

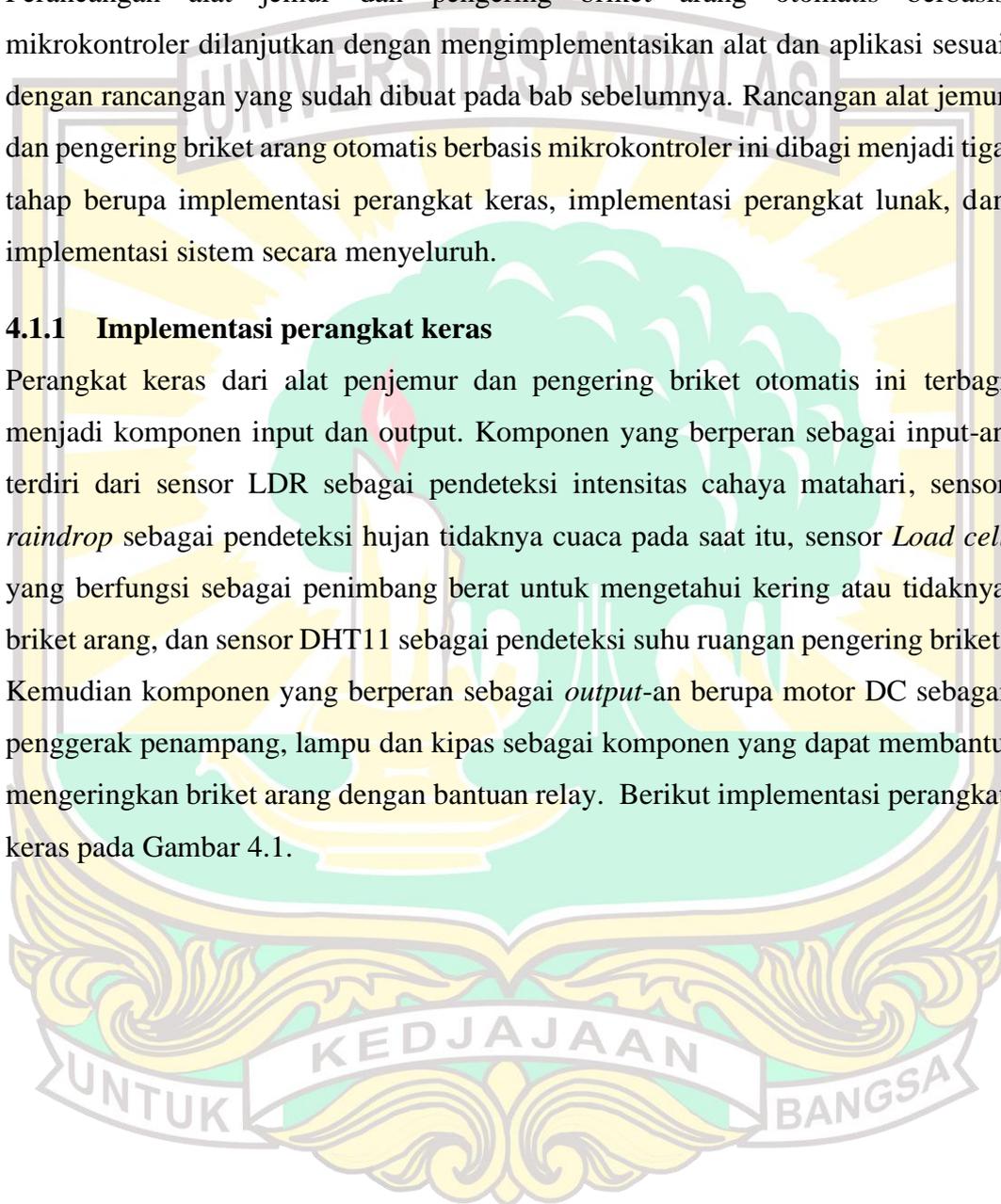
BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Perancangan alat jemur dan pengering briket arang otomatis berbasis mikrokontroler dilanjutkan dengan mengimplementasikan alat dan aplikasi sesuai dengan rancangan yang sudah dibuat pada bab sebelumnya. Rancangan alat jemur dan pengering briket arang otomatis berbasis mikrokontroler ini dibagi menjadi tiga tahap berupa implementasi perangkat keras, implementasi perangkat lunak, dan implementasi sistem secara menyeluruh.

4.1.1 Implementasi perangkat keras

Perangkat keras dari alat penjemur dan pengering briket otomatis ini terbagi menjadi komponen input dan output. Komponen yang berperan sebagai input-an terdiri dari sensor LDR sebagai pendeteksi intensitas cahaya matahari, sensor *raindrop* sebagai pendeteksi hujan tidaknya cuaca pada saat itu, sensor *Load cell* yang berfungsi sebagai penimbang berat untuk mengetahui kering atau tidaknya briket arang, dan sensor DHT11 sebagai pendeteksi suhu ruangan pengering briket. Kemudian komponen yang berperan sebagai *output*-an berupa motor DC sebagai penggerak penampang, lampu dan kipas sebagai komponen yang dapat membantu mengeringkan briket arang dengan bantuan relay. Berikut implementasi perangkat keras pada Gambar 4.1.





Gambar 4.1 Implementasi Perangkat Keras

Keterangan gambar 4.1 :

a. NodeMCU ESP32

NodeMCU ESP32 berperan sebagai pembaca *input*-an dari sensor. Kemudian hasil *input*-an dari sensor akan diproses sehingga dapat menghasilkan *output*-an sesuai dengan yang diinginkan.

b. Modul I298n

Modul I298n digunakan sebagai pengontrolan kecepatan dan arah dari komponen motor DC sebagai penggerak wadah briket

c. Relay

Relay disambungkan dengan kipas dan lampu yang berfungsi untuk mematikan dan menghidupkan kedua komponen sesuai dengan hasil dari input-an sensor DHT11 dan user melalui aplikasi telegram.

d. *Power supply*

Power supply digunakan untuk menyuplai tegangan pada sistem

e. Sensor LDR

Sensor cahaya (LDR) digunakan untuk menangkap intensitas cahaya matahari yang akan dikirimkan ke mikrokontroler sebagai indikasi bahwa cuaca sedang cerah.

f. Sensor *Raindrop*

Sensor air (*raindrop*) digunakan untuk mendeteksi adanya hujan.

g. Sensor DHT11

Sensor DHT11 digunakan untuk memeriksa suhu ruangan briket arang.

h. Kipas

Kipas berfungsi untuk membantu pengeringan di dalam wadah pengering briket arang

i. Lampu

Lampu berfungsi membantu menghangatkan suhu ruangan pengering briket sehingga estimasi waktu pengeringan lebih cepat

j. Penampang/wadah briket

Penampang/wadah briket Berfungsi sebagai tempat peletak briket arang yang akan dikeringkan

k. Motor DC

Motor DC digunakan untuk penggerak wadah penjemur briket keluar maupun masuk ruangan. Kipas dan lampu digunakan untuk mengeringkan briket arang yang belum sepenuhnya kering di dalam ruangan.

l. Load cell

Load cell digunakan sebagai pendeteksi berat briket yang akan dikeringkan.

4.1.2 Implementasi perangkat lunak

Implementasi pada perangkat lunak dibagi menjadi 2 bagian sebagai berikut:

a. Arduino IDE

Pada aplikasi ini penulis menuliskan coding atau perintah pada mikrokontroler NodeMCU ESP32. Berikut ini hasil dari implementasi coding untuk menjalankan sensor LDR di Arduino IDE pada Gambar 4.2.

```
if (otomatis == 1) {  
  if (ldr < 4000 and hujan > 4000 and kondisi == 0) {  
    digitalWrite(fan, LOW);  
    digitalWrite(relay, LOW);  
    myBot.sendMessage(ids, "Jemuran sudah dikeluarkan \r\nCuaca cerah tidak hujan");  
    Serial.println("kondisi terang, tidak hujan");  
    analogWrite(m1, 0);  
  } else if (ldr < 4000 and hujan < 4000 and kondisi == 1) {  
    digitalWrite(fan, HIGH);  
    digitalWrite(relay, HIGH);  
    myBot.sendMessage(ids, "Cuaca sedang hujan \r\nJemuran dimasukan, lampu dan kipas dihidupkan");  
    Serial.println("kondisi terang, dan hujan");  
    kondisi = 0;  
  }  
}
```

Gambar 4.2 Source Code Sensor LDR

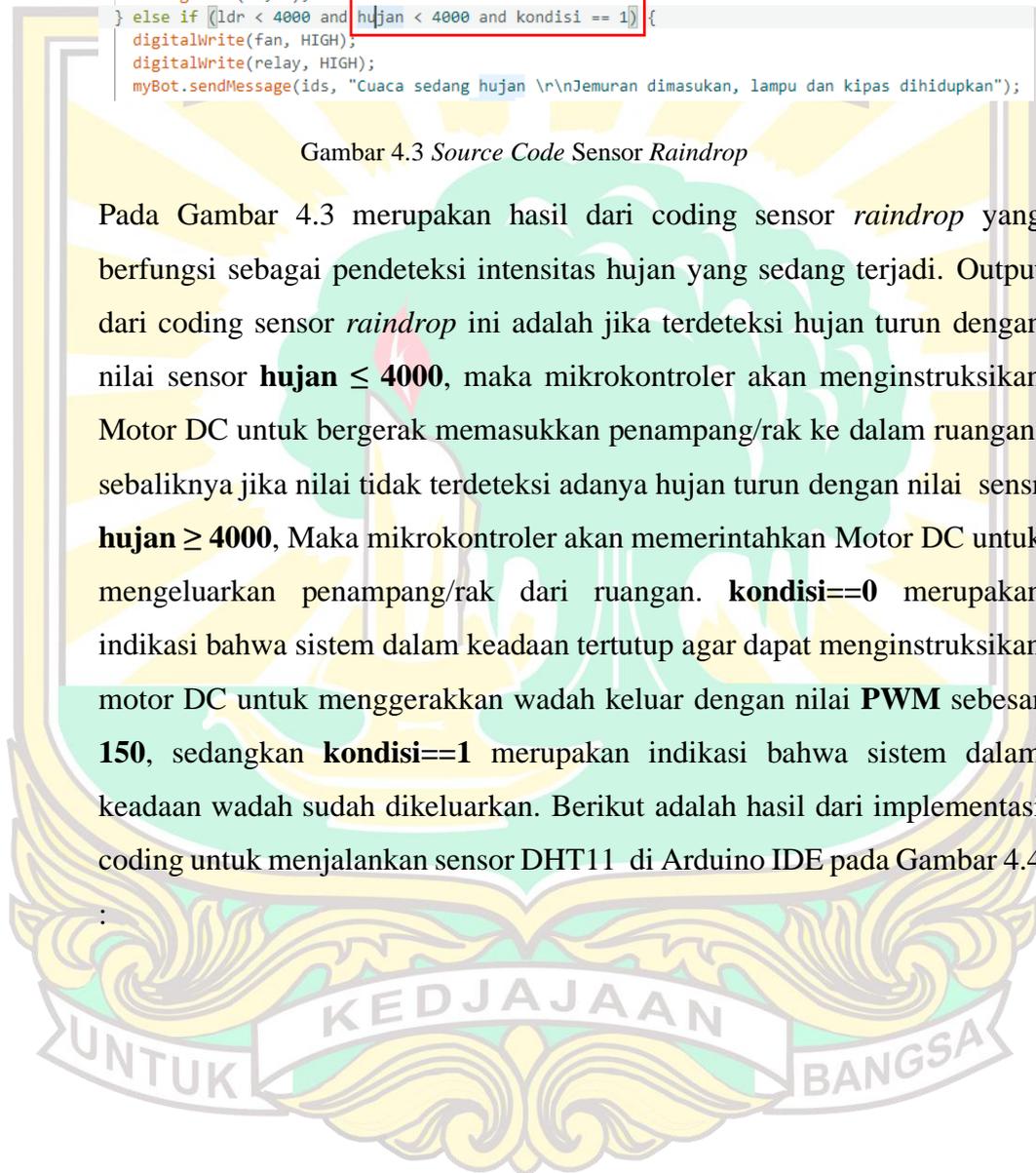
Pada Gambar 4.2 merupakan hasil dari coding sensor LDR yang berfungsi sebagai pendeteksi intensitas cahaya matahari atau cuaca yang sedang terjadi. Output dari coding sensor LDR ini adalah apabila dideteksi intensitas cahaya dalam skala nilai intensitas cahaya bernilai **LDR \geq 4000** yang berarti kondisi sedang gelap, Maka mikrokontroler akan menginstruksikan Motor DC untuk memasukkan penampang/rak ke dalam ruangan, dan sebaliknya jika nilai intensitas cahaya bernilai **LDR \leq 4000** yang berarti cuaca sedang terang, Maka mikrokontroler akan memerintahkan motor DC untuk mengeluarkan penampang/rak dari ruangan. **kondisi==0** merupakan indikasi bahwa sistem dalam keadaan tertutup agar dapat menginstruksikan motor dc untuk menggerakkan wadah keluar, sedangkan **kondisi==1** merupakan indikasi bahwa sistem dalam keadaan wadah sudah dikeluarkan. Berikut adalah hasil dari implementasi

coding untuk menjalankan sensor *raindrop* di Arduino IDE pada Gambar 4.3.

```
if (ldr < 4000 and hujan > 4000 and kondisi == 0) {  
  digitalWrite(fan, LOW);  
  digitalWrite(relay, LOW);  
  myBot.sendMessage(ids, "Jemuran sudah dikeluarkan \r\nCuaca cerah tidak hujan");  
  Serial.println("kondisi terang, tidak hujan");  
  kondisi = 1;  
  analogWrite(m1, 150);  
  delay(DELAY);  
  analogWrite(m1, 0);  
} else if (ldr < 4000 and hujan < 4000 and kondisi == 1) {  
  digitalWrite(fan, HIGH);  
  digitalWrite(relay, HIGH);  
  myBot.sendMessage(ids, "Cuaca sedang hujan \r\nJemuran dimasukan, lampu dan kipas dihidupkan");  
}
```

Gambar 4.3 *Source Code* Sensor *Raindrop*

Pada Gambar 4.3 merupakan hasil dari coding sensor *raindrop* yang berfungsi sebagai pendeteksi intensitas hujan yang sedang terjadi. Output dari coding sensor *raindrop* ini adalah jika terdeteksi hujan turun dengan nilai sensor **hujan** ≤ 4000 , maka mikrokontroler akan menginstruksikan Motor DC untuk bergerak memasukkan penampang/rak ke dalam ruangan, sebaliknya jika nilai tidak terdeteksi adanya hujan turun dengan nilai sensr **hujan** ≥ 4000 , Maka mikrokontroler akan memerintahkan Motor DC untuk mengeluarkan penampang/rak dari ruangan. **kondisi==0** merupakan indikasi bahwa sistem dalam keadaan tertutup agar dapat menginstruksikan motor DC untuk menggerakkan wadah keluar dengan nilai **PWM** sebesar **150**, sedangkan **kondisi==1** merupakan indikasi bahwa sistem dalam keadaan wadah sudah dikeluarkan. Berikut adalah hasil dari implementasi coding untuk menjalankan sensor DHT11 di Arduino IDE pada Gambar 4.4 :



```

h = dht.readHumidity();
t = dht.readTemperature();

Serial.println(msg.text);
if (msg.text.equalsIgnoreCase("/t")) {
String data = "Suhu saat ini adalah " + String(t) + "°C";
myBot.sendMessage(ids, data);
Serial.println("data terkirim");
} else if (msg.text.equalsIgnoreCase("/h")) {
String data = "Kelembaban udara saat ini adalah " + String(h) + "%";
myBot.sendMessage(ids, data);

```

Gambar 4.4 Source Code Sensor DHT11

Pada Gambar 4.4 merupakan hasil dari coding sensor DHT11 yang berfungsi sebagai pendeteksi suhu dan kelembaban ruang pengering briket arang dengan “t=dht.readTemperature” adalah inialisasi suhu dan “h=dht.readHumidity” adalah inialisasi kelembaban. Output dari coding sensor DHT11 ini adalah jika terdeteksi suhu ruangan briket arang berada pada nilai 37⁰, maka mikrokontroler akan mengirim notifikasi ke aplikasi telegram besar suhu yang dideteksi, dan sistem akan mematikan kipas dan lampu secara otomatis. kemudian pada sistem terdapat penggerak wadah briket untuk mengeluarkan dan memasukkan briket dengan komponen motor DC . Berikut adalah hasil dari implementasi coding untuk menjalankan motor DC pada Arduino IDE pada Gambar 4.5.

```

Serial.println("kondisi terang, tidak hujan");
kondisi = 1;
analogWrite(m1, 150);
delay(DELAY);
analogWrite(m1, 0);

analogWrite(m2, 150);
delay(DELAY);
analogWrite(m2, 0);
else if (ldr > 4000 and kondisi == 1) {
digitalWrite(fan, HIGH);
digitalWrite(relay, HIGH);
myBot.sendMessage(ids, "Kondisi gelap \r\nJemuran dimasukan");
Serial.println("terdeteksi gelap");
kondisi = 0;

analogWrite(m2, 150);
delay(DELAY);
analogWrite(m2, 0);

```

Gambar 4.5 Source Code Motor DC

Pada Gambar 4.5 merupakan hasil dari coding Motor DC yang berfungsi untuk memasukkan dan mengeluarkan wadah briket arang berdasarkan input dari sensor LDR dan *raindrop*. “m1” sebagai penggerak searah jarum jam yang berarti wadah briket keluar dari ruangan pengering dan m2 sebagai penggerak berlawanan jarum jam yang berarti wadah briket masuk kedalam ruangan pengering. Pada **m1** dan **m2** diberi nilai *speed* **PWM** sebesar **150** sebagai nilai dari kecepatan perputaran Motor DC dengan **delay** selama **6** detik, maka sistem dapat secara otomatis mengeluarkan dan memasukkan wadah briket arang. Berikut adalah hasil dari implementasi coding untuk menjalankan sensor *Load cell* pada Gambar 4.6.

```
if (berat2 <= berat1-(berat1*0.35) and kondisi==0){  
  Serial.println("sistem dimatikan selesai");  
  myBot.sendMessage(ids, "briket kering \r\nsistem dimatikan");  
  String data = "berat sekarang adalah " + String(berat2);  
  myBot.sendMessage(ids, data);  
  otomatis = 0;  
}
```

Gambar 4.6 Source code Load cell

Pada Gambar 4.6 merupakan hasil dari coding sensor *Load cell* yang berfungsi sebagai pendeteksi berat briket arang yang akan dikeringkan. Dengan menggunakan pengkondisian ketika berat awal dari briket arang sudah berkurang sebanyak **35%**, maka sistem akan merespon dengan mematikan kipas dan lampu secara otomatis dan sistem akan mengirim informasi bahwa sistem sudah mematikan kipas dan lampu serta mengirimkan informasi berat dari briket arang.

b. Aplikasi Telegram

Aplikasi telegram digunakan sebagai pemberi notifikasi pesan dari alat penjemur dan pengering briket secara otomatis dan *real time*.

```

#include "CTBot.h"
CTBot myBot;

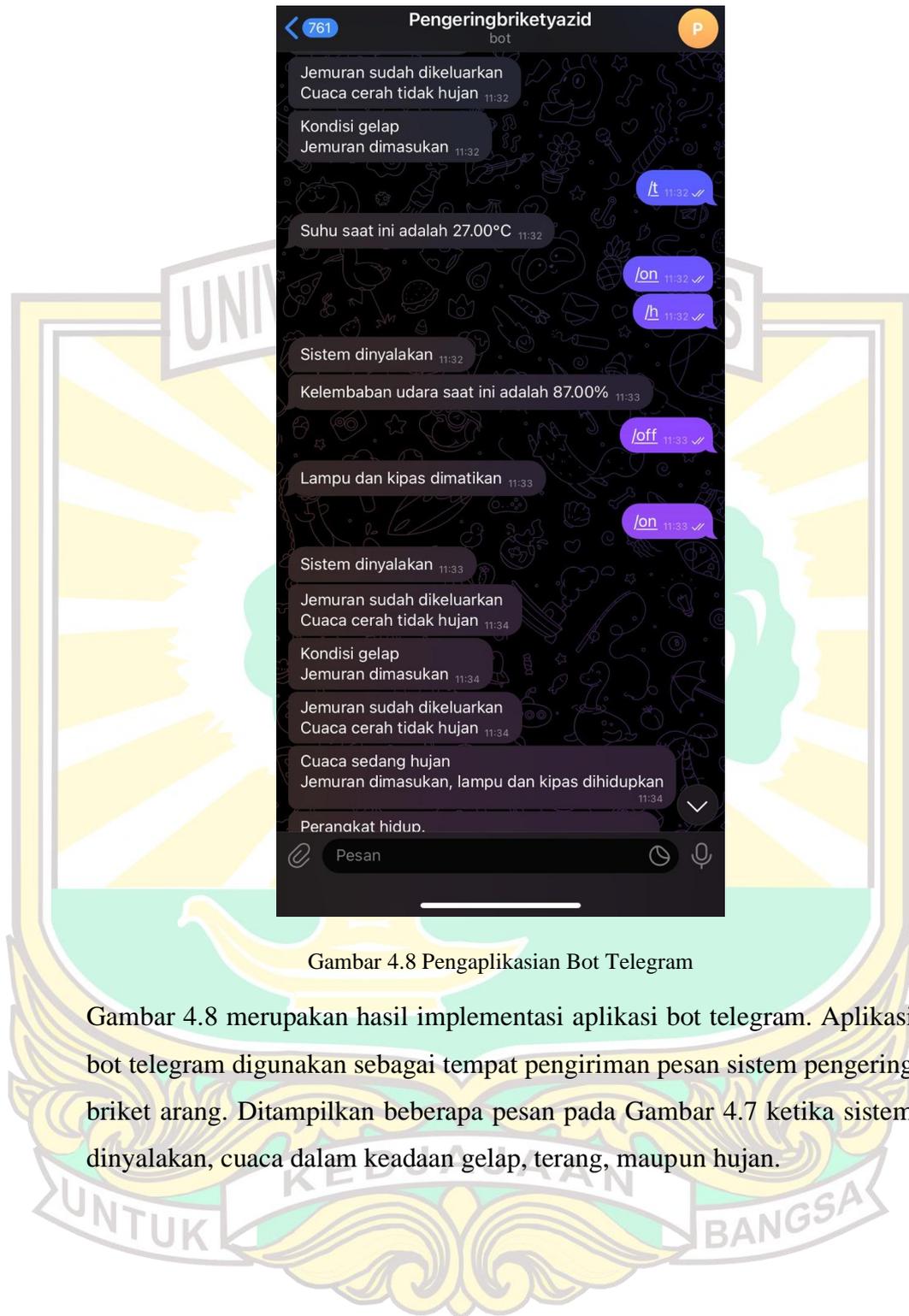
String ssid = "Viona"; // REPLACE
String pass = "yazidhanafi02"; // REPLACE
String token = "6223751423:AAFtNQu1TgXlv5a77y2U99LCbZUodYZb6Ew"; // REPLACE
int64_t ids = 1351490103;

if (CTBotMessageText == myBot.getNewMessage(msg)) {
  Serial.println(msg.text);
  if (msg.text.equalsIgnoreCase("/t")) {
    String data = "Suhu saat ini adalah " + String(t) + "°C";
    myBot.sendMessage(ids, data);
    Serial.println("data terkirim");
  } else if (msg.text.equalsIgnoreCase("/h")) {
    String data = "Kelembaban udara saat ini adalah " + String(h) + "%";
  } else if (msg.text.equalsIgnoreCase("/off") and kondisi==0) {
    digitalWrite(fan, LOW);
    digitalWrite(relay, LOW);
    otomatis = 0;
    myBot.sendMessage(ids, "Lampu dan kipas dimatikan");
  }
}

```

Gambar 4.7 Source Code Aplikasi Bot Telegram

Pada Gambar 4.7 merupakan hasil coding dari penggunaan aplikasi telegram sebagai pemberi notifikasi secara *realtime*. Dimulai dengan memasukkan ssid sebagai nama *wifi* dan kata sandi *wifi* yang akan digunakan berupa “Viona” dan “yazidhanafi02”. Kemudian memasukkan nomor token yang didapat dari aplikasi bot telegram yang dibuat sehingga antara aplikasi telegram dan perangkat dapat terhubung berupa “6223751423: AAFtNQu1TgXlv5a77y2U99LCbZUodYZb6Ew”. Serta memasukkan setiap proses yang terjadi dengan notifikasi pesan seperti pada pendeteksian sensor LDR sehingga dapat secara *real time* mengirimkan pesan keaplikasi telegram. Berikut hasil implementasi aplikasi android yang dapat dilihat pada Gambar 4.8

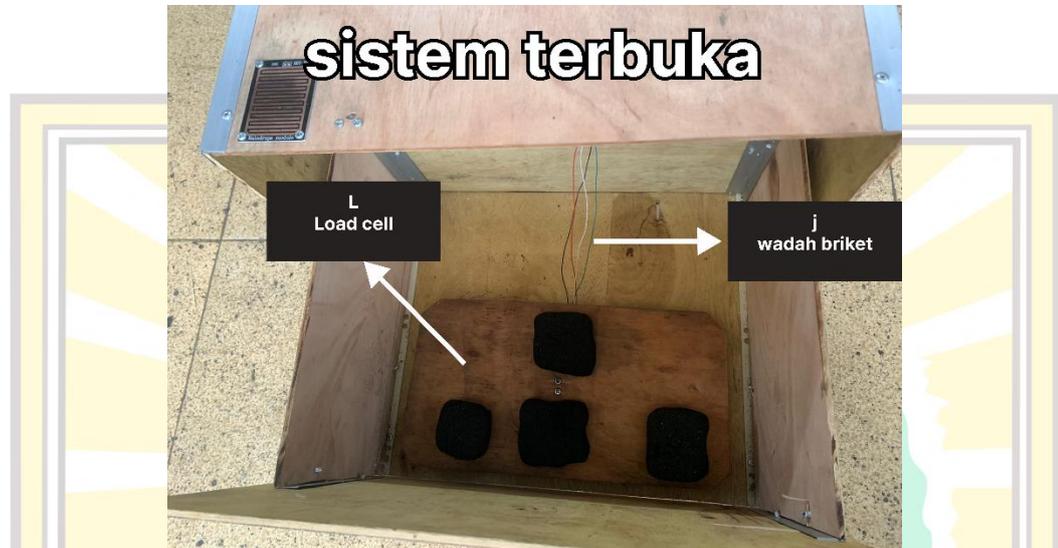


Gambar 4.8 Pengaplikasian Bot Telegram

Gambar 4.8 merupakan hasil implementasi aplikasi bot telegram. Aplikasi bot telegram digunakan sebagai tempat pengiriman pesan sistem pengering briket arang. Ditampilkan beberapa pesan pada Gambar 4.7 ketika sistem dinyalakan, cuaca dalam keadaan gelap, terang, maupun hujan.

4.1.3 Implementasi sistem

Implementasi sistem merupakan gabungan dari implementasi perangkat keras dan perangkat lunak sehingga menjadi sebuah kesatuan sistem yang hasil output dari sistem dapat dijalankan dan dapat mengirim informasi melalui aplikasi telegram.



Gambar 4.9 Implementasi Sistem

Implementasi sistem dapat dilihat pada Gambar 4.9. Sistem penjemur dan pengering briket otomatis ini dimulai dengan meletakkan objek briket arang kedalam penampang/rak, kemudian sistem akan mendeteksi cuaca pada saat itu dengan menggunakan sensor LDR dan *raindrop*, apabila cuaca sedang hujan, maka penampang akan secara otomatis bergerak masuk kedalam ruangan dengan menggunakan motor DC, kemudian sistem akan langsung menghidupkan kipas dan lampu secara otomatis ketika penampang/rak dimasukkan, serta sistem akan mendeteksi suhu ruangan dan menampilkan nilai suhu ruangan ke pesan telegram. *User* dapat mematikan lampu dan kipas melalui aplikasi telegram berdasarkan informasi suhu ruangan yang dapat dijadikan indikasi bahwa briket sudah kering.

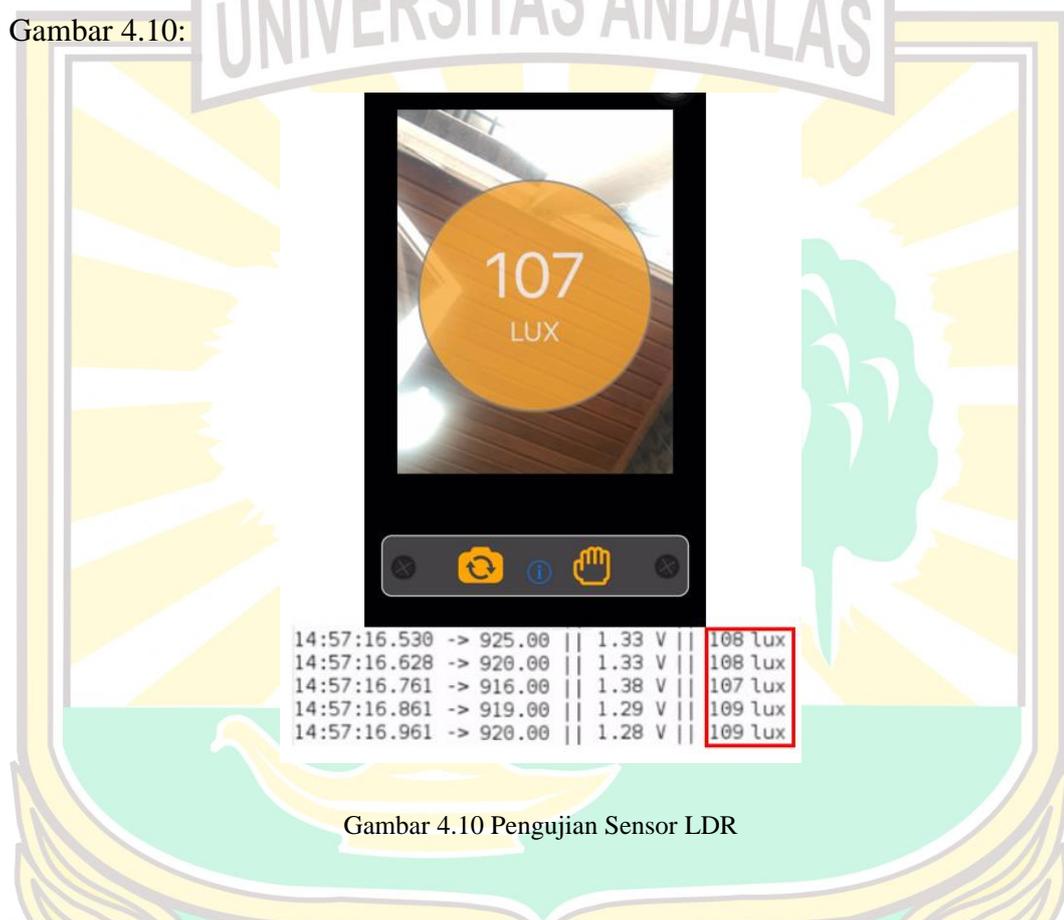
4.2 Pengujian dan Analisa

Pengujian dan analisa akan dilakukan dan dibagi menjadi tiga tahap, berupa pengujian dan analisa perangkat keras, pengujian dan analisa perangkat lunak, dan pengujian dan analisa system secara keseluruhan.

4.2.1 Pengujian dan Analisa perangkat keras

4.2.1.1 Pengujian dan analisa sensor LDR

Pengujian dan Analisa dari komponen sensor LDR akan dilakukan dengan bantuan sensor cahaya *smartphone*. Untuk mengetahui hasil pengujian, dilakukan pengujian tingkat keakuratan dari komponen sensor LDR dengan membandingkan hasil dari sensor cahaya *smartphone*. Berikut adalah hasil dari pengujian sensor LDR pada Gambar 4.10:



Gambar 4.10 Pengujian Sensor LDR

Dilakukan pengujian dan Analisa terhadap perbandingan antara komponen sensor LDR dan sensor cahaya *smartphone* dengan spesifikasi 12 MP dan $f/1.8$ aperture. $f/1.8$ aperture merupakan parameter pada sebuah *smartphone* yang menentukan seberapa banyak cahaya yang dapat ditangkap oleh sensor kamera. pengujian dilakukan sebanyak 6 kali percobaan agar didapatkan hasil pengujian yang akurat. Hasil dari pengujian komponen sensor DHT11 dapat dilihat pada Tabel 4.1 :

Table 4.1 Pengujian dan Analisa sensor LDR

| No | Sensor LDR (Lux) | Sensor Smartphone (Lux) | selisih | Persentase eror |
|-------------------|------------------|-------------------------|---------|-----------------|
| 1 | 455 | 478 | 23 | 0,48% |
| 2 | 406 | 415 | 9 | 0,21% |
| 3 | 303 | 308 | 5 | 0,10% |
| 4 | 211 | 220 | 9 | 0,70% |
| 5 | 109 | 107 | 2 | 0,018% |
| 6 | 420 | 427 | 7 | 0,16% |
| Jumlah | | | 63 | 1,668% |
| Rata-rata Selisih | | | 9,01 | |
| Rata-rata Error | | | 0,40% | |

Dilihat pada Table 4.1 dibandingkan antara komponen sensor LDR dan sensor cahaya smartphone ditempat dan lokasi yang sama. Maka dari itu untuk mendapatkan hasil rata-rata selisih dan eror, dapat menggunakan persamaan sebagai berikut:

$$\text{Rata-rata Selisih} = \frac{\sum \text{Selisih Pengukuran}}{\text{Banyak data}} \dots\dots\dots(4.1)$$

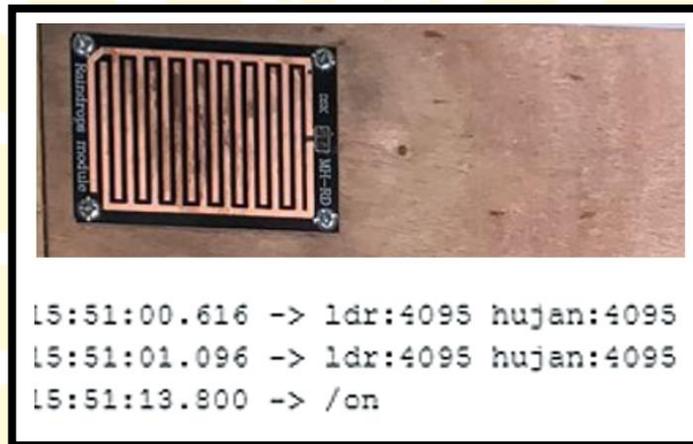
$$\% \text{Error} = \frac{|\text{Nilai Sensor} - \text{Nilai Alat Ukur}|}{\text{Nilai Alat Ukur}} \times 100\% \dots\dots\dots(4.2)$$

$$\text{Rata-rata Error} = \frac{\sum \% \text{Error}}{\text{Banyak Error}} \dots\dots\dots(4.3)$$

dengan dilakukannya pengujian dan Analisa maka didapatkan bahwa rata-rata selisih sebesar 9,01 dan rata-rata eror sebesar 0,40%.

4.2.1.2 Pengujian dan Analisa sensor *Raindrop*

Pengujian dan Analisa dari komponen sensor *raindrop* akan dilakukan dengan meneteskan air sebagai indikator pada komponen sensor. Untuk mengetahui hasil pengujian, dilakukan pengujian tingkat keakuratan dari komponen sensor *raindrop* dengan meneteskan air pada sensor dan dilihat hasil *input* pada serial monitor dengan ketentuan apabila sensor membaca nilai kurang dari 4000 maka dinyatakan hujan dan sebaliknya. Berikut adalah hasil dari pengujian sensor *raindrop* pada Gambar 4.10:



Gambar 4.11 Pengujian dan Analisa sensor *Raindrop*

Dilakukan pengujian dan Analisa terhadap komponen sensor *raindrop* dengan meneteskan air dengan intensitas yang berbeda sebanyak 20 kali percobaan agar didapatkan hasil pengujian yang akurat. Hasil dari pengujian komponen sensor *raindrop* dapat dilihat pada Tabel 4.2.

Table 4.2 Pengujian dan Analisa Sensor *Raindrop*

| No | Kondisi nyata (ditetaskan air) | Kondisi yang dibaca sensor | Keterangan |
|----|-----------------------------------|-------------------------------|------------|
| 1 | Hujan | Hujan | Benar |
| 2 | Tidak hujan | Tidak hujan | Benar |

| | | | |
|----|-------------|-------------|-------|
| 3 | Hujan | Hujan | Benar |
| 4 | Tidak hujan | Tidak hujan | Benar |
| 5 | Hujan | Hujan | Benar |
| 6 | Tidak hujan | Tidak hujan | Benar |
| 7 | Hujan | Hujan | Benar |
| 8 | Tidak hujan | Tidak hujan | Benar |
| 9 | Hujan | Hujan | Benar |
| 10 | Tidak hujan | Tidak hujan | Benar |
| 11 | Hujan | Hujan | Benar |
| 12 | Tidak hujan | Tidak hujan | Benar |
| 13 | Hujan | Hujan | Benar |
| 14 | Tidak hujan | Tidak hujan | Benar |
| 15 | Hujan | Hujan | Benar |
| 16 | Tidak hujan | Tidak hujan | Benar |
| 17 | Hujan | Hujan | Benar |
| 18 | Tidak hujan | Tidak hujan | Benar |
| 19 | Hujan | Hujan | Benar |
| 20 | Tidak hujan | Tidak hujan | Benar |

Berdasarkan Tabel 4.2 didapatkan bahwa kondisi sensor *raindrop* Ketika ditetaskan air akan bernilai dibawah 4000 pada serial monitor, dan tanpa adanya tetesan air, sensor akan bernilai dibawah 4000 pada serial monitor.

4.2.1.3 Pengujian dan Analisa sensor DHT11

Pengujian dan Analisa dari komponen sensor DHT11 akan dilakukan dengan bantuan komponen hygrometer. Untuk mengetahui hasil pengujian, dilakukan pengujian tingkat keakuratan dari komponen sensor DHT11 dengan membandingkan hasil dari komponen hygrometer dan melihat tingkat kelembaban pada ruang pengering briket arang. Berikut adalah hasil dari pengujian sensor DHT11 pada Gambar 4.11:



Gambar 4.12 Pengujian Sensor DHT11

Dilakukan pengujian dan Analisa terhadap perbandingan antara komponen sensor DHT11 dengan komponen *hygrometer* sebanyak 5 kali percobaan agar didapatkan hasil pengujian yang akurat. Hasil dari pengujian komponen sensor DHT11 dapat dilihat pada table 4.3:

Table 4.3 Pengujian dan Analisa Sensor DHT11

| No | Sensor DH11 | hygrometer | selisih | Error % | |
|-------------------|-------------|------------|---------|---------|-------|
| 1 | 81,70% | 81% | 0,70% | 0,864% | |
| 2 | 90,80% | 90% | 0.80% | 0,8% | |
| 3 | 94,30% | 94% | 0,30% | 0,31% | |
| 4 | 97,60% | 97% | 0,60% | 0,61% | |
| Jumlah | | | | 2,4% | 2,58% |
| Rata-rata Selisih | | | 0,6% | | |
| Rata-rata Error | | | 0,658% | | |

Dilihat pada Table 4.3 dibandingkan antara komponen sensor DHT11 dan komponen *hygrometer* ditempat dan lokasi yang sama. Maka dari itu untuk mendapatkan hasil rata-rata selisih dan *error*, dapat menggunakan persamaan sebagai berikut:

$$\text{Rata-rata Selisih} = \frac{\sum \text{Selisih Pengukuran}}{\text{Banyak data}} \dots\dots\dots(4.1)$$

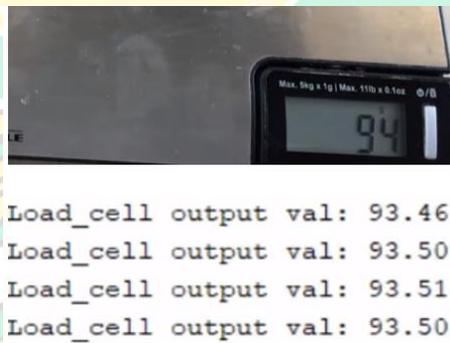
$$\% \text{Error} = \frac{|\text{Nilai Sensor} - \text{Nilai Alat Ukur}|}{\text{Nilai Alat Ukur}} \times 100\% \dots\dots\dots(4.2)$$

$$\text{Rata-rata Error} = \frac{\sum \% \text{Error}}{\text{Banyak Error}} \dots\dots\dots(4.3)$$

dengan dilakukannya pengujian dan Analisa maka didapatkan bahwa rata-rata selisih sebesar 0,6% dan rata-rata *error* sebesar 0,658%.

4.2.1.4 Pengujian dan Analisa Sensor *Load cell*

Pengujian dan analisa sensor *Load cell* akan dilakukan dengan perbandingan nilai berat objek sebenarnya. Hasil pengujian dianalisa berdasarkan tingkat keakuratan komponen sensor *Load cell* ketika menimbang suatu objek dan dibandingkan seberapa banyak presentase selisih dan error pada sensor *Load cell*. Berikut adalah hasil dari pengujian sensor *Load cell* pada Gambar 4.13.



Gambar 4.13 Pengujian dan Analisa *Load cell*

Dilakukan pengujian dan Analisa terhadap perbandingan antara komponen sensor *Load cell* dengan berat sebenarnya beberapa objek sebanyak 5 kali percobaan dengan objek dan berat yang berbeda-beda agar didapatkan hasil pengujian yang

akurat. Hasil dari pengujian komponen sensor *Load cell* dapat dilihat pada Table 4.4.

Table 4.4 Pengujian dan analisa Sensor *Load cell*

| No | Berat sebenarnya | Sensor <i>Load cell</i> | Selisih | Presentase <i>eror</i> |
|----|------------------------|-------------------------|---------|------------------------|
| 1 | 100 gr | 99,23 | 0,77 | 0,077% |
| 2 | 94 gr | 93,5 | 0,50 | 0,050% |
| 3 | 75 gr | 74,3 | 0,70 | 0,090% |
| 4 | 110 gr | 109,6 | 0,40 | 0,036% |
| 5 | 180 gr | 181,83 | 1,83 | 0,01% |
| 6 | 200 gr | 198,4 | 1,60 | 0,08% |
| 7 | 80 gr | 81,5 | 1,50 | 0,018% |
| | Jumlah | | 7,3 | 0,361% |
| | Rata-rata selisih | | 1,042 | |
| | Rata-rata <i>Error</i> | | 0,051% | |

Dilihat pada Table 4.4 dibandingkan antara komponen sensor *Load cell* dan berat asli dari suatu objek. Maka dari itu untuk mendapatkan hasil rata-rata selisih dan *eror*, dapat menggunakan persamaan sebagai berikut:

$$\text{Rata-rata Selisih} = \frac{\sum \text{Selisih Pengukuran}}{\text{Banyak data}} \dots\dots\dots(4.1)$$

$$\% \text{Error} = \frac{|\text{Nilai Sensor} - \text{Nilai Alat Ukur}|}{\text{Nilai Alat Ukur}} \times 100\% \dots\dots\dots(4.2)$$

$$\text{Rata-rata Error} = \frac{\sum \% \text{Error}}{\text{Banyak Error}} \dots\dots\dots(4.3)$$

dengan dilakukannya pengujian dan Analisa maka didapatkan bahwa rata-rata selisih sebesar 1,042 dan rata-rata *error* sebesar 0,051%.

4.2.2 Pengujian dan Analisa Perangkat Lunak

Pengujian dan Analisa perangkat lunak sistem penjemur dan pengering briket arang ini akan dibagi menjadi dua, yaitu melakukan pengujian terhadap Program.uno dan melakukan pengujian terhadap aplikasi telegram. Pengujian ini dilakukan bertujuan untuk mengetahui apakah perangkat lunak yang digunakan berjalan dengan baik dan menganalisa waktu respon dan eksekusi yang dibutuhkan sistem.

4.2.2.1 Pengujian dan Analisa Program.Ino

Dilakukan pengujian dan Analisa program.Ino dengan *profiling* untuk menghitung nilai estimasi waktu respon yang dibutuhkan sistem sehingga dapat berjalan dengannya menggunakan bantuan program tambahan pada Arduino IDE.

Table 4.5 Pengujian dan Analisa Program.Uno

| Jenis Percobaan | Waktu Eksekusi pada program (milidetik) |
|------------------------|---|
| Pembacaan sensor hujan | 4437 milidetik |
| Pembacaan sensor LDR | 4167 milidetik |
| Pembacaan sensor DHT11 | 5189 milidetik |
| Total waktu | 13,793 milidetik |

Kemudian dilakukan perhitungan besar memori yang digunakan untuk dapat mengeksekusi program masing-masing sensor.

Table 4.6 Pengujian dan Analisa *Size Memory*

| Program | File size | Memory Usage |
|-------------------------------------|-----------|--------------|
| Pembacaan sensor <i>raindrop</i> | 1 kb | 235985 bytes |
| Pembacaan sensor LDR | 1 kb | 234578 bytes |
| Pembacaan sensor DHT11 | 2 kb | 263349 bytes |

Pada Tabel 4.6 didapatkan nilai penggunaan file size dan memory usage pada program pembacaan sensor *raindrop*, LDR dan DHT11 yang dibuat serta semua proses dapat berjalan dengan baik dan berhasil. Kemudian dilakukan analisa total memori yang digunakan pada sistem dan dapat dilihat pada Gambar 4.14.

```
Sketch uses 890609 bytes (67%) of program storage space. Maximum is 1310720 bytes.
Global variables use 45052 bytes (13%) of dynamic memory, leaving 282628 bytes for local variables. Maximum is 327680 bytes
esptool.py v4.5.1
Serial port COM4
Connecting...
Chip is ESP32-D0WD-V3 (revision v3.0)
Features: WiFi, BT, Dual Core, 240MHz, VRef calibration in efuse, Coding Scheme None
```

Gambar 4.14 Total Memori yang digunakan

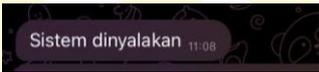
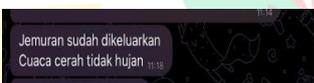
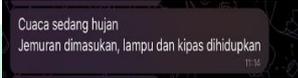
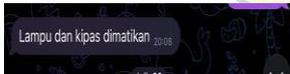
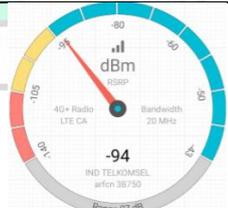
Pada Gambar 4.13 dapat dilihat penggunaan total memori yang digunakan untuk menjalankan sistem pengering briket pada mikrokontroler ESP32. Proses *compile* dapat berjalan dengan baik dan sukses karena program yang dipakai masih dalam batas wajar atau masih belum mencapai batas maksimum memori yang diperlukan dengan nilai pemakaian memori sebesar 890609 bytes dari maksimal penggunaan memori adalah 1310720 bytes.

4.2.2.2 Pengujian dan Analisis Aplikasi Notifikasi Telegram

Dilakukan beberapa pengujian terhadap aplikasi notifikasi telegram dengan koneksi internet. Penulis mengamati dan menganalisa hasil yang diterima melalui pesan yang masuk ke *room chat* aplikasi telegram serta waktu respon bot telegram

dilakukan dengan tujuan untuk menghitung lamanya aplikasi bot telegram dalam merespon setiap notifikasi yang dikirim oleh sistem.

Table 4.7 Pengujian waktu respon bot telegram

| No | Fungsi | Banyak pengujian | Rata-rata waktu respon | Kekuatan sinyal |
|------------------------------------|--|------------------|------------------------|---|
| 1 | Sistem dinyalakan  | 5 | 3,3 detik |  |
| 2 | Wadah briket dikeluarkan  | 5 | 2,1 detik |  |
| 3 | Wadah briket dimasukkan (kondisi hujan)  | 5 | 2,5 detik |  |
| 4 | Wadah briket dimasukkan (kondisi gelap/malam)  | 5 | 2,2 detik |  |
| 5 | Kipas dan lampu dimatikan  | 5 | 3,2 detik |  |
| Rata-rata waktu respon keseluruhan | | | | 2,75 detik |

Dapat ditarik kesimpulan pada Tabel 4.7 bahwa waktu respon rata-rata pada pengujian yang dilakukan sebanyak 5 kali pada aplikasi bot telegram adalah sebesar 2,75 detik.

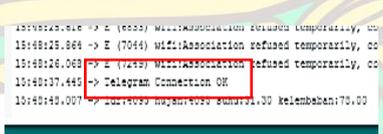
4.2.3 Pengujian dan Analisa Sistem Keseluruhan

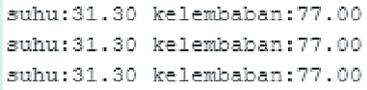
Pengujian dan Analisa sistem secara keseluruhan bertujuan untuk mengetahui apakah sistem sudah berjalan dan berfungsi sesuai dengan tujuan yang dibuat. Kemudian sistem dapat mengirimkan pesan notifikasi ke aplikasi telegram mengenai informasi alat yang dirancang, seperti pesan keadaan cuaca dan keadaan briket arang apakah basah atau kering.

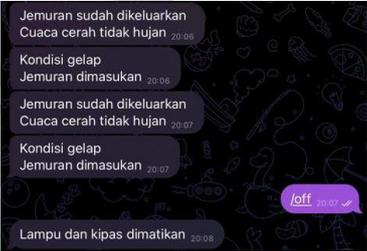
4.2.3.1 Pengujian fungsional

Pengujian fungsional merupakan pengujian yang dilakukan secara keseluruhan terhadap sistem yang dirancang untuk melihat apakah sistem yang dirancang sudah berjalan dengan baik atau sebaliknya berdasarkan rancangan umum. Berikut merupakan pengujian yang dilakukan dapat dilihat pada tabel 4.8:

Table 4.8 Pengujian Fungsional

| no | Butir uji | Indikator keberhasilan | Hasil | Kete rangan |
|----|---|---|---|-------------|
| 1 | Sistem terhubung ke jaringan | Sistem dapat terhubung ke aplikasi telegram |  <p>sistem berhasil terkoneksi dengan aplikasi telegram</p> | Berhasil |
| 2 | Sistem mendeteksi keadaan cuaca dengan sensor LDR dan <i>raindrop</i> | Sistem dapat mendeteksi keadaan cuaca apakah cerah, gelap dan hujan | <pre>58.069 -> ldr:4095 hujan:4095 58.588 -> ldr:4095 hujan:4095 59.064 -> ldr:4095 hujan:4095 59.551 -> ldr:4095 hujan:4095</pre> | Berhasil |

| | | | | |
|---|--|--|--|----------|
| | | dengan memanfaatkan sensor LDR dan <i>raindrop</i> | Sistem berhasil mendeteksi keadaan cuaca | |
| 3 | Sistem penjemur memasukkan dan mengeluarkan briket arang secara otomatis | Sistem dapat memasukkan dan mengeluarkan wadah penjemur briket sesuai dengan kondisi cuaca yang ada | Sistem berhasil memasukkan dan mengeluarkan wadah briket sesuai dengan kondisi cuaca | Berhasil |
| 4 | Sistem mendeteksi suhu ruangan dan berat briket arang | Sistem dapat mendeteksi suhu ruangan dan berat briket arang tinggi atau rendah dengan menggunakan sensor DHT11 dan sensor <i>Load cell</i> . Sehingga dengan informasi suhu ruangan dan berat briket arang, maka sistem dan user dapat mematikan pengering secara otomatis melalui aplikasi telegram |  <p>Sistem berhasil mendeteksi suhu ruangan briket arang</p> | Berhasil |
| 5 | Sistem mematikan kipas dan lampu secara otomatis | Sistem dapat mematikan lampu dan kipas berdasarkan informasi suhu ruangan pengering |  <p>sistem berhasil mematikan</p> | Berhasil |

| | | | | |
|---|--------------------------------------|---|---|----------|
| | melalui aplikasi telegram | dan perintah <i>user</i> melalui aplikasi telegram | lampu dan kipas melalui aplikasi telegram dengan mengetikkan “/off” | |
| 6 | Sistem terhubung keaplikasi telegram | Sistem dapat memberi informasi berupa notifikasi ke aplikasi telegram secara <i>real time</i> |  <p>Sistem berhasil memberi notifikasi pesan melalui telegram secara <i>real time</i></p> | Berhasil |

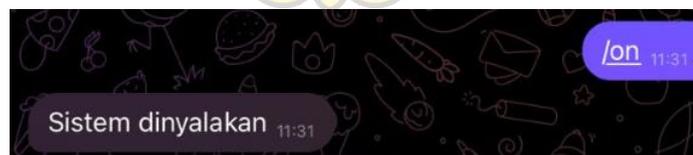
Kemudian dilakukan pengujian secara keseluruhan pada notifikasi bot telegram. Dilakukan pengujian dengan mengamati dan menganalisa kemampuan aplikasi bot telegram dalam menerima pesan yang dikirim oleh mikrokontroler selama sistem berjalan.

Table 4.9 Pengujian dan Analisa Notifikasi bot telegram

| no | Keadaan | Keadaan yang diharapkan | hasil | Indikator |
|----|----------------------------|---|--|-----------|
| 1 | Terdapat koneksi internet | Notifikasi Aplikasi Telegram masuk ke room chat | Notifikasi aplikasi telegram dapat terkirim dan tidak terjadi eror | berhasil |
| 2 | Tidak ada koneksi internet | Notifikasi Aplikasi Telegram tidak masuk ke room chat | Notifikasi aplikasi telegram tidak terkirim ke room chat | Berhasil |

| | | | | |
|---|---|---|---|----------|
| | | dan menunggu koneksi internet | | |
| 3 | sistem penjemur briket dikeluarkan | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “jemuran sudah dikeluarkan, cuaca cerah tidak hujan” | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “jemuran sudah dikeluarkan, cuaca cerah tidak hujan” | berhasil |
| 4 | sistem penjemur briket dimasukkan kedalam ruangan | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “kondisi gelap, jemuran dimasukkan” | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “kondisi gelap, jemuran dimasukkan” | berhasil |
| 5 | sistem mematikan kipas dan lampu | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “kipas dan lampu dimatikan” | Notifikasi Aplikasi Telegram masuk ke room chat dengan pesan “kipas dan lampu dimatikan” | berhasil |

Berikut merupakan tampilan pesan yang dikirim sistem ke aplikasi bot telegram:



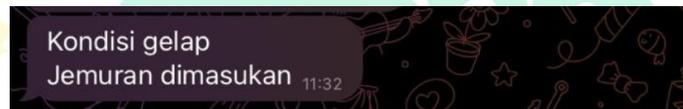
Gambar 4.15 Pesan Telegram Sistem Terkoneksi

Pada Gambar 4.15 *user* dapat mengetik “/on” yang bertujuan untuk menghidupkan sistem penjemur secara keseluruhan melalui room chat telegram.



Gambar 4.16 Pesan Telegram Jemuran dikeluarkan

Kemudian pada Gambar 4.16 diketahui sistem mengirimkan pesan “jemuran sudah dikeluarkan, cuaca cerah tidak hujan” yang berarti bahwa sensor LDR dan *raindrop* memberi input bahwa cuaca sedang cerah dan tidak hujan



Gambar 4.17 Pesan Telegram Sistem Penjemur Briket dimasukkan kedalam Ruangan

Dapat dilihat juga pada Gambar 4.17 sistem mengirimkan pesan “kondisi gelap jemuran dimasukkan” ke room chat telegram yang berarti kondisi pada saat itu sudah menjelang malam hari. Pada Tabel 4.9 merupakan hasil dari pengujian kemampuan aplikasi bot telegram dalam menerima pesan yang dikirim oleh mikrokontroler.

4.2.3.2 Pengujian Otomatisasi Sistem Pengereng Briket Arang

Pada pengujian otomatisasi sistem pengereng briket arang bertujuan untuk membuktikan bahwa komponen sistem berupa sensor LDR, sensor *raindrop*, sensor dht11, sensor *Load cell*, motor DC, lampu, kipas dan notifikasi aplikasi telegram dapat bekerja sesuai dengan tujuan utama yang telah dirancang. Sensor LDR digunakan sebagai parameter untuk mendeteksi intensitas cahaya matahari yang masuk ke sistem, sensor *raindrop* digunakan sebagai pendeteksi cuaca hujan yang turun sehingga dapat mengantisipasi briket terkena air hujan ketika sedang dijemur, sensor DHT11 digunakan sebagai pendeteksi suhu ruangan pengereng briket ketika sedang hujan dan akan memberi informasi ke aplikasi telegram sehingga *user* dapat *me-monitoring* pengereng dan dapat mematikan ketika briket sudah kering,

kemudian motor DC berperan sebagai komponen yang dapat memindahkan wadah briket masuk ke dalam ruang briket ketika cuaca sedang hidup, kipas dan lampu berfungsi untuk membantu mengeringkan briket arang ketika cuaca sedang hujan, serta aplikasi telegram berfungsi sebagai pemberi informasi ketika sistem sedang berjalan. Hasil pengujian otomatisasi sistem pengering briket arang dapat dilihat pada tabel 4.8.

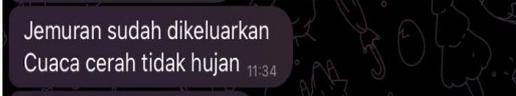
Table 4.10 Pengujian Otomatisasi Sistem Pengering Briket Arang

| Ke- | Waktu dan jam | Kon-disi cahaya | Kon-disi cuaca | Motor DC (wadah briket) | Kon-disi briket arang (gr) | Suhu ruangan | Telegram | Kondisi kipas dan lampu |
|-----|---------------|-----------------|----------------|-------------------------|----------------------------|-------------------|--------------------------|-------------------------|
| 1 | 8.00 | Terang (2688) | Cerah (4095) | Keluar (m1,150) | 100,86 | 29.2 ⁰ | Pesan briket dikeluarkan | Mati |
| 2 | 10.00 | Terang (2895) | Cerah (4095) | Keluar (m1,150) | 97,67 | - | - | Mati |
| 3 | 12.00 | Terang (2187) | Cerah (4095) | Keluar (m1,150) | 94,15 | - | - | Mati |
| 4 | 14.00 | Terang (2032) | Cerah (4095) | Keluar (m1,150) | 90,73 | - | - | Mati |
| 5 | 16.00 | Terang (2198) | Cerah (4095) | Keluar (m1,150) | 87,89 | - | - | Mati |
| 6 | 19.00 | Redup (4054) | Cerah (4095) | Masuk (m2,150) | 85,21 | 31.6 ⁰ | Pesan kondisi gelap | Hidup |
| 7 | 20.00 | Gelap (4156) | Cerah (4095) | Masuk (m2,150) | 83.43 | 31.6 ⁰ | - | Hidup |
| 8 | 8.00 | Terang (2314) | Cerah (4095) | Keluar (m1,150) | 76,52 | - | Pesan briket dikeluarkan | Mati |
| 9 | 10.00 | Terang (2295) | Cerah (4095) | Keluar (m1,150) | 73,04 | - | - | Mati |
| 10 | 12.00 | Gelap (4095) | Hujan (2034) | Masuk (m2,150) | 70,05 | 34.2 ⁰ | Pesan briket dimasukkan | Hidup |
| 11 | 14.00 | Gelap (4067) | Hujan (1045) | Masuk (m2,150) | 68,95 | 34.8 ⁰ | - | Hidup |
| 12 | 16.00 | Gelap (4087) | Hujan (1047) | Masuk (m2,150) | 66,56 | 35.1 ⁰ | - | Hidup |

| | | | | | | | | |
|----|-------|-----------------|-----------------|-------------------|------|-------------------|---|------|
| 13 | 18.00 | Gelap (4103) | Hujan (1287) | Masuk (m2,150) | 65,4 | 36,7 ^o | Sistem mematikan kipas dan lampu | Mati |
|----|-------|-----------------|-----------------|-------------------|------|-------------------|---|------|

Pada Tabel 4.10 dapat diamati pengujian otomatisasi pengeringan yang dilakukan dengan sistem pengering briket dalam waktu pengujian beberapa hari. Hasil dari pengujian yang dilakukan adalah sensor LDR membaca kondisi intensitas cahaya pada saat pengujian dan sensor *raindrop* membaca intensitas curah hujan pada saat pengujian berjalan. Ketika sensor LDR membaca bahwa kondisi cuaca sedang terang dan sensor *raindrop* membaca kondisi cuaca sedang tidak hujan, maka sistem akan menggerakkan motor DC untuk memindahkan wadah briket keluar dari ruangan pengering briket. Terdapat dua kondisi yang dapat membuat sistem menggerakkan wadah briket masuk ke dalam ruangan pengering briket. Yang pertama jika sensor LDR membaca bahwa kondisi cuaca sudah dalam keadaan gelap atau malam hari, maka sistem akan menggerakkan wadah briket masuk ke dalam ruang pengering briket, yang kedua adalah jika sensor *raindrop* membaca adanya tetesan air hujan, maka sistem akan memindahkan wadah briket secara otomatis ke dalam ruangan pengering briket. Kemudian ketika wadah briket sudah masuk ke dalam ruangan pengering, maka secara otomatis kipas dan lampu akan hidup sebagai alternatif pengering briket ketika cuaca sedang tidak memungkinkan untuk melakukan pengeringan menggunakan sinar matahari. Setelah kipas dan lampu sudah dihidupkan, maka sensor DHT11 dan *Load cell* akan memberikan informasi ke telegram mengenai berat dan suhu yang ada di ruangan pengering briket, ketika briket sudah kering berdasarkan berat dan suhu ruangan briket dengan nilai berat dibawah 35% dari berat awal briket, maka sistem dapat mematikan kipas dan lampu secara otomatis. Pada Tabel 4.11 merupakan tampilan pengujian otomatisasi sistem secara keseluruhan.

Table 4.11 Tampilan Pengujian Otomatisasi Sistem

| No | Tampilan Sistem | Hasil |
|----|--|---|
| 1 |  | <p>Cuaca sedang terang dan tidak hujan, maka wadah briket akan otomatis keluar dari ruangan</p> |
| 2 |  | <p>Ketika cuaca gelap atau sudah malam hari, maka wadah briket akan masuk secara otomatis</p> |
| |  | <p>Ketika jemuran briket sudah masuk kedalam ruangan, kemudian kipas dan lampu hidup</p> |
| 3 |  | <p>Cuaca sedang terang dan tidak hujan, maka sistem akan mengirimkan</p> |

| | | |
|---|--|---|
| | | pesan ke telegram |
| 4 | | Ketika malam hari, maka sistem akan mengirimkan pesan ke telegram |

Pada tabel 4.11 dapat dilihat bahwa sistem menggunakan komponen motor DC untuk menggerakkan wadah briket arang, ketika cuaca cerah dan tidak hujan maka motor DC akan menggerakkan wadah briket keluar dari ruangan pengering dan sebaliknya, serta sistem akan secara otomatis memberikan informasi secara *real time* ke aplikasi bot telegram sebagai sarana *monitoring user* kepada sistem yang sedang berjalan.

4.2.3.3 Pengujian perbandingan pengering briket arang dengan alat yang dirancang dengan pengeringan tanpa menggunakan alat.

Pengujian perbandingan antara pengering briket arang menggunakan sistem yang dirancang dengan pengeringan tanpa menggunakan sistem untuk membuktikan apakah dengan menggunakan sistem pengering otomatis ini, proses pengeringan dapat berjalan lebih efisien atau tidak berdasarkan tujuan dari perancangan sistem pengering ini.

Table 4.12 Pengujian dengan Menggunakan Alat

| Hari ke- | Waktu dan jam | Kon-disi cahaya | Kon-disi cuaca | Motor DC (wadah briket) | Kon-disi briket arang dan berat (gr) | Suhu ruangan | Telegram | Kondisi kipas dan lampu |
|----------|---------------|-----------------|----------------|-------------------------|--------------------------------------|-------------------|----------------------------------|-------------------------|
| 1 | 8.00 | Terang (2688) | Cerah (4095) | Keluar (m1,150) | 100,86 | 29.2 ⁰ | Pesan briket dikeluarkan | Mati |
| | 10.00 | Terang (2895) | Cerah (4095) | Keluar (m1,150) | 97,67 | - | - | Mati |
| | 12.00 | Terang (2187) | Cerah (4095) | Keluar (m1,150) | 94,15 | - | - | Mati |
| | 14.00 | Terang (2032) | Cerah (4095) | Keluar (m1,150) | 90,73 | - | - | Mati |
| | 16.00 | Terang (2198) | Cerah (4095) | Keluar (m1,150) | 87,89 | - | - | Mati |
| | 18.00 | Redup (4054) | Cerah (4095) | Masuk (m2,150) | 85,21 | 31.6 ⁰ | Pesan kondisi gelap | Hidup |
| | 20.00 | Gelap (4156) | Cerah (4095) | Masuk (m2,150) | 83,43 | 31.6 ⁰ | - | Hidup |
| 2 | 8.00 | Terang (2314) | Cerah (4095) | Keluar (m1,150) | 76,52 | - | Pesan briket dikeluarkan | Mati |
| | 10.00 | Terang (2295) | Cerah (4095) | Keluar (m1,150) | 73,04 | - | - | Mati |
| | 12.00 | Gelap (4095) | Hujan (2034) | Masuk (m2,150) | 70,05 | 34.2 ⁰ | Pesan briket dimasukkan | Hidup |
| | 14.00 | Gelap (4067) | Hujan (1045) | Masuk (m2,150) | 68,95 | 34.8 ⁰ | - | Hidup |
| | 16.00 | Gelap (4087) | Hujan (1047) | Masuk (m2,150) | 66,56 | 35.1 ⁰ | - | Hidup |
| | 18.00 | Gelap (4103) | Hujan (1287) | Masuk (m2,150) | 65,4 | 36,7 ⁰ | Sistem mematikan kipas dan lampu | Mati |

Dilihat pada tabel 4.12 dilakukan pengujian selama beberapa hari, pengering berjalan sesuai dengan keadaan cuaca pada saat itu, pada percobaan ini briket arang memiliki berat awal sebesar 100,86 gr sebanyak 4 buah briket dalam keadaan basah, pada pengujian dihari pertama, keadaan cuaca cerah dan tidak ada hujan, sehingga proses pengeringan berjalan tanpa menggunakan pengering ruangan briket pada jam 8.00- 18.00. kemudian pada jam 18.00 briket masuk kedalam ruangan pengering dengan indikasi bahwa cuaca sudah gelap dan briket dikeringkan dari dalam ruangan pengering briket, kondisi briket dengan berat 85,21 gr. Pada hari kedua wadah briket kembali keluar dari ruangan briket dengan keadaan briket dengan berat 76,52 gr. Pada pengujian hari kedua ini kondisi cuaca hujan terjadi pada jam 12.00 sehingga briket dikeringkan di dalam ruangan pengering. Pada sekitar jam 18.00 sistem mematikan kipas dan lampu mati secara otomatis karena briket sudah dalam kategori kering dengan berat 65,4 gr dan suhu ruangan mencapai 36,7⁰ atau sudah memenuhi pengkondisian berat briket sudah berkurang 35% dari berat awal penimbangan. Maka proses pengeringan dari mulai briket basah sampai dengan kering memakan waktu 2 hari.

Table 4.13 Pengujian pengeringan tanpa menggunakan alat

| Hari ke- | Waktu (Wib) | Kondisi cuaca | Kondisi briket arang | Berat briket |
|----------|-------------|---------------|----------------------|--------------|
| 1 | 8.00 | Cerah | Basah | 100 gr |
| | 10.00-07.00 | Cerah | Basah | 86 gr |
| 2 | 8.00 | Cerah | Basah | 85 gr |
| | 10.00 | Cerah | Basah | 82 gr |
| | 12.00-20.00 | hujan | Basah | 79 gr |
| 3 | 8.00 | Cerah | Basah | 77 gr |
| | 10.00 | Cerah | Lembab | 74 gr |
| | 12.00 | Cerah | Lembab | 71 gr |

| | | | | |
|--|-------|-------|--------|-------|
| | 13.00 | Cerah | Lembab | 68 gr |
| | 16.00 | Cerah | Kering | 64 gr |

Kemudian pada tabel 4.13 dilakukan juga pengujian pengeringan briket arang tanpa menggunakan sistem dihari yang sama. Pada pengujian hari pertama briket memiliki berat 100 gr dalam keadaan basah. Dalam satu hari, ketika dikeringkan briket mengalami pengurangan berat menjadi sebesar 86 gr. Pada pengujian dihari kedua, berat briket arang berada pada nilai 79 gr pada jam 20.00. Kemudian pada hari ketiga pada jam 16.00 briket sudah dalam keadaan kering dengan berat 64 gr.

