

PENGEMBANGAN ROBOT REHABILITASI LENGAN BAGI
PENDERITA STROKE BERBASIS *INTERNET OF THINGS*
MENGUNAKAN *SMARTPHONE*



SKRIPSI

Diajukan sebagai salah satu syarat untuk menyelesaikan
Pendidikan diploma empat (D-4) Teknik Mekatronika
Jurusan Teknik Mesin
Politeknik Negeri Ujung Pandang

FIKRI FATAHILLAH
FITRIANI

444 17 013
444 17 011

PROGRAM STUDI D-4 TEKNIK MEKATRONIKA
JURUSAN TEKNIK MESIN
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2021

HALAMAN PENGESAHAN

Skripsi dengan judul “**Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone***” oleh Fikri Fatahillah NIM 44417013 dan Fitriani NIM 44417011 telah diterima dan disahkan sebagai salah syarat untuk memperoleh gelar Sarjana Terapan Teknik pada program studi Teknik Mekatronika Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.



Makassar, 07 September 2021

Dosen Pembimbing I

Ir. Lewi, M.T.
NIP. 19650913 199103 1 006

Dosen Pembimbing II

Dr. Ir. Simon Ka'ka, M.T.
NIP. 19590913 198803 1 001



Mengetahui,
Koordinator Program Studi

Dr. Ir. Simon Ka'ka, M.T.
NIP. 19590913 198803 1 001

HALAMAN PENERIMAAN

Pada hari ini, Selasa tanggal 7 September 2021, tim penguji ujian sidang skripsi telah menerima skripsi mahasiswa: Fikri Fatahillah NIM 44417013 dan Fitriani NIM 44417011 dengan judul “Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone*”.



Makassar, 7 September 2021

Tim Penguji Ujian Sidang Skripsi:

1. Ir. Remigius Tandioga, M.Eng.Sc.

Ketua

(.....
.....)

2. Imran Habriansyah, S.ST., M.T.

Sekretaris

(.....
.....)

3. Dr.Eng. Abdul Kadir Muhammad, S.T., M.Eng.

Anggota

(.....
.....)

4. Dr.Eng. Akhmad Taufik, S.T., M.T.

Anggota

(.....
.....)

5. Dr. Ir. Simon Ka’Ka, M.T.

Anggota

(.....
.....)

6. Ir. Lewi, M.T.

Anggota

(.....
.....)

KATA PENGANTAR

Puji syukur atas kehadiran Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya penulisan skripsi ini, yang berjudul “Pengembangan Robot Rehabilitasi Lengan Bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone*” dapat diselesaikan dengan baik.

Skripsi ini disusun berdasarkan pengerjaan tugas akhir yang telah penulis lakukan selama kurang lebih enam bulan. Pengerjaan tugas akhir dan penyusunan skripsi ini penulis lakukan untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan D-4 Program Studi Teknik Mekatronika, Jurusan Teknik Mesin, Politeknik Negeri Ujung Pandang.

Skripsi ini dapat kami susun dengan baik karena adanya masukan dan dukungan dari berbagai pihak, baik yang berupa informasi, arahan dan bimbingan., oleh karena itu penulis mengucapkan sebanyak-banyaknya terima kasih kepada:

1. Allah SWT yang telah memberikan rahmat, ilmu, kesehatan dan perlindungan kepada penulis selama mengerjakan skripsi hingga saat ini.
2. Ibu dan ayah yang sangat penulis cintai, tidak terkira banyaknya dukungan dan doa yang diberikan kepada penulis.
3. Bapak Prof. Ir. Muhammad Anshar, M.Si., Ph.D., selaku Direktur Politeknik Negeri Ujung Pandang.
4. Bapak Rusdi Nur, S.S.T., M.T., Ph.D., selaku Ketua Jurusan Teknik Mesin.

5. Bapak Dr. Ir. Simon Ka’Ka, M.T., selaku Koordinator Program Studi Teknik Mekatronika sekaligus selaku pembimbing II.
6. Bapak Ir. Lewi, M.T., selaku pembimbing I dan bapak Dr. Ir. Simon Ka’Ka’, M.T., selaku pembimbing II. Begitu banyak bantuan yang telah diberikan kepada penulis.
7. Dosen dan Tenaga Kependidikan Politeknik Negeri Ujung Pandang.
8. Teman-teman seperjuangan dari Teknik Mekatronika yang telah banyak meluangkan waktu dan kesempatan untuk membantu dan mendukung proses pengerjaan tugas akhir serta skripsi penulis.
9. Pihak-pihak yang secara langsung maupun tidak yang telah memberikan kontribusi dalam proses pengerjaan tugas akhir serta skripsi penulis.

Penulis menyadari bahwa tentu saja ada begitu banyak kekurangan dan kesalahan dalam skripsi ini, begitu pula dengan peralatan yang bersangkutan dengan skripsi ini. Untuk itu kami mengharapkan adanya *feedback* baik berupa saran ataupun kritikan dari pembaca sehingga menjadi bahan bagi penulis untuk memperbaiki skripsi ini. Semoga skripsi ini bisa membawa manfaat bagi pembaca secara umum dan bagi penulis secara khusus.

Makassar, 01 September 2021

Penulis

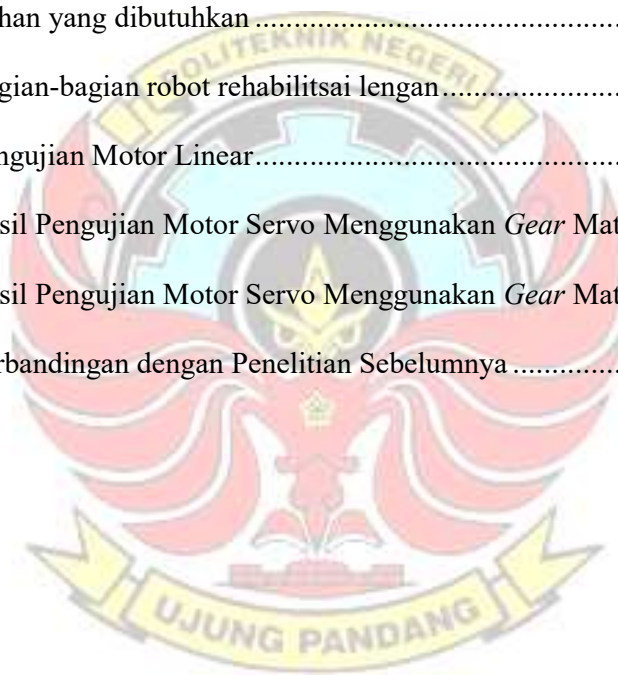
DAFTAR ISI

	hlm.
HALAMAN SAMPUL.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PENERIMAAN.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vi
DAFTAR TABEL.....	viii
DAFTAR GAMBAR.....	ix
DAFTAR LAMPIRAN.....	xi
SURAT PERNYATAAN.....	xii
SURAT PERNYATAAN.....	xiii
RINGKASAN.....	xiv
SUMMARY.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Ruang Lingkup Penelitian.....	4
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	5
BAB II TINJAUAN PUSTAKA.....	6
2.1 Stroke.....	6
2.2 Rehabilitasi Medik.....	7
2.3 Robot Rehabilitasi Stroke.....	9
2.4 <i>Internet of Things (IoT)</i>	10

2.5	Komponen Mekanik	13
2.6	Komponen Elektronik.....	17
2.7	Penelitian Sebelumnya.....	30
BAB III METODE PENELITIAN		33
3.1	Tempat dan Waktu Penelitian	33
3.2	Alat dan Bahan Penelitian.....	33
3.3	Metode Perancangan.....	35
3.4	Studi Literatur.....	36
3.5	Perancangan Perangkat Keras	36
3.6	Perancangan Perangkat Lunak	38
BAB IV HASIL DAN PEMBAHASAN		41
4.1	Hasil Perancangan Mekanik.....	41
4.2	Hasil Perancangan Elektronik	45
4.3	Hasil Perancangan Program	50
4.4	Pengujian Motor Linear	52
4.5	Pengujian Motor Servo	53
4.6	Pengujian Aplikasi.....	55
4.7	Perbandingan dengan Penelitian Sebelumnya.....	58
BAB V PENUTUP		59
5.1	Kesimpulan	59
5.2	Saran	59
DAFTAR PUSTAKA		60
LAMPIRAN		63

DAFTAR TABEL

	hlm.
Tabel 2. 1 <i>Upper Extremity</i> untuk <i>Flexi</i> dan <i>Extensi</i>	10
Tabel 2. 2 <i>Upper Extremity</i> untuk Horizontal <i>Adduksi</i> dan <i>Abduksi</i>	10
Tabel 2. 3 Deskripsi Arduino Mega 2560.....	18
Tabel 3. 1 Alat yang Dibutuhkan.....	33
Tabel 3. 2 Bahan yang dibutuhkan	34
Tabel 4. 1 Bagian-bagian robot rehabilitasi lengan.....	42
Tabel 4. 2 Pengujian Motor Linear.....	52
Tabel 4. 3 Hasil Pengujian Motor Servo Menggunakan <i>Gear</i> Material Besi	53
Tabel 4. 4 Hasil Pengujian Motor Servo Menggunakan <i>Gear</i> Material Plastik ...	54
Tabel 4. 5 Perbandingan dengan Penelitian Sebelumnya	58



DAFTAR GAMBAR

	hlm.
Gambar 2. 1 Penyebab Stroke	6
Gambar 2. 2 Stroke <i>Hemoragik</i> dan Stroke <i>Iskemik</i>	7
Gambar 2. 3 Robot Rehabilitasi	9
Gambar 2. 4 Motor Linear	14
Gambar 2. 5 Motor <i>dc</i> Power Window.....	15
Gambar 2. 6 Motor Servo.....	17
Gambar 2. 7 <i>Board</i> Arduino Mega 2560	18
Gambar 2. 8 NodeMCU.....	22
Gambar 2. 9 Adaptor 12V.....	24
Gambar 2. 10 Push Button	25
Gambar 2. 11 BTS 7960	26
Gambar 2. 12 <i>Liquid Crystal Display</i> (LCD).....	27
Gambar 2. 13 Keypad 4x4	28
Gambar 2. 14 MIT App Inventor.....	29
Gambar 2. 15 <i>Step Down</i>	29
Gambar 3. 1 <i>FlowChart</i> Sistem Perancangan	35
Gambar 3. 2 Robot Rehabilitasi Lengan. (a) Tampak Samping. (b) Tampak Depan. (c) Tampak Atas. (d) Tampak Isometrik.	37
Gambar 3. 3 Skema Rangkaian Elektronik Robot Rehabilitasi Lengan.....	38
Gambar 3. 4 Diagram Alir Sistem Pergerakan.....	40
Gambar 4. 1 Hasil Perancangan Mekanik.....	41

Gambar 4. 2 Penggantian Pergerakan Kiri-kanan dengan Motor Servo.....	43
Gambar 4. 3 <i>Gear</i> . (a) Transmisi Pergerakan Kiri-kanan Sebelumnya dan (b) Penggantian Menggunakan Gear Plastik.....	44
Gambar 4. 4 Penggantian Motor <i>Linear</i>	44
Gambar 4. 5 Rangkaian dan Komponen Elektronik.....	46
Gambar 4. 6 Tampilan Depan <i>Box Panel Controller</i>	47
Gambar 4. 7 <i>Wiring</i> Komunikasi Serial Arduino Mega dan NodeMCU.....	47
Gambar 4. 8 <i>Wiring</i> Motor dan Arduino Mega.....	48
Gambar 4. 9 <i>Limit Switch</i> dan Arduino Mega.....	48
Gambar 4. 10 <i>Button</i> dan Arduino Mega.....	49
Gambar 4. 11 <i>Keypad</i> , LCD dan Arduino Mega.....	49
Gambar 4. 12 <i>Layout</i> Papan PCB.....	50
Gambar 4. 13 Tampilan Aplikasi	51
Gambar 4. 14 Aplikasi Mengirim Data ke <i>Firestore</i>	56
Gambar 4. 15 Aplikasi Mengirim Data ke <i>Google Spreadsheet</i>	56
Gambar 4. 16 Tampilan Data dari <i>Firestore</i> ke NodeMCU	57
Gambar 4. 17 Tampilan Data dari NodeMCU diserialkan ke Arduino Mega	57

DAFTAR LAMPIRAN

	hlm.
Lampiran 1 Tahapan-tahapan pengoperasian.....	63
Lampiran 2 Program di Arduino Mega.....	65
Lampiran 3 Program di NodeMCU	84
Lampiran 4 Program di MIT App Inventor.....	86
Lampiran 5 Dokumentasi.....	87
Lampiran 6 Biodata Penulis	88



SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini:

Nama : Fikri Fatahillah

NIM : 44417013

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul “Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone*” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut di atas tidak benar, saya siap menanggung risiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 07 September 2021



10000
SUPLEKSI BERSILU BERSIPAN
METERAN
TEMPEL
89022AJX480588547

Fikri Fatahillah
44417013

SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini:

Nama : Fitriani

NIM : 44417011

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul “Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone*” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut di atas tidak benar, saya siap menanggung risiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 07 September 2021



Fitriani
44417011

PENGEMBANGAN ROBOT REHABILITASI LENGAN BAGI PENDERITA STROKE BERBASIS *INTERNET OF THINGS* MENGUNAKAN *SMARTPHONE*

RINGKASAN

Pola hidup masyarakat yang semakin modern telah menuntut keinginan menyelesaikan suatu permasalahan dengan cepat dan instan dalam kehidupan sehari-hari namun tanpa kita sadari proses suatu pengembangan pada dasarnya dapat menimbulkan efek samping dalam hal ini adalah kesehatan. Oleh karena itu, perubahan pola hidup yang terjadi akibat mengembangkan segalanya jadi lebih instan tersebut tidak dapat dikesampingkan akan timbulnya penyakit pada manusia salah satunya *Cerebrovascular Accident (CVA)* yang lazimnya disebut stroke. Seiring berkembangnya teknologi di dunia ini, banyak sekali industri yang mengembangkan robot berdasarkan kontribusinya di bidang-bidang tertentu, terutama bidang kesehatan dan teknologi. Salah satunya robot rehabilitasi stroke pada anggota tubuh contohnya lengan. Pada tahun 2017 telah ada pengembangan robot rehabilitasi lengan bagi penderita stroke dengan desain alat memudahkan pengguna robot serta bentuk fisik yang lebih kompatibel. Namun robot rehabilitasi yang telah di buat memerlukan pengembangan.

Penelitian ini dilakukan untuk mengembangkan robot rehabilitasi lengan tahun 2017 pada bagian mekanik serta sistem internet of things. Berdasarkan tujuan mengembangkan pada bagian mekanik, pengembangan dilakukan dengan mengganti motor power window menjadi motor servo. Pada bagian sistem internet of things pengembangan yang dilakukan adalah robot dapat mengirimkan data penggunaan robot agar dapat dikontrol melalui *smartphone*.

Berdasarkan hasil penelitian ini dapat di simpulkan desain dengan menggunakan motor servo dapat mengoptimalkan pergerakan motor, pada sistem Internet of thing pengamatan data penggunaan robot lebih mudah diamati, serta dapat dikontrol dari jarak jauh.

DEVELOPMENT OF AN ARM REHABILITATION ROBOT FOR STROKE PATIENTS BASED ON INTERNET OF THINGS USING A SMARTPHONE

SUMMARY

The lifestyle of an increasingly modern society has demanded the desire to solve a problem quickly and instantaneously in everyday life, but without realizing it, the process of a development can basically cause side effects, in this case health. Therefore, changes in lifestyle that occur as a result of developing everything so instantaneously cannot be ruled out for the emergence of disease in humans, one of which is the Cerebrovascular Accident (CVA) which is commonly called a stroke. As technology develops in this world, many industries are developing robots based on its contribution in certain fields, especially in the field of health and technology. One of them is a stroke rehabilitation robot on the limb, for example the arm. In 2017 there has been a development of an arm rehabilitation robot for stroke sufferers with a device design that makes it easier for robot users and a more compatible physical form. However, the rehabilitation robot that has been created requires development.

Based on the results of this study, it can be concluded that the design using a servo motor can optimize the movement of the motor, on the Internet of Things system the observation of data using robots is easier to observe, and can be controlled remotely. Based on the results of this study, it can be concluded that the design using a linear motor can smooth the motor movement, in the Internet of thing system, data observation of robot use is easier to observe, and in the PID section the controller can control motor movement to make it more stable.

Based on the results of this study, it can be concluded that the design using a servo motor can optimize the movement of the motor, on the Internet of Things system the observation of data using robots is easier to observe, and can be controlled remotely.

BAB I PENDAHULUAN

1.1 Latar Belakang

Stroke merupakan masalah kesehatan yang utama bagi masyarakat modern saat ini. Dewasa ini, stroke semakin menjadi masalah serius yang dihadapi hampir diseluruh dunia. Hal tersebut dikarenakan serangan stroke yang mendadak dapat mengakibatkan kematian, kecacatan fisik dan mental baik pada usia produktif maupun usia lanjut (Junaidi, 2011).

Pada umumnya penderita stroke akan menjadi bergantung pada bantuan orang lain dalam menjalankan aktivitas kehidupannya sehari-hari (*activities of daily living/ADL*) seperti makan dan minum, mandi, berpakaian dan sebagainya. Kemandirian dan mobilitas seseorang yang menderita stroke menjadi berkurang atau bahkan hilang. Berkurangnya tingkat kemandirian dan mobilitas seseorang dapat berpengaruh terhadap kualitas hidup yang dimiliki (Hariandja, 2013). Salah satu upaya untuk meningkatkan kualitas hidup penderita stroke adalah melalui rehabilitasi (Wirawan, 2009).

Seiring berkembangnya teknologi di dunia ini, banyak sekali industri yang mengembangkan robot berdasarkan kontribusinya di bidang-bidang tertentu, terutama bidang kesehatan dan teknologi. Terapi stroke dilakukan dengan bantuan robot untuk menggerakkan bagian tubuh pasien penderita stroke misalkan pada bagian tangan. Karena tingginya biaya pengadaan

robot rehabilitasi maka saat ini penggunaan robot untuk rehabilitasi hanya dapat ditemui di rumah sakit mewah atau pusat rehabilitasi saja, namun ini cukup mendapat respon besar bagi sebahagian orang, kerana proses rehabilitasi dengan menggunakan robot ternyata dapat mempercepat proses pemulihan yang biasanya membutuhkan waktu yang sangat lama, kebanyakan pemulihan yang dilakukan dengan menggunakan jasa psikoterapi terjadi selama 3 bulan sampai dengan 6 bulan pertama tetapi akan berlanjutan terus sampai 2 tahun atau bahkan lebih. Oleh karena itu, penelitian ini akan berusaha mengidentifikasi peluang desain dengan sistem rehabilitasi berbasis teknologi yang terjangkau (*low-cost*) untuk penderita stroke. Dimana hasil dari penelitian ini dapat membuat robot rehabilitasi bagi penderita pasca stroke khusus pada bagian tangan pasien yang komponennya terbuat dari bahan yang terjangkau dengan bentuk fisik yang kompatibel sehingga bukan hanya dapat digunakan di rumah sakit atau pusat rehabilitasi saja tapi dapat juga digunakan bagi pasien pasca stroke yang ingin berlatih di rumah sendiri.

Pada tugas akhir sebelumnya yaitu “Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things*” penulis menambahkan sistem Internet of Things (IoT) menggunakan *website*. *IoT* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan,

elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. Jadi robot rehabilitasi lengan ini dapat dikontrol lewat *website*. Serta menambahkan PID kontroler, PID adalah mekanisme *loop* kontrol yang menggunakan umpan balik yang banyak digunakan dalam sistem kontrol industri dan berbagai aplikasi lain yang membutuhkan kontrol termodulasi terus menerus.

Pada pengembangan selanjutnya yaitu “Pengembangan Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things* Menggunakan *Smartphone*” penulis akan memanfaatkan *smartphone* sebagai media untuk memantau dan mengontrol robot menggunakan aplikasi berbasis *Internet of Things (IoT)*. Selanjutnya yaitu mengubah penggunaan motor *DC* menjadi motor servo agar mekanismenya menjadi lebih efektif dimana yang awalnya menggunakan *limit switch* untuk menentukan posisi minimum dan maksimum pergerakan akan diubah menjadi pemanfaatan motor servo agar derajat pergerakannya bisa diatur untuk menentukan posisi minimum dan maksimum pergerakan.

1.2 Rumusan Masalah

Dari uraian latar belakang masalah, maka didapatkan rumusan masalah seperti berikut:

1. Bagaimana mengoptimalkan kecepatan motor linear pada robot rehabilitasi lengan ($v = 12 - 20 \text{ mm/s}$)?

2. Bagaimana mengontrol pergerakan horizontal robot agar menjadi lebih efektif menggunakan motor servo?
3. Bagaimana merancang aplikasi smartphone pada robot rehabilitasi lengan berbasis *Internet of Things*?

1.3 Ruang Lingkup Penelitian

Dalam pembuatan tugas akhir ini ada beberapa hal yang akan dibatasi yaitu:

1. Menggerakkan anggota tubuh bagian tangan.
2. Gerakan yang dihasilkan oleh robot 3DOF, yaitu 1DOF untuk gerak fleksi (menekukkan tangan) dan gerak ekstensi (meluruskan tangan) pada siku, 2DOF untuk gerak adduksi (mendekati tubuh) dan gerak abduksi (menjauhi tubuh) pada tangan yaitu merenggangkan tangan kedepan dan kesamping serta keatas dan kebawah.
3. Pengontrolan gerakan terdiri dari 2 mode yaitu lambat dan sedang
4. Pengoperasian dipermudah dengan menggunakan *Internet of Things* melalui *smartphone*.
5. Pasien yang dapat melakukan reahabilitasi adalah pasien pasca stroke bagian lengan.

1.4 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam pembuatan dan perancangan ini adalah:

1. Untuk mengoptimalkan kecepatan motor linear pada robot rehabilitasi lengan ($v = 12 - 20\text{mm/s}$).

2. Untuk mengontrol pergerakan horizontal robot agar menjadi lebih efektif menggunakan motor servo.
3. Untuk menghasilkan aplikasi smartphone yang dapat memantau dan mengontrol robot rehabilitasi lengan berbasis *Internet of Things*.

1.5 Manfaat Penelitian

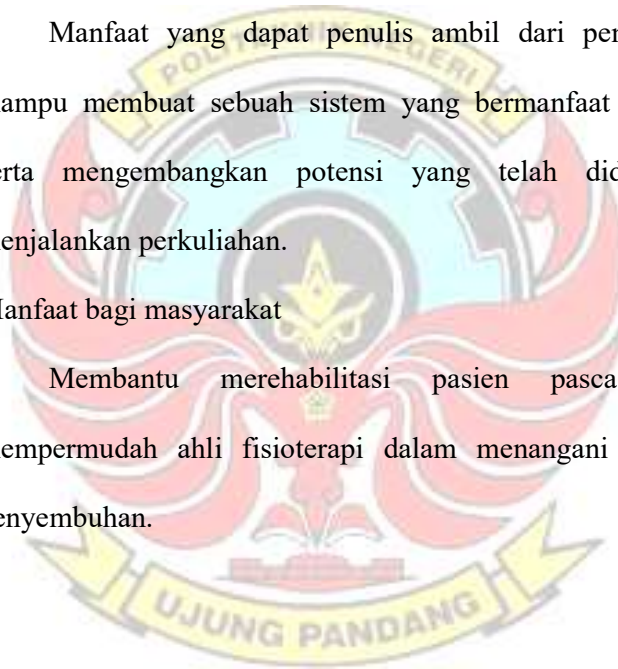
Adapun manfaat yang dapat diperoleh dari penelitian ini antara lain:

1. Manfaat bagi penulis

Manfaat yang dapat penulis ambil dari penelitian ini yaitu mampu membuat sebuah sistem yang bermanfaat bagi masyarakat serta mengembangkan potensi yang telah didapatkan selama menjalankan perkuliahan.

2. Manfaat bagi masyarakat

Membantu merehabilitasi pasien pasca stroke serta mempermudah ahli fisioterapi dalam menangani pasien di masa penyembuhan.



BAB II TINJAUAN PUSTAKA

2.1 Stroke

Stroke atau cedera *cerebrovaskuler* adalah kehilangan fungsi otak yang diakibatkan oleh berhentinya suplai darah ke bagian otak sering ini adalah kulminasi penyakit serebrovaskuler selama beberapa tahun. (Smeltzer C. S. dkk., 2002).

Menurut WHO stroke adalah adanya tanda-tanda klinik yang berkembang cepat akibat gangguan fungsi otak fokal (atau global) dengan gejala-gejala yang berlangsung selama 24 jam atau lebih yang menyebabkan kematian tanpa adanya penyebab lain yang jelas selain vaskuler. (Susilo, 2000)



Gambar 2. 1 Penyebab Stroke
(Sumber: Indozone, 2020)

Berdasarkan patologi anatomi, stroke dibagi menjadi dua yaitu stroke *iskemik* dan stroke *hemoragik* yang dapat dilihat pada gambar 2.2. Stroke *iskemik* terjadi ketika arteri yang membawa darah dan oksigen ke otak menyempit, menyebabkan penurunan aliran darah ke otak. Kondisi ini juga

dikenal sebagai *iskemia*. *Iskemik* stroke dapat dibagi lagi menjadi 2 jenis, stroke *trombotik* dan stroke *emboli*. Sedangkan, Stroke *hemoragik* terjadi saat pembuluh darah di otak pecah dan menyebabkan perdarahan. Pendarahan di otak dapat dipicu oleh kondisi yang memengaruhi peredaran darah. Kondisi tersebut termasuk hipertensi yang tidak terkendali, melemahnya dinding pembuluh darah, dan pengobatan dengan pengencer darah. Stroke *hemoragik* terdiri dari dua jenis, yaitu perdarahan *intracerebral* dan *subarachnoid*.



Gambar 2. 2 Stroke *Hemoragik* dan Stroke *Iskemik*
(Sumber: Syah, 2014)

2.2 Rehabilitasi Medik

Mengenai sejarah singkat rehabilitasi medis, menurut data yang tersedia di *Department of Physical Medicine and Rehabilitation*, Mayo Clinic, Rochester, Amerika Serikat, pada tahun 1916 terdapat wabah polio yang menyerang New York. Wabah tersebut dapat mengakibatkan kecacatan sementara bahkan seumur hidup jika tidak cepat ditangani, maka dibentuklah *Georgia Warm Springs Young Foundation* pada 1924 sebagai tanggapan terhadap wabah polio ini untuk menanggulangi akibat buruk yang ditimbulkan (Dalgas dkk, 2008).

Salah satu penerapan rehabilitasi yaitu pada penyakit stroke, dengan demikian pemulihan fungsi alat gerak (rehabilitasi) yang dijalani pasien polio itulah titik awal yang mendorong berdirinya rehabilitasi medis Sebagai upaya mengembalikan kemampuan motorik dan meningkatkan kualitas hidup para penderita stroke. Maka rehabilitasi pasca stroke yang dini secara teratur dapat mengembalikan kemampuan motorik secara bertahap hingga kesehatan mereka dapat pulih kembali secara total (Bastian, 2011).

Rehabilitasi stroke merupakan sebuah program yang terkoordinasi yang memberikan perawatan restoratif untuk memaksimalkan pemulihan dan meminimalisasi *impairment*, *disability*, dan *hadicap* yang disebabkan oleh stroke (Widiyanto, 2009). *Disability* atau ketidakmampuan didefinisikan sebagai keterbatasan atau hilangnya kemampuan untuk melakukan aktivitas yang umum dilakukan orang normal akibat *impairment* yang dideritanya (Wirawan, 2009).

Menurut Wirawan (2009), terdapat 6 prinsip dasar pada rehabilitasi stroke sebagai berikut:

1. Gerak merupakan obat yang paling mujarab.
2. Latihan yang digunakan pada terapi gerak sebaik merupakan gerak fungsional.
3. Pasien diarahkan untuk melakukan gerak dengan keadaan senormal mungkin.
4. Latihan gerak fungsional dapat dilakukan setelah stabilitas tubuh sudah tercapai.

5. Terapi gerak diberikan kepada pasien yang siap secara fisik maupun mental.
6. Hasil terapi akan optimal jika ditunjang dengan kemampuan fungsi kognitif, persepsi, dan modalitas sensoris yang baik.

2.3 Robot Rehabilitasi Stroke

Salah satu peneliti asal Inggris *Daily Mail* melakukan percobaan pertamanya di *Northumbria Healthcare NHS Foundation Trust* dan Universitas Newcastle untuk menguji coba robot rehabilitasi bagi pasien penderita stroke. Dalam uji coba, pasien stroke duduk di depan meja menghadap layar komputer dan menempatkan tangan mereka pada robot. Instruktur tersebut kemudian meminta mereka melakukan beberapa gerakan latihan, seperti menggerakkan target yang ada di layar komputer melalui alat yang digenggamnya.



Gambar 2. 3 Robot Rehabilitasi
(Sumber: Marianjoy, 2021)

Jika pasien tidak dapat menggerakkan tangan mereka, maka robot yang akan menggerakkannya. Jika pasien mulai dapat bergerak, robot akan mendampingi pasien, tujuannya untuk membantu otak dan tangan pasien kembali dapat bekerja bersama-sama, dan hasil dari uji coba tersebut kepada seorang pasien, kemudian muncul pendapat yang menggembarakan bahwa ketika mengalami stroke setiap gerakan yang dapat dilakukan adalah perubahan besar.

Upper Extremity studi penelitian ini diperoleh dari hasil wawancara langsung di rumah sakit Universitas Hasanuddin Makassar oleh Dr. Husnul Mubarak salah satu dokter ahli fisioterapi. (Rahmat, 2016)

Tabel 2. 1 *Upper Extremity* untuk *Flexi* dan *Extensi*

<i>Upper Extremity</i>	
<i>Shoulder</i>	<i>Elbow</i>
<i>Flexion 0° – 120°</i>	<i>Extension 0°</i>
<i>Extension 0° – 45°</i>	<i>Flexion 0° – 120°</i>

Tabel 2. 2 *Upper Extremity* untuk Horizontal *Adduksi* dan *Abduksi*

<i>Upper Extremity Horizontal</i>	
<i>Shoulder</i>	<i>Elbow</i>
<i>Full Adduksi 80°</i>	<i>Abduksi 100°</i>

2.4 *Internet of Things (IoT)*

Internet of Things, atau dikenal juga dengan singkatan *IoT*, merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas

internet yang terhubung secara terus-menerus. Kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Bahan pangan, elektronik, koleksi, peralatan apa saja, termasuk benda hidup yang terhubung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif. Pada aktual, *Internet of Things* mengacu pada benda yang dapat diidentifikasi secara unik sebagai representasi virtual dalam struktur berbasis Internet. Istilah *Internet of Things* berdasarkan permintaan oleh Kevin Ashton pada tahun 1999 dan mulai terkenal melalui Auto-ID Center di MIT.

Casagras (Koordinasi dan aksi dukungan untuk aktivitas dan standarisasi global terkait RFID) mendefinisikan *Internet of Things*, sebagai sebuah infrastruktur jaringan global, yang menghubungkan benda-benda fisik dan virtual melalui eksploitasi data *capture* dan kemampuan komunikasi. Infrastruktur terdiri dari jaringan yang telah ada dan internet berikut pengembangan jaringannya. Semua ini akan menawarkan layanan, sensor dan kemampuan koneksi sebagai dasar untuk pengembangan layanan dan aplikasi kooperatif yang independen. Ia juga berhubungan dengan tingkat otonom data capture yang tinggi, transfer acara, konektivitas jaringan dan interoperabilitas.

SAP (*Systeme, Anwendungen und Produkte*) mendefinisikannya sbb: Dunia di mana benda-benda fisik yang diintegrasikan ke dalam jaringan informasi secara berkesinambungan, dan di mana benda-benda fisik tersebut berperan aktif dalam proses bisnis. Layanan yang tersedia dengan obyek

pintar melalui Internet, mencari dan mengubah status mereka sesuai dengan setiap informasi yang peduli, disamping memperhatikan masalah privasi dan keamanan.

ETP EPOSS mendefinisikannya sebagai Jaringan yang dibentuk oleh hal-hal atau benda yang memiliki identitas, pada dunia maya yang beroperasi di ruang itu dengan menggunakan kecerdasan antarmuka untuk terhubung dan berkomunikasi dengan pengguna, konteks sosial dan lingkungan.

2.4.1 Cara Kerja Internet of Things

IoT bekerja dengan memanfaatkan suatu argumentasi pemrograman, dimana tiap-tiap perintah argumen tersebut bisa menghasilkan suatu interaksi antar mesin yang telah terhubung secara otomatis tanpa campur tangan manusia dan tanpa terbatas jarak berapapun jauhnya. Jadi, Internet di sini menjadi penghubung antara kedua interaksi mesin tersebut. Lalu di mana campur tangan manusia? Manusia dalam *IoT* tugasnya hanyalah menjadi pengatur dan pengawas dari mesin-mesin yang bekerja secara langsung tersebut.

2.4.2 Manfaat Internet of Things

Beberapa manfaat *IoT* mungkin tidak terlalu kelihatan, tetapi bukan berarti tidak bisa dirasakan. Di bawah ini adalah tiga manfaat utama yang akan kamu dapatkan langsung dari *IoT*:

1. Konektivitas

Di era digital ini, kamu bisa mengucapkan selamat tinggal pada era pengoperasian perangkat secara manual. Dengan *IoT*, kamu bisa mengoperasikan banyak hal dari satu perangkat, misalnya *smartphone*.

2. Efisiensi

Dengan adanya peningkatan pada konektivitas, berarti terdapat penurunan jumlah waktu yang biasanya dihabiskan untuk melakukan tugas yang sama. Misalnya, asisten suara seperti Apple's Homepod atau Amazon's Alexa dapat memberikan jawaban atas pertanyaan tanpa kamu perlu mengangkat telepon atau menghidupkan komputer.

3. Kemudahan

Perangkat *IoT* seperti *smartphone* kini mulai menjadi perangkat yang biasa dimiliki oleh sebagian besar orang.

2.5 Komponen Mekanik

2.5.1 Motor *Linear*

Linear motor adalah motor listrik yang stator dan rotornya berbentuk linier. Jadi motor ini berbentuk linier datar, tidak bulat tabung seperti motor pada umumnya. Karena bentuknya yang linier, motor ini tidak menghasilkan torsi (putaran) tetapi menghasilkan gaya yang linier. Umumnya, motor linier dibuat dengan prinsip hukum

Lorentz, yakni arus listrik yang mengalir pada medan magnet akan terkena gaya Lorentz yang arahnya tegak lurus.



Gambar 2. 4 Motor Linear
(Sumber: Dickson, 2020)

Prinsip kerja aktuator linear listrik adalah desain modular yang mengintegrasikan motor dan sekrup, mengubah gerakan putar poros motor menjadi gerakan linear batang piston aktuator linear listrik. Sementara itu, aktuator linear listrik servo mengubah keunggulan motor servo kecepatan, rotasi, kontrol torsi menjadi kecepatan, posisi, kontrol dorong yang tepat untuk mewujudkan produk revolusioner gerak linear presisi tinggi.

2.5.2 Motor *Direct Current (dc)* Power Window

Motor *dc* (Power Window) adalah suatu motor yang mengubah energi listrik searah menjadi mekanik yang berupa tenaga penggerak torsi. Motor *dc* digunakan dimana kecepatan kontrol dan kecepatan torsi diperlukan untuk memenuhi kebutuhan. Bagian *dc* yang paling penting adalah rotor dan strator. Bagian stator adalah badan motor,

sikat-sikat dan inti kutub magnet. Bagian rotor adalah bagian yang berputar dari suatu motor *dc*. Yang termasuk rotor sedang lilitan jangkar, komutator, tali, isolator, poros, bantalan dan kipas.



Gambar 2. 5 Motor *dc* Power Window
(Sumber: Yanik, 2016)

Adapun perhitungan torsi untuk mekanisme pergerakan motor yaitu:

1. Perhitungan torsi motor untuk mekanisme naik-turun

Telah diketahui bahwa:

$$T = \frac{P}{\omega} \dots\dots\dots(1)$$

Dalam hal ini:

T = Torsi [Nm]

P = Daya [Watt]

ω = Putaran [rad/s]

Jika putaran ω dinyatakan dalam rpm (revolution per minute), maka bentuk persamaan (1) akan berubah. Misalkan dari persamaan (1) :

$$T = k \frac{P}{\omega}$$

Dalam hal ini:

$$k = 1 \frac{\text{Nm.rad/s}}{\text{W}} \dots\dots\dots(2)$$

Tetapi $1 \text{ rad/s} = \frac{30}{\pi} \text{ rpm}$ sehingga persamaan (2) menjadi:

$$k = 1 \frac{\text{Nm} \frac{30}{\pi} \text{rpm}}{\text{W}} = \frac{30}{\pi} \frac{\text{Nm.rpm}}{\text{W}}$$

Jadi faktor yang harus diperkalikan terhadap persamaan (1) adalah $\frac{30}{\pi}$ sehingga berubah menjadi:

$$T = \frac{30.P}{\pi.n} \dots\dots\dots(3)$$

Dalam hal ini:

T = Torsi [Nm]

P = Daya [Watt]

n = Putaran [rpm]

2. Perhitungan torsi motor untuk mekanisme maju-mundur dan kiri-kanan

$$T = F.r \dots\dots\dots(4)$$

Dalam hal ini:

T = Torsi [Nm]

F = Gaya [N]

r = Jarak Perpindahan [m]

2.5.3 Motor Servo

Motor servo adalah motor dengan sistem umpan balik tertutup dimana posisi motor akan diinformasikan kembali ke rangkaian kontrol yang ada di motor servo. Motor ini terdiri dari motor *dc*, serangkaian roda gigi, potensiometer, dan sirkuit kontrol. Potensiometer berfungsi untuk menentukan batas sudut rotasi servo. Sedangkan sudut sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal kabel motor.



Gambar 2. 6 Motor Servo
(Sumber: Alexnld, 2021)

2.6 Komponen Elektronik

Komponen elektronik yang dibutuhkan dalam merancang robot rehabilitasi lengan, yaitu:

2.6.1 Arduino Mega 2560

Arduino Mega 2560 adalah suatu papan sirkuit dengan mikrokontroler chip ATmega2560 serta memiliki jumlah pin paling banyak di antara semua jenis Arduino lainnya.



Gambar 2. 7 *Board* Arduino Mega 2560
(Sumber: Arduino Home Page, 2012)

Tabel 2. 3 Deskripsi Arduino Mega 2560

Mikrokontroler	ATmega2560
Tegangan Operasi	5V
Input Voltage (dianjurkan)	7-12V
Input Voltage (batas)	6-20V
Digital (I/O) Pins	54 (dari yang 15 memberikan output PWM)
Pins Masukan Analog	16
dc Current per I/O Pin	20 mA
dc Saat ini untuk 3.3V Pin	50 mA
Flash Memory	256 KB yang 8KB digunakan oleh bootloader
SRAM	8 KB
EEPROM	4 KB
Kecepatan jam	16 MHz
Panjangnya	101,52 mm
Lebar	53,3 mm
Berat	37 g

Arduino dapat diberikan *power* melalui koneksi *USB* atau *power supply*. *Powernya* masuk secara otomatis. *Power supply* dapat menggunakan adaptor *dc* atau baterai. Adaptor dapat dikoneksikan dengan mencolok *jack* adaptor pada koneksi *port input supply*. *Board* arduino dapat dioperasikan menggunakan *supply* dari luar sebesar 6 - 20 volt. Jika *supply* kurang dari 7V, kadangkala pin 5V akan menyuplai kurang dari 5V dan

board bisa menjadi tidak stabil. Jika menggunakan lebih dari 12V, tegangan di regulator bisa menjadi sangat panas dan menyebabkan kerusakan pada *board*. Rekomendasi tegangan ada pada 7 sampai 12 Volt. (Arduino Home Page, 2012).

Penjelasan pada pin power adalah sebagai berikut:

1. Volume input (V_{in}), yaitu tegangan *input* ke *board* arduino ketika menggunakan tegangan dari luar (seperti yang disebutkan 5volt dari koneksi USB atau tegangan yang diregulasikan). Pengguna dapat memberikan tegangan melalui pin ini, atau jika tegangan suplai menggunakan *power jack*, aksesnya menggunakan pin ini.
2. 5 Volt, yaitu regulasi *power supply* digunakan untuk *power* mikrokontroler dan komponen lainnya pada *board*. 5V dapat melalui V_{in} menggunakan regulator pada *board*, atau *supply* oleh USB atau *supply* regulasi 5V lainnya.
3. 3V3, yaitu suplai 3.3volt didapat oleh *FTDI chip* yang ada di *board*. Arus maksimumnya adalah 50mA.
4. Pin Ground, berfungsi sebagai jalur *ground* atau jalur negatif pada arduino.
5. Memori, ATmega328 memiliki 32 KB *flash* memori untuk menyimpan kode, juga 2 KB yang digunakan untuk *bootloader*. ATmega328 memiliki 2 KB untuk SRAM dan 1 KB untuk EEPROM.

Masing-masing dari 54 pin digital pada Mega dapat digunakan sebagai *input* atau *output*, menggunakan *pinMode ()*, *digitalWrite ()*, dan *digitalRead ()*. Mereka beroperasi di 5 Volt. Setiap pin dapat memberikan atau menerima 20 mA sebagai kondisi operasi yang direkomendasikan dan memiliki resistor *pull-up* internal yang (terputus secara *default*) dari 20-50k ohm. Maksimal 40mA adalah nilai yang tidak boleh melebihi untuk menghindari kerusakan permanen ke mikrokontroler. (*Arduino Home Page, 2012*).

Arduino memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lain. ATmega328 ini menyediakan *UART TTL (5V)* komunikasi serial, yang tersedia pada pin digital 0 (RX) dan 1 (TX). *Firmware* Arduino menggunakan *USB driver* standar *COM*, dan tidak ada *driver* eksternal yang dibutuhkan. Namun, pada *Windows, file*. Ini diperlukan. Perangkat lunak arduino termasuk monitor serial yang memungkinkan data sederhana yang akan dikirim ke *board* Arduino. *RX* dan *TX LED* di *board* akan berkedip ketika data sedang dikirim melalui *chip USB-to-serial* dan koneksi *USB* ke komputer.

ArduinoMEGA dapat diprogram dengan perangkat lunak Arduino. Pada ATmega2560 di Arduino terdapat *bootloader* yang memungkinkan Anda untuk meng-*upload* kode baru tanpa menggunakan *programmer hardware eksternal*. IDE Arduino adalah

software yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

1. *Editor program*, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam *memory* di dalam papan Arduino (*Arduino Home Page, 2012*).

2.6.2 NodeMCU

NodeMCU pada dasarnya adalah pengembangan ESP8266 dengan *firmware* berbasis e-Lua. On NodeMcu dilengkapi dengan *port microUSB* yang berfungsi untuk pemrograman juga catu daya. Selain itu, NodeMCU juga dilengkapi dengan tombol push yaitu tombol *reset* dan *flash*. NodeMCU menggunakan bahasa pemrograman Lua yang merupakan paket dari esp8266. Bahasa Lua memiliki logika dan struktur pemrograman yang sama dengan bahasa c, hanya dengan sintaks yang berbeda. Jika Anda menggunakan Lua, Anda dapat menggunakan Lua loader atau uploder Lua.

Selain bahasa Lua NodeMCU juga mendukung perangkat lunak Arduino IDE dengan membuat beberapa perubahan pada manajemen pada Arduino IDE.



Gambar 2. 8 NodeMCU
(Sumber: Stans, 2017)

Berikut adalah spesifikasi NodeMCU V3 yang digunakan dalam proyek ini:

1. Menggunakan CH340G bukan CP2102.
2. NodeMCU memiliki serial USB-TTL built-in dengan kekuatan industri super andal CH340G untuk stabilitas unggul pada semua platform yang didukung.
3. Tegangan antarmuka komunikasi: 3.3V.
4. Jenis antena: Tersedia antena PCB internal.
5. Nirkabel 802.11 b / g / n standar
6. WiFi di 2.4GHz, mendukung mode keamanan WPA / WPA2
7. Mendukung tiga mode pengoperasian STA / AP / STA + AP
8. Tumpukan protokol TCP / IP bawaan untuk mendukung beberapa koneksi TCP Client (5 MAX)

9. D0 ~ D8, SD1 ~ SD3: digunakan sebagai GPIO, PWM, IIC, dll.,
Kemampuan driver port 15mA
10. AD0: 1 saluran ADC
11. Input daya: 4.5V ~ 9V (10V MAX), bertenaga USB
12. Sekarang: transmisi kontinyu: 70mA (200mA MAX), Siaga:
200uA
13. Kecepatan transfer: 110-460800bps
14. Mendukung antarmuka komunikasi data UART / GPIO
15. Peningkatan *firmware* jarak jauh (OTA)
16. Mendukung *Smart Link Smart Networking*
17. Temperatur kerja: -40 ~ + 125
18. Tipe *Drive*: *Driver* H-bridge ganda berdaya tinggi
19. Ukuran *flash*: 4Mbyte

2.6.3 Adaptor 12V

Hampir semua rangkaian elektronik membutuhkan sumber tegangan *dc* yang teratur. Pencatuan ini dapat dilakukan secara langsung oleh baterai, namun yang lebih umum catu daya yang diperoleh dari sumber *ac* standar yang kemudian diubah menjadi tegangan *dc*.



Gambar 2. 9 Adaptor 12V
(Sumber: Info Panduan Trik, 2015)

2.6.4 Push Button

Push Button adalah saklar tekan yang berfungsi untuk menghubungkan atau memisahkan bagian – bagian dari suatu instalasi listrik satu sama lain. Suatu sistem saklar tekan *push button* terdiri dari saklar tekan *start*, *Stop reset* dan saklar tekan untuk *emergency*. *Push button* memiliki kontak NC (*normally close*) dan NO (*normally open*).

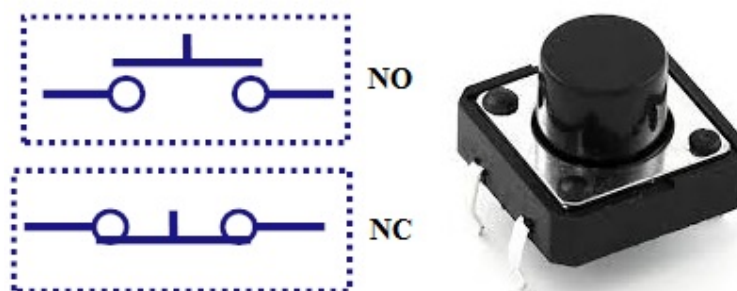
Prinsip kerja Push Button adalah apabila dalam keadaan normal tidak ditekan maka kontak tidak berubah, apabila ditekan maka kontak NC akan berfungsi sebagai *stop* (memberhentikan) dan kontak NO akan berfungsi sebagai *start* (menjalankan) biasanya digunakan pada sistem pengontrolan motor – motor induksi untuk menjalankan mematikan motor pada industri – industri. Pada pembuatan alat ini digunakan 2 buah Tipe *Normally Open* (NO).

Tombol ini disebut juga dengan tombol start karena kontak akan menutup pintu bila ditekan dan kembali terbuka bila menekan tombol

yang satu lagi. Bila tombol ditekan maka kontak bergerak akan menyentuh kontak tetap sehingga arus listrik akan mengalir.

Berdasarkan fungsi kerjanya yang menghubungkan dan memutuskan, push button switch mempunyai 2 tipe kontak yaitu NC (Normally Close) dan NO (Normally Open).

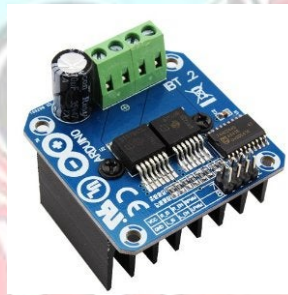
1. NO (*Normally Open*), merupakan kontak terminal dimana kondisi normalnya terbuka (aliran arus listrik tidak mengalir). Dan ketika tombol saklar ditekan, kontak yang NO ini akan menjadi menutup (*Close*) dan mengalirkan atau menghubungkan arus listrik. Kontak NO digunakan sebagai penghubung atau menyalakan sistem circuit (*Push Button ON*).
2. NC (*Normally Close*), merupakan kontak terminal dimana kondisi normalnya tertutup (mengalirkan arus listrik). Dan ketika tombol saklar push button ditekan, kontak NC ini akan menjadi membuka (*Open*), sehingga memutus aliran arus listrik. Kontak NC digunakan sebagai pemutus atau mematikan sistem circuit (*Push Button Off*).



Gambar 2. 10 Push Button
Sumber: (Kelas PLC, 2021)

2.6.5 Driver Motor

Driver motor L298N merupakan *module driver* motor *dc* yang paling banyak digunakan atau dipakai di dunia elektronika yang difungsikan untuk mengontrol kecepatan serta arah perputaran motor *dc*. IC L298 merupakan sebuah IC tipe H-bridge yang mampu mengendalikan beban-beban induktif seperti relay, solenoid, motor *dc* dan motor stepper. Pada IC L298 terdiri dari transistor-transistor logik (TTL) dengan gerbang nand yang berfungsi untuk memudahkandalam menentukan arah putaran motor *dc* maupun motor stepper.



Gambar 2. 11 BTS 7960
(Sumber: [F1 Depo](#), 2018)

BTS7960 adalah modul jembatan H arus tinggi terintegrasi penuh untuk aplikasi penggerak motor. Antarmuka ke *mikrokontroler* dipermudah oleh IC *driver* terintegrasi yang menampilkan input level logika, diagnosis dengan sensor arus, penyesuaian laju perubahan tegangan, pembangkitan waktu mati, dan perlindungan terhadap suhu berlebih, tegangan berlebih, tegangan kurang, arus berlebih, dan korsleting. BTS7960 memberikan solusi biaya optimal untuk

penggerak motor PWM arus tinggi terlindung dengan konsumsi ruang papan yang sangat rendah.

Driver yang digunakan adalah *driver* dengan *input* arus 12V *dc*, berfungsi sebagai pembalik arah putaran motor *dc* sekaligus mengatur kecepatan putaran motor *dc*. Dimana polaritas *input* tegangan motor *dc* dibalik melalui *IC* yang terdapat pada papan kontrol *driver* tersebut dengan bantuan kontroler Arduino Mega 2560.

2.6.6 *Liquid Crystal Display* (LCD)

LCD adalah media tampilan yang paling mudah untuk ditonton karena menghasilkan tampilan karakter yang baik dan cukup banyak. Pada 20×4 LCD 80 karakter dapat ditampilkan, 20 karakter di setiap baris sebanyak 4 baris. 20×4 LCD umumnya menggunakan 16 pin sebagai kontrol, tentu saja akan sangat boros ketika menggunakan 16 pin ini. Oleh karena itu, driver khusus digunakan sehingga LCD dapat dikontrol dengan jalur I2C. melalui I2C, LCD dapat dikontrol hanya menggunakan 2 pin, yaitu SDA dan SCL.



Gambar 2. 12 *Liquid Crystal Display* (LCD)
(Sumber: Sinau Programming, 2020)

2.6.7 Keypad

Keypad merupakan antarmuka antara komunikasi perangkat elektronik dengan manusia yang disebut istilah HMI (*Human Machine Interface*). Dalam penelitian ini *keypad* digunakan untuk meng-input data dalam hal ini yaitu kecepatan motor, jenis gerakan, dan jumlah gerakan.



Gambar 2. 13 Keypad 4x4
(Sumber: Ardutech, 2019)

2.6.8 MIT App Inventor

MIT App Inventor merupakan platform untuk memudahkan proses pembuatan aplikasi sederhana tanpa harus mempelajari atau menggunakan bahasa pemrograman yang terlalu banyak. Kita dapat mendesain aplikasi android sesuai keinginan dengan menggunakan berbagai macam layout dan komponen yang tersedia. Dengan app inventor, pengguna bisa melakukan pemrograman komputer untuk menciptakan aplikasi perangkat lunak dengan sistem operasi berbasis android. App inventor ini berbasis visual block programming karena memungkinkan pengguna bisa menggunakan, melihat, menyusun dan

men-drag and drops block yang merupakan simbol perintah dan fungsi event handler untuk menciptakan sebuah aplikasi yang bisa berjalan di sistem android.

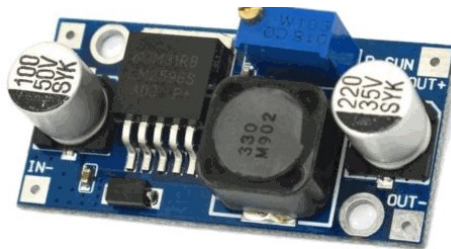


Gambar 2. 14 MIT App Inventor
(Sumber: Psti Unisayogya, 2020)

2.6.9 Step down

Fungsi *step down* adalah mengubah tegangan tinggi dengan arus rendah menjadi tegangan rendah dengan arus tinggi. Fungsi utama *step down* adalah menurunkan tegangan listrik dan menyesuaikannya dengan kebutuhan elektronika (Sendari, 2020).

Daya pada transformator diukur menggunakan produk dari tegangan dan arus. Daya pada transformator dinilai dalam Volt - Amps VA. Idealnya, daya pada setiap transformator adalah konstan, yaitu daya yang tersedia pada sekunder transformator sama dengan daya pada primer transformator.



Gambar 2. 15 Step Down
(Sumber: Hermawan, 2017)

2.7 Penelitian Sebelumnya

Dalam penulisan ini penulis mengumpulkan segala informasi dari referensi, literatur yang sesuai dengan topik dan menggunakan media internet sebagai bahan referensi tambahan dan penulis sebelumnya juga melakukan wawancara langsung dengan ahli fisioterapi oleh Dr. Husnul Mubarak di rumah sakit Universitas Hasanuddin Makassar mengenai stroke.

Penelitian pertama yaitu dari Abdul Rahmat dan Muh. Swahid Ismail pada tahun 2016 yang berjudul “Rancang Bangun Robot Rehabilitasi Lengan bagi Penderita Stroke” dalam penelitiannya menggunakan *input* analog untuk mengontrol gerakan lengan robot, perangkat tersebut dikoneksikan dengan mikrokontroler (Arduino mega) untuk memberi perintah ke lengan robot hasil pengujiannya adalah robot dapat bergerak mendekat dan menjauhi tubuh sesuai dari pemrograman yang di-*input* kan. ketika menyentuh *limit switch* untuk batas maksimum dan minimum, motor akan berhenti sejenak dan kembali bergerak berlawanan arah sampai perintah yang diberikan selesai untuk hasil robot ini telah berhasil dibuat dengan menggunakan komponen dan material yang ekonomis dan desain yang telah dibuat memiliki bentuk fisik yang kompatibel sehingga robot rehabilitasi ini bukan hanya dapat digunakan di rumah sakit atau pusat rehabilitasi saja, tapi dapat juga digunakan di rumah pasien. adapun kekurangan dari hasil rancang bangun kemarin, 3 Putaran motor *dc* dapat mengalami penurunan kecepatan dari yang ditentukan karna faktor berat lengan pasien yang akan di rehabilitasi.

Pada pengembangan pertama dari M. Agus dan Ikram pada tahun 2017 yang berjudul “Pengembangan Robot Rehabilitasi Lengan untuk Penderita Stroke” telah berhasil dikembangkan pada penggunaan robot rehabilitasi lengan ini sudah menggunakan interface berupa *LCD 20 x 4* dan *keypad* yang sangat membantu dalam pengoperasian alat. Pada bagian keamanannya, tersedia 4 pilihan tombol *stop*, *ON/OFF* Arduino, *Power Supply* yang mudah dilepas, dan tombol *Emergency*. Dalam pengontrolan gerakan, telah dibuatkan program yang dapat meng-*input* pengulangan gerakan sesuai keinginan pasien. Kekurangannya ada pada gerakan fleksi dan ekstensi sebaiknya menggunakan ulir dengan jarak *pitch* yang lebih besar agar gerakan bisa lebih cepat.

Selanjutnya dikembangkan lagi suatu robot rehabilitasi lengan (Agus dan Ikram, 2017). Ini adalah pengembangan pertama yang dilakukan oleh M. Agus dan Ikram pada tahun 2017 yang berjudul “Pengembangan Robot Rehabilitasi Lengan untuk Penderita Stroke” telah berhasil dikembangkan pada penggunaan robot rehabilitasi lengan ini sudah menggunakan *interface* berupa *LCD 20 x 4* dan *keypad* yang sangat membantu dalam pengoperasian alat. Pada bagian keamanannya, tersedia 4 pilihan tombol *stop*, *ON/OFF* Arduino, *Power Supply* yang mudah dilepas, dan tombol *Emergency*. Dalam pengontrolan gerakan, telah dibuatkan program yang dapat meng-*input* pengulangan gerakan sesuai keinginan pasien. Kekurangannya ada pada gerakan fleksi dan ekstensi sebaiknya menggunakan ulir dengan jarak *pitch* yang lebih besar agar gerakan bisa lebih cepat.

Pada pengembangan kedua dari Muh. Syafrie Afrizal dan Muhammad Fiqih Mahmuda pada tahun 2020 yang berjudul “Robot Rehabilitasi Lengan bagi Penderita Stroke Berbasis *Internet of Things*” penulis menambahkan sistem *Internet of Things (IoT)* menggunakan *website*. *IoT* merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus. Jadi robot rehabilitasi lengan ini dapat dikontrol lewat *website*. Serta menambahkan PID kontroler, PID adalah mekanisme *loop* kontrol yang menggunakan umpan balik yang banyak digunakan dalam sistem kontrol industri dan berbagai aplikasi lain yang membutuhkan kontrol termodulasi terus menerus.



BAB III METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Laboratorium Mekatronika Kelas 4 Teknik Mekatronika PNUP, Makassar, Sulawesi Selatan. Penelitian ini dilakukan pada bulan. Waktu pelaksanaan dimulai dari bulan Januari 2021 sampai dengan bulan Agustus 2021.

3.2 Alat dan Bahan Penelitian

Dalam pelaksanaannya penelitian ini memerlukan beberapa alat, bahan dan perangkat lunak guna merakit tugas akhir sesuai yang diharapkan, berikut adalah tabel dari alat dan bahan yang akan digunakan:

Tabel 3. 1 Alat yang Dibutuhkan

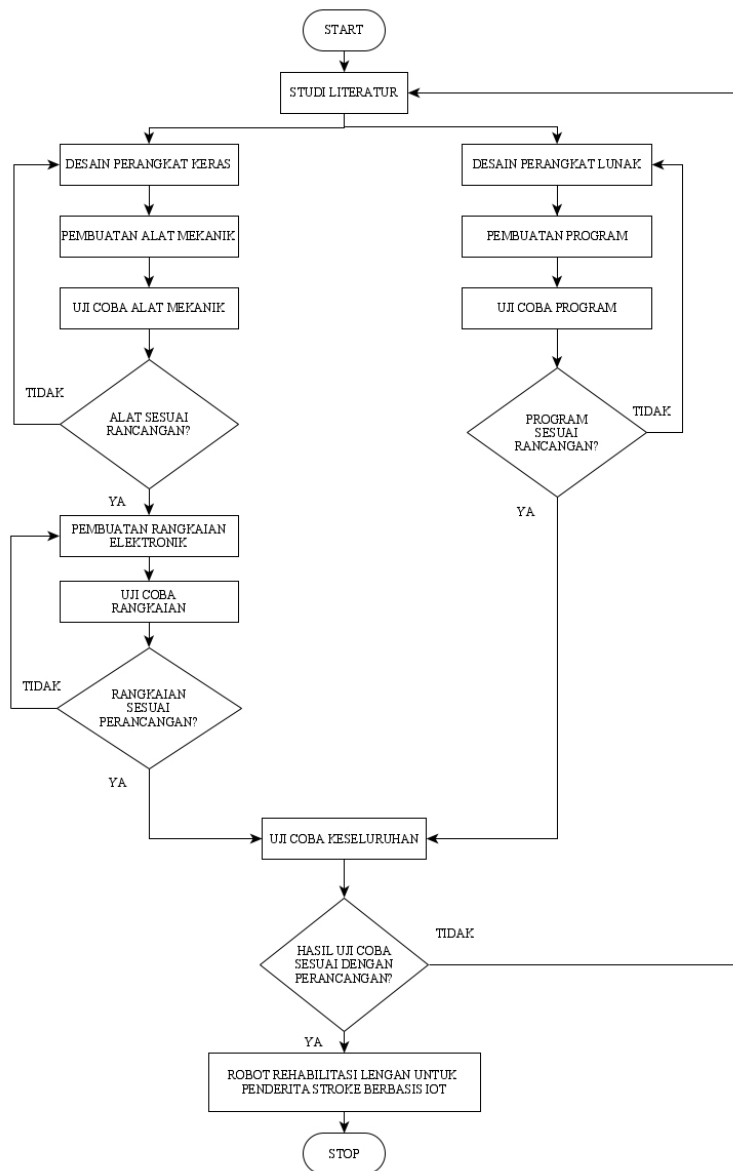
No.	Nama Alat
1.	Laptop
2.	Smartphone
3.	Gurinda
4.	Obeng
5.	Las listrik
6.	Tang
7.	Bor
8.	Solder
9.	Multimeter
10.	Gunting

Tabel 3. 2 Bahan yang dibutuhkan

No	Nama Bahan	Jumlah (buah)
1.	Arduino MEGA	1
2.	NodeMCU	1
3.	Adaptor 12V	1
4.	Motor Listrik dc <i>Power Window</i>	1
5.	Motor Linear	1
6.	Motor Servo	1
7.	BTS 7960	1
8.	Baut mur	Disesuaikan
9.	Kabel <i>jumper</i>	Disesuaikan
10.	<i>Push Button</i>	4
11.	<i>Limit Switch</i>	4
12.	<i>Keypad</i>	1
13.	<i>Heat Sink</i>	1
14.	Kabel data	1
15.	Timah	Disesuaikan
16.	Plat Alumunium	Disesuaikan

3.3 Metode Perancangan

Adapun diagram perancangan sebagai panduan dalam proses perancangan dan pembuatan robot rehabilitasi lengan bagi penderita stroke ini agar dapat memperoleh suatu alat yang baik dari segi mutu serta tetap mempertimbangkan segi ekonomisnya.



Gambar 3. 1 *FlowChart* Sistem Perancangan

3.4 Studi Literatur

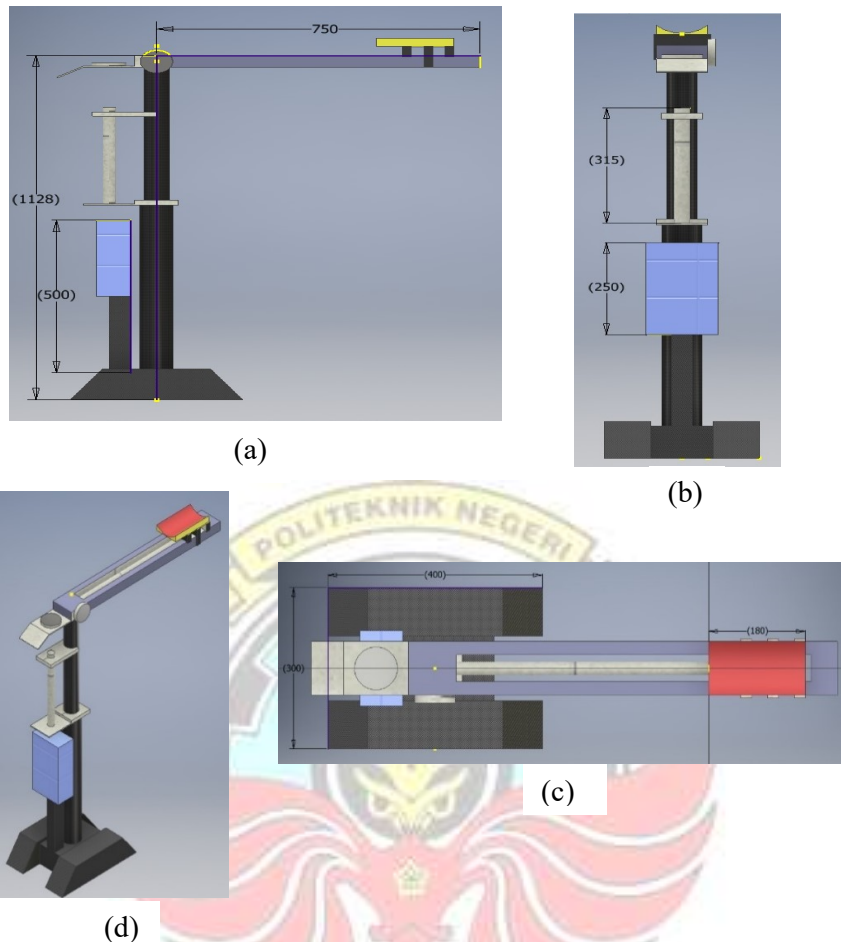
Dalam perancangan alat ini, langkah awal yang dilakukan adalah mencari sebanyak-banyaknya data serta informasi melalui berbagai media cetak maupun elektronik dimana informasi tersebut harus relevan dengan alat yang akan dibuat. Diantaranya dengan melakukan diskusi dengan penyusun sebelumnya, serta melalui media elektronik.

3.5 Perancangan Perangkat Keras

Perancangan perangkat keras merupakan bagian terpenting dalam pembuatan robot rehabilitasi penderita stroke ini. Pada bagian ini berisi mengenai perancangan elektrik dan perancangan mekanik yang akan sangat mempengaruhi kinerja dan hasil akhir dari gerakan robot rehabilitasi ini.

3.5.1 Perancangan Mekanik

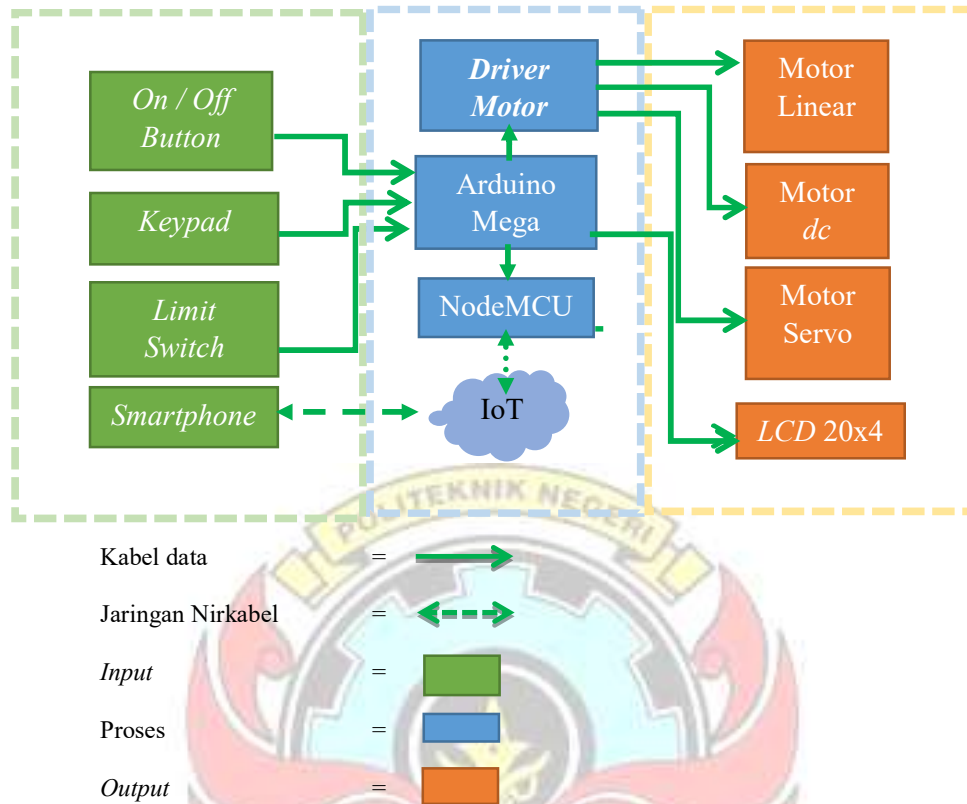
Perancangan mekanik pada robot rehabilitasi berupa sebuah desain konstruksi dan posisi dari komponen-komponen mekanik untuk membentuk sebuah robot rehabilitasi. Desain mekanik dapat dilihat pada gambar 3.2.



Gambar 3. 2 Robot Rehabilitasi Lengan. (a) Tampak Samping. (b) Tampak Depan. (c) Tampak Atas. (d) Tampak Isometrik.

3.5.2 Perancangan Elektronik

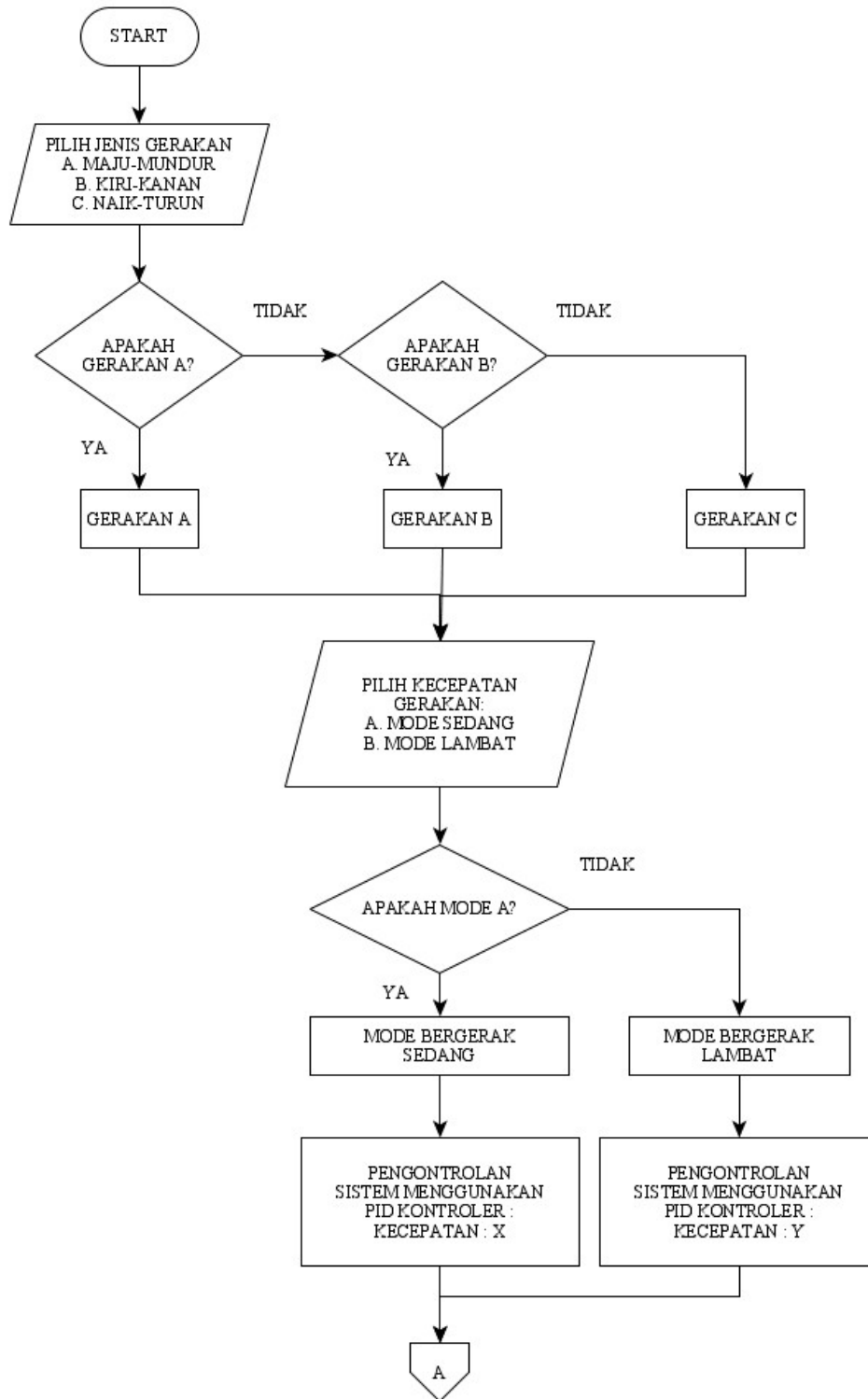
Perancangan *hardware* elektronik pada alat ini dibuat dengan menggunakan beberapa *shield* untuk memperkecil tempat atau *space* dalam peletakannya dan memperkecil ukuran dari alat ini. Adapun komponen penyusun *hardware* elektronika yang digunakan pada alat ini dapat dilihat pada gambar 3.3.

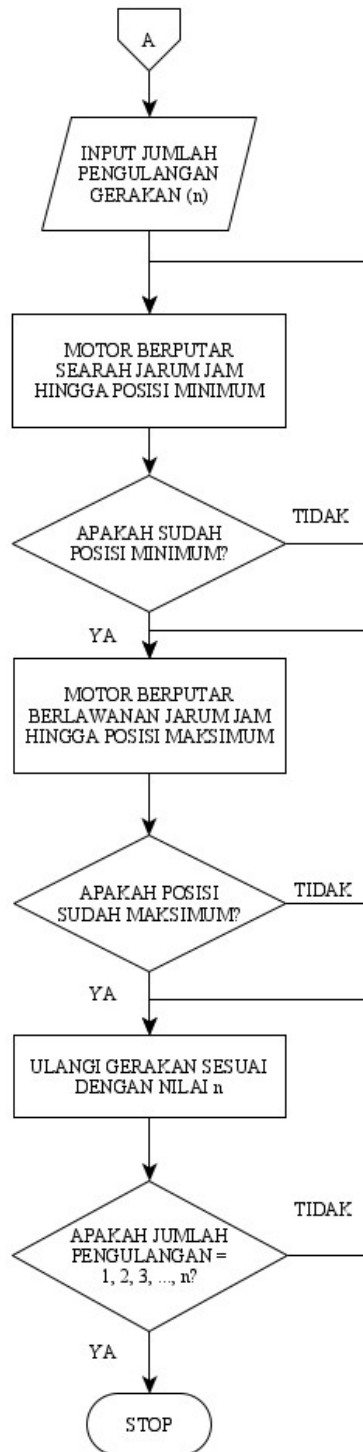


Gambar 3. 3 Skema Rangkaian Elektronik Robot Rehabilitasi

3.6 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada alat ini bertujuan untuk memudahkan pengguna untuk mengoperasikan robot rehabilitasi penderita stroke ini. Perancangan perangkat lunak pada alat ini berupa program yang dibuat dalam komputer kemudian dimasukkan ke dalam perangkat *hardware* elektronika yang telah dibuat. Program tersebut dibuat berdasarkan gerakan yang hendak digunakan dalam proses rehabilitasi yang mana gerakan tersebut didukung dengan kemampuan dan batasan pergerakan yang sesuai dengan desain dari robot rehabilitasi tersebut.





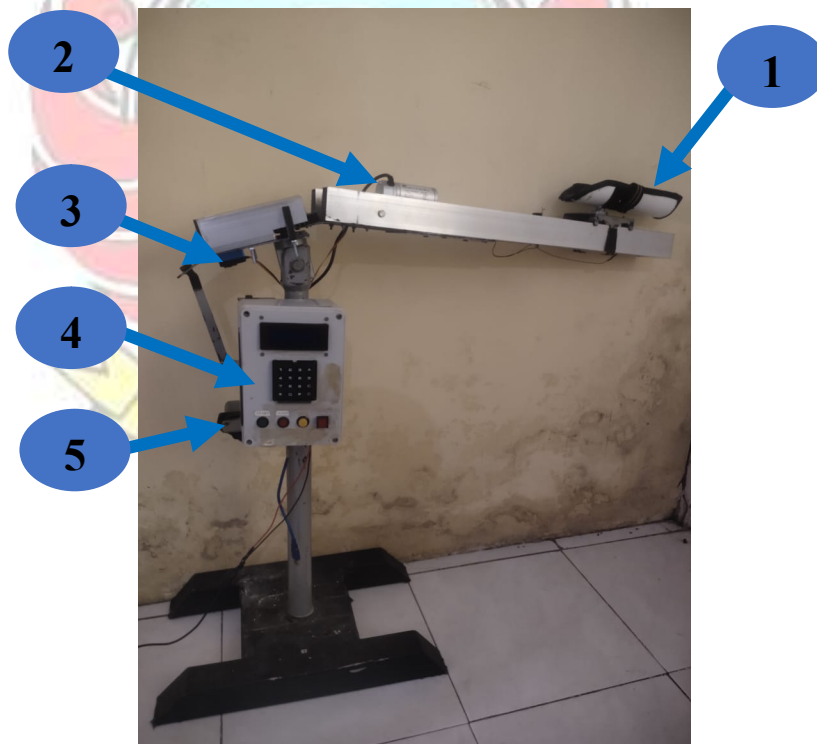
Gambar 3. 4 Diagram Alir Sistem Pergerakan

BAB IV HASIL DAN PEMBAHASAN

Hasil implementasi robot beserta dengan pembahasan pada bab ini dibagi menjadi empat bagian yaitu hasil perancangan mekanik, elektronik program dan hasil pengujian. Beberapa hasil implementasi robot rehabilitasi lengan bagi pasien pasca stroke telah dijelaskan pada Bab III. Adapun perubahan yang telah dilakukan pada proses perancangan dijelaskan pada bab ini.

4.1 Hasil Perancangan Mekanik

Setelah melakukan percobaan maka penulis mendapatkan desain robot sebagai berikut:



Gambar 4. 1 Hasil Perancangan Mekanik

Setelah melalui beberapa saran dan percobaan yang dilakukan, penelitian ini disetujui menggunakan desain mekanik yang lama, namun

dilakukan penggantian pada bagian motor demi mengoptimalkan penelitian tahun lalu. Berikut penjelasan bagian-bagian pada robot rehabilitasi lengan bagi penderita stroke.

Tabel 4. 1 Bagian-bagian robot rehabilitasi lengan

No	Nama Bagian	Fungsi
1	Dudukan tangan	Dudukan tangan didesain panjang untuk kenyamanan tangan pasien dan titik berat tangan pasien dengan dudukan tangan lebih seimbang.
2	Transmisi maju-mundur	Dalam pergerakan maju-mundur menggunakan motor <i>linear</i> sebagai aktuator dengan spesifikasi yang lebih tinggi, sehingga pergerakan lebih cepat dan halus dibandingkan menggunakan ulir.
3	Transmisi kiri-kanan	Dalam pergerakan kiri-kanan menggunakan motor servo sebagai aktuator, sehingga derajat pergerakannya dapat diatur.
4	Transmisi naik-turun	Penghubung aktuator pergerakan naik-turun.
5	<i>Panel box controller</i>	Sebagai tempat rangkaian elektronik dan kontrol alat.

Pada penelitian sebelumnya mekanisme pergerakan kiri-kanan menggunakan motor *power window* yang dimana untuk pergerakannya belum optimal dikarenakan masih menggunakan *limit switch* dan posisi

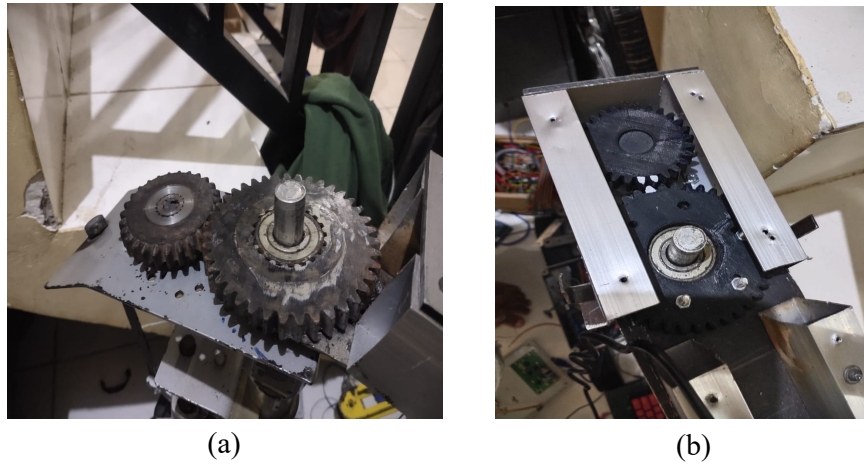
lengan robot tidak bisa berada di posisi tengah dan juga kecepatan Bergeraknya sangat cepat walaupun sudah diatur ke rpm terendah. Maka dari itu pada penelitian kali ini dilakukan perubahan pada mekanisme pergerakan kiri kanan menggunakan motor servo agar derajat atau posisi dari lengannya dapat di posisikan di tengah dan juga sudah tidak menggunakan *limit switch* untuk batas minimal dan maksimal pergerakan motor.



Gambar 4. 2 Penggantian Pergerakan Kiri-kanan dengan Motor Servo

Spesifikasi torsi motor servo yang digunakan yaitu sebesar 40kgf.cm atau jika dikonversikan menjadi 3.92266 Nm dengan tegangan input maksimal 8.4V.

Selanjutnya dilakukan penggantian *gear* transmisi pada pergerakan kiri-kanan dengan bahan plastik agar meringankan beban motor servo.



Gambar 4. 3 *Gear*. (a) Transmisi Pergerakan Kiri-kanan Sebelumnya dan (b) Penggantian Menggunakan Gear Plastik

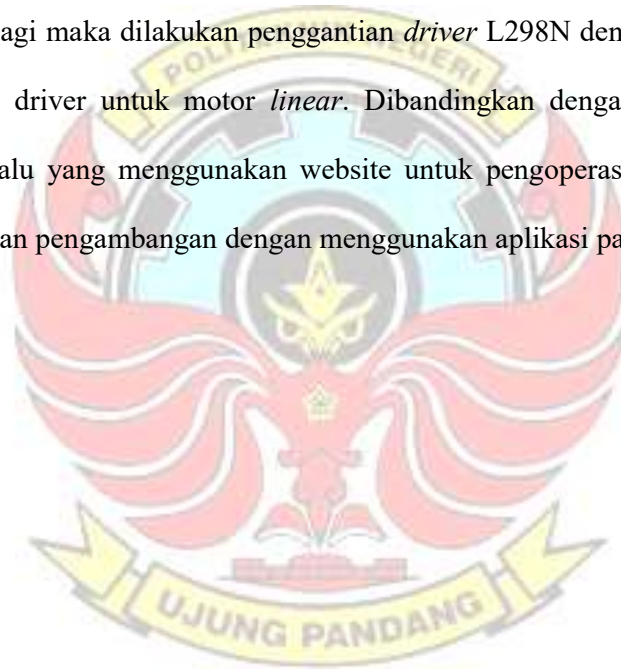
Pada pergerakan maju-mundur dilakukan pengembangan motor *linear* dengan spesifikasi yang lebih tinggi dikarenakan pada motor sebelumnya kecepatannya hanya 5mm/s dan setelah dilakukan penggantian kecepatan motor menjadi 20mm/s yang dirasa sudah cukup untuk melakukan pergerakan.

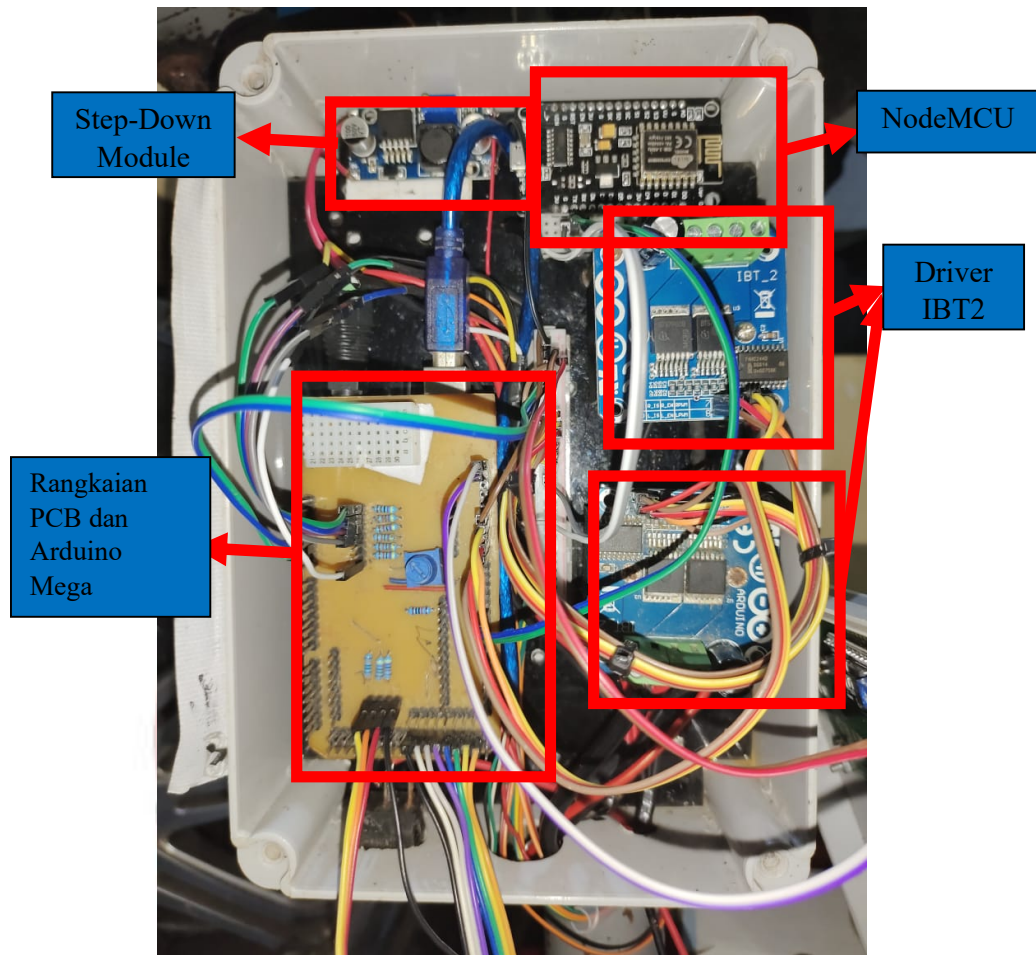


Gambar 4. 4 Penggantian Motor *Linear*

4.2 Hasil Perancangan Elektronik

Pada perancangan elektronik, pengontrolan menggunakan Arduino Mega sebagai mikrokontroler sebab memiliki jumlah *port* yang banyak yang bisa dijadikan *input* dan *output* dan juga menggunakan NodeMCU untuk komunikasi IoT-nya. Pada pengembangan sebelumnya digunakan *driver* IBT Arduino dan *driver* motor L298N dan untuk pengembangan kali ini dikarenakan sudah menggunakan motor servo yang tidak memerlukan *driver* lagi maka dilakukan penggantian *driver* L298N dengan IBT Arduino sebagai driver untuk motor *linear*. Dibandingkan dengan pengembangan tahun lalu yang menggunakan website untuk pengoperasian IoT, kalo ini dilakukan pengembangan dengan menggunakan aplikasi pada *smartphone*.





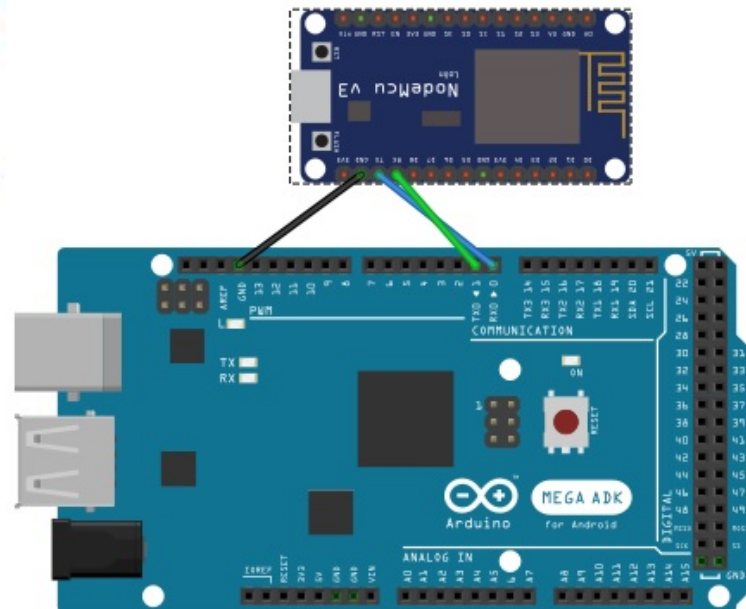
Gambar 4. 5 Rangkaian dan Komponen Elektronik





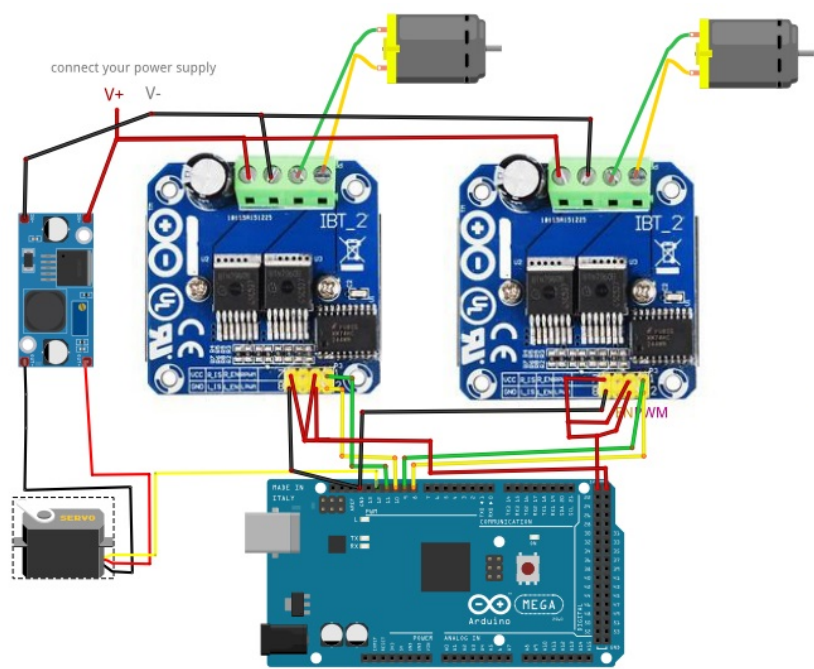
Gambar 4. 6 Tampilan Depan *Box Panel Controller*

1. Hubungan antara Arduino Mega dengan NodeMCU



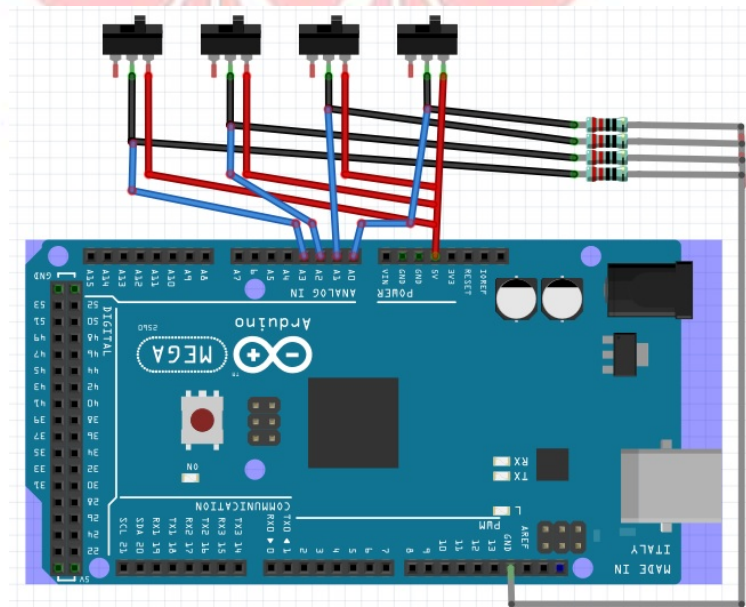
Gambar 4. 7 *Wiring* Komunikasi Serial Arduino Mega dan NodeMCU

2. Hubungan antara motor, *driver* dan Arduino Mega



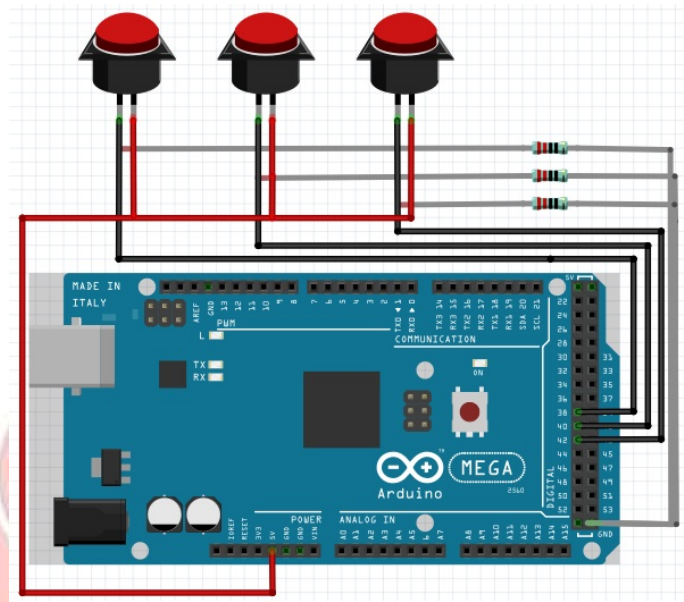
Gambar 4. 8 *Wiring* Motor dan Arduino Mega

3. Hubungan antara *limit switch* dengan Arduino Mega



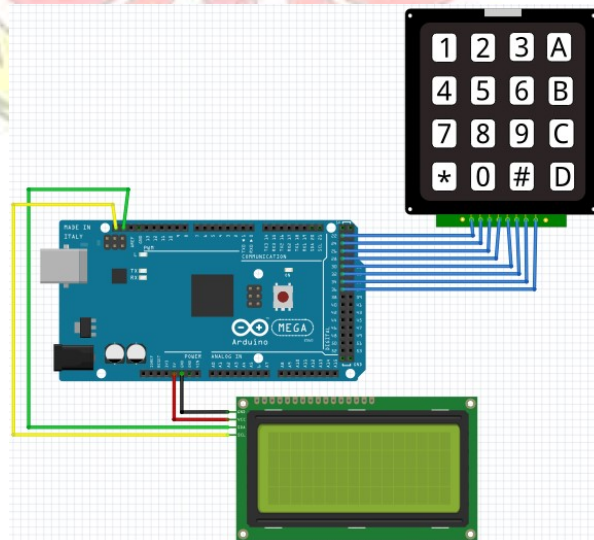
Gambar 4. 9 *Limit Switch* dan Arduino Mega

4. Hubungan antara *Button* dengan Arduino Mega



Gambar 4. 10 *Button* dan Arduino Mega

5. Hubungan antara *keypad*, lcd dan Arduino Mega



Gambar 4. 11 *Keypad*, LCD dan Arduino Mega

6. *Layout* papan PCB



Gambar 4. 12 *Layout* Papan PCB

4.3 Hasil Perancangan Program

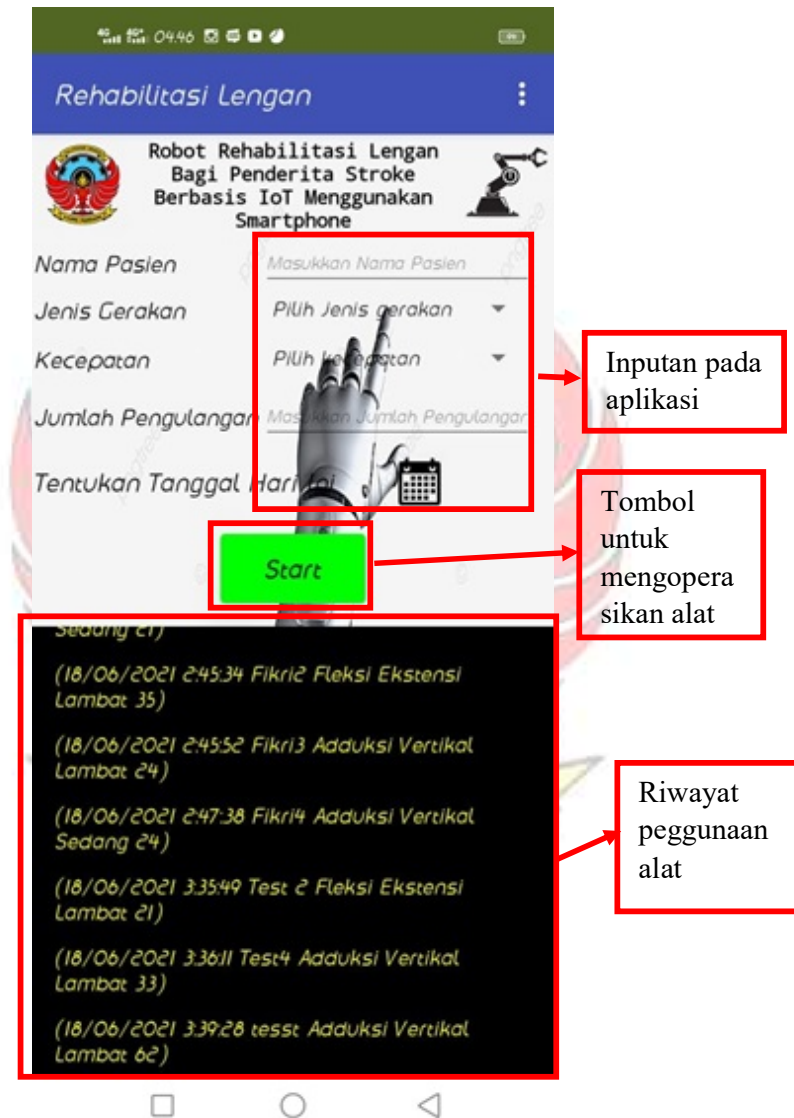
Perancangan program pada robot rehabilitasi lengan terbagi menjadi 2 tahap yaitu pemrograman pada Arduino Mega dan juga pemrograman aplikasi menggunakan *smartphone*.

1. Program pada Arduino Mega

Untuk pembuatan program dilakukan menggunakan *software* Arduino yang mengatur proses kerja dari semua peralatan yang telah dibuat. Adapun *listing* program gerakan yang dibebankan pada mikrokontroler arduino sebagai berikut:

1. Sub program untuk LCD dan *Keypad*
2. Sub program untuk tombol *Start*, *Stop* dan *Reset*
3. Sub program untuk pergerakan motor
4. Sub program untuk mengambil data dari server
5. Sub program untuk memecah data yang di terima dari server
6. Sub program untuk pertanyaan jenis gerakan

7. Sub program untuk pertanyaan kecepatan
 8. Sub program untuk pertanyaan jumlah pengulangan gerakan
 9. Program utama
2. Program aplikasi menggunakan *smartphone*



Gambar 4. 13 Tampilan Aplikasi

Tahap pembuatan aplikasi *smartphone* berbasis *Internet of Things* dibuat berupa *code block* dalam aplikasi *MIT App Inventor*. Dalam

sistem ini, *smartphone* berfungsi sebagai *controller* dimana di aplikasi pengguna dapat menginput jenis gerakan, kecepatan dan jumlah pengulangan dimanapun dan kapanpun asalkan alat terhubung ke listrik dan juga ke internet. Di dalam aplikasi juga terdapat riwayat penggunaan alat sehingga pengguna melakukan pengontrolan dan bisa mengetahui perkembangan pasien dari terapi yang dilakukannya. Dengan sistem ini diharapkan dapat mempermudah dalam pengoperasian alat.

4.4 Pengujian Motor Linear

Pada tahap pengujian motor linear, dilakukan pembacaan tegangan yang dibutuhkan terhadap beban yang diberikan. Dalam pengujian ini motor linear diberikan beban seberat 1kg untuk mengetahui tegangan yang dibutuhkan pada kecepatan tertentu. Pengujian dilakukan menggunakan adaptor 12VDC dan diukur menggunakan multimeter.

Tabel 4. 2 Pengujian Motor Linear

No.	Kecepatan Motor (mm/s)	Tegangan (V)
1	8	11.7
2	10	11.7
3	12	11.8
4	14	11.9
5	16	11.9
6	18	12
7	20	12

Berdasarkan Tabel 4.2, pengujian dengan kecepatan motor 8-10mm/s memerlukan tegangan sebesar 11.7V, pada kecepatan 12mm/s didapatkan

tegangan sebesar 11.8V dan pada kecepatan 14-16mm/s dibutuhkan tegangan sebesar 11.9V. Pada saat kecepatan motor diatur ke 20mm/s, *supply* tegangan yang dibutuhkan yaitu 12V dan dapat disimpulkan motor linear bekerja dengan baik pada tegangan tersebut.

4.5 Pengujian Motor Servo

Pada tahap pengujian motor servo, dilakukan pembacaan tegangan yang dibutuhkan terhadap beban yang diberikan. Dalam pengujian ini digunakan dua jenis material *gear* yaitu besi dan plastik dan juga penggunaan *step-down* untuk mengatur tegangan pada motor servo. Beban yang digunakan yaitu berat dari lengan alat itu sendiri ditambah dengan berat dari motor linear.

Tabel 4. 3 Hasil Pengujian Motor Servo Menggunakan *Gear* Material Besi

No	Tegangan (V)	Kondisi Servo
1	5	Tidak bergerak
2	5.5	Tidak bergerak
3	6	Tidak bergerak
4	6.5	Tidak bergerak
5	7	Tidak bergerak
6	7.5	Tidak bergerak
7	8	Tidak bergerak
8	8.2	Tidak bergerak
9	8.4	Bergerak
10	8.6	Bergerak
11	9	Tidak teratur
12	10	Tidak teratur
13	12	Tidak teratur

Tabel 4. 4 Hasil Pengujian Motor Servo Menggunakan *Gear* Material Plastik

No	Tegangan (V)	Kondisi Servo
1	5	Tidak bergerak
2	5.5	Tidak bergerak
3	6	Tidak bergerak
4	6.5	Tidak bergerak
5	7	Lemah
6	7.5	Lemah
7	8	Lemah
8	8.2	Lemah
9	8.4	Stabil
10	8.6	Stabil
11	9	Tidak teratur
12	10	Tidak teratur
13	12	Tidak teratur

Berdasarkan Tabel 4.3, pada saat tegangan yang diberikan sebesar 5-8.2V motor servo tidak dapat melakukan pergerakan atau beban terlalu besar, setelah diberikan tegangan sebesar 8,4-8,6V motor servo melakukan pergerakan tetapi dengan intensitas yang kecil dan tersendat-sendat, ketika diberikan tegangan di atas 8,6V motor servo bergerak tetapi tidak mengikuti program alias pergerakannya acak. Berbeda dengan Tabel 4.4 dapat dilihat motor servo sudah dapat melakukan pergerakan ketika diberi tegangan sebesar 7V dan setelah diberi tegangan 8,4-8,6V maka motor servo sudah dapat bergerak dengan mulus.

Dari hasil pengujian yang dilakukan dapat ditarik kesimpulan bahwa penggunaan material plastik pada *gaer* berpengaruh besar terhadap kinerja dari motor servo. Motor servo dengan spesifikasi 40kgf.cm dapat bekerja dengan optimal ditegangan 8,4-8,6V. pengujian ini dilakukan dengan mengukur tengangan menggunakan alat ukur multimeter.

Penggantian motor *power window* dengan motor servo pada pergerakan kiri-kanan dapat membuat pengoperasian alat lebih efektif dikarenakan penggunaan motor servo dapat membuat posisi lengan robot berada di tengah dengan mengatur derajat. Berbeda dengan saat menggunakan motor *power window* dimana posisi berhentinya lengan hanya bisa di posisi kiri saja. Adapun penggunaan *gear* plastik yang berbahan ABS sangat membantu untuk meringankan beban dari motor servo.

Pada saat menggunakan tegangan langsung dari arduino yaitu 5V motor servo tidak dapat bergerak maka penulis mencoba menggunakan modul *step-down* dengan inputan dari adaptor sebesar 12V dan tegangan keluaran yang diteruskan ke motor servo dapat diatur. Setelah beberapa kali percobaan didapatlah tegangan optimal untuk pergerakan motor yang stabil yaitu 8,6V. Penggunaan kabel juga dapat mempengaruhi kestabilan tegangan yang dikirim ke motor servo.

4.6 Pengujian Aplikasi

Pengujian ini dibagi menjadi 4 bagian yaitu:

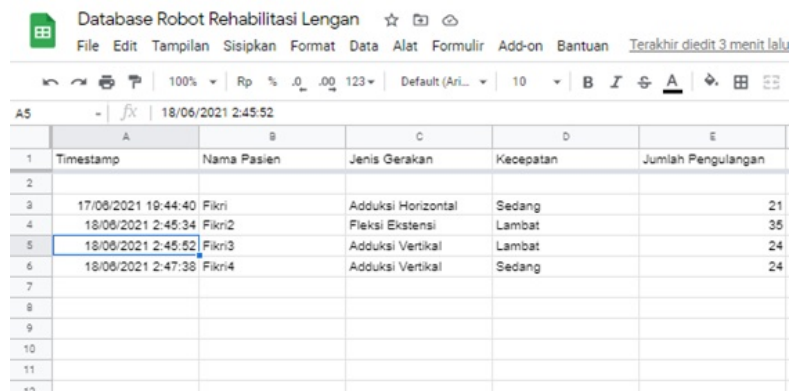
a. Aplikasi mengirim data ke server

Dilakukan pengujian dengan cara memberikan inputan pada aplikasi berupa nama pasien, jenis gerakan, kecepatan dan jumlah pengulangan yang diinginkan.



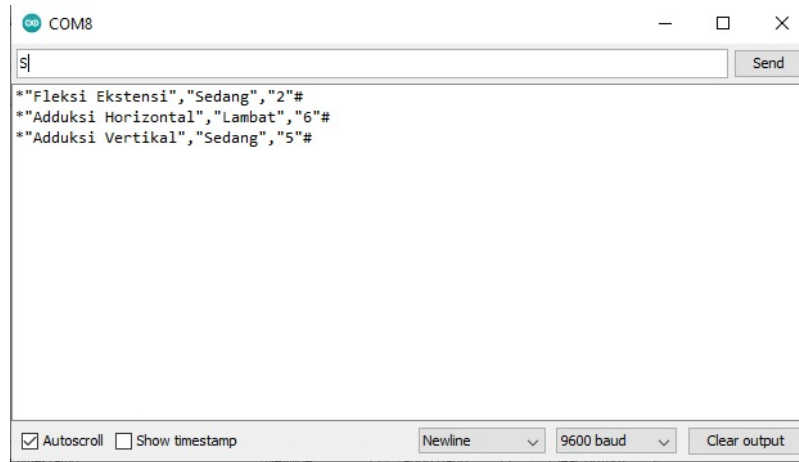
Gambar 4. 14 Aplikasi Mengirim Data ke *Firestore*

Data yang dikirim dari ke *firebase* juga dikirim ke *google spreadsheet* untuk dijadikan riwayat penggunaan alat.



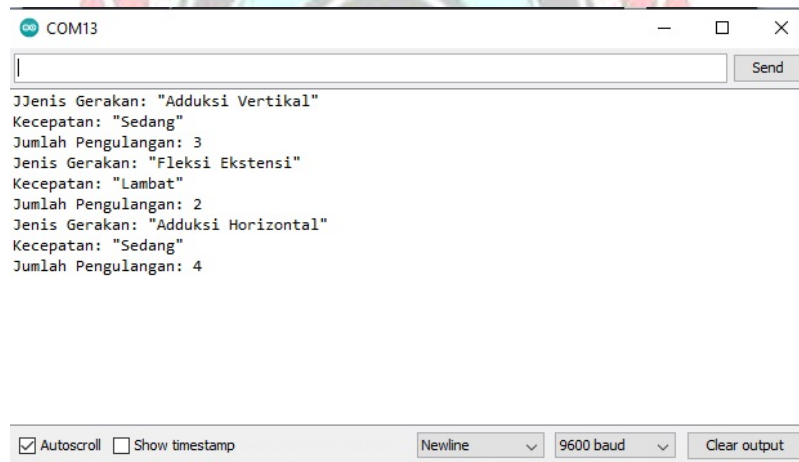
Gambar 4. 15 Aplikasi Mengirim Data ke *Google Spreadsheet*

b. NodeMCU mengambil data dari *Firebase*



Gambar 4. 16 Tampilan Data dari *Firebase* ke NodeMCU

c. NodeMCU meneruskan data ke Arduino Mega



Gambar 4. 17 Tampilan Data dari NodeMCU diserialkan ke Arduino Mega

Sebelum melakukan pengujian dipastikan NodeMCU maupun *smartphone* terhubung ke internet terlebih dahulu, setelah dipastikan sudah terhubung maka selanjutnya dilakukan penginputan pada aplikasi berupa nama, jenis gerakan, kecepatan dan berapa jumlah pengulangan yang diinginkan. Setelah melakukan penginputan maka tekan tombol

start pada aplikasi yang selanjutnya data yang telah diinput akan diteruskan ke *firebase* sebagai server. Dikarenakan *firebase* merupakan *real-time database* maka tidak dapat melihat riwayat data apa saja yang telah dikirimkan, maka dari itu dibuatlah *google spreadsheet* sebagai server penyimpan riwayat data yang telah dikirim aplikasi. Selanjutnya NodeMCU mengambil data yang tersimpan di *firebase* yang kemudian akan diteruskan ke Arduino Mega menggunakan komunikasi serial untuk selanjutnya diolah dan mengoperasikan alat.

4.7 Perbandingan dengan Penelitian Sebelumnya

Pada penelitian ini memiliki beberapa perbedaan yang signifikan dengan penelitian sebelumnya yaitu sistem pergerakan dan sistem IoT.

Tabel 4. 5 Perbandingan dengan Penelitian Sebelumnya

Aspek	Penelitian Sebelumnya	Penelitian Sekarang
Pergerakan maju-mundur	Menggunakan motor linear dengan spesifikasi kecepatan 5mm/s	Mengganti motor linear dengan spesifikasi kecepatan 20mm/s
Pergerakan kiri-kanan	Menggunakan motor dc	Mengganti dengan menggunakan motor servo
Sistem IoT	Menggunakan <i>website</i> untuk <i>monitoring</i> alat	Menggunakan <i>smartphone</i> untuk mengontrol dan <i>monitoring</i> penggunaan alat

BAB V PENUTUP

5.1 Kesimpulan

Dari hasil dan pembahasan penelitian ini, dapat ditarik beberapa kesimpulan sebagai berikut

1. Dengan dilakukannya penggantian motor *linear* dengan spesifikasi yang lebih tinggi yang sebelumnya kecepatannya hanya 5mm/s menjadi motor dengan kecepatan 20mm/s lebih optimal untuk melakukan pergerakan.
2. Pengontrolan pergerakan kiri-kanan robot lebih efektif menggunakan motor servo dikarenakan penggunaan motor servo dapat membuat posisi lengan robot berada di tengah dengan mengatur derajat.
3. Dengan dibuatnya aplikasi smartphone untuk robot rehabilitasi lengan, maka dapat mempermudah dalam pengoperasian dan memonitoring kerja alat.

5.2 Saran

Berikut ini beberapa saran untuk pengembangan robot rehabilitasi lengan kedepannya:

1. Menambahkan fitur pada aplikasi yang membuat dokter dapat menjadwalkan dan mendapat notifikasi ketika alat digunakan.
2. Pergerakan *adduksi* horizontal sebaiknya menggunakan transmisi gear yang lebih diperhitungkan lagi agar dapat membuat pergerakan lebih baik dengan tidak terlalu membebankan motor.

DAFTAR PUSTAKA

- Agus, Muh. dan Ikram. 2017. *Pengembangan Robot Rehabilitasi Lengan untuk Penderita Stroke*. Skripsi. Makassar: Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.
- Alexnld. 2021. *Coreless Digital Servo Large Torque Metal Gear For Rc Robot*. (online), (<https://alexnld.com/product/spt-spt5632-270-32kg-coreless-digital-servo-large-torque-metal-gear-for-rc-robot/>), Diakses 8 Februari 2021.
- Arduino Home Page. 2012. *Arduino Mega Getting Started*. (online), (<http://www.arduino.cc/en/Main/arduinoBordMega>), Diakses 02 Februari 2021.
- Ardutech. 2019. *Cara Menyambung Keypad dengan Arduino*. (online), (<https://www.ardutech.com/cara-menyambung-keypad-dengan-arduino/>). Diakses 10 Februari 2021.
- Bastian, Y. D. 2011. *Rehabilitasi Stroke. RS. Mitra Keluarga, Depok*. (online), (<http://www.mitrakeluarga.com/depok/rehabilitasi-stroke>), Diakses 02 Februari 2021.
- Dalgas, U. dkk. 2008. Multiple sclerosis and physical exercis: recommendations for the application of resistance, endurance and combined-training. *Multiple Sclerosis*, 14(1): 35-53.
- Dickson, Robbie. 2020. *How Does A Linear Actuator Work*. (online), (<https://www.firgelliauto.com/id/blogs/news/how-does-a-linear-actuator-work>), Diakses 10 Februari 2021.
- F1 Depo. 2018. *BTS7960B 20 Amper Motor Sürücü Devresi (BTS7960)*. (Online), (<https://www.f1depo.com/BTS7960B-20-Amper-Motor-Surucu-Devresi-BTS7960-,PR-190.html>), Diakses 7 Februari 2021.
- Hariandja, Johanna Renny Octavia. 2013. *Identifikasi Kebutuhan akan Sistem Rehabilitasi Berbasis Teknologi Terjangkau untuk Penderita Stroke di Indonesia*. Skripsi. Bandung: Lembaga Penelitian dan Pengabdian kepada Masyarakat Universitas Katolik Parahyangan.

- Hermawan, Trian. 2017. *Menaikan Atau Menurunkan Tegangan Dc.* (Online), (<http://trianhermawan.blogspot.com/2017/11/menaikan-atau-menurunkan-tegangan-dc-to.html?m=1>), Diakses 10 Februari 2021
- Indozone. 2020. *Kenali Gejala Dan Tanda Awal Stroke dengan Fast Apakah Itu.* (Online), (<https://www.indozone.id/health/XxsEPQo/kenali-gejala-dan-tanda-awal-stroke-dengan-fast-apakah-itu/read-all>), Diakses 9 Februari 2021.
- Info Panduan Trik. 2015. *Pengertian Adaptor dan Jenisnya.* (Online), (<http://infopanduantrik.blogspot.com/2015/09/pengertian-adaptor-dan-jenisnya.html?m=1>), Diakses 10 Februari 2021.
- Junaidi, Iskandar. 2011. *Stroke Waspada Ancaman.* Yogyakarta: Andi
- Kelas PLC. 2021. *Macam-Macam Saklar dan Gambarnya Saklar Mekanik.* (Online), (<https://www.kelasplc.com/macam-macam-saklar-dan-gambarnya-saklar-mekanik/>), Diakses 10 Februari 2021.
- Marianjoy. 2021. *Rehab Technology.* (Online), (<https://www.marianjoy.org/rehab-technology/tellabs/armeopower.aspx>), Diakses 02 Februari 2021.
- Motl, R. W dan Gosney J. L., 2008. Effect of exercise training on quality of life in multiple sclerosis: a meta-analysis. *Multiple Sclerosis*, 14 (1): 129-135.
- Mubarak, Husnul. 2016. Wawancara langsung, Fisioterapi, RS. Universitas Hasanudin Makassar.
- Psti Unisayogya. 2020. *Apa Itu Mit App Inventor Berikut Penjelasannya.* (online), (<https://psti.unisayogya.ac.id/2020/01/06/apa-itu-mit-app-inventor-berikut-penjasannya/>), Diakses 7 Februari 2021.
- Rahmat, Abdul dan Muh. Syahid Iswal. 2016. *Rancang Bangun Robot Rehabilitasi Lengan Bagi Penderita Stroke.* Skripsi. Makassar: Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.
- Sendari, Anugerah Ayu. 2020. Liputan6. (online), (<https://hot.liputan6.com/read/4207016/fungsi-transformator-step-down-pahami-cara-kerjanya>), Diakses 18 Agustus 2021.

- Sinau Programming. 2020. *Menampilkan Text Pada Lcd 16x2 Arduino*. (online), (<https://www.sinauprogramming.com/2020/10/menampilkan-text-pada-lcd-16x2-arduino.html?m=1>), Diakses 10 Februari 2021.
- Smeltzer C. S, Brunner dan Suddarth. 2002. *Buku Ajar Keperawatan Medikal Bedah*. Jakarta: EGC.
- Stans. 2017. *NodeMCU Esp8266 Iot Development Board Overview*. (online). (<https://42bots.com/esp8266/nodemcu-esp8266-iot-development-board-overview/>). Diakses 7 Februari 2021.
- Susilo. 2000. *Simposium Stroke, Patofisiologi Dan Penanganan Stroke, Suatu Pendekatan Baru Millenium III*. Bangkalan: IDI
- Syah, Efran. 2014. *Dampak Stroke dan Siapa Saja yang Berisiko*. (online), (<https://www.medkes.com/2014/07/dampak-stroke-dan-siapa-saja-yang-berisiko.html>), Diakses 02 Februari 2021
- Widiyanto. 2009. *Terapi Gerak Bagi Penderita Stroke*. Universitas Negeri Yogyakarta. Indonesia.
- Wirawan, R. 2009. Rehabilitasi Stroke pada Pelayanan Kesehatan Primer. Volume 59. *Majalah Kedokteran Indonesia*. Jakarta. (online). (www.indonesia.digitaljournals.org). Diakses 02 Februari 2021.
- Yanik, Kiki Aprilia. 2016. *Aplikasi Motor DC sebagai Penggerak Tabung Pengisian Baglog pada Alat Pengepres Baglog Jamur Otomatis*. Skripsi. Yogyakarta. Jurusan Pendidikan Teknik Elektro Fakultas Teknik Universitas Negeri Yogyakarta.

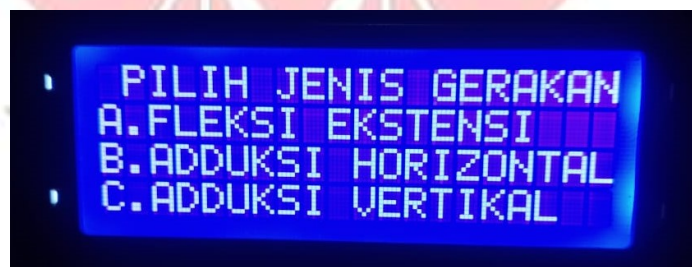
LAMPIRAN

Lampiran 1 Tahapan-tahapan pengoperasian

1. Hubungkan adaptor 12Vdc ke listrik 220 Vac.
2. Tekan tombol ON pada panel.
3. Selanjutnya muncul tampilan penulis serta pembimbing seperti pada gambar berikut:



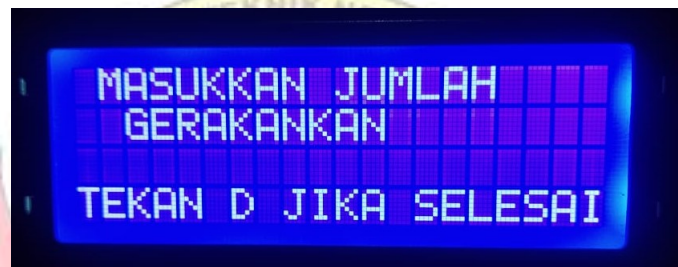
4. Tekan *start* maka muncul pertanyaan untuk memilih jenis gerakan yang diinginkan seperti gambar berikut:



5. Setelah meng-*input* jenis gerakan, maka muncul menu selanjutnya untuk memilih kecepatan motor yang diinginkan seperti gambar berikut:



6. Setelah meng-*input* kecepatan motor, maka muncul menu selanjutnya untuk memilih berapa jumlah gerakan pengulangan yang diinginkan seperti gambar berikut:

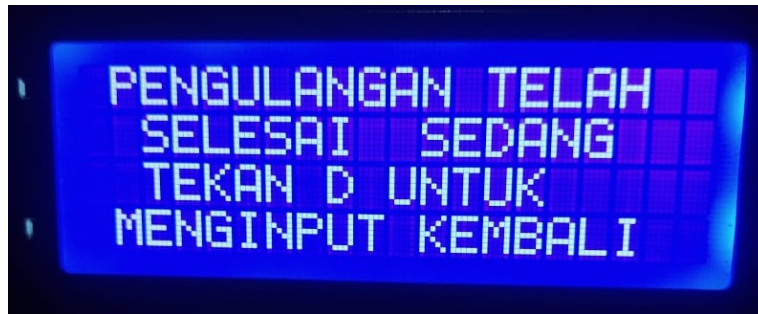


7. Jika kita telah menginput jumlah gerakan yang diinginkan, maka akan muncul tiga parameter untuk meyakinkan bahwa gerakan yang diinginkan sudah sesuai dengan perintah untuk dijalankan dalam program seperti gambar berikut:



8. Jika data selesai maka tekan tombol *start* untuk melakukan proses rehabilitasi dan motor akan bergerak otomatis sesuai perintah yang di-*input* sebelumnya, tapi jika merasa ada yang salah dalam melakukan input

dari 3 parameter yang ada, maka bisa menekan D untuk meng-*input* kembali.



9. Proses pergerakan terhitung satu kali jika kedua *limit switch* pada setiap lengan tertekan.
10. Setelah menggunakan alat maka lepas adaptor.
11. Selesai.

Lampiran 2 Program di Arduino Mega

```
#include <Wire.h>
#include <Liquidcrystal_I2C.h>
#include <Keypad.h>
#include <Servo.h>
Servo myservo;

void(*Reset_ya)(void)= 0;
/////RESET
MICON//////////////////////////////////////
/////

const int Start =38;
const int Stopp =40;
const int Reset =42;
const int PWM_1 =10;
const int PWM_2 =11;
const int PWM_3 =9;
const int PWM_4 =8;
const int Limit_Max1 = A2;
const int Limit_Min1 = A3;
const int Limit_Max2 = A0;
```



```

const int Limit_Min2 = A1;

////////MENGANTI NAMA
PIN////////////////////////////////////

String dataIn;
String dt[10];
String InputPengulangan;
String getdata_JenisGerakan;
String getdata_Kecepatan;
String getdata_JumlahPengulangan;
int i;
int count =0;
int a      =0;
int K      =0;
int S      =0;
int l      =0;
int k      =0;
int kec    =0;
int pin1   =0;
int pin2   =0;
int no_input;
int ulang ;
int Pengulangan;
int InputOk;
int InputModeOk;
int Input_Ok;
int Output_pengulangan;
int Jumlah ;
int J_Pengulangan;
int b =0;
int pos = 45;
boolean parsing=false;
int LED1 = 4;
int LED2 = 5;
int LED3 = 6;

////////SET
VARIABEL////////////////////////////////////
////////

const byte ROWS = 4;
const byte COLS = 4;

char hexaKeys[ROWS][COLS] = {

```



```

    {'1', '4', '7', '*'},
    {'2', '5', '8', '0'},
    {'3', '6', '9', '#'},
    {'A', 'B', 'C', 'D'}
};

byte rowPins[COLS] = {36, 34, 32, 30};
byte colPins[ROWS] = {28, 26, 24, 22};

Keypad myKeypad = Keypad(makeKeymap(hexaKeys), rowPins,
colPins, ROWS, COLS);

/////////KEYPAD/////////
/////////

LiquidCrystal_I2C lcd(0x27, 20, 4);

/////////LCD/////////
/////////

void setup()
{
  pinMode (Start, INPUT);
  pinMode (Reset, INPUT);
  pinMode (Stopp, INPUT);
  pinMode (Limit_Max1, INPUT);
  pinMode (Limit_Min1, INPUT);
  pinMode (Limit_Max2, INPUT);
  pinMode (Limit_Min2, INPUT);
  pinMode (PWM_1, OUTPUT);
  pinMode (PWM_2, OUTPUT);
  pinMode (PWM_3, OUTPUT);
  pinMode (PWM_4, OUTPUT);
  pinMode (LED1, OUTPUT);
  pinMode (LED2, OUTPUT);
  pinMode (LED3, OUTPUT);
  dataIn="";
  myservo.attach(12);
  Serial.begin(9600);

/////////SET INPUT & OUTPUT/////////

  lcd.backlight();
  lcd.init();
  lcd.setCursor(0,0);

```

```

lcd.print      (".....WELCOME.....");
lcd.setCursor(0,1);
lcd.print      (" ROBOT REHABILITASI ");
lcd.setCursor(0,2);
lcd.print      ("PEMBIMBING:");
lcd.setCursor(0,3);
lcd.print      ("PAK LEWI & PAK SIMON");
delay(2000);
lcd.setCursor(0,0);
lcd.print      (".....WELCOME.....");
lcd.setCursor(0,1);
lcd.print      (" ROBOT REHABILITASI ");
lcd.setCursor(0,2);
lcd.print      (" BY:FIKRI & FITRI ");
lcd.setCursor(0,3);
lcd.print      ("TEKAN START U/ MULAI");
}

////////TAMPILAN UTAMA LCD SAAT DINYALAKAN//////////

void parsingData()
{
  int j=0;
  dt[j]="";
  for(i=1;i<dataIn.length();i++)
  {
    if ((dataIn[i] == '#') || (dataIn[i] == ','))
    {
      j++;
      dt[j]=""; //inisialisasi variabel array dt[j]
    }
    else
    {
      dt[j] = dt[j] + dataIn[i];
    }
  }
  getdata_JenisGerakan = String(dt[0]);
  getdata_Kecepatan = String(dt[1]);
  getdata_JumlahPengulangan = String(dt[2]);
  getdata_JumlahPengulangan.trim();
  J_Pengulangan =
  String(getdata_JumlahPengulangan[1]).toInt();
  Serial.print("Jenis Gerakan: ");
  Serial.println(getdata_JenisGerakan);
}

```

```

Serial.print("Kecepatan: ");
Serial.println(getdata_Kecepatan);
Serial.print("Jumlah Pengulangan: ");
Serial.println(J_Pengulangan);
}

////////SUB PROGRAM PEMECAHAN DATA////////////////////////////////////

void RESET ()
{
  if(digitalRead(Reset)== HIGH)
  {
    Serial.println("reset ok");
    Reset_ya();
  }
}
////////SUB PROGRAM RESET////////////////////////////////////

void Stop ()
{
  if(digitalRead(Stopp)==HIGH)
  {
    Serial.println("Stop");
    lcd.clear( );
Wait_PB_Start:
    lcd.setCursor(0,0);
    lcd.print("  STATUS = STOP  ");
    lcd.setCursor(0,2);
    lcd.print("  START => LANJUT  ");
    lcd.setCursor(0,3);
    lcd.print(" RESET => MENU AWAL");
    RESET();
    MotorStop();
    if(digitalRead(Start)==HIGH)
    {
      goto Poin3;
    }
    goto Wait_PB_Start;
Poin3:
    Serial.println("Start");
    S=1;
  }
}
}

```

```
////////SUB PROGRAM STOP////////////////////////////////////
```

```
void MotorMaju()  
{  
  analogWrite(pin1, 0);  
  analogWrite(pin2, kec);  
}
```

```
////////SUB PROGRAM MOTOR MAJU////////////////////////////////////
```

```
void MotorMundur()  
{  
  analogWrite(pin1, kec);  
  analogWrite(pin2, 0);  
}
```

```
////////SUB PROGRAM MOTOR MAJU////////////////////////////////////
```

```
void MotorStop()  
{  
  analogWrite(pin1, 0);  
  analogWrite(pin2, 0);  
  myservo.write(45);  
}
```

```
////////SUB PROGRAM MOTOR STOP////////////////////////////////////
```

```
void MotorKiri()  
{  
  for (pos = 45; pos <= 90; pos += 1)  
  {  
    myservo.write(pos);  
    delay(kec);  
  }  
  for (pos = 90; pos >= 0; pos -= 1)  
  {  
    myservo.write(pos);  
    delay(kec);  
  }  
}
```

```
////////SUB PROGRAM MOTOR KIRI////////////////////////////////////
```

```
void MotorKanan()
```

```

{
  for (pos = 0; pos <= 45; pos += 1)
  {
    myservo.write(pos);
    delay(kec);
  }
}

////////SUB PROGRAM MOTOR KANAN////////////////////////////////////

void PertanyaanJenisGerakan()
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" PILIH JENIS GERAKAN ");
  lcd.setCursor(0,1);
  lcd.print("A.FLEKSI EKSTENSI");
  lcd.setCursor(0,2);
  lcd.print("B.ADDUKSI HORIZONTAL");
  lcd.setCursor(0,3);
  lcd.print("C.ADDUKSI VERTIKAL");
  InputModeLagi:
  Stop();
  char keypressed = myKeypad.getKey();
  if (keypressed == 'A')
  {
    pin1=PWM_3;
    pin2=PWM_4;
    l=1;
    K=1;
    Serial.println("Fleksi Ekstensi");
    goto InputModeOk;
  }
  else if (keypressed=='B')
  {
    l=2;
    K=2;
    Serial.println("Adduksi Horizontal");
    goto InputModeOk;
  }
  else if (keypressed == 'C')
  {
    pin1=PWM_2;
    pin2=PWM_1;
  }
}

```

```

        l=3;
        K=3;
        Serial.println("Adduksi Vertikal");
        goto InputModeOk;
    }
    else
    {
        goto InputModeLagi;
    }
InputModeOk:
    Serial.println("Input Jenis Gerakan Ok");
}

////////SUB PROGRAM JENIS GERAKAN MOTOR////////////////////////////////////

void PertanyaanKecepatan()
{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" PILIH KEC. GERAKAN ");
    lcd.setCursor(0,2);
    lcd.print("A MODE SEDANG");
    lcd.setCursor(0,3);
    lcd.print("B MODE LAMBAT");
InputKecLagi:
    Stop();
    char keypressed = myKeypad.getKey();
    if((keypressed == 'A') &&(K==1))
    {
        kec = 255;
        Serial.println("kec=100");
        goto InputOk;
    }
    else if((keypressed == 'B') &&(K==1))
    {
        kec = 200;
        Serial.println("kec=70");
        goto InputOk;
    }
    else if((keypressed == 'A') &&(K==2))
    {
        kec = 20;
        Serial.println("Kec=225");
        goto InputOk;
    }
}

```

```

}
else if ((keypressed == 'B') && (K==2))
{
    kec = 40;
    Serial.println("kec=100");
    goto InputOk;
}
else if ((keypressed == 'A') && (K==3))
{
    kec = 225;
    Serial.println("kec=225");
    goto InputOk;
}
else if ((keypressed == 'B') && (K==3))
{
    kec = 150;
    Serial.println("Kec=150");
    goto InputOk;
}
else
{
    Serial.println("Input lagi");
    goto InputKecLagi;
}
InputOk:
Serial.println("input ok");
}

////////SUB PROGRAM KECEPATAN MOTOR////////////////////////////////////

void JumlahPengulangan()
{
Input_Pertama:
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" MASUKKAN JUMLAH ");
    lcd.setCursor(0,1);
    lcd.print(" GERAKKAN ");
    lcd.setCursor(0,3);
    lcd.print("TEKAN D JIKA SELESAI");
Input_Lagi:
    Stop();
    char keypressed = myKeypad.getKey();
    if(keypressed == 'D')

```



```

    {
        goto Input_Ok;
    }
    if(keypressed != NO_KEY)
    {
        keypressed;
        count++;
        if(count>=0&&count<=20)
        {
            lcd.setCursor((9+a),2);
            lcd.print(keypressed);
        }
        a++;
        InputPengulangan+=keypressed;
        Pengulangan = InputPengulangan.toInt();
        Serial.print(Pengulangan);
        Jumlah = Pengulangan;
        if((keypressed == '*') || (keypressed == '#') || (keypressed
== 'C') || (keypressed == 'B') || (keypressed == 'A'))
        {
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("  INPUT SALAH  ");
            lcd.setCursor(0,1);
            lcd.print("  MOHON MENGGUNAKAN  ");
            lcd.setCursor(0,2);
            lcd.print("  ANGKA 0-9  ");
            InputPengulangan = "";
            keypressed="";
            a=0;
            delay(2500);
            goto Input_Pertama;
        }
    }
    goto Input_Lagi;
Input_Ok:
a=0;
lcd.clear( );
}

////////SUB PROGRAM PENGULANGAN////////////////////////////////////

void loop()
{

```

```

if(Serial.available(>0)
{
    char inChar = (char)Serial.read();
    dataIn += inChar;
    if (inChar == '\n') {parsing = true;}
}
if(parsing)
{
    parsingData();
    parsing=false;
    dataIn="";
}

Wait_Start:
if(digitalRead(Start)==HIGH)
{
    Serial.println("Start");
    goto Poin1;
}
Stop();
goto Wait_Start;
Poin1:
PertanyaanJenisGerakan();
PertanyaanKecepatan();
JumlahPengulangan();
lcd.clear();
Wait_Start2:
if(K==1)
{
    lcd.setCursor(0,0);
    lcd.print(" FLEKSI EKSTENSI ");
}
if(K==2)
{
    lcd.setCursor(0,0);
    lcd.print(" ADDUKSI HORIZONTAL ");
}
if(K==3)
{
    lcd.setCursor(0,0);
    lcd.print(" ADDUKSI VERTIKAL ");
}
if((kec==255)&&(K==1))
{

```



```

    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==200)&&(K==1))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN LAMBAT ");
}
if((kec==20)&&(K==2))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==40)&&(K==2))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN LAMBAT ");
}
if((kec==225)&&(K==3))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==150)&&(K==3))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN LAMBAT ");
}
lcd.setCursor(0,2);
lcd.print(" JUM. GERAKAN ");
lcd.setCursor(15,2);
lcd.print(Pengulangan);
lcd.setCursor(0,3);
lcd.print("START=0K  B=ULANG");
Stop();
char keypressed = myKeypad.getKey();
if(digitalRead(Start)==HIGH)
{
    Serial.println("Start");
    goto poin6;
}

else if (keypressed=='B')
{

```

```

a          =0;
Pengulangan =0;
InputPengulangan ="";
kec        =0;
k          =0;
pin1       =0;
pin2       =0;
goto Poin1;
l=0;
K=0;
}
goto Wait_Start2;
poin6:
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" STATUS = RUN ");
if((kec==255)&&(K==1))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==200)&&(K==1))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN LAMBAT ");
}
if((kec==20)&&(K==2))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==40)&&(K==2))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN LAMBAT ");
}
if((kec==225)&&(K==3))
{
    lcd.setCursor(0,1);
    lcd.print(" KECEPATAN SEDANG ");
}
if((kec==150)&&(K==3))
{
    lcd.setCursor(0,1);

```

```

    lcd.print(" KECEPATAN LAMBAT ");
}
if(K==1)
{
    lcd.setCursor(0,2);
    lcd.print(" FLEKSI EKSTENSI ");
}
if(K==2)
{
    lcd.setCursor(0,2);
    lcd.print(" ADDUKSI HORIZONTAL ");
}
if(K==3)
{
    lcd.setCursor(0,2);
    lcd.print(" ADDUKSI VERTIKAL ");
}
lcd.setCursor(0,3);
lcd.print(" SISA GERAKAN ");
lcd.setCursor(16,3);
lcd.print(Pengulangan);
S=0;
poin3:
if(S==1)
{
    goto poin6;
}
if((Pengulangan!=0) && (l==1))
{
    Serial.println("L1");
    MotorMundur();
    Stop();
    if(digitalRead(Limit_Min1)==HIGH)
    {
        Poin2:
        delay(150);
        Serial.println("Limit_Min1 ON");
        Stop();
        MotorMaju();
    }
    if(digitalRead(Limit_Max1)==HIGH)
    {
        delay(150);
        Serial.println("Limit_Max1 ON");
        Pengulangan--;
    }
}

```

```

        lcd.setCursor(16,3);
        lcd.print(Pengulangan);
        Serial.print(Pengulangan);
        goto poin3;
    }
    goto Poin2;
}
goto poin3;
}
if((Pengulangan!=0) && (l==2))
{
    Serial.println("L2");
    MotorKiri();
    MotorKanan();
    Pengulangan--;
    lcd.setCursor(16,3);
    lcd.print(Pengulangan);
    Serial.print(Pengulangan);
    goto poin3;
}
if((Pengulangan!=0) && (l==3))
{
    Serial.println("L3");
    MotorMundur();
    Stop();
    if(digitalRead(Limit_Min2)==HIGH)
    {
        Poin4:
        Serial.println("limit_Min2 ON");
        Stop();
        MotorMaju();
        if(digitalRead(Limit_Max2)==HIGH)
        {
            Serial.println("Limit_Max2 ON");
            Pengulangan--;
            lcd.setCursor(16,3);
            lcd.print(Pengulangan);
            Serial.print(Pengulangan);
            delay(150);
            goto poin3;
        }
        goto Poin4;
    }
}
goto poin3;

```

```

}
if((J_Pengulangan != 0)&&(getdata_JenisGerakan ==
"\Fleksi Ekstensi\")==&&(getdata_Kecepatan == "\Sedang\"))
{
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    pin1=PWM_3;
    pin2=PWM_4;
    kec = 255;
    MotorMundur();
    delay(6000);
    MotorMaju();
    delay(6000);
    J_Pengulangan--;
    Serial.println(J_Pengulangan);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" STATUS = RUN ");
    lcd.setCursor(0,1);
    lcd.print(" FLEKSI EKSTENSI ");
    lcd.setCursor(0,2);
    lcd.print(" KECEPATAN SEDANG ");
    lcd.setCursor(0,3);
    lcd.print(" SISA GERAKAN ");
    lcd.setCursor(16,3);
    lcd.print(J_Pengulangan);
}
if((J_Pengulangan != 0)&&(getdata_JenisGerakan ==
"\Fleksi Ekstensi\")==&&(getdata_Kecepatan == "\Lambat\"))
{
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    pin1=PWM_3;
    pin2=PWM_4;
    kec = 200;
    MotorMundur();
    delay(7300);
    MotorMaju();
    delay(7300);
    J_Pengulangan--;
    Serial.println(J_Pengulangan);
    lcd.clear();
}

```

```

        lcd.setCursor(0,0);
        lcd.print(" STATUS = RUN ");
        lcd.setCursor(0,1);
        lcd.print(" FLEKSI EKSTENSI ");
        lcd.setCursor(0,2);
        lcd.print(" KECEPATAN SEDANG ");
        lcd.setCursor(0,3);
        lcd.print(" SISA GERAKAN ");
        lcd.setCursor(16,3);
        lcd.print(J_Pengulangan);
    }
    if((J_Pengulangan != 0)&&(getdata_JenisGerakan ==
"\Adduksi Horizontal\")==&&(getdata_Kecepatan ==
"\Sedang\")==)
    {
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, HIGH);
        digitalWrite(LED3, LOW);
        kec = 20;
        myservo.write(45);
        MotorKiri();
        MotorKanan();
        J_Pengulangan--;
        Serial.println(J_Pengulangan);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print(" STATUS = RUN ");
        lcd.setCursor(0,1);
        lcd.print(" ADDUKSI HORIZONTAL ");
        lcd.setCursor(0,2);
        lcd.print(" KECEPATAN SEDANG ");
        lcd.setCursor(0,3);
        lcd.print(" SISA GERAKAN ");
        lcd.setCursor(16,3);
        lcd.print(J_Pengulangan);
    }
    if((J_Pengulangan != 0)&&(getdata_JenisGerakan ==
"\Adduksi Horizontal\")==&&(getdata_Kecepatan ==
"\Lambat\")==)
    {
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, HIGH);
        digitalWrite(LED3, LOW);
        kec = 40;
        myservo.write(45);

```



```

MotorKiri();
MotorKanan();
J_Pengulangan--;
Serial.println(J_Pengulangan);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" STATUS = RUN ");
lcd.setCursor(0,1);
lcd.print(" ADDUKSI HORIZONTAL ");
lcd.setCursor(0,2);
lcd.print(" KECEPATAN LAMBAT ");
lcd.setCursor(0,3);
lcd.print(" SISA GERAKAN ");
lcd.setCursor(16,3);
lcd.print(J_Pengulangan);
}
if((J_Pengulangan != 0)&&(getdata_JenisGerakan ==
"\Adduksi Vertikal\"")&&(getdata_Kecepatan ==
"\Sedang\""))
{
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, HIGH);
    pin1=PWM_2;
    pin2=PWM_1;
    kec = 225;
    MotorMundur();
    delay(8500);
    MotorMaju();
    delay(8500);
    J_Pengulangan--;
    Serial.println(J_Pengulangan);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" STATUS = RUN ");
    lcd.setCursor(0,1);
    lcd.print(" ADDUKSI VERTIKAL ");
    lcd.setCursor(0,2);
    lcd.print(" KECEPATAN SEDANG ");
    lcd.setCursor(0,3);
    lcd.print(" SISA GERAKAN ");
    lcd.setCursor(16,3);
    lcd.print(J_Pengulangan);
}

```

```

if((J_Pengulangan != 0)&&(getdata_JenisGerakan == "\"Adduksi
Vertikal\"")&&(getdata_Kecepatan == "\"Lambat\""))
{
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, HIGH);
    pin1=PWM_2;
    pin2=PWM_1;
    kec = 150;
    MotorMundur();
    delay(12000);
    MotorMaju();
    delay(12000);
    J_Pengulangan--;
    Serial.println(J_Pengulangan);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" STATUS = RUN ");
    lcd.setCursor(0,1);
    lcd.print(" ADDUKSI VERTIKAL ");
    lcd.setCursor(0,2);
    lcd.print(" KECEPATAN SEDANG ");
    lcd.setCursor(0,3);
    lcd.print(" SISA GERAKAN ");
    lcd.setCursor(16,3);
    lcd.print(J_Pengulangan);
}
if(J_Pengulangan == 0)
{
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    MotorStop();
    lcd.setCursor(0,0);
    lcd.print(" STATUS = DONE ");
    lcd.setCursor(0,1);
    lcd.print(" GUNAKAN APLIKASI ");
    lcd.setCursor(0,2);
    lcd.print(" UNTUK MENGINPUT ");
    lcd.setCursor(0,3);
    lcd.print(" KEMBALI ");
}
if(Pengulangan == 0)
{

```

```

MotorStop();
TungguPresD:
lcd.setCursor(0,0);
lcd.print(" PENGULANGAN TELAH ");
lcd.setCursor(0,1);
lcd.print(" SELESAI ");
lcd.setCursor(0,2);
lcd.print(" TEKAN D UNTUK ");
lcd.setCursor(0,3);
lcd.print(" MENGINPUT KEMBALI ");

char keypressed = myKeypad.getKey();
if(keypressed=='D')
{
  a          =0;
  Pengulangan =0;
  InputPengulangan ="";
  kec        =0;
  k          =0;
  pin1       =0;
  pin2       =0;
  l=0;
  K=0;
  goto Poin1;

}
goto TungguPresD;
}
}
////////PROGRAM UTAMA//////////

```

Lampiran 3 Program di NodeMCU

```

#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>

// Set these to run example.
#define FIREBASE_HOST "robot-rehabilitasi-default-rtbd.firebaseio.com"
#define FIREBASE_AUTH "dxWhMXL4hTXU4RlU1Xja9JneikFKUTCclmAmDiPI"
#define WIFI_SSID "Daftar Hitam"
#define WIFI_PASSWORD "123456788"
String getdata_JenisGerakan="";

```

```

String getdata_Kecepatan="";
String getdata_JumlahPengulangan="";
boolean kondisi;

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

  // connect to wifi.
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}

void loop() {
  kirimdata();
}

void kirimdata(){
  if(!kondisi)
  {
    getdata_JenisGerakan =
    Firebase.getString("Rehabilitasi_Database/JenisGerakan");
    getdata_Kecepatan =
    Firebase.getString("Rehabilitasi_Database/Kecepatan");
    getdata_JumlahPengulangan =
    Firebase.getString("Rehabilitasi_Database/JumlahPengulangan"
    );
    Serial.println ("*" + getdata_JenisGerakan + "," +
    getdata_Kecepatan + "," + getdata_JumlahPengulangan + "#");
    kondisi=true;
  }
  else
  {

if(getdata_JenisGerakan!=Firebase.getString("Rehabilitasi_Da
tabase/JenisGerakan")||getdata_Kecepatan !=
Firebase.getString("Rehabilitasi_Database/Kecepatan")||

```

```

        getdata_JumlahPengulangan !=
        Firebase.getString("Rehabilitasi_Database/JumlahPengulangan"
        )){
            getdata_JenisGerakan =
            Firebase.getString("Rehabilitasi_Database/JenisGerakan");
            getdata_Kecepatan =
            Firebase.getString("Rehabilitasi_Database/Kecepatan");
            getdata_JumlahPengulangan =
            Firebase.getString("Rehabilitasi_Database/JumlahPengulangan"
            );
            Serial.println ("*" + getdata_JenisGerakan + "," +
            getdata_Kecepatan + "," + getdata_JumlahPengulangan + "#");
        }
    }
}
}

```

Lampiran 4 Program di MIT App Inventor

The screenshot shows the MIT App Inventor code editor with several event-driven blocks:

- Initialize global (data) to create empty list**: A block to initialize a global list.
- when Screen1 - Initialize**: Contains a `set Web2 - Url to https://docs.google.com/spreadsheets/d/1Sm9uK0Md...` and a `call Web2 - Get` block.
- when Screen1 - BackPressed**: Contains a `close application` block.
- when Start - Click**: Contains a `call FirebaseDB1 - GetValue` block with a `tag Nama - Text` and a `valueIfTagNotThere` block.
- when Web2 - GotText**: Contains a `set global data to list from csv table text get responseC`, a `remove list item list get global data index 1`, and a `set ListView1 - Elements to get global data` block.
- when Web1 - GotText**: Contains a `call Web2 - Get` block, followed by `set Nama - Text to`, `set JumlahPengulangan - Text to`, `set Kecepatan - Selection to`, and `set JenisGerakan - Selection to` blocks.

The screenshot shows the MIT App Inventor code editor with several event-driven blocks:

- when Notification - AfterChoosing**: Contains a `choice` block, a `do if get Choice is 'K3'` block, and a `then` block with multiple `call FirebaseDB1 - StoreValue` blocks for `tag Nama - Text`, `tag JenisGerakan - Selection`, `tag Kecepatan - Selection`, and `tag JumlahPengulangan - Text`. It also includes a `set Web1 - Url to https://docs.google.com/forms/d/1OJ_Ga1268NGQ5X...` block and a `call Web1 - Get` block.
- when FirebaseDB1 - GetValue**: Contains a `do if value` block, a `call Toastert - ShowChooseDialog` block with `message`, `title Peningkatan`, `button1Text 'Ya'`, `button2Text 'Tidak'`, and `cancelable false`.

Lampiran 5 Dokumentasi



Lampiran 6 Biodata Penulis



Fikri Fatahillah. Lahir di Makassar pada tanggal 16 Juni 1999 dari ayah Abu Bakar dan ibu Sitti Djuhari. Penulis adalah anak pertama dari empat bersaudara. Tahun 2011 penulis menyelesaikan Pendidikan Sekolah Dasar di SD Inpres Tamalanrea IV. Pada tahun itu juga penulis melanjutkan Pendidikan di SMPN 30 Makassar dan tamat pada tahun 2014, kemudian melanjutkan Sekolah Menengah Atas di SMAN 21 Makassar dan lulus pada tahun 2017. Pada tahun 2017 penulis diterima sebagai mahasiswa program studi Teknik Mekatronika, Jurusan Teknik Mesin di Politeknik Negeri Ujung Pandang. Penulis pernah mengikuti Praktek Kerja Lapangan di PT. Pelabuhan Indonesia IV (Persero) Cabang Makassar, Kota Makassar, Provinsi Sulawesi Selatan pada tahun 2020.



Fitriani. Lahir di Sumpang Jeru pada tanggal 25 Desember 1998 dari ayah Aburahuddin dan ibu Diana. Penulis adalah anak pertama dari tiga bersaudara. Tahun 2011 penulis menyelesaikan Pendidikan Sekolah Dasar di SDN 70 Libukang. Pada tahun itu juga penulis melanjutkan Pendidikan di SMPN 1 Liliraja dan tamat pada tahun 2014, kemudian melanjutkan Sekolah Menengah Atas di SMAN 1 Liliraja dan lulus pada tahun 2017. Pada tahun 2017 penulis diterima sebagai mahasiswa program studi Teknik Mekatronika, Jurusan Teknik Mesin di Politeknik Negeri Ujung Pandang. Penulis pernah mengikuti Praktek Kerja Lapangan di PT. Pelabuhan Indonesia IV (Persero) Cabang Makassar *New Port*, Kota Makassar, Provinsi Sulawesi Selatan pada tahun 2020.

LAMPIRAN BERITA ACARA PELAKSANAAN
UJIAN SIDANG SKRIPSI

Nama Mahasiswa : Fikri Fatahillah/Titriani
NIM : 444 17 013/444 17 011

Catatan/Daftar Revisi Penguji:

No.	Nama	Uraian	Tanda Tangan
1	P. Kadir	- Perbaiki video pengujian dgn bahan	
2	Imra	- Sistem IoT harus connect ke HP desktop	
3	P. Kani	- Perbaikan bagian di jwbark pada skripsi - Absa pada skripsi	

Makassar, 07 September 2021
Sekretaris Penguji



Imran Habriansyah, S.ST., M.T.
NIP. 19881005 201903 1 009

Catatan: Jika ada perubahan Judul Tugas Akhir konfirmasikan secepatnya ke bagian Akademik.