 1 (TX). The Pin RX is for received the TTL serial data, the RX pin located in pin 0. And Pin TX pin in the 1, the pin itself means transfer TTL serial data. There is the Figure 2.8 of Pin Configuration of Arduino UNO.

![Figure 2.9 Arduino UNO pin configuration](image)

2.5. Arduino UNO IDE
In this project for the software I use Arduino UNO IDE (Integrated Development Environment). This application can support project to running the project. The Arduino IDE interface can be seen in the Figure 2.9 below.
Arduino IDE is a based on processing and library function to easily program the microcontroller. This microcontroller is typically programmed using a dialect of features from the programming languages C and C++. It can be used to make the programming the code and also uploading the code to the device that we use.

### 2.6. Relay Module 2 Channels

Figure 2.10 above is the figure of 2 channel relay module that I used in this project to support PIR motion sensor and also for the controller lamp via android. Relay is a switch that open and close circuit electromechanically or electronically. This relay can be controlled directly with 3.3V or 5V logic signals from a microcontroller witch is Arduino UNO that is used in this project. It has 1x4 pin header for connection power (5V and 0V), and for controlling the 2 relay. The pin are marked:

- **GND** – connect 0V to this pin.
- IN1 – Control relay 1, active low. Relay will turn on input goes below about 2.0 V
- IN2 – Control relay 2, active low. Relay will turn on input goes below about 2.0 V
- VCC – Connect 5V to this pin.

2.7. Fitting Lamp

![Figure 2.12 Fitting lamp](image)

Figure 2.12 above is a the figure of fitting lamp that I used in this project. Because the project is controlling the lamp this component must be able. Fitting lamp is an electrical connection to the lamp and ensures the lamp is held in the correct position or place to put a lamp to be connected into the electrical lines. There are so many fitting lamp, in this project I use type E27.

2.8. Plug – in

![Figure 2.13 Plug-in](image)

Figure 2.13 above is the figure of Plug-In that I used in this project. The lamp in this project needs the power supply for the lamp. AC power plug is devices that allow electrically operated equipment to be connected to primary AC power supply. Plug in this project have 2 pins, no grounding, and usually used 220-240 V.
2.9. Dimmer

These project the author use dimmer. Dimmer is a device used to adjust the glow of incandescent light from out, dim, bright, to very bright. Dimmer can be installed with incandescent light bulb up to 100 watt power. Other than that by using potentiometer the strength of the refractive light is adjusted according to our wish by turn right and left. There are supporting component for support the potentiometer for example resistors, capacitors, IC 555, TRIAC, diode. The dimmer can be seen in the Figure 2.13 below.

Figure 2.14 Dimmer
CHAPTER 3
DESIGN IMPLEMENTATIONS

3.1. Introductory Remarks

This design implementation of the system is categorized into two parts: hardware implementation and software implementation. The general work procedure is the first part to be explained in block diagram followed by flowchart of the whole system, the requirement of component, hardware design, electrical design, and consist of Arduino program coding. The block diagram of the device shown in figure 3.1 below.

**Figure 3.1 Block diagram of the installation system**

Lamp Controller with Arduino based on android via Wi-Fi with motion sensor, the Arduino UNO as the controller of the whole system. Arduino UNO process the input signal from PIR sensor and ESP8266, the output signal will turn on each relay for each lamp.

Flowchart is the full step how the prototype will be work and the block diagram is the simple explanation of the system. The flowchart of the installation can have divided into two sections. First, we can control a lamp with motion sensor if first motion detected until the fourth motion detected. All motion can control the lamp with difference intensity from 25% until 100% and OFF of the brightness. Each lamp can be change because of the relay for each brightness. There are 3 relays to control a lamp with PIR (Passive Infrared Receiver).
The motion must be given through PIR motion sensor and the PIR motion sensor will send the input into Arduino UNO. If the PIR motion sensor detects the motion, relay 1 will turn ON and the lamp will have intensity 25%. If there is a second motion detected by the PIR sensor, relay 2 will turn ON and the lamp 1 will have intensity 50% and the relay 1 will be OFF. If there is third motion detected by the PIR sensor relay 3 will turn ON and the lamp 1 will have intensity 100% and he relay 2 will be OFF. Lastly, if there is a forth motion detected by the sensor it will turn OFF the lamp because the relay 3 is OFF, this sequence will be repeated and continuously happen if there’s another motion detected by the PIR sensor.

The ESP8266 is a module that author use in this project. ESP8266 is the Wi-Fi module for the microcontroller to connect to Wi-Fi. The ESP8266 in this project will catch the Wi-Fi signal it means the serial initialization in flowchart system. Before the module receive the signal from the router, the router transfer the signal to the smartphone first (Android) and then the android transfer the signal to the module and if the module is already connected with the Wi-Fi connection and we can control a lamp with our smartphone android.

In android there will be application to control lamp. First to turn ON a lamp 2 press button ON and the button will change the color from the red to the green, if the button already changes the color it means a lamp is ON and the relay 4 is ON too. For turn OFF a lamp 2 just press the button OFF and it will change color from the green to the red, if the color is already change it means the lamp 2 already OFF by the Android.

The important things are the Arduino can read all the data and the connection of he Wi-Fi must be strong enough for all component work properly. The PIR (Passive Infrared Receiver) also can work well if the place of the sensor in a very minimum disturbance motion so that it will not disturb the reading. All the system will still work if supply is still available from Arduino. Figure 3.2 below shows overall flowchart of the installation.
Figure 3.2 Overall flow chart of the installation

When the PIR and the Arduino is active, we are able to control lamp 1 with the motion around PIR sensor. For lamp 2, when Android is connected we are able to send a data to
the module and the data will be send to microcontroller and will be processed into certain command. Table 3.1 shown the list of that active command.[8]

<table>
<thead>
<tr>
<th>NO</th>
<th>Relay</th>
<th>Appliance</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Relay 1</td>
<td>Lamp, PIR sensor</td>
<td>Turn ON Lamp 1 with intensity 25%</td>
</tr>
<tr>
<td>2</td>
<td>Relay 2</td>
<td>Lamp, PIR sensor</td>
<td>Turn ON Lamp 1 with intensity 50%</td>
</tr>
<tr>
<td>3</td>
<td>Relay 3</td>
<td>Lamp, PIR sensor</td>
<td>Turn ON Lamp 1 with intensity 100%</td>
</tr>
<tr>
<td>4</td>
<td>Relay 4</td>
<td>Lamp</td>
<td>Turn ON and OFF Lamp 2</td>
</tr>
</tbody>
</table>

### 3.2. Hardware Implementation

#### 3.2.1. PIR Motion Sensor Implementation

PIR sensor consists of 3 wires, yellow, black, and red. The positive connect to the power 5V in Arduino, the negative one black wire connect to GND in Arduino and the relay, the last OUT connect to pin 2 in Arduino. Green wire from the Arduino connect to VCC, red wire in Arduino connect to VCC in relay Table 3.2 and Figure 3.4 below.

It will be powered up from the Arduino 5V pin and the output signal will be read by the digital input pin 2 the Arduino.

<table>
<thead>
<tr>
<th>Device</th>
<th>Pin Configurations</th>
<th>5V</th>
<th>GND</th>
<th>Pin 2</th>
<th>Pin 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PIR Sensor</td>
<td>Red Wire</td>
<td>Black Wire</td>
<td>Yellow Wire</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Relay 5V 2 Channels</td>
<td>Red wire</td>
<td>Black wire</td>
<td></td>
<td>Green wire</td>
<td></td>
</tr>
</tbody>
</table>
3.2.2. ESP8266 Wi-Fi Module Implementation

ESP8266 Wi-Fi module is used to communicate between internets with the Arduino. ESP8266 consist of 6 pins connected to the Arduino UNO board to send and receive an input signal. The pin configurations of ESP8266 with Arduino are shown in Table 3.3 and Figure 3.4 below and ESP8266 with the relay 2x2 channels are shown in Figure 3.5 and Table 3.4 below.

<table>
<thead>
<tr>
<th>Design</th>
<th>Pin Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO</td>
<td>3.3 V</td>
</tr>
<tr>
<td>ESP8266</td>
<td>VCC and CH_PD</td>
</tr>
</tbody>
</table>

Figure 3.3 PIR, Arduino uno, and relay implementation

Figure 3.4 ESP8266 implementation between Arduino and ESP8266
3.3. Arduino, Relay, Dimmer, and Lamp Implementation

Figure 3.5 ESP8266 implementation between Arduino with 2x2 relay 5V 2 channels

Table 3.4 Pin Configuration between Arduino UNO and 2x2 Relay 5V 2 Channels

<table>
<thead>
<tr>
<th>Device</th>
<th>Pin Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO</td>
<td>Pin 5V  Pin GND</td>
</tr>
<tr>
<td>Relay 5V 2 Channels</td>
<td>VCC</td>
</tr>
<tr>
<td></td>
<td>GND</td>
</tr>
<tr>
<td></td>
<td>IN 1</td>
</tr>
<tr>
<td></td>
<td>IN 2</td>
</tr>
<tr>
<td></td>
<td>IN 3</td>
</tr>
<tr>
<td></td>
<td>Pin 5</td>
</tr>
<tr>
<td></td>
<td>Pin 6</td>
</tr>
<tr>
<td></td>
<td>Pin 4</td>
</tr>
</tbody>
</table>

Figure 3.6 Arduino, relay, dimmer, and lamp implementation
### Table 3.5 Pin Configuration between Arduino UNO, Relay 5V with 2x2 2 Channels, Dimmer and Lamp

<table>
<thead>
<tr>
<th>Device</th>
<th>Pin Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino UNO</td>
<td>Pin 5V GND Pin 5</td>
</tr>
<tr>
<td>Relay 5V 2 Channels</td>
<td>VCC GND IN 1 IN 2</td>
</tr>
<tr>
<td>Dimmer Diminishes</td>
<td>IN dimmer IN dimmer Out dimmer</td>
</tr>
<tr>
<td>Dimmer Increases</td>
<td>IN dimmer IN dimmer OUT dimmer</td>
</tr>
<tr>
<td>Lamp</td>
<td></td>
</tr>
</tbody>
</table>

### 3.4. Software Implementations

To implement the program to the microcontroller, the author used Arduino IDE 1.6.14. The program will consist of PIR motion sensor, and ESP8266 control module Wi-Fi. Full sketch of the program can be seen in Appendix A.

### 3.5. Hardware Design

![Prototype lamp controller design](image)

**Figure 3.7 Prototype lamp controller design**

### 3.6. Electrical Design (wiring diagram)

The wiring diagram of this final project can be seen in Appendix B.
CHAPTER 4
RESULTS AND DISCUSSIONS

4.1. Results and Discussions

The miniature is designed to control the electrical appliance by android application via Wi-Fi and motion sensor (PIR). The miniature divided into two sections of part. The part one control the lamp by PIR and the second one control the lamp by Android via Wi-Fi. Figure 4.1 shown the miniature appearance.

Figure 4.1 Prototype lamp controller
4.2. Experimental Result

The result is obtained by controlling the lamp with Android and motion sensor (PIR) that has been done during the experiment. Table 4.1 table of result below.

**Table 4.1 Table of Experiment Result of Lamp Controller**

<table>
<thead>
<tr>
<th>No</th>
<th>Command</th>
<th>Appliance Status</th>
<th>Expected</th>
<th>Observation</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Turn ON Lamp 1 intensity 25%</td>
<td>Lamp 1 ON intensity 25%</td>
<td>Lamp 1 ON intensity 25%</td>
<td>Lamp 1 ON intensity 25%</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 ON intensity 25%</td>
<td>Lamp 1 ON intensity 25%</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Turn ON Lamp 1 intensity 50%</td>
<td>Lamp 1 ON intensity 50%</td>
<td>Lamp 1 ON intensity 50%</td>
<td>Lamp 1 ON intensity 50%</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 ON intensity 50%</td>
<td>Lamp 1 ON intensity 50%</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Turn ON Lamp 1 intensity 100%</td>
<td>Lamp 1 ON intensity 100%</td>
<td>Lamp 1 ON intensity 100%</td>
<td>Lamp 1 ON intensity 100%</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 ON intensity 100%</td>
<td>Lamp 1 ON intensity 100%</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Turn OFF Lamp 1</td>
<td>Lamp 1 ON intensity 100%</td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 OFF</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 OFF</td>
<td>Lamp 1 OFF</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Turn ON Lamp 2</td>
<td>Lamp 2 ON</td>
<td>Lamp 2 ON</td>
<td>Lamp 2 ON</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 2 OFF</td>
<td>Lamp 2 ON</td>
<td>Lamp 2 ON</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Turn OFF Lamp 2</td>
<td>Lamp 2 ON</td>
<td>Lamp 2 OFF</td>
<td>Lamp 2 OFF</td>
<td>[✓] Accepted [ ] Rejected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lamp 2 OFF</td>
<td>Lamp 2 OFF</td>
<td>Lamp 2 OFF</td>
<td></td>
</tr>
</tbody>
</table>
This project has two type of controls, with PIR motion sensor and Android. First, the author will control the lamp 1 with PIR. There are 3 motion will be detected by PIR motion sensor. The first motion detected by PIR and the lamp 1 will be on with 25% intensity, shown in figure 4.2 below.

![Figure 4.2 Lamp with 25% intensity](image)

The second motion will change the lamp 1 to 50% intensity if the motion detected. But if the motion not detected the lamp still in 25% intensity. The lamp with 50% intensity shown in figure 4.3 below.

![Figure 4.3 Lamp with 50% intensity](image)
The third motion if this motion detected the lamp 1 will be changed from 50% intensity to 100% intensity or very bright. The lamp with 100% intensity shown in figure 4.4 below.

![Figure 4.4 Lamp with 100% intensity](image)

The last motion will be read to turn lamp 1 OFF, as before if the motion did not detect the lamp still in 100% intensity. Lamp OFF Shown in figure 4.5 below.

![Figure 4.5 Lamp off with PIR](image)
For the ESP8266 the android will control all of the changing of the lamp. Figure 4.6 and 4.7 below show the condition of the lamp if they are controlled by android via Wi-Fi. The figure 4.6 below is the figure when the author presses ON button on Android and lamp 2 is ON. Otherwise, Figure 4.7 is the figure when the author press the OFF button and the lamp 2 will automatically OFF.

Figure 4.6 Lamp 2 on

Figure 4.7 Lamp 2 off
4.3. Experimental Method

This final project board is a prototype. This prototype is small scale installation in real life but the system is same. Once the android smartphone is already paired with Wi-Fi, the signal transfer to the module ESP8266 and the lamp can controlled by Android.

The signal from Android will be sent to the Wi-Fi module and the module will send the data to be read by Arduino then Arduino will translate the data sent from the module to turn ON the relay which will make the lamp ON. Controlling with Android by pressing button in application, red button for turn OFF the lamp and green button for turn ON the lamp.

In motion sensor, or we called it PIR motion sensor the sensor is very sensitive. The PIR sensor can receive the motion and transmit the signal to the Arduino. The Arduino receive the signal and process the signal to turn ON or turn OFF the relay. In this project every motion will be divided into 4 section intensity of the lamp. First motion the sensor will catch the motion to change the lamp with 25% intensity, and second motion the sensor will catch the motion to change the lamp with 50% intensity, the third motion the sensor will catch the motion to change the lamp with 100% intensity, and last motion the lamp will be OFF.

The sensor only catches the motion with the radius between 0-5 meters long without obstacle. The ESP8266 module working base on the internet speed, if the speed of internet good the output will be good working but if the speed of internet low the output will be not good result.
4.4. Strength and Weakness

The strength of the prototype in this final project are:

- The controlling system of the prototype run well as the program implemented in microcontroller.
- The main system also run as expected.
- The motion sensor also runs with a good output.
- The sensor can control three types of brightness of the lamp as we want.
- The communication with Wi-Fi can easily control and monitor the electricity in really long distance.

The weaknesses of the prototype in this final project are:

- Sometime there is a delay in reading data in Arduino.
- Due to sensitivity, it is better to place the sensor in a very minimum disturbance motion so that it will not disturb the reading.
CHAPTER 5
CONCLUSIONS AND FUTURE DEVELOPMENTS

5.1. Conclusions

Based on the result of the project, “Prototype Lamp Controller with Arduino via Wi-Fi with Motion Sensor”, there are two conclusions that can be presented:

1. The application on android is successfully implemented as the controller a lamp with Wi-Fi connection and the prototype can connect with ESP8266 module Wi-Fi.
2. The PIR motion sensor is successfully implemented as the intensity controller in prototype with four differences of brightness.

5.2. Recommendations and Future Developments

There are several possibilities to develop and improve the result of this final project. Two recommendations for the prototype resulted from this final project are:

1. In order to enhance the range of the intensity, it is recommended to implement the dimmer function on Android Application. So, the intensity can be change based on the user wants.
2. That gadget is used not only Android but also other gadget, such as IOS in iPhone and other.
REFERENCES


## APPENDIX A

### PROGRAM CODE

<table>
<thead>
<tr>
<th>Program Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>#include &lt;crc16.h&gt;</code></td>
<td>//Library for crc 16</td>
</tr>
<tr>
<td><code>#include &lt;espduino.h&gt;</code></td>
<td>//Library for espduino (ESP8266 library)</td>
</tr>
<tr>
<td><code>#include &lt;FP.h&gt;</code></td>
<td>//Library for FP</td>
</tr>
<tr>
<td><code>#include &lt;mqtt.h&gt;</code></td>
<td>//Library for mqtt</td>
</tr>
<tr>
<td><code>#include &lt;rest.h&gt;</code></td>
<td>//Library for rest</td>
</tr>
<tr>
<td><code>#include &lt;ringbuf.h&gt;</code></td>
<td>//Library for ringbuf</td>
</tr>
<tr>
<td><code>#include &lt;SoftwareSerial.h&gt;</code></td>
<td>//Library for softwareserial (software serial for ESP8266)</td>
</tr>
<tr>
<td><code>#include &lt;espduino.h&gt;</code></td>
<td></td>
</tr>
<tr>
<td>`#define Rwifi 3</td>
<td>//Relay 4 pin for Wi-Fi</td>
</tr>
<tr>
<td>`#define R1 4</td>
<td>//Relay 1 pin for PIR Sensor</td>
</tr>
<tr>
<td>`#define R2 6</td>
<td>//Relay 2 pin for PIR Sensor</td>
</tr>
<tr>
<td>`#define R3 5</td>
<td>//Relay 3 pin for PIR Sensor</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><code>int indikator = 13;</code></td>
<td>//Indicator for LED</td>
</tr>
<tr>
<td><code>int inputVout = 2;</code></td>
<td>//Indicator for Vout PIR</td>
</tr>
<tr>
<td><code>int statusPIR = 0;</code></td>
<td>//Indicator for status logical PIR</td>
</tr>
<tr>
<td><code>int data = 0;</code></td>
<td>//Temporary variable for collecting data of PIR</td>
</tr>
<tr>
<td><code>int i = 0;</code></td>
<td></td>
</tr>
<tr>
<td><code>int w = 0;</code></td>
<td></td>
</tr>
<tr>
<td><code>SoftwareSerial espPort(10, 11);</code></td>
<td></td>
</tr>
<tr>
<td><code>ESP esp(&amp;espPort, &amp;Serial, 9);</code></td>
<td></td>
</tr>
<tr>
<td><code>REST rest(&amp;esp);</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><code>boolean wifiConnected = false;</code></td>
<td>//Indicator the motion of PIR</td>
</tr>
<tr>
<td><code>boolean data1 = false;</code></td>
<td></td>
</tr>
<tr>
<td><code>int loop_count = 0;</code></td>
<td></td>
</tr>
<tr>
<td><code>char response[266];</code></td>
<td>//Indicator ESP8266 serial &amp; port Restart of ESP8266</td>
</tr>
<tr>
<td><code>char buff[64];</code></td>
<td></td>
</tr>
<tr>
<td><code>String strId,strData,strCode;</code></td>
<td>//Stating the variable for Wi-Fi communication</td>
</tr>
<tr>
<td><code>String strData_Last1,strData_Last2,strData_Last3,strData_Last4;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td><code>//void(* resetFunc) (void) = 0;</code></td>
<td>//Response sequence for Wi-Fi data</td>
</tr>
<tr>
<td><code>void clearBuffer(void) {</code></td>
<td></td>
</tr>
<tr>
<td>for (int i = 0;i&lt;266;i++ ) {</td>
<td></td>
</tr>
<tr>
<td>response[i]=0;</td>
<td></td>
</tr>
<tr>
<td>}</td>
<td></td>
</tr>
</tbody>
</table>
```cpp
//status when ESP connected to Wi-Fi
if(status == STATION_GOT_IP) {
    Serial.println("TERHUBUNG KE WIFI");
    wifiConnected = true;
} else {
    wifiConnected = false;
}
}

void wifiCb(void* response) {
    uint32_t status;
    RESPONSE res(response);
    if(res.getArgc() == 1) {
        res.popArgs((uint8_t*)&status, 4);
        if(status == STATION_GOT_IP) {
            Serial.println("TERHUBUNG KE WIFI");
            wifiConnected = true;
        } else {
            wifiConnected = false;
        }
    }
}

void setup() {
    pinMode(Rwifi, OUTPUT);
    digitalWrite(Rwifi, HIGH);
    pinMode(indikator, OUTPUT); // set pin 13 sbg output
    pinMode(inputVout, INPUT); // set pin 2 sbg input
    Serial.begin(9600); // serial monitor
    pinMode(R1, OUTPUT);
    pinMode(R2, OUTPUT);
    pinMode(R3, OUTPUT);
    digitalWrite(R1, HIGH);
    digitalWrite(R2, HIGH);
    digitalWrite(R3, HIGH);
    Serial.begin(9600);
    espPort.begin(19200);
    esp.enable();
    delay(500);
    esp.reset();
    delay(500);
    while(!esp.ready());
    Serial.println("ARDUINO: Setup client");
    if(!rest.begin("api.thingspeak.com")) {
        //If the condition of wifiConnected is true, then ESP is connected
        //if the condition of wifiConnected is false then ESP is not connected
        //A Function that is running only once at the beginning and to declare the component as an input output
        //pinMode() command it configures the pin inside the bracket to behave either as an input or output, according the programmer needs
        //Setting the relay condition
        // Declaring the communication between Arduino and Serial monitor
        //Declaring the ESP bit rate communication
        //for recall library thingspeak
    }
```
```cpp
Serial.println("ARDUINO: Gagal Setup client");
    while(1);
}
Serial.println("ARDUINO: Menghubungkan dengan Wifi");

esp.wifiCb.attach(&wifiCb);

esp.wifiConnect("Andromax-M3S-8949","73122302");
Serial.println("ARDUINO: System sudah siap!");
}

void loop()
{
    pir();
    esp.process();
}

void pir(){
data = digitalRead(inputVout); // baca input dr Vout
if ((data == HIGH) && (statusPIR == LOW))
{
    i++;
    w++;
    statusPIR = HIGH;
}
if(i == 0)
{
digitalWrite(R1,HIGH);
digitalWrite(R2,HIGH);
digitalWrite(R3,HIGH);
}
if(i== 1)
{
digitalWrite(R1,LOW);
digitalWrite(R2,HIGH);
digitalWrite(R3,HIGH);
}
if(i==2)
{
digitalWrite(R1,HIGH);
digitalWrite(R2,LOW);
digitalWrite(R3,HIGH);
}
if(i==3){
digitalWrite(R1,HIGH);
```

//For show the word of ARDUINO: Menghubungkan dengan Wifi in serial monitor

//The SSID used is Andromax-M3S-8949,with password 73122302
//For show the word of ARDUINO:

//loop() is a function which always running repeatedly until there is no power flows in Arduino
//pir() is a function to PIR works

//reading signal from PIR sensor, if the data high and status PIR low so the status PIR will be HIGH and the lamp 1 will be ON.(with motion)

//it means if the value of input is equal to 0, all Relay is off

//it means if the value of input is equal to 1, the lamp 1 will be on with 25% intensity while R1 on and both R2, R3 is off.

//it means of the value of input is equal to 2, the lamp 1 will be on with 50% intensity while R2 is on and both R1 and R3 is off.

//it means of the value of input is equal to 3, the lamp 1 will be on with 100% intensity while both R1 and
digitalWrite(R2,HIGH);
digitalWrite(R3,LOW);
if(i==4){
i=0;
}
else {
    if ((data == LOW) && (statusPIR == HIGH)){
        statusPIR = LOW;
    }
}

void wifi(){
  loop_start:
  esp.process();
  if(wifiConnected) {
    char str_field1[6] , str_field2[6];
    sprintf(buff, "/channels/403272/fields/1/last");
    Serial.println(buff);
    rest.get((const char*)buff);
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
      strId = "";
      strData = "";
      strCode = "";
      getData();
      if (strId == "1" || strId == "1.0" || strId == "1.00"){
        digitalWrite(Rwifi,LOW);
        data1 = true;
        digitalWrite(R2,HIGH);
        digitalWrite(R3,LOW);
      } else{ // hardReset();
    }
  } else{ //wifi() is a function of wifi when the wifi is active, and looping started
    rest.get((const char*)buff);
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
      strId = "";
      strData = "";
      strCode = "";
      getData();
      if (strId == "1" || strId == "1.0" || strId == "1.00"){
        digitalWrite(Rwifi,LOW);
        data1 = true;
        digitalWrite(R2,HIGH);
        digitalWrite(R3,LOW);
      } else{ // hardReset();
    }
  } 

R2 is off and R3 is on.

//it means if the data low and status PIR high so the status PIR will be HIGH and the lamp 1 will be OFF.(with motion)

//wifi() is a function of wifi when the wifi is active, and looping started

//esp will be process if the wifi already connected

//waiting for the data from “Reconnect” application

//it means if the data equal to 1,1.0,or 1.00 relay wifi will be on and the lamp 2 will be on

//it means if the data equal to 0,-1,0.1,or 0.1 relay wifi will be off and the lamp 2 will be off

//for reset the hardware
// resetFunc();
{
    delay(5);
    loop_count++;
    Serial.println("LOOP :");
    Serial.println(loop_count);
    if(loop_count == 2 ){
        loop_count = 0;
        if(data1) {
            dtostrf(1, 1, 1, str_field1);
        }else{
            dtostrf(0.1, 2, 1, str_field1);
        }
        sprintf(buff, 
"//update?key=1S3925WE7FNOUS4C&field1=%s&field2=%s&field3=%s&field4=%s",str_field1,str_field2);
        Serial.println(buff);
        rest.get((const char*)buff);
        Serial.println("ARDUINO: Mengirim data terbaru");
        if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
            Serial.println("ARDUINO: Berhasil GET Data");
            strId = ""; strData = ""; strCode = "";
            getData();
        }
        delay(5);
    }
}
else{
}
}

void getData(){
    int i=0,j=0,k=0;
    for (i = 0; i < 10; i++){
        if((response[i] == \r') || (response[i] == \n')) {}
```c
else{
    strId += response[i];
}

if (response[i] == '\n'){
    i++;
    break;
}
}

Serial.println('');
Serial.print("ID : ");
Serial.print(strId);

for (j = i; j < (i+20); j++){
    if((response[j] == '\r') || (response[j] == '\n')) {
    }
    else{
        strData += response[j];
    }
    if (response[j] == '\n'){
        j++;
        break;
    }
}
Serial.println('');
Serial.print("Data : ");
Serial.print(strData);

for (k = j; k < (j+10); k++){
    if((response[k] == '\r') || (response[k] == '\n')) {
    }
    else{
        strCode += response[k];
    }
    if (response[k] == '\n'){
        break;
    }
}
Serial.println('');
Serial.print("Code : ");
Serial.print(strCode);
Serial.println('');
```

//Print strId data

//"j" is the variable do the sequence response to get the strData data

//"k" is the variable do the sequence response to get the strCode data

//Print strCode to serial monitor
boolean hardReset() {
    String tmpData;
}
APPENDIX B
WIRING DIAGRAM