

BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. HASIL IMPLEMENTASI

Setelah sistem dianalisis dan dirancang secara rinci, maka akan menuju tahap implementasi alat, adapun hasil dari tahap implementasi tersebut sebagai berikut :



Gambar 5.1 Bentuk Fisik Alat

Alat diatas merupakan alat pengemas gula berbasis mikrokontroler arduino uno yang telah penulis rancang. Penyangga alat ini terbuat dari kayu, lalu terdapat

sensor *load cell*, relay, LCD 16x2, motor servo, *push button*, pemanas, serta rangkaian keseluruhan dari alat pengemas gula tersebut. Sensor *load cell* digunakan untuk mendeteksi berat gula yang dikeluarkan motor servo, LCD berfungsi sebagai output seberapa banyak gula yang terdeteksi oleh *load cell*.

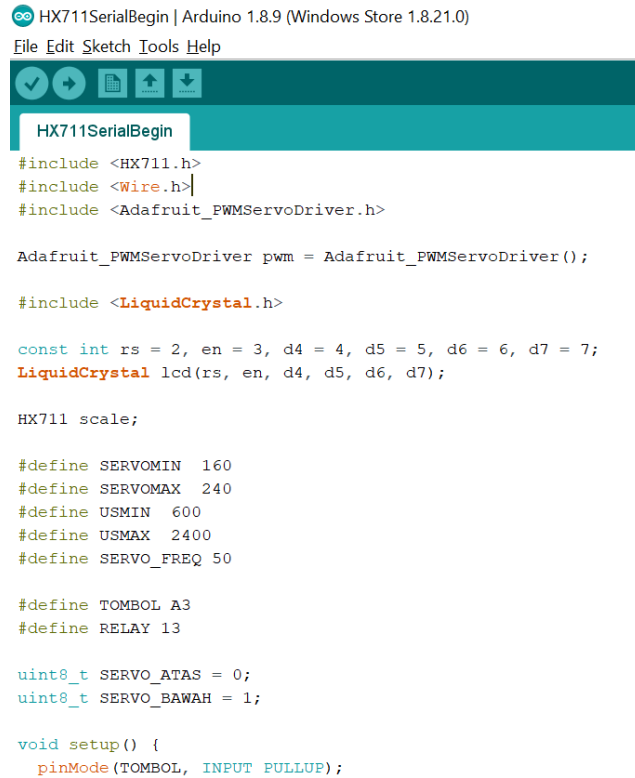
5.2 PENGUJIAN SISTEM

Pengujian sistem bertujuan untuk memastikan bahwa semua fungsi sistem berfungsi dengan baik dan untuk mencari kesalahan yang mungkin terjadi. Pada tahap pengujian sistem terdapat 2 jenis yaitu pengujian perangkat lunak dan pengujian perangkat keras.

5.2.1 Pengujian Perangkat Lunak

5.2.1.1 Arduino IDE

Dalam pembuatan alat ini penulis menggunakan Arduino IDE sebagai *software* pembuatan koding program. Program dibuat agar alat dapat bekerja sesuai dengan fungsi yang sudah dijelaskan oleh penulis. Berikut ini merupakan gambar program yang dibuat untuk alat ini :



```
HX711SerialBegin | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help
HX711SerialBegin
#include <HX711.h>
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>

Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver();

#include <LiquidCrystal.h>

const int rs = 2, en = 3, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

HX711 scale;

#define SERVOMIN 160
#define SERVOMAX 240
#define USMIN 600
#define USMAX 2400
#define SERVO_FREQ 50

#define TOMBOL A3
#define RELAY 13

uint8_t SERVO_ATAS = 0;
uint8_t SERVO_BAWAH = 1;

void setup() {
  pinMode(TOMBOL, INPUT_PULLUP);
```

Gambar 5.2 Pendeklarasian

Pada gambar 5.2 merupakan koding untuk mendeklarasikan alat sehingga lebih mudah untuk di panggil.



```
HX711SerialBegin | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help

HX711SerialBegin

void setup() {
  pinMode(TOMBOL, INPUT_PULLUP);
  pinMode(RELAY, OUTPUT);
  Serial.begin(9600);
  scale.begin(A1, A0);
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  scale.set_scale(2280.f);
  scale.tare();
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  pwm.begin();
  pwm.setOscillatorFrequency(27000000);
  pwm.setPWMFreq(SERVO_FREQ);
  lcd.begin(16, 2);
  delay(10);
  buka_bawah();
  tutup_atas();
}
```

Gambar 5.3 Tampilan Program *Load Cell*

Pada gambar 5.3 merupakan program load cell untuk mendapatkan nilai ketika sensor menerima input.




```
HX711SerialBegin | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help

HX711SerialBegin

void setup() {
  pinMode(TOMBOL, INPUT_PULLUP);
  pinMode(RELAY, OUTPUT);
  Serial.begin(9600);
  scale.begin(A1, A0);
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  scale.set_scale(2280.f);
  scale.tare();
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  pwm.begin();
  pwm.setOscillatorFrequency(27000000);
  pwm.setPWMFreq(SERVO_FREQ);
  lcd.begin(16, 2);
  delay(10);
  buka_bawah();
  tutup_atas();
}
```

Gambar 5.4 Tampilan Program Motor Servo

Pada gambar diatas PWM untuk motor servo diatur frekuensinya sebesar 27000000.



```

HX711SerialBegin | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help
HX711SerialBegin

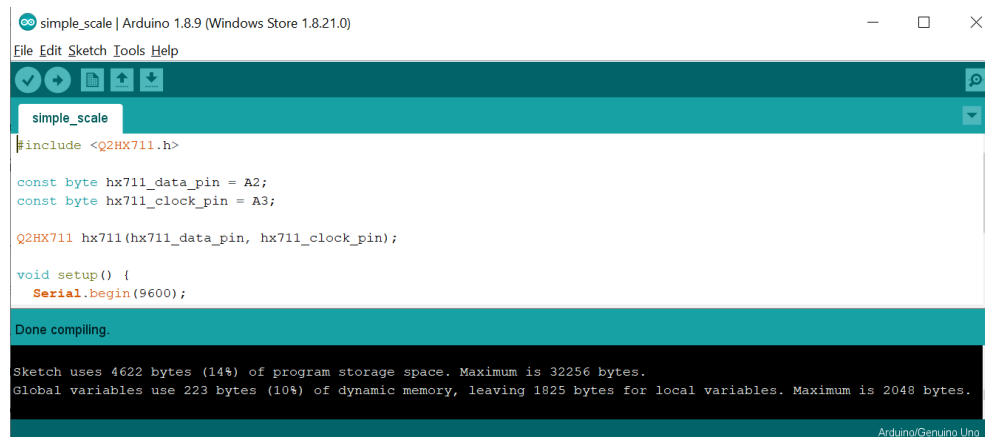
void setup() {
  pinMode(TOMBOL, INPUT_PULLUP);
  pinMode(RELAY, OUTPUT);
  Serial.begin(9600);
  scale.begin(A1, A0);
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  scale.set_scale(2280.f);
  scale.tare();
  Serial.println(scale.read());
  Serial.println(scale.read_average(20));
  Serial.println(scale.get_value(5));
  Serial.println(scale.get_units(5), 1);
  pwm.begin();
  pwm.setOscillatorFrequency(27000000);
  pwm.setPWMFreq(SERVO_FREQ);
  lcd.begin(16, 2);
  delay(10);
  buka_bawah();
  tutup_atas();
}

```

Gambar 5.5 Tampilan Program LCD

Pada gambar 5.5 program LCD dimasukan kearduino. Program delay sendiri untuk memberikan jeda berapa lama nilai yang terdeteksi muncul pada LCD.

Setelah *sketch* program telah selesai dibuat, maka tahap berikutnya adalah menyimpan *sketch* yang telah dibuat dengan memilih menu File kemudian *Save*, setelah disimpan langkah selanjutnya *verify/compile* program yang dibuat dengan menekan tombol CTRL+R, apabila program yang kita buat tidak ada kesalahan pengetikkan maka akan tampil seperti dibawah ini :



```

simple_scale | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help
simple_scale
#include <Q2HX711.h>

const byte hx711_data_pin = A2;
const byte hx711_clock_pin = A3;

Q2HX711 hx711(hx711_data_pin, hx711_clock_pin);

void setup() {
  Serial.begin(9600);
}

Done compiling.

Sketch uses 4622 bytes (14%) of program storage space. Maximum is 32256 bytes.
Global variables use 223 bytes (10%) of dynamic memory, leaving 1825 bytes for local variables. Maximum is 2048 bytes.
Arduino/Genuino Uno

```

Gambar 5.6 Tampilan Program yang Tidak Error

Namun jika terdapat kesalahan dalam pengetikkan program maka akan tampil seperti dibawah ini :



```

simple_scale | Arduino 1.8.9 (Windows Store 1.8.21.0)
File Edit Sketch Tools Help
simple_scale
#include <Q2HX711.h>

const byte hx711_data_pin = A2;
const byte hx711_clock_pin = sA3;

Q2HX711 hx711(hx711_data_pin, hx711_clock_pin);

void setup() {
  Serial.begin(9600);
}

'sA3' was not declared in this scope
exit status 1
'sA3' was not declared in this scope
Copy error messages

```

Gambar 5.7 Tampilan Program yang Terdapat Error

5.3 PENGUJIAN ALAT

Pengujian perangkat keras ini dilakukan untuk mengetahui benar atau tidaknya sebuah rangkain listrik yang telah di rangkai. Pengujian tegangan rangkaian alat dilakukan guna mengetahui arus yang diterima oleh rangkaian, pengujian ini menggunakan alat multi tester. Setelah pengujian tegangan

rangkaian, dilanjutkan dengan pengujian fungsi dari masing-masing rangkaian apakah berjalan sesuai dengan yang diinginkan atau tidak. Tahap terakhir adalah melakukan pengujian keseluruhan.

5.3.1 Pengujian Perangkat Keras

5.3.1.1 Pengujian Tegangan Arduino

Rangkaian ini merupakan otak dari seluruh rangkaian. Semua rangkaian yang terdapat pada alat ini dikendalikan *input output*-nya oleh rangkaian mikrokontroler Arduino ini. Pengujian rangkaian ini dilakukan dengan menghubungkan pin VCC(+) dan pin GND(-) pada multi tester. Adapun hasil dari pengujian tegangan ini dapat dilihat pada tabel 5.1 :

Tabel 5.1 Pengujian Tegangan Arduino

Sumber Arus	Tegangan Input	Tegangan <i>Output</i>
Adaptor 5 Volt	5 Volt	4.6 Volt

5.3.1.2 Pengujian Sensor *Load Cell*

Pengujian ini dilakukan untuk mengetahui apakah sensor *load cell* dapat mendeteksi berat gula dari 0 kg sampai 1 kg. Sensor ini menerima input tegangan dari 5V sampai 10V. Berikut ini merupakan tabel hasil pengujian sensor *load cell* :

Tabel 5.2 Pengujian Sensor *Load Cell*

Pengujian (kg)				Rata-Rata
1	2	3	4	
0 kg	0 kg	0 kg	0 kg	0 kg
0,55 kg	0,54 kg	0,51kg	0,69 kg	0,57 kg
1,11 kg	1,01 kg	1,04 kg	1,09 kg	1,06 kg

5.3.1.3 Pengujian *Push Button*

Pengujian ini dilakukan agar dapat mengetahui apakah tombol yang terdapat pada rangkaian berfungsi. Pengujian ini dilakukan agar dapat mengetahui apakah *push button* ketika ditekan dapat mengaktifkan servo penampung gula dan alat-alat yang terdapat pada alat pengemas.

Tabel 5.3 *Push Button*

Push Button		
Tekan	0	Low
Tidak	1	High

Tabel 5.3 menjelaskan bahwa ketika *push button* ditekan akan memberikan nilai 0 yang akan mengaktifkan alat untuk berfungsi. Namun ketika tidak ditekan maka nilainya 1 sehingga alat tidak akan berfungsi.

5.3.1.4 Pengujian Rangkaian Pemanas

Pengujian ini dilakukan agar dapat mengetahui apakah rangkaian pemanas dapat memanaskan besi elemen impuls ketika diberi tegangan 220 V AC.

Tabel 5.4 Pemanas

Arus AC (V)	Elemen Impuls
220 V	Panas
0 V	Tidak Panas

Berdasarkan tabel 5.4 ketika elemen impuls diberikan tegangan sebesar 220 V AC maka besi akan panas. Namun ketika besi elemen impuls tidak diberikan tegangan maka besi elemen impuls tidak akan panas.

5.3.1.5 Pengujian Rangkaian Relay dan Pemanas

Pengujian ini dilakukan agar dapat mengetahui apakah rangkaian pemanas dapat berfungsi dengan baik.

Tabel 5.5 Relay dan Pemanas

Input	Relay		Pemanas
	NO	NC	
0	Terhubung	Tidak Terhubung	Hidup
1	Tidak Terhubung	Terhubung	Mati

Pada tabel 5.5 menjelaskan kondisi pemanas ketika diberi input 0 maka akan hidup, sedangkan ketika diberi input 1 maka pemanas akan mati.

5.3.1.6 Pengujian LCD

Pengujian LCD dilakukan untuk mengetahui hasil dari sensor *load cell* yang akan ditampilkan pada LCD. Pengujian rangkaian menggunakan multi tester, yang didapatkan bahwa LCD menerima tegangan sebesar 5V. Berikut merupakan tabel hasil pengujian

Tabel 5.6 Pengujian LCD

Pengujian	Tampilan LCD
1	0,55 kg
2	0,54 kg
3	0,51 kg
4	0,69 kg

Setelah dilakukan pengujian, berdasarkan data dari tabel 5.6 LCD berhasil untuk menampilkan berat dari gula yang diterima oleh sensor.

5.3.1.7 Pengujian Motor Servo

Posisi servo dikendalikan menggunakan pwm 50Hz sinyal. Pengujian dilakukan dengan memberikan pulsa 0.3 ms, dan 0.6 ms.

Tabel 5.7 Pengujian Motor Servo

Periode Sinyal (ms)	Arah Putaran Servo
0.3	Kearah 45 derajat
0.6	Kearah 60 derajat

Pada tabel 5.7 menjelaskan bahwa ketika motor servo diberikan sinyal 0.3 ms akan bergerak kearah 45 derajat. Lalu jika diberikan sinyal selama 0.6 ms maka motor servo akan bergerak kearah 60 derajat.

Tabel 5.8 Pengujian Motor Servo 1

PULSA(ms)	Gerakan
190 ms	Terbuka/ 90 Derajat
280 ms	Tertutup/ 0 Derajat

Pada tabel 5.8 dapat disimpulkan bahwa ketika servo 1 diberi pulsa selama 190 ms, servo bergerak sebesar 90 derajat atau terbuka. Sedangkan ketika servo 1 diberi pulsa selama 280 ms maka servo akan tertutup.

Tabel 5.9 Pengujian Motor Servo 2

PULSA (ms)	Pemanas
160	Mengepres

240	Tidak Mengepres
-----	-----------------

Dari tabel 5.9 dapat diketahui bahwa ketika servo 2 diberikan pulsa selama 160 ms maka alat pemanas akan bergerak untuk mengepres plastik, sedangkan ketika servo 2 diberikan pulsa selama 240 ms maka alat pemanas tidak mengepres plastik.

5.3.1.8 Pengujian Keberhasilan Mengepres Plastik

Disini pengujian menggunakan 2 jenis plastik. Plastik A memiliki bahan yang lebih tebal, sedangkan plastik B memiliki bahan yang lebih tipis dan tidak sekaku plastik A, pengujian dilakukan sebanyak 5 kali.

Tabel 5.10 Keberhasilan Mengepres Plastik

Pengujian	Jenis Plastik	
	A	B
1	Berhasil	Gagal
2	Berhasil	Gagal
3	Gagal	Berhasil
4	Gagal	Gagal
5	Berhasil	Gagal

Pada tabel 5.10 didapatkan hasil bahwa ketika menggunakan plastik jenis A percobaan yang berhasil dilakukan adalah 3 kali, sedangkan ketika menggunakan plastik jenis B percobaan yang berhasil hanya 1 kali.

5.3.1.9 Pengujian Alat Keseluruhan

Setelah semua rangkaian dan alat telah diuji apakah berfungsi atau tidak, maka langkah selanjutnya yaitu menguji alat secara keseluruhan dan

membandingkan seberapa efektif penggunaan alat pengemas gula ini dengan pengemasan secara tradisional.

Tabel 5.11 Pengemasan Menggunakan Alat Pengemas Gula

No	Pengujian(s)
1	55 s
2	57 s
3	58 s
4	60 s
5	55 s
6	54 s
7	60 s
8	52 s
9	59 s
10	62 s
Rata-Rata	57,2 s

Pada tabel 5.11, setelah dilakukan pengujian sebanyak 10 x didapatkan bahwa rata-rata untuk menyelesaikan 1 pengemasan plastik dibutuhkan waktu selama 57,2 sekon/detik.

Tabel 5.12 Pengemasan Menggunakan Cara Tradisional

No	Pengujian(s)
1	63 s
2	65 s
3	70 s
4	61 s
5	59 s
6	69 s
7	71 s
8	68 s
9	60 s
10	65 s
Rata-	65,1 s

Rata	
------	--

Pada tabel 5.12, setelah dilakukan percobaan sebanyak 10 x dengan menggunakan cara tradisional waktu rata-rata yang dibutuhkan untuk menyelesaikan 1 pengemasan plastik yaitu sebesar 65,1 sekon/detik.

5.4 ANALISIS SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan analisa rangkaian keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, dapat dikatakan bahwa alat ini berfungsi sesuai yang penulis inginkan.

Pengujian ini dilakukan untuk menunjukkan bahwa perancangan alat pengemas gula ini dapat bekerja sesuai dengan tujuan dari pembuatan. Pengujian ini dilakukan dengan cara sebagai berikut :

1. Buka file program Arduino yang telah dibuat setelah itu klik “ tools “ untuk mencari Port COM yang tersedia, sambungkan dahulu USB alat ke laptop, selanjutnya klik “ Serial Monitor “ untuk melihat nilai program yang ingin diuji.
2. Setelah semua program dijalankan maka langkah selanjutnya menguji alat, ketika USB tersambungkan, alat otomatis hidup tetapi belum mulai melakukan proses pengemasan.
3. Langkah selanjutnya yaitu meletakkan plastik ukuran 1 kg, kemudian tekan tombol yang akan mengaktifkan motor servo 1 yang berada di wadah penampung gula. Kemudian setelah sensor *load cell* menerima inputan

sebesar 1 kg yang jumlahnya akan muncul pada LCD, maka otomatis motor servo 1 akan mati. Setelah itu motor servo 2 yang berada pada pemanas akan bergerak untuk merekatkan plastik yang sudah berisi gula seberat 1 kg.