

BAB III

PERANCANGAN ALAT DAN PROGRAM

3.1 Gambaran Umum Alat

Alat control atap ini menggunakan Arduino Uno sebagai pengkoreksi data yang dihasilkan oleh sensor, sensor yang digunakan yaitu menggunakan sensor hujan. Sensor ini yang nantinya akan mendeteksi air hujan, kemudian sensor ini akan memberi keluaran dan akan dikoreksi oleh Mikrokontroler, kemudian akan memberi inputan kepada modul L298N, sehingga motor akan bekerja menutup atap stadion dan LED menyala sebagai output.

3.2 Spesifikasi Alat

Alat – alat yang digunakan pada perancangan sistem otomatisasi atap stadion ini memiliki spesifikasi sebagai berikut:

Tabel 3.1 Spesifikasi Alat

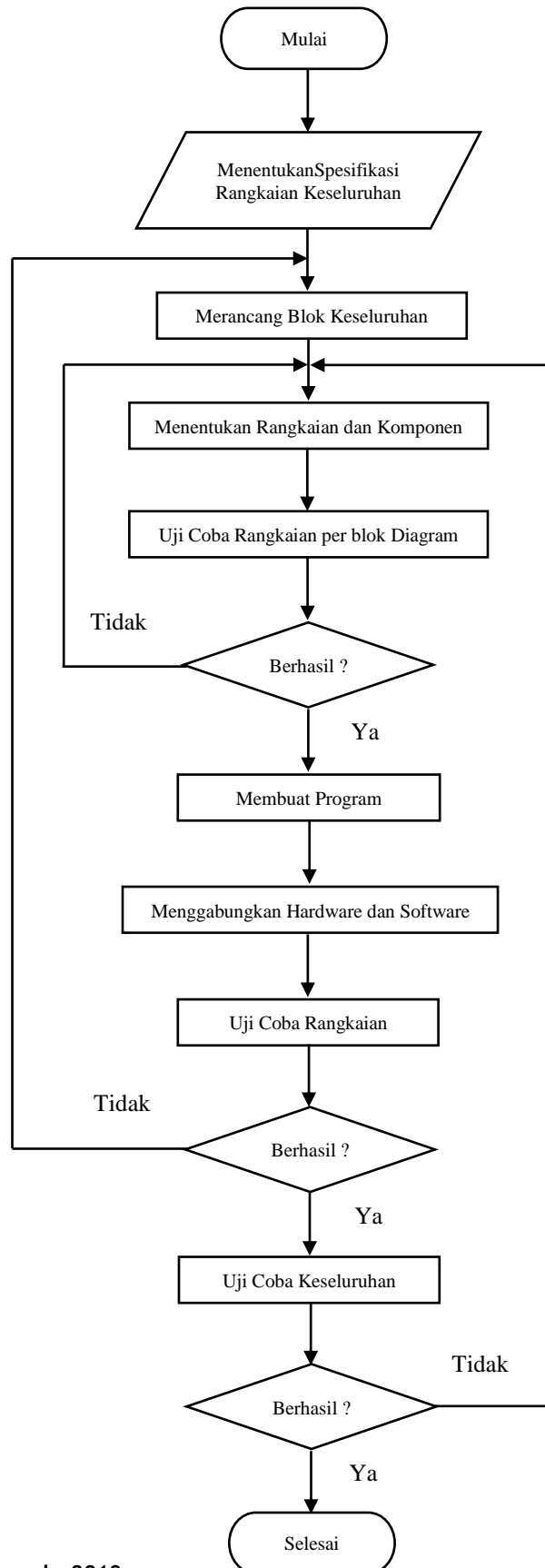
Komponen	Keterangan
Power Supply	Input 220V AC diturunkan menjadi 12V AC disearahkan sehingga dapat menghasilkan tegangan tetap 12V DC – 5V DC
Mikrokontroler Arduino UNO	Mikrokontroler ATmega328 beroperasi 5 V , Input Voltage (recommended):7 – 12 V Input Voltage (limits) : 6 – 20 V Digital I/O pins : 14 (6 provide PWM output) Analog input pins : 6 DC current per I/O pin : 40 mA DC current for 3.3 V pin : 50 mA Clock speed : 16 MHz
Motor DC	Dc motor FF-130S (R)/H Dimension : 40.7 x 19.7 x 42.9 mm approx. Stall torque : 0.57-0.65A Operating speed : 4280-4920 rpm Operation Voltage : 4.5-12 V Running Current 500mA 900mA (6 V) Stall current 2.5 A (6V)

Sensor Hujan	<p>Sensor ini bermaterial dari FR-04 dengan dimensi 5cm x 4cm berlapis nikel dan dengan kualitas tinggi pada kedua sisinya</p> <p>Pada lapisan module mempunyai sifat anti oksidasi sehingga tahan terhadap korosi</p> <p>Tegangan kerja masukan sensor 3.3V – 5V</p> <p>Menggunakan IC comparator LM393 yang stabil</p> <p>Output dari modul comparator dengan kualitas sinyal bagus lebih dari 15mA</p> <p>Dilengkapi lubang baut untuk instalasi dengan modul lainnya</p> <p>Terdapat potensiometer yang berfungsi untuk mengatur sensitifitas sensor</p> <p>Terdapat 2 Output yaitu digital (0 dan 1) dan analog (tegangan)</p> <p>Dimensi PCB yaitu 3.2 cm x 1.4 cm</p>
L298N	<p>Menggunakan IC L298N (Double H bridge Drive Chip)</p> <p>Tegangan minimal untuk masukan power antara 5V-35V</p> <p>Tegangan operasional : 5V</p> <p>Arus untuk masukan antara 0-36mA</p> <p>Arus maksimal untuk keluaran per Output A maupun B yaitu 2A</p> <p>Daya maksimal yaitu 25W</p> <p>Dimensi modul yaitu 43 x 43 x 26mm</p> <p>Berat : 26g</p>

3.3 Tahapan Perencanaan Pembuatan Alat

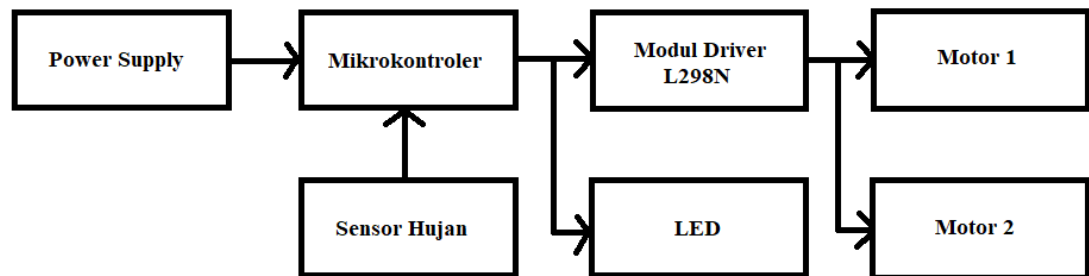
Dalam perencanaan sistem ini akan dibahas tentang kebutuhan-kebutuhan yang harus dipenuhi agar alat pendeteksi dapat bekerja sesuai dengan sistem yang diinginkan. Maka dari itu dalam merencanakan pembuatan alat ini diperlukan langkah-langkah sebagai berikut:

1. Membuat diagram blok rancangan alat pendeteksi logam
2. Membuat *flowchart* kerja Arduino
3. Mengimplementasikan rangkaian dengan program
4. Pengujian alat
5. Analisis dan simpulan
6. Membuat laporan



3.3.1 Perancangan Alat

Dalam perancangan sebuah alat yang meliputi blok diagram rangkaian dan penjelasan dari masing-masing blok diagram rangkaian pada rancang bangun *Prototype* atap otomatis stadion berbasis Arduino.



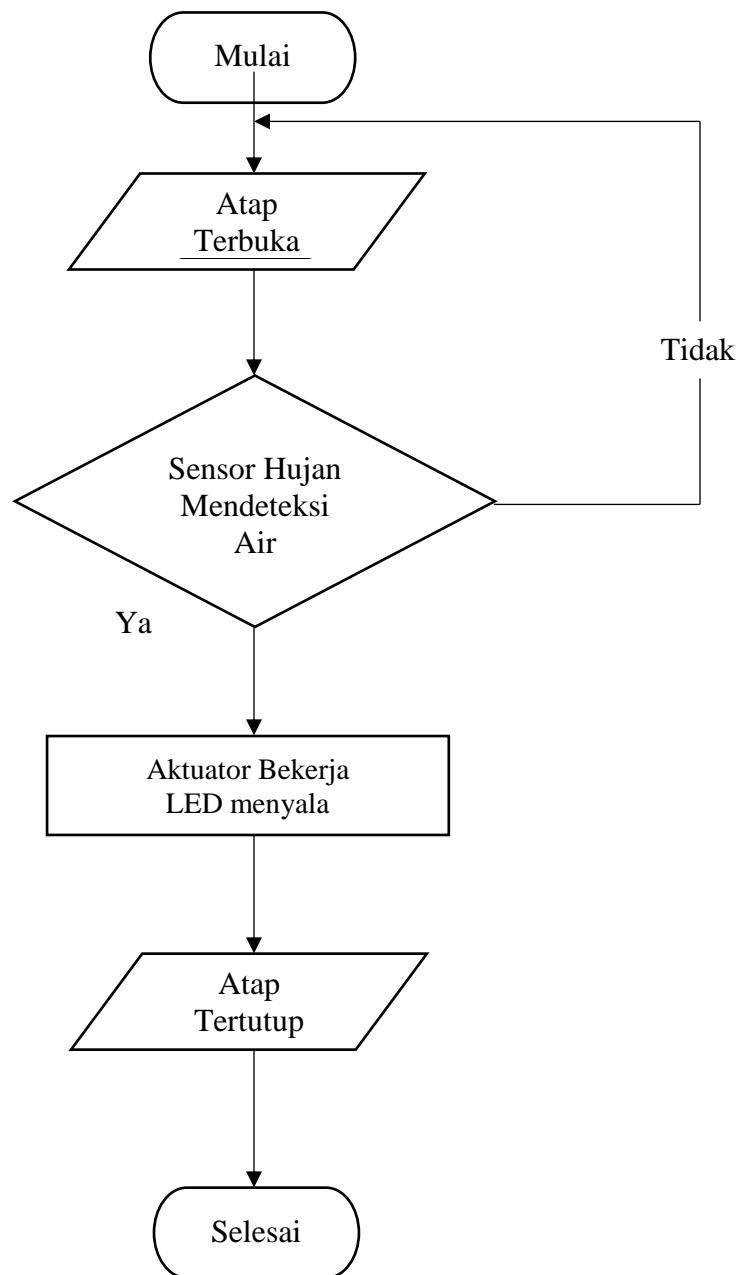
Gambar 3.2 Diagram Blok

Berdasarkan gambar diatas dapat disimpulkan bahwa proses penutupan atap secara otomatis berawal dari sensor hujan yang mendeteksi adanya air lalu memberikan sinyal *output* pada mikrokontroler yang kemudian akan diolah sedemikian rupa untuk memberikan output sinyal pada modul driver untuk memutar 2 motor menjadi kondisi tertutup kemudian mikrokontroler memberikan output juga untuk menyalakan lampu LED.

3.3.2 Flowchart

Flowchart ‘diagram alir’ telah dikenal luas dan umum digunakan untuk menggambarkan alur proses atau langkah-langkah secara berurutan. Banyak digunakan antara lain untuk menggambarkan proses bisnis, langkah-langkah penyelesaian masalah, atau *Standard Operational Procedure* (SOP).

Flowchart berikut menunjukkan proses dari sebuah awal mula atap terbuka kemudian sensor hujan yang akan membaca objek akan memberikan sinyal pada mikrokontroler, pada tahap ini mikrokontroler akan memproses data dari pendeteksian sensor hujan, kemudian mikrokontroler akan memberikan sinyal *output* 12V DC kepada modul driver kemudian modul driver akan menggerakkan aktuatuor sehingga atap menutup dan LED menyala dengan tegangan 5V.



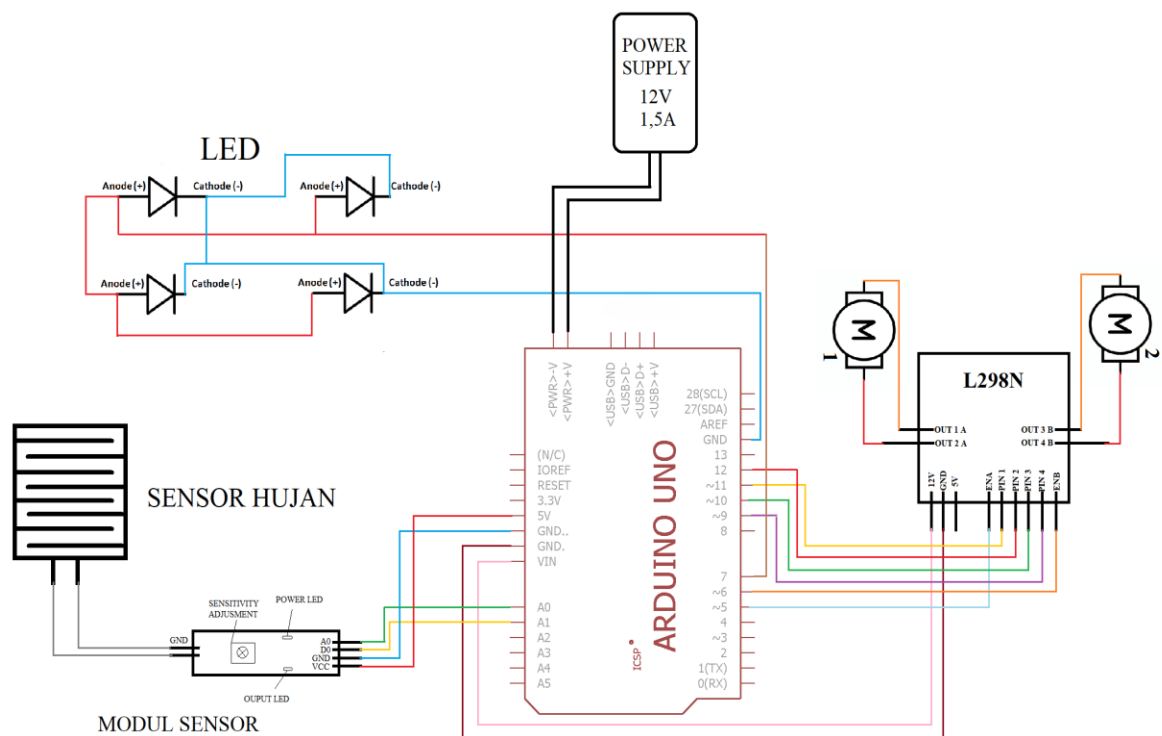
Gambar 3.3 *Flowchart* kerja alat

3.4 Langkah Pembuatan Alat

Langkah pembuatan alat pada tugas akhir ini terdiri dari perancangan dan pemograman

3.4.1 Perancangan Alat

Pada gambar 3.4 menunjukkan gambar rangkaian schematic. Dari gambar sebelumnya yaitu gambar 3.2 telah dijelaskan bahwa sinyal masukan yang diberikan oleh sensor akan diproses oleh suatu alat pengendali, yaitu sebuah Mikrokontroler Arduino Uno. Sinyal didapat dari sebuah sensor, berupa Sensor Hujan.



Gambar 3.4 Schematic

Perancangan skema elektronika dalam alat buka tutup atap stadion otomatis ini menggunakan mikrokontroler Arduino Uno untuk mengeluarkan pulsa PWM (Pulse Width Modulation). Kemudian sensor adalah transduser yang merubah kondisi sekitar menjadi besaran listrik yang dikirimkan ke Arduino Uno untuk diproses menjadi PWM. Ketika Sensor Hujan mendeteksi adanya air maka Mikrokontroler akan memberikan sinyal kepada modul driver, lalu modul driver akan menyalurkan tegangan kepada motor DC yang sudah terpasang dengan atap stadion, maka atap pun akan menutup dan LED akan menyala, begitupun sebaliknya saat sensor sudah tidak mendeteksi adanya air maka atap akan terbuka kembali dan lampu ikut mati.

3.4.2 Pemrograman

Hal terpenting dalam proses perencanaan pembuatan alat atap otomatis ini adalah program dari Mikrokontroler itu sendiri karena yang dapat mengolah semua sistem agar dapat bekerja dengan semestinya tergantung bagaimana kita memprogramnya, dalam proses memprogram kita dapat membuatnya sesuai dengan kebutuhan.

a. Proses Data

Data digital yang dikeluarkan oleh sensor hujan, kemudian diproses oleh mikrokontroler yang dimana mikrokontroler tersebut mengolah data sesuai program yang penulis buat.

b. Tampilan Program

- Langkah 1 yang telah diproses kemudian ditampilkan.



```

DEJET_PWM | Arduino 1.8.5
File Edit Sketch Tools Help

DEJET_PWM

//Inialisasi pin sensor hujan dan ldr
const int pinHujan = A1;

//Motor
const int motorPin3 = 10;
const int motorPin4 = 9;
const int motorPin1 = 11;
const int motorPin2 = 12;
const int ENA = 5;
const int ENB = 6;

const int Lampu = 7;
int data ;

void setup()
{

```

Gambar 3.5 Tampilan Penulisan Program Menggunakan *Software* Arduino

- Langkah 2 yang telah diproses kemudian ditampilkan.

```

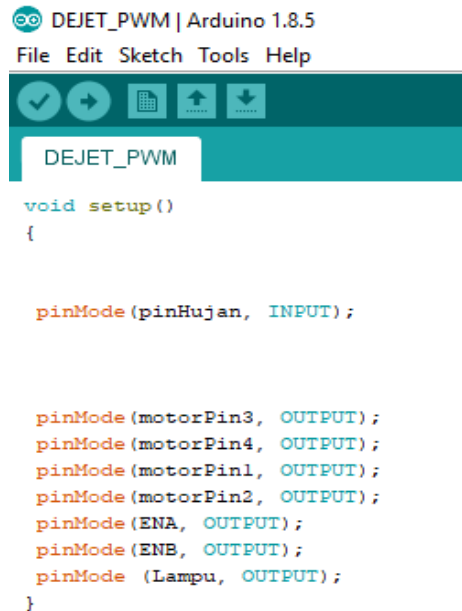
pinMode(pinHujan, INPUT);

pinMode(motorPin3, OUTPUT);
pinMode(motorPin4, OUTPUT);
pinMode(motorPin1, OUTPUT);
pinMode(motorPin2, OUTPUT);
pinMode(ENA, OUTPUT);
pinMode(ENB, OUTPUT);
pinMode(Lampu, OUTPUT);
}

```

Gambar 3.6 Tampilan Penulisan Program Menggunakan *Software* Arduino

- Langkah 3 yang telah diproses kemudian ditampilkan.



```

DEJET_PWM | Arduino 1.8.5
File Edit Sketch Tools Help

DEJET_PWM

void setup()
{

  pinMode(pinHujan, INPUT);

  pinMode(motorPin3, OUTPUT);
  pinMode(motorPin4, OUTPUT);
  pinMode(motorPin1, OUTPUT);
  pinMode(motorPin2, OUTPUT);
  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(Lampu, OUTPUT);
}

```

Gambar 3.7 Tampilan Penulisan Program Menggunakan *Software* Arduino

- Langkah 4 yang telah diproses kemudian ditampilkan.

```

void loop()
{
  data=analogRead(pinHujan);

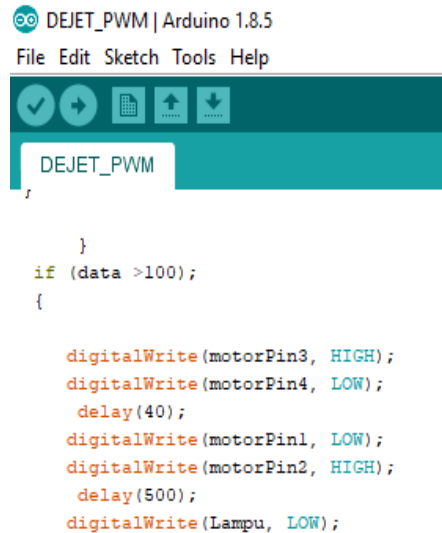
  if (data <= 40)
  {

    digitalWrite(motorPin3, LOW);
    digitalWrite(motorPin4, HIGH);
    delay(40);
    digitalWrite(motorPin1, HIGH);
    digitalWrite(motorPin2, LOW);
    analogWrite(ENA, 150);
    analogWrite(ENB, 150);
    delay(500);
  }
}

```

Gambar 3.8 Tampilan Penulisan Program Menggunakan *Software* Arduino

- Langkah 5 yang telah diproses kemudian ditampilkan.



```

}
if (data >100);
{
    digitalWrite(motorPin3, HIGH);
    digitalWrite(motorPin4, LOW);
    delay(40);
    digitalWrite(motorPin1, LOW);
    digitalWrite(motorPin2, HIGH);
    delay(500);
    digitalWrite(Lampu, LOW);

```

Gambar 3.9 Tampilan Penulisan Program Menggunakan *Software* Arduino

- Langkah 6 yang telah diproses kemudian ditampilkan.

```

while(1) {
digitalWrite(motorPin3, HIGH);
digitalWrite(motorPin4, HIGH);
digitalWrite(motorPin1, HIGH);
digitalWrite(motorPin2, HIGH);

data=analogRead(pinHujan);
if (data <= 40) {break;}
}
}
}

```

Gambar 3.10 Tampilan Penulisan Program Menggunakan *Software* Arduino

Program dibuat dengan menggunakan bahasa C atau bahasa Arduino yang dibuat menggunakan *compiler* Arduino IDE, *Software* inilah yang paling utama, membantu menjembatani antara bahasa mesin yang begitu rumit sehingga menjadi bahasa dan logic yang lebih mudah dimengerti manusia. Pemilihan *compiler* Arduino IDE dikarenakan penggunaannya yang mudah karena *library* yang digunakan lengkap dan bahasa C yang digunakan lebih mudah dipahami oleh manusia merupakan tampilan dari *software* Arduino.

Selanjutnya penulisan program pada Arduino IDE, sebelum penulisan program perlu dilakukanya inisialisasi menggunakan *library* yang telah disediakan oleh arduino. Tampilan penulisan program menggunakan *software* arduino..