

BAB III

METODOLOGI PERANCANGAN

3.1 Komponen dan Persediaan

Langkah pertama dalam perancangan ini adalah membuat tabel komponen dan persediaan. Tabel komponen dan persediaan pada perancangan ini berisi tentang komponen-komponen utama dengan kuantitas lebih dari kebutuhan perancangan karena untuk persediaan bila terjadi kegagalan. Tabel komponen dan persediaan pada perancangan ini dapat dilihat pada Tabel 3.1 dan 3.2.

Tabel 3. 1 Komponen dan Persediaan

No.	Nama Komponen & Persediaan	Jumlah
1	Arduino Pro Mini 168 3.3V	1
2	Arduino Pro Mini 328P 5V	1
3	Tower Pro MG90s	8
4	Futaba 3003S	1
5	Adafruit 16-chanel servo driver	1
6	Kapasitor Elektrolit 2200uF 16V	1
7	ESP01	2
8	HMC5883L	1
9	GY61	1
10	LM35	1
11	Relay 1 CH	1
12	LCD 16x2	1
13	Buzzer	2
14	Blower	1
15	MP1584	4
16	Adapter 5V/2A	1
17	Adapter 12V/3A	1
18	Jack Power DC	2
19	PCB Bolong	2
20	Resistor Variabel 10kOhm	1
21	Tombol On/Off	4
22	Pin Header Male	8
23	Pin Header Female	1
24	Kabel Jumper Female-Female	80

Tabel 3. 2 Komponen dan Persediaan (lanjutan)

No.	Nama Komponen & Persediaan	Jumlah
25	Kabel Jumper Male-Female	40
26	Tinol	1
27	Akrilik lembaran 700x400	1
28	<i>Cable Ties</i>	100
29	<i>Rubber Spon</i>	1
30	Mur dan Baut 3x20	20
31	Mur dan Baut 2x20	40
32	Sarung Tangan	1
33	Perekat Kain	1
34	Lem Akrilik	1
35	Lem Bakar	2
36	Isolasi Hitam	1
37	Benang Jahit	1

3.2 Alat dan Mesin yang Diperlukan

Langkah kedua pada perancangan ini adalah membuat tabel alat dan mesin yang diperlukan. Alat dan mesin yang dimaksud adalah alat dan mesin yang bertugas hanya untuk membantu perancangan atau dalam kata lain tidak menjadi bagian komponen rancangan. Alat dan mesin yang diperlukan pada perancangan ini dapat dilihat pada Tabel 3.3.

Tabel 3. 3 Alat dan Mesin yang Diperlukan

No.	Nama Alat dan Mesin	Jumlah
1	FTDI	2
2	Obeng	1
3	Tang	1
4	Jarum Jahit	1
5	Gunting	1
6	Penggaris 30cm	1
7	Bor	1
8	Mata bor diameter 2 mm	1
9	Korek Api	1
10	Mata bor diameter 3 mm	1
11	Solder	1
12	MiFi	2

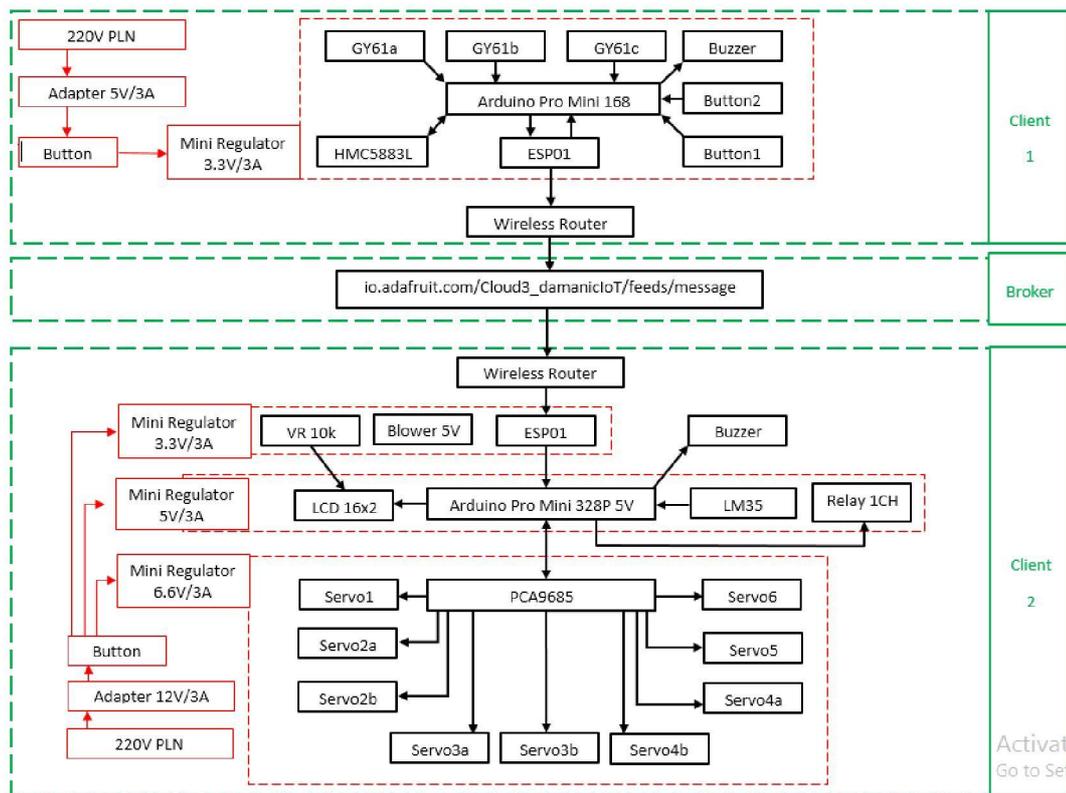
3.3 Aplikasi dan Layanan Online

Langkah ketiga adalah membuat tabel aplikasi dan layanan online. Pada perancangan ini digunakan 4 buah aplikasi dan 1 layanan online. Aplikasi dan layanan online pada perancangan ini dapat dilihat pada Tabel 3.4.

Tabel 3. 4 Aplikasi dan Layanan Online

No.	Nama Aplikasi dan Layanan Online
1	SketchUp
2	Adobe Illustrator
3	Arduino Ide
4	Adobe Premier
5	Adafruit IO

3.4 Sistem



Gambar 3. 1 Diagram blok Sistem

Langkah keempat adalah pembuatan rancangan sistem. Pembuatan rancangan sistem pada langkah ini didasarkan pada komponen-komponen yang telah disebutkan pada Tabel 3.1.

Gambar 3.1 adalah diagram blok sistem alat yang dibagi oleh garis beda warna yaitu, warna merah, hitam, dan hijau. Berikut penjelasannya:

1. Garis putus-putus merah adalah tanda penyebaran daya sesuai dengan tegangan yang di butuhkan tiap komponen. Dimulai dari 220VAC PLN sebagai sumber, lalu diteruskan ke adapter untuk dikonversi menjadi tegangan searah 12V dan 5V. Kemudian diteruskan ke tombol sebagai saklar daya yang masuk ke regulator. Regulator memiliki fungsi utama mengubah nilai tegangan keluaran adapter menjadi tegangan masukan sesuai kebutuhan dan juga menstabilkan tegangan outputnya, yaitu 3.3V, 5V, dan 6.6V. Kemudian tegangan tersebut diteruskan ke pin VCC dan GND dikomponen-komponen yang membutuhkan tegangan
2. Garis putus-putus hitam adalah hubungan masukan dan keluaran antar komponen dalam memproses data. Dimulai dari sensor (GY61 dan HMC5883L) sebagai masukan data, lalu di proses oleh mikrokontoler (Arduino Pro Mini), dan ESP8266 (ESP01), *wireless router* yang di dalamnya terdapat kartu GSM berisi kuota internet untuk menghubungkan alat dengan jaringan internet, kemudian *platform IoT (Internet of Things)* (io.adafruit.com/Cloud3_damanicIoT/message) sebagai server internet, lcd sebagai monitor mode dan posisi sendi-sendi robot, servo driver (PCA9685) sebagai tambahan pin PWM pada papan arduino, hingga rotor servo yang bergerak sesuai dengan data sudut yang diterimanya
3. Garis putus-putus hijau adalah hubungan komponen-komponen MQTT antara *client 1* dan *client 2* yang dihubungkan dalam sebuah server (broker) io.adafruit.com melalui feed yang telah dibuat, yaitu “*message*” pada akun “*Cloud3_damanicIoT*”. Dalam sistem ini, *client 1* adalah *publisher* dan *Client 2* adalah *subscriber*

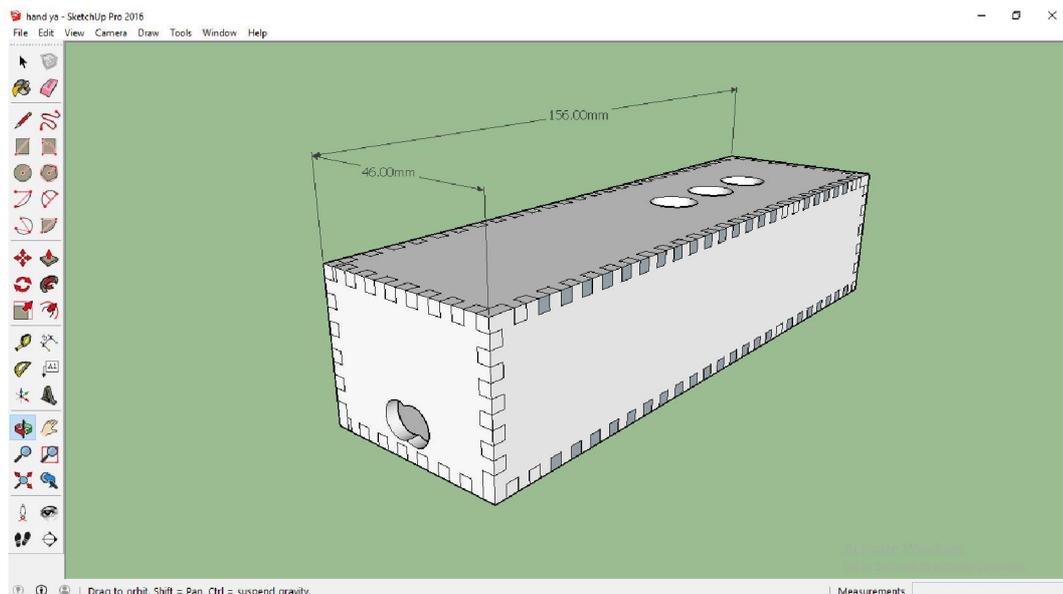
3.5 Desain

Langkah kelima adalah pembuatan desain. Pembuatan desain diawali dengan membuat desain 3D menggunakan aplikasi Sketch Up. Setelah desain 3D selesai kemudian dikonversi menjadi desain 2D menggunakan Adobe Illustrator. Pembuatan desain 3D pada perancangan ini di bagi menjadi 2, yaitu bagian pengendali dan robot.

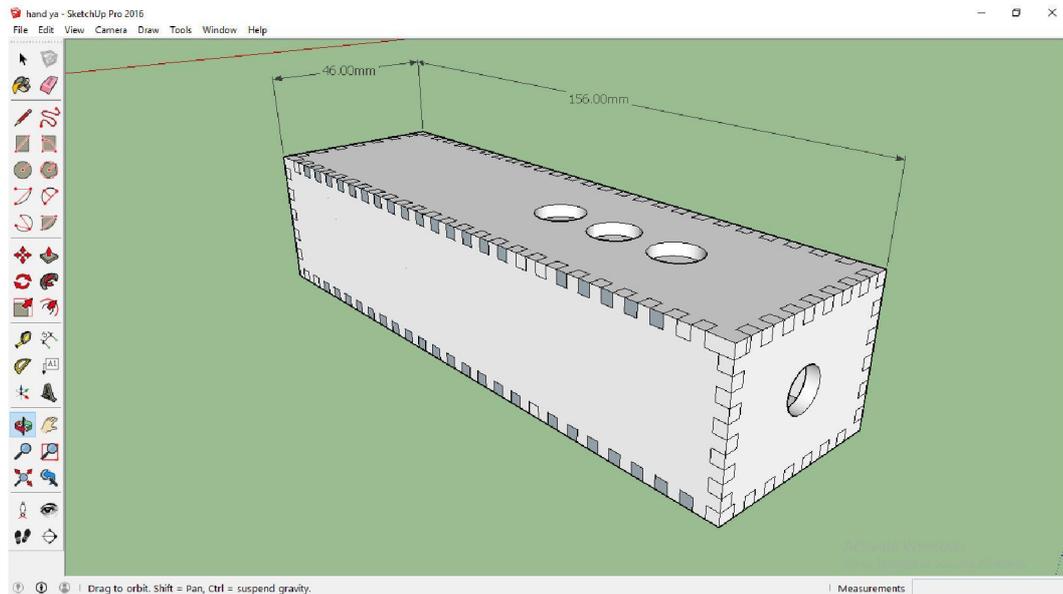
3.5.1 Pengendali

Desain 3D pengendali terdiri dari box, perekat kain, dan sarung tangan. Berikut penjelasannya:

1. Box berfungsi sebagai rumah bagi komponen-komponen elektronika selain sensor. Lubang pada sisi depan box berfungsi sebagai tempat keluarnya kabel-kabel yang terkoneksi dengan sensor. Lubang pada sisi belakang box berfungsi sebagai tempat masuknya terminal adapter ke rangkaian. Dan pada sisi atas terdapat 3 buah lubang berbentuk lingkaran sebagai tempat di letakkannya tiga buah tombol. Desain 3D box pengendali dapat dilihat pada Gambar 3.2 dan Gambar 3.3.

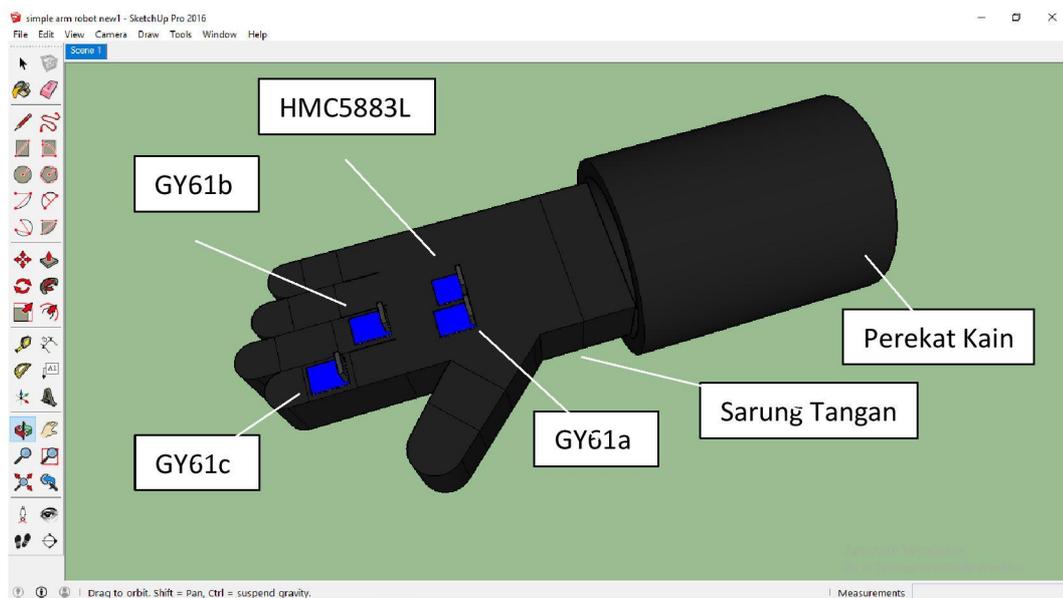


Gambar 3. 2 Desain 3D Box Pengendali (Sisi Kiri)



Gambar 3. 3 Desain 3D Box Pengendali (Sisi Kanan)

2. Perkat kain berfungsi untuk merekatkan box dengan tangan manusia. Perkat dipasang saling bertolak belakang dan dijahit dengan benang jahit supaya kuat. Perkat didesain seperti Gambar 3.4 agar dapat digunakan untuk berbagai ukuran tangan manusia.



Gambar 3. 4 Desain 3D Sarung, Tangan, Perkat, dan Peletakan Sensor

3. Sarung tangan berfungsi sebagai tempat diletakkannya 4 buah sensor, yaitu 3 buah GY61 dan sebuah HMC5883L

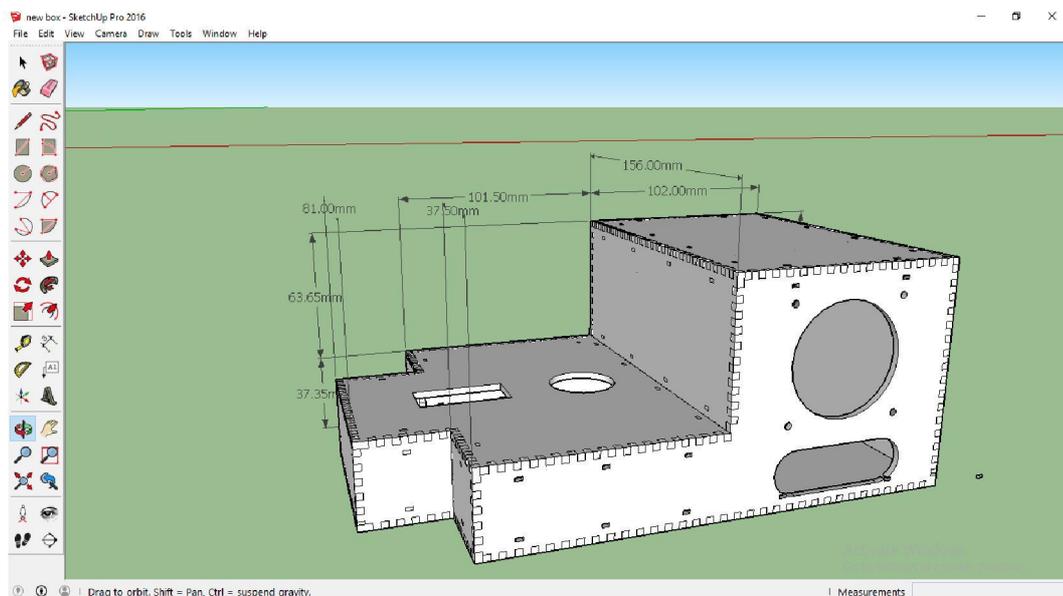
3.5.2 Robot

Desain 3D robot terdiri dari desain *base box* dan robot lengan. Berikut penjelasannya:

3.5.2.1 *Base Box*

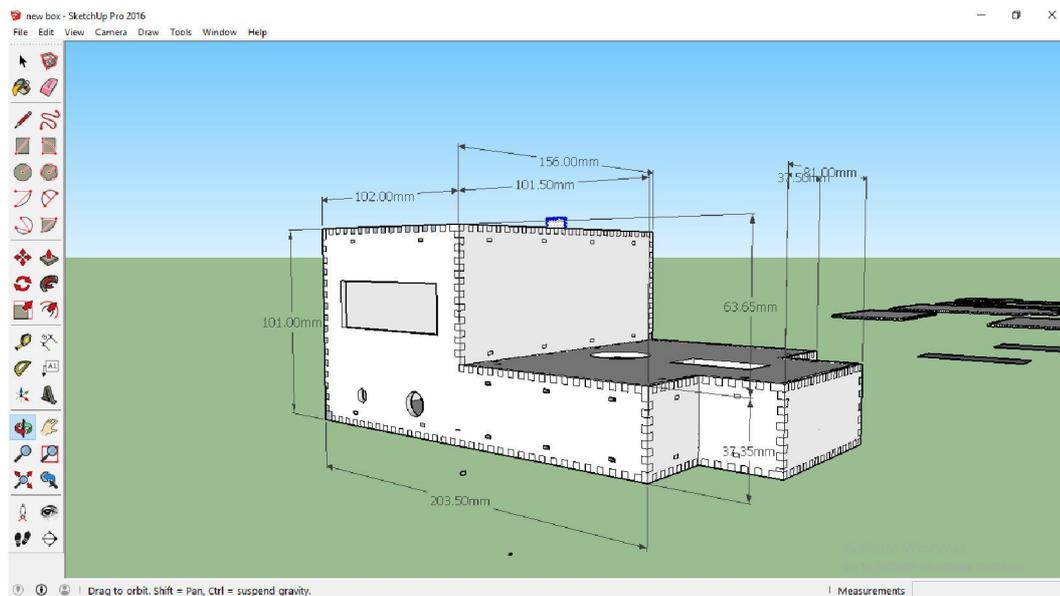
Base Box dibuat dengan bahan akrilik berketebalan 3mm dengan ukuran tiap sisi dapat dilihat pada Gambar 3.5 dan 3.6. *Base Box* berfungsi sebagai rumah semua komponen elektronik selain servo. Tujuannya adalah melindungi rangkaian dari kemungkinan yang dapat mengganggu sistem dan menambah nilai estetika.

Pada sisi kiri, terdapat lubang berbentuk lingkaran sebagai tempat peletakan blower 5V. Sementara lubang di bawahnya adalah tempat untuk terminal masukkan tegangan dari adapter 12V dan juga berfungsi sebagai *exhaust* udara *box*.



Gambar 3. 5 Desain 3D Box Robot (Sisi Kiri)

Pada sisi kanan, terdapat lubang berbentuk persegi panjang sebagai tempat peletakan LCD 16x2. Sementara di bawahnya terdapat 2 lubang berbentuk lingkaran besar dan kecil. Lingkaran besar adalah tempat peletakan tombol power, dan lingkaran kecil untuk tempat resistor variabel 10k dalam mengatur kontras LCD 16x2.

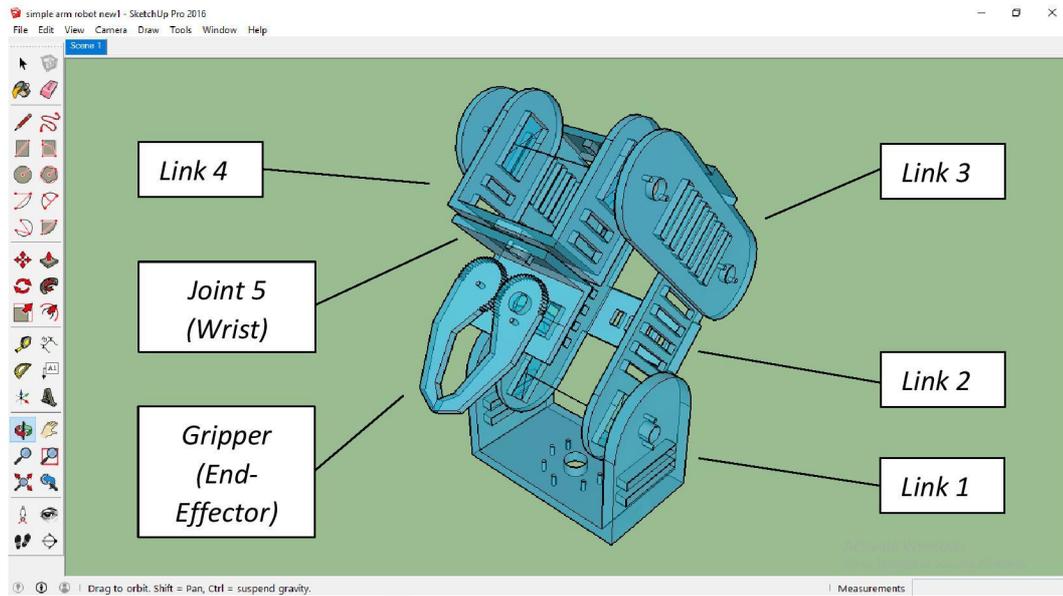


Gambar 3. 6 Desain 3d Box Robot (Sisi Kanan)

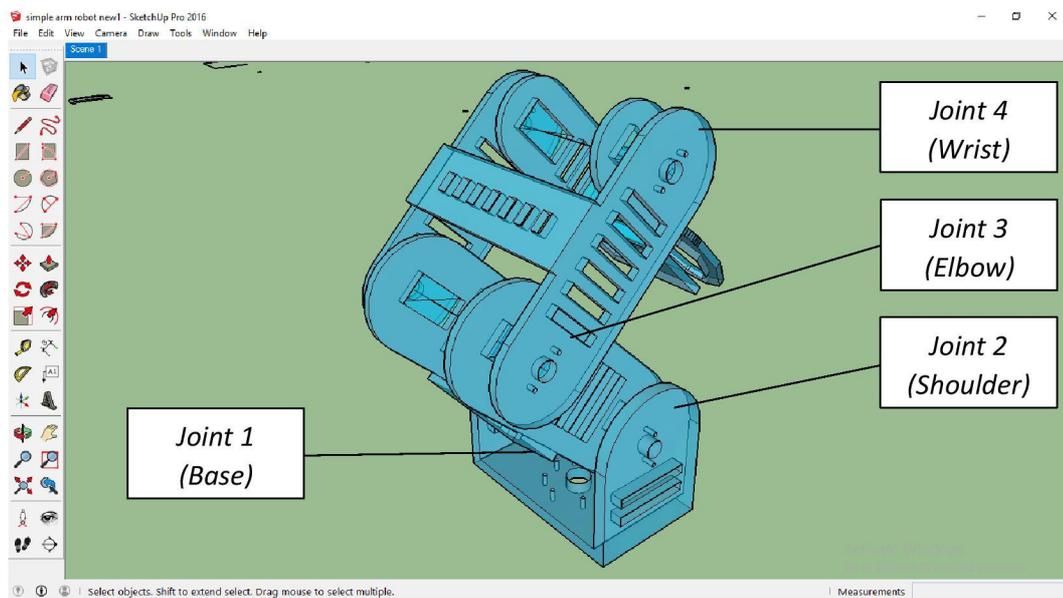
Pada sisi atas terdapat lubang berbentuk persegi panjang sebagai tempat peletakan servo yang bertugas sebagai titik tumpu beban robot. Sementara lubang berbentuk lingkaran sebagai tempat masuknya kabel-kabel motor servo pada robot.

3.5.2.2 Robot Lengan

Robot Lenan dibuat dengan bahan akrilik berketebalan 3mm. Lubang lingkaran pada desain adalah tempat baut-baut berukuran diameter 2mm dan 3mm, lubang berbentuk persegi panjang untuk menempelkan servo dengan badan akrilik, dan lubang rongga berbentuk persegi panjang berfungsi untuk mengurangi beban akrilik tanpa mengurangi kekuatan akrilik tersebut. Desain 3d robot dapat dilihat pada Gambar 3.7 dan 3.8.



Gambar 3. 7 Desain 3D Robot (Sisi Kiri)



Gambar 3. 8 Desain 3D Robot (Sisi Kanan)

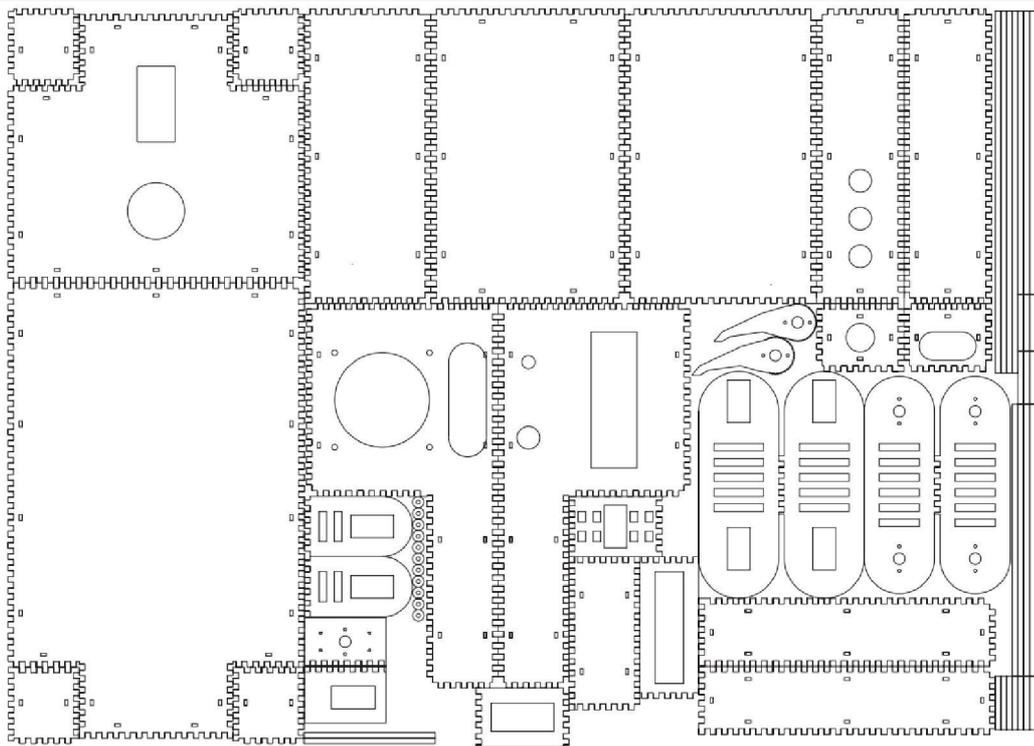
Lebar setiap *link* memiliki ukuran yang berbeda-beda. Semakin jauh dari *base* lebar *link* semakin mengecil. Hal ini bertujuan supaya beban yang ditopang *base* tidak terlalu berat. Sementara, panjang pada rancangan robot disesuaikan dengan dimensi komponen-komponen-komponen rancangan. Dimensi robot dapat dilihat pada Tabel 3.5.

Tabel 3. 5 Dimensi tiap Lengan Robot (link)

No	Lengan Penghubung (<i>link</i>)	Panjang (mm)	Lebar (mm)
1	<i>Link 1</i>	63.5	47
2	<i>Link 2</i>	107	42
3	<i>Link 3</i>	107	37
4	<i>Link 4</i>	63.5	32

3.5.3 Konversi Desain 3 Dimensi ke 2 Dimensi

Konversi dilakukan karena dalam proses pemotongan akrilik hanya bisa dilakukan dengan desain 2D. Pemotongan akrilik dilakukan menggunakan *laser cutting*. Konversi desain ke bentuk 2D dilakukan dengan menggunakan aplikasi Adobe Illustrator. Hasil konversi dapat dilihat pada Gambar 3.9.

**Gambar 3. 9** Konversi Desain 3D ke 2D

3.6 Pengkabelan

Langkah selanjutnya adalah membuat tabel pengkabelan. Pengkabelan pada perancangan ini dibagi menjadi dua sistem, yaitu pengendali dan robot. Berikut penjelasan pengkabelan pada perancangan ini:

3.6.1 Pengkabelan Sistem Pengendali

Tabel pengkabelan sistem pengendali dapat dilihat pada Tabel 3.6 dan 3.7. Tegangan yang dibutuhkan untuk komponen-komponen pada sistem ini adalah 3.3V.

Tabel 3. 6 Pengkabelan Bagian Pengendali

Dari		Ke	
Komponen	Pin	Pin	Komponen
Arduino Pro Mini 168 3.3V	VCC	VCC	MP1584
	GND	GND	
ESP01	VCC	VCC	Arduino Pro Mini 168 3.3V
	GND	GND	
	RX	TX	
	TX	RX	
HMC5883L	VCC	VCC	MP1584
	GND	GND	
	SCL	SCL	Arduino Pro Mini 168 3.3V
	SDA	SDA	
GY61a	VCC	VCC	MP1584
	GND	GND	
	Y	Pin	Arduino Pro Mini 168 3.3V
GY61b	VCC	VCC	MP1584
	GND	GND	
	Y	Pin	Arduino Pro Mini 168 3.3V
GY61c	VCC	VCC	MP1584
	GND	GND	
	X	Pin	Arduino Pro Mini 168 3.3V
	Y	Pin	
Tombol Merah	UJUNG A	Pin	MP1584
	Ujung B	GND	
Tombol Kuning	Ujung A	Pin	Arduino Pro Mini 168 3.3V
	Ujung B	GND	MP1584

Tabel 3. 7 Pengkabelan Bagian Pengendali (lanjutan)

Dari		Ke	
Komponen	Pin	Pin	Komponen
Tombol Biru	Ujung A	Pin	Arduino Pro Mini 168 3.3V
	Ujung B	GND	MP1584
Buzzer	VCC	Pin	Arduino Pro Mini 168 3.3V
	GND	GND	MP1584
MP1584	VCC (input)	VCC	Adapter 5V
	GND (input)	GND	

3.6.2 Pengkabelan Sistem Robot

Pengkabelan sistem robot dapat dilihat pada Tabel 3.8 dan Tabel 3.9. Pada pengkabelan sistem robot pemecahan alur listrik dari adapter dibagi menjadi 3, yaitu 3.3V, 5V, dan 6V. Pemecahan jalur ini dilakukan dengan menggunakan pin header yang di solder ke papan PCB.

Tabel 3. 8 Pengekabelan Bagian Robot

Dari		Ke	
Komponen	Pin	Pin	Komponen
Arduino Pro Mini 328 5V	VCC	VCC	MP1584 output 3.3V
	GND	GND	
ESP01	VCC	VCC	Arduino Pro Mini 328 5V
	GND	GND	
	RX	TX	
Tombol Merah	UJUNG A	Pin	MP1584
	UJUNG B	GND	
Buzzer	VCC	Pin	Arduino Pro Mini 328 5V
	GND	GND	MP1584
Relay 1CH	VCC	VCC	MP1584 output 5V
	GND	GND	Arduino Pro Mini 328 5V
	CH	Pin	
PCA9685	VCC	VCC	MP1584 output 5V
	GND	GND	Arduino Pro Mini 328 5V
	SCL	SCL	

Tabel 3. 9 Pengekabelan Bagian Robot (lanjutan)

Dari		Ke	
Komponen	Pin	Pin	Komponen
PCA9685	SDA	SDA	Arduino Pro Mini 328 5V
LCD16x2	VCC	VCC	MP1584 output 5V
	GND	Back Light (+)	LCD16x2
		GND	MP1584 output 5V
		Back Light (+)	LCD16x2
	RW		
	Contrast	CH	Potentio 10k
	RS	Pin 2	Arduino Pro Mini 328 5V
	EN	Pin 3	
	D0	Pin	
	D1	Pin	
D2	Pin		
Potensio 10k	VCC	VCC	MP1584 output 3.3V
	GND	GND	
Blower	VCC	VCC	MP1584 output 3.3V
	GND	GND	
MP1584 output 3.3V	GND	GND	MP1584b output 5V
		GND	MP1584c output 6.6V
	VCC (input)	VCC	Adapter 12V/3A
GND (input)	GND		
MP1584b output 5V	VCC (input)	VCC	
	GND (input)	GND	
MP1584c output 6.6V	VCC (input)	VCC	
	GND (input)	GND	

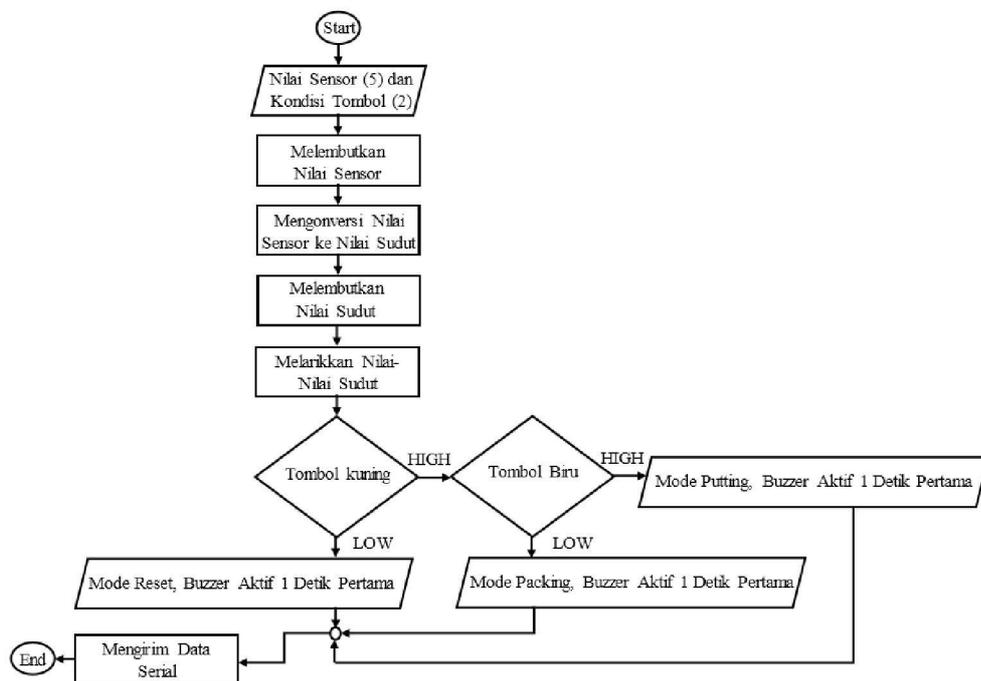
3.7 Diagram Alir Program

Langkah selanjutnya adalah membuat diagram alir program. Diagram alir program dibuat guna mempermudah penulis dalam menentukan prinsip kerja rancangan secara rinci. Diagram alir program pada perancangan ini dibagi menjadi beberapa bagian. Berikut penjelasannya:

3.7.1 Diagram Alir Program bagian Pengendali

Komponen-komponen yang diprogram pada bagian pengendali adalah Arduino Pro Mini dan ESP01. Arduino pro mini diprogram untuk mengolah data dari sensor lalu dikeluarkan ke ESP01. ESP01 menerima data dari Arduino Pro mini lalu meneruskannya ke server MQTT.

3.7.1.1 *Arduino Pro Mini*



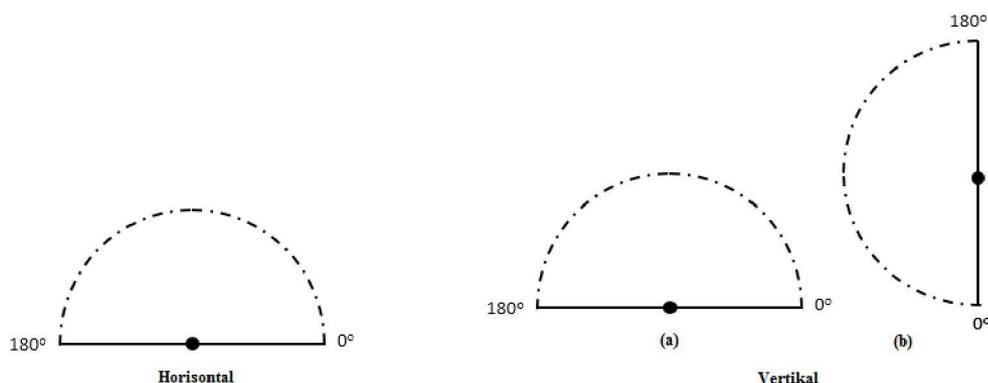
Gambar 3. 10 Diagram Alir Program pada Arduino Pro Mini Bagian Pengendali

Diagram alir program pada Arduino Pro Mini bagian pengendali dapat dilihat pada Gambar 3.10. Diagram alir dimulai dengan *start* kemudian dilanjutkan

pembacaan nilai 5 buah nilai sensor, yaitu pin y GY61a, GY61b, dan GY61c, pin x GY61c, serta pin I2C HMC5883L.

Hasil perubahan nilai keluaran sensor tidak tenang, sehingga membutuhkan proses pelembutan nilai sensor, yaitu dengan menjumlahkan 13 data pembacaan terakhir kemudian dibagi 13. Keluaran dari proses ini adalah perubahan keluaran pembacaan sensor lebih tenang.

Servo-servo pada robot lengan hanya dapat bergerak dalam sudut 0° hingga 180° . Oleh karena itu, nilai-nilai sensor yang telah dihaluskan dikonversi menjadi nilai sudut 0° hingga 180° dengan ketentuan seperti pada Gambar 3.11.



Gambar 3. 11 Arah Derajat Kebebasan Pengendali

Gambar horisontal adalah arah nilai sudut untuk HMC5883L. Sementara, gambar vertikal (a) untuk pin x GY61c, dan gambar vertikal (b) untuk pin y GY61a, GY61b, dan GY61c.

Adapun masukan nilai sudut yang tidak didapat dari pembacaan sensor melainkan dari program, yaitu nilai untuk membuka dan menutup penjepit robot. Dimana ketika untuk membuka penjepit, diinisialkan dengan nilai 0 dan untuk menutup penjepit diinisialkan dengan nilai 90.

Untuk mempermudah pengolahan data selanjutnya maka, keluran-keluaran nilai sudut variabelkan dengan nama seperti pada Tabel 3.10.

Tabel 3. 10 Pengubahan Nama Sensor ke dalam Variabel Program

Variabel Lama	Variabel Baru
HMC5883L	Sudut1
Pin y GY61a	Sudut2
Pin y GY61b	Sudut3
Pin y GY61c	Sudut4
Pin y GY61c	Sudut5
Kondisi Penjepit	Sudut6

Setelah dikonversi dan diinisialkan, tahap selanjutnya adalah pelembutan nilai sudut. Pelembutan nilai sudut dilakukan karena rentang nilai sensor yang dikonversi lebih sempit dari rentang sudut konversi sehingga keluaran nilai sudut tidak tenang dan jarak antar perubahan nilai sudut besar. Caranya untuk melembutkan nilai ini sama seperti pelembutan nilai keluaran sensor, yaitu dengan menjumlahkan 13 data pembacaan terakhir kemudian dibagi 13. Keluaran dari proses ini adalah keluaran nilai sudut lebih tenang dan jarak antar perubahan nilai sudut kecil.

Tombol yang digunakan sebagai masukan data berjumlah 2, yaitu tombol kuning dan biru. Arduino Pro Mini membaca kondisi tombol sebagai masukan, dimana ketika tombol *on* maka, kondisi tombol yang terbaca adalah *LOW*, dan ketika tombol *off* maka, kondisi tombol yang terbaca adalah *HIGH*.

Ketika kondisi tombol kuning *LOW* artinya mode *reset*. Mode *reset* adalah mode untuk mengembalikan ke pengaturan awal titik 90° dari keluaran HMC5883L. Mode *reset* diinisialkan dengan angka 1 dan *buzzer* aktif pada detik pertama. Ketika mode *reset* aktif, pengendali akan mem-*publish* pesan “*1,90,220,90,0,90,90#” setiap dua detik sekali.

Ketika tombol kuning *HIGH* dan tombol biru *HIGH* artinya mode *putting*. Mode *putting* maksudnya robot dapat bergerak sesuai gerakan tangan. Mode *putting* diinisialkan dengan angka 2 dan *buzzer* aktif pada detik pertama. Ketika mode *reset* aktif, pengendali akan mem-*publish* pesan dengan nilai berubah-ubah sesuai hasil pengolahan dari masukan gestur tangan. *Publish* pesan pada mode *putting* terjadi setiap 2 detik sekali.

Ketika tombol kuning *HIGH* dan tombol biru *LOW* artinya mode *packing*. Mode *packing* maksudnya robot dapat mengambil dan memindah barang secara otomatis. Mode *packing* diinisialan dengan angka 3 dan *buzzer* aktif pada detik pertama. *Publish* pesan pada mode ini terjadi setiap 4 detik sekali. Pesan yang di-*publish* pengendali pada mode ini berubah otomatis sesuai dengan urutan pada Tabel 3.11.

Tabel 3. 11 Urutan keluaran Pesan Mode Packing

<i>Publish ke</i>	<i>Pesan</i>
1	*3,92,117,36,20,127,90#
2	*3,93,117,35,17,90,90#
3	*3,87,220,90,45,90,90#
4	*3,0,220,90,45,90,90#
5	*3,0,90,90,90,90,90#
>5	*3,0,90,90,90,90,0#

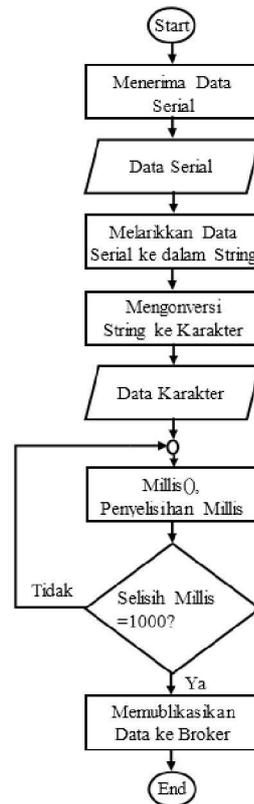
Sebelum data-data pembacaan dikirimkan ke ESP01 melalui komunikasi serial, data-data tersebut dibuat larik terlebih dahulu, yaitu diawali tanda ‘!’, dipisahkan dengan tanda ‘,’ dan diakhiri dengan tanda ‘#’. Format pelarikan data dilihat ada Gambar 3.12. Kemudian, larik-larik tersebut dikirimkan ke ESP01 setiap 0.5 detik sekali melalui komunikasi serial.

! mode , sudut1 , sudut2 , sudut3 , sudut4 , sudut5 , sudut6 #

Gambar 3. 12 Format Pelarikan Data

3.7.1.2 ESP01

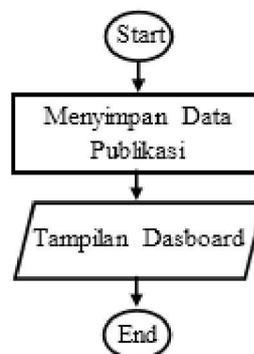
Diagram alir program pada ESP01 dapat dilihat pada Gambar 3.13. Program pada ESP01 diawali dengan menerima data masukan dari Arduino Pro Mini melalui komunikasi serial. Karena pengiriman dilakukan melalui komunikasi serial maka, data larik yang telah diterima harus dibuat larik lagi. Pelarikan ini hanya bisa dilakukan dengan *type data string*. Setelah terlarik, langkah selanjutnya adalah mengubah *type data string* menjadi data karakter. Pengubahan ini harus dilakukan karena MQTT tidak dapat mempublikasi ke broker jika data yang dipublikasi bertipe data *string*.



Gambar 3. 13 Diagram Alir Program pada ESP01 bagian Pengendali

Langkah selanjutnya adalah memanfaatkan fungsi *millis()*. Fungsi *millis()* digunakan untuk menempatkan proses *publish* esan larik pada tiap 2 detik. Sehingga pesan dikirimkan pada Adafuit IO tiap 2 detik sekali.

3.7.2 Broker



Gambar 3. 14 Diagram Alir Kerja Broker

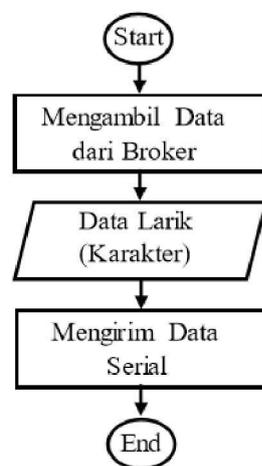
Diagram alir pada broker dapat dilihat pada Gambar 3.14. Dalam MQTT pada perancangan ini, broker hanya dapat menerima pesan, menyimpan pesan, dan menampilkan isi pesan pada *dashboard* pada fasilitas *monitor*-nya.

3.7.3 Robot

Secara umum diagram alir pada robot lengan hanya mempresentasikan data yang dikirim oleh pengendali. Bagian-bagian yang diprogram adalah ESP01 dan Arduino Pro Mini, serta penjelasan singkat mengenai prinsip kerja PCA9685. Berikut penjelasannya:

3.7.3.1 ESP01

Diagram alir ESP01 bagian robot dapat dilihat pada Gambar 3.15. Pada ESP01 bagian robot, ESP01 bertugas mengambil pesan yang ada pada *feed* broker yang telah ditentukan. Pesan yang dapat diambil oleh ESP01 adalah pesan terbaru yang diterima oleh broker waktu pengambilan pesan.

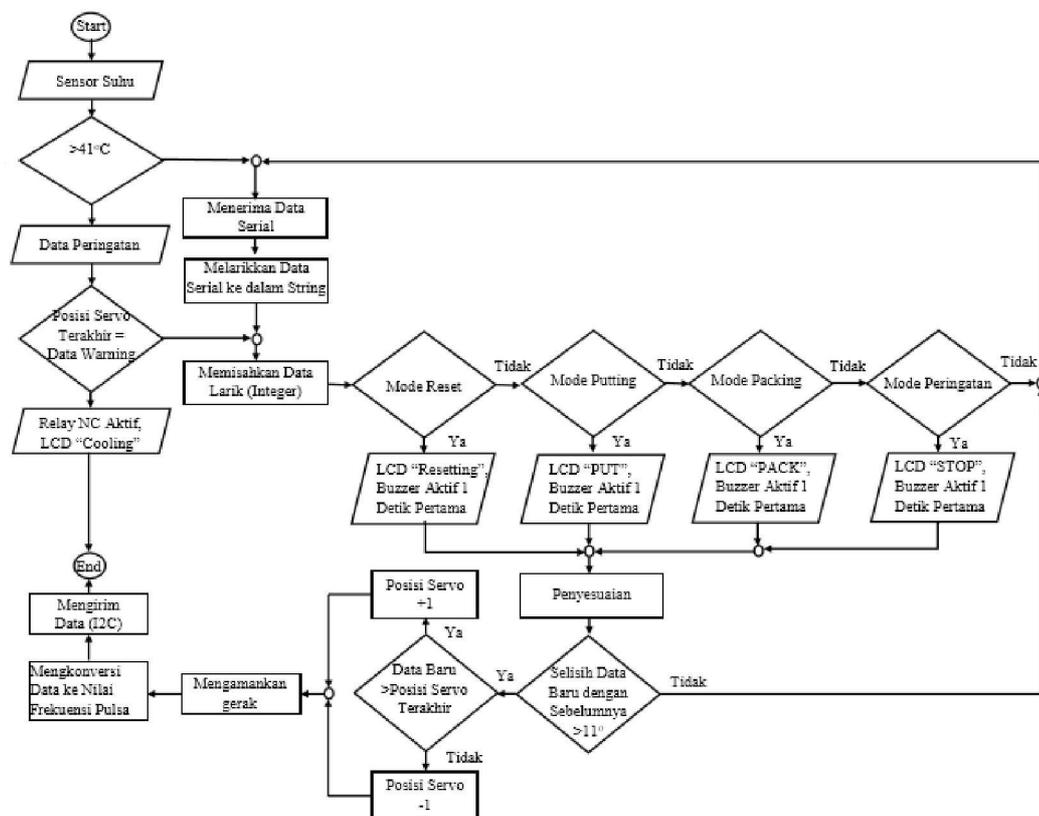


Gambar 3.15 Diagram Alir Program pada ESP01 bagian Robot

Pesan di MQTT dibaca oleh ESP01 sebagai pesan dengan tipe data karakter. Kemudian pesan tersebut dikirimkan ke Arduino Pro Mini melalui komunikasi serial.

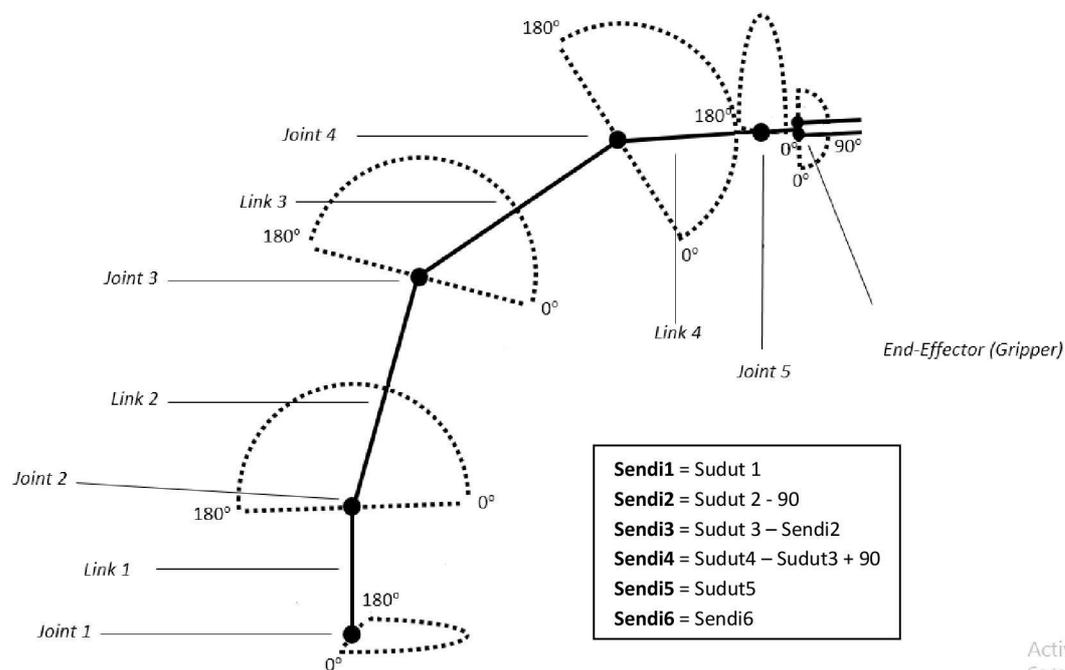
3.7.3.2 Arduino Pro Mini

Diagram alir Arduino Pro Mini bagian robot dapat dilihat pada Gambar 3.16. Diagram alir di sini diawali dengan nilai pembacaan sensor suhu yang terpasang pada sendi 2. Sendi 2 merupakan sendi yang tersusun dari 2 buah servo MG90s sehingga hanya memiliki torsi sebesar 4.4kgcm untuk mengangkat beban serta suhu operasi antara 0°C hingga 55°C. Oleh karena itu, supaya sendi 2 tidak cepat rusak maka, di pasang sensor suhu.



Gambar 3. 16 Penyesuaian Nilai Arah Sendi Robot

Batas suhu maksimal kerja sendi 2 diatur $\frac{3}{4}$ dari 55°C atau sama dengan 41°C. Jika suhu sendi melewati 41°C maka, masukan data robot akan menggerakkan robot dalam posisi siap pendinginan. Pengarahan ini dinamakan dengan “*mode peringatan*” dan LCD akan menampilkan karakter “*STOP*” serta buzzer aktif pada 1 detik pertama.



Gambar 3. 17 Penyesuaian Nilai Arah Sendi Robot

Ketika posisi robot sudah dalam posisi siap pendinginan maka, relay NC akan aktif dan robot akan mati. Saat ini terjadi, LCD akan menampilkan karakter “Colling”. Kondisi ini akan berlangsung hingga sendi 2 bersuhu 28°C dengan memanfaatkan fungsi “for”. Ketika sendi 2 bersuhu 28°C maka, relay NC akan mati, dan robot hidup kembali.

Apabila nilai suhu sendi 2 $< 41^{\circ}\text{C}$ maka, Arduino Pro Mini dapat menerima data larik dari ESP01 melalui komunikasi serial. Karena menggunakan komunikasi serial maka, data harus dibuat larik kembali ke dalam sebuah *string*. Data larik yang telah dibuat menjadi satu *string* selanjutnya dipisahkan (*parsing*) menjadi 7 buah variabel *integer*, yaitu mode, sudut1, sudut2, sudut3, sudut4, sudut5, dan sudut6.

Ketika mode bernilai 1 artinya mode *reset* dan LCD akan menampilkan karakter “Reset”. Ketika mode bernilai 2 artinya mode *putting* dan LCD akan menampilkan karakter “Putting”. Ketika mode bernilai 3 artinya mode *packing* dan LCD akan menampilkan karakter “Packing”.

Nilai-nilai sudut yang telah dipisahkan selanjutnya diolah dengan penyesuaian posisi kebebasan sendi-sendi robot (*link*). Penyesuaian yang dilakukan dapat dilihat pada Gambar 3.17.

Selanjutnya, yaitu membandingkan apakah nilai sudut yang baru $>$ atau $<$ dari posisi servo sebelumnya. Jika $>$ maka, posisi servo sebelumnya tambah 2° . Jika $<$ maka posisi servo sebelumnya dikurang 2° . Penjumlahan dan pengurangan ini dilakukan setiap 15 milidetik sekali. Tujuannya supaya perubahan posisi servo halus.

Tahap selanjutnya adalah keamanan gerak masing-masing sendi. Keamanan ini dilakukan dengan membatasi gerak sendi. Hal ini didasarkan pada kekuatan maksimal torsi pada masing-masing sendi yang berbeda. Untuk lebih jelasnya dapat dilihat pada Tabel 3.12.

Setelah data melalui tahap keamanan selanjutnya masuk ke tahap konversi dari nilai sudut ke nilai periode pulsa servo, yaitu sudut nilai 0° hingga 180° menjadi 150 hingga 600.

Tabel 3. 12 Batas Sudut Gerak Robot

Sendi (Link)	Maksimal ($^\circ$)	Minimal ($^\circ$)
1	18	162
2	68	162
3	18	162
4	18	162
5	18	162
6	18	90

3.7.3.3 PCA9685

Setelah dikonversi, data dikirim ke PCA9685 melalui komunikasi I2C. PCA9685 pun menerima data dan pin-pin PCA9685 yang telah ditentukan membangkitkan pulsa sesuai nilai-nilai periode pulsa yang masuk. Gelombang pulsa ini kemudian dikirimkan ke servo melalui pin CH servo dan servo pun

bergerak sesuai gelombang yang diterimanya. Diagram alir prinsip kerja PCA9685 yang telah dijelaskan di atas dapat dilihat pada Gambar 3. 18.



Gambar 3. 18 Diagram Alir Kerja PCA9685

3.8 Pengujian

Setelah alat jadi, alat masuk ke tahap pengujian. Pengujian dilakukan untuk mengetahui alat sudah bekerja dengan baik atau belum. Pada tahap ini misi alat adalah memindahkan benda dari alas ke kantong. Proses pengujian dibantu dengan busur derajat yang tercetak pada kertas. Guna busur deraja ini adalah untuk membantu peletakan sudut benda dari titik *base* robot dengan tepat. Pengujian ini dilakukan sebanyak tiga kali percobaan dengan masing-masing letak benda di setiap percobaan berbeda. Peletakan benda pada masing-masing percobaan dapat dilihat pada Tabel 3.13.

Tabel 3. 13 Peletakan Benda

Percobaan	Jarak dari Base Robot	Sudut dari Base Robot
I	9.5	5°
II	9.5	90°
III	9.5	135°

3.8.1 Prosedur Pengujian

Berikut prosedur pengujian yang dilakukan ada perancangan ini:

1. Pasang sarung tangan pengendali ke tangan, kemudian Rekatkan box pengendali ke tangan
2. Pasang masing-masing interface pada box ke sensor, yaitu Ab ke As, Bb ke Bs, Cb ke Cs, dan Db ke Ds
3. Letakkan kantong pada sudut 0° dari titik *base* robot dengan jarak 130 mm
4. Adapter 5V/2A dihubungkan ke sumber tegangan 220VAC dan jack DC pada pengendali. Sementara adapter 12V/3A dihubungkan ke sumber tegangan 220VAC dan jack DC pada robot.
5. Benda diletakkan dengan posisi seperti Tabel 3.13
6. MiFi diaktifkan
7. Kunjungi io.adafruit.com/Cloud3_damanicIoT/dashboards/message pada *browser* untuk memonitor waktu dan pesan yang ter-*publish* ke broker
8. Pengendali dan robot diaktifkan dengan cara menekan tombol merah pada sisi keduanya
9. Robot akan mulai hidup pada detik ke-9 setelah tombol merah robot di tekan
10. LED ESP01 pengendali akan berkedip ketika pengendali sudah terhubung ke broker. Ini mengartikan bahwa *publish* data telah dimulai
11. Robot sudah mulai *subscribe* pesan dari broker ditandai dengan hidupnya LED ESP01 pada box robot.
12. Tekan tombol kuning pada pengendali untuk menggerakkan robot ke posisi *reset*.
13. Ketika robot hidup, tekan kembali tombol kuning pengendali, kemudian gerakkan robot dengan gestur tangan dan arahkan *gripper* ke benda
14. Setelah posisi *gripper* siap mengangkat benda, kemudian tekan tombol biru, maka secara otomatis robot akan mengambil, dan memindahkan benda ke kantong.

3.9 Perbaikan

Ketika komponen belum bekerja sesuai tujuan maka, perbaikan dilakukan. Kemudian diuji kembali.