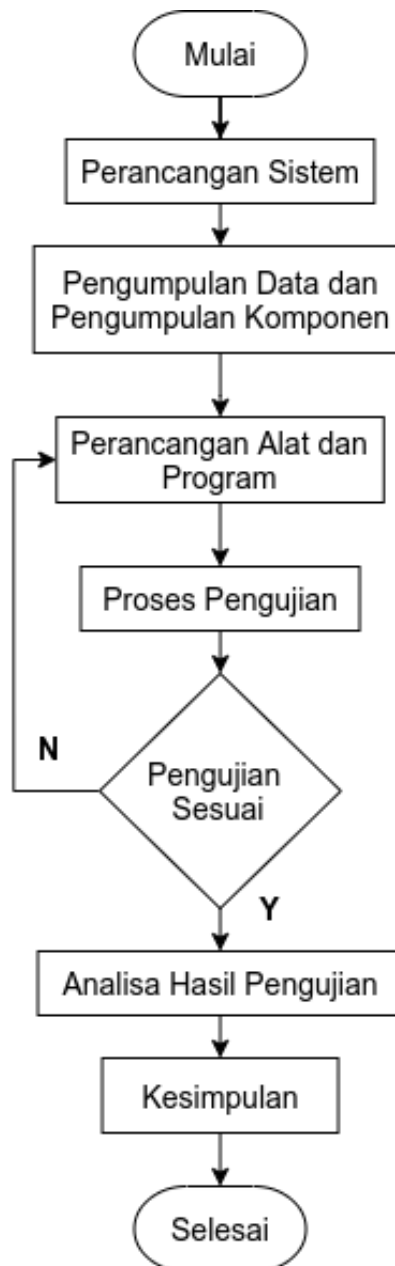


**BAB III**  
**METODE PENELITIAN**

**1.1 *Flowchat* Penelitian**



Gambar 3.1 *Flowchart* Penelitian

Penjelasan tahapan pada gambar 3.1:

1. Tahap pertama adalah mulai penelitian.
2. Tahap kedua adalah perencanaan sistem mengenai alat yang akan dibuat.
3. Tahap ketiga adalah pengumpulan data dan pengumpulan komponen yang dibutuhkan dalam pembuatan alat.
4. Tahap keempat adalah pembuatan alat sesuai dengan perencanaan.
5. Tahap kelima adalah pembuatan program menggunakan software Arduino Ide dan sudah dihubungkan ke NodeMCU dan Arduino Uno.
6. Tahap keenam adalah pengujian alat dan program, ini dilakukan untuk mengetahui apakah program yang dibuat sudah sesuai dengan alat yang dibuat.
7. Tahap ketujuh adalah analisa hasil pengujian.
8. Kesimpulan.
9. Selesai penelitian.

## **1.2 Tempat Penelitian**

Kegiatan ini dilaksanakan di lokasi studi yaitu di Laboratorium Teknik Elektro Universitas Siliwangi jln. Siliwangi No.24 Kota Tasikmalaya.

## **1.3 Studi Literatur**

Melakukan pembacaan beberapasumber seperti jurnal-jurnal yang telah dipublikasikan ataupun buku buku yang berkaitandengan teori mengenai *smart PJU* dengan kendali *power lane* kontroler.

## **1.4 Metode Pengambilan data**

Metode yang digunakan dalam mengumpulkan data pada penelitian ini adalah sebagai berikut:

### **1. Teknik observasi**

Metode ini merupakan teknik pengamatan yang dilakukan secara langsung untuk mengamati waktu kirim dan menerima *Master* terhadap *Slave* dalam *power lane*.

### **2. Studi pustaka**

Metode ini dilakukan dengan cara membaca dan mengumpulkan teori-teori penting yang dijadikan sebagai landasan atau kajian deduktif. Studi pustaka yang penulis lakukan berhubungan dengan *smart* PJU dengan kendali *power lane* kontroler.

## **1.5 Pemodelan Alat**

Pada tahap pemodelan alat ini meliputi beberapa tahap yaitu perancangan komponen yang digunakan untuk pemodelan perangkat keras dan perangkat lunak pada alat *smart* PJU ini.

### **1.5.1 Komponen**

Komponen yang akan digunakan dalam pemodelan alat ini terbagi menjadi 2 bagian yaitu komponen elektronik dan peralatan. Komponen elektronik akan digunakan untuk merancang rangkaian elektronik alat. Dan peralatan digunakan untuk membantu dalam pengerjaan alat seperti ditunjukkan pada tabel 3.1 dan 3.2.

Tabel 3.1 Peralatan

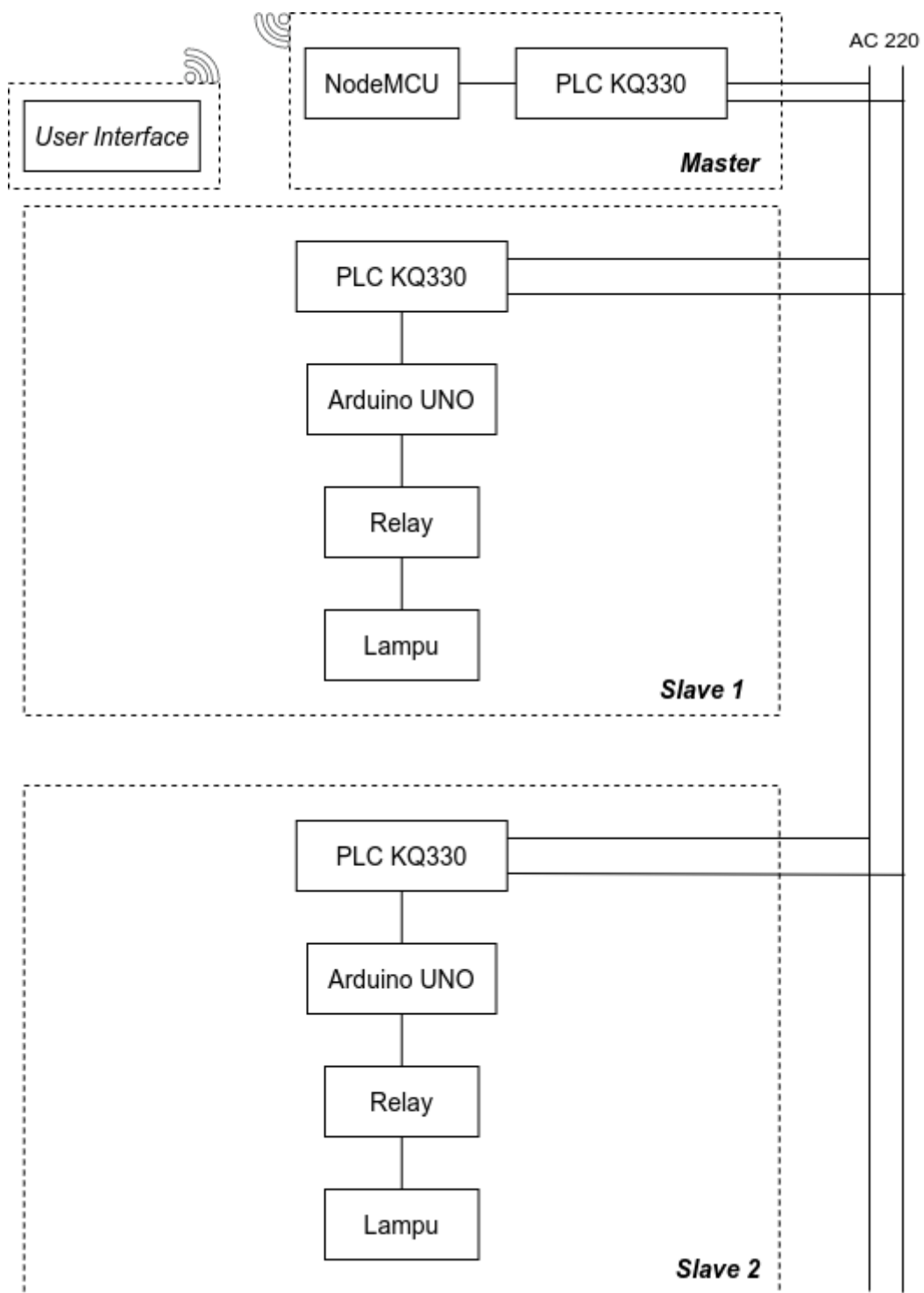
| <b>Nama Alat</b>   | <b>Fungsi</b>                     |
|--------------------|-----------------------------------|
| Bor                | Melubangi papan PCB, Box, dll.    |
| Solder dan Timah   | Soldering                         |
| Penggaris          | Alat ukur                         |
| Obeng              | Memasang dan membuka baut         |
| Tang               | Memotong kabel, mengunci mur      |
| <i>Multitester</i> | Digital pengukuran satuan listrik |

Tabel 3.2 Komponen Elektronik

| <b>Nama</b>                               | <b>Fungsi</b>                             | <b>Jumlah</b> |
|---|---|---------------|
| <i>NodeMCU</i>                            | <i>Processor</i> utama pada <i>Master</i> | 1             |
| <i>Breadboard</i>                         | Penghasil energi listrik                  | 1             |
| <i>Power Line Carrier</i>                 | Penghubung <i>Master</i> ke <i>Slave</i>  | 3             |
| <i>Arduino UNO</i>                        | <i>Processor</i> utama pada <i>Slave</i>  | 2             |
| Modul Relay 1<br><i>Channel</i>           | Mendeteksi Tegangan yang dihasilkan       | 2             |
| Kabel <i>Jumper</i><br><i>male-male</i>   | Penghubung Modul ke Modul                 | 10            |
| Kabel <i>Jumper</i><br><i>male-female</i> | Penghubung Modul ke Modul                 | 10            |
| <i>Power Supply 12</i><br><i>Volt</i>     | <i>Supply</i> Tegangan Komponen           | 3             |

### 1.5.2 Pemodelan Sistem

Dalam proses pemodelan terdapat beberapa tahapan untuk *Smart PJU* dengan Kendali *Power line controller* seperti gambar 3.2.



Gambar 3.2 Blok diagram pemodelan perangkat keras

*Hardware* terbagi menjadi tiga yaitu meliputi *hardware Master* sebagai kendali *Slave* dan *Master* langsung terhubung ke *User Interface*. dan dua *hardware Slave* sebagai penerima data berupa nyala matinya lampu Penerangan Jalan Umum.

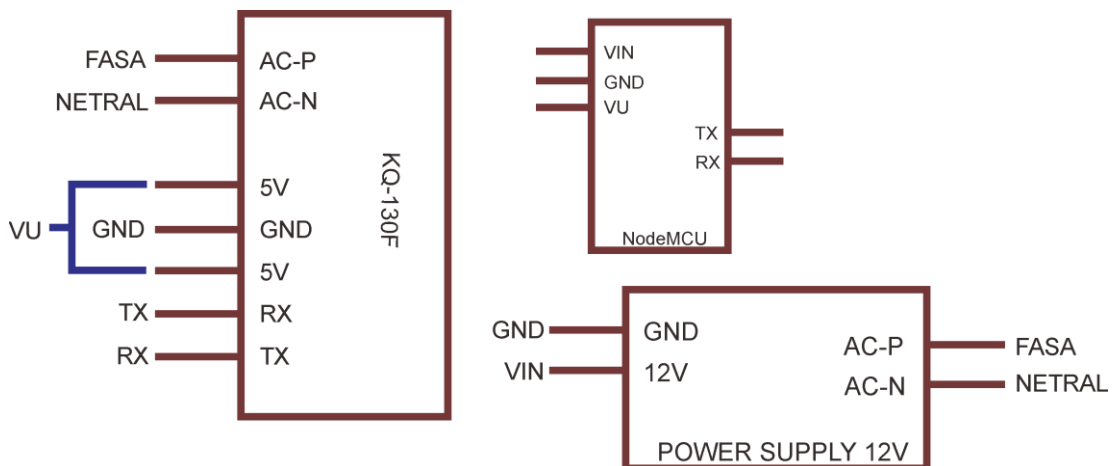
Pemodelan software meliputi program pengirim dan penerima data melalui modul *Power Line Carrier*, program pengirim terhadap *tiap-tiap Slave*, dan program *User Interface*

### 1.5.3 Pemodelan Perangkat Keras

Komponen utama dalam pemodelan perangkat keras adalah NodeMCU V3. Mikrokontroler ini di integrasikan dengan komponen-komponen lainnya agar dapat penyalakan lampu.

#### 1. Pemodelan *Hardware (Master)*

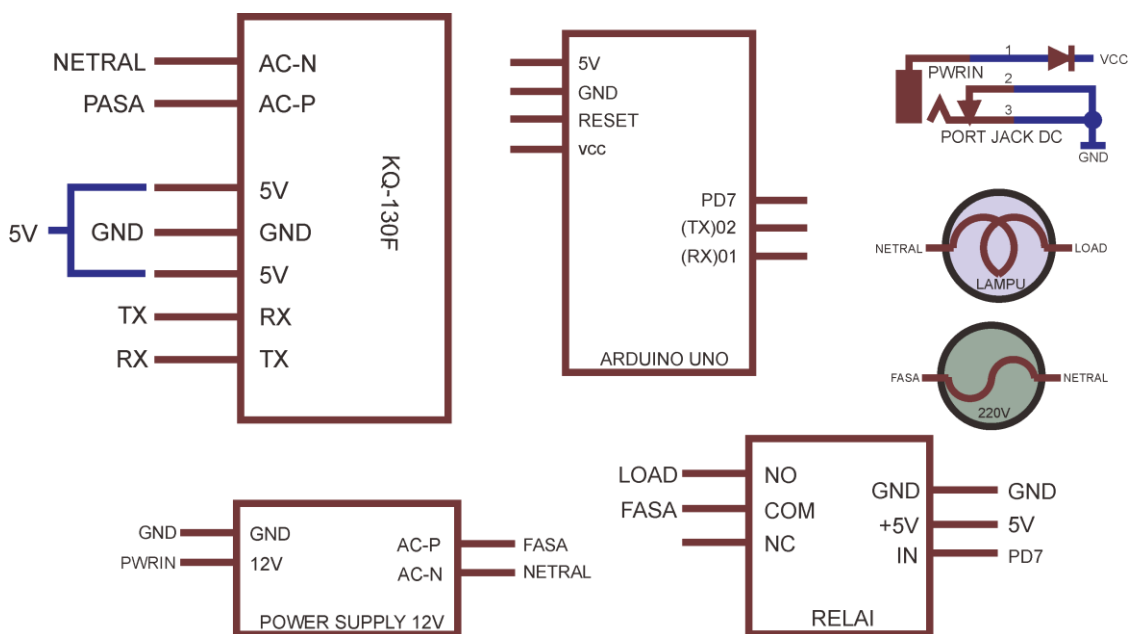
Pemodelan alat ini terdiri atas terdiri atas Arduino Mega, modul *Power Line Carrier* KQ330. Arduino Mega bertindak sebagai pengatur dari semua komponen yang terpasang di alat ini sesuai dengan program yang di-upload. Modul *Power Line Carrier* KQ330 berfungsi sebagai media komunikasi untuk mengirim dan menerima data ke *Power Line Carrier* pada perangkat *Slave*.



Gambar 3.3 Skema *hardware* pada *Master*

## 2. Pemodelan *Hardware (Slave)*

Pemodelan alat ini terdiri atas terdiri atas Arduino Uno, modul *Power Line Carrier KQ330*, dan relai. Arduino Uno berfungsi sebagai pengendali dari semua komponen yang terpasang di alat ini sesuai dengan program yang diunggah. Modul *Power Line Carrier KQ330* berfungsi sebagai media komunikasi pada modul *Power Line Carrier* di perangkat *Master*.



Gambar 3.4 Skema *hardware* pada *Slave*

## 3. Pemodelan Modul *Power Line Carrier (PLC) KQ330* pada *Master*

Modul *Power Line Carrier* di alat ini berfungsi untuk mengirim data pada *Slave*. Jenis komunikasi data yang digunakan adalah komunikasi data serial. Komunikasi jenis ini digunakan untuk komunikasi antara dua board Arduino lain. Dalam alat ini, NodeMCU berkomunikasi serial melalui listrik 220 V memanfaatkan modul PLC Komunikasi ini menggunakan dua buah pin yaitu pin RX untuk menerima data dan pin TX untuk mengirimkan data. Pin ini

terdapat di NodeMCU dan PLC. Pin TX dan RX NodeMCU disebut pin Serial. Modul PLC KQ330 seperti pada Gambar 3.5 yang memiliki sembilan pin yang deskripsinya tertera pada Tabel 3.3, dan spesifikasinya pada Tabel 3.4.

Tabel 3.3 Konfigurasi Pin Modul *Power Line Carrier* KQ330

| Pin Modul PLC | Fungsi  |
|---------------|---|
| AC            | 220 VAC   |
| AC            | 220 VAC   |
| ± 12V         | +12 power supply (300mA)  |
| GND           | Digital Circuit Ground  |
| ± 5V          | +5V power supply (12 mA)  |
| RX            | TTL level, pembawa data masuk dari mikrokontroler port TX                 |
| TX            | TTL level, pembawa data keluar dari mikrokontroler port RX                |
| Mode          | Pemilih mode Mode <i>High</i> = floating atau 5V Mode <i>low</i> = ground |
| NC/RST        | Pin reset ( <i>active low</i> ) Digunakan untuk mode atur frekuensi       |



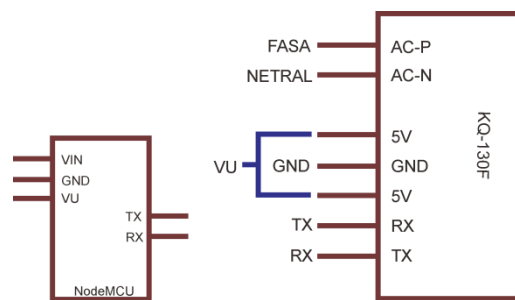
Gambar 3.5 Modul *Power Line Carrier* (PLC) KQ330

Tabel 3.4 Spesifikasi Modul *Power Line Carrier* (PLC) KQ330

| Spesifikasi         | Nilai           |
|---------------------|-----------------|
| Frekuensi Kerja     | 120kHz -135 kHz |
| Interface baud rate | 9600 bps        |
| Actual baud rate    | 100 bps         |



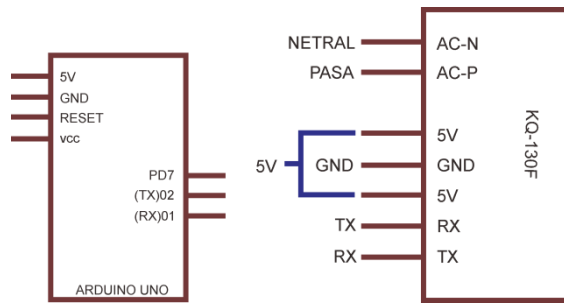
|   |   |
|---|---|
| Panjang maksimal transmisi data one frame terus-menerus | $\leq 252 \text{ bytes}$                    |
| Sensitivitas penerimaan data                            | $\leq 1 \text{ mVAC/1 min}$<br>$5\text{mA}$ |
| <i>Out-of-band rejection</i>                            | $\geq 60 \text{ dB}$                        |
| <i>Permissible Error</i>                                | $\leq 10 \text{ kHz}$                       |
| Bandwidth   | DC +5 V $I_{\text{max}} \leq 20$<br>mA      |



Gambar 3.6 Skema Modul *Power Line Carrier* (PLC) KQ330 pada *Master*

#### 4. Pemodelan Modul *Power Line Carrier* (PLC) KQ330 pada *Slave*

Pemasangan Modul *Power Line Carrier* pada *Slave* ini dimaksudkan untuk menerima data yang dikirim oleh *Master*. Komunikasi serial untuk mengirim data-data dengan melibatkan dua buah pin yaitu pin RX untuk menerima data dan pin TX untuk mengirimkan data. Pin ini terdapat di Arduino UNO dan PLC. Serial yang dimanfaatkan untuk memasang modul PLC adalah Serial 0 dengan pin 1. Secara umum, modul PLC KQ330 terdiri atas sembilan pin dengan deskripsi yang sudah tertera. Dalam alat ini, Pin TX dan RX modul PLC dihubungkan ke port RX dan TX Arduino UNO. Pin 5-12V pada PLC disambung ke pin VU Arduino UNO. Pin GND pada PLC disambung ke GND Arduino UNO. 2 pin AC terhubung ke listrik 220 V.



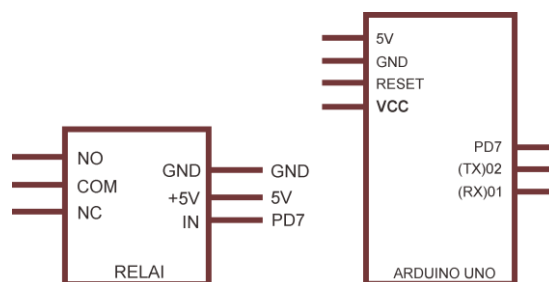
Gambar 3.7 Skema Modul *Power Line Carrier* (PLC) KQ330 pada *Slave*

## 5. Relai

Dalam rangkaian ini relai berperan sebagai media atau saklar antara sumber tegangan PLN (220VAC) dan beban lampu penrangan jalan. Penggunaan relay bertujuan untuk mengaktifkan dan menonaktifkan aliran arus dari PLN ke beban lampu penerangan. Untuk memberikan kondisi saklar *ON* dan *OFF* pada relay, digunakan mikrokontroler sebagai pengendali relay berdasarkan kondisi dan parameter yang telah ditetapkan.



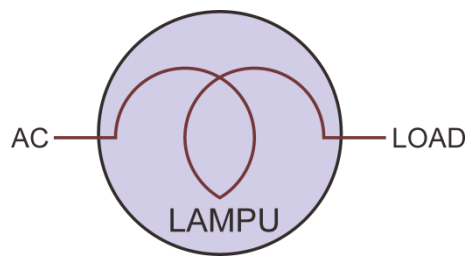
Gambar 3.8 Relai



Gambar 3.9 Skema relai ke arduino uno

## 6. Lampu penerangan

Lampu penerangan merupakan *output* dari rangkaian sistem. Lampu penerangan yang digunakan adalah lampu dengan tegangan 220 volt.

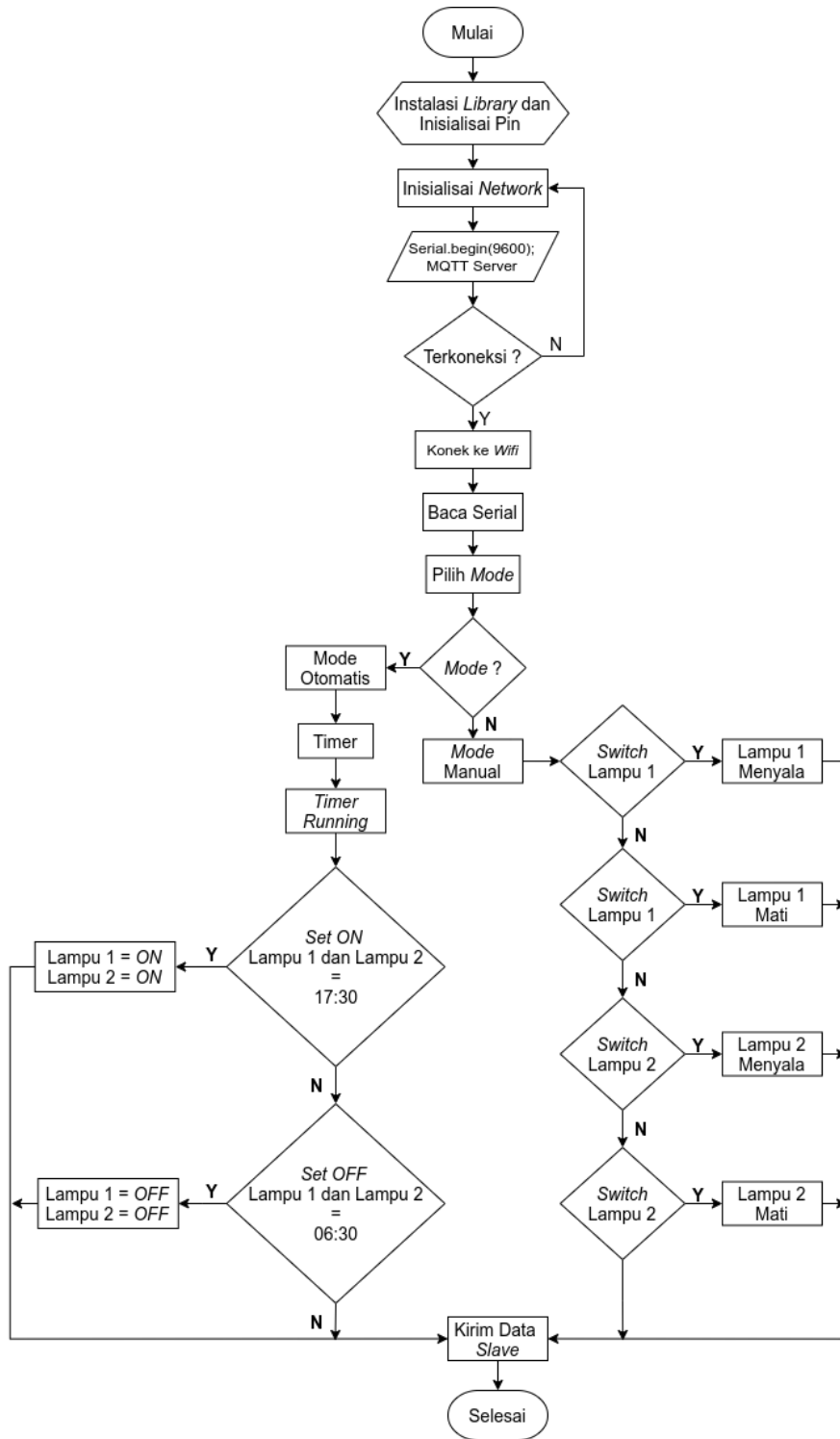


Gambar 3.10 Skema Lampu 220 volt

### 1.5.4 Pemodelan Perangkat lunak

Pemodelan perangkat lunak untuk menjalankan alat, pemrograman digunakan agar mampu membuat interaksi antar komponen.

# 1. Pemodelan Perangkat lunak *Master*



Gambar 3.11 Diagram *Master* perangkat lunak

Berdasarkan gambar 3.11 diagram *Master* perangkat lunak pada perangkat *Master* terdiri atas program Instalasi Wifi, pengiriman data melalui *Power Line Carrier* (PLC), Mode Otomati dan Mode manual.

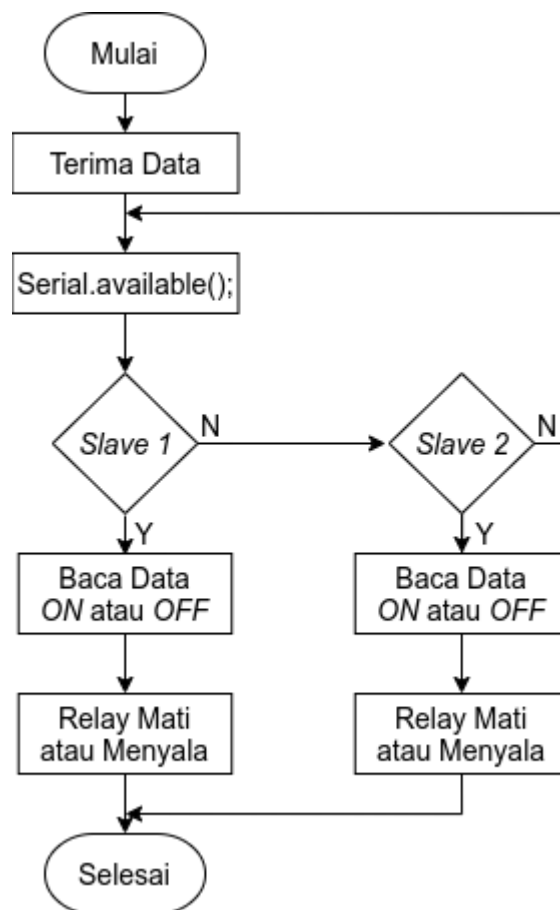
Proses instalasi *WIFI* adalah memberikan program untuk menghubungkan ke jaringan *WIFI*. Proses yang dilakukan dengan memasukan alamat *SSID* dan *Password router* atau *WIFI* yang digunakan. Proses tersebut secara berulang dan akan selesai sampai NodeMCU terhubung pada jaringan. Setelah NodeMCU terhubung maka akan memproses *input* server MQTT Broker yang telah dimasukan ke dalam program yaitu test.mosquitto.org serta memberikan Topik berupa “Penerangan Jalan Umum”.

Program pada NodeMCU terdiri dari 2 main program yaitu fungsi manual dan fungsi otomatis. Mode manual digunakan untuk mengendalikan lampu secara manual, *user* dapat mengendalikan lampu sesuai dengan keinginan. Pada fungsi manual dilakukan dengan memilih menu manual pada interface kemudian pilih *Switch* lampu 1 atau lampu 2 yang akan di nyalakan/dimatikan secara manual. Setelah proses manual digunakan maka masing-masing *Slave* akan menerima dan membaca data yang diperoleh.

Fungsi *Timer* atau otomatis digunakan untuk mengendalikan Penerangan Jalan Umum sesuai jadwal menyala atau matinya lampu sesuai waktu yang telah ditentukan. Waktu tersebut diperoleh dari waktu jam global yang sudah terkoneksi dengan NodeMCU, setelah waktu sudah sesuai dengan set nyala dan matinya lampu maka akan memberikan perintah kepada *Slave*.

## 2. Pemodelan Perangkat lunak *Slave*

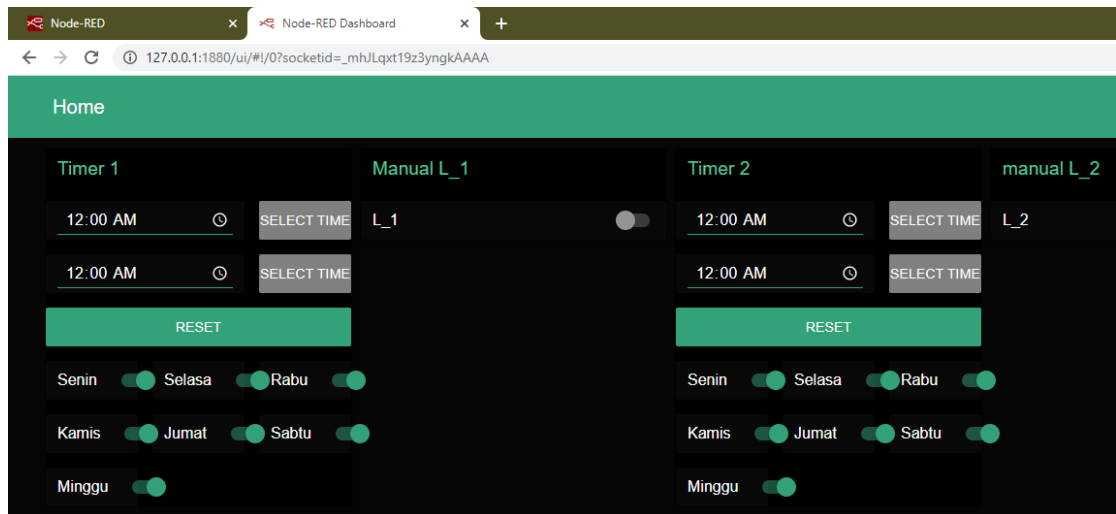
Pemodelan software pada perangkat *Slave* terdiri atas program komunikasi data melalui *Power Line Carrier* (PLC) dan Relay.



Gambar 3.12 Diagram *Slave* perangkat

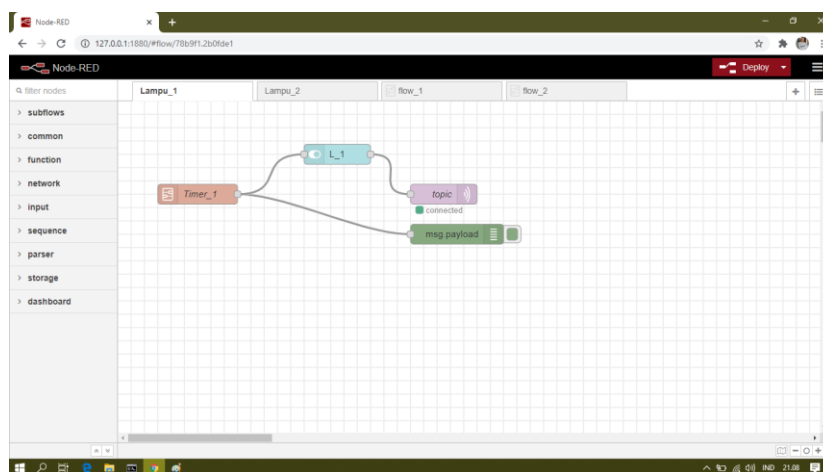
Program pada *Slave* hanya berupa program penerimaan data yang dikirim oleh perangkat *Master*, program tersebut berupa menyala atau matinya lampu. data yang diterima akan diproses oleh masing-masing *Slave*, tiap *Slave* mempunyai alamat tersendiri agar tidak terjadi kesalahan ketika memproses data *Slave 1* atau *Slave 2*.

### 3. Pemodelan *Dashboard* Node-Red



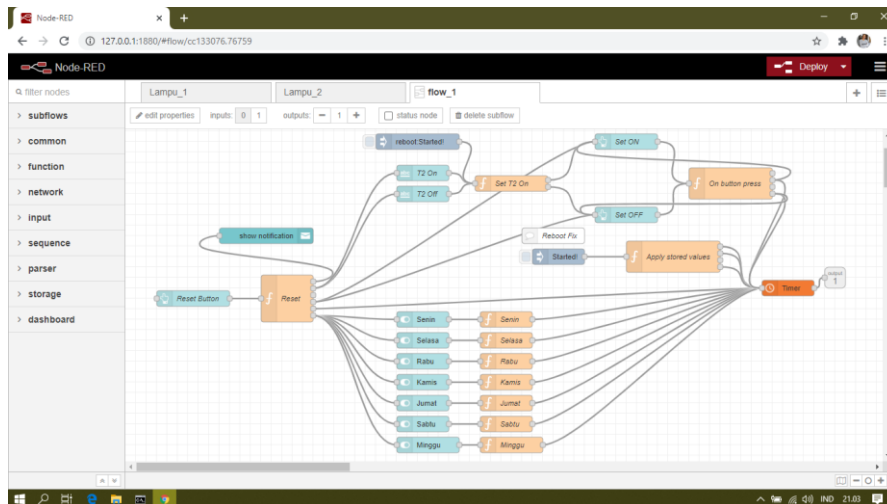
Gambar 3.13 Tampilan *user interface*

*Software* yang digunakan untuk *user interface* (UI) pada sistem ini dibuat menggunakan aplikasi programming Node-RED. Node-RED digunakan untuk membuat *dashboard* monitoring dan pengendalian menjadi tampilan UI. Memprogram Node-RED dilakukan dengan menyusun palette yang terdapat pada sistem Node-RED. Palette adalah node-node yang berguna dalam pembuatan *dashboard*, node-node tersebut memiliki fungsi yang berbeda-beda tergantung kegunaan masing-masing palette.



Gambar 3.14 flow proses switch lampu

Gambar 3.14 merupakan flow switch lampu. Pada proses ini terdapat Palette switch yang memberikan kondisi hidup atau mati pada relay. Setelah kondisi ditentukan proses masuk ke fungsi parsing yang bertujuan untuk menguraikan data. Tipe data yang ditentukan switch adalah String, dalam proses parsing diubah menjadi nilai “0” dan “1”. Selain nilai kondisi switch terdapat juga nilai Timer yang diisi pada nilai setpoint. Setelah nilai diuraikan dengan kondisi switch dan nilai Timer pada setpoint kemudian dikirim menggunakan palletes MQTT *output* ke MQTT broker.



Gambar 3.15 flow proses otomatis

Gambar 3.15 merupakan flow proses otomatis. Pada proses ini digunakan Palette slider yang berfungsi memberikan nilai Timer. Nilai Timer pada slider berupa waktu penjadwalan hidup atau mati lampu. Setelah nilai ditentukan pengguna fungsi parsing akan menguraikan data. Data yang dikirim berisi nilai setpoint dan diberikan nilai Timer pada kondisi switch lampu. Setelah nilai diuraikan nilai-nilai tersebut dikirim menggunakan Palette MQTT *output* ke MQTT Broken.