

BAB III

IMPLEMENTASI

3.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan aplikasi lampu lalu lintas (*traffic light*) pada pertigaan yang menggunakan Arduino Uno sebagai perangkat keras utama terdapat komponen-komponen dan lain sebagainya yang diperlukan :

a. Komponen

1. Mikrokontroler Arduino Uno
2. Resistor 220k
3. Buzzer
4. LED Hijau, Kuning dan Merah

b. bahan

1. PCB
2. Tenol
3. Kabel Jumper
5. Port USB
4. Kertas/sticker

c. Alat

1. solder
2. Cutter
3. Gunting

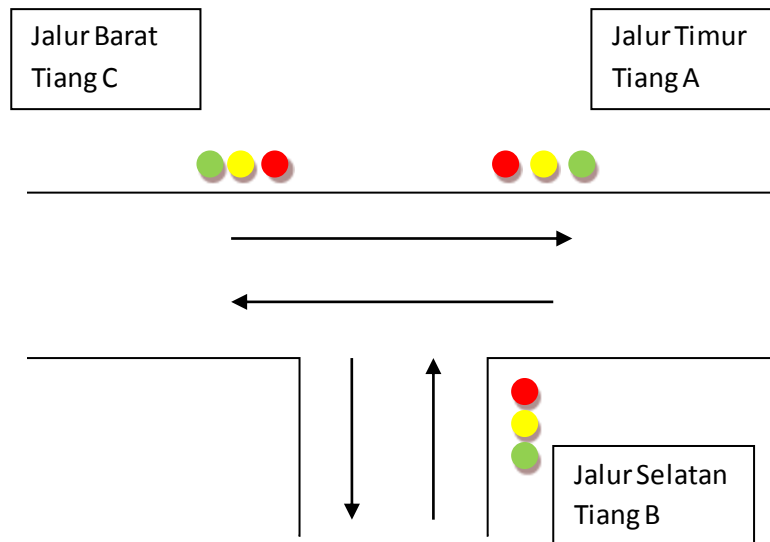
3.1.1. Implementasi Miniatur Traffic Light

Aplikasi ini bekerja dengan nyalanya lampu LED secara bergantian, waktu nyala lampu merah disetiap jalur sangat mempengaruhi oleh lamanya penyalan lampu kuning dan hijau (masing – masing berbeda). Perbedaan waktu nyala lampu hijau ini dilakukan untuk menentukan adanya prioritas pada jalur tertentu, berikut tabel lamanya waktu penyalan pada tiap – tiap jalur:

Tabel 3. Waktu Lampu Lalu Lintas

No	Warna Lampu LED	Waktu (detik)
1.	Lampu LED Merah	2 Detik
2.	Lampu LED Kuning	1 Detik
3.	Lampu LED Hijau	4 Detik

Untuk pengaturan lampu lalu lintas pertigaan pada sistem ini, jika lampu lalu lintas sebelah selatan dan barat dalam posisi berhenti (lampu menyala warna merah) maka lampu sebelah timur dalam posisi jalan (lampu menyala warna hijau),



Gambar 2.6 Skema Lampu Lalu Lintas Pertigaan

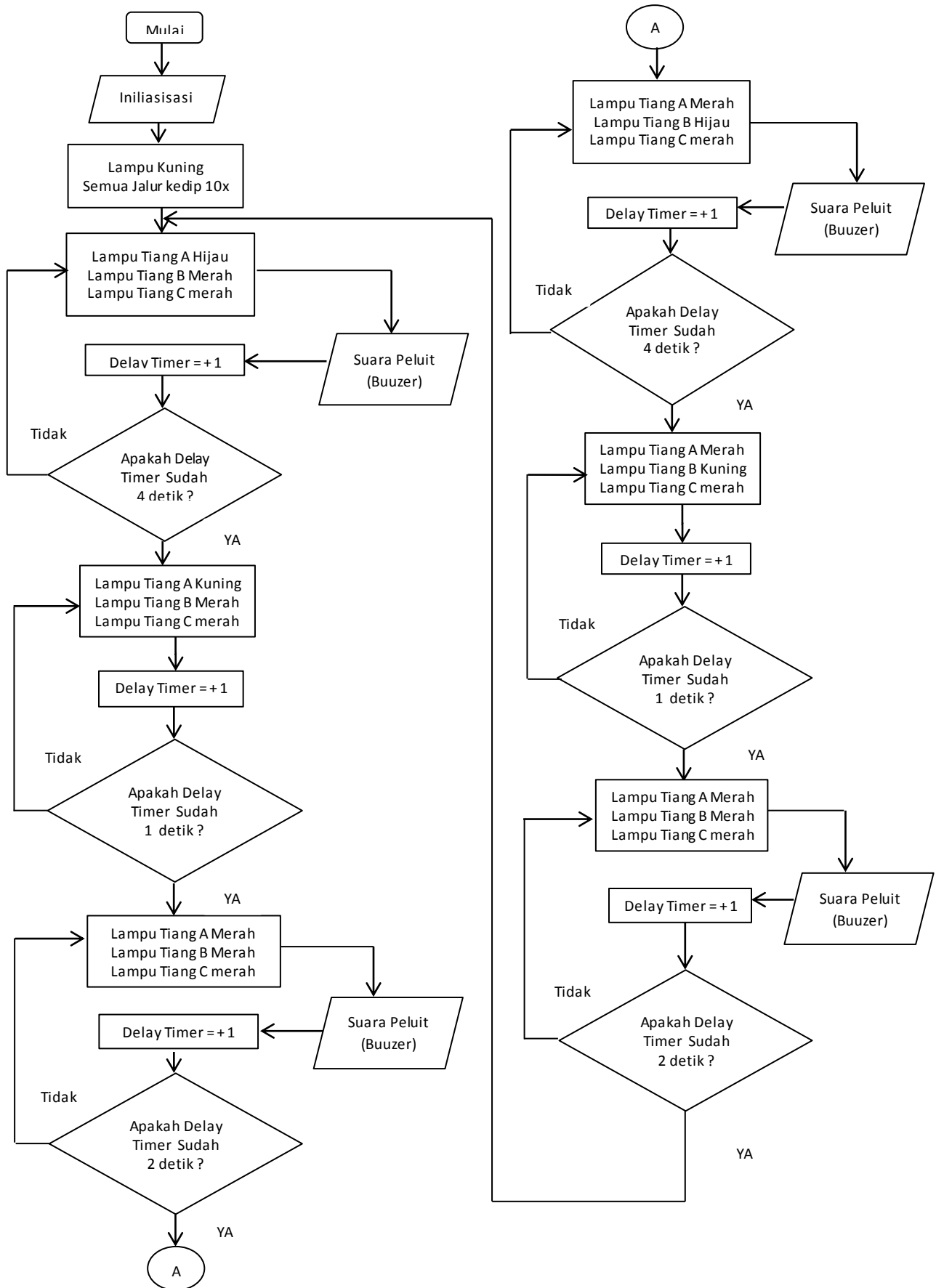
Sistem pengaturan lampu lalu lintas ini dirancang untuk simulasi diruas jalan pertigaan dengan menggunakan Aduino Uno yang didalamnya terdapat memori untuk menyimpan program yang akan dijalankan pada aplikasi *traffic light*.

Aplikasi *traffic light* ini akan berfungsi setelah di berikan program dari software Arduino dan kemudian akan diupload ke perangkat keras Arduino Uno. Jika program aplikasi *traffic light* ini sudah benar dan terupload, pertama akan menyalakan lampu warna hijau dijalur timur tiang A, dan akan melakukan interupsi, yaitu penghentian sementara pada jalur yang sedang berjalan untuk melayani jalur selanjutnya.

3.2. Implementasi Perangkat Lunak

3.2.1. Pembahasan Diagram Alir Program Utama

Pada gambar 2.7 dibawah ini merupakan system diagram alir program utama, ada beberapa fungsi atau penjelasan pada gambar dibawah ini yaitu :



Gambar 2.7 Diagram Alir Program Utama

1. Inisialisasi

Pada flowchart dapat dijelaskan pada saat mulai maka program akan membaca inputan maka kemudian akan diproses pada INISIALISASI yang ada pada flowchart kemudian masukan yang telah diproses tersebut akan diproses pada masing – masing program lampu led. Disini lampu led bertujuan sebagai sebuah pengambil keputusan dari masukan yang bernilai *high*, apabila masukanya sesuai dengan program maka akan menghasilkan keluaran, jika tidak sesuai pembandingan akan kembali membandingkan sampai sesuai dengan program.

2. Lampu Kuning Semua Jalur Kedip 10x

Pada program awal dilakukan inisialisasi dan kemudian untuk memulai nyala lampu *Traffic Light*, lampu kuning semua jalur akan berkedip 10x (HIGH) selama 1 detik dan kemudian sampai semua lampu kuning kembali off atau dengan keadaan *LOW*. Dalam hal ini semua jalur lampu kuning berkedip, pada lampu hijau dan merah disemua jalur dalam kondisi *LOW* atau dalam keadaan Off.

3. Lampu Tiang A Hijau, Lampu Tiang B Merah dan Lampu Tiang C Merah

Lampu *traffic light* bekerja secara bergantian pada tiap jalur sesuai dengan waktu yang sudah ditentukan dengan urutan menyala lampu hijau, lampu kuning dan lampu merah. Untuk lampu *traffic light* menggunakan LED untuk menampilkan simulasi nyala lampu lalu lintas, dimana warna LED yang digunakan adalah merah, kuning dan hijau, sebanyak 9 LED.

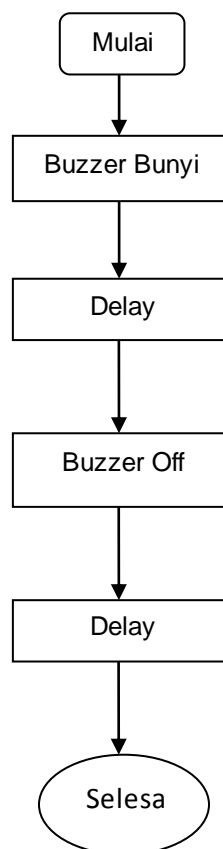
Adapun urutan dan maksud nyala lampu sebagai berikut :

- a. Nyala merah : waktu yang diperuntukan bagi pengguna jalan untuk berhenti.
- b. Nyala kuning : waktu yang diperlukan untuk memberikan peringatan kepada pengguna jalan bahwa nyala lampu akan berganti.
- c. Nyala hijau : waktu yang diperuntukan bagi pengguna jalan untuk melewati persimpangan.

4. Suara Peluit (BUZZER)

Pada proses ini buzzer sebagai untuk mengeluarkan suara atau bunyi peluit disaat lampu LED hijau dalam keadaan HIGH dan lampu LED merah HIGH sampai lampu LED hijau dan merah kembali dalam keadaan LOW. Dan diulang – ulang disetiap jalur secara bergantian nyala *traffic light*.

3.2.2. Diagram Alir



Gambar 2.8 Diagram Alur Program Buzzer

Dari gambar 2.7 tersebut adalah diagram alir dari program buzzer dimana akan mengeluarkan suara atau bunyi dan kemudian akan kembali Off (tidak bunyi) setelah delayMicroseconds atau untuk tundaan eksekusi baris program selanjutnya dalam microseconds. Dan program kembali ke inisialisasi proses suara peluit bunyi secara berulang – ulang.

3.2.3. Membuat Program Menggunakan Audio dan Pengujiannya

Pada pembuatan program ini akan menggunakan fungsi delay pada program. Program bertujuan untuk membuat buzzer menyala dan mati. Konfigurasi nyala dan mati buzzer dengan sela waktu tertentu ini akan membuat suara seperti peluit. Membuat program audio dengan menggunakan sketch lalu melakukan compiling kode kemudian mengupload kode ke board Arduino Uno bila kode tersebut berhasil diunggah, akan terdengar suara seperti peluit. Untuk membunyikan audio dengan nada frekuensi dan durasi tertentu di Arduino, yaitu dengan menggunakan fungsi `tone()`. Fungsi `tone()` ini menerima 2 atau 3 parameter sebagai masukannya. Dengan kode sebagai berikut :

```

int speakerPin = 12;
int length = 13;
char notes [] = {"ccc"};
int beats [] = { 2,5,10};
int tempo = 100;
void playTone(int tone, int duration) {
    for (long i = 0; i < duration * 1000L; i += tone * 2) {
        digitalWrite(speakerPin, HIGH);
        delayMicroseconds(tone);
        digitalWrite(speakerPin, LOW);
        delayMicroseconds(tone);
    }
}

void playNote(char note, int duration) {
    char names[] = { 'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C' };
    int tones[] = {10000, 1700, 1519, 1432, 1275, 1136, 1014,
9500 };
    // play the tone corresponding to the note name
    for (int i = 0; i < 8; i++) {
        if (names[i] == note) {
            playTone(tones[i], duration);
        }
    }
}

void setup () {
    pinMode(speakerPin, OUTPUT);
}
void loop () {
    for (int i = 0; i < length; i++) {

        if (notes[i] == ' ') {
            delay(beats[i] * tempo); // rest
        } else {
            playNote(notes[i], beats[i] * tempo);
        }
        delay(tempo / 4);
    }
}

```

Penjelasan dalam sintaks ini adalah pada fungsi pin 12 merupakan penginialisasi untuk keluaran. Dan pada fungsi *void loop()* yaitu untuk pengulangan untuk menampilkan nilai output pada pin 12, jika nilai input nya 1 maka keluarannya NYALA atau HIGH dan jika nilainya inputnya 0, maka keluarannya MATI atau

LOW. Sedangkan fungsi delay adalah waktu penampilan untuk keluarannya. Setiap sketch Arduino wajib memiliki fungsi `setup()` dan `loop()`. Fungsi `setup()` dipanggil hanya sekali saat pertama kali program berjalan. `pinMode` fungsi yang berguna untuk memberitahu Arduino bahwa pin pada board akan digunakan sebagai input atau output. Fungsi `loop()` adalah program utama yang dipanggil secara continue selama Arduino menyala (dialiri power). Setiap perintah dalam fungsi `loop()` akan dipanggil satu persatu sampai perintah terakhir dalam blok `loop` dicapai, lalu Arduino akan kembali ke awal perintah diblok fungsi `loop()`. Dengan struktur seperti program diatas akan menjalankan kode yang ada didalam kurung kurawal jika kondisi `TRUE`, dan jika tidak (`FALSE`) maka akan diperiksa apakah kondisi pada `else if` dan jika kondisinya `FALSE` maka kode pada `else` yang akan dijalankan.

3.2.4. Pengujianya

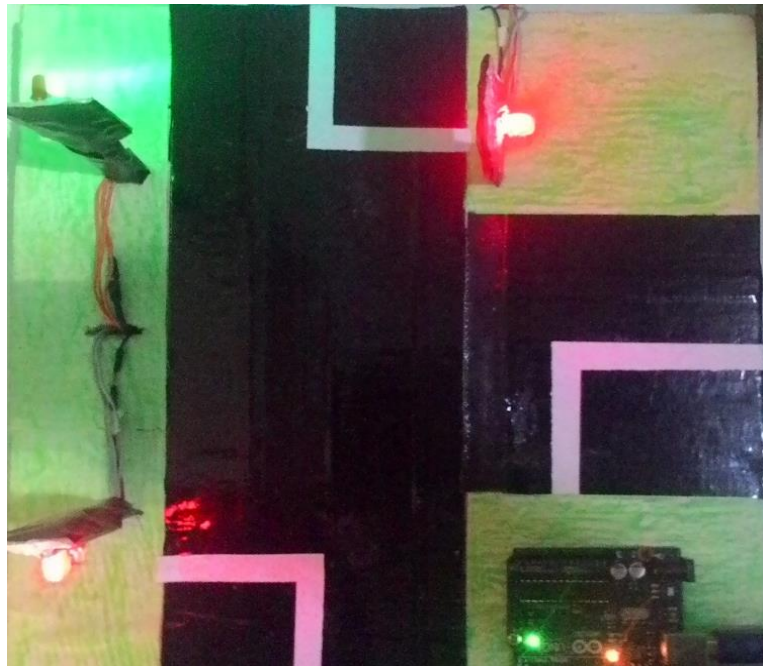
Pada pengujian audio (buzzer) ini dilakukan dengan memberikan program yang sebelumnya sudah dibuat disketch Arduino (software arduino) dengan cara mengupload kode ke dalam board Arduino, kemudian audio akan mengeluarkan suara/bunyi peluit. Berdasarkan pengujian ini dapat disimpulkan bahwa program audio bekerja dengan baik, namun suara peluit belum sempurna seperti bunyi peluit sungguhan.

3.3. Pengujian Lampu Traffic Light

Pada pengujian dibagian ini dilakukan dengan memberikan program pada board Arduino Uno. Setelah memberikan program pada board Arduino Uno maka rangkaian aplikasi *traffic light* pada pertigaan akan menyala secara bergantian, pada tiap jalur sesuai dengan waktu yang sudah ditentukan dengan urutan menyala lampu hijau, lampu kuning dan lampu merah.

Nyala padamnya lampu dikendalikan dengan program pada Arduino Uno yang sudah dirancang pada awal pembuatan sampai hasil akhir yang dicapai. Pada saat padam dan menyalanya masing – masing lampu ini dijelaskan sebagai berikut :

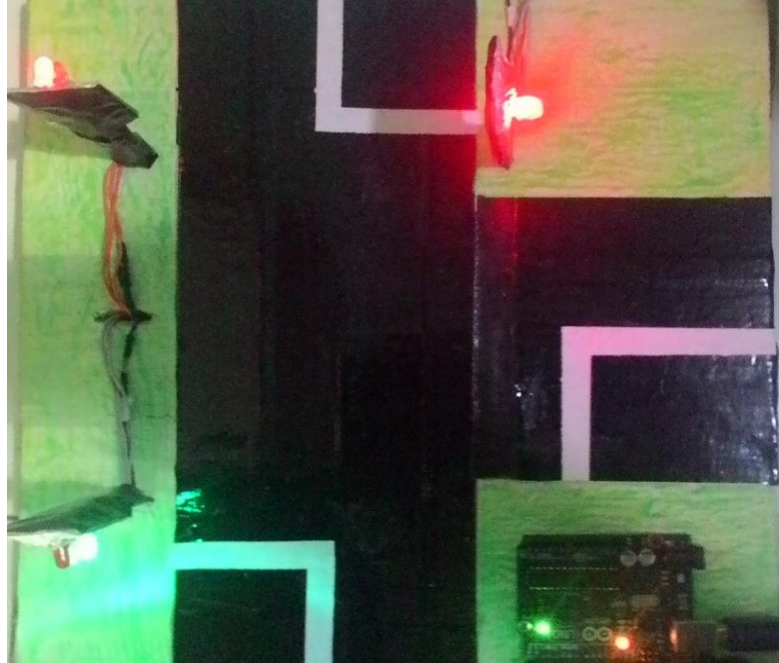
- Awalnya lampu merah menyala selama 2 detik, menandakan kendaraan untuk berhenti, kemudian 4 detik sebelum lampu hijau menandakan bahwa pengendara harus bersiap – siap, saat lampu hijau menyala, saat lampu hijau menyala, kedua lampu merah dan kuning dengan keadaan mati, lalu 1 detik sebelum lampu merah menyala, maka lampu kuning menyala dan seterusnya akan diulang – ulang nyala lampu lalu lintas secara bergiliran di jalur selanjutnya.



Gambar 2.9 Pengujian di tiang A lampu hijau menyala



Gambar 2.10 Di tiang B lampu hijau menyala



Gambar 2.11 Di tiang C lampu hijau menyala

3.3.1. Pengujian Rangkaian Keseluruhan

Pengujian keseluruhan dilakukan setelah semua rangkaian selesai dibuat dan sudah diisi program. Saat board Arduino Uno telah dihubungkan dengan catu daya dan juga konverternya, maka aplikasi lampu lalu lintas pada pertigaan jalan dapat di amati tiap – tiap tiang A, B dan tiang C dan laptop sebagai penghubung catu daya dengan board Arduino Uno. Ketika komputer atau laptop mengirimkan data berupa program *traffic light* pertigaan ke board Arduino Uno maka secara otomatis program akan menyesuaikan dengan data yang tersimpan di dalam chip Arduino tersebut akan menyalakan skema *traffic light* pertigaan. Apabila program Arduino telah merubah waktu delay lampu hijau menyala suara peluit, kuning dan merah menyala suara peluit maka dapat diartikan bahwa skema *traffic light* pertigaan berjalan dan program sudah benar.