

BAB I

PENDAHULUAN

A. Latar Belakang

Teknologi merupakan salah satu bagian penting dalam kehidupan manusia di berbagai bidang industri maupun dalam kehidupan sehari-hari. Seiring dengan naiknya kebutuhan manusia dan teknologi yang semakin hari semakin canggih, maka dibuatlah suatu sistem yang dapat membantu pekerjaan manusia yang disebut dengan robot.

Lengan robot merupakan jenis lengan mekanik yang kemudian diprogram dengan fungsi menyerupai dengan manusia. Sandi pada lengan robot biasanya digerakkan oleh motor ataupun sejenisnya, lengan robot diklasifikasikan berdasarkan jumlah derajat kebebasannya. Derajat kebebasan merupakan sambungan pada lengan, dapat dibengkokkan, digeser ataupun diputar. Derajat kebebasan digunakan untuk mengetahui cara robot bergerak yang biasanya, jumlah derajat kebebasan sama dengan jumlah sandi yang menggerakkan robot.

(Purwono Prasetyawan Dkk, 2018).

Teknologi zaman sekarang, berdampak pada segala bidang kehidupan. Teknologi sangat membantu dalam perkembangan inovasi yang terus bermunculan, Tak terkecuali dengan banyaknya pembuatan robot mulai dari robot mainan sampai pada robot pelayanan industri, Lengan robot empat dof menggunakan aplikasi smartphone arduino bluetooth controller, Sistem yang lebih

canggih yang dapat mengendalikan robot dari jarak jauh khususnya di bidang elektronika.

Oleh karena itu penelitian ini membahas “**Sistem Kendali Lengan Robot 4 Dof Melalui Push Button Dan Bluetooth (Joystick, Terminal, Voive)**”. Lengan robot empat dof merupakan suatu alat penggabungan mekanik dan elektronika yang dirancang untuk mengendalikan robot dengan menggunakan *push button* dan aplikasi *arduino bluetooth controller* untuk memudahkan pengguna untuk mengontrol pergerakan robot secara wireless, Untuk penelitian ini terfokus pada komunikasi aplikasi *smartphone* dengan robot melalui operasi *wireless* sehingga gerakan robot akan bergerak sesuai dengan gerakan Aplikasi *smartphone* yang telah di program.

B. Rumusan Masalah

Dengan mengacu pada latar belakang diatas maka disusun sebuah rumusan masalah yang akan dibahas yakni: bagaimana mengendalikan lengan robot empat dof menggunakan *Push button* dan Aplikasi *smartphone Arduino bluetooth controller* dengan tiga fitur yaitu, *Joystick, Terminal, dan, Voice*.

C. Tujuan Penelitian

Adapun tujuan penelitian dalam perancangan alat ini yaitu dengan mengembangkan lengan robot, Pengendali *push button* dan aplikasi *arduino bluetooth controller (Joystick, Terminal, Voice)* untuk memudahkan pengguna dalam mengontrol pergerakan robot secara *wireless* dan *push button*.

D. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah:

1. Pergerakan lengan robot dikontrol menggunakan *push button* dan aplikasi *Arduino bluetooth controller (joystick, terminal, voice)*.
2. Derajat pergerakan servo dan mode control di tampilkan di LCD

E. Manfaat Penelitian

Adapun manfaat penulisan ini adalah dengan memberikan pengetahuan bagaimana sebuah sistem bekerja dalam proses kendali mengatur pergerakan lengan robot dengan menggunakan *push button* dan aplikasi *Arduino bluetooth controller (joystic, terminal, voice)*, serta dapat menjadi bahan referensi dalam pengembangan mahasiswa dalam bidang robotika.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

Untuk membuat rancangan sistem kendali lengan robot empat *dof* melalui *push button* dan *bluetooth (joystick, terminal, voice)*, sebelumnya dapat dijelaskan tinjauan pustaka dan sistem yang terkait didalamnya yang dapat dijelaskan sebagai berikut:

1. Definisi Robot

Robot pertama kali muncul pada tahun 1921 yang berasal dari bahasa Ceko “ROBOTA” yang berarti *Forced Labor* dari kata “*Robotics*” berasal dari sebuah karya fiksi ilmiah karangan Issac Asimov pada tahun 1942 dengan judul “*Runaround*”. Cerita tersebut kemudian dimasukkan oleh Asimov kedalam buku karangannya yang terkenal “*L Robot*” (Siswaja).

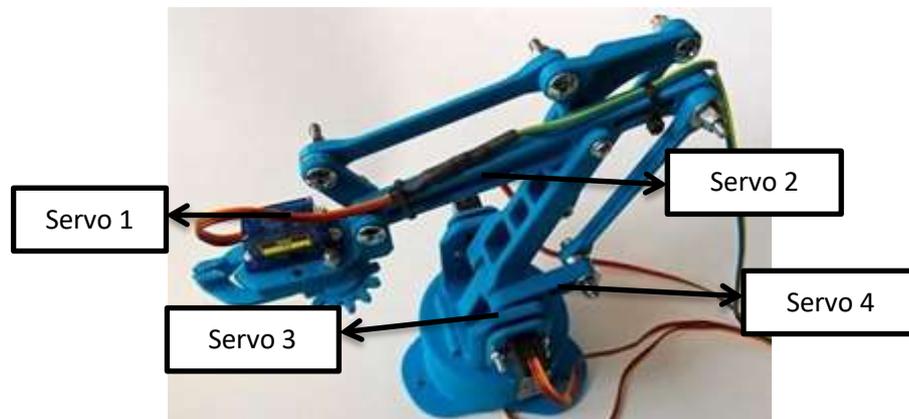
Nainggolan, dkk (2015), Robot adalah sebuah hasil rekayasa teknologi yang dirancang untuk mempermudah pekerjaan manusia. Robot dirancang dari perangkat mekanik yang dapat melakukan tugas fisik, dibawah kendali dan pengawasan kontrol manusia ataupun menggunakan program yang telah di defenisikan sebagai mesin dengan kecerdasan komputer dan dikontrol oleh komputer, dan memiliki kemampuan fisik seperti manusia.

Pada umumnya robot merupakan rangkaian elektromagnetik yang dapat bergerak untuk mencapai suatu titik tertentu melalui empat gerakan yaitu bergerak maju mundur, kiri kanan, atas bawah, dan membuka menutup capit. Secara umum, robot memiliki sifat-sifat atau karakteristik sebagai berikut:

- a. Sebuah robot alami dan merupakan hasil rekaan
- b. Dapat merasakan kondisi lingkungan
- c. Mampu memanipulasi benda yang berada di lingkungan
- d. Mempunyai tingkat kecerdasan tertentu yang dapat bergerak secara terkontrol dan secara otomatis.
- e. Dapat diprogram
- f. Dapat membuat pergerakan yang terkoordinasi dengan baik.

2. Lengan Robot

lengan robot adalah robot mekanik yang gerakannya dibuat menyerupai pergerakan lengan manusia. Lengan robot memiliki beberapa sendi pergerakan yang disesuaikan dengan kebutuhan lengan robot, juga memiliki komponen sebagai bagian pembentukannya seperti *aktuator* dan *controller*. Lengan robot memiliki derajat kebebasan yang berfungsi untuk menentukan banyaknya gerakan pada robot. (Bagus Utomo, dkk 2020).



Gambar 2. 1 Lengan robot

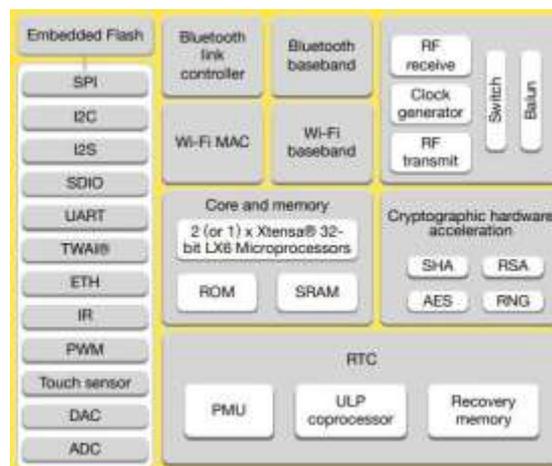
Pada sistem mekanik lengan robot setidaknya terdiri dari 4 *servo* yaitu:

1. *Base servo* 1 bagian atas memutar pergerakan kekanan atau kekiri
2. *Shoulder servo* 2 bagian lengan yang bergerak keatas dan kebawah
3. *Wrist servo* 3 bagian pergelangan tangan bergerak kedepan dan kebelakang
4. *Gripper servo* 4 bagian servo yang berfungsi untuk menjepit benda kerja

3. Mikrokontroler

Syah Alam, (2021). ESP32 adalah *Mikrokontroler System on Chip (SoC)* berbiaya rendah dari *Espressif Systems*, yang juga sebagai pengembang dari *SoC ESP8266* yang terkenal dengan *Node MCU*. *ESP32* adalah penerus *SoC ESP8266* dengan menggunakan *Mikroprosesor Xtensa LX6 32-bit Tensilica* dengan *Wi-Fi dan Bluetooth* yang terintegrasi.

Hal yang baik tentang *ESP32*, seperti *ESP8266* adalah komponen *RF* terintegrasi seperti *Power Amplifier, Low-Noise Receive Amplifier, Antena Switch, dan Filter*. Hal ini membuat perancangan *hardware* pada *ESP32* menjadi sangat mudah karena hanya memerlukan sedikit komponen *eksternal*.



Gambar 2.2 Sistem *ESP32*
(sumber: Syah Alam,2021)

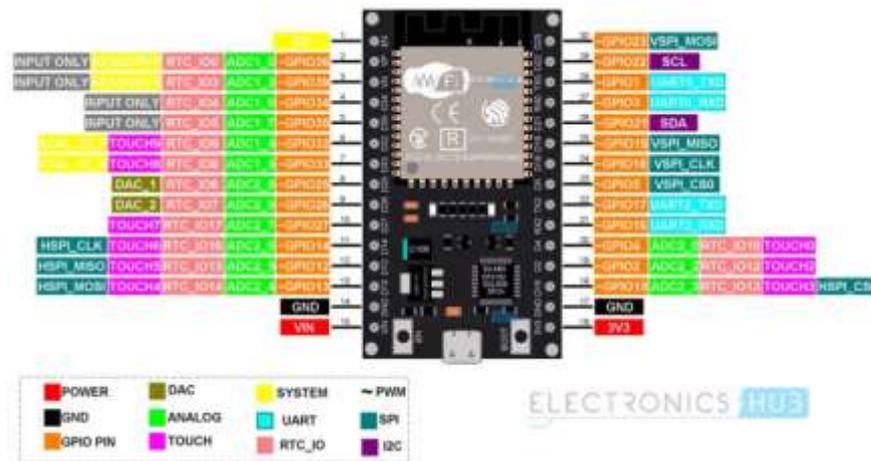
Hal penting yang perlu diketahui tentang *ESP32* adalah diproduksi menggunakan teknologi 40 nm *ultra-low-power TSMC*. Jadi, dapat dioperasikan dengan baterai yang umum seperti yang sudah digunakan pada perangkat perlengkapan audio, *monitoring, smartwatch, dll*.

ESP32 memiliki lebih banyak fitur daripada *ESP8266*. Memulai dengan *ESP32* ini. Berikut ini daftar beberapa spesifikasi penting dari *ESP32*. Tetapi untuk spesifikasi lengkap, dapat melihat pada Data sheet.

- *Single or Dual-Core 32-bit LX6 Microprocessor with clock frequency up to 240 MHz.*
- *520 KB of SRAM, 448 KB of ROM and 16 KB of RTC SRAM.*
- *Supports 802.11 b/g/n Wi-Fi connectivity with speeds up to 150 Mbps.*
- *Support for both Classic Bluetooth v4.2 and BLE specifications.*
- *34 Programmable GPIOs.*
- *Up to 18 channels of 12-bit SAR ADC and 2 channels of 8-bit DAC*
- *Serial Connectivity include 4 x SPI, 2 x I²C, 2 x I²S, 3 x UART.*
- *Ethernet MAC for physical LAN Communication (requires external PHY).*
- *1 Host controller for SD/SDIO/MMC and 1 Slave controller for SDIO/SPI.*
- *Motor PWM and up to 16-channels of LED PWM.*
- *Secure Boot and Flash Encryption.*
- *Cryptographic Hardware Acceleration for AES, Hash (SHA-2), RSA, ECC and RNG.*

ESP32 memiliki total 48 pin yang multi fungsi. Penggunaan pin berbeda-beda tergantung fungsinya. Detail pin dapat dilihat pada gambar disamping.

Keunggulan *ESP32* adalah memiliki banyak pin yang dapat berfungsi sebagai analog atau digital sesuai dengan konfigurasi.



Gambar 2.3 pinout ESP32 board
(sumber: Syah Alam,2021)

4. Push button

Push button adalah satu komponen elektronika yang dapat memutus dan mengalirkan arus listrik dalam suatu rangkaian *project Arduino*, Dimana pemutusan dan pengaliran ini terjadi karena prinsip pengalihan dari satu konduktor ke konduktor lain. Caranya dengan pengoperasian langsung secara manual oleh pengguna.

Seperti yang dijelaskan pada poin sebelumnya, bahwa fungsi *push button* adalah untuk memutus dan menyambungkan arus Listrik, Biasanya *push button* ini digunakan untuk memicu jalannya suatu perangkat output seperti *relay*, *buzzer*, *LED*, maupun yang lainnya, Bahkan kerennya lagi *push button* juga dapat dipadukan dengan *software Matlab* maupun *database website*.



Gambar 2.4 *push button*
(*sumber: Aldy Razor. 2020*)

Pada dasarnya, prinsip kerja *push button* adalah pemutus dan penyambung aliran listrik, Namun dalam hal ini ia tak bersifat mengunci Jadi ia akan kembali ke posisi semua saat selesai ditekan. Saat *push button* ditekan, ia menjadi bernilai *HIGH* dan akan menghantarkan arus listrik. Sedangkan apabila dilepas, maka ia bernilai *LOW* dan memutus arus listrik. Namun cara kerja saklar *push button* kadang berbeda tergantung dari jenisnya. (Aldy Razor. 2020)

5. Aplikasi Arduino Bluetooth Controller

(Apps valley.2019). Tentang Pengontrol *Arduino Bluetooth Controller*, dikembangkan oleh *My Valley Apps* dan versi terbaru *Arduino Bluetooth Controller 5.4* telah diperbarui pada 15 Mei 2023. *Arduino Bluetooth Controller* termasuk dalam kategori Alat. Anda dapat memeriksa semua aplikasi dari pengembang *Arduino Bluetooth Controller* dan menemukan 72 aplikasi alternatif untuk *Arduino Bluetooth Controller* di Android. Saat ini aplikasi ini gratis. Aplikasi ini dapat diunduh di Android 4.4+ di *APKFab* atau *Google Play*. Semua file *APK/XAPK* di *APKFab.com* asli dan 100% aman dengan unduhan cepat.

Betapa kerennya mengendalikan perangkat elektronik dengan berbagai cara menggunakan Aplikasi Arduino bluetooth *Controller*, Aplikasi Pengontrol *Bluetooth Arduino* memungkinkan untuk mengontrol perangkat secara *wireless* dengan dengan modul bluetooth dan papan *arduino*.

Perangkat Android menjadi remote control untuk pengontrol *mikro* apa pundenan *modul bluetooth*.

Dapat menggunakan kontrol *Bluetooth Arduino* Semua dalam satu untuk:

- Sistem Otomasi Rumah Pintar
- Sistem perintah suara
- Pengontrolan kendaraan
- Kendali cahaya
- Kendali Led
- Dan lain sebagainya

1. Fitur utama *arduino bluetooth controller*

- 1) Pengontrol *REMOTE* untuk mengontrol kendaraan seperti mobil, motor dan perangkat sejenis lainnya
- 2) *DIMMER* digunakan untuk tinggi dan rendahnya kecerahan *led* atau kecepatan perangkat lainnya.
- 3) *TERMINAL* untuk mengirim perintah menggunakan *keyboard smartphone*.
- 4) Tombol *ON / OFF* untuk menuji perangkat berpungsi dengan normal.
- 5) *VOICE* Controller untuk memudahkan dalam pengontrolan melalui suara.

- 6) TIMER untuk mengatur durasi *on / off* dengan penghitungan mundur.



Gambar 2.5 Logo Aplikasi Arduino bluetooth controller
(sumber: Apps valley.2019)

6. Motor Servo

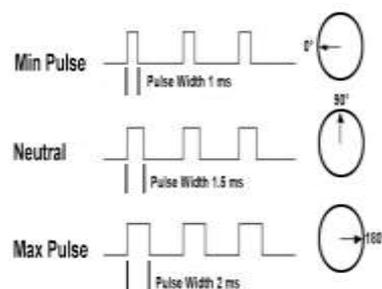
Motor *Servo* merupakan sebuah perangkat atau actuator putar (motor) yang dirancang dengan sistem kontrol umpan balik loop tertutup sehingga dapat di atur untuk menentukan posisi sudut dari poros *output* yang dimana posisi rotornya akan di informasikan kembali ke rangkaian kontrol yang ada di dalam motor *servo*. Motor ini terdiri dari motor DC, rangkaian gear, serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor *servo*. Sedangkan potensiometer, dan rangkaian control, Potensiometer berfungsi untuk menentukan batas sudut dari putaran *servo* dan sudut dari sumbu motor *servo* diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor *servo*. Prinsip kerja motor *servo* dikendalikan dengan memberikan *pulse width modulation* melalui kabel control, dengan durasi yang diberikan akan menentukan posisi sudut putaran dari poros motor *servo*. Poros motor *servo* akan bergerak dan bertahan di posisi yang telah diperintahkan ketika durasi telah diberikan (NewbloggerThemes.2020).

Tabel 2. 1 Spesifikasi Motor *Servo mg996R*

No.	Spesifikasi	
1	Dimensi	40,7 X 19,7 X42,9 mm
2	Tegangan Kerja	4,8 VDC
3	Berat	55 g
4	Kecepatan Reaksi	0,17 s/60 derajat
5	<i>Stall Torque</i>	9,4 kg/cm
6	Suhu Kerja	0-55 C
7	<i>Dead Band Width</i>	5 us
8	Stall current	2,5 A (6V)
9	<i>Running current</i>	500 mA-900 Ma

**Gambar 2.6** Motor servo
(sumber: NewbloggerThemes.2020)

Untuk mempertahankan posisinya, rangkaian control melakukan sinyal PWM (*Pulse Width Modulation*), Lebar sinyal di atur antara 1ms hingga 2ms (mili detik), motor akan berputar dari titik 0 hingga maksimal 180° atau 360° tergantung tipenya jika diberikan sinyal pada rentang waktu tersebut, sinyal pwm ini

**Gambar 2.7** pulsa kendali motor servo
(sumber: NewbloggerThemes.2020)

Motor servo akan bekerja dengan baik jika diberikan sinyal pwm dengan frekuensi 50 HZ pada pin kontrolnya, jika sinyal dengan frekuensi 50 HZ di capai pada kondisi ton *duty cycle* dari sinyal yang di berikan kurang dari 1,5ms maka motor akan berputar berlawanan arah jarum jam atau membentuk sudut yang besarnya linear terhadap besarnya ton *duty cycle* dan akan bertahan pada posisi tersebut.

7. Modul *Stepdown LM2595*

Module Regulator LM 2596 adalah rangkaian modul konverter DC / DC dengan frekuensi tetap 150 kHz fixed-voltage (PWM step-down) menggunakan IC Regulator LM2596, yang mampu menggerakkan beban 5A dengan efisiensi tinggi, derek rendah dan regulasi garis dan beban yang sangat baik. Membutuhkan jumlah minimum komponen eksternal, regulator mudah digunakan dan termasuk kompensasi frekuensi internal dan osilator frekuensi tetap.

Modul regulator LM2596 dapat bekerja dengan suplai tegangan 4V-32V dan suhu operasinya -40 - +85 degrees. Pada *module regulator* LM2596 menggunakan ic SMD (*Surface Mount Device*) dan terdapat sebuah *potensio* untuk mengatur tegangan masukannya dari 4V – 24V DC pada frekuensi kerja 150 kHz sehingga dapat disesuaikan dengan kebutuhan tegangan.(sejati, 2019)



Gambar 2.8 Stepdown LM2596

(sumber: <https://ecadio.com/jual-modul-step-down-dc-lm2596>)

8. Baterai

Battery adalah perangkat yang mengandung sel listrik yang dapat menyimpan energy yang dapat di konversi menjadi daya. *Battery* menghasilkan listrik melalui proses kimia. *Battery* atau akkumulator adalah sebuah sel listrik dimana di dalamnya berlangsung proses elektrokimia yang *reversible* (dapat berkebalikan) dengan efisiennya yang tinggi. Yang dimaksud dengan reaksi kimia *reversible* adalah di dalam *battery* dapat berlangsung proses perubahan kimia menjadi tenaga listrik (proses pengosongan) dan sebaliknya dari tenaga listrik menjadi tenaga kimia (proses pengisian) dengan cara proses regenerasi dari eltroda-elektroda yang dipakai yaitu, dengan melewati arus listrik dalam arah polaritas yang berlawanan di dalam sel.

Battery terdiri dari dua jenis yaitu, *battery primer* dan *battery sekunder*. *Battery primer* merupakan *battery* yang hanya dapat di pergunakan sekali pemakaian saja dan tidak dapat di isi ulang. Hal ini terjadi karena reaksi kimia material aktifnya tidak dapat di kembalikan. Sedangkan *battery sekunder* dapat di isi ulang, karena material aktifnya di dalam dapat di putar kembali. Kelebihan dari pada *battery sekunder* adalah harganya lebih efisien untuk penggunaan jangka waktu yang panjang. (prayogo, 2016)

9. LCD 20x4 I2C

LCD (Liquid Crystal Display) adalah perangkat yang berfungsi sebagai media penampil dengan memanfaatkan kristal cair sebagai objek penampil utama. *LCD* tentunya sudah sangat banyak digunakan untuk berbagai macam keperluan seperti media elektronik televisi, kalkulator, atau layar komputer sekalipun.



Gambar 2.9 LCD 20x4 I2C

(sumber: <https://shopee.co.id/LCD-20X4-BLUE-WHITE-20-X-4-I2C-i.19286514.6167375949>)

LCD yang digunakan adalah LCD berukuran 20x4 karakter dengan tambahan *chip module I2C* untuk mempermudah *programmer* nantinya dalam mengakses LCD tersebut. Sebab dengan digunakannya modul *I2C* akan lebih memperhemat penggunaan pin *arduino* yang akan digunakan, contohnya saja dengan menggunakan modul *I2C* maka hanya diperlukan 4 buah pin *arduino*, yaitu pin *SCL*, pin *SDA*, pin *VCC* dan pin *GND*.(Zamrodah, 2016)

Tabel 2.2 Konfigurasi pin LCD 20x4

Pin	Symbol	Details
1	GND	Ground
2	Vcc	Supply Voltage +5V
3	Vo	Contrast adjustment
4	RS	0-> Control input, 1-> Data input
5	R/W	Read/Write
6	E	Enable
7 to 14	D0 to D7	Data
15	VB1	Backlight +5V
16	VB2	Backlight ground

B. Kajian Hasil Penelitian Terkait

Adapun penelitian terdahulu yang menjadi salah satu acuan penulis dalam melakukan penelitian, sehingga penulis dapat memperbanyak teori yang digunakan dalam mengkaji penelitian yang dilakukan. Penulis sengaja mengangkat beberapa penelitian sebelumnya untuk dijadikan sebagai referensi dalam memperkaya bahan kajian pada penelitian ini.

Berikut ini merupakan penelitian terdahulu berupa jurnal yang terkait dengan penelitian yang dilakukan oleh penulis dalam penelitian yang pertama dilakukan oleh Abdul Muid pada penelitian rancang bangun lengan robot sebagai alat pemindah barang berdasarkan warna menggunakan sensor *fotodiode* dengan kajian hasil penelitian yaitu membuat sebuah lengan robot yang dapat memindahkan barang berdasarkan warna dengan mendeteksi keberadaan benda lalu mengambilnya kemudian menyortir warna benda yang diambil tersebut untuk dimasukkan kedalam wadah yang sesuai. (Abdul Muid. 2015).

Penelitian yang kedua dilakukan oleh Anak Agung Gde Ekayana dan Gusti Ngurah Kade Ary P, pada judul penelitian rancang bangun *prototype* sistem kendali lengan robot menggunakan *interface wireless 2.4 Ghz* dengan kajian hasil penelitian yaitu mewujudkan model *prototype* sistem kendali lengan robot yang dapat dipergunakan sebagai media pembelajaran dalam mata kuliah robotika dengan menggunakan lengan robot. (Anak Agung Gde Ekayana.2017).

Penelitian yang ketiga dilakukan oleh Syukranullah pada judul penelitian Rancang Bangun Robot lengan Berbasis *Microcontroller Arduino Uno*, dengan kajian hasil penelitian yaitu Menghasilkan desain rancang bangun lengan robot

pemindah barang berbasis arduino uno dengan skala kecil dan menganalisa kemampuan aktuator terhadap kinerja robot yang menggunakan motor servo sebagai penggerak lengan robot. (Syukranullah.2019).

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Metode penelitiannya menggunakan metode eksperimen dengan menguji fungsi rangkaian alat apakah sesuai yang diharapkan serta mengacu pada studi pustaka terdahulu yang telah dikumpulkan. Pada penelitian ini penulis merancang sebuah alat yang terfokus pada koneksi *push button* dan aplikasi *arduino Bluetooth controller* untuk pengontrol lengan robot empat *dof*.

B. Lokasi dan Waktu

Lokasi dan Waktu

Lokasi penelitian ini dilakukan di laboratorium teknik elektro Universitas Muhammadiyah Parepare. Untuk waktu penelitian dapat dilihat pada table dibawah ini:

Tabel 3. 1 Waktu penelitian

No	Kegiatan	Waktu kegiatan											
		Bulan ke-1				Bulan ke-2				Bulan ke-3			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literatul	■	■										
2	Perancangan desain alat			■	■								
3	Instalasi komponen utama					■							
4	Pembuatan kerangka alat						■	■					
5	Uji lapangan								■				
6	Evaluasi dan analisi hasil uji									■			
7	Pembuatan laporan hasil penelitian										■	■	■

C. Alat dan Bahan

Alat dan bahan yang digunakan dalam penelitian ini yaitu:

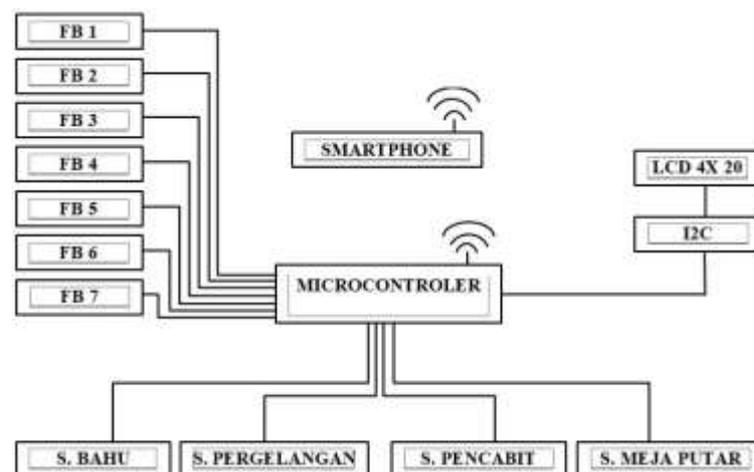
Tabel 3. 2 Alat dan bahan beserta fungsinya

No	Alat dan Bahan	Jumlah
1	<i>Mikrokontroler ESP32</i>	1
2	<i>Push button</i>	7
3	<i>Aplikasi arduino bluetooth kontroller</i>	1
4	<i>Motor servo MG90S</i>	1
5	<i>Motor servo MG996R</i>	3
6	<i>Baterai VTC6 3000mah</i>	2
7	<i>Step Down LM2596</i>	1
8	<i>LCD I2C 20x4</i>	1

D. Rancangan Alat Penelitian

1. Perancangan elektronik

Pada bagian ini akan dibahas suatu pernyataan rangkaian alat yang akan dirancang. Berikut blok diagram dari prerancangan system kendali lengan robot empat *dof* melalui *push button* dan aplikasi *arduino bluetooth controller*



Gambar 3. 1 Blok diagram

Blok diagram sistem keseluruhan dari alat ini terdiri dari tiga bagian yang meliputi: *Push button*, aplikasi *arduino bluetooth kontroller*, *mikrokontroler*, pergerakan lengan robot. Penjelasan pada bagian blok diagram sebagai berikut:

A. *Fush Button*

Pada bagian ini terdapat tujuh buah tombol yang dapat di tekan kemudian mengirim perintah atau sinyal ke mikrokontroler,

- 1) *Push Button* 1 sebagai bahu berpungsi untuk memaju mundur lengan robot.
- 2) *Push Button* 2 Sebagai pergelangan berpungsi untuk menaik turunkan lengan robot.
- 3) *Push Button* 3 Sebagai pencabit berpungsi untuk membuka menutup gengaman lengan robot.
- 4) *Push button* 4 Sebagai meja putar berpungsi untuk menggerakkan robot kekiri dan kekanan.
- 5) *Push button* 5 Sebagai tombol opsi timbal balik dari *fush button* 1 sampai 4.
- 6) *Push Button* 6 Sebagai tombol mode control *fush button*, *terminal*, *joystick*, *voice*.
- 7) *Push Button* 7 Sebagai tombol *on/off*.

B. Aplikasi Arduino bluetooth Kontroller



Pada bagian ini terdapat tiga fitur yang dapat mengontrol setiap *servo*, dapat mengontrol menggunakan *Joystick*, Terminal, dan, *Voice*. Ketiga fitur ini terdapat dalam satu aplikasi yang dapat di gunakan pada smartphone sebagai pegendali jarak jauh.

1. Joystick

Pada fitur ini terdapat dua fungsi yang masing-masing memiliki 10 buah tombol yang dapat digunakan untuk memberi perintah pergerakan ke atas ke bawah, kanan ke kiri, maju dan mundur.



2. Terminal

Pada fitur ini terdapat kolom *enter command here* bertujuan untuk mengganti mode kendali *push button*, *joystick* dan *voice* dengan cara memasukkan kode yang telah ditentukan.



3. Voice

Pada fitur ini terdapat *icon microphone* yang dapat di tekan untuk memberikan perintah melalui suara yang sesuai data yang telah di program.



A. Pengolah data / mikrokontroler

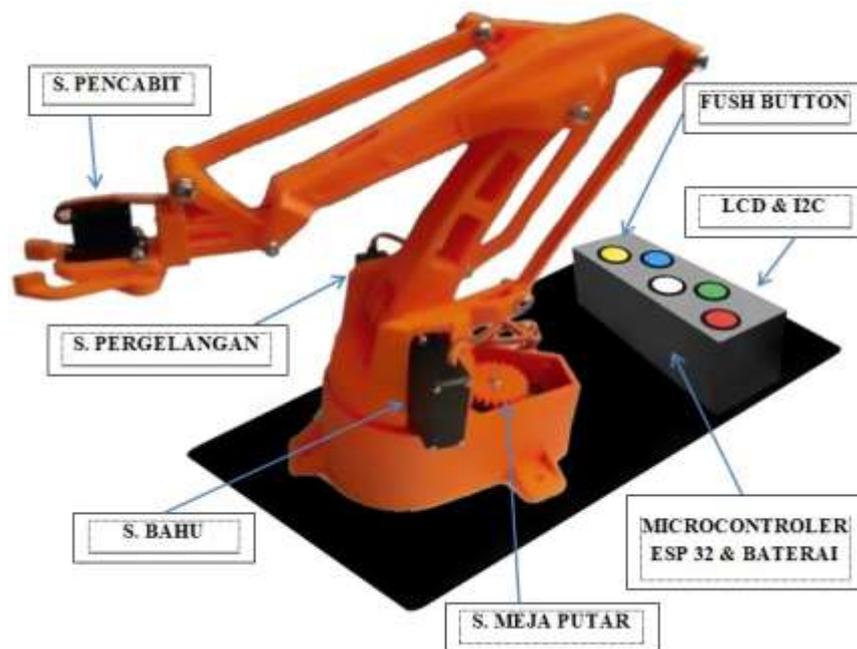
Pada pengolahan data terdapat sinyal dari aplikasi yang sudah diterima kemudian akan di proses ke mikrokontroler yaitu *ESP32* untuk mengambil keputusan sehingga setiap *variable* yang ada pada sistem akan di proses berdasarkan perintah pada program.

B. Pergerakan lengan robot

Pada bagian ini terdiri dari 4 motor *servo*, base motor *servo* yang pertama pada bagian alas berfungsi untuk memutar pergerakan lengan robot secara horizontal atau memutar pergerakan lengan robot kekanan dan kekiri, shoulder *servo* yang kedua berfungsi untuk membantu menggerakkan lengan robot secara vertikal atau pergerakan lengan robot keatas dan kebawah, wrist *servo* yang ketiga berfungsi sebagai pergelangan tangan untuk bergerak keatas dan kebawah dan gripper *servo* 4 berfungsi untuk menjepit benda kerja pada lengan robot.

C. Perancangan mekanik

Pada perancangan alat ini menggunakan bahan dari akrilik pada rangka robot dengan ukuran panjang 20 cm, lebar 26 cm. Terdapat beberapa komponen yang akan digunakan pada perancangan mekanik yaitu lengan robot yang memiliki empat motor *servo*, *mikrokontroler*, baterai.



Gambar 3. 2 Perancangan mekanik

E. Teknik pengumpulan data

Penelitian ini dilakukan beberapa tahapan antara lain:

1. Studi literature

Pengumpulan data serta informasi yang terdapat pada beberapa sumber referensi atau teori yang menjadi landasan teori yang mengenai perencanaan alat. Perencanaan alat disini yaitu gambaran dari lengan robot yang di kendalikan oleh fush button dan aplikasi.

2. Perancangan dan Realisasi

Pada tahapan ini dilakukan proses perancangan dan realisasi alat. langkah pertama yang dilakukan yaitu membuat suatu rancangan perangkat keras sebagai acuan dalam membuat alat yang diinginkan. Langkah pertama yang dilakukan yaitu membuat desain mekanik seperti lengan robot yang terdiri dari empat buah

servo, serta pemasangan komponen seperti pemasangan mikrokontroler dan penghubungan pin-pin. Kemudian yang dilakukan pada langkah kedua yaitu membuat rancangan perangkat lunak berupa program yang menggunakan aplikasi *arduino IDE* sebelum di *input* kedalam mikrokontroler.

3. Pengujian

Setelah melalui tahap perancangan mekanik, selanjutnya dilakukan pengujian. pengujian tahap pertama dimulai dari pengujian kelayakan pada masing-masing komponen seperti kelayakan pakai pada *ESP32*, motor *servo*, fush button. Pengujian tahap dua yaitu pengujian yang dilakukan sistem secara keseluruhan untuk mengetahui apakah semua komponen yang digunakan dapat berfungsi secara normal.

BAB IV

HASIL DAN PEMBAHASAN

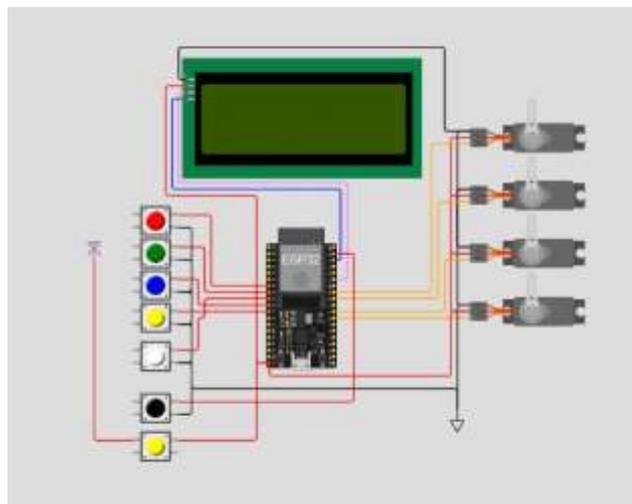
A. Rancangan Sistem

Pada perancangan sistem terbagi atas dua bagian yaitu perancangan perangkat keras (*Hardware*) dan perangkat lunak (*Software*) dari sistem yang di gunakan dan untuk lebih jelasnya dapat dilihat pada gambar di bawah ini:

1. Rangkaian *Hardware*

a. Rangkaian penerima (RX)

Pada rangkaian penerima atau RX (*resiver*) menggunakan *bluetooth* dari *ESP32* yang berfungsi menerima data dari aplikasi *bluetooth controller*. Pin-pin yang terhubung ke arduino yaitu pin CLK terhubung ke pin 2 arduino, pin CS terhubung ke pin 3 arduino, pin CMD terhubung ke pi 4 arduino, dan pin DAT terhubung ke pin 5 arduino.



Gambar 4. 1 Rangkaian penerima data

Komponen perangkat keras yang digunakan terdiri dari:

- 1) *ESP32* berfungsi sebagai pengendali utama dalam menerima dan memberi perintah pada setiap servo,
- 2) *Servo 1* berfungsi sebagai penggerak lengan robot ke kanan dan ke kiri,
Servo 2 berfungsi sebagai penggerak lengan robot ke depan dan kebelakang,
Servo 3 berfungsi sebagai penggerak lengan robot ke atas dan ke bawah,
Servo 4 berfungsi sebagai penggerak lengan robot membuka dan menutup
- 3) *Push button 1* berfungsi untuk menggerakkan *servo 1*,
Push button 2 berfungsi untuk menggerakkan *servo 2*,
Push button 3 berfungsi untuk menggerakkan *servo 3*,
Push button 4 berfungsi untuk menggerakkan *servo 4*,
Push button 5 berfungsi sebagai rotasi atau pengubah arah sebaliknya dari *servo 1* sampai 4
Push button 6 berfungsi sebagai mode ganti kontrol dari push button, joystick, dan voice
Push button 7 berfungsi sebagai *on off*.
- 4) *LCD* berfungsi untuk menampilkan sudut derajat pergerakan *servo* dan mode kontrol

2. Perangkat lunak (*Software*)

a. Rangkain pengirim (TX)

Pada rangkaian pengirim atau TX (*transmitter*) menggunakan aplikasi arduino bluetooth controller versi 6.1.0. Aplikasi *arduino bluetooth controller* menggunakan teknologi *wireless* untuk koneksi ke rangkaian penerima atau RX (*resiver*) yang ada pada robot. Adapun spesifikasi dari aplikasi *arduino bluetooth controller* yaitu, Memiliki beberapa pitur pengontrol yaitu, *joystick controller*, *voice conroller*, *terminal controller*, *buttons controller*, *timer controller*, *dimmer controller*, Koneksi *wireless* dengan jarak 10 m,



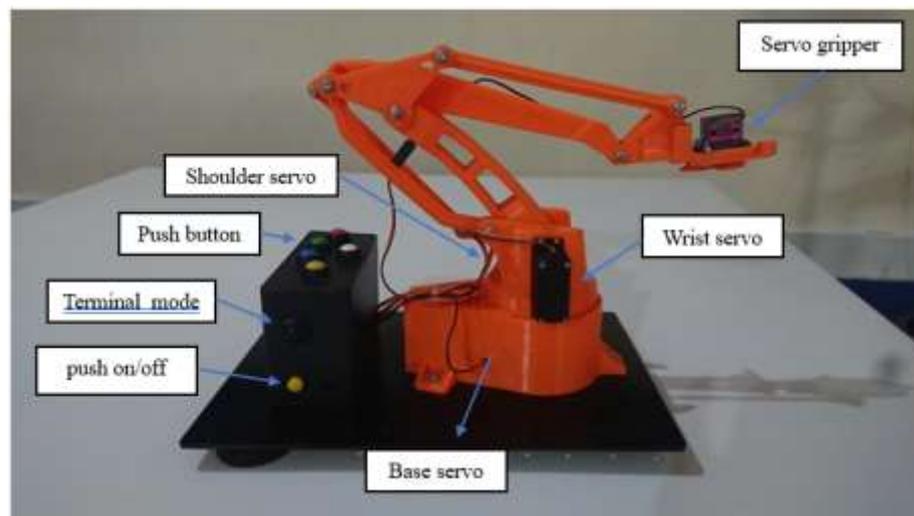
Gambar 4. 2 Logo aplikasi *Bluetooth conroller*

B. Rancangan Mekanik

Perancangan ini dapat kita lihat dengan dengan tinggi satu 20 cm dan luas 20 x 26 cm dengan menggunakan rangka yang bahannya dari besi, akrilit, karet, dan terdapat beberapa komponen di dalamnya. Terdapat dua bagian perancangan mekanik yaitu:

1. Bagian luar

Rancangan mekanik pada bagian luar yang akan diteliti dapat dilihat pada gambar perancangan mekanik dibawah ini.



Gambar 4. 3 Bagian luar rancangan mekanik

Pada bagian rangkaian ini terdapat *servo gripper* yang berfungsi untuk menggerakkan capit pada lengan robot, *wrist servo* berfungsi untuk menggerakkan lengan robot ke atas dan ke bawah, *shoulder servo* berfungsi untuk menggerakkan pergelangan ke depan dan ke belakang pada lengan robot, *base servo* berfungsi untuk menggerakkan pergelangan ke kiri dan ke kanan.

2. Bagian dalam

Rancangan mekanik pada bagian dalam yang akan diteliti dapat dilihat pada gambar perancangan mekanik dibawah ini.



Gambar 4. 4 Bagian dalam perancangan mekanik

Pada rangkaian ini terdapat komponen yang memiliki fungsi pada alat seperti:

Base servo yang berfungsi sebagai penggerak lengan robot ke kanan dan ke kiri, *mikrokontroler ESP32* berfungsi sebagai mengatur atau memproses perintah untuk semua komponen, *step down* berfungsi untuk mengatur tegangan yang akan masuk ke komponen, *LCD & I2C* berfungsi untuk menampilkan mode control dan sudut derajat *servo*, *baterai* berfungsi sebagai *power supply 5v*.

C. Perancangan Perangkat Lunak

Dalam tugas akhir ini menggunakan aplikasi *IDE Arduino* yang mengontrol bagian perangkat lunak ini adalah merupakan bagian pembuatan program yang akan dimasukkan ke *mikrokontroler* yaitu berupa perintah untuk mengolah data masukan untuk diaplikasikan. Di sini akan digunakan *software* pendukung.

1. Uraian program

Adapun *softwer* yang digunakan dalam membuat *sketch* pemrograman dan berguna untuk mengedit, membuat, meng-*upload* ke *board*, dan meng-*coding* program tertentu.

a. Menentukan *library* yang akan dipakai.

```
#include <Servo.h>
#include <BluetoothSerial.h>
#if !defined(CONFIG_BT_ENABLED) ||
!defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig`
to and enable it
#endif
#if !defined(CONFIG_BT_SPP_ENABLED)
#error Serial Bluetooth not available or not enabled. It is
only available for the ESP32 chip.
#endif

BluetoothSerial SerialBT;
String bluetooth_name = "ESP32-BT-Slave";
String dataBTH = "";

#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4);
Servo myservo1;
Servo myservo2;
Servo myservo3;
Servo myservo4;
const int SV[]={16,17,19,18}; //pin Servo
const int Btn[] = {27,26,33,32}; //pin tombol
const int SW=23, Rotate=25;

byte sudut[]={90,90,45,90};
byte CsrX[]={5, 15, 5, 15};
byte CsrY[]={2, 2, 3, 3};
byte SdtMin[]={30, 90, 45, 0}; //Sudut_Maks
byte SdtMax[]={150,170,90,180}; //Sudut_Maks
byte i, P=1;
```

Pada program diatas terdapat *library* yang di pakai untuk menjalankan alat yaitu *library* pada servo dan *library* pada *bluetooth ESP32* yang terhubung ke aplikasi *arduino bluetooth controller* di *smartphone* dan terdapat *konfigurasi* pada aplikasi *arduino bluetooth controller* untuk koneksi ke alat.

b. Program penggerak lengan robot menggunakan *push button*.

```
// Melalui PUSH BUTTON
void model () {
    if(!digitalRead(Btn[0])) {
        if(sudut[0]<SdtMax[0] && digitalRead(Rotate))
        sudut[0]++;
        if(sudut[0]>SdtMin[0] && !digitalRead(Rotate))
        sudut[0]--;
        myservo1.write(sudut[0]);
    }
    else if(!digitalRead(Btn[1])) {
        if(sudut[1]<SdtMax[1] && digitalRead(Rotate))
        sudut[1]++;
        if(sudut[1]>SdtMin[1] && !digitalRead(Rotate))
        sudut[1]--;
        myservo2.write(sudut[1]);
    }
    else if(!digitalRead(Btn[2])) {
        if(sudut[2]<SdtMax[2] && digitalRead(Rotate))
        sudut[2]++;
        if(sudut[2]>SdtMin[2] && !digitalRead(Rotate))
        sudut[2]--;
        myservo3.write(sudut[2]);
    }
    else if(!digitalRead(Btn[3])) {
        if(sudut[3]<SdtMax[3] && digitalRead(Rotate))
        sudut[3]++;
        if(sudut[3]>SdtMin[3] && !digitalRead(Rotate))
        sudut[3]--;
        myservo4.write(sudut[3]);
    }
    for(i=0; i<4; i++) { tampilkan(i); delay(1); }
}
```

Pada program diatas terdiri dari beberapa *push button* yang dapat menjalankan perintah yang berbeda, *push button mode* berpungsi untuk mengubah tiga *mode control push button, joystick, voice, push on \off* untuk menghidupkan dan mematikan robot, *push button servo* berpungsi untuk mengendalikan *servo, push button rotation* berpungsi untuk mengubah arah sebaliknya dari *push botton servo*.

c. Program penggerak lengan robot menggunakan terminal

```

//Melalui TERMINAL BTH
void mode2 () {
  if(dataBTH=="u" && sudut[0]<SdtMax[0]) sudut[0]++;
  //up
  if(dataBTH=="f" && sudut[1]<SdtMax[1]) sudut[1]++;
  //forwad
  if(dataBTH=="o" && sudut[2]<SdtMax[2]) sudut[2]++;
  //open
  if(dataBTH=="r" && sudut[3]<SdtMax[3]) sudut[3]++;
  //right
  if(dataBTH=="d" && sudut[0]>SdtMin[0]) sudut[0]--;
  //down
  if(dataBTH=="b" && sudut[1]>SdtMin[1]) sudut[1]--;
  //back
  if(dataBTH=="c" && sudut[2]>SdtMin[2]) sudut[2]--;
  //close
  if(dataBTH=="l" && sudut[3]>SdtMin[3]) sudut[3]--;
  //left
  if(dataBTH=="s") dataBTH=""; //Reset variable
  myservo1.write(sudut[0]);
  myservo2.write(sudut[1]);
  myservo3.write(sudut[2]);
  myservo4.write(sudut[3]);
  for(i=0; i<4; i++) { tampilkan(i); delay(10); }
}

```

Program di atas untuk mengubah *mode control* dari tiga *mode* yaitu, *push button*, *joystick*, *terminal*, dan *voice*. Dengan mengetikkan perintah M1 kendali push button, M2 kendali Joystick, M3 kendali Terminal, M4 kendali voice, Juga dapat mengendalikan pergerakan lengan robot

d. Program penggerak lengan robot menggunakan *joystick*

```

void mode3() { //Melalui JOYSTICK BTH
  if(dataBTH=="U" && sudut[0]<SdtMax[0]) sudut[0]++;
  //up
  if(dataBTH=="F" && sudut[1]<SdtMax[1]) sudut[1]++;
  //forwad
  if(dataBTH=="O" && sudut[2]<SdtMax[2]) sudut[2]++;
  //open
  if(dataBTH=="R" && sudut[3]<SdtMax[3]) sudut[3]++;
  //right
  if(dataBTH=="D" && sudut[0]>SdtMin[0]) sudut[0]--;
  //down
  if(dataBTH=="B" && sudut[1]>SdtMin[1]) sudut[1]--;
  //back
  if(dataBTH=="C" && sudut[2]>SdtMin[2]) sudut[2]--;
  //close
  if(dataBTH=="L" && sudut[3]>SdtMin[3]) sudut[3]--;
  //left
  if(dataBTH=="0" || dataBTH=="o") dataBTH=""; //Reset
  variable

  myservo1.write(sudut[0]);
  myservo2.write(sudut[1]);
  myservo3.write(sudut[2]);
  myservo4.write(sudut[3]);
  for(i=0; i<4; i++) { tampilkan(i); delay(10); }
}

```

Program diatas untuk menjalankan aplikasi *control Bluetooth* menggunakan *joystick*, yang di mana data U sebagai perintah pergerakan ke atas, data F sebagai perintah pergerakan ke depan, data O sebagai perintah pergerakan menutup, data R sebagai perintah pergerakan ke kanan, data D sebagai perintah pergerakan ke bawah, data B sebagai perintah pergerakan membuka, data C sebagai perintah pergerakan ke belakang, data L sebagai perintah pergerakan ke kiri, data 0 sebagai perintah mereset data.

e. Program penggerak lengan robot menggunakan *voice*.

```

//Melalui VOICE BTH
void mode4() {
  if(dataBTH == "naik"      && sudut[0]<SdtMax[0])
    sudut[0]++;
  else if(dataBTH == "maju" && sudut[1]<SdtMax[1])
    sudut[1]++;
  else if(dataBTH == "buka" && sudut[2]<SdtMax[2])
    sudut[2]++;
  else if(dataBTH == "kanan" && sudut[3]<SdtMax[3])
    sudut[3]++;
  else if(dataBTH == "turun" && sudut[0]>SdtMin[0])
    sudut[0]--;
  else if(dataBTH == "mundur" && sudut[1]>SdtMin[1])
    sudut[1]--;
  else if(dataBTH == "jepit" && sudut[2]>SdtMin[2])
    sudut[2]--;
  else if(dataBTH == "kiri" && sudut[3]>SdtMin[3])
    sudut[3]--;
  else if(dataBTH == "tahan") dataBTH=""; //Reset variable

  myservo1.write(sudut[0]);
  myservo2.write(sudut[1]);
  myservo3.write(sudut[2]);
  myservo4.write(sudut[3]);
  for(i=0; i<4; i++) { tampilkan(i); delay(50); }
}

```

Program diatas untuk menjalankan aplikasi *control Bluetooth* menggunakan *voice*, data program *voice* yang digunakan yaitu kata yang di ucapkan untuk mengontrol pergerakan robot.

 f. Tampilan *LCD*

```

void pilihan() {
  lcd.clear(); delay(1000);
  if(P==1) {
    lcd.setCursor(0,0); lcd.print("  Kontrol Melalui
");
    lcd.setCursor(0,1); lcd.print("    PUSH BUTTON
");
  }
  if(P==2) {
    lcd.setCursor(0,0); lcd.print("  Kontrol Melalui
");
    lcd.setCursor(0,1); lcd.print("  TERMINAL Bluetooth
");
  }
  if(P==3) {
    lcd.setCursor(0,0); lcd.print("  Kontrol Melalui
");
    lcd.setCursor(0,1); lcd.print("  JOYSTICK Bluetooth
");
  }
  if(P==4) {
    lcd.setCursor(0,0); lcd.print("  Kontrol Melalui
");
    lcd.setCursor(0,1); lcd.print("  VOICE Bluetooth
");
  }
  delay(2000);
  intro();
}

void intro() {
  lcd.setCursor(0,2); lcd.print(" SV1:      SV2:      ");
  lcd.setCursor(0,3); lcd.print(" SV3:      SV4:      ");
  for(i=0; i<4; i++) { tampilkan(i); delay(10); }
}

void tampilkan(byte a) {
  lcd.setCursor(CsrX[a],CsrY[a]);
  if(sudut[a]<100) {
    lcd.print(" "); if(sudut[a]<10) lcd.print(" ");
  }
  lcd.print(sudut[a]);
  lcd.write(223);
}

```

Program di atas untuk menampilkan *mode control* dan sudut derajat *servo* di *LCD*.

g. Program keseluruhan.

```

void setup() {
  Serial.begin(115200);
  Serial.println("Nama Bluetooth : "+
String(blueetooth_name));
  SerialBT.begin(blueetooth_name);

  pinMode(Btn[0], INPUT_PULLUP);
  pinMode(Btn[1], INPUT_PULLUP);
  pinMode(Btn[2], INPUT_PULLUP);
  pinMode(Btn[3], INPUT_PULLUP);
  pinMode(SW, INPUT_PULLUP);
  pinMode(Rotate, INPUT_PULLUP);
  lcd.begin();
  lcd.backlight();
  pilihan();
  myservo1.attach(SV[0]); //shoulder (up-down)
  myservo2.attach(SV[1]); //elbow (forwr-d-back)
  myservo3.attach(SV[2]); //gripper (open-close)
  myservo4.attach(SV[3]); //base (right-left)
  myservo1.write(sudut[0]);
  myservo2.write(sudut[1]);
  myservo3.write(sudut[2]);
  myservo4.write(sudut[3]);
  delay(2000);
}

void loop() {
  byte tunggu=0;
  while(!digitalRead(SW) && tunggu<10) {tunggu++;
delay(150); }
  if(tunggu>=10) {
    tunggu=0; P++; if(P>4) P=1; pilihan();
  }

  if(SerialBT.available()>0) dataBTH="";
  while(SerialBT.available()>0) {
    char c = SerialBT.read();
    dataBTH += c;
  }
  if(dataBTH.length()>0) {
    Serial.println("dataBTH: "+String(dataBTH));
    if(dataBTH=="M1") { dataBTH=""; P=1; pilihan();}
    if(dataBTH=="M2") { dataBTH=""; P=2; pilihan(); }
    if(dataBTH=="M3") { dataBTH=""; P=3; pilihan(); }
    if(dataBTH=="M4") { dataBTH=""; P=4; pilihan(); }
    if(P==2) mode2();
    if(P==3) mode3();
    if(P==4) mode4();
  }
  if(P==1) mode1();
}

```

D. Pengujian Alat

Pengujian pada alat untuk uji respon perintah pada lengan robot dalam beberapa tahap percobaan sebagai berikut:

1. Uji respon pergerakan *servo* dengan kendali fush button

1. Uji respon pergerakan *servo* bahu.

Percobaan pertama dilakukan uji respon perintah *servo* kiri pada lengan dengan sudut 90° sampai 170°

Tabel 4. 1 Uji respon pergerakan *servo* kiri

No	waktu pergerakan (s)	
	Ke Depan 90°	Ke Belakang 170°
1	3,21	3,19
2	3,18	3,18
3	3,20	3,15
4	3,15	3,16
5	3,19	3,18
Rata-rata	3,18	3,17

Pada percobaan uji respon pergerakan *servo* bahu dengan sudut 90° sampai 170° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke depan 3,21 detik dan ke belakang 3,19 detik, Pada pengujian kedua waktu pergerakan ke depan 3,18 detik dan ke belakang 3,18 detik, Pada pengujian ketiga waktu pergerakan ke depan 3,20 detik dan ke belakang 3,15 detik, Pada pengujian keempat waktu pergerakan ke depan 3,15 detik dan ke belakang 3,16 detik, dan Pada pengujian kelima waktu pergerakan ke depan 3,19 detik dan ke belakang 3,18 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan ke depan yaitu 3,18 detik dan ke belakang 3,17 detik. Adapun dokumentasi pengukuran pada pergerakan lengan ke depan dan ke belakang dapat dilihat pada gambar 4.5 dan 4.6.



Gambar 4. 5 pengukuran pergerakan lengan ke bawah



Gambar 4. 6 pengukuran pergerakan lengan ke atas

2. Uji respon pergerakan *servo* lengan.

Percobaan kedua dilakukan uji respon perintah *servo* kanan pada lengan dengan sudut 30° sampai 150°

Tabel 4. 2 Uji respon pergerakan *servo* kanan

No	waktu pergerakan (s)	
	Ke Atas 30°	Ke Bawah 150°
1	4,47	4,46
2	4,45	4,48
3	4,43	4,45
4	4,46	4,44
5	4,42	4,40
Rata rata	4,44	4,44

Pada percobaan uji respon pergerakan *servo* lengan dengan sudut 30° sampai 150° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke atas 4,47 detik dan ke bawah 4,46 detik, Pada pengujian kedua waktu pergerakan ke atas 4,45 detik dan ke bawah 4,48 detik, Pada pengujian ketiga waktu pergerakan ke atas 4,43 detik dan ke bawah 4,45 detik, Pada pengujian keempat waktu pergerakan ke atas 4,46 detik dan ke bawah 4,44 detik, dan Pada pengujian kelima waktu pergerakan ke atas 4,42 detik dan ke bawah 4,40 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan ke atas yaitu 4,44 detik dan ke bawah 4,44 detik. Adapun dokumentasi pengukuran pada pergerakan lengan ke atas dan ke bawah dapat dilihat pada gambar 4.7 dan 4.8.



Gambar 4. 7 pengukuran pergerakan lengan ke bawah



Gambar 4. 8 pengukuran pergerakan lengan ke atas

3. Uji respon pergerakan *gripper servo*.

Percobaan ketiga dilakukan uji respon perintah *gripper servo* pada lengan robot dengan sudut 45° sampai 90° .

Tabel 4. 3 Uji respon pergerakan *gripper servo*

No	waktu pergerakan (s)	
	Membuka 45°	Menutup 90°
1	1,90	1,66
2	1,86	1,64
3	1,72	1,63
4	1,71	1,65
5	1,71	1,63
Rata-rata	1,78	1,64

Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 45° sampai 90° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan membuka 1,90 detik dan menutup 1,66 detik, Pada pengujian kedua waktu pergerakan membuka 1,86 detik dan menutup 1,64 detik, Pada pengujian ketiga waktu pergerakan membuka 1,72 detik dan menutup 1,63 detik, Pada pengujian keempat waktu pergerakan membuka 1,71 detik dan menutup 1,65 detik, Pada pengujian kelima waktu pergerakan membuka 1,71 detik dan menutup 1,63 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan membuka yaitu 1,78 detik dan menutup yaitu 1,64 detik. Adapun dokumentasi pengukuran pada pergerakan lengan membuka dan menutup dapat dilihat pada gambar 4.9 dan 4.10.



Gambar 4. 9 Pengukuran pergerakan lengan membuka



Gambar 4. 10 Pengukuran pergerakan lengan menutup

4. Uji respon pergerakan *base servo*

Percobaan keempat dilakukan uji respon perintah *base servo* pada lengan dengan sudut 0° sampai 180°

Tabel 4. 4 Uji respon pergerakan *base servo*

No	waktu pergerakan (s)	
	Ke Kiri 0°	Ke Kanan 180°
1	6,83	6,81
2	6,81	6,82
3	6,83	6,81
4	6,84	6,80
5	6,82	6,83
Rata-rata	6,82	6,81

Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 0° sampai 180° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke kiri 6,83 detik dan ke kanan 6,81 detik, Pada pengujian kedua waktu pergerakan ke kiri 6,81 detik dan ke kanan 6,82 detik, Pada pengujian ketiga waktu pergerakan ke kiri 6,83 detik dan ke kanan 6,81 detik, Pada pengujian keempat ke kiri 6,84 detik dan ke kanan 6,80 detik, dan Pada pengujian kelima waktu pergerakan ke kiri 6,82 detik dan ke kanan 6,83 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan ke kiri yaitu 6,82 detik dan ke kanan yaitu 6,81 detik. Adapun dokumentasi pengukuran pada pergerakan lengan ke kiri dan ke kanan dapat dilihat pada gambar 4.11 dan 4.12



Gambar 4. 11 Pengukuran pergerakan lengan ke kiri



Gambar 4. 12 Pengukuran pergerakan lengan ke kanan

2. Uji respon pergerakan *servo* dengan kendali aplikasi *bluetooth control joystick*

1. Uji respon pergerakan *servo* bahu, Percobaan pertama dilakukan uji respon perintah *servo* kiri pada lengan dengan sudut 90° sampai 170°

Tabel 4. 5 Uji respon pergerakan *servo* kiri

No	waktu pergerakan (s)	
	Ke Depan 90°	Ke Belakang 170°
1	6,20	6,21
2	6,24	6,26
3	6,22	6,25
4	6,21	6,23
5	6,23	6,22
Rata-rata	6,22	6,23

Pada percobaan uji respon pergerakan *servo* bahu dengan sudut 90° sampai 170° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke depan 6,20 detik dan ke belakang 6,21 detik, Pada pengujian kedua waktu pergerakan ke depan 6,24 detik dan ke belakang 6,26 detik, Pada pengujian ketiga waktu pergerakan ke depan 6,22 detik dan ke belakang 6,25 detik, Pada pengujian keempat waktu pergerakan ke depan 6,21 detik dan ke belakang 6,23 detik, dan Pada pengujian kelima waktu pergerakan ke depan 6,23 detik dan ke belakang 6,22 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan kedepan yaitu 6,22 detik dan kebelakang 6,23 detik.

2. Uji respon pergerakan servo lengan, Percobaan kedua dilakukan uji respon perintah *servo* kanan pada lengan dengan sudut 30° sampai 150°

Tabel 4. 6 Uji respon pergerakan servo kanan

No	waktu pergerakan (s)	
	Ke Atas 30°	Ke Bawah 150°
1	9,01	9,05
2	9,07	9,09
3	9,10	9,08
4	9,06	9,06
5	9,08	9,09
Rata-rata	9,06	9,07

Pada percobaan uji respon pergerakan *servo* lengan dengan sudut 30° sampai 150° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke atas 9,01 detik dan ke bawah 9,05 detik, Pada pengujian kedua waktu pergerakan ke atas 9,07 detik dan ke bawah 9,08 detik, Pada pengujian ketiga waktu pergerakan ke atas 9,10 detik dan ke bawah 9,08 detik, Pada pengujian keempat waktu pergerakan ke atas 9,06 detik dan ke bawah 9,06 detik, dan Pada pengujian kelima waktu pergerakan ke atas 9,08 detik dan ke bawah 9,09 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan ke atas yaitu 9,06 detik dan ke bawah yaitu 9,07 detik.

3. Percobaan ketiga dilakukan uji respon perintah *gripper servo* pada lengan robot dengan sudut 45° sampai 90° .

Tabel 4.7 Uji respon pergerakan *gripper servo*

No	waktu pergerakan (s)	
	Membuka 45°	Menutup 90°
1	3,97	3,96
2	3,93	3,95
3	3,91	3,94
4	3,96	3,96
5	3,99	3,98
Rata-rata	3,95	3,95

Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 45° sampai 90° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan membuka 3,97 detik dan menutup 3,96 detik, Pada pengujian kedua waktu pergerakan membuka 3,93 detik dan menutup 3,95 detik, Pada pengujian ketiga waktu pergerakan membuka 3,91 detik dan menutup 3,94 detik, Pada pengujian keempat membuka 3,96 detik dan menutup 3,96 detik, dan Pada pengujian kelima waktu pergerakan membuka 3,99 detik dan menutup 3,98 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan membuka yaitu 3,95 detik dan menutup yaitu 3,95 detik.

4. Percobaan keempat dilakukan uji respon perintah base *servo* pada lengan dengan sudut 0° sampai 180°

Tabel 4. 8 Uji respon pergerakan *base servo*

No	waktu pergerakan (s)	
	Ke Kiri 0°	Ke Kanan 180°
1	13,46	13,48
2	13,42	13,41
3	13,44	13,45
4	13,43	13,48
5	13,47	13,42
Rata-rata	13,44	13,44

Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 0° sampai 180° dilakukan lima kali pengujian dengan dua pergerakan. Pada pengujian pertama waktu pergerakan ke kiri 13,46 detik dan ke kanan 13,48 detik, Pada pengujian kedua waktu pergerakan ke kiri 13,42 detik dan ke kanan 13,41 detik, Pada pengujian ketiga waktu pergerakan ke kiri 13,44 detik dan ke kanan 13,45 detik, Pada pengujian keempat ke kiri 13,43 detik dan ke kanan 13,48 detik, dan Pada pengujian kelima waktu pergerakan ke kiri 13,47 detik dan ke kanan 13,42 detik. Adapun nilai rata-rata waktu yang didapat pada lima kali pengujian pada pergerakan ke kiri yaitu 13,44 detik dan ke kanan yaitu 13,44 detik.

3. Uji respon *control bluetooth terminal mode*

Tabel 4. 9 uji respon *control bluetooth terminal mode*

NO	Jarak respon tanpa penghalang				Jarak respon dengan penghalang		
	5m	10m	15m	20m	5m	10m	15m
1	✓	✓	✓	✓	✓	✓	✗
2	✓	✓	✓	✗	✓	✓	✗
3	✓	✓	✓	✓	✓	✓	✗
4	✓	✓	✓	✗	✓	✓	✗
5	✓	✓	✓	✗	✓	✓	✗
6	✓	✓	✓	✓	✓	✓	✓
7	✓	✓	✓	✗	✓	✓	✗
8	✓	✓	✓	✗	✓	✓	✗
9	✓	✓	✓	✗	✓	✓	✗
10	✓	✓	✓	✓	✓	✓	✗
Rata rata	100%	100%	100%	40%	100%	100%	10%

Pada percobaan uji respon *control* aplikasi *bluetooth* dilakukan lima kali pengujian dengan dua tahap yaitu jarak respon tanpa penghalang dan jarak respon dengan penghalang menggunakan tembok, yang dilakukan lima kali pengujian setiap jaraknya, dengan simbol centang (✓), menandakan bahwa alat yang di kontrol dengan jarak yang di tentukan dapat respon, dan silang (✗),menandakan bahwa alat yang di kontrol dengan jarak yang di tentukan tidak dapat respon.

Pengujian jarak respon tanpa penghalang,

Pada pengujian pertama jarak respon tanpa penghalang di jarak 5m sampai 20m dapat merespon. Pada pengujian ke dua jarak respon tanpa penghalang di jarak 5m sampai 15m dapat merespon dan jarak 20m tidak dapat merespon. Pada pengujian ke tiga jarak respon tanpa penghalang di jarak 5m sampai 20m dapat merespon. Pada pengujian ke empat jarak respon tanpa

penghalang di jarak 5m sampai 15m dapat merespon dan jarak 20m tidak dapat merespon.

Pada pengujian ke lima jarak respon tanpa penghalang di jarak 5m sampai 15m dapat merespon dan jarak 20m tidak dapat merespon.

Pada pengujian ke enam jarak respon tanpa penghalang di jarak 5m sampai 20m dapat merespon. Pada pengujian ke tuju jarak respon tanpa penghalang di jarak 5m sampai 15m dapat merespon dan jarak 20m tidak dapat merespon. Pada pengujian ke delapan jarak respon tanpa penghalang di jarak 5m sampai 15m dapat merespon 20m tidak dapat merespon. Pada pengujian ke sembilan jarak respon tanpa penghalang di jarak 5m sampai 15m dapat merespon dan jarak 20m tidak dapat merespon. Pada pengujian ke sepuluh jarak respon tanpa penghalang di jarak 5m sampai 20m dapat merespon.

Adapun tingkat keberhasilan yang dihasilkan pada jarak 5 meter 100% keberhasilan, jarak 10 meter 100% keberhasilan, 15 meter 100% keberhasilan, 20 meter 40% keberhasilan.

Pengujian jarak respon dengan penghalang.

Pada pengujian pertama jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

pada pengujian ke dua jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

pada pengujian ke tiga jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 25m tidak dapat merespon.

pada pengujian ke empat jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

pada pengujian ke lima jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

Pada pengujian enam jarak respon dengan penghalang di jarak 5m sampai 15m dapat merespon.

pada pengujian ke tujuh jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

pada pengujian ke delapan jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 25m tidak dapat merespon.

pada pengujian ke sembilan jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

pada pengujian ke sepuluh jarak respon dengan penghalang di jarak 5m sampai 10m dapat merespon dan di jarak 15m tidak dapat merespon.

Adapun tingkat keberhasilan jarak respon yang dihasilkan pada jarak 5 meter 100% keberhasilan, jarak 10 meter 100% keberhasilan, 15 meter 10% keberhasilan.

4. Pada percobaan uji respon *control* aplikasi *bluetooth voice*

Tabel 4. 10 Uji respon *control bluetooth voice*

NO	Respon <i>control bluetooth voice</i>			
	Volume sedang, pada saat tenang	Volume keras, pada saat tenang	Volume sedang, pada saat ribut	Volume keras, pada saat ribut
1	✓	✓	✓	✗
2	✓	✓	✗	✗
3	✓	✓	✗	✓
4	✓	✓	✓	✗
5	✓	✓	✓	✓
6	✓	✓	✓	✗
7	✓	✓	✗	✗
8	✓	✓	✗	✓
9	✓	✓	✓	✗
10	✓	✓	✓	✓
Rata rata	100%	100%	60%	40%

Pada percobaan uji respon *control* aplikasi *bluetooth voice* dilakukan lima kali pengujian dengan empat tahap yaitu, respon *control bluetooth voice* dengan *volume* sedang pada saat tenang, respon *control bluetooth voice* dengan *volume* keras pada saat tenang, dan respon *control bluetooth voice* dengan *volume* sedang pada saat ribut, respon *control bluetooth voice* dengan *volume* keras pada saat ribut, yang dilakukan lima kali pengujian setiap jaraknya, dengan simbol centang (✓), menandakan bahwa alat yang di kontrol dengan *volume* suara dan kondisi yang ditentukan dapat respon, dan silang (✗),menandakan bahwa alat yang di kontrol dengan *volume* suara dan kondisi yang ditentukan tidak dapat respon.

Pada pengujian pertama respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume*

sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke dua respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut tidak dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke tiga respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut tidak dapat merespon, suara *volume* keras saat ribut dapat merespon.

Pada pengujian ke empat respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke lima respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke enam respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke tujuh respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut tidak dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke delapan respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut tidak dapat merespon, suara *volume* keras saat ribut dapat merespon.

Pada pengujian ke sembilan respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Pada pengujian ke sepuluh respon suara *volume* sedang saat tenang dapat merespon, suara *volume* keras saat tenang dapat merespon dan, suara *volume* sedang saat ribut dapat merespon, suara *volume* keras saat ribut tidak dapat merespon.

Adapun tingkat keberhasilan respon suara *volume* sedang saat tenang 10 kali percobaan dapat merespon 100% keberhasilan, Suara *volume* keras saat tenang 10 kali percobaan dapat merespon 100% keberhasilan, Suara *volume* sedang saat ribut 10 kali percobaan dapat merespon 60% keberhasilan, Suara *volume* keras saat ribut 10 kali percobaan dapat merespon 40% keberhasilan.

BAB V

PENUTUP

A. Kesimpulan

Setelah melakukan penelitian pada sistem kendali lengan robot 4 *dof* melalui *push button* dan aplikasi *arduino bluetooth kontrol (joystick, terminal, voice)*, dapat disimpulkan bahwa:

1. Pada hasil pengujian pergerakan lengan robot menggunakan *control push button*, Pada percobaan uji respon pergerakan *servo* bahu dengan sudut 90° sampai 170° , rata-rata waktu pergerakan ke depan yaitu 3,18 detik dan ke belakang 3,17 detik, Pada percobaan uji respon pergerakan *servo* lengan dengan sudut 30° sampai 150° , rata-rata waktu pergerakan ke atas yaitu 4,44 detik dan ke bawah 4,44 detik, Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 45° sampai 90° , rata-rata waktu pergerakan membuka yaitu 1,78 detik dan menutup yaitu 1,64 detik, Pada percobaan uji respon pergerakan *base servo* dengan sudut 0° sampai 180° rata-rata waktu pergerakan ke kiri yaitu 6,82 detik dan ke kanan yaitu 6,81 detik,
2. Pada hasil pengujian pergerakan lengan robot menggunakan *control aplikasi Arduino bluetooth controller joystick*, Pada percobaan uji respon pergerakan *servo* bahu dengan sudut 90° sampai 170° , rata-rata waktu pergerakan ke depan yaitu 6,22 detik dan ke belakang 6,23 detik, Pada percobaan uji respon pergerakan *servo* lengan dengan sudut 30° sampai 150° , rata-rata waktu

pergerakan ke atas yaitu 9,06 detik dan ke bawah yaitu 9,07 detik, Pada percobaan uji respon pergerakan *servo* pencapit dengan sudut 45° sampai 90° , rata-rata waktu pergerakan membuka yaitu 3,95 detik dan menutup yaitu 3,95 detik, Pada percobaan uji respon pergerakan *base servo* dengan sudut 0° sampai 180° rata-rata waktu pergerakan ke kiri yaitu 13,44 detik dan ke kanan yaitu 13,44 detik.

3. Pada percobaan uji respon *control aplikasi bluetooth voice* dilakukan lima kali pengujian dengan empat tahap yaitu, respon *control bluetooth voice* dengan *volume* sedang pada saat tenang, respon *control bluetooth voice* dengan *volume* keras pada saat tenang, dan respon *control bluetooth voice* dengan *volume* sedang pada saat ribut, respon *control bluetooth voice* dengan *volume* keras pada saat ribut, Adapun tingkat keberhasilan respon suara *volume* sedang saat tenang 10 kali percobaan, dapat merespon 100% keberhasilan, suara *volume* pada saat tenang 10 kali percobaan dapat merespon 100% keberhasilan, suara *volume* sedang saat ribut 10 kali percobaan dapat merespon 60% keberhasilan, suara *volume* keras saat ribut 10 kali percobaan dapat merespon 40% keberhasilan.
4. Jarak maksimal respon dari aplikasi *Bluetooth control* ke robot yaitu 20 Meter tanpa penghalang.
5. Daya ketahanan baterai dapat bertahan selama 30 menit saat baterai terisi full, dengan menggunakan *Baterai VTC6 3000mah* sebanyak dua buah.

6. Sistem kendali push button lebih responsif di banding sistem kendali aplikasi bluetooth karna bluetooth memerlukan delay respon dari RX aplikasi menuju TX ESP32

B. Saran

1. Dibutuhkan penelitian lanjutan pada lengan robot untuk dikembangkan yang nantinya bisa digunakan sebagai robot pengangkut barang.
2. Membutuhka pengembangan lanjutan untuk system kendali aplikasi Bluetooth agar delay servo tidak selisih jauh dari system kendali push button.

DAFTAR PUSTAKA

- Abdul Muid. (2015). Rancang Bangun Lengan Robot Sebagai Alat Pemindah Barang Berdasarkan Warna Menggunakan Sensosr *Fotodioda*. Argon jurnal. 9(2):50-51.
- Aldy Razor. 2020. *Push Button Arduino, Pengertian, Fungsi, dan Prinsip Kerja*, Diperoleh dari: Push Button Arduino: Pengertian, Fungsi, dan Prinsip Kerja - Aldyrazor.com. (Diakses 02 Agustus 2023).
- Anak Agung Gde Ekayana & Gusti Ngurah Kade Ary P, (2017). *Rancang Bangun Prototype Sistem Kendali Lengan Robot Menggunakan Interface Wireless 2.4 Ghz*.Simetris. 6(2):43-45.
- Apps valley. 2019. *Arduino Bluetooth Controller - All in One for Android*. Diperoleh dari: Arduino Bluetooth Controller - All in One - Free download and software reviews - CNET Download. (Diakses 02 Agustus 2023).
- Bagus Utomo, Noor Yulita Dwi Setyaningasih, Mohammad Iqbal. (2020). *Kendali Robot Lengan 4 Dof Berbasis Arduino Uno dan Sensor MPU-6050*. Kudus: bagusutomo0301@gmail.com
- Nainggolan, E.M., Rusdinar, A., & Sunarya, U. (2015). *Perancangan dan implementasi tangan robot dengan menggunakan elektromiogram*. Proceedings of engineering, 2(2).
- NewbloggerThemes. 2020. *Motor Servo - Pengertian, Jenis, Prinsip Kerja & Bagian Motor Servo*. Diperoleh dari: Motor Servo - Pengertian, Jenis, Prinsip Kerja & Bagian Motor Servo ~ Studi Elektronika (webstudi.site). (Diakses 02 Agustus 2023).
- Prayogo TS. Sistem Kontrol Charger Handphone Otomatis Berbasis Android. Dr Diss Inst Bisnis dan Inform Stikom Surabaya. 2016;7(2):107–15.
- Purwono Prasetyawan, Yopan Ferdianto, Syaiful Ahdan, Fika Trisnawati. (2018). *Pengendali Lengan Robot Dengan Mikrokontroler Arduino Berbasis Smartphone*. Lampung: purwono.prasetyawan@teknokrat.ac.id
- Sejati H. Gambar 2.1 ARDUINO UNO R3 2.2 Perangkat Lunak Arduino IDE. Modul Convert LM2596. 2019;3–12.
- Syah Alam, Gunawan Tjahjadi, Nur Rahma Yenita, Supriyadi. (2021). *Rancang Bangun Prototipe Pengendalian Lengan Robot (Robotic Arm) sebagai Pemindah Barang Berbasis Internet Of Things*. Jakarta Barat: syah.alam@trisakti.ac.id
- Syukranullah. (2019). Rancang Bangun Robot lengan Berbasis Microcontroler Arduino Uno. Medan: Syukranullahza92@gmail.com

Sejati H. Gambar 2.1 ARDUINO UNO R3 2.2 Perangkat Lunak Arduino IDE. Modul Convert LM2596. 2019;3–12.

Prayogo TS. Sistem Kontrol Charger Handphone Otomatis Berbasis Android. Dr Diss Inst Bisnis dan Inform Stikom Surabaya. 2016;7(2):107–15.

Zamrodah Y. Monitoring Penggunaan Daya Listrik Secara Detail Menggunakan Mikrokotroller Arduino Uno. 2016;15(2):1–23.