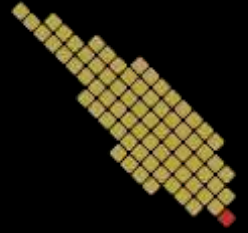


Laporan Praktikum Embedded System (IF3122)

Program Studi Teknik Informatika
Institut Teknologi Sumatera
2021



Judul : Sistem Pengendali Atap Jemuran Otomatis
Nama : - Mahdia Nisrina Maharani M (118140025)
- Romantika Banjarnahor (118140045)
- Juni Rianto Simamora (118140047)
- Syabana Minggu N (118140122)
- Ezra Kornel Naibaho (118140171)

Pernyataan Orisinalitas

Dengan ini penulis bertanggung jawab sepenuhnya atas isi dari dokumen ini dan menyatakan bahwa seluruh isi dokumen ini adalah hasil karya penulis sendiri, dan setiap karya orang lain yang digunakan dalam dokumen ini telah diparafrase dan sudah disitasi sesuai dengan ketentuan yang ada.

Lampung Selatan, 2 Januari 2022

Kelompok 5

Daftar Isi

1. Pendahuluan	2
1.1 Latar Belakang	2
1.2 Schematic Diagram/Arsitektur Sistem	3
2. Landasan Teori	4
2.1. Mikrokontroler	4
2.2. Sensor	4
2.3. Sensor Cahaya (Light Dependent Resistor)	4
2.4. Sensor Ultrasonik	5
2.5. Sensor Air Hujan	5
2.6. Servo	5
3. Implementasi	6
3.1 Struktur Kode	6
3.2 Foto dan Hasil Implementasi	10
Referensi	14

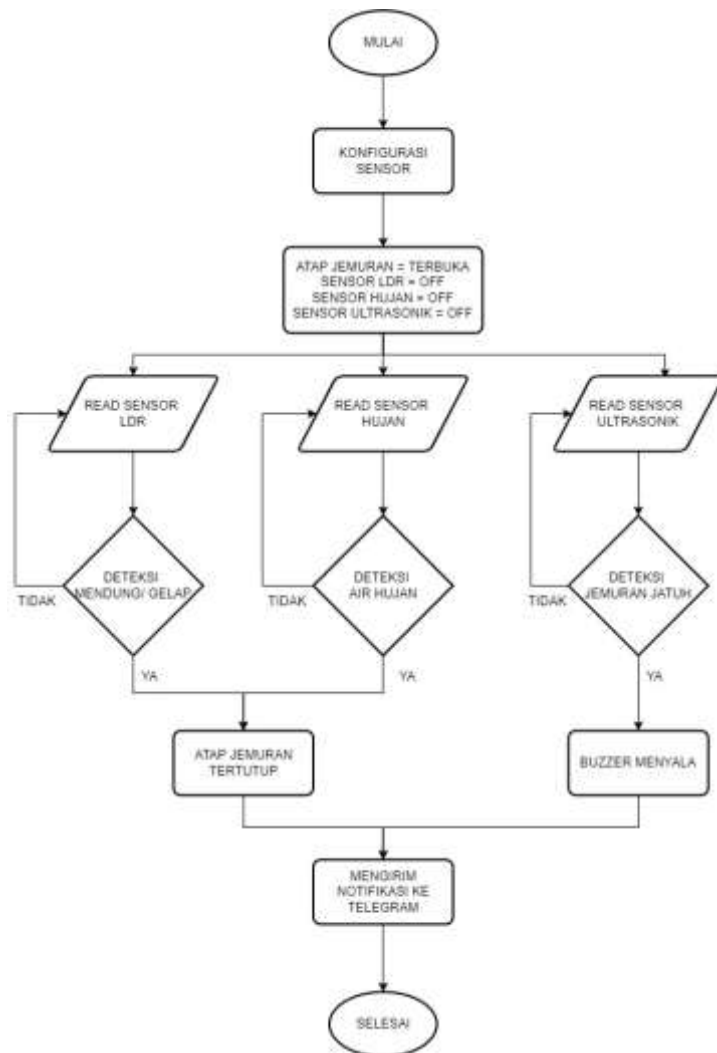
1. Pendahuluan

1.1 Latar Belakang

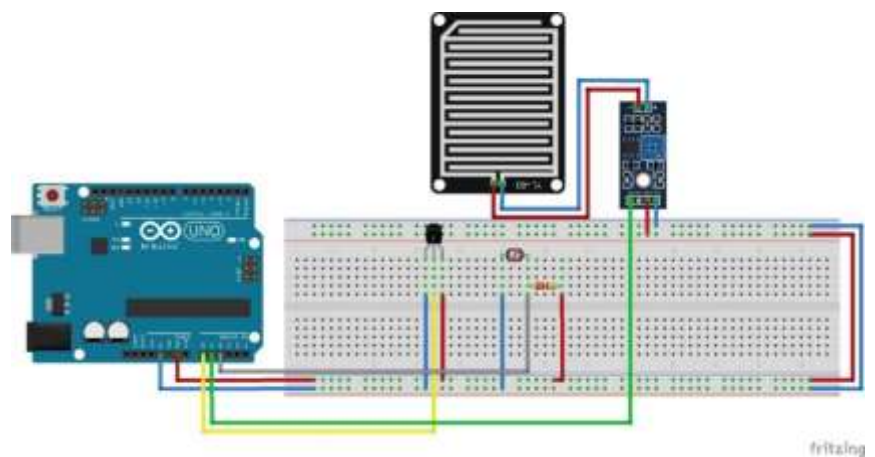
Mencuci pakaian adalah salah satu kegiatan yang sering dilakukan oleh masyarakat Indonesia dalam kehidupan sehari-hari. Kegiatan mencuci selalu berhubungan dengan kegiatan menjemur pakaian dengan tujuan untuk mengeringkan. Sebagian besar masyarakat beraktivitas di luar rumah dan mereka akan meninggalkan jemuran mereka sepanjang hari. Untuk menjaga jemuran mereka agar tetap kering dan terhindar dari hujan dan udara lembab pada malam hari ketika mereka tidak di rumah maka sangat diperlukan alat yang dapat mengendalikan atap jemuran yang dapat terbuka dan tertutup otomatis. Indikator yang mempengaruhi alat pengendali ini adalah sensor suhu, sensor cahaya, dan sensor air hujan. Berdasarkan permasalahan tersebut, peneliti merancang sebuah alat pengendali atap jemuran otomatis yang dikendalikan oleh mikrokontroler.

1.2 Schematic Diagram/Arsitektur Sistem

a. Desain Flowchart



b. Schematic Diagram



2. Landasan Teori

2.1. Mikrokontroler

Mikrokontroler merupakan alat yang berukuran mikro yang berfungsi sebagai pengendali atau pengontrol. Sebelum adanya mikrokontroler maka yang digunakan adalah mikroprosesor. Namun mikrokontroler lebih unggul dibandingkan dengan mikroprosesor. Alasan dari keunggulan tersebut ada beberapa yakni tersedianya I/O I/O dalam mikrokontroler sudah tersedia sementara pada mikroprosesor dibutuhkan IC tambahan untuk menangani I/O tersebut. Alasan lainnya adalah memori Internal Memori merupakan media untuk menyimpan program dan data sehingga mutlak harus ada. Mikroprosesor belum memiliki memori internal sehingga memerlukan IC memori eksternal [1].

2.2. Sensor

Sensor adalah alat yang mampu menangkap fenomena fisika atau kimia kemudian mengubahnya menjadi sinyal elektrik baik arus listrik ataupun tegangan. Fenomena fisik yang mampu menstimulasi sensor untuk menghasilkan sinyal elektrik meliputi temperatur, tekanan, gaya, medan magnet cahaya, pergerakan dan sebagainya [2].

2.3. Sensor Cahaya (Light Dependent Resistor)

Light Dependent Resistor (LDR) terdiri dari sebuah cakram semikonduktor yang mempunyai dua buah elektroda pada permukaannya. Resistansi LDR berubah seiring dengan perubahan intensitas cahaya yang mengenainya. Dalam keadaan gelap resistansi LDR sekitar 10 Ω dan dalam keadaan terang sebesar 1 k Ω atau kurang. LDR terbuat dari bahan semikonduktor seperti cadmium sulfide. Dengan bahan ini energi dari cahaya yang jatuh menyebabkan lebih banyak muatan yang dilepas atau arus listrik meningkat. Artinya resistansi bahan telah mengalami penurunan. LDR digunakan untuk mengubah energi cahaya menjadi energi listrik. Berikut cara kerja sensor LDR. Pada saat gelap atau cahaya redup, bahan dari cakram tersebut menghasilkan elektron bebas dengan jumlah yang relatif kecil. Sehingga hanya ada sedikit elektron untuk mengangkut muatan elektrit. Artinya pada saat cahaya redup, LDR menjadi konduktor yang buruk, atau bisa disebut juga LDR memiliki resistansi yang besar pada saat

gelap atau cahaya redup [3].

2.4. Sensor Ultrasonik

Sensor ultrasonic merupakan merupakan salah satu sensor yang memiliki fungsi untuk mengubah besaran fisis atau bunyi menjadi besaran listrik dan juga sebaliknya. Cara kerja pada sensor ini adalah dengan cara pantulan suatu gelombang suara yang dapat digunakan untuk menafsirkan eksistensi atau jarak suatu pada benda dengan frekuensi tertentu

2.5. Sensor Air Hujan

Sensor air hujan dirancang untuk mendeteksi air pada saat turun hujan tetapi juga dapat digunakan untuk mendeteksi level ketinggian air. Rangkaian sensor air hujan dapat dibuat dengan menggunakan komponen resistor sebagai komponen utama dan elektroda sebagai pendeteksi air [2].

2.6. Servo

Servo merupakan motor listrik yang dimana sistem dari servo ini yaitu closed loop. Sistem ini digunakan untuk mengendalikan akselerasi dan kecepatan pada sebuah motor listrik dengan keakuratan yang tinggi.

Pada umumnya motor servo dapat terbagi menjadi dua jenis berdasarkan beban arusnya, yaitu:

Motor Servo AC

Motor Servo DC

3. Implementasi

3.1 Struktur Kode

Dibawah ini adalah struktur kode program yang digunakan pada penelitian kali ini :

```
1  #include <Servo.h>
2
3  #define trig 3
4  #define echo 4
5  unsigned long duration;
6  unsigned int distance;
7  String cuaca, rain, jarak;
8
9  const int ldr = A0;
10 const int hujan = A1;
11 int led = 12;
12 int buzzer = 7;
13 Servo servol;
14 int posl = 90;
15
16 int cek = 0;
17
18 unsigned long startHujan = 0;
19 unsigned long startMendung = 0;
20 unsigned long startUltrasonic = 0;
21 unsigned long currentHujan = 0;
22 unsigned long currentMendung = 0;
23 unsigned long currentUltrasonic = 0;
24 unsigned long intervalHujan = 0;
25 unsigned long intervalMendung = 0;
26 unsigned long intervalUltrasonic = 0;
27 int kirimHujan = 0;
28 int kirimMendung = 0;
29 int kirimled = 0;
30 int kirimUltrasonic = 0;
31
```

```
1 void setup() {
2   pinMode(buzzer, OUTPUT);
3   pinMode(trig, OUTPUT);
4   pinMode(echo, INPUT);
5   pinMode(ldr, INPUT);
6   servol.attach(11);
7   servo2.attach(10);
8   Serial.begin(115200);
9   servol.write(90); //Posisi awal servol
10 }
```

```
1 void loop() {
2
3   //start Ultrasonic
4   digitalWrite(trig, LOW);
5   delayMicroseconds(5);
6   digitalWrite(trig, HIGH);
7   delayMicroseconds(5);
8   digitalWrite(trig, LOW);
9   duration = pulseIn(echo, HIGH);
10  distance = duration/58.2;
11  if(distance ≤ 40){
12    if(kirimUltrasonic == 0){
13      jarak = "Ada pakaian jatuh";
14      digitalWrite(buzzer, HIGH);
15      String data = (String)jarak;
16      Serial.println(data);
17      delay(1000);
18      digitalWrite(buzzer, LOW);
19    }
20    kirimUltrasonic++;
21  }
22
23  else if(distance > 40){
24    kirimUltrasonic = 0;
25    jarak = "";
26  }
27  //end Ultrasonik
```

```
1 //start LDR
2 int intensity = analogRead(ldr);
3 int mapLDR = map(intensity, 0, 1023, 1, 3);
4 if(mapLDR == 1){
5   digitalWrite(led, LOW);
6   if(startMendung ≠ 0){
7     startMendung = 0;
8     intervalMendung = 0;
9     currentMendung = 0;
10    kirimMendung = 0;
11    cuaca = "";
12    if(pos1≠90){
13      buka();
14      Serial.println("terbuka");
15    }
16  }
17 }
```



```

1 //start LDR
2 int intensity = analogRead(ldr);
3 int mapLDR = map(intensity, 0, 1023, 1, 3);
4 if(mapLDR == 1){
5     digitalWrite(led, LOW);
6     if(startMendung != 0){
7         startMendung = 0;
8         intervalMendung = 0;
9         currentMendung = 0;
10        kirimMendung = 0;
11        cuaca = "";
12        if(pos1 != 90){
13            buka();
14            Serial.println("terbuka");
15        }
16    }
17 }
18
19 else if(mapLDR == 2){
20     if(startMendung == 0){
21         startMendung = millis();
22         digitalWrite(led, HIGH);
23     }
24     currentMendung = millis();
25     intervalMendung = currentMendung - startMendung;
26     if(intervalMendung ≥ 2000){
27         digitalWrite(led, LOW);
28     }
29
30     if(intervalMendung ≥ 8000 && kirimMendung == 0){
31         cuaca = "Cuaca mendung";
32         String data = (String)cuaca;
33         Serial.println(data);
34         kirimMendung++;
35     }else if(kirimMendung > 0){
36         cuaca = "";
37     }
38
39     if(kirimMendung > 0 && intervalMendung ≥ 8000){
40         if(pos1 == 90){
41             tutup();
42         }
43     }
44 }
45
46 else{
47     if(startMendung == 0){
48         startMendung = millis();
49         digitalWrite(led, HIGH);
50     }
51     currentMendung = millis();
52     intervalMendung = currentMendung - startMendung;
53     if(intervalMendung ≥ 2000){
54         digitalWrite(led, LOW);
55     }
56
57     if(intervalMendung ≥ 8000 && kirimMendung == 0){
58         cuaca = "Cuaca mendung";
59         String data = (String)cuaca;
60         Serial.println(data);
61         kirimMendung++;
62     }else if(kirimMendung > 0){
63         cuaca = "";
64     }
65
66     if(kirimMendung > 0 && intervalMendung ≥ 8000){
67         if(pos1 == 90){
68             tutup();
69         }
70     }
71 }
72 //end LDR

```

```

1 //start Hujan
2 int sensorHujan = analogRead(hujan);
3 int mapHujan = map(sensorHujan, 0, 1023, 0, 3);
4 switch(mapHujan){
5     case 0:
6         if(startHujan == 0){
7             startHujan = millis();
8         }
9         currentHujan = millis();
10        intervalHujan = currentHujan - startHujan;
11        if(intervalHujan ≥ 6000 && kirimHujan == 0){
12            rain = "Rain Warning";
13            String data = (String)rain;
14            Serial.println(data);
15            kirimHujan++;
16        }else if(kirimHujan > 0){
17            rain = "";
18        }
19
20        if(intervalHujan ≥ 6000 && kirimHujan > 0){
21            if(pos1 == 90){
22                tutup();
23            }
24        }
25        break;
26    case 1:
27        if(startHujan == 0){
28            startHujan = millis();
29        }
30        currentHujan = millis();
31        intervalHujan = currentHujan - startHujan;
32        if(intervalHujan ≥ 6000 && kirimHujan == 0){
33            rain = "Rain Warning";
34            String data = (String)rain;
35            Serial.println(data);
36            kirimHujan++;
37        }else if(kirimHujan > 0){
38            rain = "";
39        }
40
41        if(intervalHujan ≥ 6000 && kirimHujan > 0){
42            if(pos1 == 90){
43                tutup();
44            }
45        }
46        break;
47    case 2:
48        if(startHujan ≠ 0){
49            startHujan = 0;
50            intervalHujan = 0;
51            currentHujan = 0;
52            kirimHujan = 0;
53            rain = "";
54        }
55        if(pos1 == 180 && mapLDR == 1){
56            buka();
57            Serial.println("terbuka");
58        }
59        break;
60 }
61 //end Hujan
62
63 delay(1000);
64 }
65 //endloop

```

```
1 //Fungsi untuk buka servo
2 void buka(){
3     for(int i=0;i<90;i++){
4         pos1--;
5         servo1.write(pos1);
6         pos2++;
7         servo2.write(pos2);
8         delay(15);
9     }
10 }
11
12 //Fungsi untuk tutup servo
13 void tutup(){
14     for(int i=0;i<90;i++){
15         pos1++;
16         servo1.write(pos1);
17         pos2--;
18         servo2.write(pos2);
19         delay(15);
20     }
21 }
```

3.2 Foto dan Hasil Implementasi







Referensi

- [1] B. Agus, & AVR Rahasia Kemudahan Bahasa C Dalam Mikrokontroler ATmega 8535, Yogyakarta: Graha Ilmu, 2008.
- [2] O. P. Adha, A. Muid and Y. Brianorman, "Prototipe Sistem Buka Tutup Atap Jemuran Pakaian Menggunakan Mikrokontroler ATmega8," *Jurnal Komputer dan Aplikasi*, vol. 3, no. 1, 2015.
- [3] N. Damastuti, "Sistem Otomasi Atap Bangunan pada Gudang Pengeringan Jagung Berbasis Arduino Uno," *e-narodroid*, vol. 2, no. 1, pp. 111-116, 2016.
- [4] I. G. S. Widharma, "Sensor Suhu Dalam Telemetry Berbasis IoT Sistem Kendali Analog," pp. 1-23, 2020.