

## BAB IV

### PENGUJIAN DAN PEMBAHASAN

Pengujian alat bertujuan untuk mengetahui kinerja baik pada masing – masing komponen atau kinerja secara keseluruhan. Hasil dari pengujian diharapkan dapat digunakan sebagai sumber data yang valid untuk mengetahui apakah alat sudah bekerja dengan baik.

#### A. Hasil Pengujian

##### 1. Pengujian Tegangan Catu Daya

###### a. Pengujian Tegangan Tanpa Beban

Tabel 23. Hasil Pengujian Tegangan Tanpa Beban

No	Pengukuran Pada	Pengukuran Ke-	V-Out (Volt)	V-Out Terukur (Volt)	Selisih Ukur	Error (%)
1.	<i>Switching Power Supply</i>	1	12	12	0	0
		2	12	11.9	0.1	0.83
		3	12	12	0	0
2.	LM2596 pada SIM800L	1	4.0	3.9	0.1	2.5
		2	4.0	4.0	0	0
		3	4.0	3.9	0.1	2.5

Tabel 23 merupakan tabel hasil pengujian tegangan tanpa beban yang diukur menggunakan multimeter. Presentase *error* pengukuran didapatkan dengan membagi nilai selisih pembacaan dengan nilai terbaca kemudian dikali 100 %.

$$Error (\%) = \frac{(V_{out\ terukur} - V_{out})}{V_{out\ Terukur}} \times 100\%$$

Berdasarkan rumus di atas, diperoleh hasil perhitungan sebagai berikut:

$$Error (\%) = \frac{(12 - 11.9)}{12} \times 100\%$$

$$Error (\%) = \frac{0.1}{12} \times 100\%$$

$$Error (\%) = 0.83 \%$$

b. Pengujian Tegangan Dengan Beban

Tabel 24. Hasil Pengujian Tegangan Dengan Beban

No	Pengukuran Pada	Pengukuran Ke-	V-Out (Volt)	V-Out Terukur (Volt)	Selisih Ukur	Error (%)
1.	<i>Switching Power Supply</i>	1	12	11.8	0.2	1.66
		2	12	11.9	0.1	0.83
		3	12	12	0	0
2.	LM2596 pada SIM800L	1	4.0	3.9	0.1	2.5
		2	4.0	3.8	0.2	5
		3	4.0	4.0	0	0

Tabel 24 merupakan tabel hasil pengujian tegangan dengan beban yang diukur menggunakan multimeter. Pengujian dilakukan dengan beban *input* sensor serta *output* motor servo dan SIM800L. Presentase *error* pengukuran didapatkan dengan membagi nilai selisih pembacaan dengan nilai terbaca kemudian dikali 100 %.

$$Error (\%) = \frac{(V_{out\ terukur} - V_{out})}{V_{out\ Terukur}} \times 100\%$$

Berdasarkan rumus di atas, diperoleh hasil perhitungan sebagai berikut:

$$Error (\%) = \frac{(12 - 11.8)}{12} \times 100\%$$

$$Error (\%) = \frac{0.2}{12} \times 100\%$$

$$Error (\%) = 1.66 \%$$

## 2. Pengujian sensor

### a. Pengujian Sensor Kelembaban Tanah

Tabel 25. Hasil Pengujian Sensor Kelembaban Tanah

No.	Pembacaan Sensor	Nilai Kelembaban (Rh %)	Soilmeter (%)	Selisih	Error (%)
1.	273	73	75	2.0	2.6
2.	444	56	55	1.0	1.8
3.	499	51	49	2.0	4.0
4.	263	74	70	4.0	5.7
5.	272	73	70	3.0	4.2
6.	311	69	70	1.0	1.4
7.	274	73	70	3.0	4.2
8.	276	73	70	3.0	4.2
9.	335	67	65	2.0	3.0
10.	736	28	30	2.0	6.6
Rata – rata					3.77

Tabel 25 menunjukkan perbandingan hasil pengujian sensor kelembaban tanah dengan soilmeter. Setelah diketahui hasil perbandingan pengukuran maka didapatkan *error* selisih pembacaan. Presentase *error* pengukuran didapatkan dengan membagi nilai selisih pembacaan dengan nilai terbaca kemudian dikali 100 %.

$$Error (\%) = \frac{(kelembaban - soilmeter)}{kelembaban} \times 100\%$$

Berdasarkan rumus di atas, diperoleh hasil perhitungan sebagai berikut:

$$Error (\%) = \frac{(75 - 73)}{75} \times 100\%$$

$$Error (\%) = \frac{2}{75} \times 100\%$$

$$Error (\%) = 2.6 \%$$

b. Pengujian Sensor DHT11

Tabel 26. Hasil Pengujian Sensor DHT11

No.	Waktu	DHT11 (°C)	Thermometer (°C)	Selisih	Error (%)
1.	03:12	29	28.8	0.2	0.69
2.	04:30	29	28.7	0.3	1.04
3.	05:19	28	28.3	0.3	1.06
4.	07:10	28	28	0	0
5.	07:28	30	30.3	0.3	0.99
6.	11:22	33	33.3	0.3	0.90
7.	13:25	35	34	1	2.94
8.	20:33	33	32.6	0.4	1.22
9.	21:47	31	30.4	0.6	1.97
10.	22:42	33	30.6	2.4	7.84
Rata – rata					1.865

Tabel 26 menunjukkan perbandingan hasil pengujian sensor suhu dengan termometer. Pengujian dilakukan sebanyak 10 kali dengan waktu yang bervariasi. Setelah diketahui hasil perbandingan pengukuran maka didapatkan *error* selisih pembacaan. Presentase *error* pengukuran didapatkan dengan membagi nilai selisih pembacaan dengan nilai terbaca kemudian dikali 100 %.

$$Error (\%) = \frac{(DHT - termometer)}{DHT} \times 100\%$$

Berdasarkan rumus di atas, diperoleh hasil perhitungan sebagai berikut:

$$Error (\%) = \frac{(29 - 28.8)}{29} \times 100\%$$

$$Error (\%) = \frac{0.2}{29} \times 100\%$$

$$Error (\%) = 0.69 \%$$

### 3. Pengujian RF 433 MHz

Tabel 27. Hasil Pengujian RF 433 MHz

No.	Kondisi	Jarak (m)	Hasil Transmisi	
			Diterima	Ditolak
1.	Tanpa Penghalang	2	√	-
		4	√	-
		6	√	-
		8	√	-
		10	√	-
2.	Terdapat Penghalang	2	√	-
		4	-	√
		6	-	√
		8	-	√
		10	-	√

Pengujian transmisi RF 433 MHz dilakukan sebanyak dua kali percobaan, yaitu tanpa penghalang dan dengan penghalang. Pada pengujian tanpa penghalang, jarak maksimum transmisi yaitu sejauh 10 meter, sedangkan pada pengujian dengan penghalang pada jarak 2 meter transmisi data masih dapat dilakukan, namun pada jarak lebih dari 2 meter tidak terjadi transmisi data. Jarak transmisi dipengaruhi oleh tegangan serta jenis antena yang dipakai. Pada alat ini hanya menggunakan kawat tembaga sebagai antena, sehingga mengakibatkan jarak transmisi terbatas di 10 meter.

#### 4. Pengujian Unjuk Kerja

Tabel 28. Hasil Pengujian Unjuk Kerja

No.	Kel. tanah (%)	Water Level (cm)		Push Button		Motor Servo			Keterangan
		1	2	Up	Dwn	1	2	3	
1.	65	0	0	-	-	-	√	√	TERBUKA
		2	0	-	-	-	-	√	TERBUKA
		2	2	-	-	-	-	√	TERBUKA
		0	2	-	-	-	√	√	TERBUKA
		0	0	√	-	-	√	√	TERBUKA
		0	0	-	√	-	√	-	TERTUTUP
2.	75	2	0	-	-	-	-	√	TERBUKA
		2	2	-	-	-	-	-	TERTUTUP
		3	2	-	-	√	√	-	TERTUTUP
		0	0	-	-	-	√	-	TERTUTUP
		0	0	√	-	-	√	√	TERBUKA
		0	0	-	√	-	√	-	TERTUTUP

Tabel 28 menunjukkan unjuk kerja alat secara keseluruhan. Pengujian unjuk kerja alat dilakukan pada kelembaban tanah 65 % dan 75 % dengan ketinggian air bervariasi. Pada rangkaian dam terdapat dua tombol yaitu buka dan tutup yang digunakan sebagai kontrol manual servo 3. Alat akan mulai bekerja saat pertama diaktifkan, dan akan berhenti bekerja sesuai batas yang ditentukan pada program. Pada pengujian ini alat akan berhenti 15 menit sejak pertama diaktifkan.

Berdasarkan tabel di atas, pada kolom keterangan hanya menampilkan keadaan servo 3, yaitu servo yang berhadapan langsung dengan saluran irigasi.

## **B. Pembahasan**

Berdasarkan pengujian pada setiap bagian secara keseluruhan, dapat diuraikan pembahasan kinerja alat sebagai berikut:

### **1. Catu Daya**

Catu daya digunakan sebagai pemasok tegangan pada setiap komponen pada alat. Sumber tegangan berasal dari listrik AC 220 Volt yang masuk ke *switching power supply* untuk diubah tegangan keluaran menjadi DC 12 Volt. Tegangan DC 12 Volt tersebut digunakan untuk mensuplai Arduino Nano dan Arduino Mega. LM2596 merupakan *stepdown dc-dc* untuk mensuplai SIM800L. Tegangan 5 Volt dari keluaran pin 5 Volt Arduino Mega diubah menjadi DC 4 Volt.

Berdasarkan data yang diperoleh pada pengujian tegangan catu daya, bahwa pada pengujian tanpa beban rata-rata presentase *error* pada *switching power supply* sebesar 0.26 %, dan pada LM2596 rata-rata presentase *error* 1.66 %. Pada pengujian dengan beban diperoleh rata-rata presentase *error* pada *switching power supply* sebesar 0.8 %, dan pada LM2596 rata-rata presentase *error* sebesar 2.5 %.

### **2. Mikrokontroler Arduino**

Pada alat ini menggunakan dua Arduino, yaitu:

#### **a. Arduino Nano**

Arduino Nano digunakan sebagai pusat kontrol sensor kelembaban tanah, *water level 1*, modul *transmitter*, serta servo 1 dan servo 2 pada rangkaian node. Dari hasil pengujian Arduino Nano dapat bekerja dengan baik. Adapun penggunaan port pada Arduino Nano yaitu:

1) *Port A0*

*Port A0* digunakan sebagai *input* yang terhubung dengan sensor kelembaban tanah YL-69.

2) *Port A1 dan A2*

*Port A1 dan A2* digunakan sebagai *input* yang terhubung dengan sensor *water level 1*.

3) *Port D2*

*Port D2* digunakan sebagai *output* yang terhubung dengan servo 1.

4) *Port D3*

*Port D3* digunakan sebagai *output* yang terhubung dengan servo 2.

5) *Port D12*

*Port D12* digunakan sebagai *output* yang terhubung dengan modul *transmitter*.

b. *Arduino Mega2560*

*Arduino Mega* digunakan sebagai pusat kontrol modul *receiver*, servo 3, SIM800L, *push button*, DHT11, *water level 2*, dan LCD. Dari pengujian *Arduino Mega* dapat bekerja dengan baik. Adapun penggunaan port pada *Arduino Mega* sebagai berikut:

1) *Port PF2/ADC2*

*Port PF2/ADC2* digunakan sebagai *input water level 2*.

2) *Port PD0/SCL/INT0*

*Port PD0/SCL/INT0* digunakan sebagai *output* yang terhubung pada pin SCL I2C LCD.



- 3) *Port PD1/SDA/INT1*  
*Port PD1/SDA/INT1* digunakan sebagai *output* yang terhubung pada pin SDA I2C LCD.
- 4) *Port PD3/TXD1/INT3*  
*Port PD3/TXD1/INT3* digunakan sebagai *output* yang terhubung pada pin Rx SIM800L.
- 5) *Port PD2/RXD1/INT2*  
*Port PD2/RXD1/INT2* digunakan sebagai *output* yang terhubung pada pin Tx SIM800L.
- 6) *Port PJ0/RXD3/PCINT9*  
*Port PJ0/RXD3/PCINT9* digunakan sebagai *input* yang terhubung pada *push button up*.
- 7) *Port PH1/TXD2*  
*Port PH1/TXD2* digunakan sebagai *input* yang terhubung pada *push button down*.
- 8) *Port PH6/OC2B*  
*Port PH6/OC2B* digunakan sebagai *output* yang terhubung dengan servo 3.
- 9) *Port PB4/OC2A/PCINT4*  
*Port PB4/OC2A/PCINT4* digunakan sebagai *input* yang terhubung dengan DHT11.
- 10) *Port PB5/OC1A/PCINT5*  
*Port PB5/OC1A/PCINT5* digunakan sebagai *input* yang terhubung dengan pin modul *receiver*.

### 3. Sensor

#### a. Sensor Kelembaban Tanah YL-69

Sensor YL-69 dihubungkan dengan pin A0 pada Arduino Nano dengan tegangan operasi 5 volt. Hasil keluaran dari sensor berupa data analog, sehingga perlu dikonversi untuk melihat tingkat kelembaban tanah. Nilai analog pada sensor berada pada *range* 0 – 1023. Semakin besar nilai analog yang terbaca maka kelembaban tanah semakin kecil. Adapun baris program untuk mengukur kelembaban tanah sebagai berikut:

```
soilValue = analogRead(A0);  
    Serial.print("Tanah : ");  
    Serial.println(soilValue);  
    str_soilValue = String(soilValue);  
    persen = (100-(soilValue*0.0977));  
    Serial.print("Persentase : ");  
    Serial.println(persen);
```

Dari program di atas “ *soilValue* “ menunjukkan nilai analog yang dibaca sensor, sedangkan “ *persen* “ menunjukkan nilai konversi dari analog ke persentase. Untuk melakukan konversi dapat menggunakan rumus sebagai berikut:

$$\text{Kelembaban} = (100 - (\text{nilai sensor} \times 0.0977))$$

0.0977 diperoleh dari  $\frac{100}{1023}$  dimana 100 merupakan nilai konversi dalam bentuk persen dan 1023 merupakan nilai maksimum pembacaan sensor. Berdasarkan rumus tersebut didapatkan hasil konversi dalam bentuk persen (%) dengan persamaan sebagai berikut:

Misalkan data pembacaan sensor adalah 273. Maka dapat dihitung:

$$\begin{aligned}
\text{Kelembaban} &= (100 - (\text{nilai sensor} \times 0.0977)) \\
&= (100 - (273 \times 0.0977)) \\
&= 100 - 26.67
\end{aligned}$$

$$\text{Kelembaban} = 73.33 \%$$

b. Sensor DHT11

Pengujian sensor suhu DHT11 dilakukan dengan membandingkan hasil pembacaan sensor dengan termometer. Pengujian diambil sebanyak 10 kali dengan waktu yang berbeda. Berdasarkan pengujian secara keseluruhan maka didapatkan rata – rata *error* sebesar 1.85 %, yang berarti keakuratan sensor sudah cukup baik.

#### 4. RF 433 MHz

a. *Transmitter*

Inti modul *transmitter* merupakan resonator SAW (*Surface acoustic wave*) yang disetel untuk frekuensi 433.xx MHz. Ketika diberi logika tinggi pada *input* data, osilator akan menghasilkan *output carrier wave* RF konstan pada 433.x MHz dan pada saat diberi logika rendah osilator berhenti. Modul *transmitter* terhubung dengan Arduino Nano yang terletak pada rangkaian Node. *Transmitter* berfungsi mengirimkan hasil pembacaan sensor kelembaban dan *water level*. Adapun baris program untuk *transmitter* sebagai berikut:

```

    void bacaSensor() {
    soilValue = analogRead(A0);
    Serial.print("Tanah : ");
    Serial.println(soilValue);
    str_soilValue = String(soilValue);
    persen = (100-(soilValue*0.0977));
    Serial.print("Persentase : ");
    Serial.println(persen);

    levelValue1 = analogRead(level_sensor1);
    str_levelValue = String(levelValue1 );
    Serial.print("Level1 : ");
    Serial.println(levelValue1);

    levelValue2 = analogRead(level_sensor2);
    Serial.print("Level2 : ");
    Serial.println(levelValue2);
    delay(1000);
    }

    void dataKirim() {
    bacaSensor();
    allSensor = str_soilValue + "," + str_levelValue;

    static char *msg = allSensor.c_str();
    rf_driver.send((uint8_t *)msg, strlen(msg));
    rf_driver.waitPacketSent();
    delay(1000);
    }

```

RF 433 MHz hanya dapat mengirimkan data dalam bentuk *String* sehingga perlu konversi dari tipe *Int* ke *String*. Berdasarkan program di atas terdapat dua variabel pembacaan sensor yang akan dikirim, yaitu kelembaban tanah dan ketinggian air pada titik pertama. Untuk dapat mengirimkan data, maka kedua

variabel tersebut digabungkan pada program “ `allSensor = str_soilValue +  
", " + str_levelValue; “.`

#### b. *Receiver*

*Receiver* terdiri dari sirkuit *RF tuned* dan beberapa Op-Amp untuk memperkuat gelombang pembawa yang diterima dari *transmitter*. Sinyal yang telah diperkuat akan diumpankan ke PLL (*Phase Lock Loop*) yang memungkinkan decoder untuk mengunci ke aliran bit digital yang memberikan *output* lebih baik. Modul *receiver* dihubungkan dengan Arduino Mega pada rangkaian dam. *Receiver* berfungsi menerima data dari *transmitter*. Adapun baris programnya sebagai berikut:

```
void dataTerima() {  
  uint8_t buf[8];  
  uint8_t buflen = sizeof(buf);  
  if (rf_driver.recv(buf, &buflen)) {  
  
    allSensor= String((char*)buf);  
  
    for(int i=0; i<allSensor.length(); i++){  
if(allSensor.substring(i,i+1)== ","){  
    soilValue = allSensor.substring(0,i);  
    levelValue1 = allSensor.substring(i+1);  
    break;  
  }}  
}}
```

Sebelumnya pada baris program *transmitter* variabel pembacaan sensor digabungkan sebelum dikirim. Program di atas digunakan untuk memisahkan

kembali dua variabel yang telah diterima. Variabel tersebut masih dalam tipe data *String* yang selanjutnya harus dikonversi ke tipe *Int*.

## 5. Pengiriman Data ke Web Thinger.io

Thinger.io berfungsi sebagai *server* untuk menampilkan dan menyimpan data hasil pembacaan sensor. Adapun baris programnya sebagai berikut:

```
#include <ThingerTinyGSM.h>
#define USERNAME "ipung"
#define DEVICE_ID "TA_Irigasi_Drainase"
#define DEVICE_CREDENTIAL "bismillah1998"
#define APN_NAME "3gprs"
#define APN_USER "3gprs"
#define APN_PSWD "3gprs"
ThingerTinyGSM thing(USERNAME, DEVICE_ID,
DEVICE_CREDENTIAL, SerialAT);
void setup(){
thing.setAPN(APN_NAME, APN_USER, APN_PSWD);

thing["Buka"] << digitalPin(2);
thing["Tutup"] << digitalPin(3);

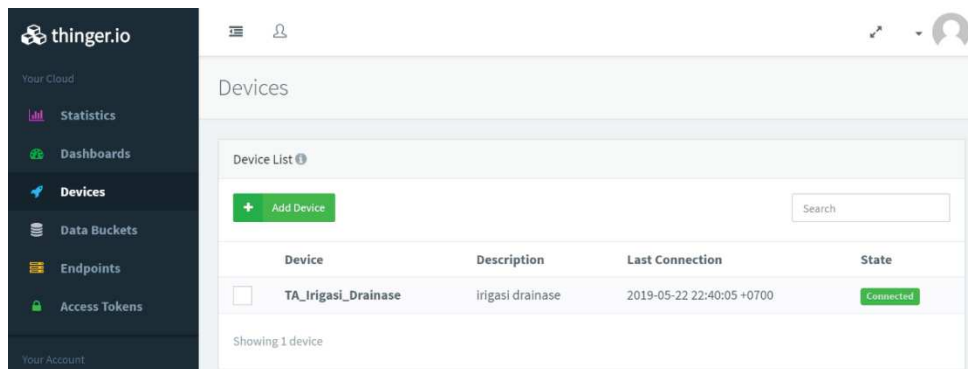
thing["sensor"]>>[] (pson& out){
    out["Kelembaban Tanah (%)"] =
    str_PercensoilValue;
    out["Suhu Udara(C)"] = temp;
    out["Kelembaban Udara(%)"] = hum;
    out["Pintu Air"] = Status;
};
}
void loop(){
thing.handle();}
```

Sebelum membuat program di atas terlebih dahulu membuat akun untuk *Device* pada Thinger.io. Pada saat membuat akun terdapat blok untuk mengisi

*Device ID* dan *Device Credential*. Selanjutnya yaitu menuliskan *Device ID* dan *Device Credential* kembali pada program. Pada program diatas “ USERNAME “ diisi nama akun Thinger.io. Kemudian pada baris yang terdapat kata “ APN “ diisi sesuai dengan *provider* yang digunakan.

Berikut adalah tampilan data pada thinger.io:

- a. Saat alat sudah terhubung ke internet maka pada halaman *device* akan terdapat keterangan “*connected*” seperti pada gambar berikut :



Gambar 26. Tampilan Alat Sudah Terkoneksi dengan Thinger.io

b. Untuk melihat hasil pembacaan sensor yaitu dengan masuk pada halaman *Dashboards*, maka akan muncul tampilan seperti pada gambar berikut :



Gambar 27. Tampilan Kondisi Pertama

Gambar di atas merupakan tampilan pada kondisi pertama, dimana saat kelembaban tanah 67 % sehingga status pintu air dalam keadaan terbuka.

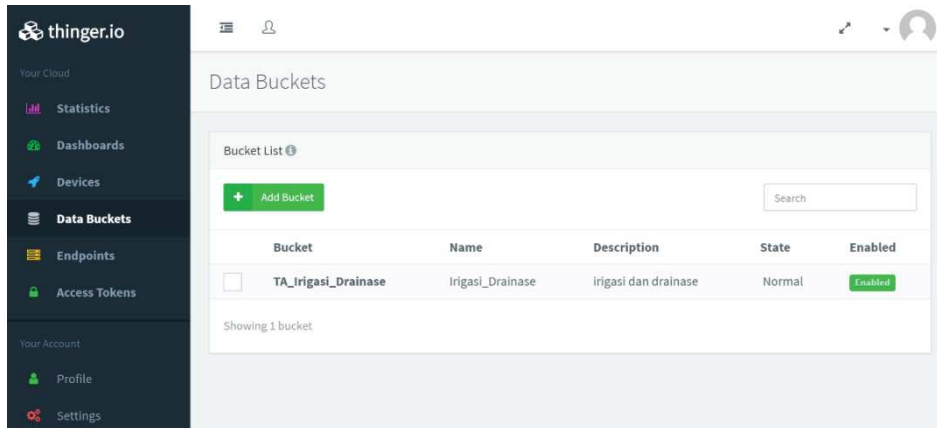


Gambar 28. Tampilan Kondisi Kedua

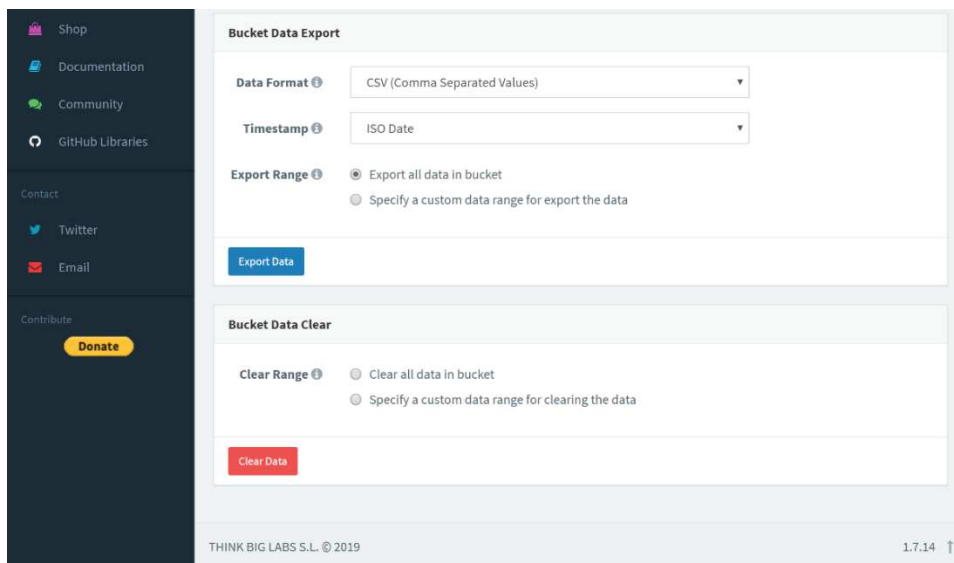
Gambar di atas merupakan tampilan pada kondisi kedua, dimana saat kelembaban tanah 77 % sehingga status pintu air berubah menjadi tertutup.



c. Mengunduh riwayat pembacaan sensor yaitu dengan masuk halaman *Data Buckets* kemudian pilih *Buckets* lalu *Export Data* seperti gambar berikut :



Gambar 29. Halaman *Data Buckets*



Gambar 30. *Export Data* Pada Halaman *Data Buckets*

Kemudian pilih *Export all data in bucket* dan klik *Export Data*. Data yang telah di ekspor dapat diunduh melalui notifikasi email dari thinger.io

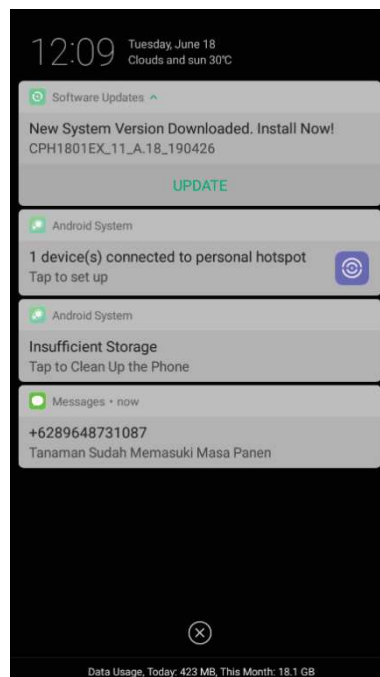
d. Data yang diunduh berupa tabel yang ditampilkan pada data Excel seperti berikut :

	A	B	C	D	E
1	ts	Kelembaban Tanah (%)	Kelembaban Udara(%)	Pintu Air	Suhu Udara(C)
2	2019-06-18T04:52:18.998	0	80.0	TERBUKA	31.0
3	2019-06-18T04:54:00.060	0	80.0	TERBUKA	32.0
4	2019-06-18T04:55:14.398	67	79.0	TERBUKA	32.0
5	2019-06-18T04:56:28.598	65	78.0	TERBUKA	32.0
6	2019-06-18T04:57:42.700	80	78.0	TERTUTUP	32.0
7	2019-06-18T04:58:56.859	78	78.0	TERTUTUP	31.0
8	2019-06-18T05:00:11.019	77	78.0	TERBUKA	32.0
9	2019-06-18T05:01:25.239	78	78.0	TERTUTUP	32.0
10	2019-06-18T05:02:39.819	78	76.0	TERBUKA	31.0
11	2019-06-18T05:03:53.458	78	77.0	TERTUTUP	32.0
12	2019-06-18T05:05:07.940	77	76.0	TERTUTUP	32.0
13	2019-06-18T05:06:43.299	78	77.0	TERTUTUP	33.0
14	2019-06-18T05:08:05.079	77	77.0	TERTUTUP	33.0

Gambar 31. Tampilan Data Pada Microsoft Excel

Pada tampilan di atas dapat dilihat beberapa kondisi. Banyaknya kondisi tergantung dari lamanya alat beroperasi.

e. Pemberitahuan melalui SMS



Gambar 32. Pemberitahuan melalui SMS

Setelah alat bekerja selama periode tertentu maka saat tanaman sudah memasuki masa panen, alat akan memberikan notifikasi melalui SMS.

#### **6. Kinerja Alat Secara Keseluruhan**

Unjuk kerja alat secara keseluruhan merupakan kombinasi dari rangkaian *hardware* dan *software* yang terintegrasi menjadi sebuah sistem. Sesuai dengan hasil pengujian alat sebelumnya, alat ini dapat bekerja dengan baik. Saat pertama alat ini dijalankan maka pada rangkaian sensor akan langsung bekerja. Namun pada rangkaian Dam akan terdapat sedikit *delay* yaitu saat alat mencari koneksi. Setelah mendapat koneksi maka sistem akan langsung bekerja.