



UNIVERSITAS INDONESIA

**SISTEM KONTROL LENGAN ROBOT MENGGUNAKAN SINYAL EMG
BERBASIS MIKROKONTROLER *H8/3069F***

SKRIPSI

YULIA NUR FITRIANA

0706262943

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

PROGRAM STUDI FISIKA

DEPOK

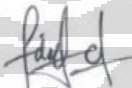
MEI 2012

HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : YULIA NUR FITRIANA

NPM : 0706262943

Tanda Tangan : 

Tanggal : 29 Mei 2012

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama : YULIA NUR FITRIANA
NPM : 0706262943
Program Studi : Instrumentasi Elektronika S1
Judul Skripsi :

SISTEM KONTROL LENGAN ROBOT MENGGUNAKAN SINYAL EMG
BERBASIS MIKROKONTROLER *H8/3096F*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sains pada Program Studi Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia

DEWAN PENGUJI

Pembimbing I : Prof. Dr. BEF Da Silva, M.Sc



Pembimbing II : Dr. Eng. Supriyanto, M.Sc



Penguji I : Dr. Lingga Hermanto, M.Si



Penguji II : Dr. Tony Mulia



Ditetapkan di : Depok

Tanggal : 29 Mei 2012

KATA PENGANTAR

Alhamdulillah puji syukur hanya kepada Allah SWT yang selalu memberikan anugrah terindah-Nya kepada penulis dan telah menuntun penulis dalam menyelesaikan skripsi ini dengan baik.

Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Sains Jurusan Fisika pada Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia.

Penulis menyadari bahwa, selesainya skripsi ini tidak terlepas dari bantuan, bimbingan, dorongan dan doa yang tulus dari banyak pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini. Tanpa itu semua sangatlah sulit bagi penulis untuk menyelesaikan skripsi ini. Untuk itu penulis ingin menyampaikan ucapan terima kasih yang setulus - tulusnya kepada:

1. Bapak BEF Da Silfa dan Pak Supri selaku dosen pembimbing yang telah membimbing penulis dalam segala hal, baik dalam ilmu pengetahuan, nasehat, dorongan, semangat serta segala fasilitas yang telah diberikan dalam menyelesaikan tugas akhir ini.
2. Pak Toni dan Pak Lingga selaku dosen penguji yang telah memberikan banyak masukan pada penulisan skripsi ini.
3. Pak Karno selaku pembimbing ketiga yang telah bersedia meluangkan waktu untuk diskusi dan bimbingan sehingga skripsi ini dapat selesai dengan baik.
4. Dosen-dosen instrumen: Pak Prawito, Pak Sastra, Pak Santoso, Pak Tony, Pak Marta, Pak Lingga, Pak Arif yang telah memberikan ilmu yang sangat bermanfaat.
5. Pak Katman dan Pak Budi yang telah meminjamkan, menyediakan, memberikan alat-alat dan komponen-komponen yang saya butuhkan selama penelitian. Pak Heri yang telah meminjamkan bukunya dan Pak Parno yang telah membuatkan alat.
6. Teman-teman seperjuangan dalam menyusun skripsi : Dewi Imas atas perhatiannya, diskusinya yang sangat bermanfaat dan selalu mengingatkan ini itu, Rusyda Taqiyya atas dorongan, ilmu dan

kebersamaannya disaat-saat genting (*p_dut*), Mergoramadhayenti atas arahan dan refreshing moment-nya, Nur Aisyah atas doa dan bantuannya dan Adi Prasetyo yang telah meminjamkan adaptornya selama pengerjaan tugas akhir ini. Oh iy Ami atau Fahmi nuklir 2008 makasih juga...

7. Semua anggota tim riset H8 : Ka Muchtar, Ka Ilham, Ka Tyo, Ka Thea, Ka Tiwi, Ka Nurma, Ka Moku, Ka Ami, Ka Rizki, Ka Lia, Ka Faizal, Nella, Harits, Fikri, Salas, Herlina, Yusuf, Faldo, Ferry, Vika, dan Cherry.
8. Teman-teman kosan : Leni, Sri, Ana, Fiki, Mba Ayu dan Mba Asmi. And special thanks to my lovely friends: Mba Sulis, Mba Nila, Putri Marlina dan Mba Rita. To former and later Momochan. I love you all...
9. Mbak Ratna dan Pak Dwi di sekret, terima kasih atas segala pelayanan yang sudah diberikan untuk kita yang bolak-balik ngurus berkas-berkas.

Semoga skripsi ini dapat berguna bagi siapa saja yang mengkajinya, serta dapat dikembangkan dan disempurnakan agar lebih bermanfaat untuk kepentingan orang banyak.

Depok, 29 Mei 2012

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Yulia Nur Fitriana
NPM : 0706262943
Program Studi : Instrumentasi Elektronika S1
Departemen : Fisika
Fakultas : Matematika dan Ilmu pengetahuan Alam
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

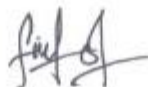
**SISTEM KONTROL Lengan ROBOT MENGGUNAKAN SINYAL EMG
BERBASIS MIKROKONTROLER H8/3069F**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalih media /formatkan, mengelola dalam bentuk pangkalan data (*Database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 29 Mei 2012

Yang menyatakan



(Yulia Nur Fitriana)

Nama : Yulia Nur Fitriana
Program studi : Fisika
Judul skripsi : Sistem Kontrol Lengan Robot Menggunakan Sinyal EMG
Berbasis Mikrokontroler *H8/3069F*

ABSTRAK

Rancangan sistem kontrol lengan robot dengan menggunakan sinyal elektromiogram (EMG) telah dibuat dengan elektroda permukaan sebagai transduser. Sinyal EMG diolah dengan sistem pengolahan sinyal dan diakuisisi dengan menggunakan mikrokontroler *H8/3069F*. Data pengamatan ditampilkan dalam bentuk *Graphical User Interface* (GUI) yang dibuat dengan bahasa pemrograman Python dan disimpan dalam database Microsoft Access. Kontrol lengan robot dilakukan berdasarkan gerakan fleksi-ekstensi pergelangan tangan. Sinyal EMG dikarakterisasi berdasarkan *root mean square* (RMS) sehingga sinyal EMG dapat diklasifikasikan. Gerakan fleksi memiliki RMS antara 0.01 - 0.13 V dan gerakan ekstensi memiliki RMS antara 0.69 - 1.19 V. Sinyal EMG yang telah diklasifikasi ini digunakan sebagai input untuk mengontrol servo motor pada lengan robot.

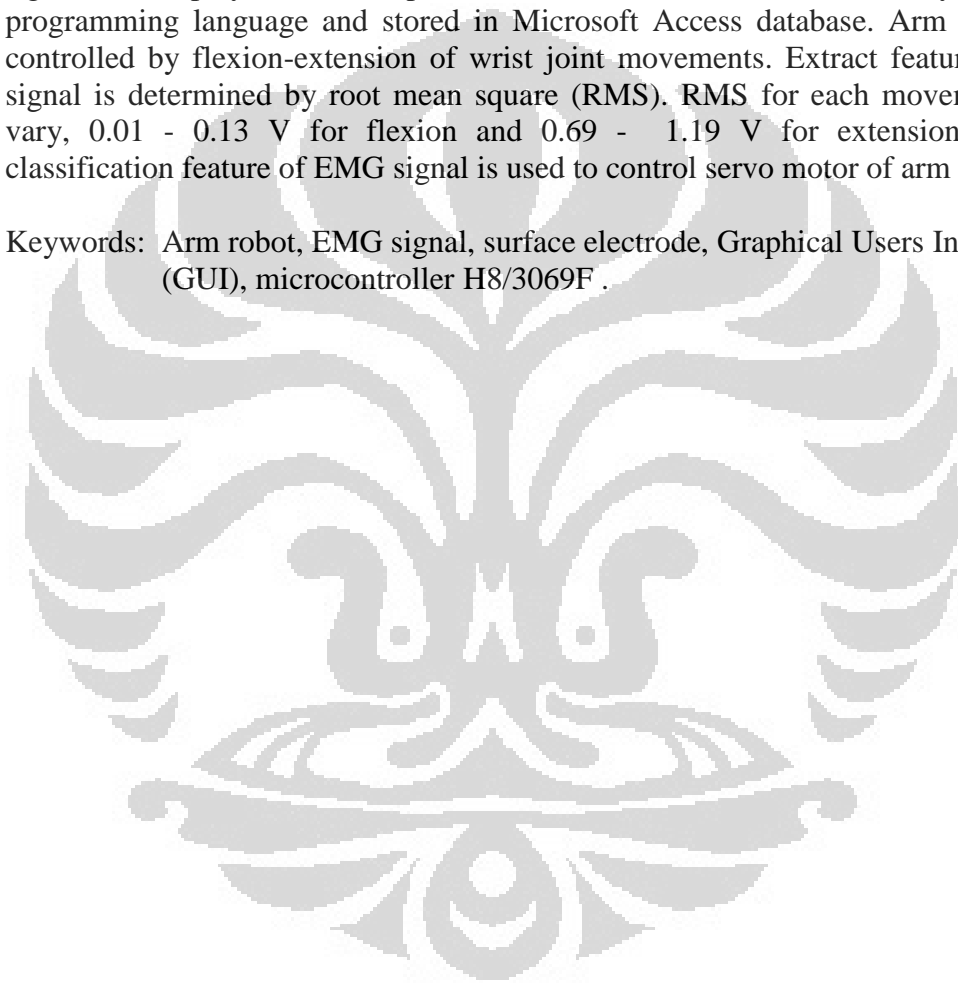
Kata kunci : lengan robot, sinyal EMG, elektroda permukaan, *root mean square*, *Graphical Users Interface* (GUI), mikrokontroler *H8/3069F*.

Name : Yulia Nur Fitriana
Program study : Physics
Title of essay : Arm Robot Control System Using EMG Signal Based
Microcontroller H8/3069F

ABSTRACT

Designing control system of arm robot using electromyogram (EMG) signal have been made with surface electrode as transducer. EMG signal is processed by signal conditioning system dan acquired by microcontroller H8/3069F . Recording EMG signal is displayed on Graphical User Interface (GUI) with Python as programming language and stored in Microsoft Access database. Arm robot is controlled by flexion-extension of wrist joint movements. Extract feature EMG signal is determined by root mean square (RMS). RMS for each movements is vary, 0.01 - 0.13 V for flexion and 0.69 - 1.19 V for extension. These classification feature of EMG signal is used to control servo motor of arm robot.

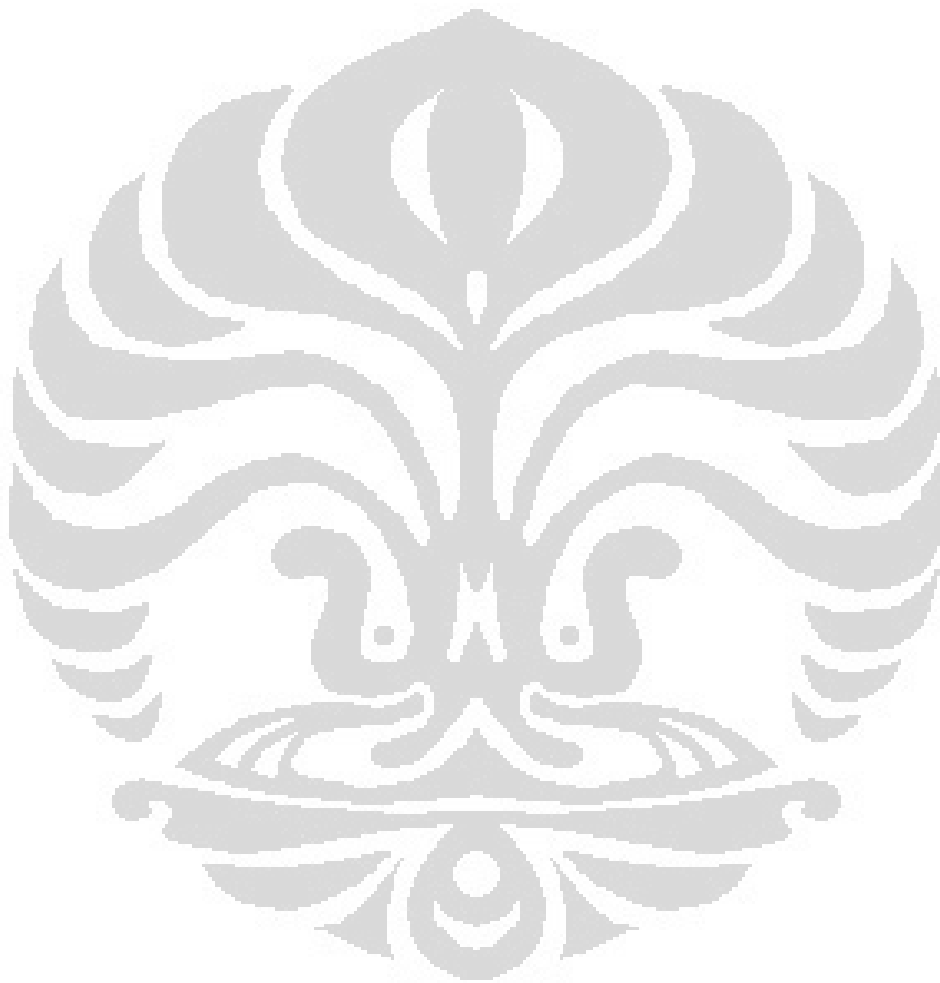
Keywords: Arm robot, EMG signal, surface electrode, Graphical Users Interface (GUI), microcontroller H8/3069F .



DAFTAR ISI

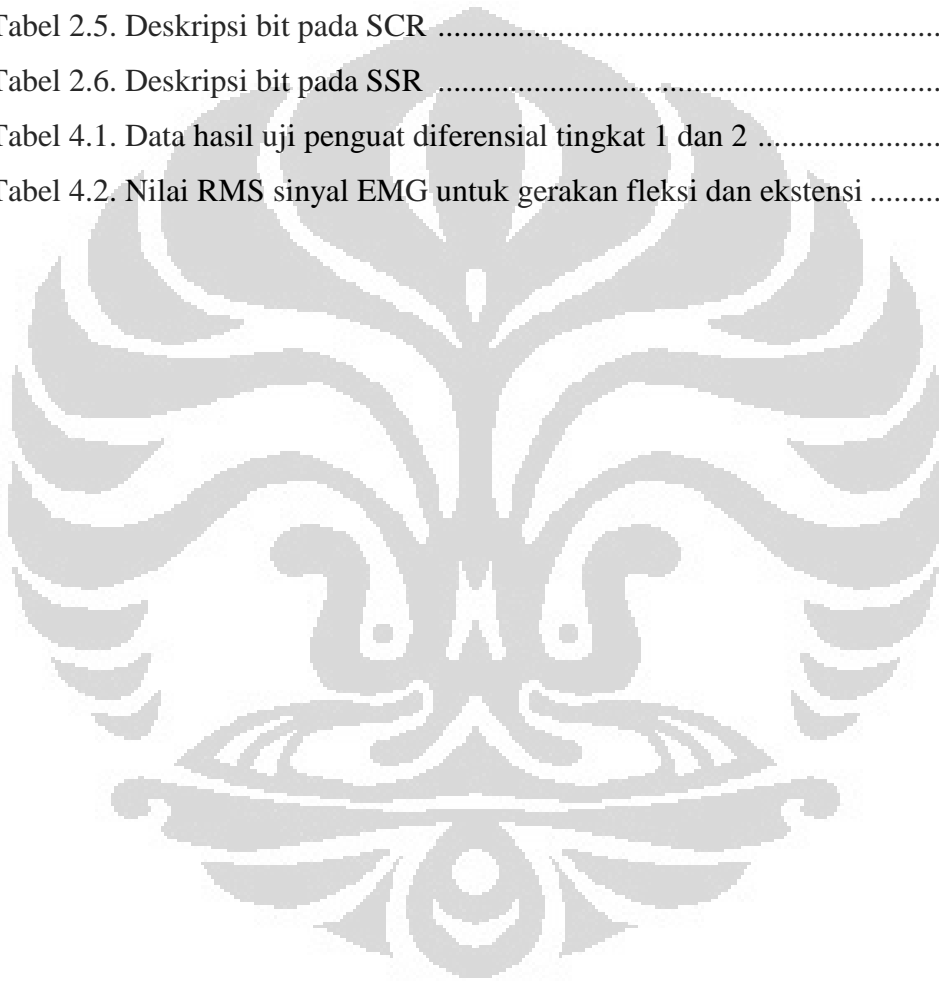
	Halaman
HALAMAN JUDUL.....	i
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH	v
ABSTRAK.....	vi
<i>ABSTRACT</i>	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN.....	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Batasan Masalah	2
1.3 Tujuan dan Manfaat Penelitian.....	2
1.4 Metodologi Penelitian.....	2
1.5 Sistematika Penulisan	5
BAB 2 LANDASAN TEORI	6
2.1 Lengan Robot	6
2.1.1 <i>Degrees Of Freedom (DOF)</i>	6
2.1.2 Servo Motor	7
2.2 Sinyal EMG	8
2.2.1 Sumber Biopotensial	9
2.2.2 Elektroda Permukaan	12
2.2.3 <i>Interface</i> Elektroda-Jaringan	14
2.2.4 <i>Motion Artifact</i>	14
2.2.5 Anatomi Otot Tangan	14
2.3 Mikrokontroler <i>H8/3069F</i>	14
2.3.1 <i>Analog to Digital Converter</i>	14
2.3.2 <i>PWM Mode</i>	14
2.3.3 <i>Serial Communication Interface</i>	14
2.4 RS 232	14
BAB 3 PERANCANGAN ALAT	16
3.1 Perangkat Keras	16
3.1.1 Elektroda Permukaan.....	19
3.1.2 Rangkaian Pengolah Sinyal.....	19
3.1.3 Perancangan Lengan Robot.....	23
3.2 Perangkat Lunak	25
3.2.1 Pemrograman pada mikrokontroler <i>H8/3069F</i>	25
3.2.2 Tampilan GUI Python	25
BAB 4 ANALISA HASIL PENELITIAN	28
4.1 Hasil Penelitian	28
4.1.1 Hasil Uji Rangkaian Penguat Diferensial.....	38
4.1.2 Hasil Uji Rangkaian Filter.....	39
4.1.3 Hasil Uji Rangkaian <i>Direct Adder</i>	40

4.1.4 Hasil Uji Kontrol Motor Servo.....	40
4.1.5 Sinyal EMG.....	41
4.2 Analisis.....	32
BAB 5 PENUTUP	34
5.1 Kesimpulan	34
5.2 Saran	34
DAFTAR PUSTAKA	35
LAMPIRAN	37



DAFTAR TABEL

Tabel 2.1. Deskripsi bit pada ADDRA, B, C dan D	15
Tabel 2.2. Deskripsi pasangan input kanal analog dan A/D data Register.....	15
Tabel 2.3. Deskripsi bit pada ADCSR.....	15
Tabel 2.4. Deskripsi bit pada SMR	18
Tabel 2.5. Deskripsi bit pada SCR	18
Tabel 2.6. Deskripsi bit pada SSR	19
Tabel 4.1. Data hasil uji penguat diferensial tingkat 1 dan 2	38
Tabel 4.2. Nilai RMS sinyal EMG untuk gerakan fleksi dan ekstensi	42

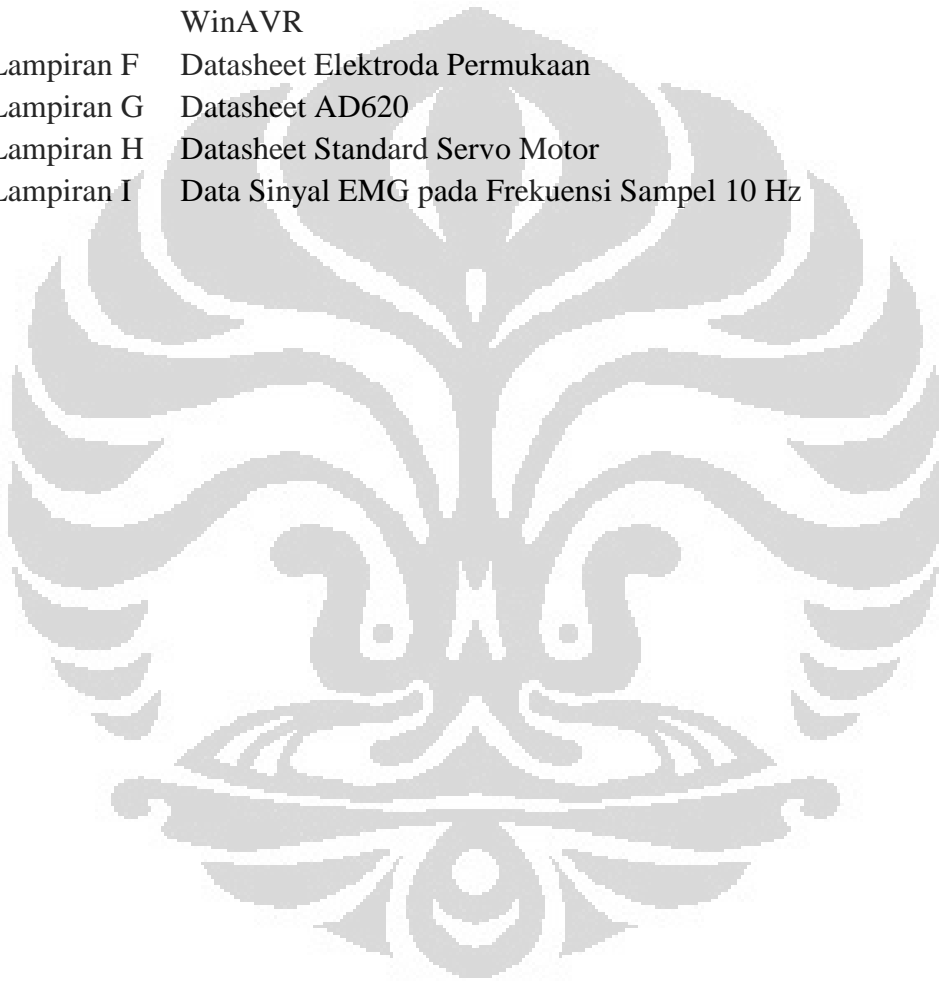


DAFTAR GAMBAR

Gambar 1.1.	Diagram alir langkah-langkah penelitian.....	3
Gambar 2.1.	Lengan robot 5 DOF menurut FBD.....	7
Gambar 2.2.	Kontrol motor dengan PWM	7
Gambar 2.3.	Proses depolarisasi dan repolarisasi di membran sel	10
Gambar 2.4.	Bagian <i>motor unit</i>	11
Gambar 2.5.	Superposisi sinyal EMG.....	11
Gambar 2.6.	Rangkaian ekivalen interface elektroda - jaringan.....	12
Gambar 2.7.	Rangkaian ekivalen interface elektroda – jaringan untuk pasangan elektroda	13
Gambar 2.8.	(a) Posterior dan (b) Anterior otot lengan bawah.....	14
Gambar 2.9.	(a) Mode PWM dengan counter di-clear-kan oleh GRA, b) Mode PWM dengan counter di-clear-kan oleh GRB.....	17
Gambar 2.10.	Format data komunikasi asinkron.....	18
Gambar 2.11.	Port DB9 <i>female</i>	20
Gambar 3.1.	Blok diagram sistem kontrol lengan robot.....	21
Gambar 3.2.	Elektroda permukaan	22
Gambar 3.3.	Lokasi penempatan elektroda yang tepat.....	22
Gambar 3.4.	Konfigurasi penguat diferensial. Sinyal EMG ditunjukkan oleh ' <i>m</i> ' dan noise ditunjukkan oleh ' <i>n</i>	23
Gambar 3.5.	Rangkaian skematik AD620	23
Gambar 3.6.	Rangkaian <i>high pass filter orde 2</i>	25
Gambar 3.7.	Rangkaian <i>low pass filter orde 2</i>	26
Gambar 3.8.	Rangkaian <i>direct adder</i>	26
Gambar 3.9.	Rangkaian pengolah sinyal yang terdiri dari penguat diferensial, <i>high pass filter</i> , <i>low pass filter</i> dan <i>direct adder</i>	27
Gambar 3.10.	Lengan robot	28
Gambar 3.11.	Kondisi lengan robot saat torsi pada M_1 maksimum.....	28
Gambar 3.12.	Diagram alir akuisisi sinyal EMG dan kontrol motor.....	31
Gambar 3.13.	Diagram alir konversi A/D.....	32
Gambar 3.14.	Diagram alir komunikasi serial asinkron	33
Gambar 3.15.	Diagram alir kontrol motor dengan menggunakan PWM mode ..	34
Gambar 3.16.	Tampilan sinyal EMG untuk frekuensi sampel 1kHz.....	36
Gambar 3.17.	Tampilan GUI (<i>Graphical User Interface</i>).....	37
Gambar 4.1.	<i>Display</i> keluaran penguat diferensial.....	39
Gambar 4.2.	Grafik respon <i>high pass filter</i>	39
Gambar 4.3.	Grafik respon <i>low pass filter</i>	40
Gambar 4.4.	<i>Display</i> input dan output <i>direct adder</i>	40
Gambar 4.5.	Posisi servo motor untuk setiap <i>delay</i> yang diberikan.....	41
Gambar 4.6.	Posisi tangan pada saat pengambilan data.....	41
Gambar 4.7.	Data sinyal EMG.....	42

DAFTAR LAMPIRAN

- Lampiran A Tabel Data Amplitudo Sinyal EMG untuk Gerakan Fleksi dan Ekstensi
- Lampiran B Pemrograman C Untuk Akuisisi Data Dan Kontrol Lengan Robot Pada Mikrokontroler
- Lampiran C Pemrograman GUI Python
- Lampiran D Pengaturan Perintah *make* dan *clean* Programmer Notpad di WinAVR
- Lampiran F Datasheet Elektroda Permukaan
- Lampiran G Datasheet AD620
- Lampiran H Datasheet Standard Servo Motor
- Lampiran I Data Sinyal EMG pada Frekuensi Sampel 10 Hz



BAB 1

PENDAHULUAN

1.1. Latar Belakang Masalah

Penggunaan robot dalam kehidupan sehari-hari terus meningkat karena robot memiliki banyak keunggulan. Robot memiliki tingkat ketelitian dan produktivitas kerja yang tinggi sehingga dapat meningkatkan volume produksi di suatu industri. Robot mampu bekerja dalam operasi-operasi yang dianggap berisiko tinggi bagi manusia. Selain itu, robot mampu mengerjakan proses secara terus-menerus yang tidak mampu dilakukan oleh manusia [1].

Lengan robot adalah salah satu bentuk produksi dalam bidang robotika. Lengan robot dirancang menyerupai tangan manusia agar dapat berfungsi sebagai organ tangan. Lengan robot dilengkapi dengan actuator dan memiliki 3 *Degree of Freedom* (DOF) atau lebih. Bagian ujung lengan robot disebut dengan *end-effector* [2]. Bagian ini bisa berupa perangkat pengelasan, pengecatan, peralatan mesin atau *gripper* (berfungsi untuk menggenggam). Untuk melaksanakan fungsinya lengan robot dapat dikontrol dengan menggunakan mikrokontroler oleh pengguna.

Lengan robot ini dapat dikontrol oleh sinyal elektromiogram (EMG). Sinyal EMG adalah sinyal yang dihasilkan oleh aktivitas otot tubuh manusia. Sinyal yang dihasilkan oleh otot tangan dapat diakuisisi dengan menggunakan mikrokontroler dan digunakan untuk mengontrol lengan robot sehingga lengan robot dapat melakukan gerakan sesuai dengan gerakan tangan. System control lengan robot menggunakan sinyal EMG ini telah banyak dikembangkan dalam bidang penerbangan dan dalam bidang medis untuk pasien amputasi.

Hal ini membuat peneliti tertarik untuk merancang system lengan robot dengan input sinyal EMG sebagai pengontrolnya. Pengontrolan lengan robot dengan menggunakan mikrokontroler *H8/3069F* telah dilakukan oleh Britantyo Witcaksono (2003). System lengan robot ini dapat dikontrol secara otomatis atau manual dengan menggunakan komputer desktop. Peneliti menambahkan input berupa sinyal EMG dari otot tangan ke dalam system control lengan robot.

Lengan robot ini dapat bergerak sesuai dengan gerakan tangan. Gerakan tangan yang akan dikontrol yaitu gerakan fleksi-ekstensi pergelangan tangan.

1.2. Batasan Masalah

Lengan robot dirancang untuk memiliki 3 DOF dan menggunakan aktuator berupa standar servo motor yang dikendalikan oleh PWM. Lengan robot dikontrol oleh sinyal EMG pada otot *extensor carpi radialis longus* dan *flexor carpi ulnaris* yaitu otot yang digunakan untuk mengontrol gerakan fleksi-ekstensi pergelangan tangan. Sinyal EMG dideteksi dengan surface elektroda kemudian ditampilkan di komputer dengan menggunakan GUI Python dan diklasifikasikan dengan menggunakan bahasa C. Lengan robot dapat melakukan gerakan sesuai dengan gerakan tangan manusia yaitu fleksi-ekstensi.

1.3. Tujuan dan Manfaat Penelitian

Penelitian bertujuan untuk:

- a. Menampilkan sinyal EMG dengan menggunakan GUI python
- b. Merancang lengan robot yang dapat dikontrol oleh sinyal EMG sehingga bisa melakukan gerakan sesuai dengan gerakan yang diinginkan.

1.4. Metodologi Penelitian

Metode penelitian yang akan dilakukan untuk membantu pelaksanaan, pembuatan, dan penganalisaan alat ini meliputi :

- a. Studi Literatur

Untuk memperoleh landasan teori dalam penelitian dan pembuatan alat maka tahap pertama yang harus dilakukan adalah mengumpulkan bermacam-macam informasi yang berkaitan dengan sinyal EMG dan system kontrol lengan robot. Adapun sumber media yang digunakan yaitu buku-buku acuan, jurnal-jurnal, artikel-artikel, serta informasi yang diperoleh dari internet.

- b. Pembuatan *Hardware* dan *Software*

Pembuatan lengan robot 3 DOF dengan bahan acrilik dan servo motor DC. Perancangan dan pembuatan rangkaian pengolah sinyal yang terdiri dari instrumentasi amplifier, *filter* dan *direct adder*. Kemudian, pembuatan program diisikan dari komputer ke mikrokontroler *H8/3069F* sehingga sinyal EMG mampu mengontrol lengan robot

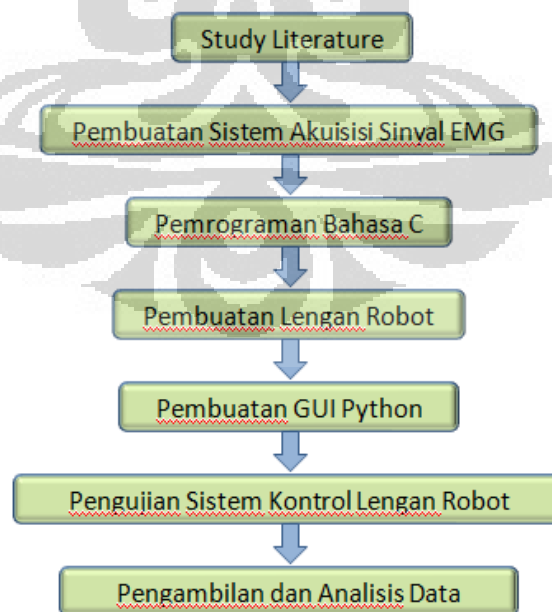
c. Pengujian Alat dan Pengambilan Data

Pengujian rangkaian pengolahan sinyal dilakukan dengan menggunakan *oscilloscop*. Pengambilan data berupa sinyal EMG dilakukan untuk setiap jenis gerakan yaitu fleksi dan ekstensi pergelangan tangan. Sinyal EMG ditampilkan dengan menggunakan GUI Python untuk mempelajari karakteristik sinyal EMG.

d. Analisis

Peneliti menganalisis karakteristik sinyal EMG masing-masing gerakan. Peneliti juga menganalisis keakuratan kontrol lengan robot terhadap sinyal EMG.

Berikut ini merupakan skematik langkah-langkah penelitian yang dilakukan dan skematik sistem pengolahan data:



Gambar 1.1. Diagram alir langkah-langkah penelitian

Penelitian ini dimulai *study literature* tentang akuisisi sinyal EMG, dan pengontrolan standar servo motor. Langkah selanjutnya, perancangan dan pengujian rangkaian pengolah sinyal untuk akuisisi sinyal EMG. Rangkaian pengolah sinyal yang dibutuhkan terdiri dari amplifier, filter dan *direct adder*. Keluaran pengolah sinyal masih dalam bentuk sinyal analog. Agar dapat ditampilkan di PC (Personal Computer), sinyal EMG dikonversi menjadi sinyal digital dengan menggunakan Mikrokontroler *H8/3069F*. Sinyal EMG ditampilkan dengan menggunakan GUI Python.

Perancangan lengan robot dapat dilakukan dengan menggunakan bahan akrilik dan dilengkapi dengan standar servo motor. Lengan robot ini akan dikontrol dengan menggunakan PWM oleh mikrokontroler. Sinyal EMG yang telah diperoleh akan diklasifikasikan dengan menggunakan bahasa C sesuai dengan gerakan yang diinginkan. Setelah itu, lengan robot dapat dikontrol dengan sinyal EMG oleh mikrokontroler.

Langkah terakhir adalah pengambilan data dan analisis keakuratan kontrol lengan robot dengan sinyal EMG. Lengan robot digerakkan sesuai dengan gerakan tangan. Jenis gerakan yang diujikan adalah fleksi-ekstensi.

1.5. Sistematika Penulisan

Sistematika penulisan skripsi terdiri atas lima bab yang secara garis besar dapat diuraikan sebagai berikut:

1. Bab 1 Pendahuluan

Bab ini memuat tentang latar belakang dari penelitian, tujuan, metode yang digunakan, dan juga pembatasan masalah pada penelitian yang dilakukan.

2. Bab 2 Landasan Teori

Bab ini memuat secara garis besar teori dasar yang berhubungan dengan penelitian.

3. Bab 3 Perancangan Alat

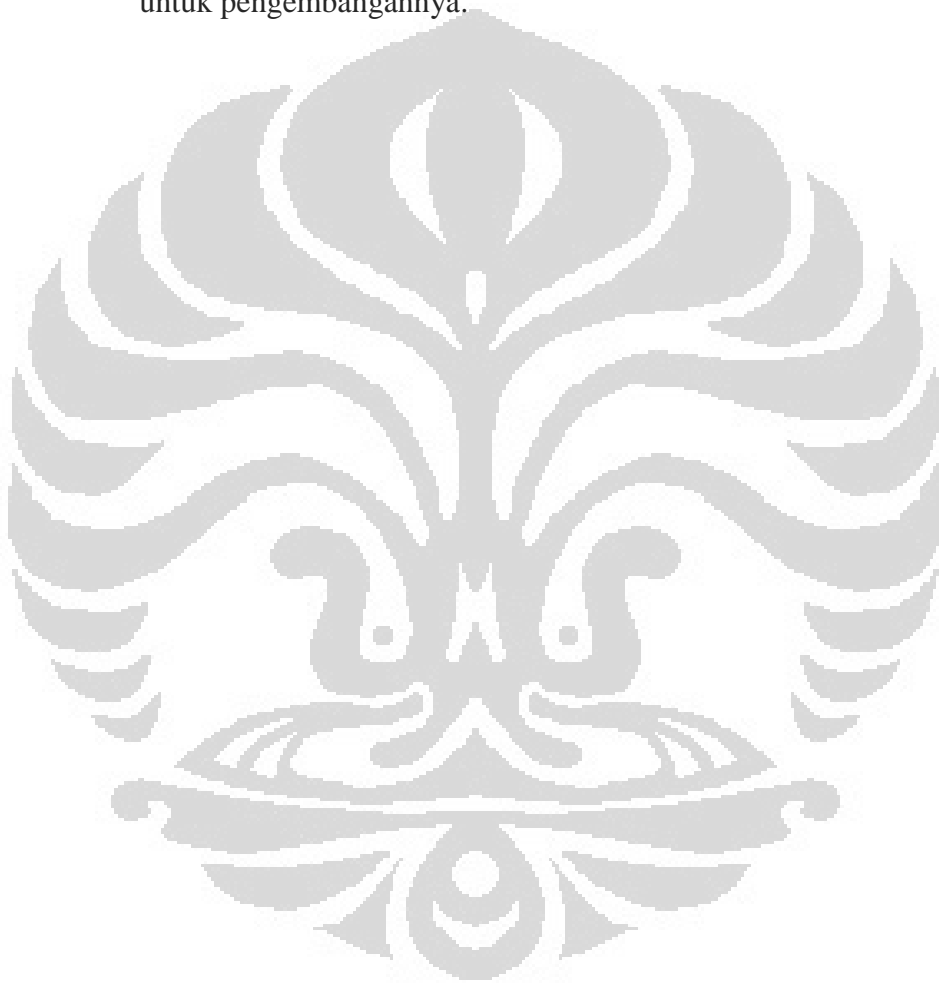
Bab ini memuat penggunaan perangkat keras dan perangkat lunak yang mendukung aplikasi Python pada sistem pemantauan proses pengeboran dengan menggunakan Mikrokontroler *H8/3069F*.

4. Bab 4 Analisis Hasil Penelitian

Bab ini berisi penjelasan mengenai hasil penelitian sensor pengeboran dan juga analisis dari sistem pemantauan proses pengeboran yang telah dibuat.

5. Bab 5 Penutup

Bab ini berisi kesimpulan atas hasil analisis dan saran yang mendukung penelitian agar memberikan hasil yang lebih baik lagi untuk pengembangannya.



BAB 2

LANDASAN TEORI

2.1. Lengan Robot

Lengan robot terdiri dari tiga bagian yaitu struktur mekanik (manipulator), penggerak dan system kontrol. Manipulator adalah susunan benda-benda kaku (*rigid bodies*) dan lengan (*link*) yang satu sama lain terhubung oleh sendi (*joint*). Pangkal lengan dapat dipasang pada kerangka dasar. Sedangkan ujung lengan (*end-effector*) dapat dihubungkan dengan alat tertentu sesuai dengan fungsi lengan robot [3].

Lengan robot dapat dikontrol dengan menggunakan sensor dan aktuator. Ada beberapa jenis aktuator yaitu aktuator listrik, hidrolik, pneumatik dan piezoelektrik. Masing-masing jenis aktuator memiliki tingkat kendali yang berbeda. Contohnya, aktuator listrik lebih mudah dikendalikan. Aktuator listrik memiliki beberapa kelebihan yaitu akurasi tinggi, torsi yang ideal untuk pergerakan dan tingkat efisiensi yang tinggi [4].

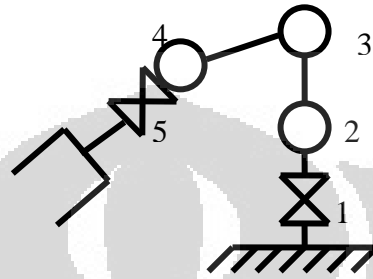
Ada dua jenis system kontrol lengan robot, yaitu system kontrol loop terbuka dan system kontrol loop tertutup. Pada system kontrol loop terbuka output tidak diumpan balikkan ke input [5].

2.1.1. *Degree of Freedom*

Derajat kebebasan (*Degree of Freedom*) adalah sambungan pada lengan, dapat dibengkokkan, diputar, maupun digeser. Derajat kebebasan digunakan untuk mengetahui cara robot bergerak, tingkat kerumitan algoritma kendali dan jumlah motor lengan robot yang digunakan. Penentuan jumlah DOF dilakukan berdasarkan jumlah gerakan yang dapat dilakukan oleh lengan robot atau jumlah aktuator lengan robot [6].

Ruang kerja robot atau ruang jangkauan robot adalah semua tempat yang dapat dijangkau oleh *end effector*. Hal ini tergantung pada sudut derajat kebebasan dan panjang jangkauan lengan. Ruang kerja ini juga bergantung pada konfigurasi robot yang dibuat.

Kaidah Denavit-Hartenberg merupakan aturan yang digunakan dalam perancangan robot yang diperkenalkan oleh Jaques Denavit dan Ricard S Hartenberg. Aturan tersebut menyatakan hanya terdapat dua gerakan yang mungkin terjadi yaitu bergeser dan berputar serta hanya terdapat 3 sumbu yang dapat terjadi yaitu sumbu x,y dan z. Contoh dari penggambaran lengan robot 5 DOF dengan menggunakan metode FBD dapat dilihat pada gambar 2.1.



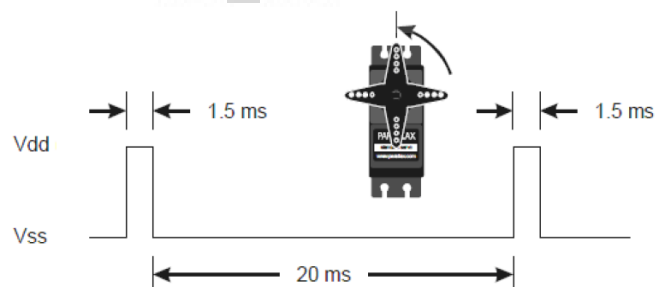
Gambar 2.1 Lengan robot 5 DOF menurut FBD

Pada *end-effector* dapat ditambahkan satu derajat kebebasan berupa *gripper*. Masing-masing sendi dapat berputar 180° atau 360° sesuai dengan fungsi dan aktuator yang diinginkan.

2.1.2. Servo Motor

Servo motor mempunyai keluaran sebuah poros dengan torsi besar. Poros ini dapat dikontrol melalui modulasi lebar pulsa. Poros dapat diatur untuk bertahan pada posisi tertentu, berputar *clock wise* atau *counter clock wise* [6].

Servo motor terdiri dari servo *continuous* dan servo standar. Servo motor continuous dapat berputar 360, sedangkan servo standar dapat berputar 180. Berikut ini adalah gambar pengaturan lebar pulsa yang diberikan pada servo standar.



Gambar 2.2 Kontrol motor dengan PWM

Berdasarkan Gambar 2.2 servo standar motor membutuhkan *delay* 1.5 ms untuk logika 1 dan 20 ms untuk logika 0 pada putaran *counter clock wise*.

2.2. Sinyal EMG

Dalam rangkaian pasif, listrik dihasilkan oleh elektron bebas. Sedangkan dalam jaringan biologis, listrik dihasilkan oleh ion yang berada dalam larutan elektrolit. Ada beberapa hukum yang mengatur peristiwa listrik oleh ion dalam larutan yaitu *Fick's law for diffusion*, *Particle drift*, dan *The Einstein relationship* [7].

1. *Fick's Law for Diffusion*

Konsentrasi partikel merupakan fungsi dari posisi. Dalam larutan terdapat region yang memiliki konsentrasi yang lebih tinggi dari region lain. System selalu bergerak menuju kesetimbangan, akibatnya terjadi difusi partikel dari region berkonsentrasi tinggi menuju region berkonsentrasi rendah[8].

$$J = -D \frac{d[C]}{dx} \quad (1)$$

J adalah kerapatan arus dalam satuan (A/m^2). [C] adalah konsentrasi partikel sebagai fungsi jarak dalam satuan (mol/liter). D adalah konstanta difusi atau *diffusivity* dalam satuan liter amper per mol liter ($L \cdot A/mol \cdot m$) [8].

2. *Drift Particle*

Partikel bermuatan atau ion mendapat gaya medan listrik sehingga partikel akan bergerak di bawah pengaruh gaya tarik menarik. Aliran ion ini disebut *drift current*. Arus drift berbanding lurus dengan tegangan, jumlah muatan dalam ion dan konsentrasi. Hubungan ini dapat ditunjukkan oleh persamaan sebagai berikut:

$$J_{drift} = -\mu Z \frac{dV}{dx} [C] \quad (2)$$

μ adalah mobility yang diberikan dalam liter ampe per volt meter mol. Sedangkan $-dV/dx$ merupakan medan listrik, E, diberikan dalam volt per meter.

3. *The Einstein Relationship*

Kedua konstansta pada persamaan di atas yaitu *diffusivity* dan *mobility* saling berhubungan satu sama lain. *Diffusivity* dan *mobility* sama-sama dipengaruhi oleh

temperatur. Hubungan ini dikenal dengan *Einstein relationship*. Hubungan ini ditunjukkan oleh persamaan sebagai berikut:

$$\frac{D}{\mu} = \frac{kT}{q} \quad (3)$$

Dengan k adalah konstanta Boltzman, q adalah muatan dan T adalah temperatur absolute [8].

2.2.1. Sumber Biopotensial Listrik

Biopotensial adalah tegangan listrik yang dihasilkan oleh perpindahan ion melalui jaringan biologis. Pada sel tubuh terdapat membran yang berfungsi sebagai filter ion. Bagian eksternal sel terdapat cairan yang konduktif terhadap ion sodium (Na^+), potassium (K^+) dan klorida (Cl^-). Sifat membrane sel yang permeable terhadap ion potassium menyebabkan ketidakseimbangan konsentrasi ion. Jumlah ion sodium di bagian eksternal sel lebih banyak daripada di bagian internal sel. Pengukuran biopotensial dilakukan dari bagian internal sel terhadap eksternal sel sehingga potensial ini bernilai negative. Pada saat ini potensial membran disebut potensial istirahat. Sel dalam kondisi ini disebut terpolarisasi. [9] Dalam beberapa penelitian dilaporkan bahwa potensial istirahat berkisar antara -60 mV sampai -100 mV [10].

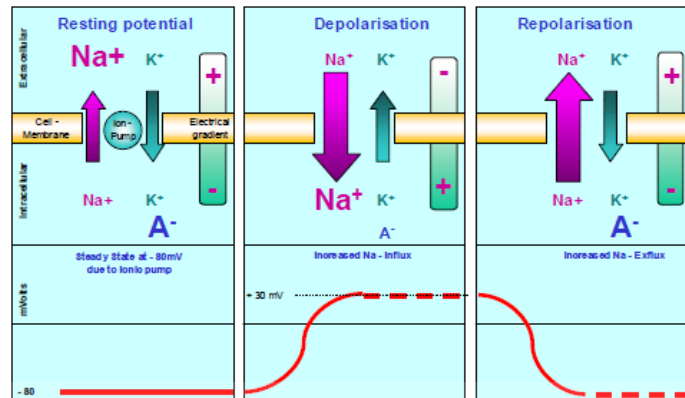
Potensial istirahat sel dapat ditentukan dengan menggunakan persamaan Goldman yaitu sebagai berikut:

$$V_m = -\frac{kT}{q} \ln \left(\frac{P_K [K^+]_i + P_{Na} [Na^+]_i + P_{Cl} [Cl^-]_o}{P_K [K^+]_o + P_{Na} [Na^+]_o + P_{Cl} [Cl^-]_i} \right) \quad (4)$$

Persamaan di atas menunjukkan bahwa potensial istirahat dipengaruhi oleh temperature. Kemudian persamaan di atas direduksi menjadi sebagai berikut:

$$V_m = -\frac{kT}{q} \ln \left(\frac{P_K [K^+]_i}{P_K [K^+]_o} \right) \quad (5)$$

Persamaan di atas disebut dengan *Nernst's equation*. berdasarkan hasil perhitungan persamaan Nernst lebih akurat 9.2% dibandingkan dengan persamaan Goldman. Hal ini menunjukkan bahwa potensial istirahat didominasi oleh aliran ion potassium [7].

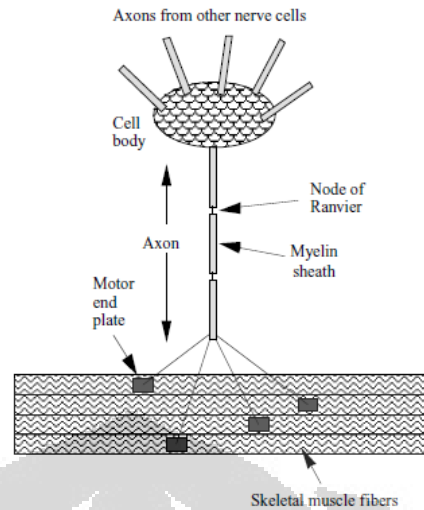


Gambar 2.3 Proses depolarisasi dan repolarisasi di membran sel

Pada saat membran sel terstimulasi oleh aliran ion maka membran akan bersifat permeable terhadap ion sodium. Sodium bergerak ke dalam sel untuk mencapai kesetimbangan dengan ion yang ada di luar sel. Ion potassium bergerak ke luar sel. Namun kemampuan potassium untuk melewati membrane tidak secepat sodium. Akibatnya terjadi ketidakseimbangan konsentrasi potassium. Potensial ini disebut dengan potensial aksi yang bernilai kira-kira 20mV. Proses perubahan potensial istirahat menjadi potensial aksi disebut depolarisasi.

Potensial sel selalu sama untuk setiap sel, tinggi total potensial aksi didefinisikan sebagai perbedaan potensial aksi dan potensial istirahat [10]. Melalui proses yang disebut pompa sodium, ion sodium akan bergerak keluar sel sehingga kembali ke keadaan potensial istirahat. Proses ini disebut dengan repolarisasi [9]. Jumlah pemompaan ini sebanding dengan konsentrasi ion sodium di dalam sel. Proses ini juga diikuti dengan perpindahan ion potassium ke dalam sel [10].

Potensial aksi menunjukkan adanya aktivitas otot atau kontraksi otot [9]. Kontraksi otot dihasilkan oleh aktivitas sejumlah *motor unit*. *Motor unit* adalah unit terkecil dari otot yang dapat dikontrol oleh system saraf. *Motor unit* terdiri dari sel tubuh, dendrit, saraf motorik, akson, dan serat otot seperti pada Gambar 2.4 [11]. Pada saat potensial aksi sampai pada *motor end plate* (ujung sel saraf), sejumlah asetilkolin dibebaskan, yang menghasilkan sebuah potensial aksi dalam sel otot. Masing-masing sel otot menghantarkan potensial aksi sama sebagaimana dihantarkan oleh akson [12].

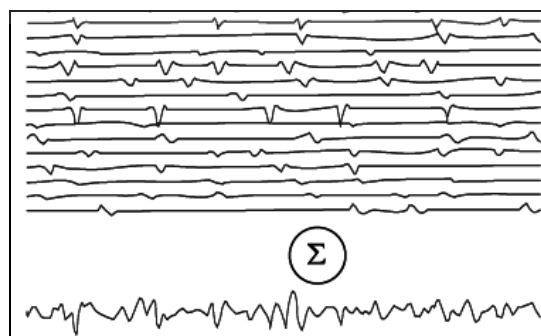
Gambar 2.4 Bagian *motor unit*

Instrumen dan teknik yang meliputi pengembangan, perekaman dan analisis potensial aksi ini disebut dengan elektromiograf [11][7]. Sedangkan potensial aksi yang terukur pada elektromiograf disebut elektromiogram (EMG) [7][9]. Bentuk gelombang sinyal EMG berupa gelombang *random noise* [10].

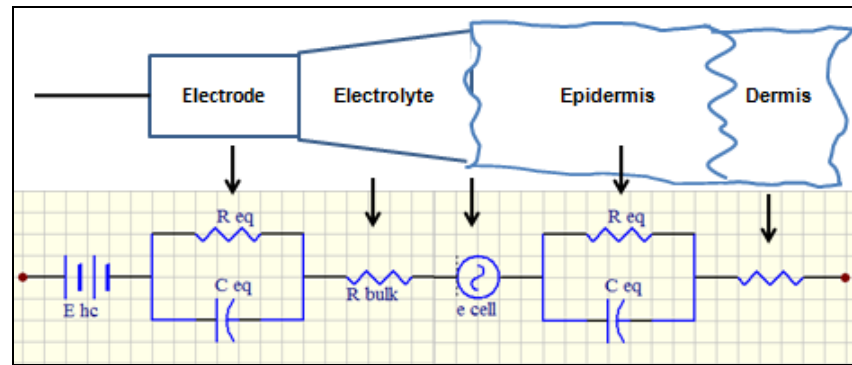
Gambar 2.5 menunjukkan bahwa sinyal EMG yang terdeteksi merupakan penjumlahan dari beberapa potensial aksi *motor unit*. Nilai sinyal EMG ini bergantung pada jarak antar elektroda dan resolusi elektroda. Selain itu, orientasi geometri serat terhadap posisi elektroda juga mempengaruhi besar sinyal EMG [11].

2.2.2. Elektroda Permukaan

Akuisisi sinyal EMG dilakukan dengan menggunakan elektroda permukaan yang terbuat dari bahan Ag/AgCl. Elektroda permukaan dapat ditempelkan di kulit.



Gambar 2.5 Superposisi sinyal EMG



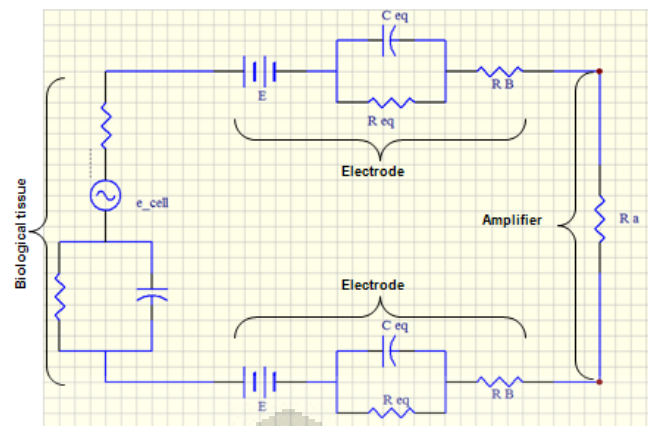
Gambar 2.6 Rangkaian ekivalen interface elektroda - jaringan

Ada beberapa karakteristik yang harus dimiliki oleh elektroda. Elektroda harus bersifat *non-polarizable* sehingga tidak terjadi polarisasi pada saat deteksi sinyal EMG. Elektroda Ag/AgCl bersifat *nontoxic*, tidak menghasikan racun pada saat kontak dengan kulit. Elektroda dirancang agar dapat digunakan kembali atau yang disebut juga dengan *disposable electrode*. Elektroda harus bersifat adesif, memiliki daya ikat yang bagus sehingga tidak menyebabkan *motion artifact*. Selain itu elektroda harus bersifat stabil terhadap temperatur, tahan terhadap oksidasi oleh udara, garam, berbagai jenis larutan dan tahan terhadap air [9]. Elektroda dilapisi dengan cairan elektrolit untuk menghasilkan konduktivitas yang lebih tinggi.

2.2.3. Interface Elektroda-Jaringan

Interface elektroda dengan jaringan dapat dimodelkan seperti pada Gambar 2.5. Baterai, E , adalah *half-cell potential*. Pada daerah transisi elektroda-elektrolit akan terjadi distribusi muatan dari elektroda menuju elektrolit begitu juga sebaliknya. Distribusi muatan ini akan menghasilkan potensial yang disebut dengan *half-cell potential*. Muatan tersebut akan membentuk dua lapisan muatan atau *double layer*. *Double layer* ini dapat dianalogikan sebagai dua plat yang sejajar yang memiliki muatan yang berlawanan dan terpisah oleh bahan dielektrik [7].

Pada Gambar 2.5 resistansi, R_B , menunjukkan resistansi bulk elektrolit. Elemen C_{eq} dan R_{eq} memodelkan karakteristik pada *interface* elektroda dan elektrolit [8]. Kombinasi R_B , C_{eq} dan R_{eq} ini merupakan impedansi elektroda.



Gambar 2.7 Rangkaian ekivalen interface elektroda – jaringan untuk pasangan elektroda

Hubungan impedansi dengan elemen tersebut adalah sebagai berikut:

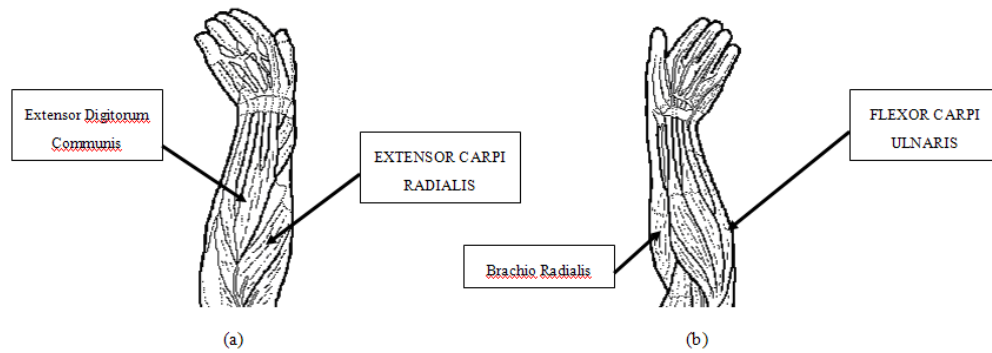
$$Z = R_s + \frac{R_d}{1 + j2\pi f C_d R_d} \quad (6)$$

Persamaan di atas menunjukkan bahwa impedansi elektroda berbanding terbalik dengan frekuensi.

Keluaran elektroda permukaan ini akan menjadi masukan pada penguat diferensial. Untuk menghasilkan beda potensial yang sama dengan beda potensial jaringan, E_{cell} , maka impedansi input amplifier, R_a dirancang setinggi mungkin seperti pada Gambar 2.6 [7].

2.2.4. Motion Artifacts

Motion artifacts merupakan eror yang dihasilkan dalam pengukuran sinyal EMG. Gerakan subjek pada saat pengukuran menghasilkan *artifact* yang dapat menutupi sinyal yang sesungguhnya atau menyebabkan perubahan *baseline*. *Artifact* juga dapat menyebabkan tampilan sinyal tidak dapat dibaca. Pada saat elektroda berada kontak dengan pasta elektrolitik, *half cell potensial* dihasilkan pada *interface* elektroda-pasta. *Interface* elektroda-pasta bisa menjadi sumber *motion artifact*. Ketika pasta terganggu, *half cell potensial* akan bervariasi karena perubahan gradient ion elektroda pada *interface* [9].



Gambar 2.8 (a) Posterior dan (b) Anterior otot lengan bawah.

2.2.5. Anatomi Otot Tangan

Hanya dua elektroda yang digunakan untuk mendeteksi sinyal EMG. Elektroda ini ditempatkan pada kulit diatas otot ekstensor *carpiradialis longus* dan *fleksor carpi ulnaris*. Kedua otot tersebut mengatur gerakan fleksi dan ekstensi tangan [13]. Dua otot ini merupakan otot yang paling tebal pada bagian lengan bawah manusia.

2.3 Mikrokontroler H8/3069F

Mikrokontroler *H8/3069F* adalah salah satu mikrokontroler yang menggunakan processor *H8/300H*. *H8/3069F* mempunyai sebuah *internal wriTabel flash memory* yang menggunakan *single power supply* (5V). Karena *flash memori* adalah ROM maka data dapat disimpan dan tidak akan hilang walaupun *power*-nya telah mati [15].

2.3.1 Analog to Digital Converter

Mikrokontroler *H8/3069F* memiliki 8 input kanal ADC dengan resolusi 10 bit dan waktu konversi maksimum adalah 3.5 μ s per kanal (dengan system *clock* 20 MHz). Konversi ADC *H8/3069F* terdiri dari 2 mode yaitu *single mode* dan *scan mode*. Pada *single mode* konversi dilakukan per kanal ADC. Ada 4 buah register, masing-masing memiliki kapasitas 16 bit data. Data hasil konversi dapat disimpan di register ini.

Metode konversi yang digunakan dalam perangkat keras mikrokontroler ini adalah *Successive Approximation Register ADC* (SAR ADC). Metode ini menggunakan komparator digital. Komparator digital melakukan perbandingan

antara sinyal input analog dengan sinyal digital. Awalnya *Most Significant Bit* (MSB) di-*set* 1. Jika input lebih kecil dari sinyal digital pembanding maka MSB bernilai 0. Kemudian geser bit pembanding ke kanan dan lakukan pembandingan lagi. Namun, jika sinyal input lebih besar maka MSB tetap bernilai 1. Kemudian set nilai bit sebelah kanan menjadi 1. Demikian seterusnya hingga hasil konversi sesuai.

Mikrokontroler *H8/3069F* memiliki tiga register, dua diantaranya yaitu sebagai berikut:

1. A/D Data Registers A to D (ADDRA to ADDR D)

Data hasil konversi A/D akan disimpan di Register A, B, C dan D. Register ini merupakan 16 bit *read-only register* seperti yang ditunjukkan oleh Tabel 2.1. Setiap register dapat menyimpan satu pasang kanal seperti yang dideskripsikan pada Tabel 2.2.

Tabel 2.1 Deskripsi bit pada ADDRA, B, C dan D

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRn	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Read/Write (n = A to D)	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
	A/D conversion data 10-bit data giving an A/D conversion result										Reserved bits					

Tabel 2.2 Deskripsi pasangan input kanal analog dan A/D data register

Analog Input Channel		
Group 0	Group 1	A/D Data Register
AN ₀	AN ₄	ADDRA
AN ₁	AN ₅	ADDRB
AN ₂	AN ₆	ADDRC
AN ₃	AN ₇	ADDRD

Tabel 2.3 Deskripsi bit pada ADCSR

Bit	7	6	5	4	3	2	1	0
	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0
Initial value	0	0	0	0	0	0	0	0
Read/Write	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W	R/W

2. A/D Control/Status Register (ADCSR)

ADCSR adalah register yang dapat ditulis dan dibaca untuk memilih mode konversi dan untuk mengontrol konversi A/D. Register ini terdiri dari 8 bit dengan spesifikasi yang dapat dilihat pada Tabel 2.3.

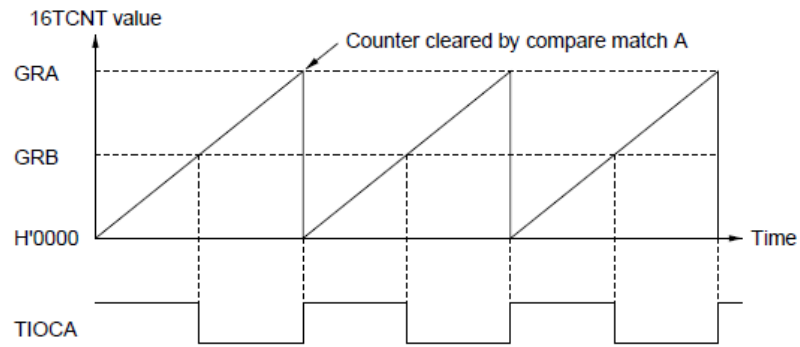
Di bawah ini adalah penjelasan dari fungsi masing-masing bit ADCSR.

- Bit 0-2 *Channel Select* berfungsi untuk memilih kanal yang akan digunakan
- Bit 3 *Clock Select* (CKS) berfungsi untuk mengatur waktu konversi A/D
- Bit 4 *Scan Mode* adalah register untuk pemilihan mode konversi single atau scan
- Bit 5 *A/D Start* adalah register pengatur *start/stop* konversi A/D
- Bit 6 *A/D Interrupt Enable* adalah register untuk aktivasi interup setelah konversi A/D berakhir
- Bit 7 *A/D End Flag* adalah register untuk menyatakan status konversi A/D telah selesai

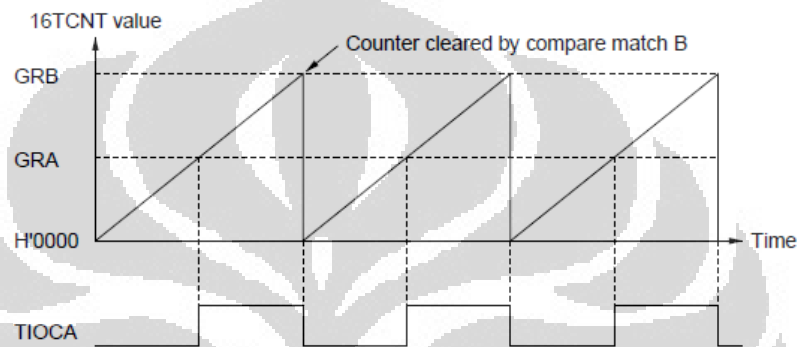
2.4.2 PWM Mode

Untuk menjalankan *standard motor servo*, diperlukan teknik PWM (Pulse Width Modulation). Teknik PWM yang diterapkan di mikrokontroler *H8/3069F* memanfaatkan 16 bit *timer* yang terintegrasi di dalamnya. Beberapa fitur dari *timer* ini adalah :

1. Memiliki enam general register (GRA dan GRB di tiap kanal) yang dapat berfungsi sebagai *output compare* dan *input capture*
2. Terdapat delapan sumber *clock* untuk tiap kanal. Internal *clock* terdiri dari Φ , $\Phi/2$, $\Phi/4$ dan $\Phi/8$. Sedangkan eksternal *clock* terdiri dari TCLKA, TCLKB, TCLKC dan TCLKD
3. Ada lima jenis mode operasi pada tiap kanalnya : *waveform output by compare match*, *input capture function*, *counter clearing function*, *synchronization*, dan *PWM mode*
4. Akses kecepatan tinggi via bus internal 16 bit
5. Memiliki sembilan sumber *interrupt*



a. Counter cleared by GRA (TOA = 1)



b. Counter cleared by GRB (TOA = 0)

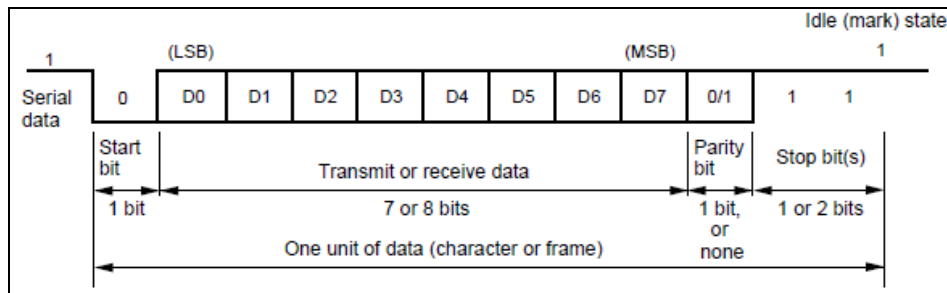
Gambar 2.9. a) Mode PWM dengan counter di-clear-kan oleh GRA

b) Mode PWM dengan counter di-clear-kan oleh GRB

Pada mikrokontroler *H8/3069F*, PWM dapat dihasilkan dari pin TIOCA. Logika 1 pada PWM diatur dengan memasukkan sebuah nilai pada GRA, sementara logika 0 pada PWM dilakukan dengan memasukkan nilai pada GRB. Pengaturan GRA dan GRB dapat dilihat pada Gambar 2.8.

2.4.3 Serial Communication Interface

Transfer data oleh mikrokontroler *H8/3069F* dilakukan melalui komunikasi serial. Metode yang digunakan dapat berupa sinkron atau asinkron. Metode sinkron menunjukkan bahwa *clock* transfer data berasal dari satu sumber *clock*. Sedangkan, pada metode asinkron, transfer data dilakukan berdasarkan *clock* masing-masing pengirim dan penerima data. Kemampuan transfer data metode asinkron lebih cepat daripada metode sinkron. Format data asinkron yang ditransfer terdiri dari *start bit*, 8 bit data, *parity bit*, dan *stop bit* seperti yang ditunjukkan oleh Gambar 2.9.



Gambar 2.10. Format data dalam komunikasi asinkronous

Mikrokontroler *H8/3069F* memiliki tiga kanal komunikasi serial interface. Masing-masing kanal terdiri dari pin serial *clock*, *receive data* dan *transmit data*. Mikrokontroler *H8/3069F* memiliki delapan register komunikasi serial, tiga diantaranya yaitu sebagai berikut:

1. *Serial Mode Register (SMR)*

Tabel 2.4 Deskripsi bit pada SMR

Bit	7	6	5	4	3	2	1	0
	C/A	CHR	PE	O/E	STOP	MP	CKS1	CKS0
Initial value	0	0	0	0	0	0	0	0

Berikut ini adalah fungsi masing-masing bit SMR:

- Bit 0 dan 1 *Clock Source* untuk memilih *baud rate* yang akan digunakan
- Bit 2 *Multiprocessor Mode* untuk memilih fungsi multiprosesor
- Bit 3 *Stop Bit Length* untuk memilih panjang *stop bit*
- Bit 4 *Parity Mode* untuk memilih jenis paritas genap atau ganjil
- Bit 5 *Parity Enable* untuk aktivasi penggunaan paritas
- Bit 6 *Character Length* untuk mengatur panjang karakter pada komunikasi serial asinkron
- Bit 7 *Communication Mode* untuk memilih jenis komunikasi serial yang digunakan sinkron atau asinkron.

2. *Serial Control Register (SCR)*

Tabel 2.5 Deskripsi bit pada SCR

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Initial value	0	0	0	0	0	0	0	0

Berikut ini adalah fungsi masing-masing bit SCR:

- Bit 0 dan 1 *Clock Enable* untuk memilih sumber *clock* antarmuka komunikasi serial
- Bit 2 *Transmit-end Interrupt Enable* untuk aktivasi interup apabila data telah selesai ditransmisikan
- Bit 3 *Multiprocessor Interrupt Enable* untuk aktivasi interup multiprosesor
- Bit 4 *Receive Enable* untuk aktivasi reseiver
- Bit 5 *Transmit Enable* untuk aktivasi transmiter
- Bit 6 *Receive Interrupt Enable* untuk aktivasi interup apabila semua data telah diterima (saat RDRF=1) dan apabila terjadi eror.
- Bit 7 *Transmit Interrupt Enable* untuk aktivasi interup apabila semua data telah terkirim (saat TDRE=1)

3. Serial Status Register (SSR)

Tabel 2.6 Deskripsi bit pada SSR

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER/ERS	PER	TEND	MPB	MPBT
Initial value	1	0	0	0	0	1	0	0

Berikut ini adalah fungsi masing-masing bit SSR:

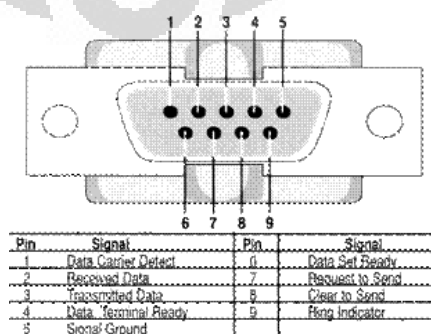
- Bit 0 *Multiprocessor Bit Transfer* untuk mengatur nilai multiprosesor yang akan dikirim
- Bit 1 *Multiprocessor Bit* untuk menyimpan nilai multiprosesor
- Bit 2 *Transmit End* untuk mengindikasikan bahwa data telah selesai ditransmisikan
- Bit 3 *Parity Error* untuk mengindikasikan bahwa paritas yang diterima eror
- Bit 4 *Framing error (FER)/Error signal status (ERS)* untuk mengindikasikan adanya eror pada *framing data* yang ditransfer
- Bit 5 *Overrun Error* untuk mengindikasikan bahwa data yang diterima tidak normal

- Bit 6 *Receive Data Register Full* untuk mengindikasikan bahwa RDR berisi data baru
- Bit 7 *Transmit Data Register Empty* untuk mengindikasikan bahwa data telah dikirim dari TDR menuju TSR dan TDR siap untuk menerima data baru

2.4. RS 232

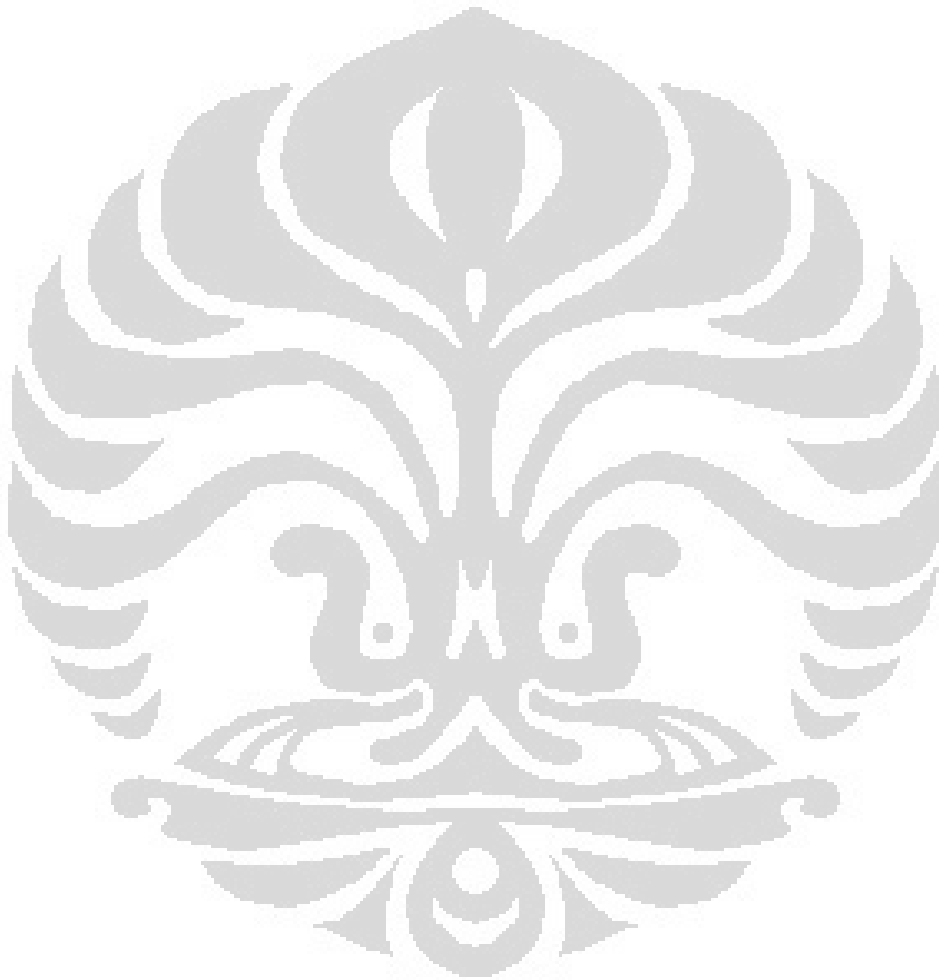
RS 232 adalah sebuah standar antarmuka komunikasi serial asinkron. RS 232 memiliki konektor yang disebut *full-duplex* karena konektor untuk mengirim dan menerima data dirancang terpisah. RS 232 adalah antarmuka standar yang digunakan untuk antarmuka mikrokomputer dengan modem, *keyboard*, printer, dan komputer lain. Data dari mikrokomputer berbentuk paralel sehingga dibutuhkan rangkaian untuk mengonversi data paralel menjadi serial sesuai dengan format RS 232. Rangkaian ini disebut dengan *universal asynchronous receiver transmitter* (UART). Pada *Windows* fitur untuk UART disebut dengan *hyperterminal* [16].

Konektor yang digunakan pada komunikasi serial data terdiri dari dua jenis yaitu serial port yang dapat berupa konektor 25 pin (DB25) dan konektor 9 pin (DB9) yang berpasangan (*male* dan *female*). Pada penelitian ini konektor yang digunakan adalah DB-9 seperti pada Gambar 2.9 . Konektor ini dapat berfungsi sebagai penerima atau pengirim data. Sebuah penerima data harus menerjemahkan sinyal antara +3 dan +25 V sebagai logika 1 dan sinyal antara -3 dan -25 V sebagai logika 0. Sedangkan, sebuah pengirim data harus menerjemahkan sinyal +5 dan +15 V sebagai logika 1 dan sinyal -5 dan -15 sebagai logika 0.



Gambar 2.11 Port DB9 *female*

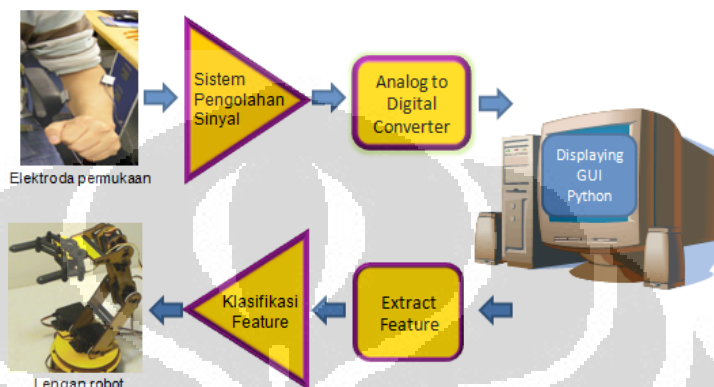
Keluaran TTL standar tidak bisa lebih besar dari 5V. Oleh karena itu, dibutuhkan driver untuk konversi sinyal dari dan menuju TTL (logika 1 > 2V dan logika 0 < 0.8 V). Kecepatan transfer data umumnya diatur pada *baud rate* 9600 (bit/detik) [12].



BAB 3 PERANCANGAN ALAT

3.1 Perangkat Keras

Adapun blok diagram sistem kontrol lengan robot ini adalah sebagai berikut:



Gambar 3.1 Blok diagram sistem kontrol lengan robot

Gambar 3.1 menunjukkan sinyal EMG dideteksi oleh elektroda permukaan. Keluaran elektroda sangat kecil, maka sinyal ini diolah terlebih dahulu dengan rangkaian pengolahan sinyal. Keluaran pengolah sinyal masih dalam bentuk sinyal analog. Agar dapat ditampilkan di PC (Personal Computer), sinyal EMG dikonversi menjadi sinyal digital dengan menggunakan Mikrokontroler *H8/3069F*. Sinyal EMG ditampilkan dengan menggunakan GUI Python. Sinyal EMG yang telah diperoleh dikarakterisasi (*extract feature*) dan diklasifikasikan dengan menggunakan bahasa pemrograman Python sesuai dengan gerakan yang diinginkan. Gerakan yang dikontrol adalah gerakan fleksi-ekstensi pergelangan tangan. Setelah itu, lengan robot dapat dikontrol dengan sinyal EMG oleh mikrokontroler.

3.1.1 Elektroda Permukaan

Elektroda permukaan yang digunakan adalah elektroda merek Ambu. Elektroda ini terbuat dari bahan Ag/AgCl dan telah dilapisi dengan elektrolit *wet gel*. Resolusi sensor ini cukup tinggi yaitu 27 mm^2 , memiliki impedansi AC yang



Gambar 3.2 Elektroda permukaan

kecil yaitu 400Ω , tegangan DC offset 0.2 mV. Gambar 3.1 menunjukkan penampang elektroda permukaan.

Dalam konfigurasi elektroda ada beberapa hal yang harus diperhatikan agar memperoleh data yang kualitatif yaitu sebagai berikut:

1. *Signal-to-noise ratio*

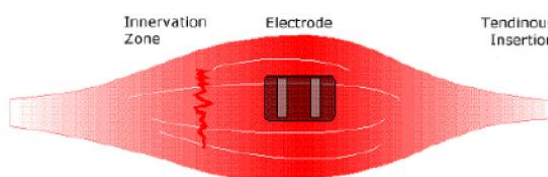
Signal-to-noise ratio (SNR) merupakan fungsi dari interaksi elektrolit dalam kulit dengan elektroda. Noise dapat direduksi dengan menggunakan elektrolit konduktif dan membuang sel kulit mati. Ditemukan bahwa elektroda dengan lebar 1 mm dan panjang 1 cm menghasilkan SNR $>99.5\%$.

2. Bandwith

Bandwith sinyal EMG dipengaruhi oleh jarak antara elektroda dan kecepatan konduksi potensial aksi di sepanjang serat otot. Jarak antar elektroda dapat dianalogikan sebagai *bandpass filter*. Untuk jara antar elektroda 1 cm, pada kecepatan 4 m/dt memiliki *passband* 200 Hz dan *stopband* pada 400 Hz [14].

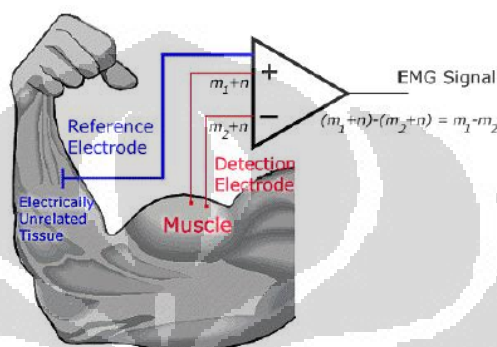
3. Lokasi dan orientasi penempelan elektroda

Gambar 3.2 menunjukkan lokasi dan orientasi penempelan elektroda. Gerakan fleksi dan ekstensi pergelangan tangan dikontrol oleh otot ekstensor *carpiradialis longus* dan *fleksor carpi ulnaris*. penempelan kedua elektroda harus sejajar dengan serat otot.



Gambar 3.3. Lokasi penempatan elektroda yang tepat

Dalam akuisisi sinyal EMG pada satu daerah target membutuhkan tiga elektroda permukaan. Dua elektroda ditempelkan di bagian atas otot yang akan dideteksi. Sedangkan elektroda yang ke-3 ditempelkan pada bagian siku sebagai referensi. Bagian siku merupakan titik yang tepat untuk dijadikan sebagai titik ground karena tidak dilapisi oleh otot. Kabel penghubung elektroda dengan rangkaian pengolah sinyal dirancang sependek mungkin agar sinyal EMG langsung diperkuat. Perancangan seperti ini disebut dengan elektroda aktif [14].



Gambar 3.4. Konfigurasi penguat diferensial. Sinyal EMG ditunjukkan oleh 'm' dan noise ditunjukkan oleh 'n'

Sinyal EMG yang dihasilkan oleh elektroda sangat kecil yaitu beberapa mikrovolt sampai milivolt dengan range frekuensi 20Hz-500Hz. Oleh karena itu, dibutuhkan rangkaian penguat dan rangkaian filter. Rangkaian filter yang digunakan terdiri dari *high pass filter* dan *low pass filter*. Rangkaian *notch filter* tidak dapat diaplikasikan, meskipun banyak noise dari *power supply* pada frekuensi 50-60 Hz. Sinyal EMG dominan pada 50 – 100 Hz, penggunaan *notch filter* akan menyebabkan kehilangan banyak data [14]. Sinyal EMG ini merupakan sinyal AC sehingga dibutuhkan rangkaian *direct adder* untuk mengubah sinyal menjadi sinyal DC. Kemudian sinyal EMG dikonversi menjadi sinyal digital oleh mikrokontroler *H8/3069F*.

3.1.2 Rangkaian Pengolah Sinyal

Adapun pengolahan sinyal yang digunakan dalam penelitian ini yaitu sebagai berikut:

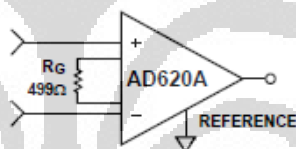
1. Rangkaian Penguat Diferensial
2. Rangkaian *High Pass Filter*

3. Rangkaian *Low Pass Filter*
4. Rangkaian *Direct Adder*

1. Rangkaian Penguat Diferensial

Sinyal dideteksi oleh kedua input penguat diferensial. Jika terdapat perbedaan di antara kedua sinyal tersebut maka perbedaan tersebut akan diperkuat. Noise akan terdeteksi sama di kedua input penguat diferensial sehingga noise dapat dieliminasi. Kemampuan untuk memperkuat perbedaan sinyal dan mengeliminasi sinyal yang sama ini dapat ditentukan dengan melihat *common-mode rejection ratio* (CMRR).

Penguat diferensial ini dirancang dengan menggunakan AD620. AD620 merupakan instrumentasi amplifier yang hanya membutuhkan satu buah resistor untuk mengatur penguatan 1-10.000. Tegangan offset maksimalnya hanya 50 μ V, offset drift 0.6 μ V/ $^{\circ}$ C. AD620 memiliki CMMR yang tinggi yaitu 130 dB.



Gambar 3.5. Rangkaian skematik AD620

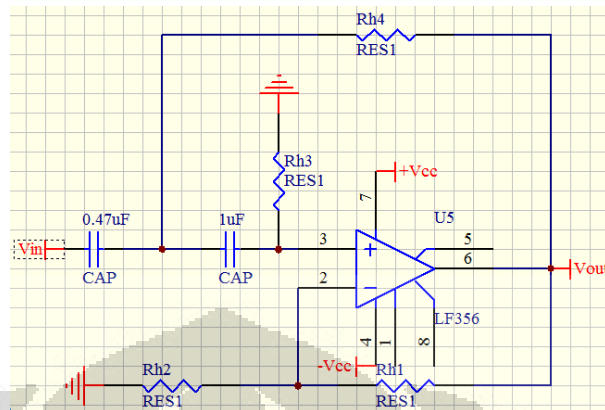
Rancangan penguat diferensial ini berupa penguat dua tingkat. Masing-masing tingkat memiliki penguatan sebesar 100x. penguatan dapat diatur dengan menggunakan resistor 499 Ω . Rangkaian ini membutuhkan *power supply* V_{cc} +/- 5V dan ground sebagai referensi. Persamaan untuk menentukan nilai resistor yaitu sebagai berikut:

$$R_G = \frac{49.4k\Omega}{G-1} \quad (7)$$

2. Rangkaian *High Pass Filter*

Sinyal EMG berada pada frekuensi 20-500Hz. Oleh karena itu dibutuhkan rangkaian *high pass filter* untuk melewatkan sinyal di atas 20Hz dan mengeliminasi sinyal di bawah 20Hz. Respon filter yang digunakan adalah Butterworth karena memiliki respon yang datar. Rangkaian filter ini dirancang

dengan Q faktor 0.707 atau disebut juga *high pass filter* orde 2. Rangkaian ini memiliki roll-off yang cepat pada daerah transisi.



Gambar 3.6. Rangkaian *high pass filter* orde 2

Berikut adalah persamaan untuk menentukan resistor yang digunakan agar diperoleh frekuensi cut-off dan Q faktor yang sesuai untuk respon Butterworth orde 2 :

$$f_c = \frac{1}{2\pi\sqrt{RRCC}} \quad (8)$$

$$R = \frac{1}{2\pi 20\text{Hz}\sqrt{1\mu\text{F}0.47\mu\text{F}}} = 11.614\text{k}\Omega$$

Untuk Q faktor 0.707 digunakan $\alpha=1.414$

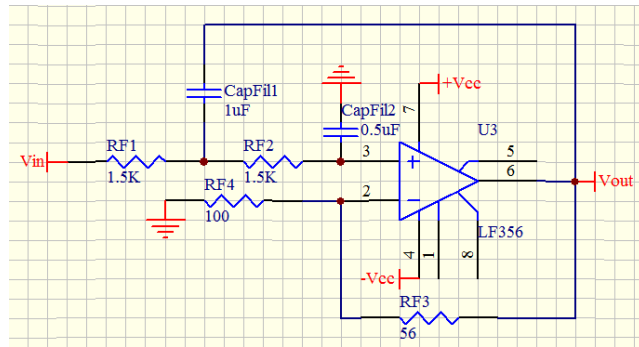
$$R_B = (2 - \alpha)R_A \quad (9)$$

$$R_B = (2 - 1.414)100\Omega = 58.6\Omega$$

$$Ap = \frac{R_B}{R_A} + 1 \quad (10)$$

$$Ap = \frac{58.6\Omega}{100\Omega} + 1 = 0.68$$

3. Rangkaian *Low Pass Filter*

Gambar 3.7. Rangkaian *Low Pass Filter* orde 2

Sinyal EMG berada pada frekuensi 20-500Hz. Oleh karena itu dibutuhkan rangkaian *high pass filter* untuk melewatkan sinyal di bawah 500Hz dan mengeliminasi sinyal di atas 500Hz. Rangkaian *low pass filter* dirancang untuk respon Butterworth orde 2 dengan Q faktor 0.707 sehingga memiliki *roll-off* yang cepat pada daerah transisi. Persamaan untuk menentukan nilai piranti pada rangkaian low pass filter sama dengan persamaan pada *high pass filter*.

$$f_c = \frac{1}{2\pi\sqrt{RRCC}}$$

$$R = \frac{1}{2\pi f_c \sqrt{CC}}$$

$$R = \frac{1}{2\pi 500\text{Hz} \sqrt{1\mu\text{F} \cdot 0.47\mu\text{F}}} = 464.5\Omega$$

Penentuan nilai R_A dan R_B pada rangkaian *low pass filter* sama dengan *high pass filter*.

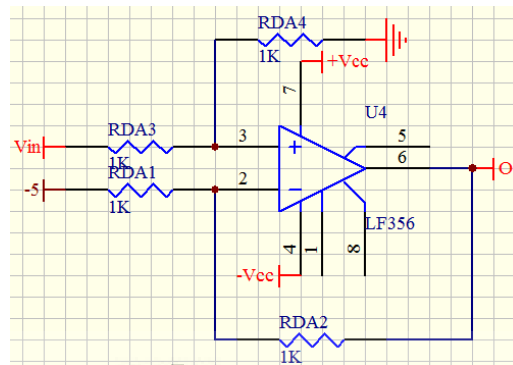
4. Rangkaian *Direct Adder*

Rangkaian *Direct Adder* berfungsi untuk menaikkan level tegangan Rangkaian ini menggunakan LF356 dengan $V_{cc} = \pm 5V$. Semua resistor diberi nilai yang sama yaitu 10 k Ω . Dengan demikian akan diperoleh tegangan output:

$$V_{out} = V_{in} \frac{R_4}{R_3} - V_{in} \frac{R_2}{R_1} \quad (11)$$

$$V_{out} = V_{in} \frac{10k\Omega}{10k\Omega} - (-2.5V) \frac{10k\Omega}{10k\Omega}$$

$$V_{out} = V_{in} + 2.5V \quad (12)$$



Gambar 3.8. Rangkaian *direct adder*



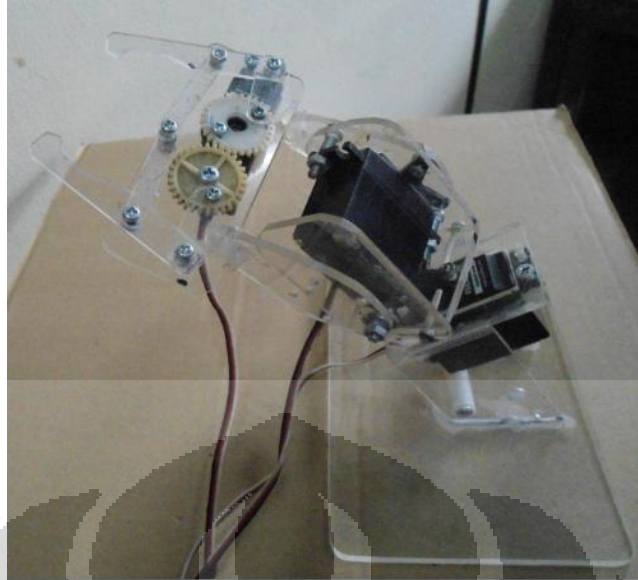
Gambar 3.9. Rangkaian pengolah sinyal yang terdiri dari penguat diferensial, *high pass filter*, *low pass filter* dan *direct adder*

3.1.3. Perancangan Lengan Robot

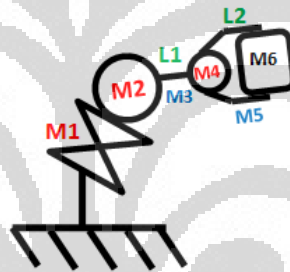
Lengan robot dirancang agar dapat melakukan gerakan fleksi-ekstensi dan menggenggam. Lengan robot ini memiliki 3 DOF seperti pada Gambar 3.8. Torsi maximum dari servo motor produksi Parallax adalah 3.4 kg-cm, massa masing-masing servo adalah 44 g dan massa rangka akrilik $M_3=54.2$ g dan $M_5= 44,2$ g. Panjang $L_1 = 4$ cm dan $L_2 = 8.5$ cm.

Perancangan lengan robot ditujukan untuk dapat sesuai dengan gerakan tangan manusia bukan untuk aplikasi angkat beban. Namun, penentuan beban maksimum yang dapat diangkat tetap dilakukan. Beban maksimum yang mampu dipertahankan oleh motor dapat ditentukan dengan persamaan sebagai berikut:

$$\tau = r \times F \quad (13)$$



Gambar 3.10. Lengan robot

Gambar 3.11 Kondisi lengan robot saat torsi pada M_1 maksimum

Dimana τ adalah torsi, r adalah panjang dan F adalah gaya. Momen gaya pada joint kedua (M_1) pada lengan robot 3 DOF (Gambar 3.11.) dapat ditentukan sebagai:

$$\tau_1 = \frac{1}{2}L_1M_3 + L_1M_4 + 2\left(\frac{1}{2}L_2M_5\right) + \tau_{ujung} \quad (14)$$

$$\tau_{ujung} = (L_1 + L_2)M_6 \quad (15)$$

Karena torsi pada M_1 maximum maka rumus τ maximum yang dapat diangkat (τ_{ujung}) pada kondisi ini adalah :

$$\tau_{ujung} = \tau_1 - \left(\frac{1}{2}L_1M_3 + L_1M_4 + 2\left(\frac{1}{2}L_2M_5\right)\right) \quad (16)$$

Bila bilangan yang didapat dari pengukuran dimasukkan kedalam rumus maka didapat torsi maximum yang dapat diangkat oleh lengan.

$$\tau_{ujung} = 3400g \cdot cm - \frac{1}{2} \times 4cm \times 54.2g - 4cm \times 44g - 2 \left(\frac{1}{2} \times 8.5cm \times 44.2g \right) \quad (17)$$

$$\tau_{ujung} = 2739.9g \cdot cm \quad (18)$$

Berdasarkan persamaan (18) torsi maksimum yang dapat diangkat adalah 2739.9 g.cm. Massa total dari lengan yang akan digerakkan M_1 adalah 142.4 g. Maka titik pusat massa robot dapat dinyatakan sebagai :

$$\begin{aligned} 3400g \cdot cm &= r \times 142.4g \\ r &= \frac{3400g \cdot cm}{142.4g} = 23.87cm \end{aligned} \quad (19)$$

Dengan hasil pada persamaan (18), maka sisa torsi yang masih dapat diangkat adalah 2752.7 g.cm.

3.2 Perangkat Lunak

Perangkat lunak yang digunakan untuk penelitian ini adalah WinAVR dan GNU H8. WinAVR berfungsi untuk proses kompilasi file input (.c) menjadi file asm (.S). Untuk menjalankan fungsi tersebut, dibutuhkan pengaturan perintah *make* dan *clean* pada Menu Tool dalam Programmer Notepad. Pada saat instalasi WinAVR akan terinstal Programmer Notepad. Programmer Notepad berfungsi sebagai tempat menulis file (.c). Perintah *make* akan mengompilasi file (.c) menjadi file (.hex) yaitu file yang menggunakan bahasa yang dimengerti oleh mikrokontroler. GNU H8 merupakan perangkat lunak yang berisi *library GCC*, *assembler* dan *linker* yang akan mengubah file asm (.s) menjadi file objek (.o) dan menghubungkan file-file menjadi file (.elf) dan kemudian mengubahnya menjadi file (.hex). Agar WinAVR dan GNU H8 dapat bekerja sama maka kedua perangkat lunak ini harus diletakkan pada *directory* yang sama.

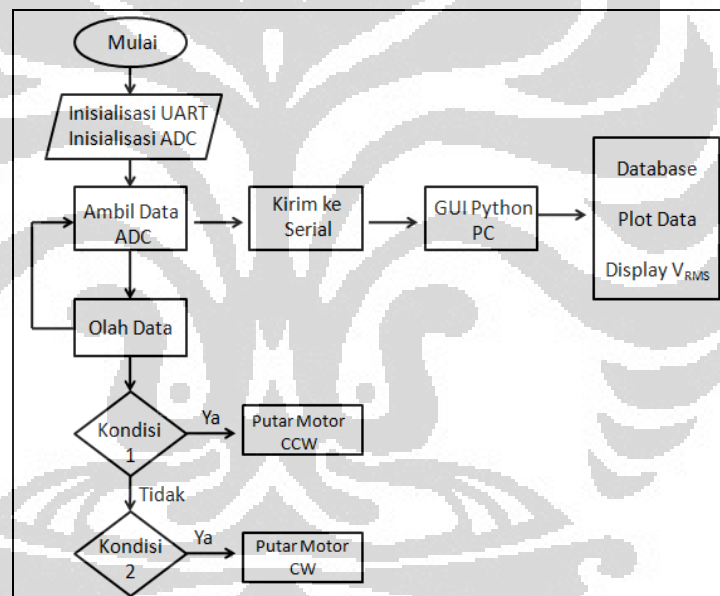
3.2.1 Pemrograman pada mikrokontroler H8/3069F

Akuisisi sinyal EMG dan kontrol lengan robot dilakukan oleh mikrokontroler H8/3069F dengan menggunakan bahasa pemrograman C. Alur pemrograman bahasa C pada mikrokontroler H8/3069F dapat dilihat pada Gambar 3.12. Program dimulai dengan inisialisasi UART dan ADC. Kemudian dilanjutkan dengan pengambilan data. Data tersebut diolah untuk menentukan nilai *root mean*

square (RMS). Persamaan untuk menentukan RMS ditunjukkan oleh persamaan 20. Nilai RMS hanya akan menunjukkan 2 kondisi berbeda. Jika nilai RMS menunjukkan kondisi 1 maka putar motor *clock wise*, sedangkan jika menunjukkan kondisi 2 maka putar motor *counter clock wise*. Data dikirim ke komputer dengan menggunakan kabel serial (RS 232). Data ini disimpan di dalam Microsoft Access dan ditampilkan di GUI Python dalam bentuk grafik dan angka.

$$RMS = \sqrt{\frac{\sum_{i=1}^n A_i^2}{n}} \quad (20)$$

A_i adalah amplitudo yang dicuplik dari segmen waktu ke i sampai dengan total data n .



Gambar 3.12 Diagram alir akuisisi sinyal EMG dan kontrol motor

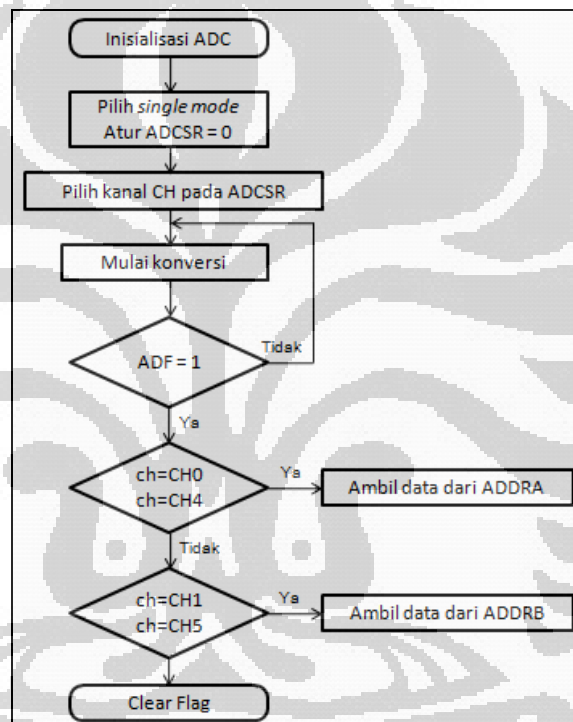
Diagram alir pada Gambar 3.12 mendeskripsikan struktur pemrograman pada *main file*. Pada *main file* ini, ada file yang harus disertakan yaitu *library file* atau *header file*. *Library file* berisi definisi-definisi yang digunakan dalam program. Salah satunya yaitu *ioh83069.h*, file ini berisi definisi register-register fungsi khusus dan pin-pin mikrokontroler *H8/3069F*. Pada *main file* ini hanya berisi perintah untuk membaca data ADC dan kontrol motor, sedangkan program ADC (*Analog to Digital Conversion*) dan komunikasi serial atau UART berada pada file (.c) lain. Namun, file ini dicantumkan sebagai *source file* pada *makefile*.

Makefile juga berisi file-file lain yang dibutuhkan untuk kompilasi. File hasil kompilasi memiliki ekstensi (.mot). Format file ini siap di tulis ke mikrokontroler *H8/3069F* . Penulisan program dilakukan di *command prompt*. Berhasil atau tidaknya penulisan program akan terlihat pada *command prompt*.

Di bawah ini, pemrograman konversi A/D, komunikasi serial asinkron (UART) dan kontrol motor dengan mode PWM akan dijelaskan lebih terperinci.

1. Pemrograman konversi A/D

Berikut adalah diagram alir pemrograman konversi A/D pada mikrokontroler *H8/3069F*:



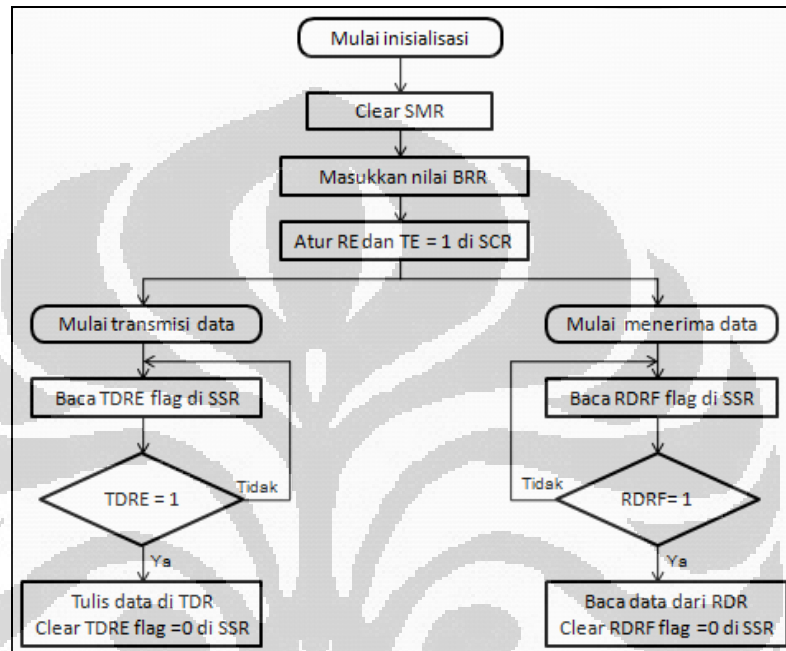
Gambar 3.13. Diagram alir konversi A/D

Akuisisi sinyal EMG menggunakan port ADC pada mikrokontroler *H8/3069F* dengan *single mode* karena sinyal EMG yang diakuisisi hanya satu kanal. Pemrograman konversi A/D dimulai dengan inisialisasi ADC, memilih *single mode* dengan mengatur $ADCSR = 0$ dan memilih kanal pada ADCSR. Kemudian konversi dapat dimulai dengan melihat status *flag* $ADF = 1$, jika tidak maka ulangi langkah sebelumnya sampai $ADF=1$. Langkah selanjutnya lihat apakah kanal yang digunakan CH0 atau CH4, jika iya maka ambil data dari ADDRA. Jika tidak, lihat apakah kanal yang digunakan CH1 atau CH5, jika iya

ambil data dari ADDR. Langkah terakhir, atur $flag = 0$ untuk mengindikasikan konversi A/C telah selesai.

2. Pemrograman Komunikasi Serial Asinkron

Di bawah ini adalah diagram alir komunikasi serial asinkron pada mikrokontroler *H8/3069F*.



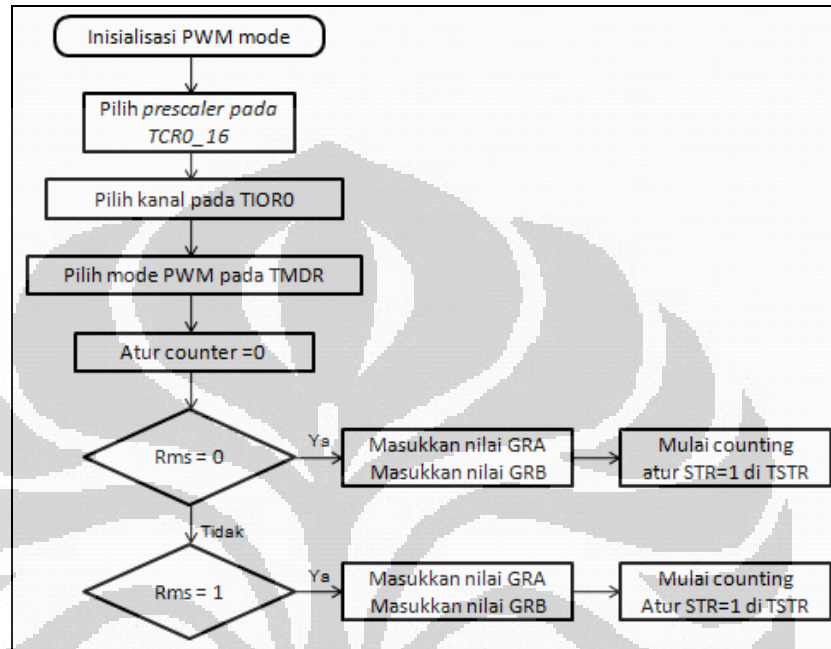
Gambar 3.14. Diagram alir komunikasi serial asinkron

Tahap inisialisasi UART yaitu sebagai berikut:

1. Mengatur format komunikasi serial yang akan digunakan asinkron atau sinkron dengan cara meng-*clear* SCR
2. Mengatur kecepatan transfer data dengan mengatur baud rate pada BRR
3. Mengaktifkan register yang mengatur pengiriman dan penerimaan data dengan cara *set* TE dan RE = 1 pada SCR

Pengiriman data serial dilakukan dengan beberapa tahap yaitu pertama, periksa apakah data telah dikirimkan ke TSR dan TDR telah kosong dengan memastikan nilai TDRE =1. Kedua, jika TDRE =1 maka tulis data ke TDR. Ketiga, atur TDRE=0 untuk mengindikasikan bahwa TDR belum siap untuk menerima data baru.

Ada beberapa tahap dalam proses penerimaan data serial yaitu pertama, periksa apakah RDR berisi data yang baru diterima dengan melihat status RDRF=1. Kedua, baca data dari RDR jika RDRF = 1. Ketiga, atur RDRF = 0 untuk menyatakan RDR telah kosong.



Gambar 3.15 Diagram alir kontrol motor dengan menggunakan PWM mode

3. Pemrograman *PWM Mode* untuk Kontrol Motor

Gambar 3.15 menunjukkan diagram alir pemrograman kontrol motor dengan *PWM mode*. Pemrograman kontrol motor dengan menggunakan *PWM mode* dimulai dengan inisialisasi yang terdiri dari

- Pilih *prescaler* yang dilakukan dengan cara *set* bit TPSC0 pada register 16TCR
- Pilih kanal (*channel 0*) dengan mengatur bit pada TIOR0
- Pilih *PWM mode* dengan *set* TMDR
- Atur *counter* bernilai nol dengan *set* TSTR untuk menunjukkan bahwa perhitungan belum dimulai

Setelah proses inisialisasi selesai, pemrograman akan membaca apakah RMS bernilai 0 atau 1. Jika RMS bernilai 1 maka masukkan pada GRB waktu yang

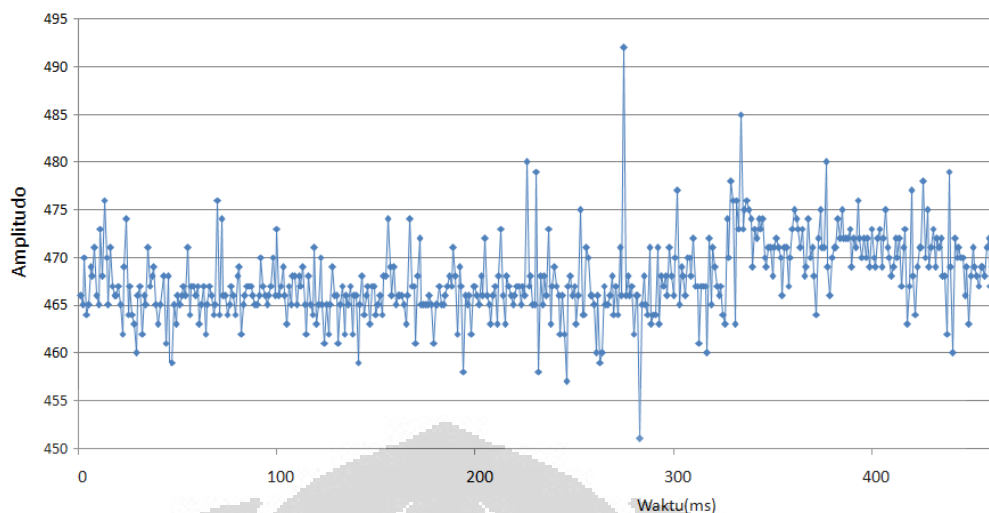
dibutuhkan oleh motor untuk mencapai posisi 90° yaitu 4375. Jika RMS bernilai 0 maka masukkan pada GRB waktu yang dibutuhkan oleh motor untuk mencapai posisi 0° yaitu 1875. Sedangkan pada GRA, masukkan nilai *prescaler* yang digunakan yaitu 62500. Kemudian, proses *counting* dapat dimulai dengan *set STR* di TSTR. Dengan demikian motor akan berputar menuju posisi 90° dan 0° .

3.2.2. Tampilan GUI Python

Sinyal EMG yang diperoleh di tampilkan di PC dalam bentuk GUI (*Graphical User Interface*). GUI dirancang dengan menggunakan bahasa pemrograman Python. Pemrograman bahasa Python untuk GUI meliputi pengiriman data dari serial, tampilan grafik, *database* dan nilai sinyal EMG. Tampilan grafik ini menggunakan kanvas Matplotlib sedangkan system *database* menggunakan SQL dan data disimpan di Microsoft Acces.

Sinyal EMG yang telah diakuisisi oleh mikrokontroler dikirim ke komputer dengan menggunakan komunikasi serial dan dikirim ke GUI Python untuk diolah. Pengolahan sinyal EMG cukup sulit karena merupakan proses stokastik, bersifat random dan tidak mempunyai pola [16]. Amplitudo sinyal EMG bervariasi dari 0-10 mV (puncak ke puncak) atau 0-1.5 mV (rms) [14]. Oleh karena itu, dibutuhkan metodologi yang sistematis untuk memperoleh karakteristik sinyal EMG agar dapat diklasifikasikan.

Sebelum data ditampilkan dengan menggunakan GUI python, data diaukuisisi dengan menggunakan hyperterminal. Hyperterminal hanya dapat menampilkan data dan tidak dapat menyimpan data. Oleh karena itu, data di hyperterminal ini di kemudian di pindahkan ke dalam file excel untuk diplot seperti pada Gambar 3.16. Hal ini bertujuan untuk *plotting* data cuplik pada frekuensi sample 1 kHz, sedangkan GUI python hanya mampu menampilkan data dengan frekuensi sample 10 Hz secara *real time*.



Gambar 3.16 Tampilan sinyal EMG untuk frekuensi sample 1kHz

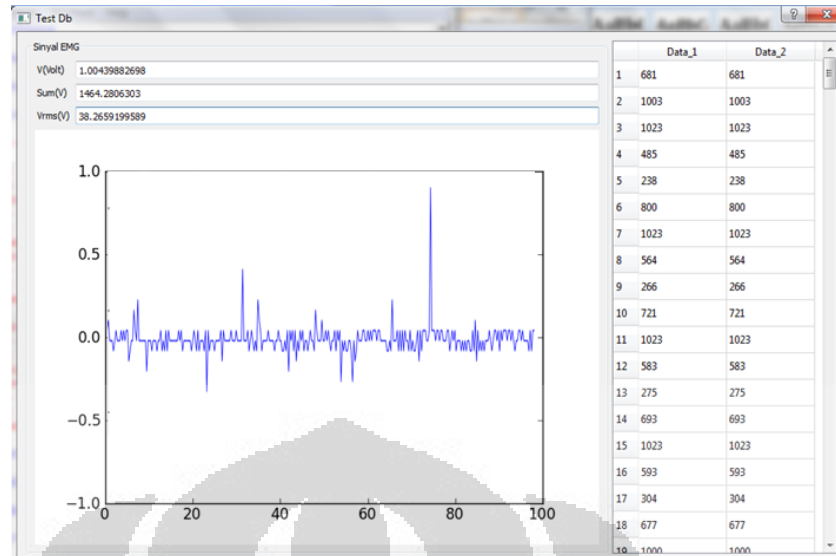
Berdasarkan Gambar 3.16 sinyal EMG ditampilkan dalam bentuk data biner dan belum dikonversi ke dalam volt. Sinyal EMG ini memiliki *baseline* pada 466 dan kemudian berubah naik menjadi 477. Perubahan *baseline* ini dapat disebabkan oleh *motion artifact*. Noise ini dapat dihilangkan dengan memastikan *grounding electrode* yang baik.

Hasil yang ditampilkan pada GUI merupakan hasil pengolahan dari data-data yang dikirimkan dari sensor. Hasil keluaran dari sensor diolah dengan menggunakan pemrograman python untuk dikalibrasi. Kalibrasi dilakukan dengan menetapkan *baseline* pada titik 0. Sinyal EMG bervariasi antara 0-1.5 mV maka *display* data akan berada di sekitar *baseline*. Kalibrasi *baseline* ini sangat penting karena *baseline* dapat berubah akibat *motion artifact*.

Kalibrasi *baseline* dilakukan dengan cara mengkonversi nilai biner menjadi nilai dalam satuan volt. Berikut adalah persamaan untuk kalibrasi *baseline* sinyal EMG:

$$\text{volt} = ((\text{float}(y)) * 5 / 1023) - (2.063)$$

volt adalah amplitudo (V), y adalah amplitudo dalam bentuk biner, 5 adalah nilai tegangan referensi mikrokontroler, 1023 adalah resolusi ADC (10bit), 2.063 adalah faktor koreksi yang diberikan agar *baseline* tepat pada garis nol. Grafik hasil kalibrasi dapat dilihat pada Gambar 3.17.



Gambar 3.17 Tampilan GUI sinyal EMG

Sinyal dapat langsung diamati dengan menggunakan grafik. Jika terdapat eror seperti perubahan *baseline* atau terdapat data yang berada diluar nilai *threshold* ($data > 1.7 \text{ V}$ dan $data < -1.7 \text{ V}$) maka pengambilan data dapat dihentikan. Pengambilan data dapat dilakukan kembali setelah memperbaiki konfigurasi elektroda sehingga error akibat *motion artifact* dapat dieliminasi.

BAB 4

ANALISIS HASIL PENELITIAN

4.1. Hasil Penelitian

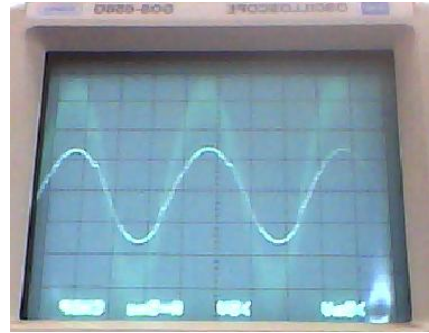
Sebelum pengambilan data terlebih dahulu dilakukan uji rangkaian pengolah sinyal. Rangkaian pengolah sinyal yang diuji yaitu rangkaian penguat diferensial, rangkaian *high pass filter*, rangkaian *low pass filter* dan rangkaian *direct adder*.

4.1.1 Hasil Uji Rangkaian Penguat Diferensial

Rangkaian penguat diferensial diberikan input AC mulai dari 2mVpp – 50mVpp. Rangkaian ini diuji dengan menggunakan osiloskop dengan kanal 1 (5mV/div) untuk *display* input dan kanal 2 (5mV/div) untuk *display* output seperti pada Gambar 4.1. Pada saat *display* kedua kanal maka akan terlihat kedua input sefase atau berhimpit karena sinyal AC input dimasukkan ke input noninverting sedangkan input inverting diberi masukan 0 V. Berikut adalah data hasil uji rangkaian penguat diferensial.

Tabel 4.1. Data hasil uji penguat diferensial tingkat 1 dan 2

Vin(mVpp)	Vout(Vpp)	Vin(mVpp)	Vout(Vpp)
2	0.2	2	0.2
3	0.3	3	0.32
4	0.38	4	0.4
5	0.48	5	0.48
6	0.6	6	0.6
7	0.7	7	0.7
8	0.8	8	0.8
9	0.9	9	0.9
10	1	10	1
20	2	20	2
30	3	30	3
40	4	40	4
50	5	50	5

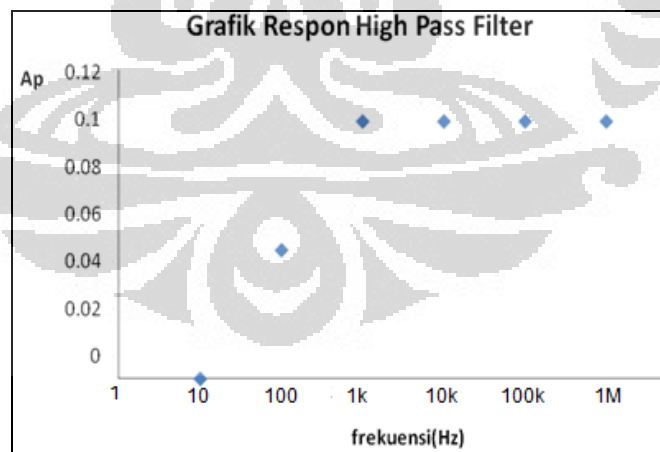


Gambar 4.1 *Display* keluaran penguat diferensial

Berdasarkan Gambar 4.1 dapat dilihat bahwa sinyal input diperkuat 100x.

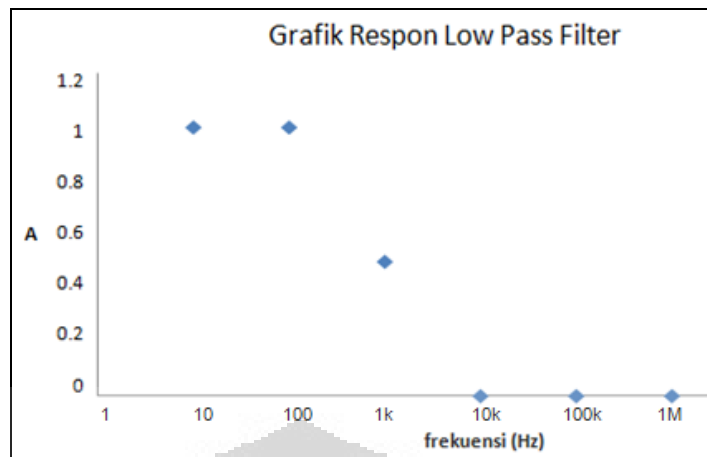
4.1.2 Hasil Uji Rangkaian Filter

Rangkaian filter diuji dengan menggunakan *function generator* dan osiloskop. Berdasarkan hasil uji rangkaian filter diperoleh grafik data seperti pada Gambar 4.2. Gambar 4.2 menunjukkan rangkaian *high pass filter* memiliki penguatan 0.05 pada saat frekuensi cut off 100Hz. Nilai ini mendekati nilai penguatan hasil perhitungan sebelumnya yaitu 0.68. Grafik respon high pass filter ini terbukti dapat melewati frekuensi di atas 100 Hz. Rangkaian ini dapat mengeliminasi interferensi *power supply* yang berfrekuensi 50 Hz.



Gambar 4.2. Grafik respon *high pass filter*

Gambar 4.3 menunjukkan rangkaian *low pass filter* orde 2 dengan respon Butterworth. Sesuai dengan rancangan, rangkaian ini akan memiliki output



Gambar 4.3. Grafik respon *low pass filter*

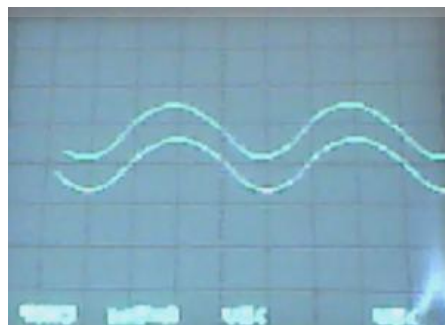
setengah dari input pada saat frekuensi *cut-off* dan memiliki respon yang datar pada *passband*.

4.1.3 Hasil Uji Rangkaian Direct Adder

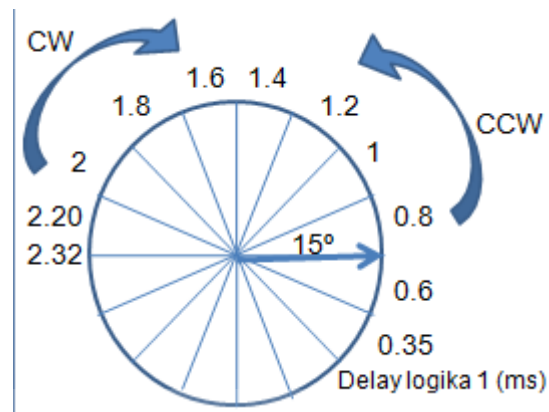
Rangkaian direct adder berfungsi untuk menaikkan level tegangan maka respon rangkaiannya akan berupa grafik yang dideskripsikan oleh Gambar 4.4. Pada Gambar 4.4 sinyal input akan diangkat sebesar 5 V. Akan tetapi untuk rangkaian *direct adder*, sinyal input akan dinaikkan sebesar 2.5V karena mikrokontroler tidak dapat menerima masukan lebih besar dari pada 5 V

4.1.4 Hasil Uji Kontrol Servo Motor

Pengujian servo motor dilakukan dengan menggunakan mikrokontroler *H8/3069F* dengan memberikan *delay* 1.3 ms untuk logika 0 dan *delay* 2.32 ms untuk logika 1. Standar servo motor mempunyai gerakan poros mulai dari 0° sampai 180°, tetapi pada kenyataannya gerakan poros ini mencapai 200°. Hal ini



Gambar 4.4. Display input dan output *direct adder*



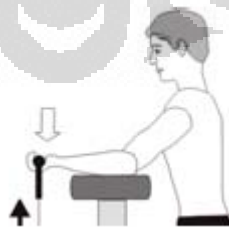
Gambar 4.5. Posisi servo motor untuk setiap *delay* yang diberikan

dimaksudkan untuk menjaga agar servo motor tidak cepat rusak karena pemberian sinyal yang melewati batas [6].

Pengaturan posisi servo motor dilakukan dengan memberikan *delay* logika 1 yang berbeda, sedangkan logika 0 diberi *delay* yang sama yaitu 1.3 ms. Variasi *delay* logika 1 untuk setiap posisi servo motor dapat dilihat pada Gambar 4.5. Berdasarkan Gambar 4.5 dapat dilihat bahwa servo motor membutuhkan *delay* yang sama untuk putaran *clock wise* dan *counter clock wise*.

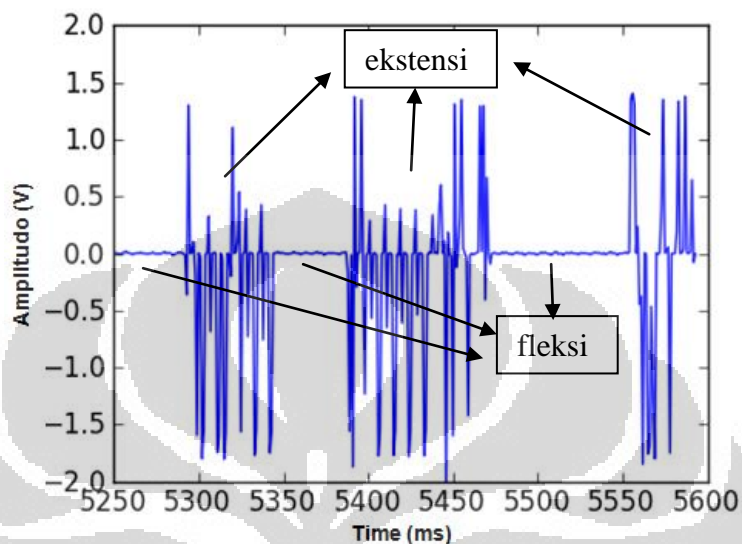
4.1.5 Sinyal EMG

Pengambilan data sinyal EMG dilakukan dengan cara meminta subjek untuk melakukan gerakan fleksi dan ekstensi pergelangan tangan secara bergantian. Tangan subjek harus di letakkan di atas meja agar tidak menghasilkan gerakan selain gerakan fleksi – ekstensi pergelangan tangan seperti pada Gambar 4.6 . Untuk gerakan fleksi tangan harus berada pada posisi yang relaks.



Gambar 4.6 Posisi tangan pada saat pengambilan data

Pada saat sinyal EMG yang terdeteksi sangat kecil. Sedangkan pada saat gerakan ekstensi beberapa *motor unit* akan diaktivasi. Data sinyal EMG dapat dilihat pada Gambar 4.7



Gambar 4.7. Data sinyal EMG

Pada Gambar 4.7 sinyal EMG untuk gerakan fleksi dan ekstensi dapat dibedakan dengan jelas. Gerakan fleksi menghasilkan sinyal yang sangat kecil sehingga terlihat berupa garis lurus sedangkan gerakan ekstensi menghasilkan sinyal dengan amplitudo cukup tinggi.

Karakterisasi sinyal EMG dilakukan dengan menentukan nilai RMS dari data yang dicuplik. Nilai RMS sinyal EMG dapat dilihat pada Tabel 4.2.

Tabel 4.2. Nilai RMS sinyal EMG untuk gerakan fleksi dan ekstensi

Jenis Gerakan	RMS (V)				
	Percobaan ke 1	Percobaan ke 2	Percobaan ke 3	Percobaan ke 4	Percobaan ke 5
Fleksi	0.02	0.13	0.01	0.02	0.01
Ekstensi	0.91	1.04	1.19	0.88	0.69

Berdasarkan Tabel 4.2 dapat dilihat bahwa nilai RMS sinyal EMG untuk gerakan fleksi dan ekstensi memiliki perbedaan yang signifikan. RMS untuk gerakan fleksi berkisar antara 0.01 – 0.13 V dan RMS untuk gerakan ekstensi

berkisar antara 0.69 - 1.19 V. Berdasarkan nilai RMS ini, klasifikasi sinyal EMG dapat dilakukan.

4.2. Analisis

Gambar 4.7 menunjukkan terdapat perbedaan antara gerakan fleksi dan gerakan ekstensi. Pada saat gerakan fleksi, membran sel memiliki potensial istirahat sebesar -60 mV sampai -100 mV. Setiap *motor unit* akan memiliki beda potensial dalam range potensial istirahat ini. Pada saat deteksi sinyal EMG dengan elektroda permukaan, dimana elektroda mendeteksi beberapa *motor unit*, sinyal ini akan dieliminasi oleh penguat diferensial. Oleh karena itu, pada saat gerakan fleksi sinyal berkisar antara 0-0.02V.

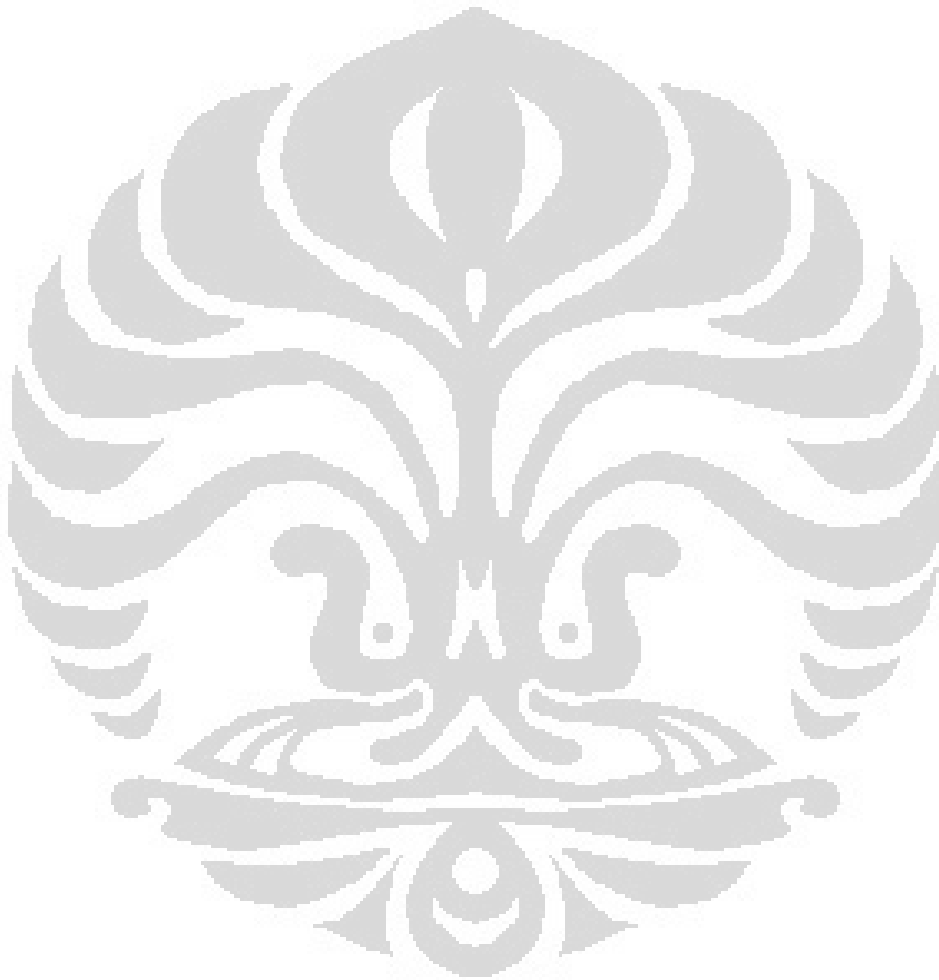
Pada saat gerakan ekstensi, beberapa *motor unit* diaktifkan sehingga terdapat beberapa *motor unit* dengan beda potensial yang berbeda-beda. Perbedaan potensial ini akan diperkuat oleh penguat diferensial. Sinyal EMG untuk gerakan ekstensi ini berkisar antara -1.7 V sampai dengan 1.7 V.

Kemungkinan adanya noise sangat kecil karena penguat diferensial yang digunakan memiliki CMRR 130 dB. Nilai CMRR ini telah memenuhi standar yang telah di sepakati yaitu >90dB [11].

Karakterisasi (*extract feature*) sinyal EMG dapat dilakukan dengan membandingkan nilai amplitudo dan *root mean square (RMS)*. RMS gerakan fleksi berkisar antara 0.01 – 0.13 V sedangkan gerakan ekstensi antara 0.69 - 1.19V.

Pengklasifikasian sinyal EMG berdasarkan gerakan bertujuan untuk menggerakkan lengan robot sesuai dengan gerakan yang diuji. Sebelumnya kontrol lengan robot dengan menggunakan PWM telah berhasil dilakukan dengan memberikan *delay* yang sesuai sehingga diperoleh putaran motor yang sesuai seperti pada Gambar 4.5. Sinyal EMG yang telah diklasifikasikan berdasarkan nilai RMS dikodekan untuk menggerakkan motor. Gerakan fleksi menunjukkan posisi 0° pada motor dan gerakan ekstensi menunjukkan posisi 90° pada motor. Dengan demikian, untuk gerakan fleksi dengan RMS antara 0.01 – 0.13 V diberi *delay* 0.6 ms sedangkan untuk gerakan ekstensi dengan RMS antara 0.69 - 1.19 V diberi *delay* 1.4 ms untuk menggerakkan motor.

Pada pengklasifikasian sinyal EMG muncul error yang mengakibatkan berubahnya posisi garis ground. Nilai RMS untuk gerakan fleksi akan berada pada range nilai RMS gerakan ekstensi. Akibatnya lengan robot akan melakukan gerakan yang tidak sesuai dengan gerakan tangan. Hal ini dapat dicegah dengan menjaga posisi elektroda yang ditempelkan pada siku tetap konstan.



BAB 5

PENUTUP

5.1 Kesimpulan

Bedasarkan hasil yang didapat dari kontrol lengan robot menggunakan sinyal EMG dapat diambil kesimpulan, yaitu:

1. Akuisisi sinyal EMG dapat ditampilkan dalam bentuk GUI python. Sinyal EMG yang ditampilkan berupa sinyal acak karena merupakan interferensi beberapa potensial aksi *motor unit*.
2. Lengan robot dapat bergerak sesuai dengan gerakan fleksi-ekstensi pergelangan tangan.

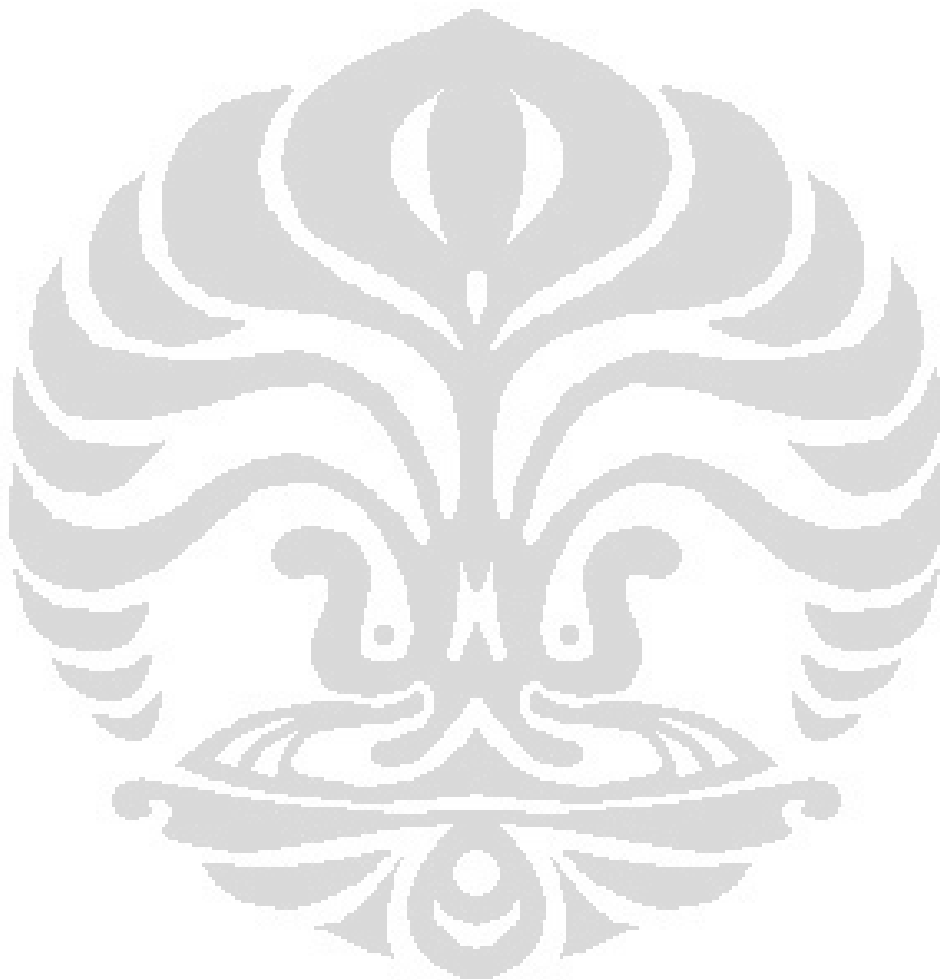
5.2 Saran

Penelitian lebih lanjut untuk menambah hasil-hasil yang telah dilakukan pada penelitian ini dapat dilakukan dengan penyempurnaan sistem kontrol lengan robot. Penyempurnaan dapat dilakukan dengan penambahan gerakan lengan yang dapat dikontrol seperti *gripper*, gerakan pronasi-supinasi, dan gerakan fleksi-ekstensi tangan. Untuk itu, sistem ini membutuhkan sensor elektroda permukaan, sistem pengolahan sinyal, dan jumlah DOF lengan robot yang lebih banyak.

DAFTAR PUSTAKA

- [1] Yani, Irsyadi & Thamrin, Ismail. “ Perancangan Prototipe Lengan Robot Lima Derajat Kebebasan (5-DOF)”. Teknik Mesin Universitas Sriwijaya. Inderalaya
- [2] Andreas W.S dkk. Robot Lengan 3 DOF dengan Input Sinyal EMG. Teknik Elektronika Politeknik Negeri Surabaya. Surabaya
- [3] Balza Achmad, Musthofa Sunaryo, Agus Arif. Simulator Lengan Robot Enam Derajat Kebebasan Menggunakan Opengl, Jurusan Teknik, Fakultas Teknik Universitas Gadjah Mada.
- [4] Diktat Kuliah Mekantronika Teknik Mesin Universitas Widyagama Malang Bab III Aktuator Elektrik
- [5] Hari S Basuki. Puslitbang INKOM-LIPI Bulletin IPT, No.1 Vol.II, April/Mei 1996
- [6] Britantyo Wicaksono. 2007. Sistem Kontrol Lengan Robot Menggunakan Mikrokontroler 16-BIT *H8/3069F* .Departemen Fisika. Universitas Indonesia
- [7] Richard Aston. Principle Biomedical Instrumentation
- [8] A. Terry Bahill.1981. Bioengineering, Biomedical, Medical And Clinical Engineering. Prentice Hall. Englewood Cliffs
- [9] R.S. Khandpur. 2005. Biomedical Instrumentation Technology And Application. Mcgraw-Hill. United Stated Of Amerika
- [10] Leslie Cromwell, Fred J. Weibell, Erich A Pfeiffer. 1980. Biomedical Instrumentation And Measurement. Prentice Hall
- [11] Peter Conrad. Abc of EMG A Practical Introduction To Kinesiological Electromyography. Noraxon Inc. United Stated of America
- [12] Stephen Derenzo. 2003. Practical Interfacing In The Laboratory. Cambridge University
- [13] Beng Guey Lau. 2009. An Intelligent Prosthetic Hand Using Hybrid Actuation And Myoelectric Control. The University Of Leeds School Of Mechanical Engineering

- [14] Carlo J. De Luca. 2002. Surface Electromyography : Detection and Recording. Delsys Incorporated
- [15] Renesas Technology. 2005. H8/3068F-ZTAT Hardware Manual.
- [16] Edgard Lamounier, Alcimar soares, Adriano Andrade, Renato Carrijo. 2008. A Virtual Prosthesis Control Based On Neural Networks For EMG Pattern Classification. IEEE Transaction and Biomedical Engineering, USA.



LAMPIRAN A

Tabel Data Amplitudo Sinyal EMG untuk Gerakan Fleksi dan Ekstensi

fleksi (V)	Ekstensi (V)	Fleksi (V)	Ekstensi (V)	Fleksi (V)	Ekstensi (V)	Fleksi (V)	Ekstensi (V)	Fleksi (V)	Ekstensi (V)
0.01	-1.74	0	0.02	0	-1.7	0	-1.64	0	-0.66
-0.09	-1.56	0	1.28	-0.01	-0.01	-0.01	-0.02	0.01	1.05
0	0	0	-1.76	0	1.26	0	0.94	0	-0.12
0	0.01	0	-1.77	0	1.32	0	0.31	0.01	-0.02
0	-1.78	0	0.01	0.01	-1.89	0.01	-1.81	0	-0.03
-0.01	1.35	0	1.28	0	0	0	0.51	-0.01	-0.04
0	0	-0.01	-1.69	-0.01	-0.02	0	-0.04	0	0.88
-0.01	0.01	0.63	-1.74	0	-1.8	0	-0.01	-0.02	-1.69
0.01	0.14	0	0.01	-0.01	-1.68	-0.01	-1.61	0	-0.09
0	0.78	0	1.28	0.01	-0.01	0	-0.13	0.01	-0.01
0.01	-1.55	0	0.03	0	-0.31	0	1.04	0.01	-1.82
0	-0.01	0	-0.62	0.02	-1.8	0.01	-0.39	-0.01	0.55
-0.01	0	0	0	0	-1.62	0.08	-0.02	0	0.93
0	0.13	-0.01	-0.11	0	0	0	-0.01	0	-0.23
-0.01	-0.48	0	0.43	0	-0.56	0	-0.03	-0.02	-0.02
0	0	0	-0.3	-0.02	-1.46	-0.01	-0.03	0	0.11
0	0	0.01	-1.75	-0.01	1.19	0	-0.03	0	0.76
0.01	-0.03	-0.01	0.06	0	-0.01	-0.01	-0.03	0	-0.67
0	-1.53	0.01	-0.9	0	-0.02	0	-0.04	0	-0.02
0	1.35	0	1.28	0	-1.54	0	-1.69	0.01	-0.03
0	0	-0.02	-1.34	0	0.33	0.01	-0.63	0	-0.03
-0.02	0.01	0	0.01	0	-1.74	0	-0.02	0	-0.02
-0.01	0.14	0	1.3	-0.01	0.01	0.01	-1.84	0	-0.02
0	0.62	0	-0.05	0	1.26	-0.01	-1.35	-0.01	-0.03
0	-1.57	0	0.66	0	-1.6	0	-0.15	0	1

LAMPIRAN B

Pemrograman C Untuk Akuisisi Data dan Kontrol Lengan Robot pada Mikrokontroler

```
/******
```

```
#include "interrupt.h"
#include "ioh83069.h"
#include "uart.h"
#include "delay.h"
#include <stdio.h>
#include <stdlib.h>
#include "adc.h"
#include <math.h>
```

```
#define PIN_H() (P1DR |= (1 << PIN0))
#define PIN_L() (P1DR &= ~(1 << PIN0))
```

```
unsigned short n=300;
unsigned int a[500],v[500],sum;
unsigned short i;
char buf[500], data, cmdbuf[20];
unsigned int rms,ave;
```

```
void TISRA_IMIA0(void)
{
  TISRA &= ~(1 << IMFA0);//clear flag
}
```

```
void ISR_RXI1 (void)
{
  data = RDR1;
  SSR1 &= ~(1 << RDRF);
}
```

```
void write(void)
{
  sprintf(&buf[0], "%u,%u\r\n",ave,v[0]);
  uart_str(&buf[0]);
  sum=0;
}
```

```
int main()
{
  uart_init();
  uart_strP("Inisialisasi Uart\r\n");
  adc_init();
```

```

uart_strP("Inisialisasi ADC\r\n");

TCR0_16 |= 0x23;
TIOR0 |= 0x99; TMDR |= 0x01; //pwm mode 0
TCNT0_16 = 0x0000; //counter mulai dari 0x0000
TSTR &= 0x00; //belum mulai counting

sei();
sum=0;

while(1)
{
    for (i=0;i<n;i++){
        v[i]=adc_getval_ch(CH1);
        a[i]=adc_getval_ch(CH1)*5000/1023;
        sum += a[i];
        _delay_ms(0.5);
        i++;
    }
    ave=sum*100/n;
    rms=sqrt(ave);
    for (i=0;i<1;i++){
        i++;
        _delay_ms(1000);
    }
    write();
    if (ave==1){
        GRA0 = 62500;
        GRB0 = 4375;
        TSTR |= (1 << STR0); //mulai counting
        TISRA |= (1 << IMIEA0); //Nyalakan interrupt IMIA
        TISRA_IMIA0();
    }else if (ave == 0){
        GRA0 = 62500;
        GRB0 = 1875;
        TSTR |= (1 << STR0); //mulai counting
        TISRA |= (1 << IMIEA0); //Nyalakan interrupt IMIA
        TISRA_IMIA0();
    }
}
return 0;
}

```

LAMPIRAN C

Pemrograman GUI Python

```
#####
from PySide.QtGui import *
from PySide.QtCore import *
from PySide.QtSql import *
from ui_testdb3 import Ui_testDb
from serialcom import SerialCom
import numpy as np
import matplotlib.pyplot as plt
from pylab import *

class testDb(QDialog, Ui_testDb):
    def __init__(self, parent=None):
        QDialog.__init__(self, parent)
        self.setupUi(self)
        self.viewTabel()

        self.x = 0.0
        self.y = 0.0

        self.com=SerialCom(self)
        self.com.readyRead.connect(self.serialHandler)
        self.com.start()

    def viewTabel(self):
        model = QSqlQueryModel()
        model.setQuery("SELECT * FROM Akuisisi_Sinyal_EMG")
        #model.setHeaderData(0,Qt.Horizontal,"Amplitudo")

        self.TabelView.setModel(model)
        self.TabelView.show()

    @Slot(str)
    def serialHandler(self,data):

        query = QSqlQuery()
        tmp = data.split(',')
        y=0.0
        y=float(tmp[0])
        volt=0.0
        volt=((float(y))*5/1023)-(2.09-0.027)

        self.y =volt
        self.grafik.addPoint(QPointF(self.x,self.y))
```



```

self.x += 1
dt = 0.0005
n=301
sum=0
sum1=0
sum2=0
Vrms=0.0
average1=0.0
j=0.0
j1=0.0

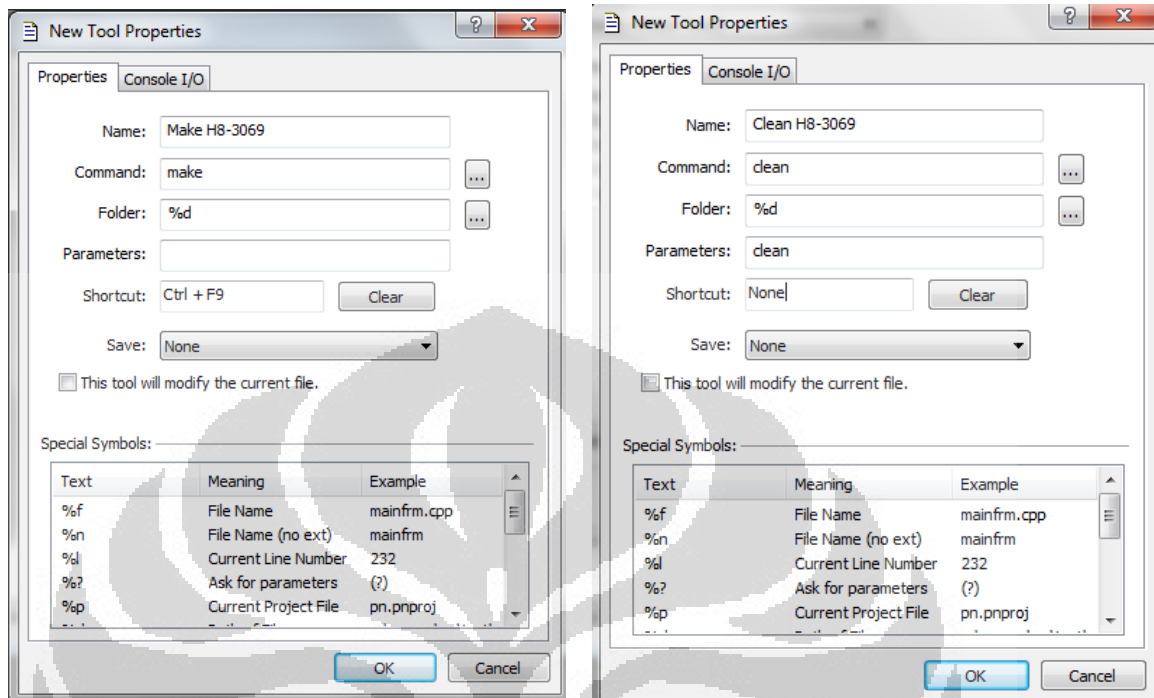
for i in range(n-1) :
    j=int(tmp[i])
    sum+=j**2
average=float (sum/n)
Vrms=(math.sqrt(average))
Vrms1=Vrms*5/1023-2.09
#self.com.write("$ %s\r" %str(Vrms1))

if average>0.0 :
    for i in range (n-1):
        j=int(tmp[i])
        dev=(j-average)**2
        sum1+=dev
    average1=float((sum1)/n)
    stand_dev=(%.2f%(math.sqrt(average1)))

def closeEvent(self,event):
    self.com.stop()

```

LAMPIRAN D

Pengaturan Perintah *make* dan *clean* Programmer Notpad di WinAVRTampilan dan Pengaturan *make file*

```

Programmers Notepad 2 - [Makefile *]
File Edit View Tools Window Help
Makefile *
PREFIX=h8300-elf
DEBUG_FLAGS=-gdwarf-2
# Specify compiler to be used
CC = $(PREFIX)-gcc
# Specify Assembler to be used
AS = $(PREFIX)-as
# Specify linker to be used
LD = $(PREFIX)-ld
# Specify CPU flag
CPU = -mh
# Common compiler flags
CFLAGS = -nostartfiles -wall
ALL_FLAGS = $(CFLAGS) $(DEBUG_FLAGS) $(CPU)
# Application name
APPNAME = imas2
# Specify all objects that you are going to link together
OBJS = start.o vects.o uart.o delay.o adc.o $(APPNAME).o
SOURCES = start.s vects.c uart.c delay.c adc.c $(APPNAME).c


$(APPNAME).mot : $(APPNAME).out
$(PREFIX)-objcopy -O srec $(APPNAME).out $@

$(APPNAME).lss : $(APPNAME).out
$(PREFIX)-objdump -h -S $< > $@

# Define ROMSTART if compiling for ROM
$(APPNAME).out : $(SOURCES) rom3068f.ld makefile
$(CC) $(ALL_FLAGS) -Trom3068f.ld -xlinker -Map $(APPNAME).map -o $@ $(SOURCES) -lptc -loptm

```

LAMPIRAN E



Key features

- Highly conductive wet gel
- Superior adhesion
- Large measuring area
- Unique offset concept
- High-quality Ag/AgCl sensor

Ambu[®] Blue Sensor R

The large Ambu Blue Sensor R features highly conductive wet gel, superior adhesion, and a large measuring area to ensure an optimal signal during stress tests and medium- to long-term applications. Thanks to the comfortable foam backing, the electrode is gentle to the skin.

DIMENSIONS

Electrode size (max. L x W or diameter in mm)	57 x 48
Skin contact size (max. L x W or diameter in mm)	48
Adhesive area (in mm ²)	1555
Height excluding connector/wire (in mm)	2

SENSOR

Sensor material	Silver/silver chloride (Ag/AgCl)
Gel system	Wet gel
Sensor area (in mm ²)	27
Gel area/measuring area (in mm ²)	254

LIFETIME

In opened pouches	1 month
Unopened pouches*	24 months

*from date of production

X-RAY & MRI

X-ray-translucent	No
MRI-safe	No

ELECTRICAL DATA (ANSI/AAM)

AC impedance - typical	400 Ω
DC offset voltage - typical	0.2 mV
Defibrillation overload recovery - typical	6.8 mV
Rate of change of polarization potential - typical	0.2 mV/s
Combined offset instability and internal noise	<15 μV
Bias current tolerance (over 8 hours)	<5 mV

Airway Management | ■ Patient Monitoring & Diagnostics

ENVIRONMENT

PVC-free electrode	No
Latex-free electrode	Yes
PVC-free lead wire	-
Latex-free lead wire	-
PVC-free packaging	Yes

STANDARD PACKAGING

Quantity/liner	1	
Quantity/pouch	3, 10, 25	
Quantity/box	252/504, 500/1000	
Minimum sales	504, 1000	

OPTIONS

Connectors	S, A, F
Pre-wired	-
Pre-wired connectors	-

AVAILABLE CONFIGURATIONS*

Quantity/pouch	Qty per inner/outer box		
	3	10	25
R-00-S	252/504	500/1000	500/1000
R-00-A, F		500/1000	

■	PRODUCT
■	WIRE LENGTH
■	CONNECTOR

R-00-S

* Some configurations may not be available worldwide.

MATERIALS - ELECTRODE

Bio-compatible	Yes
Sensor	Silver/silver chloride (Ag/AgCl)
Sponge	Polyurethane foam (PUR)
Outer carrier	Polyvinyl chloride foam (PVC)
Outer adhesive	Acrylate
Inner carrier (if any)	-
Inner adhesive (if any)	-
Upper part	Polyvinyl chloride film (PVC)
Fitting	Polypropylene (PP)
Connector (stud)	Nickel-plated brass
Lead wire core	-
Lead wire insulation	-
Lead wire connector	-
Liner	Polyester/polyethylene film (PET/PE)

MATERIALS - PACKAGING

Pouches, inner layer	Polyethylene (PE)
Pouches, centre layer	Aluminium (Al)
Pouches, outer layer	Polyester (PET)
Boxes	Cardboard

CONNECTORS



S = Stud



A = 4 mm fitting





Low Cost Low Power Instrumentation Amplifier

AD620

FEATURES

Easy to use

- Gain set with one external resistor (Gain range 1 to 10,000)
- Wide power supply range (± 2.3 V to ± 18 V)
- Higher performance than 3 op amp IA designs
- Available in 8-lead DIP and SOIC packaging
- Low power, 1.3 mA max supply current

Excellent dc performance (B grade)

- 50 μ V max, input offset voltage
- 0.6 μ V/ $^{\circ}$ C max, input offset drift
- 1.0 nA max, input bias current
- 100 dB min common-mode rejection ratio (G = 10)

Low noise

- 9 nV/ $\sqrt{\text{Hz}}$ @ 1 kHz, input voltage noise
- 0.28 μ V p-p noise (0.1 Hz to 10 Hz)

Excellent ac specifications

- 120 kHz bandwidth (G = 100)
- 15 μ s settling time to 0.01%

APPLICATIONS

- Weigh scales
- ECG and medical instrumentation
- Transducer interface
- Data acquisition systems
- Industrial process controls
- Battery-powered and portable equipment

CONNECTION DIAGRAM

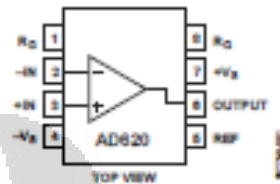


Figure 1. 8-Lead PDIP (N), CERDIP (C), and SOIC (R) Packages

PRODUCT DESCRIPTION

The AD620 is a low cost, high accuracy instrumentation amplifier that requires only one external resistor to set gains of 1 to 10,000. Furthermore, the AD620 features 8-lead SOIC and DIP packaging that is smaller than discrete designs and offers lower power (only 1.3 mA max supply current), making it a good fit for battery-powered, portable (or remote) applications.

The AD620, with its high accuracy of 40 ppm maximum nonlinearity, low offset voltage of 50 μ V max, and offset drift of 0.6 μ V/ $^{\circ}$ C max, is ideal for use in precision data acquisition systems, such as weigh scales and transducer interfaces. Furthermore, the low noise, low input bias current, and low power of the AD620 make it well suited for medical applications, such as ECG and noninvasive blood pressure monitors.

The low input bias current of 1.0 nA max is made possible with the use of Superbeta processing in the input stage. The AD620 works well as a preamplifier due to its low input voltage noise of 9 nV/ $\sqrt{\text{Hz}}$ at 1 kHz, 0.28 μ V p-p in the 0.1 Hz to 10 Hz band, and 0.1 pA/ $\sqrt{\text{Hz}}$ input current noise. Also, the AD620 is well suited for multiplexed applications with its settling time of 15 μ s to 0.01%, and its cost is low enough to enable designs with one-in-amp per channel.

Table 1. Next Generation Upgrades for AD620

Part	Comment
AD8221	Better specs at lower price
AD8222	Dual channel or differential out
AD8226	Low power, wide input range
AD8220	JFET input
AD8228	Best gain accuracy
AD8295	+2 precision op amps or differential out
AD8429	Ultra low noise

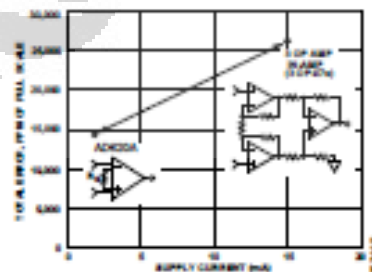


Figure 2. Three Op-Amp IA Designs vs. AD620

Rev. H

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
 Tel: 781.329.4700 www.analog.com
 Fax: 781.326.8700 © 2003–2011 Analog Devices, Inc. All rights reserved.

SPECIFICATIONS

Typical @ 25°C, $V_s = \pm 15$ V, and $R_b = 2$ k Ω , unless otherwise noted.

Table 2.

Parameter	Conditions	AD620A			AD620B			AD620S ¹			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
GAIN	$G = 1 + (49.4 \text{ k}\Omega/R_b)$										
Gain Range		1		10,000	1		10,000	1		10,000	
Gain Error ²	$V_{out} = \pm 10$ V										
G = 1			0.03	0.10		0.01	0.02		0.03	0.10	%
G = 10			0.15	0.30		0.10	0.15		0.15	0.30	%
G = 100			0.15	0.30		0.10	0.15		0.15	0.30	%
G = 1000			0.40	0.70		0.35	0.50		0.40	0.70	%
Nonlinearity	$V_{out} = -10$ V to $+10$ V										
G = 1–1000	$R_b = 10$ k Ω		10	40		10	40		10	40	ppm
G = 1–100	$R_b = 2$ k Ω		10	95		10	95		10	95	ppm
Gain vs. Temperature	G = 1			10			10			10	ppm/°C
	Gain > 1 ²			-50			-50			-50	ppm/°C
VOLTAGE OFFSET	(Total RTI Error = $V_{os} + V_{eor}/G$)										
Input Offset, V_{os}	$V_s = \pm 5$ V to ± 15 V		30	125		15	50		30	125	μ V
Overtemperature	$V_s = \pm 5$ V to ± 15 V			185			85			225	μ V
Average TC	$V_s = \pm 5$ V to ± 15 V		0.3	1.0		0.1	0.6		0.3	1.0	μ V/°C
Output Offset, V_{eor}	$V_s = \pm 15$ V		400	1000		200	500		400	1000	μ V
	$V_s = \pm 5$ V			1500			750			1500	μ V
Overtemperature	$V_s = \pm 5$ V to ± 15 V			2000			1000			2000	μ V
Average TC	$V_s = \pm 5$ V to ± 15 V		5.0	15		2.5	7.0		5.0	15	μ V/°C
Offset Referred to the Input vs. Supply (PSR)	$V_s = \pm 2.3$ V to ± 18 V										
G = 1		80	100		80	100		80	100		dB
G = 10		95	120		100	120		95	120		dB
G = 100		110	140		120	140		110	140		dB
G = 1000		110	140		120	140		110	140		dB
INPUT CURRENT											
Input Bias Current			0.5	2.0		0.5	1.0		0.5	2	nA
Overtemperature				2.5			1.5			4	nA
Average TC			3.0			3.0			8.0		pA/°C
Input Offset Current			0.3	1.0		0.3	0.5		0.3	1.0	nA
Overtemperature				1.5			0.75			2.0	nA
Average TC			1.5			1.5			8.0		pA/°C
INPUT											
Input Impedance											
Differential			10 ¹²			10 ¹²			10 ¹²		G Ω , pF
Common-Mode			10 ¹²			10 ¹²			10 ¹²		G Ω , pF
Input Voltage Range ³	$V_s = \pm 2.3$ V to ± 5 V	$-V_s + 1.9$		$+V_s - 1.2$	$-V_s + 1.9$		$+V_s - 1.2$	$-V_s + 1.9$		$+V_s - 1.2$	V
Overtemperature	$V_s = \pm 5$ V to ± 18 V	$-V_s + 2.1$		$+V_s - 1.3$	$-V_s + 2.1$		$+V_s - 1.3$	$-V_s + 2.1$		$+V_s - 1.3$	V
		$-V_s + 1.9$		$+V_s - 1.4$	$-V_s + 1.9$		$+V_s - 1.4$	$-V_s + 1.9$		$+V_s - 1.4$	V
Overtemperature		$-V_s + 2.1$		$+V_s - 1.4$	$-V_s + 2.1$		$+V_s - 2.1$	$-V_s + 2.3$		$+V_s - 1.4$	V

Regardless of the system in which it is being used, the AD620 provides greater accuracy at low power and price. In simple systems, absolute accuracy and drift errors are by far the most significant contributors to error. In more complex systems with an intelligent processor, an autogain/autozero cycle removes all absolute accuracy and drift errors, leaving only the resolution errors of gain, nonlinearity, and noise, thus allowing full 14-bit accuracy.

Note that for the homebrew circuit, the OP07 specifications for input voltage offset and noise have been multiplied by $\sqrt{2}$. This is because a three op amp type in-amp has two op amps at its inputs, both contributing to the overall input error.

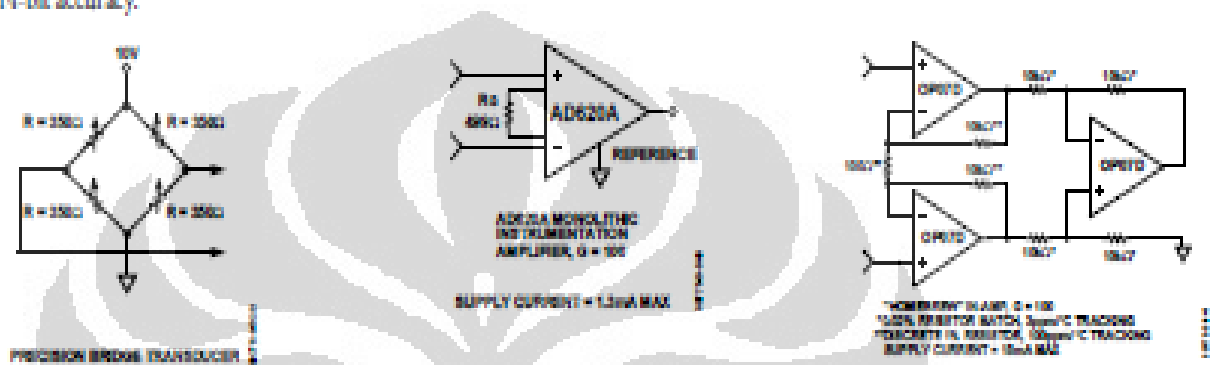


Figure 37. Make vs. Buy

Table 4. Make vs. Buy Error Budget

Error Source	AD620 Circuit Calculation	"Homebrew" Circuit Calculation	Error, ppm of Full Scale	
			AD620	Homebrew
ABSOLUTE ACCURACY at $T_a = 25^\circ\text{C}$				
Input Offset Voltage, μV	125 $\mu\text{V}/20\text{ mV}$	(150 $\mu\text{V} \times \sqrt{2})/20\text{ mV}$	6,250	10,607
Output Offset Voltage, μV	1000 $\mu\text{V}/100\text{ mV}/20\text{ mV}$	(150 $\mu\text{V} \times 2)/100/20\text{ mV}$	500	150
Input Offset Current, nA	2 nA $\times 350\ \Omega/20\text{ mV}$	(6 nA $\times 350\ \Omega)/20\text{ mV}$	18	53
CMR, dB	110 dB(3.16 ppm) $\times 5\text{ V}/20\text{ mV}$	(0.02% Match $\times 5\text{ V})/20\text{ mV}/100$	791	500
		Total Absolute Error	7,559	11,310
DRIFT TO 85°C				
Gain Drift, ppm/ $^\circ\text{C}$	(50 ppm + 10 ppm) $\times 60^\circ\text{C}$	100 ppm/ $^\circ\text{C}$ Track $\times 60^\circ\text{C}$	3,600	6,000
Input Offset Voltage Drift, $\mu\text{V}/^\circ\text{C}$	1 $\mu\text{V}/^\circ\text{C} \times 60^\circ\text{C}/20\text{ mV}$	(2.5 $\mu\text{V}/^\circ\text{C} \times \sqrt{2} \times 60^\circ\text{C})/20\text{ mV}$	3,000	10,607
Output Offset Voltage Drift, $\mu\text{V}/^\circ\text{C}$	15 $\mu\text{V}/^\circ\text{C} \times 60^\circ\text{C}/100\text{ mV}/20\text{ mV}$	(2.5 $\mu\text{V}/^\circ\text{C} \times 2 \times 60^\circ\text{C})/100\text{ mV}/20\text{ mV}$	450	150
		Total Drift Error	7,050	16,757
RESOLUTION				
Gain Nonlinearity, ppm of Full Scale	40 ppm	40 ppm	40	40
Typ 0.1 Hz to 10 Hz Voltage Noise, μV p-p	0.28 μV p-p/20 mV	(0.38 μV p-p $\times \sqrt{2})/20\text{ mV}$	14	27
		Total Resolution Error	54	67
		Grand Total Error	14,663	28,134

G = 100, $V_s = \pm 15\text{ V}$.

(All errors are min/max and referred to input.)

LAMPIRAN G



Web Site: www.parallax.com
 Forum: forums.parallax.com
 Sales: sales@parallax.com
 Technical: support@parallax.com

Office: (916) 624-8333
 Fax: (916) 624-8000
 Sales: (888) 512-1024
 Tech Support: (888) 997-6267

Parallax Standard Servo (#900-00005)

The Parallax Standard Servo is ideal for robotics and basic movement projects.

Features

- Holds any position between 0 and 180 degrees
- 43.1 oz-in torque at 6 V
- Accepts four mounting screws
- Easy to interface with any Parallax microcontroller
- High precision gear made of the POM (polyacetal) resin makes the operation smooth causing no backlash.
- Manufactured for Parallax exclusively by Futaba



Technical Specifications

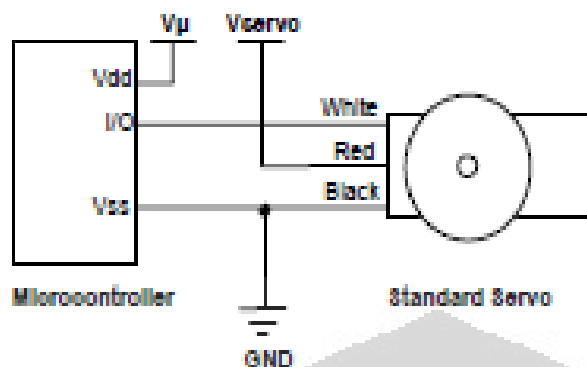
- Power requirements: 4 to 6 VDC* (see Power Requirement Notes below)
- Maximum current draw: 140 +/- 50 mA at 6 VDC when operating in no load conditions
15 mA when in static state
- Communication: Pulse-width modulation
- Dimensions approx 2.2 x 0.8 x 1.6 in (5.58x 1.9 x 40.6 cm) excluding servo horn
- Operating temperature range: 14 to 122°F (-10 to 50°C)
- Weight: 1.55 oz (44 g)

*Power Requirement Notes

Futaba specifies 4-6 VDC for this servo. However, we find that this servo is tolerant of a 9 V battery for short periods of time when there is no load, as used in some activities in the Stamps in Class series of tutorials. (Slight jittering may be observed when batteries are fresh; this does not cause damage). Do not use this servo with an unregulated wall-mount supply, or a regulated wall mount supply exceeding 6 VDC.

Servo current draw can spike while under load. Be sure that your application's power supply and voltage regulator is prepared to supply adequate current for all servos used. Do not try to power this servo directly from a BASIC Stamp module's Vdd or Vin pins; do not connect the servo's Vss line directly to the BASIC Stamp module's Vss pin.

Quick-Start Circuit



V_p = microcontroller voltage supply

V_{servo} = 4 to 6 VDC, regulated or battery (See Error! Reference source not found., page Error! Bookmark not defined.)

I/O = PWM TTL or CMOS output signal, 3.3 to 5 V, not to exceed V_{servo} + 0.2 V

Specifications

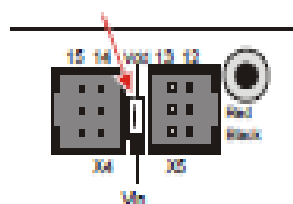
Pin	Name	Description	Minimum	Typical	Maximum	Units
1 (White)	Signal	Input; TTL or CMOS	3.3	5.0	V _{servo} + 0.2	V
2 (Red)	V _{servo}	Power Supply	4.0	5.0	6.0	V
3 (Black)	V _{ss}	Ground		0		V

Power Precautions

- Do not use this servo with an unregulated wall-mount supply. Such power supplies may deliver variable voltage far above the stated voltage.
- Do not power this servo through the BASIC Stamp Module's V_{dd} pin.
- Servo current draw can spike while under peak load; be sure your application's regulator is prepared to supply adequate current for all servos used in combination.
- Some Stamps in Class tutorials, such as "What's a Microcontroller?" instruct the user to power these servos with a 9V battery when using a HomeWork Board; this does not cause damage.

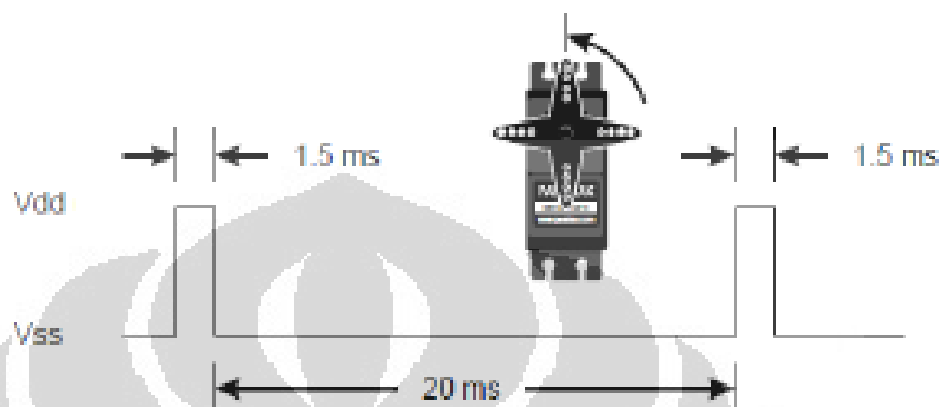
Board of Education Jumper Connection

When connecting the servo to the Board of Education Rev C's servo header, be sure the jumper is set to V_{dd} as shown in the figure below. Failure to place the jumper at this setting can cause damage your servo.



Communication Protocol

The Parallax Standard Servo is controlled through pulse width modulation, where the position of the servo shaft is dependent on the duration of the pulse. In order to hold its position, the servo needs to receive a pulse every 20 ms. Below is a sample timing diagram for the center position of the Parallax Standard Servo.



BASIC Stamp Programming Examples

PBASIC has a PULSOUT command that sets the I/O pin to an output and sends a pulse of a specified duration.

PULSOUT *Pin, Duration*

The example shown below for a BASIC Stamp 2 causes a servo connected to BASIC Stamp I/O pin 1 to turn to and hold its center position for approximately 5 seconds.

```

* ($STAMP BS2)
* ($PBASIC 2.5)

counter VAR Word

FOR counter = 1 TO 100

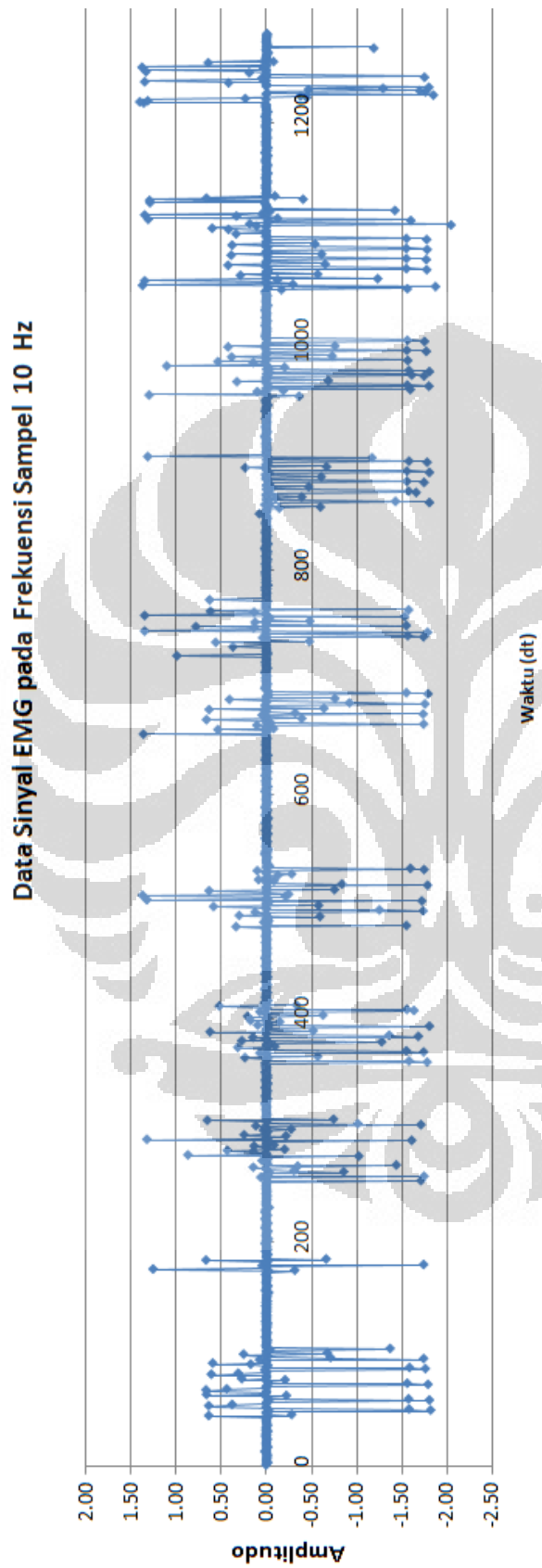
    PULSOUT 1, 750
    PAUSE 20

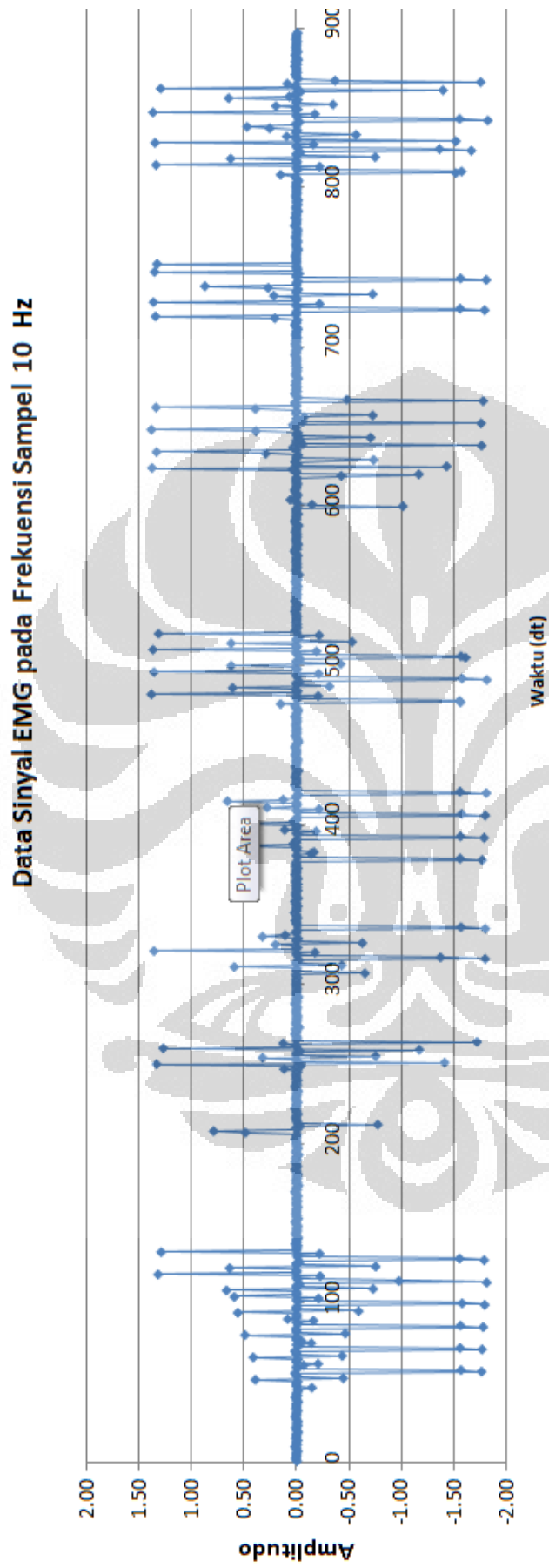
NEXT
  
```

For more examples with the BASIC Stamp 2, see "What's a Microcontroller?" Chapter 4, available for free download from the 90005 product page at www.parallax.com.

Different BASIC Stamp modules use different units for the PULSOUT command's *Duration* argument. When adapting BS2 code to another BASIC Stamp model, you may need to make adjustments. The table below lists the PULSOUT ranges for each BASIC Stamp microcontroller. See the BASIC Stamp Manual or BASIC Stamp Editor Help for more information.

LAMPIRAN H





Data Sinyal EMG pada Frekuensi Sampel 10 Hz

