

**PENGENDALIAN BILATERAL ROBOT BERBASIS PC
DENGAN MIKROKONTROLER H8/3052F SEBAGAI UNIT
DATA AKUISISI**

SKRIPSI

HARTANTO RAHARJO

04 04 03 043 1



**UNIVERSITAS INDONESIA
FAKULTAS TEKNIK
PROGRAM TEKNIK ELEKTRO
DEPOK
JULI 2008**

**PENGENDALIAN BILATERAL ROBOT BERBASIS PC
DENGAN MIKROKONTROLER H8/3052F SEBAGAI UNIT
DATA AKUISISI**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

HARTANTO RAHARJO

04 04 03 043 1



**UNIVERSITAS INDONESIA
FAKULTAS TEKNIK
PROGRAM TEKNIK ELEKTRO
DEPOK
JULI 2008**

HALAMAN PERNYATAAN ORISINALITAS

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

Nama : Hartanto Raharjo

NPM : 0404030431

Tanda Tangan :

Tanggal : 14 Juli 2008



HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Hartanto Raharjo
NPM : 0404030431
Program Studi : Teknik Elektro
Judul Skripsi : Pengendalian Bilateral Robot Berbasis PC Dengan
Mikrokontroler H8/3052F Sebagai Unit Data
Akuisisi

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Dr. Abdul Muis, ST., M.Eng (.....)
Penguji : Dr. Ir. Feri Yusivar, M.Eng (.....)
Penguji : Dr. Ir. Ridwan Gunawan, MT (.....)

Ditetapkan di : Depok

Tanggal : 16 Juli 2008

UCAPAN TERIMA KASIH

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Abdul Muis, ST., M.Eng, Ph.D selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
2. Orang tua dan keluarga saya yang telah memberikan bantuan dukungan material dan moral; dan
3. Para sahabat yang telah banyak membantu saya dalam menyelesaikan skripsi ini.

Akhir kata, saya berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 14 Juli 2008

Penulis

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

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

Nama : Hartanto Raharjo
NPM : 0404030431
Program Studi : Teknik Elektro
Departemen : Elektro
Fakultas : Teknik
Jenis Karya : Skripsi

demikian 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:

**Pengendalian Bilateral Robot Berbasis PC dengan
Mikrokontroler H8/3052F sebagai Unit Data Akuisisi**

berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta dan pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 14 Juli 2008

Yang menyatakan

(Hartanto Raharjo)

ABSTRAK

Nama : Hartanto Raharjo
Program Studi : Teknik Elektro
Judul : Pengendalian Bilateral Robot berbasis PC dengan Mikrokontroler H8/3052F sebagai Unit Data Akuisisi.

Sistem telerobotik banyak dikembangkan untuk melakukan pekerjaan yang berbahaya yang tidak dapat diakses oleh operator secara langsung. Pada pengendali telerobotik, umumnya terdapat dua buah manipulator yang terkoneksi satu sama lain melalui suatu jaringan. Pada sistem telerobotik biasa, gerakan yang dibuat oleh operator pada manipulator master akan mengirimkan sinyal informasi yang cukup untuk membuat manipulator pada sisi *slave* mengikuti gerakan manipulator *master*. Suatu sistem telerobotik dapat dikategorikan sebagai bilateral robot apabila terdapat pertukaran sinyal informasi sebagai masukan pengendali bergerak dari *master* ke *slave* dan sebaliknya. Sehingga memungkinkan operator pada sisi *master* untuk mengendalikan manipulator *slave* dan merasakan objek yang dirasakan/ disentuh oleh manipulator pada sisi *slave*.

Dalam Skripsi ini dilakukan pembuatan perangkat keras, dan perangkat lunak pada sisi komputer dan mikrokontroler H8/3052F untuk mendukung pengujian terhadap performa aplikasi bilateral robot yang dibuat.

Evaluasi kinerja sistem dilakukan dengan memperhatikan respon manipulator *slave* terhadap perubahan posisi sudut yang diberikan pada manipulator *master*, serta respon sistem ketika terdapat objek yang menahan pergerakan manipulator *slave*.

Kata Kunci:

Telerobotik, Bilateral Robot, Komunikasi Serial, RS-232, Jaringan, Mikrokontroler H8/3052F

ABSTRACT

Name : Hartanto Raharjo
Study Program : Electrical Engineering
Title : PC Based Bilateral Robot Control With Microcontroller
H8/3052F As Data Acquisition Unit

Tele robotic systems have been developed to perform dangerous tasks or any other tasks, which are inaccessible to operators directly. Generally, in tele robotic control, there are two manipulators, which is connected one to another through network. In common tele robotic system, movement made by operator on master manipulator will send enough information signal(s) to make manipulator on slave side follow master manipulator's movement. Tele robotic system can be categorized as bilateral robot if the information signal flows in both directions between master and slave. So, operator on master's side can controls slave manipulator, and senses the object, touched by slave manipulator.

In this bachelor thesis, hardware, and software on both computers and H8/3052F microcontrollers are built. These are done to support performance test of bilateral robot application that is made before.

Performance evaluation of this system takes focus on slave manipulator's response to the change of master manipulator's angle position, and system's response when there is object, which is hold slave manipulator's movement.

Keywords:

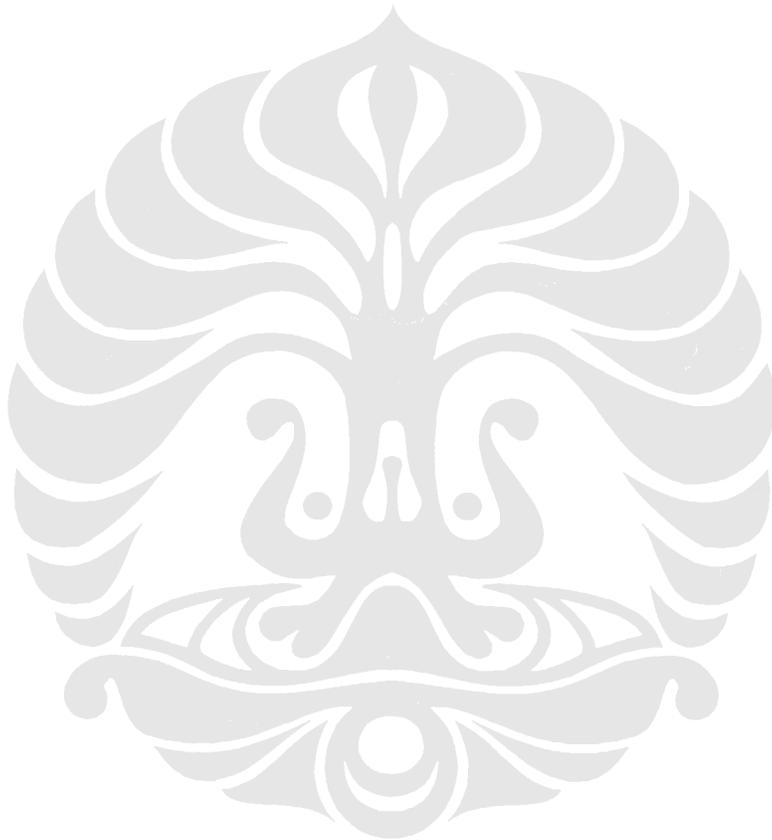
Tele robotic, Bilateral Robot, Serial Communication, RS-232, Network, H8/3052F Microcontroller

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
UCAPAN TERIMA KASIH.....	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH.....	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xii
DAFTAR SINGKATAN	xiii
DAFTAR ISTILAH/ SIMBOL.....	xiv
DAFTAR LAMPIRAN.....	xv
1. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Tujuan Penelitian	2
1.4. Pembatasan Masalah.....	2
1.5. Metodologi Penelitian.....	2
1.6. Sistematika Penelitian.....	3
1.7. Sistematika Penulisan	3
2. LANDASAN TEORI	4
2.1. Motor DC Servo.....	4
2.2. Potensiometer Sebagai Sensor Putaran.....	4
2.3. Spesifikasi Mikrokontroler H8/3052F [2]	5
2.4. Konverter A/D pada Mikrokontroler H8/3052F [2]	7
2.4.1. Overview ADC pada Mikrokontroler H8/3052F.....	7
2.4.2. Konfigurasi dan Deskripsi Register ADC Mikrokontroler H8/3052F.....	7
2.4.2.1. Register Data A/D (ADDRA – ADDRD)	7
2.4.2.2. Register Kontrol/ Status A/D (ADCSR).....	8
2.4.2.3. Register Kontrol A/D (ADCR).....	10
2.5. Unit Pewaktuan Terintegrasi pada Mikrokontroler H8/3052F [2].....	11
2.5.1. Register – register ITU.....	12
2.5.1.1. Register Mulai Pewaktuan (TSTR)	12
2.5.1.2. Register Mode Pewaktuan (TMDR).....	12
2.5.1.3. Register Timer Counter (TCNT)	13
2.5.1.4. Register General (GRA, GRB).....	13
2.5.1.5. Register Pengontrol Pewaktuan (TCR)	13
2.5.1.6. Timer I/O Control Register (TIOR)	14
2.5.2. Operasi ITU pada Mode PWM.....	14
2.6. Komunikasi Serial pada Mikrokontroler H8/3052F [2].....	15
2.6.1. Pin dan Register SCI.....	15
2.6.1.1. Shift Register Penerima (RSR).....	15

2.6.1.2. Register Data Penerima (RDR)	15
2.6.1.3. Shift Register Pengirim (TSR)	16
2.6.1.4. Register Data Penerima (TDR)	16
2.6.1.5. Register Mode Serial (SMR)	16
2.6.1.6. Register Kontrol Serial (SCR).....	18
2.6.1.7. Register Status Serial (SSR).....	19
2.6.1.8. Register Bit Rate (BRR).....	21
2.6.2. Operasi SCI pada Mode Asinkron pada Mikrokontroler H8/3052F.....	22
2.6.2.1. Inisialisasi Serial pada SCI.....	22
2.6.2.2. Transmisi Data Serial pada SCI	24
2.6.2.3. Penerimaan Data Serial pada SCI.....	25
2.7. Komunikasi Serial dengan RS-232C	26
2.7.1. Konfigurasi Pin RS-232C DB-9	26
2.7.2. Konfigurasi Hubungan Kabel Pin SCI dengan RS-232C DB-9	26
2.8. Pemrograman Port Serial dengan Bahasa Pemrograman C	27
2.9. Dasar Pemrograman Soket UDP	28
2.9.1. Pengenalan Terhadap Soket API	28
2.9.2. Pemrograman Soket UDP dengan Bahasa Pemrograman C	29
3. PERANCANGAN	31
3.1. Konsep Dasar	31
3.2. Perancangan Perangkat Keras	32
3.2.1. Penggunaan Mikrokontroler H8/3052F	32
3.2.2. Spesifikasi PC	33
3.2.3. Rangkaian Filter <i>Low Pass</i>	33
3.2.4. Rangkaian Voltage Regulator	34
3.2.5. Rangkaian <i>Optocoupler</i>	34
3.2.6. Perancangan Lengan Motor	35
3.2.7. Hasil Perancangan Perangkat Keras	35
3.3. Perancangan Perangkat Lunak	36
3.3.1. Program pada Sisi Mikrokontroler.....	36
3.3.2. Program pada Sisi PC	38
3.4. Perancangan Program Grafis dengan OpenGL.....	40
3.5. Kalibrasi Nilai Keluaran Konverter A/D dan GRB	40
4. ANALISA.....	42
4.1. Pengujian Respon Aktuator Slave	42
4.1.1. Prosedur Pengujian Respon Aktuator Slave	42
4.1.2. Hasil Pengujian Respon Aktuator Slave	43
4.1.3. Error Respon Aktuator Slave Terhadap Master.....	46
4.2. Pengujian Pengaruh Objek yang Menahan Pergerakan Manipulator Slave	47
4.2.1. Prosedur Pengujian Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave	47
4.2.2. Hasil Pengujian Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave	48

5. KESIMPULAN	51
DAFTAR ACUAN	52
DAFTAR PUSTAKA.....	53
LAMPIRAN.....	54

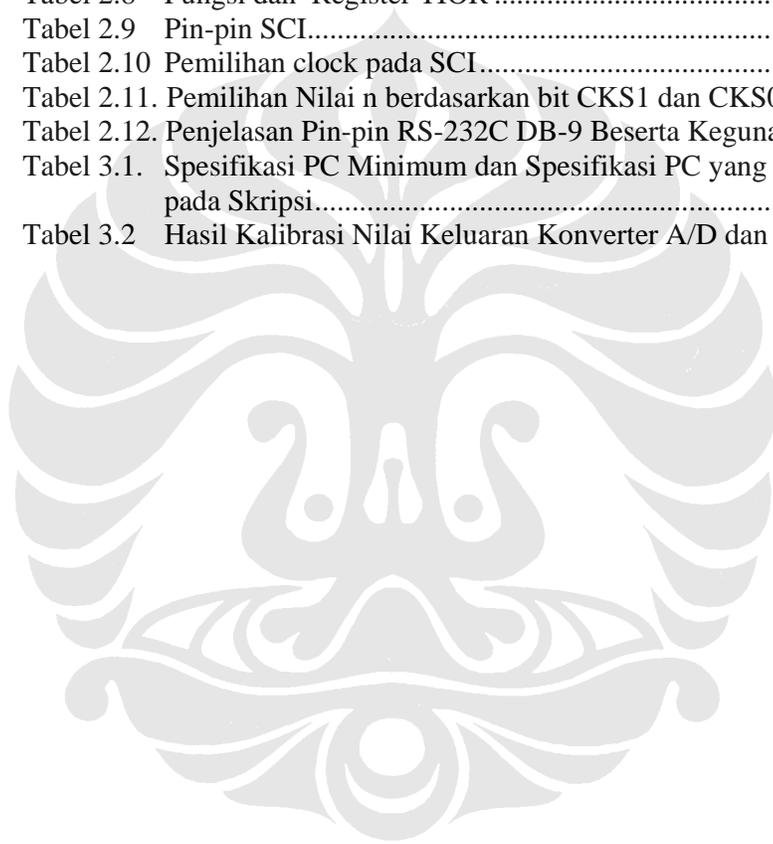


DAFTAR GAMBAR

Gambar 1.1. Ilustrasi Sistem Bilateral Kontrol yang Dirancang	1
Gambar 2.1. Pengaturan Sudut Motor dengan Sinyal PWM	4
Gambar 2.2 Penggunaan Potensiometer sebagai Sensor Putaran	5
Gambar 2.3. Struktur dari Register Data A/D.....	7
Gambar 2.4. Struktur Register Kontrol/ Status A/D (ADCSR)	8
Gambar 2.5. Struktur Register Kontrol A/D (ADCR)	10
Gambar 2.6. Diagram Alir Operasi ADC pada Mode Single (a); dan Proses Inisialisasi ADC (b)	11
Gambar 2.7. Struktur TSTR.....	12
Gambar 2.8. Struktur TMDR	12
Gambar 2.9. Diagram Alir Operasi ITU pada Mode PWM.....	14
Gambar 2.10 Struktur dari Register Mode Serial (SMR)	16
Gambar 2.11. Struktur Register Kontrol Serial (SCR)	18
Gambar 2.12. Struktur Register Status Serial (SSR)	19
Gambar 2.14. Diagram Alir proses inisialisasi serial pada operasi asinkron.....	23
Gambar 2.15. Diagram Alir proses transmisi data serial pada operasi asinkron ..	24
Gambar 2.16. Diagram Alir proses penerimaan data serial pada operasi asinkron.....	25
Gambar 2.17 RS-232C DB-9 Konektor Tampak Depan	26
Gambar 2.18 Koneksi Kabel Pin Serial SCI dengan Port Serial Komputer	27
Gambar 3.1. Diagram Blok Konsep Dasar Sistem Bilateral Kontrol yang Dirancang.....	31
Gambar 3.2. Rancangan Rangkaian Filter Low Pass.....	33
Gambar 3.3. Rancangan Rangkaian Voltage Regulator	34
Gambar 3.4. Rancangan Rangkaian Optocoupler	35
Gambar 3.5. Rangkaian hasil perancangan perangkat keras untuk pengendalian bilateral robot.....	36
Gambar 3.6. Diagram Alir Algoritma Program pada Sisi Mikrokontroler.....	37
Gambar 3.7. Diagram Alir Algoritma Program Pada Sisi PC	39
Gambar 3.8. Ilustrasi Kerja Program Grafis dengan OpenGL.....	40
Gambar 4.1. Ilustrasi proses pengujian respon aktuator slave.....	42
Gambar 4.1 Respon Aktuator Slave terhadap Perubahan Sudut pada Aktuator Master dengan Tempo Lambat	43
Gambar 4.2. Perbandingan Nilai GRB Master dan Slave untuk Tempo Lambat .	44
Gambar 4.3 Respon Aktuator Slave terhadap Perubahan Sudut pada Aktuator Master dengan Tempo Cepat	45
Gambar 4.4. Perbandingan Nilai GRB Master dan Slave untuk Tempo Cepat....	45
Gambar 4.5. Error Posisi Sudut Slave Terhadap Master untuk Tempo Lambat...	46
Gambar 4.6. Error Posisi Sudut Slave Terhadap Master untuk Tempo Cepat.....	47
Gambar 4.7 Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave.....	48
Gambar 4.8. Perbandingan Nilai GRB Master dan Slave Pada Pengujian Adanya Objek yang Menahan Pergerakan Slave.....	49

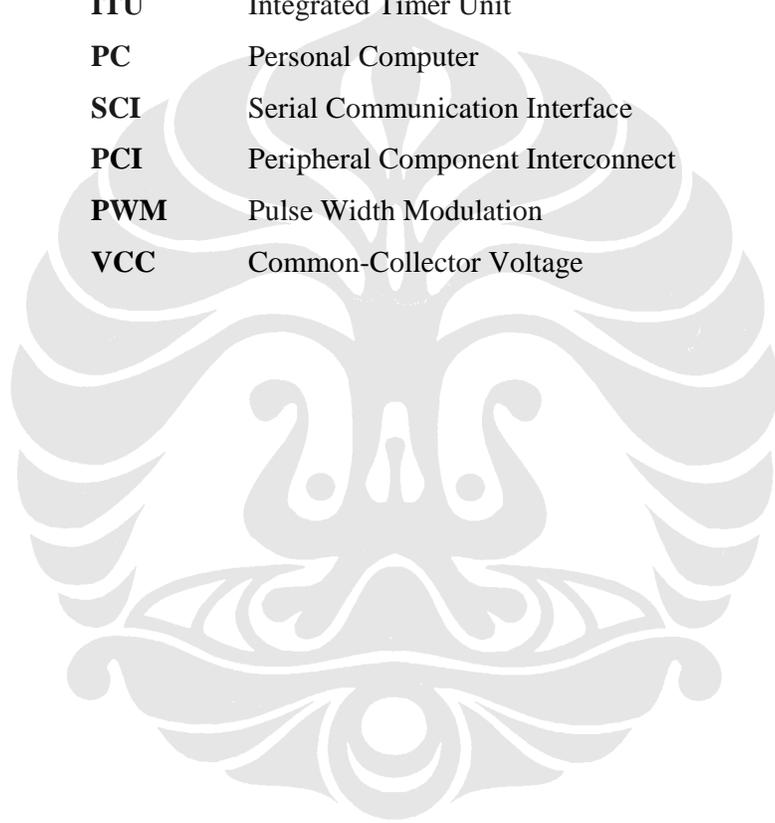
DAFTAR TABEL

Tabel 2.1. Pengaruh Isi Bit ADF Pada Proses Konversi A/D.....	8
Tabel 2.2. Pengaruh Isi Bit ADIE Pada Proses Konversi A/D.....	8
Tabel 2.3. Pengaruh Isi Bit ADST Pada Proses Konversi A/D.....	9
Tabel 2.4. Pemilihan Mode Konversi Melalui Bit SCAN.....	9
Tabel 2.5. Pemilihan Waktu Konversi Melalui Bit CKS.....	9
Tabel 2.6. Pemilihan Kanal Konversi Melalui Bit-bit CH.....	10
Tabel 2.7. Fungsi dan Struktur TCR.....	13
Tabel 2.8. Fungsi dan Register TIOR.....	14
Tabel 2.9. Pin-pin SCI.....	15
Tabel 2.10. Pemilihan clock pada SCI.....	17
Tabel 2.11. Pemilihan Nilai n berdasarkan bit CKS1 dan CKS0.....	21
Tabel 2.12. Penjelasan Pin-pin RS-232C DB-9 Beserta Kegunaannya.....	26
Tabel 3.1. Spesifikasi PC Minimum dan Spesifikasi PC yang Digunakan pada Skripsi.....	33
Tabel 3.2. Hasil Kalibrasi Nilai Keluaran Konverter A/D dan GRB.....	41



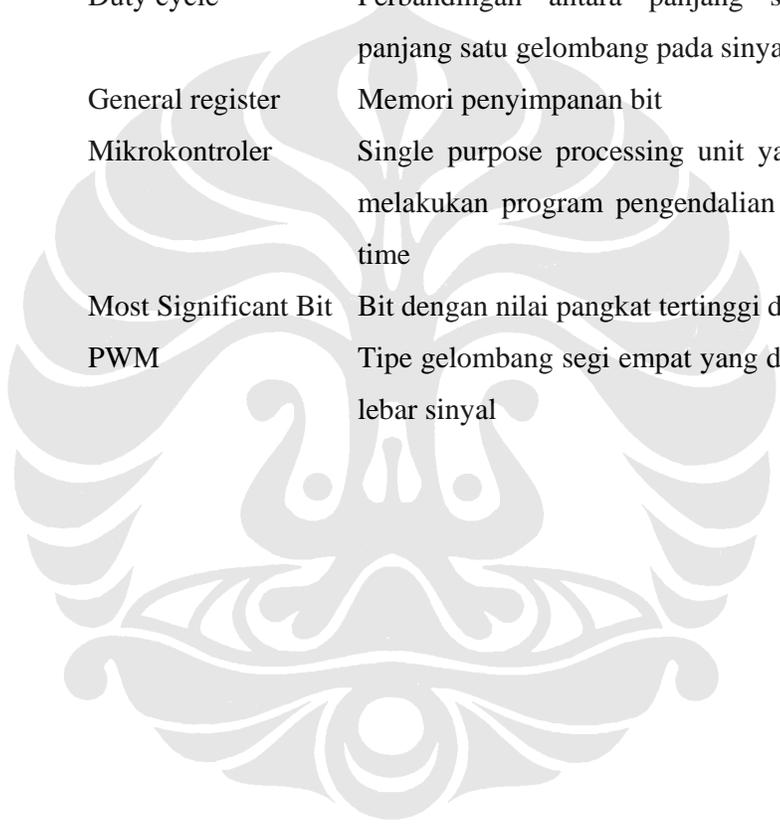
DAFTAR SINGKATAN

ADC	Analog to Digital Converter
CPU	Central Processing Unit
DAQ	Data Acquisition
GND	Ground
I/O	Input/ Output
ITU	Integrated Timer Unit
PC	Personal Computer
SCI	Serial Communication Interface
PCI	Peripheral Component Interconnect
PWM	Pulse Width Modulation
VCC	Common-Collector Voltage



DAFTAR ISTILAH/ SIMBOL

Φ	Frekuensi clock mikrokontroler
Bit	Digit dalam sistem bilangan biner, dapat bernilai 0 atau 1
Byte	8-bit
Degree of Freedom	Derajat kebebasan gerak
Duty cycle	Perbandingan antara panjang sinyal on dengan panjang satu gelombang pada sinyal PWM
General register	Memori penyimpanan bit
Mikrokontroler	Single purpose processing unit yang didesain untuk melakukan program pengendalian kecil, kadang real time
Most Significant Bit	Bit dengan nilai pangkat tertinggi dalam satuan bit
PWM	Tipe gelombang segi empat yang dimodulasi terhadap lebar sinyal



DAFTAR LAMPIRAN

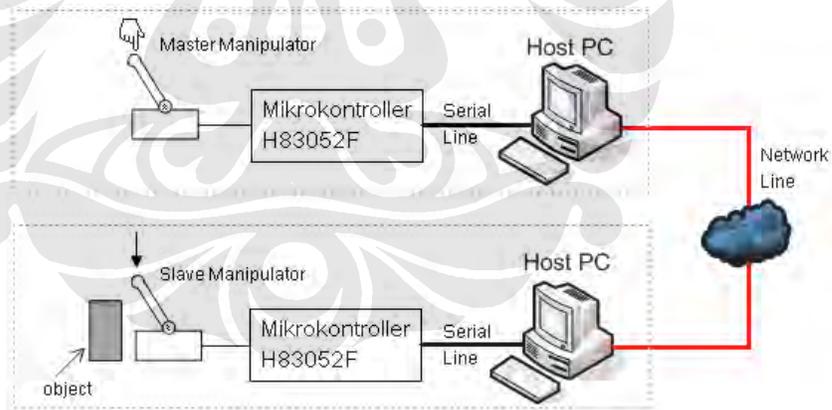
Lampiran I. Rangkaian Hasil Perakitan Perangkat Keras.....	56
--	----



BAB 1 PENDAHULUAN

1.1. Latar Belakang

Pada saat ini sistem telorobotik banyak dikembangkan untuk melakukan pekerjaan yang berbahaya dan tidak dapat diakses oleh operator secara langsung [1]. Pada pengendali telorobotik, umumnya terdapat dua buah manipulator yaitu *master* dan *slave* yang terkoneksi satu sama lain melalui suatu jaringan. Pada sistem telorobotik biasa, gerakan yang dibuat oleh operator pada manipulator master akan mengirimkan sinyal informasi yang cukup untuk membuat manipulator pada sisi slave mengikuti gerakan manipulator *master*. Suatu sistem telorobotik dapat dikategorikan sebagai bilateral robot apabila terdapat pertukaran sinyal informasi sebagai masukan pengendali dari *master* ke *slave* dan sebaliknya sehingga memungkinkan operator pada sisi *master* untuk mengendalikan manipulator *slave* dan merasakan objek yang dirasakan/ disentuh oleh manipulator pada sisi slave. Gambar 1.1 menggambarkan ilustrasi sistem yang dirancang.



Gambar 1.1. Ilustrasi Sistem Bilateral Kontrol yang Dirancang

Manipulator robot yang dibuat berupa lengan dengan satu sendi (degree of freedom/ d.o.f.) yang masing-masing digerakkan oleh sebuah motor dengan sensor putaran.

Dalam membangun suatu sistem bilateral kontrol yang dimaksud, selain sensor dan aktuator berupa motor dc servo, juga dibutuhkan suatu unit data

akuisisi dan unit pengendali. Unit data akuisisi (misalnya DAQ PCI card) merupakan suatu perangkat yang digunakan sebagai penghubung untuk melakukan fungsi konversi A/D, D/A, fungsi *counter*, atau fungsi I/O. Adapun untuk pembuatan aplikasi bilateral kontrol pada skripsi ini digunakan mikrokontroler H8/3052F sebagai pengganti DAQ PCI card untuk menekan biaya perangkat.

1.2. Perumusan Masalah

Permasalahan yang akan diselesaikan mencakup pembangunan suatu sistem bilateral kontrol berbasis PC dengan menggunakan mikrokontroler sebagai unit data akuisisi pengganti DAQ PCI card. Sistem yang dibuat harus menjamin pergerakan lengan *slave* mengikuti perubahan pergerakan manipulator lengan *master*. Selain itu, operator pada sisi *master* harus dapat merasakan objek yang menahan pergerakan manipulator *slave*.

1.3. Tujuan Penelitian

Tujuan penelitian yang dilakukan pada skripsi ini adalah untuk merancang suatu sistem bilateral kontrol berbasis PC dengan menggunakan mikrokontroler H8/3052F sebagai unit data akuisisi dan melakukan evaluasi terhadap sistem bilateral kontrol yang dibangun.

1.4. Pembatasan Masalah

Pekerjaan yang didefinisikan pada skripsi ini dibatasi pada perancangan sistem bilateral kontrol untuk lengan dengan sebuah sendi dan evaluasi kerja pada sistem bilateral kontrol yang dibuat terhadap beberapa perlakuan yang diberikan. Sudut pergerakan sendi yang dibuat juga dibatasi dari 0° hingga 150°.

1.5. Metodologi Penelitian

Penelitian yang didefinisikan pada skripsi ini dilakukan melalui:

1. Pendekatan tinjauan pustaka yaitu melakukan studi literatur dari buku-buku pustaka, sumber-sumber di Internet, dan buku manual dari perangkat yang digunakan.

2. Pendekatan diskusi dengan pembimbing skripsi yang berkaitan dengan topik bahasan skripsi.
3. Perancangan dan pembuatan perangkat keras.
4. Pembuatan program pada komputer dan mikrokontroler untuk melakukan pengujian terhadap batasan masalah yang dibahas pada skripsi ini.

1.6. Sistematika Penelitian

Hasil kerja pada penelitian ini akan dievaluasi kualitas kerja sistem, dan sejauh mana sistem mampu melakukan fungsinya sebagai aplikasi bilateral kontrol. Data yang diambil berupa sudut-sudut aktuator *master* dan *slave* untuk berbagai perlakuan yang diberikan pada sistem bilateral kontrol yang dibuat.

1.7. Sistematika Penulisan

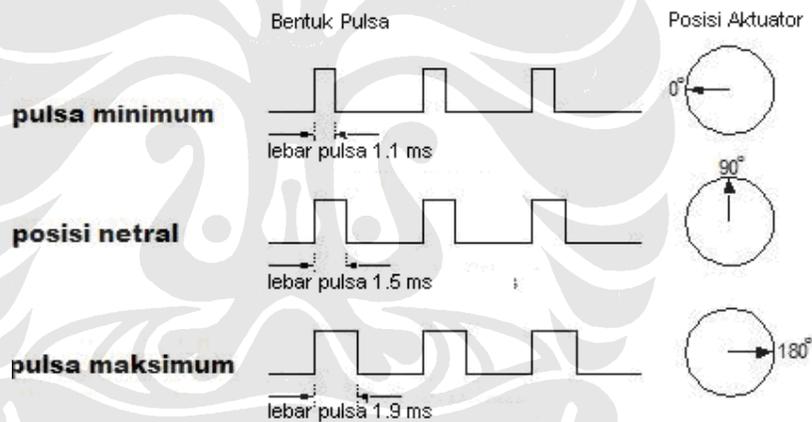
Guna membuat pembahasan masalah pada skripsi menjadi lebih sistematis, skripsi ini dibagi menjadi beberapa bab.

Bab Pertama, Pendahuluan, memuat latar belakang permasalahan, tujuan, pembatasan masalah, metodologi penelitian, sistematika penelitian, dan sistematika penulisan. **Bab Kedua**, Landasan Teori, menggambarkan landasan pengetahuan mengenai komunikasi serial, mikrokontroler, dan hal-hal lain yang digunakan pada skripsi ini. **Bab Ketiga**, Perancangan, menggambarkan langkah pengerjaan skripsi ini mulai dari perancangan sistem hingga pengaturan konfigurasi yang digunakan. **Bab Keempat**, Analisa, berisi analisa dan pembahasan mengenai evaluasi kerja dengan berbagai aspek yang menjadi fokus penelitian. **Bab Kelima**, Penutup, bab ini merangkum pembahasan skripsi ini.

BAB 2 LANDASAN TEORI

2.1. Motor DC Servo

Aktuator yang akan dikendalikan pada skripsi ini berupa motor dc servo Hitech HSR 5995TG. Sebuah motor dc servo terdiri dari sebuah motor dc beserta pengendali internal, sehingga pengendali posisi putaran cukup diatur dengan masukan sinyal. Sebuah motor dc servo umumnya dilengkapi tiga buah kabel yang harus dihubungkan yaitu kabel sumber tegangan (VCC), kabel ground (GND), dan kabel sinyal/ pulsa. Sinyal yang digunakan untuk mengatur/mengendalikan sudut putaran aktuator berupa sinyal *Pulse Width Modulation* (PWM). Lebar pulsa PWM ini menentukan besar kecilnya sudut yang akan dituju suatu aktuator, seperti yang diilustrasikan pada gambar 2.1.



Gambar 2.1. Pengaturan Sudut Motor dengan Sinyal PWM

2.2. Potensiometer Sebagai Sensor Putaran

Pada skripsi ini potensiometer digunakan sebagai sensor putaran. Pada prinsipnya, potensiometer merupakan suatu resistor variabel yang besar hambatan antara pin keluaran dengan pin *ground* dapat diubah-ubah berdasarkan putaran yang diberikan pada pemutarnya. Jika pada antara pin masukan dan pin *ground* diberikan beda potensial yang tetap, maka tegangan antara pin keluaran dan *ground* dapat berubah-ubah sesuai dengan sudut putaran yang diberikan pada potensiometer. Tegangan keluaran inilah yang digunakan untuk mengidentifikasi posisi sudut putar pada aktuator yang digunakan. Secara teori, besar perubahan

hambatan antara pin keluaran dan pin *ground* sebanding terhadap perubahan sudut putar yang diberikan terhadap potensiometer.

Hubungan tegangan, arus, dan hambatan listrik ditunjukkan pada persamaan 2.1

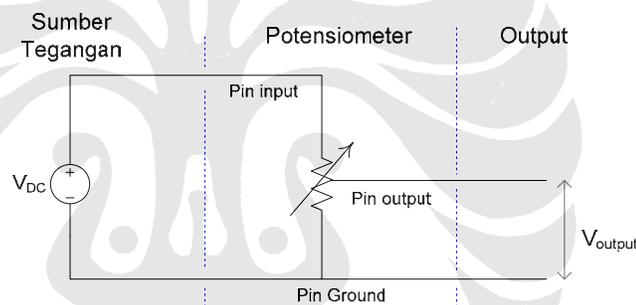
$$V = I R \quad (2.1)$$

Keterangan: V = Tegangan

I = Arus

R = Hambatan Listrik

Dengan besar arus yang konstan, maka perubahan tegangan akan linier terhadap hambatan listrik, sehingga perubahan sudut putar pada potensiometer akan menyebabkan perubahan tegangan keluaran potensiometer secara linier. Gambar 2.2 menunjukkan ilustrasi penggunaan potensiometer sebagai sensor putaran.



Gambar 2.2 Penggunaan Potensiometer sebagai Sensor Putaran

2.3. Spesifikasi Mikrokontroler H8/3052F [2]

Mikrokontroler H8/3052F merupakan salah satu mikrokontroler yang memiliki fitur yang cukup lengkap untuk menggantikan akuisisi data PCI card pada aplikasi yang sederhana. Beberapa fitur yang dimiliki oleh mikrokontroler H8/3052F sebagai pengganti DAQ PCI card diantaranya:

1. Fitur CPU
 - a. Terdapat 16 buah general register 16-bit (dapat digunakan sebagai 8 buah general register 32-bit).
 - b. Beroperasi dengan kecepatan tinggi yaitu pewaktuan/ clock maksimum 25 MHz, waktu operasi penjumlahan dan pengurangan sekitar 80ns,

waktu operasi pengalihan dan pembagian sekitar 560ns, serta alokasi pengalamatan 16-Mbyte.

- c. Memiliki instruksi yang cukup lengkap diantaranya instruksi logika, aritmatika, transfer data, manipulasi bit, perkalian dan pembagian dengan dan tanpa tanda (*signed* dan *unsigned*).
2. Fitur Memori yaitu besar memori flash sebesar 512 *kbytes* dan RAM sebesar 8 *kbytes*
3. Fitur Unit Pewaktu Terintegrasi (ITU) 16-bit
 - a. 5 kanal timer 16-bit, yang dapat memproses sampai 12 pulsa output atau 10 pulsa input
 - b. Timer counter 16-bit pada setiap kanal (kanal 0 – 4)
 - c. Operasi dapat disinkronisasi
 - d. Mode PWM dapat beroperasi pada tiap kanal
 - e. Mode *Phase counting* dapat beroperasi pada kanal 2
 - f. Buffering pada kanal 3 dan 4
 - g. Mode *Reset Synchronized* PWM pada kanal 3 dan 4
 - h. Mode *Complementary* PWM pada kanal 3 dan 4
4. Fitur Watch Dog Timer (WDT)
5. Fitur Interface Komunikasi Serial (SCI)
 - a. Dua kanal independen (SCI0 dan SCI1)
 - b. *Full Duplex*, yaitu dapat melakukan transmisi dan penerimaan secara bersamaan
 - c. Mode Asinkron dan Sinkron
6. Fitur konversi A/D, dengan resolusi 10-bit, 8 buah kanal, mode *single* dan *scan*, range konversi tegangan analog yang beragam, dan fungsi *sampling* dan *hold*.
7. Fitur konversi D/A, dengan resolusi 8-bit, dan 2 buah kanal
8. Fitur I/O, yaitu 70 pin yang dapat berfungsi sebagai pin masukan atau keluaran, dan 9 pin yang berfungsi sebagai pin masukan saja.

2.4. Konverter A/D pada Mikrokontroler H8/3052F [2]

ADC (*Analog to Digital Converter*) adalah alat yang digunakan untuk mengkonversi besaran tegangan analog ke bentuk digital.

2.4.1. Overview ADC pada Mikrokontroler H8/3052F

ADC pada mikrokontroler H8/3052F memiliki banyak fitur yang dapat digunakan diantaranya resolusi 10-bit, 8 buah kanal input analog, jangkauan tegangan yang dapat dipilih melalui tegangan pada pin V_{REF} , proses konversi yang cepat yaitu sekitar 5,4 μ s maksimum tiap kanal (dengan pewaktuan 25 MHz).

2.4.2. Konfigurasi dan Deskripsi Register ADC Mikrokontroler H8/3052F

Pada operasi *Analog to Digital Converter* (ADC), terdapat 3 jenis register yang digunakan diantaranya Register Data A/D (ADDRA – ADDRD), Register Kontrol/ Status A/D (ADCSR), dan Register Kontrol A/D (ADCSR).

2.4.2.1. Register Data A/D (ADDRA – ADDRD)

Pada mikrokontroler H8/3052F, terdapat 4 buah register data A/D. Register ini merupakan register yang menyimpan hasil konversi dari besaran tegangan analog ke dalam bentuk digital. Struktur dari register data A/D dapat dilihat pada gambar 2.3.

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	-	-	-	-	-	-
Nilai inisial	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
	10 Bit Data Hasil Konversi A/D										Reserved Bit					

* n = A - D

Gambar 2.3. Struktur dari Register Data A/D

Hasil konversi tegangan analog ke dalam bentuk digital adalah 10-bit data yang disimpan pada register data A/D sesuai dengan kanal yang dipilih. Bit 5 – 0 dari register data adalah bit yang nilainya tidak dapat digunakan dan selalu dibaca 0 oleh CPU.

2.4.2.2. Register Kontrol/ Status A/D (ADCSR)

ADCSR merupakan register 8-bit yang digunakan untuk memilih mode dan mengontrol proses konversi A/D. Struktur ADCSR ditunjukkan oleh gambar 2.4.

Bit	7	6	5	4	3	2	1	0
ADCSR	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	R/(W)*	R/W	R/W	R/W	R/W	R/W	R/W	R/W

* Hanya 0 yang dapat ditulis untuk clear flag

Gambar 2.4. Struktur Register Kontrol/ Status A/D (ADCSR)

Konfigurasi dan penjelasan dari tiap bit pada struktur register ADCSR untuk pengaplikasiannya terhadap alat yang akan dibangun diberikan oleh:

1. Bit 7 – ADF (A/D End Flag), merupakan flag yang digunakan untuk mengindikasikan berakhirnya proses konversi analog ke bentuk digital. Untuk melakukan proses konversi, maka bit ini harus bernilai 0.

Tabel 2.1. Pengaruh Isi Bit ADF Pada Proses Konversi A/D

Bit 7 - ADF	Deskripsi
0	(Kondisi Clear) Nilai 0 akan ditulis saat membaca ADF = 1
1	(Kondisi Setting) Mode Single: Proses konversi A/D berhenti Mode Scan: Proses konversi pada tiap kanal dihentikan

2. Bit 6 – ADIE (A/D Interrupt Enable), merupakan bit yang mengatur ada atau tidaknya permintaan interupsi pada akhir konversi A/D.

Tabel 2.2. Pengaruh Isi Bit ADIE Pada Proses Konversi A/D

Bit 6 - ADIE	Deskripsi
0	Permintaan interupsi berakhirnya A/D (ADI) dimatikan
1	Permintaan interupsi berakhirnya A/D (ADI) diaktifkan

3. Bit 5 – ADST (A/D Start), merupakan bit yang mengatur proses mulai dan berhentinya suatu konversi A/D. bit ADST harus diset tetap menjadi 1 saat

proses konversi berlangsung. Bit ini juga dapat diset menjadi 1 melalui masukan trigger eksternal pada pin \overline{ADTRG} .

Tabel 2.3. Pengaruh Isi Bit ADST Pada Proses Konversi A/D

Bit 5 - ADST	Deskripsi
0	Proses konversi A/D dihentikan
1	Mode Single: Proses Konversi dimulai, Bit ini akan secara otomatis di-clear menjadi 0 ketika proses konversi berakhir Mode Scan: Proses konversi dimulai dan berlanjut, bersiklus diantara kanal-kanal yang dipilih, hingga ADST di-clear menjadi 0 oleh perangkat lunak, reset, atau transisi ke mode standby

4. Bit 4 – SCAN (Mode Scan), yaitu bit yang digunakan untuk memilih mode *single* atau *scan*. Mode single digunakan ketika mikrokontroler hanya memerlukan sebuah proses konversi A/D. Sedangkan mode scan digunakan ketika mikrokontroler memerlukan lebih dari satu proses konversi A/D

Tabel 2.4. Pemilihan Mode Konversi Melalui Bit SCAN

Bit 4 - SCAN	Deskripsi
0	Mode single
1	Mode scan

5. Bit 3 – CKS (Clock Select), merupakan bit yang digunakan untuk menentukan waktu konversi A/D. Untuk mengubah waktu konversi yang digunakan, maka perlu menghentikan proses konversi dahulu dengan meng-clear nilai ADST menjadi 0.

Tabel 2.5. Pemilihan Waktu Konversi Melalui Bit CKS

Bit 3 - CKS	Deskripsi
0	Waktu konversi = 266 state (maksimum)
1	Waktu konversi = 134 state (maksimum)

6. Bit 2 to 0 – CH2 to CH0 (Channel Select 2 to 0), merupakan bit yang digunakan untuk memilih kanal masukan analog.

Tabel 2.6. Pemilihan Kanal Konversi Melalui Bit-bit CH

Pemilihan Grup	Pemilihan kanal		Deskripsi	
	CH1	CH0	Mode Single	Mode Scan
0	0	0	AN ₀	AN ₀
		1	AN ₁	AN ₀ , AN ₁
	1	0	AN ₂	AN ₀ – AN ₂
		1	AN ₃	AN ₀ – AN ₃
1	0	0	AN ₄	AN ₄
		1	AN ₅	AN ₄ , AN ₅
	1	0	AN ₆	AN ₄ – AN ₆
		1	AN ₇	AN ₄ – AN ₇

2.4.2.3. Register Kontrol A/D (ADCR)

ADCR merupakan register 8-bit yang berfungsi untuk mengaktifkan atau mematikan trigger eksternal dari konversi A/D. Struktur dari register ini ditunjukkan oleh gambar 2.5.

Bit	7	6	5	4	3	2	1	0
ADCR	TRGE	-	-	-	-	-	-	-
Nilai inisial	0	1	1	1	1	1	1	0
R/W	R/(W)	-	-	-	-	-	-	-

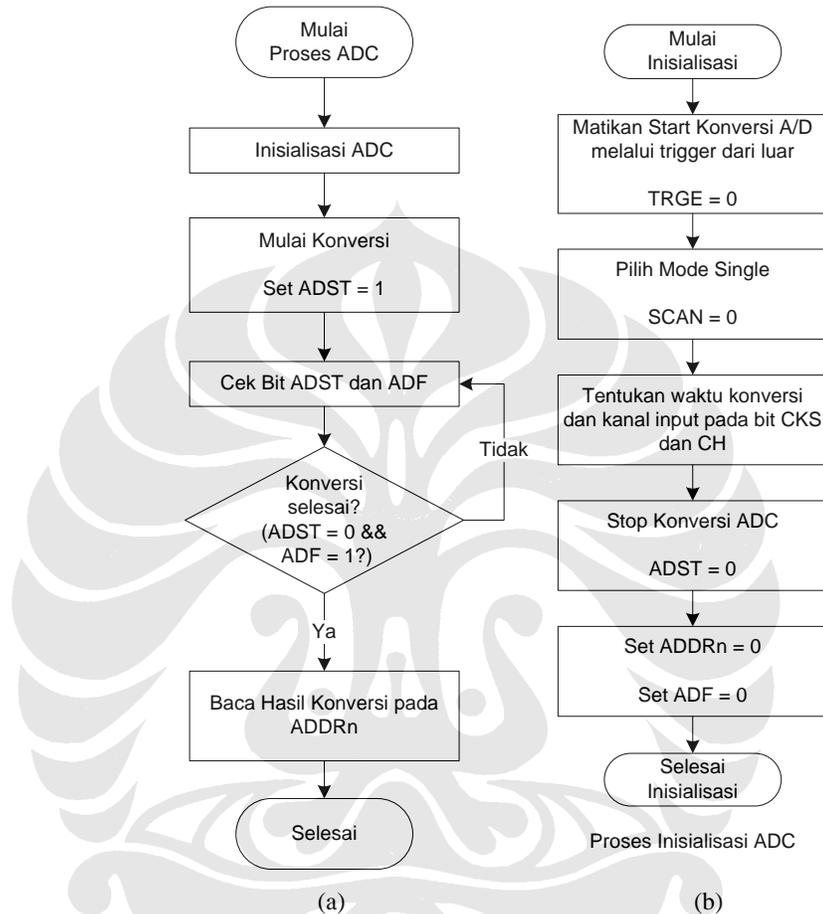
Gambar 2.5. Struktur Register Kontrol A/D (ADCR)

Fungsi dari ADCR hanya diatur oleh bit 7 – TRGE (Trigger Enable). Jika bit ini diset menjadi 0, maka proses konversi A/D tidak dapat ditrigger dari luar. Namun, jika bit ini bernilai 1, maka proses konversi A/D akan dimulai pada saat sinyal trigger eksternal \overline{ADTRG} berubah dari 1 ke 0.

2.4.3. Operasi ADC pada Mikrokontroler H8/3052F dengan Mode Single

Operasi ADC dengan mode single dapat dipilih ketika hanya dibutuhkan sebuah proses konversi A/D pada sebuah kanal. Untuk menggunakan mode ini maka bit SCAN pada ADCSR harus diset menjadi 0. Proses konversi ini dimulai ketika bit ADST diset menjadi 1 yang dapat dilakukan oleh perangkat lunak atau trigger dari luar. Bit ADST akan tetap bernilai 1 ketika proses konversi berlangsung dan secara otomatis akan di-clear menjadi 0 ketika proses konversi berakhir.

Setelah proses konversi berakhir bit ADF juga akan diset menjadi 1. Setelah bit ADST menjadi 0 dan ADF menjadi 1, maka hasil konversi A/D dapat dibaca pada register ADDRA – ADDRD. Gambar 2.6 menunjukkan diagram alir dari inialisasi ADC dan operasi ADC pada mode *single*.



Gambar 2.6. Diagram Alir Operasi ADC pada Mode Single (a); dan Proses Inialisasi ADC (b)

2.5. Unit Pewaktuan Terintegrasi pada Mikrokontroler H8/3052F [2]

Mikrokontroler H8/3052F memiliki lima buah unit pewaktuan (ITU) yang mampu memproses hingga 12 pulsa keluaran atau 10 pulsa masukan. Masing-masing unit pewaktuan ini memiliki dua buah register *general* (GRA dan GRB) yang dapat berfungsi sebagai *output compare match* atau *input compare match*. Mode unit pewaktuan yang dapat dipilih pada mikrokontroler ini diantaranya:

- Mode Normal, yaitu sebagai *counter up* untuk beroperasi sebagai *free-running counter* dan *counter periodik*.

- Mode Sinkron, yaitu operasi *counter* up yang bekerja secara sinkron pada beberapa kanal pewaktuan.
- Mode PWM, yaitu menghasilkan keluaran sinyal PWM pada pin TIOCA.
- Mode *Reset-Synchronized* PWM
- Mode *Complementary* PWM
- Mode *Phase-Counting*

Mode yang digunakan pada skripsi ini adalah mode PWM untuk menghasilkan sinyal PWM yang menggerakkan motor dc servo.

2.5.1. Register – register ITU

Register-register pada ITU berfungsi untuk pemilihan mode operasi, pengaturan perhitungan pewaktuan, dan pengontrolan terhadap lebar sinyal PWM yang dihasilkan.

2.5.1.1. Register Mulai Pewaktuan (TSTR)

TSTR (*Timer Start Register*) merupakan register 8-bit yang digunakan untuk memulai dan menghentikan perhitungan terhadap pewaktuan *counter* (TCNT). Struktur dari TSTR ditunjukkan pada gambar 2.7.

Bit	7	6	5	4	3	2	1	0
TSTR	-	-	-	STR4	STR3	STR2	STR1	STR0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W

Gambar 2.7. Struktur TSTR

2.5.1.2. Register Mode Pewaktuan (TMDR)

TMDR (*Timer Mode Register*) merupakan register 8-bit yang digunakan untuk memilih mode operasi pada ITU mikrokontroler H8/3052F. Struktur dari TMDR ditunjukkan gambar 2.8.

Bit	7	6	5	4	3	2	1	0
TMDR	-	MDF	FDIR	PWM4	PWM3	PWM2	PWM1	PWM0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Gambar 2.8. Struktur TMDR

Penjelasan dari bit-bit pada register mode pewaktuan yaitu:

1. Bit 6 – MDF, digunakan untuk memilih mode operasi *phase counting* pada kanal 2.
2. Bit 5 – FDIR, digunakan untuk pengaturan kondisi *flag overflow* (OVF) pada TSR2.
3. Bit 4 sampai 0 – PWMn, digunakan untuk pemilihan mode PWM pada kanal n.

2.5.1.3. Register Timer Counter (TCNT)

TCNT (*Timer Counter*) merupakan register 16 bit yang menghitung masukan pulsa dari sumber pewaktuannya.

2.5.1.4. Register General (GRA, GRB)

Register General adalah register 16 bit yang dapat berfungsi sebagai register *output compare* atau *input capture*. ITU mempunyai 10 general register, dua di setiap kanalnya.

Ketika berfungsi sebagai register *output compare*, isi register ini akan dibandingkan dengan nilai TCNT. Ketika keduanya cocok, *flag* IMFA atau IMFB menjadi 1 pada TSR.

2.5.1.5. Register Pengontrol Pewaktuan (TCR)

TCR (*Timer Control Register*) adalah register 8 bit yang memilih sumber pewaktuan, memilih dan memilih cara *counter* di-clear-kan. Tabel 2.7 merangkum fungsi dan struktur TCR.

Tabel 2.7. Fungsi dan Struktur TCR

Bit	Nama	Singkatan	R/W	Fungsi
0	Timer prescaler 0	TPSC0	R/W	memilih pewaktuan counter
1	Timer prescaler 1	TPSC1	R/W	
2	Timer prescaler 2	TPSC2	R/W	
3	Clock Edge 0	CKEG0	R/W	memilih clock edge eksternal
4	Clock Edge 1	CKEG1	R/W	
5	Counter clear 0	CCLR0	R/W	memilih sumber pembersihan counter
6	Counter clear 1	CCLR1	R/W	

2.5.1.6. Timer I/O Control Register (TIOR)

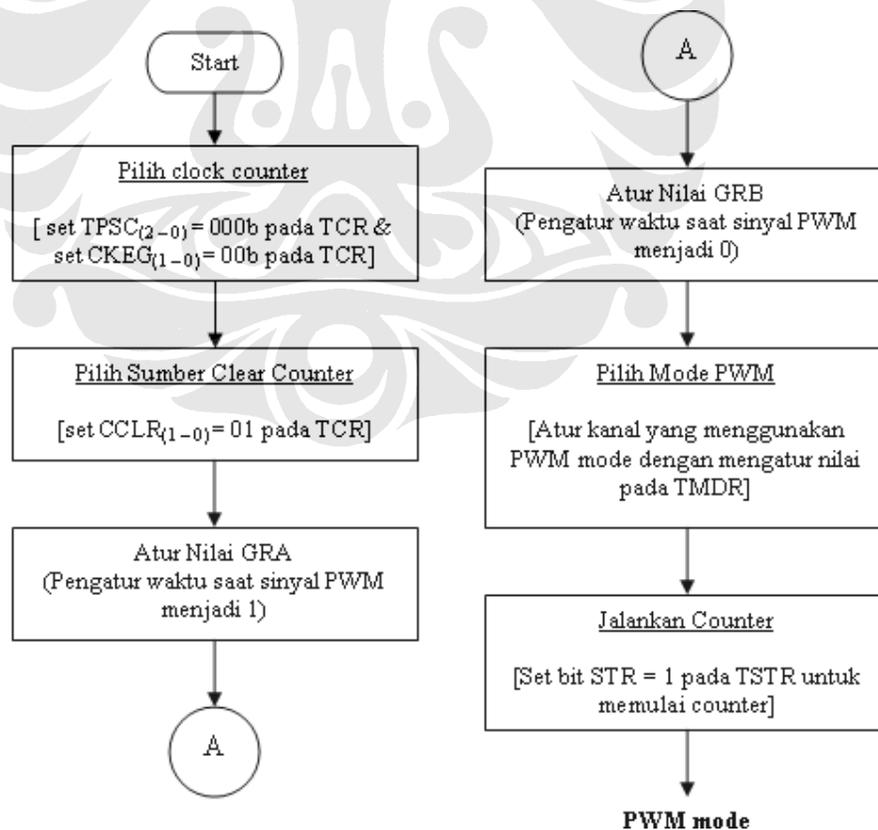
TIOR adalah register baca/tulis 8 bit yang menentukan fungsi GRA dan GRB sebagai *output compare match* atau *input capture match*. Tabel 2.8 merangkum fungsi dan struktur TIOR.

Tabel 2.8 Fungsi dan Register TIOR

Bit	Nama	Singkatan	R/W	Fungsi
0	I/O control A0	IOA0	R/W	memilih fungsi GRA
1	I/O control A1	IOA1	R/W	
2	I/O control A2	IOA2	R/W	
4	I/O control B0	IOB0	R/W	memilih fungsi GRB
5	I/O control B1	IOB1	R/W	
6	I/O control B2	IOB2	R/W	

2.5.2. Operasi ITU pada Mode PWM

Pada mode PWM, nilai GRA dan GRB digunakan untuk menghasilkan sinyal PWM pada pin TIOCA. Diagram alir operasi ITU pada mode PWM ditunjukkan pada gambar 2.9.



Gambar 2.9. Diagram Alir Operasi ITU pada Mode PWM

2.6 Komunikasi Serial pada Mikrokontroler H8/3052F [2]

Komunikasi serial pada mikrokontroler H8/3052F digunakan untuk pengiriman data dari mikrokontroler ke PC dan sebaliknya. Komunikasi serial (SCI) pada mikrokontroler H8/3052F dapat berkomunikasi dengan mode sinkron dan asinkron.

2.6.1. Pin dan Register SCI

Komunikasi serial (SCI) pada mikrokontroler H8/3052F memiliki konfigurasi pin untuk tiap-tiap kanal seperti yang ditunjukkan tabel 2.9.

Tabel 2.9 Pin-pin SCI

Kanal	Nama	Singkatan	I/O	Fungsi
0	Pin Serial Clock	SCK ₀	I/O	Clock I/O SCI ₀
	Pin Penerimaan Data	RxD ₀	I	Penerimaan data input SCI ₀
	Pin Pengiriman Data	TxD ₀	O	Pengiriman data output SCI ₀
1	Pin Serial Clock	SCK ₁	I/O	Clock I/O SCI ₁
	Pin Penerimaan Data	RxD ₁	I	Penerimaan data input SCI ₁
	Pin Pengiriman Data	TxD ₁	O	Pengiriman data output SCI ₁

Register-register internal pada SCI berfungsi dalam pemilihan mode, format data, bit rate, dan pengontrolan bagian pengiriman dan penerimaan.

2.6.1.1. Shift Register Penerima (RSR)

RSR merupakan register 8-bit yang berperan dalam penerimaan data serial. Cara kerjanya adalah dengan memasukkan masukan data serial pada pin RxD ke dalam RSR dengan urutan penerimaan LSB (bit 0) terlebih dahulu lalu mengkonversinya ke bentuk data paralel.

2.6.1.2. Register Data Penerima (RDR)

RDR berfungsi sebagai register 8-bit yang menyimpan data serial yang diterima. Ketika SCI selesai memperoleh 1 byte data serial, maka secara otomatis SCI akan memindahkan data tersebut ke RDR untuk disimpan. Setelah itu, maka RSR akan siap untuk menerima data berikutnya. Hal ini memungkinkan penerimaan data secara kontinu.

2.6.1.3. Shift Register Pengirim (TSR)

TSR merupakan register 8-bit yang berperan dalam pengiriman data serial. Cara kerja register ini adalah dengan memindahkan data dari TDR ke TSR, kemudian data akan ditransmisikan secara serial dari pin TxD dengan urutan pengiriman LSB (bit 0) terlebih dahulu. Perpindahan data dari TDR ke TSR hanya dapat dilakukan jika flag TDRE pada register SSR tidak bernilai 1.

2.6.1.4. Register Data Penerima (TDR)

TDR berfungsi sebagai register 8-bit yang menyimpan data serial untuk transmisi serial. Ketika SCI mendeteksi bahwa isi dari TSR kosong, maka secara otomatis SCI akan memindahkan data dari TDR ke TSR. Pengiriman data serial juga dapat dilakukan secara kontinu dengan cara menuliskan data ke TDR saat transmisi serial sedang dilakukan dari TSR.

2.6.1.5. Register Mode Serial (SMR)

SMR merupakan register 8-bit yang berfungsi untuk menentukan format komunikasi serial pada SCI dan memilih sumber clock untuk menghasilkan baud rate. Pada gambar 2.10 ditunjukkan struktur dari register SMR.

Bit	7	6	5	4	3	2	1	0
SMR	C/ \bar{A}	CHR	PE	O/E	STOP	MP	CKS1	CKS0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Gambar 2.10 Struktur dari Register Mode Serial (SMR)

Konfigurasi dan penjelasan dari tiap bit pada struktur register SMR untuk pengaplikasiannya terhadap alat yang akan dibangun diberikan oleh:

1. Bit 7 – C/ \bar{A} (Communication Mode), merupakan bit yang menentukan apakah SCI beroperasi pada mode asinkron (bit 0) atau mode sinkron (bit 1). Komunikasi serial yang digunakan pada aplikasi alat ini adalah asinkron sehingga diset C/ \bar{A} = 1.
2. Bit 6 – CHR (Character Length), merupakan bit yang digunakan untuk menentukan panjang tiap data serial apakah 8-bit (bit 0) atau 7-bit (bit 1).

Isi dari bit ini hanya akan berlaku jika mode yang dipilih adalah mode asinkron ($C/\bar{A} = 1$). Panjang data yang digunakan untuk transmisi adalah 8-bit sehingga diset $CHR = 0$.

3. Bit 5 – PE (Parity Enable), merupakan bit yang digunakan untuk menentukan format dari data serial pada mode asinkron menggunakan tambahan parity (bit 1) atau tidak ada parity (bit 0). Pada format data yang dikirimkan digunakan tambahan parity untuk memberi kemampuan receiver mendeteksi adanya single bit error sehingga diset $PE = 1$.
4. Bit 4 – O/\bar{E} (Parity Mode), merupakan bit yang digunakan untuk memilih mode parity: Even Parity (bit 0) atau Odd Parity (bit 1). Bit ini hanya berlaku jika berada pada mode asinkron ($C/\bar{A} = 1$) dan $PE = 1$. Adapun pada tulisan ini, format data dipilih parity genap sehingga diset $O/\bar{E} = 0$.
5. Bit 3 – STOP (Stop bit length), merupakan bit yang digunakan untuk memilih panjang bit stop pada mode asinkron yaitu sebuah bit stop (bit 0) atau 2 buah bit stop (bit 1). Adapun pada tulisan ini, format data serial dipilih 1 bit stop sehingga diset $STOP = 0$.
6. Bit 2 – MP (Multiprocessor Mode), merupakan bit yang digunakan untuk memilih fungsi multiprosesor (bit 1) atau tidak (bit 0). Pada aplikasi yang ingin dibangun tidak diperlukan multiprosesor sehingga diset $MP = 0$.
7. Bit 1 dan 0 – CKS1/0 (Clock Select 1 dan 0), merupakan bit-bit yang digunakan untuk memilih sumber clock dari penghasil baud rate pada chip yang ada. 4 buah clock yang mungkin yang ditunjukkan tabel 2.10:

Tabel 2.10 Pemilihan clock pada SCI

Bit 1: CKS1	Bit 0: CKS0	Deskripsi
0	0	Φ
	1	$\Phi/4$
1	0	$\Phi/16$
	1	$\Phi/64$

Pada komunikasi serial dipilih sumber clock ϕ dengan $\phi = 25$ MHz. Sehingga diset $CKS1 = 0$ dan $CKS0 = 0$.

2.6.1.6. Register Kontrol Serial (SCR)

SCR merupakan register 8-bit yang berperan dalam pengaktifan pengirim dan penerima SCI, clock keluaran pada mode asinkron, interupsi, dan sumber clock transmisi/penerimaan. Struktur dari register SCR ditunjukkan oleh gambar 2.11.

Bit	7	6	5	4	3	2	1	0
SCR	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Gambar 2.11. Struktur Register Kontrol Serial (SCR)

Penjelasan dan fungsi dari tiap bit dalam register kontrol serial (SCR) diberikan oleh:

1. Bit 7 – TIE (Transmit Interrupt Enable), merupakan bit yang digunakan untuk mengaktifkan (bit 1) atau mematikan (bit 0) fungsi permintaan TXI (Transmit-Data-Empty Interrupt) ketika flag TDRE pada SSR menjadi 1 saat pengiriman data serial dari TDR ke TSR.
2. Bit 6 – RIE (Receive Interrupt Enable), merupakan bit yang digunakan untuk mengaktifkan (bit 1) atau mematikan (bit 0) fungsi interupsi yang menandakan masih terdapat data pada RDR, dan interupsi yang menandakan adanya error pada penerimaan data
3. Bit 5 – TE (Transmit Enable), merupakan bit yang digunakan untuk menonaktifkan (bit 0) dan mengaktifkan (bit 1) mulainya pengiriman data serial.
4. Bit 4 – RE (Receive Enable), merupakan bit yang digunakan untuk menonaktifkan (bit 0) dan mengaktifkan (bit 1) mulainya penerimaan data serial.
5. Bit 3 – MPIE (Multiprocessor Interrupt Enable), merupakan bit yang digunakan untuk menonaktifkan (bit 0) atau mengaktifkan (bit 1) interupsi multiprosesor. Bit ini harus diset menjadi 0 pada mode non multiprosesor.
6. Bit 2 – TEIE (Transmit-End Interrupt Enable), merupakan bit yang digunakan untuk menonaktifkan (bit 0) dan mengaktifkan (bit 1)

permintaan TEI (Transmit-End Interrupt) jika TDR tidak mengandung data baru yang ingin ditransmisikan ketika MSB telah ditransmisikan.

7. Bit 1 dan 0 – CKE1/0 (Clock Enable 1 dan 0), merupakan bit-bit yang digunakan untuk memilih sumber clock dan aktif tidaknya keluaran clock dari pin SCK.

2.6.1.7. Register Status Serial (SSR)

SSR merupakan register 8-bit yang mengandung nilai bit multiprosesor, dan status flag-flag yang mengindikasikan status operasi SCI. Struktur dari Register Status Serial ditunjukkan gambar 2.12.

Bit	7	6	5	4	3	2	1	0
SSR	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Nilai inisial	0	0	0	0	0	0	0	0
R/W	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W

* Hanya 0 yang dapat ditulis untuk clear flag

Gambar 2.12. Struktur Register Status Serial (SSR)

Penjelasan dari tiap-tiap bit yang ada pada register tersebut adalah:

1. Bit 7 – TDRE (Transmit Data Register Empty), merupakan bit yang mengindikasikan bahwa SCI telah memasukkan data yang ditransmisikan dari TDR ke TSR dan data transmisi serial berikutnya dapat ditulis pada TDR.

Jika TDRE = 0 (*Clearing Condition*), artinya TDR mengandung data transmisi yang valid. Nilai ini diakibatkan program membaca ketika TDRE diset menjadi 1 kemudian menuliskan 0 pada bit tersebut, ataupun saat DMAC menulis data pada TDR. Sebaliknya jika TDRE = 1 (*Setting Conditions*), artinya TDR tidak mengandung data transmisi yang valid. Nilai ini dapat diakibatkan saat chip direset atau pada mode standby, Bit TDRE pada SCR diclear menjadi 0, ataupun isi dari TDR dimasukkan pada TSR, sehingga data baru dapat dituliskan pada TDR.

2. Bit 6 – RDRF (Receive Data Register Full), merupakan bit yang mengindikasikan bahwa RDR mengandung data penerima yang baru.

Jika $RDRF = 0$ (*Clearing Condition*) artinya RDR tidak mengandung data penerimaan yang baru. Nilai ini dapat diakibatkan oleh chip yang direset atau berada pada mode standby, program membaca RDRF ketika diset menjadi 1 kemudian menuliskan 0 pada bit tersebut, ataupun saat DMAC membaca data dari RDR. Sebaliknya jika $RDRF = 1$ (*Setting Condition*) artinya RDR mengandung data penerimaan baru. Nilai ini diakibatkan ketika data serial diterima secara normal dan ditransfer dari RSR ke RDR.

3. Bit 5 – ORER (Overrun Error), merupakan bit yang mengindikasikan penerimaan data dihentikan secara tidak normal akibat adanya error overrun.

Jika $ORER = 0$ artinya penerimaan dalam proses atau telah diakhiri secara normal. Sedangkan jika $ORER = 1$ artinya terjadi error overrun. Penerimaan data serial pada saat ini tidak dapat dilanjutkan karena RDR akan tetap menahan data sebelum terjadi error overrun selama $ORER = 1$.

4. Bit 4 – FER (Framing Error), yaitu bit yang mengindikasikan penerimaan data berakhir secara tidak normal akibat error framing pada mode asinkron.

Jika $FER = 0$ artinya penerimaan dalam progress atau telah diakhiri secara normal. Sedangkan jika $FER = 1$ artinya terjadi error framing dimana bit stop tidak bernilai 1. Pada pengecekan error framing, jika panjang bit stop adalah 2-bit, maka hanya bit pertama yang diperiksa.

5. Bit 3 – PER (Parity Error), yaitu bit yang mengindikasikan penerimaan data berakhir secara tidak normal akibat adanya error parity pada mode asinkron.

Jika $PER = 0$ artinya penerimaan dalam progress atau telah diakhiri secara normal. Sedangkan jika $PER = 1$ artinya terjadi error parity yang diakibatkan jumlah bit 1 pada data termasuk bit parity tidak sesuai dengan pengaturan parity genap atau ganjil pada bit O/\bar{E} pada register SMR.

6. Bit 2 – TEND (Transmit END), yaitu bit yang mengindikasikan bahwa ketika bit terakhir pada suatu karakter ditransmisikan TDR tidak mengandung data transmit yang baru, sehingga transmisi dihentikan.

Jika $TEND = 0$ artinya proses transmisi sedang dalam progress. Sedangkan jika $TEND = 1$ artinya proses transmisi telah berakhir.

7. Bit 1 – MPB (Multiprocessor Bit), yaitu bit yang menyimpan nilai dari bit multiprosesor pada data yang diterima ketika format multiprosesor digunakan pada mode asinkron.
8. Bit 0 – MPBT (Multiprocessor Bit Transfer), yaitu bit yang menyimpan nilai dari bit multiprosesor yang ditambahkan pada data yang ditransmisikan ketika mode multiprosesor dipilih untuk pengiriman pada mode multiprosesor.

2.6.1.8. Register Bit Rate (BRR)

BRR merupakan register 8-bit yang bersama-sama dengan bit CKS1 dan CKS0 pada SMR bertugas untuk memilih sumber clock penghasil baud rate, dan menentukan bit rate komunikasi serial. Pada tiap-tiap kanal SCI terdapat kontrol penghasil baud rate yang independen sehingga nilai yang berbeda dapat dituliskan pada 2 kanal yang berbeda. Konfigurasi dan pengaturan isi BRR pada mode asinkron dihitung berdasarkan rumus berikut:

$$N = \frac{\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (2.2)$$

Keterangan: B = Bit rate (bit/s)

N = BRR setting untuk penghasil baud rate ($0 \leq N \leq 255$)

ϕ = Frekuensi sistem clock (MHz)

n = Sumber clock penghasil baud rate (n = 0, 1, 2, 3)

Tabel 2.11. Pemilihan Nilai n berdasarkan bit CKS1 dan CKS0

n	Sumber Clock	Setting SMR	
		CKS1	CKS0
0	Φ	0	0
1	$\Phi/4$	0	1
2	$\Phi/16$	1	0
3	$\Phi/64$	1	1

Sedangkan error bit rate pada mode asinkron dapat dihitung berdasarkan persamaan berikut:

$$Error(\%) = \left\{ \frac{\phi \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100 \quad (2.3)$$

2.6.2. Operasi SCI pada Mode Asinkron pada Mikrokontroler H8/3052F

Pada komunikasi serial dengan mode asinkron setiap karakter atau frame yang dikirim atau diterima dimulai dengan sebuah bit start dan diakhiri dengan bit stop. Pada gambar 2.13 diperlihatkan format komunikasi dan pengaturannya pada SMR sesuai dengan format yang diinginkan.

Setting SMR															
Bit	CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11
	0	1	0	1	S	8-Bit Data								P	Stop

Gambar 2.13. Format Komunikasi Serial dengan 8-bit Data dengan Parity dan 1Bit Stop

Proses transmisi dan penerimaan pada SCI bersifat independen, sehingga memungkinkan komunikasi secara *full duplex* dengan perangkat lain. Selain itu baik pengiriman maupun menerima pada SCI memiliki buffer, sehingga memungkinkan proses penerimaan dan pengiriman data saat melakukan proses terhadap data lain.

Pewaktuan/ *clock* pada SCI dapat dilakukan melalui dua pilihan yaitu secara internal dan eksternal yang pemilihannya berdasarkan konfigurasi bit $\overline{C/\overline{A}}$ pada SMR dan bit CKE1 dan CKE0 pada SCR. Pewaktuan secara internal dilakukan berdasarkan pewaktuan yang dihasilkan oleh baud rate generator yang terdapat pada chip yang digunakan. Sedangkan pewaktuan secara eksternal dilakukan melalui pemberian sinyal masukan pewaktuan pada pin SCK.

Agar pengiriman dan penerimaan data serial dapat dilakukan pada SCI, maka terdapat beberapa hal yang harus dilakukan secara berurutan yaitu:

1. Inisialisasi serial pada SCI H8/3052F
2. Pengaktifan bit TE atau RE menjadi 1
3. Melakukan proses penerimaan atau pengiriman secara serial.

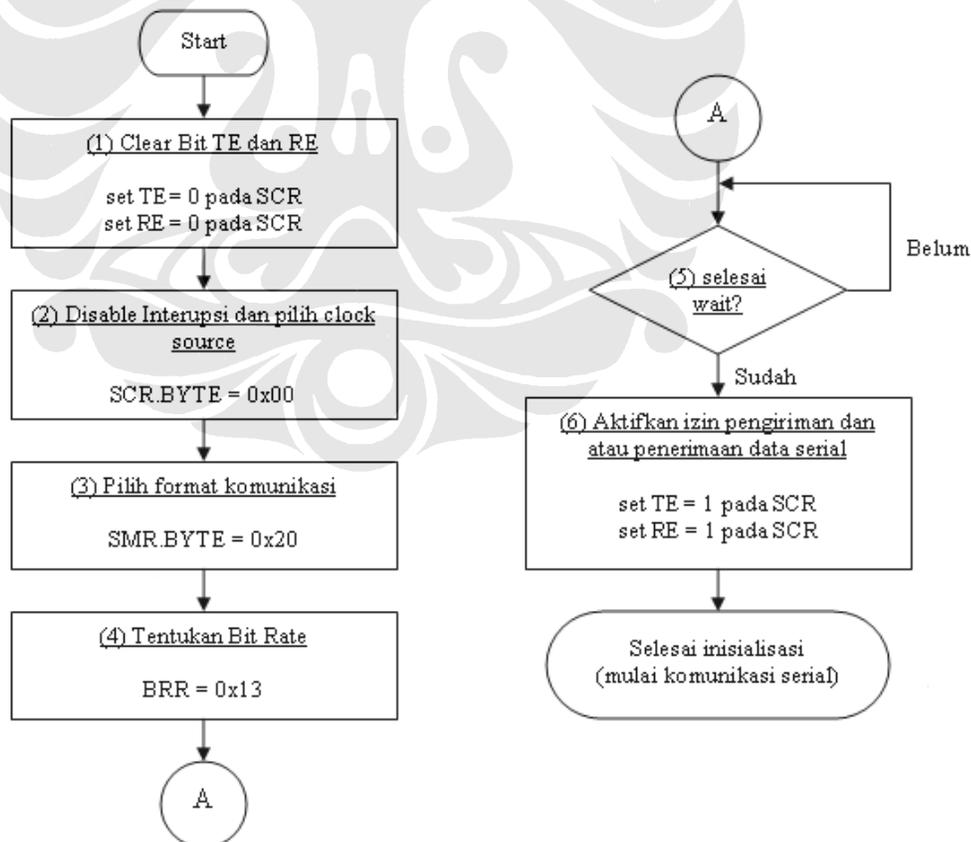
2.6.2.1. Inisialisasi Serial pada SCI

Proses inisialisasi serial pada SCI digunakan untuk menentukan format dan cara komunikasi suatu pengiriman dan penerimaan data secara serial pada SCI

dilakukan. Proses inisialisasi serial dilakukan berdasarkan urutan kerja sebagai berikut:

1. Meng-clear bit TE dan RE pada SCR menjadi 0
2. menonaktifkan semua interupsi SCI dan memilih sumber pewaktuan pada bit CKE1 dan CKE0 pada SCR.
3. Memilih format komunikasi untuk pengiriman dan atau penerimaan secara serial pada SMR
4. Menentukan bit rate untuk menentukan jumlah data yang dapat dikirim dan diterima setiap waktu (sekon) melalui BRR.
5. Menunggu paling tidak 1 interval waktu yang digunakan untuk mengirim dan menerima data.
6. Mengubah bit TE dan atau RE pada SCR menjadi 1.
7. Melakukan proses pengiriman dan atau penerimaan data serial.

Pada gambar 2.14 berikut ditunjukkan diagram alir proses inisialisasi serial pada SCI untuk operasi asinkron.



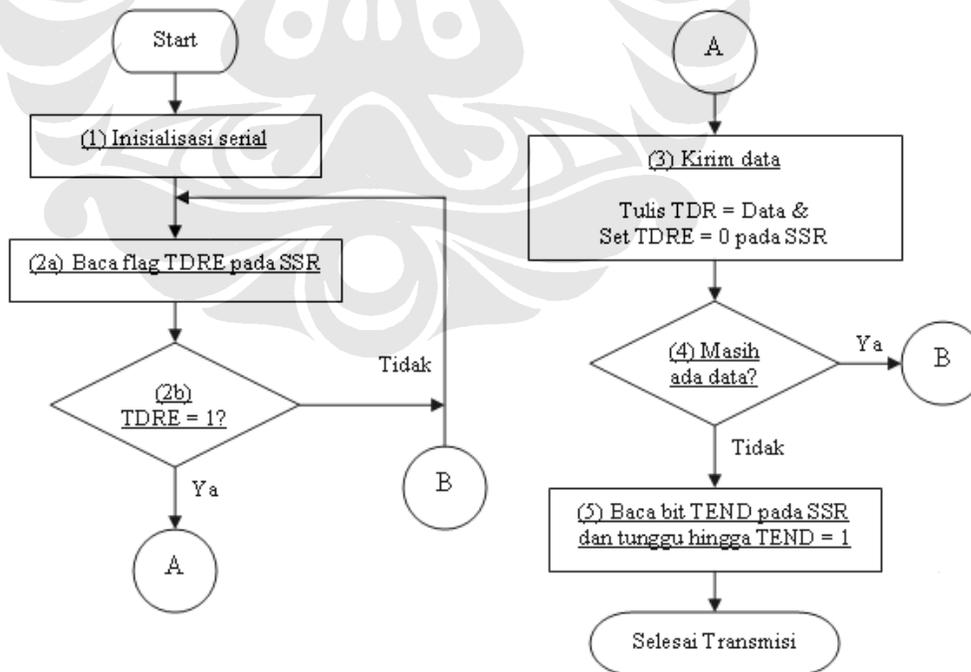
Gambar 2.14. Diagram Alir proses inisialisasi serial pada operasi asinkron

2.6.2.2. Transmisi Data Serial pada SCI

Proses transmisi data secara serial pada SCI hanya dapat dilakukan jika pada akhir proses inialisasi bit TE pada SCR diset menjadi 1 (*transmit enable*). Proses transmisi data serial dilakukan berdasarkan urutan kerja sebagai berikut:

1. Melakukan proses inialisasi serial
2. Melakukan pemeriksaan adanya data yang akan ditransmisikan pada TSR melalui flag TDRE. Flag TDRE akan bernilai 1 jika dideteksi terdapat data yang akan dikirimkan secara serial pada TSR.
3. Jika flag TDRE = 1, maka tuliskan data pada TDR dan set TDRE = 0 untuk melakukan proses transmisi data serial.
4. Proses 2 – 3 akan terus dilakukan jika masih ada data lain yang akan dikirimkan.
5. Jika sudah tidak ada data yang akan dikirimkan, maka akan dilakukan pembacaan terhadap bit TEND pada SSR secara terus-menerus sampai terbaca nilai TEND = 1 yang menandakan dihentikannya proses transmisi.

Pada gambar 2.15 ditunjukkan diagram alir proses transmisi data serial pada operasi asinkron oleh mikrokontroler H8/3052F.



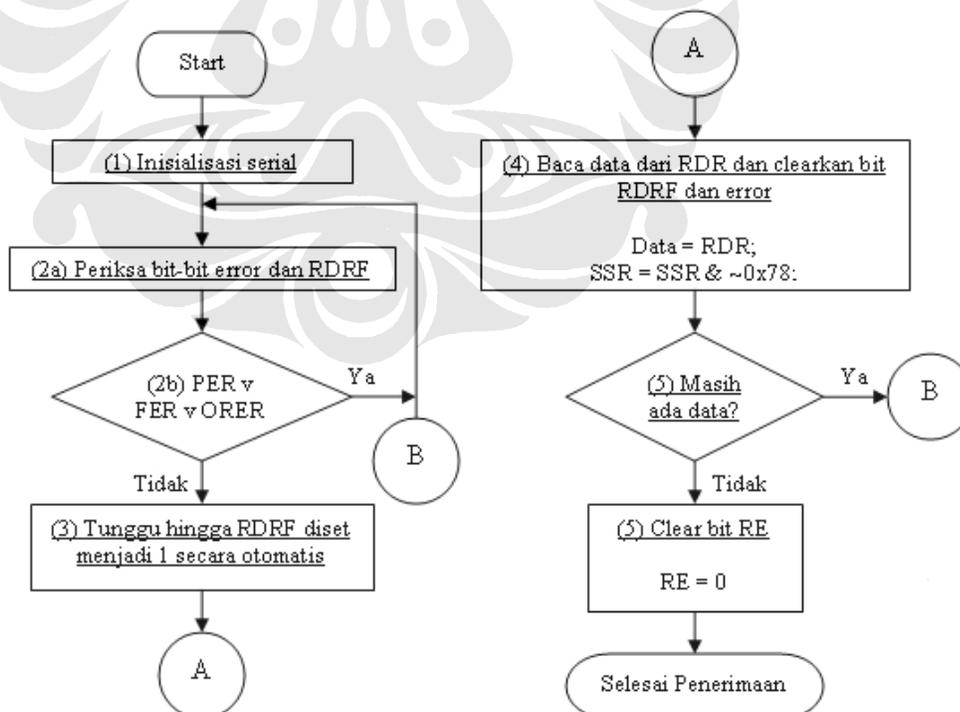
Gambar 2.15. Diagram Alir proses transmisi data serial pada operasi asinkron

2.6.2.3. Penerimaan Data Serial pada SCI

Proses penerimaan data secara serial pada SCI hanya dapat dilakukan jika pada akhir proses inialisasi bit RE pada SCR diset menjadi 1. Proses transmisi data serial dilakukan berdasarkan urutan kerja sebagai berikut:

1. Melakukan proses inialisasi serial
2. Periksa bit-bit pertanda error (parity, frame, dan overrun) dan bit RDRF yang menandakan RDR kosong atau berisi . Tunggu hingga tidak ada error dan RDR kosong (PER, FER, ORER, dan RDRF bernilai 0). Pengecekan status melalui RDRF harus bernilai 0 agar dapat dilakukan transfer data dari RSR ke RDR.
3. Jika tidak terdapat error, maka RDRF akan diset menjadi 1 secara otomatis
4. Baca data yang diterima RDR dan *clear*-kan bit-bit RDRF dan error-error yang ada sebelumnya pada SSR
5. Ulangi proses 2 – 4, jika masih ada data yang akan diterima
6. Jika proses penerimaan telah selesai, maka *clear*-kan nilai RE menjadi 0.

Pada gambar 2.16 ditunjukkan diagram alir proses penerimaan data serial pada operasi asinkron oleh mikrokontroler H8/3052F.



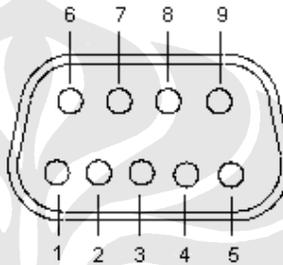
Gambar 2.16. Diagram Alir proses penerimaan data serial pada operasi asinkron

2.7. Komunikasi Serial dengan RS-232C

Untuk melakukan komunikasi serial antara PC dengan mikrokontroler H8/3052F digunakan interface RS-232C (DB-9) yang dihubungkan ke komputer dan pin-pin SCI pada mikrokontroler.

2.7.1. Konfigurasi Pin RS-232C DB-9

RS-232C DB-9 memiliki sembilan buah pin yang digunakan untuk komunikasi serial seperti yang ditunjukkan gambar 2.17.



Gambar 2.17 RS-232C DB-9 Konektor Tampak Depan

Sedangkan penjelasan pin-pin pada RS-232C ditunjukkan pada tabel 2.12.

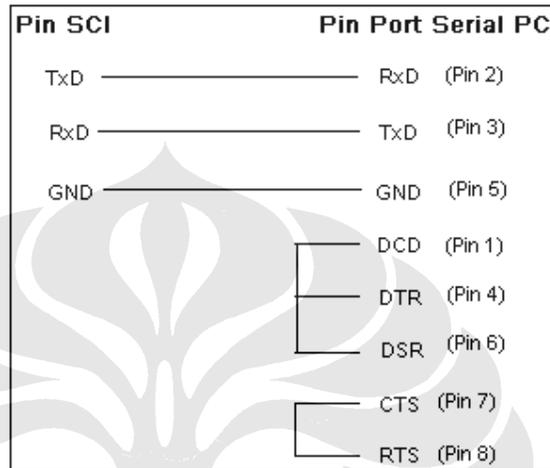
Tabel 2.12. Penjelasan Pin-pin RS-232C DB-9 Beserta Kegunaannya

Pin #	Akronim	Nama Lengkap	Arah Komunikasi	Kegunaan
1	DCD	Data Carrier Detect	<<	Indikasi adanya koneksi serial
2	RxD	Receive Data	<<	Menerima data byte
3	TxD	Transmit Data	>>	Mengirim data byte
4	DTR	Data Terminal Ready	>>	Sisi terminal siap berkomunikasi
5	GND	Ground	--	Ground
6	DSR	Data Set Ready	<<	Sisi serial yang lain siap berkomunikasi
7	RTS	Request To Send	>>	RTS/CTS flow control
8	CTS	Clear To Send	<<	RTS/CTS flow control
9	RI	Ring Indicator	<<	Adanya permintaan koneksi

2.7.2. Konfigurasi Hubungan Kabel Pin SCI dengan RS-232C DB-9

Untuk melakukan komunikasi serial dengan komputer, mikrokontroler H8/3052F hanya memerlukan 3 buah pin yaitu TxD pada SCI yang dihubungkan dengan RxD pada sisi komputer, RxD pada SCI yang dihubungkan pada TxD

pada sisi komputer, dan GND SCI yang dihubungkan dengan GND pada sisi komputer. Pada konektor RS-232C di sisi komputer selain hal tersebut, perlu dilakukan hubungan singkat antara pin 1 (DCD), 4 (DTR), dan 6 (DSR) serta antara pin 7 (RTS) dan pin 8 (CTS). Koneksi hubungan kabel RS-232C pada port serial komputer dan pin serial pada SCI ditunjukkan gambar 2.18 [3].



Gambar 2.18 Koneksi Kabel Pin Serial SCI dengan Port Serial Komputer

Tujuan pin 1, 4 dan 6 dihubung singkat adalah agar komputer selalu mendeteksi adanya data carrier pada pin 1 yang menandakan adanya perangkat lain yang terhubung padanya meskipun belum dihubungkan pada mikrokontroler. Selain itu juga bertujuan agar komputer selalu siap untuk berkomunikasi melalui port serial akibat pin 4 dan 6 yang dihubung singkat.

2.8. Pemrograman Port Serial dengan Bahasa Pemrograman C

Bahasa pemrograman yang akan digunakan untuk membuat program komunikasi serial pada komputer yaitu bahasa pemrograman C. Sedangkan sistem operasi yang digunakan pada komputer adalah sistem operasi Linux Redhat 9.0.

Pada sistem operasi yang berbasis UNIX ataupun Linux, semua perangkat baik perangkat char seperti port I/O dan perangkat blok seperti cd-rom dapat dianggap dan diperlakukan sebagai suatu file yang berada dibawah direktori /dev sehingga memudahkan pengaksesan terhadap perangkat tersebut. Karena perangkat tersebut dianggap sebagai suatu file, maka dapat digunakan perintah

open() untuk membuka akses pada perangkat yang akan diakses. Prototipe fungsi dari perintah open yaitu:

```
fd = open("Lokasi File", "Flag-flag");
```

Pada sistem operasi linux, file perangkat port serial berada pada lokasi /dev/ttyS0 dan /dev/ttyS1 yang setara dengan COM1 dan COM2 pada sistem operasi berbasis Windows.

Sedangkan untuk membaca dan menuliskan data byte dari dan ke port serial, dapat digunakan perintah read(), dan write() yang sebelumnya harus menginisialisasi port tersebut dengan mengkonfigurasi port berdasarkan bit rate, *flow control*, dan format data yang akan dikomunikasikan. Prototipe fungsi dari perintah read dan write yaitu:

```
read(fd, *received_data, n_byte);
write(fd, *transmited_data, n_byte);
```

Jika telah selesai melakukan komunikasi data antar perangkat, maka dapat digunakan perintah close() untuk menutup akses terhadap port serial pada komputer guna menghentikan komunikasi data antar kedua perangkat. Prototipe fungsi dari perintah close yaitu:

```
close(fd);
```

2.9. Dasar Pemrograman Soket UDP

Pada skripsi ini, pemrograman soket digunakan untuk melakukan komunikasi antar komputer melalui jaringan.

2.9.1. Pengenalan Terhadap Soket API

Soket merupakan perangkat komunikasi dua arah yang dapat digunakan untuk berkomunikasi dengan proses lain pada komputer yang sama atau dengan sebuah proses yang berjalan pada komputer lain.

Soket API pertama kali diperkenalkan pada sistem operasi BSD UNIX 4.2 dan merupakan standar *de facto* untuk pemrograman jaringan. Soket menyediakan kemampuan untuk membangun aplikasi komunikasi kompleks dengan perangkat jaringan lainnya. Beberapa jenis soket yang digunakan dalam pemrograman jaringan yaitu [4]:

1. Soket Datagram, yaitu soket yang digunakan dalam pengiriman paket berupa datagram. Datagram merupakan paket yang digunakan pada sistem pengiriman yang tidak menjamin hasil pengiriman data. Umumnya, soket ini lebih dikenal dengan soket UDP (*User Datagram Protocol*).
2. Soket Stream, yaitu soket yang menggunakan mekanisme pengiriman data yang memiliki level yang lebih tinggi daripada soket datagram sehingga menyediakan koneksi yang *reliable* dan menjamin pengiriman dengan benar.
3. Soket Raw, yaitu soket yang digunakan untuk melakukan akses oleh protokol level bawah, dan hanya dapat digunakan oleh program yang memiliki hak akses pada soket tersebut.
4. DAS (*Dynamically Assigned Socket*) dan SAS (*Statically Assigned Socket*), merupakan soket yang digunakan untuk pengiriman datagram pada jaringan berbasis AppleTalk.

Jenis soket yang digunakan dalam skripsi ini adalah soket datagram.

2.9.2. Pemrograman Soket UDP dengan Bahasa Pemrograman C

Untuk memulai komunikasi dengan soket, maka pada awal komunikasi perlu dilakukan pembuatan soket yang dimaksud. Untuk membuat soket UDP digunakan perintah `socket()`, contoh pembuatan soket UDP yaitu:

```
sock_fd = socket(AF_INET, SOCK_DGRAM, 0);
```

Selain membuat soket, pada sisi *server* diperlukan suatu fungsi lainnya yang digunakan untuk memberikan alamat IP dan nomor port pada soket tersebut. Fungsi ini dipenuhi dengan menggunakan perintah `bind()`. Prototipe fungsi perintah `bind()` yaitu:

```
bind(int sock_fd, struct sockaddr *my_addr, int addr_len);
```

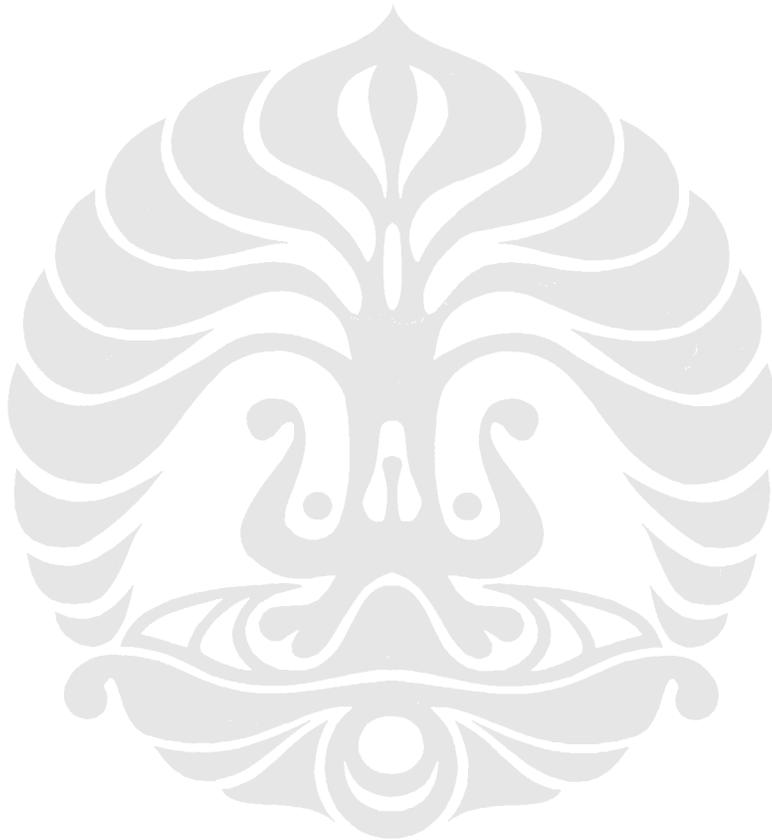
Untuk melakukan pengiriman dan penerimaan data *byte* atau *string* melalui jaringan komputer pada soket UDP digunakan perintah `sendto()` dan `recvfrom()`. Prototipe fungsi perintah `sendto()` dan `recvfrom()` yaitu:

- `sendto(int sockfd, const void *msg, int len, unsigned int flags, const struct sockaddr *to, socklen_t tolen);`

- `recvfrom(int sockfd, void *buf, int len, unsigned int flags, struct sockaddr *from, int *fromlen);`

Jika telah selesai melakukan komunikasi data antar komputer, maka dapat digunakan perintah `close()` untuk menutup komunikasi antar komputer. Prototipe fungsi dari perintah `close()` yaitu:

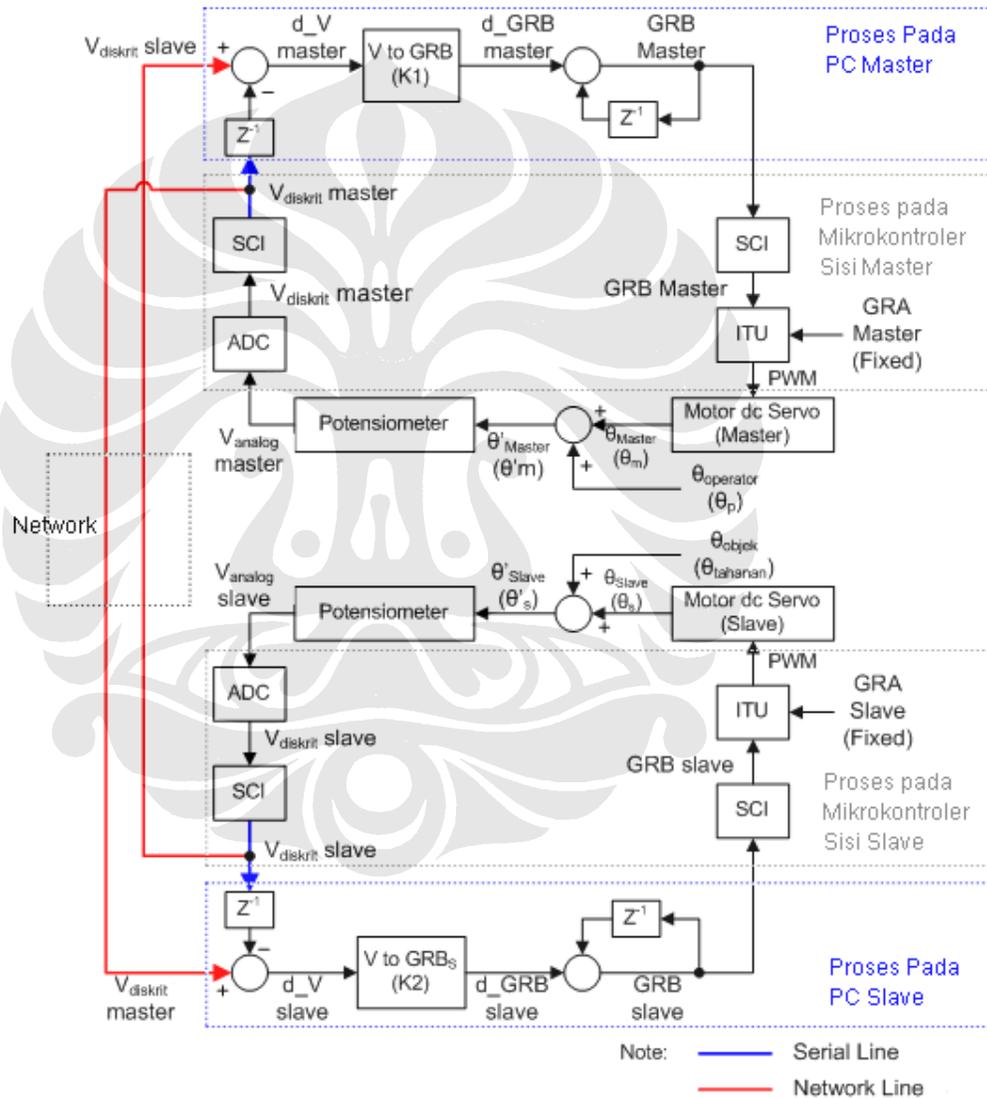
```
close(sock_fd);
```



BAB 3 PERANCANGAN

3.1. Konsep Dasar

Diagram blok konsep dasar dari sistem bilateral kontrol yang dirancang pada skripsi ini ditunjukkan gambar 3.1.



Gambar 3.1. Diagram Blok Konsep Dasar Sistem Bilateral Kontrol yang Dirancang

Perubahan sudut motor pada sisi master yang diberikan oleh operator menimbulkan perubahan keluaran tegangan pada sensor potensiometer. Tegangan

ini dikonversi menjadi bentuk digital oleh koverter A/D yang kemudian dikirim pada PC *master* melalui koneksi serial. Tegangan diskrit yang diterima PC *master* digunakan sebagai umpan balik sistem pada sisi *master* dan dikirimkan ke PC *slave* melalui jaringan komputer. Tegangan diskrit yang dikirimkan ke PC *slave* akan menjadi referensi masukan pada sistem *slave* sehingga terjadi pergerakan sudut motor *slave* mengikuti pergerakan motor *master*.

Jika terdapat objek yang menahan pergerakan lengan robot *slave*, operator dapat merasakan objek tersebut. Hal ini dikarenakan pada sistem yang dirancang, sudut motor *slave* akan dikirimkan ke PC *master* sebagai masukan referensi. Sehingga, motor *master* akan berusaha mengikuti sudut motor *slave*.

3.2. Perancangan Perangkat Keras

Perangkat keras yang digunakan untuk mendukung aplikasi bilateral kontrol yang dibuat pada skripsi ini diantaranya:

1. Dua buah mikrokontroler H8/3052F
2. Dua buah PC
3. Rangkaian filter *low pass*
4. Rangkaian *Voltage Regulator*
5. Rangkaian *optocoupler*
6. Dua buah motor, dua buah batangan aluminium, dan dua buah kabel serial.
7. Sumber tegangan

3.2.1. Penggunaan Mikrokontroler H8/3052F

Pada skripsi ini, mikrokontroler H8/3052F digunakan sebagai unit data akuisisi. Fungsi yang diaktifkan pada mikrokontroler ini yaitu fungsi konverter A/D, fungsi *counter*, dan fungsi komunikasi serial.

Fungsi konverter A/D digunakan untuk mengkonversi keluaran tegangan analog potensiometer menjadi bentuk digital. Potensiometer yang digunakan pada skripsi ini *dimounting* pada motor untuk digunakan sebagai sensor putaran.

Fungsi *counter* digunakan untuk menghasilkan sinyal PWM sesuai nilai GRB yang dikirimkan dari PC ke mikrokontroler.

Fungsi komunikasi serial digunakan untuk melakukan pengiriman dan penerimaan data antar PC dan mikrokontroler H8/3052F.

3.2.2. Spesifikasi PC

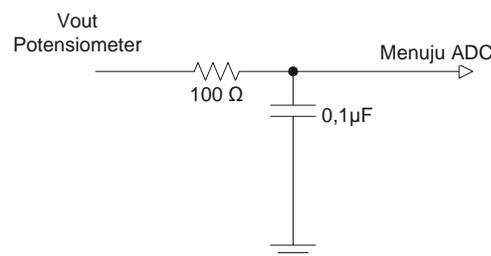
Pada skripsi ini, PC digunakan untuk mengolah data yang dikirimkan dari mikrokontroler. Data hasil olahan komputer berupa nilai GRB. Nilai GRB ini akan dikembalikan ke mikrokontroler untuk menghasilkan sinyal PWM dengan *duty cycle* tertentu. Spesifikasi minimum PC untuk menjalankan fungsinya, sistem operasi dan mengolah data, beserta spesifikasi PC yang digunakan pada skripsi ini ditunjukkan pada tabel 3.1.

Tabel 3.1. Spesifikasi PC Minimum dan Spesifikasi PC yang Digunakan pada Skripsi

	Spesifikasi Minimum	Spesifikasi yang dipakai pada Skripsi
CPU	450 MHz	3 GHz
Hard Drive	3 Gbytes	15 Gbytes
RAM	128 Mbytes	512 Mbytes
Video RAM	32 Mbytes	128 Mbytes
Keyboard	Ya	Ya
Mouse	Tidak	Ada
USB Interface	USB 1.1	USB 2.0
Port Serial	Ya (minimal 1)	Ya (1 buah)
Ethernet Interface	Ya (10Mbps)	Ya (10/100Mbps)

3.2.3. Rangkaian Filter *Low Pass*

Untuk mengurangi derau tegangan dan memperbaiki bentuk gelombang tegangan yang menuju ke konverter A/D, digunakan rangkaian filter *low pass* dengan cut off frequency 15,916kHz. Rangkaian filter *low pass* yang dirancang ditunjukkan pada gambar 3.2.

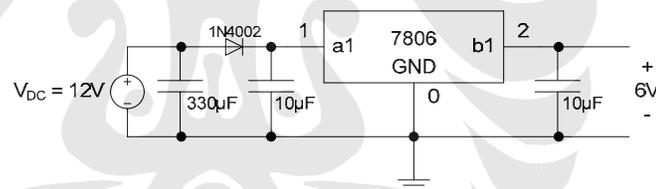


Gambar 3.2. Rancangan Rangkaian Filter Low Pass

3.2.4. Rangkaian Voltage Regulator

Pada aplikasi yang dibuat pada skripsi ini, tegangan masukan motor dc servo menggunakan sumber tegangan yang terpisah dari sumber tegangan mikrokontroler. Tujuan penggunaan sumber tegangan yang terpisah ini adalah agar mikrokontroler dapat memperoleh suplai arus yang mencukupi kebutuhannya. Pada saat bergerak, motor memerlukan suplai arus lebih besar. Sehingga jika menggunakan sumber tegangan yang sama, mikrokontroler tidak mendapatkan suplai arus yang mencukupi kebutuhannya. Hal ini dapat menyebabkan mikrokontroler menjadi *ter-reset* dan mengulang pekerjaannya dari awal.

Sumber tegangan yang diberikan pada motor harus berada pada daerah kerja tegangan dari motor yang digunakan. Selain itu, sumber tegangan motor dc servo pada skripsi ini diharapkan stabil sehingga perlu dirancang suatu rangkaian voltage regulator. Rangkaian *voltage regulator* yang dirancang pada skripsi ini ditunjukkan pada gambar 3.3.



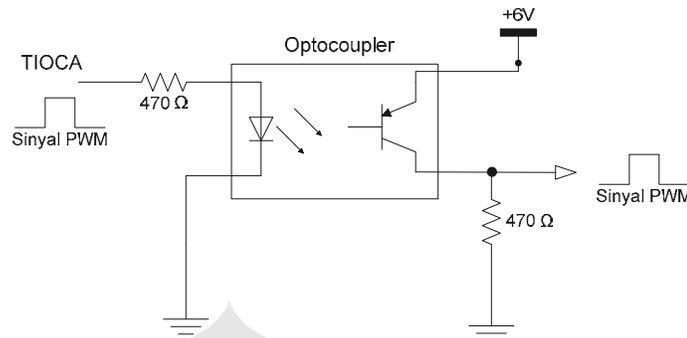
Gambar 3.3. Rancangan Rangkaian Voltage Regulator

Pada rancangan yang dibuat pada skripsi ini, digunakan diode untuk memastikan polaritas tegangan masukan adalah positif. Sedangkan IC 7806 digunakan untuk menghasilkan tegangan keluaran ke motor yang stabil yaitu sekitar 6 volt.

3.2.5. Rangkaian *Optocoupler*

Pada rancangan perangkat keras yang dibuat pada skripsi ini, rangkaian motor dan mikrokontroler H8/3052F harus benar-benar terpisah. Hal ini bertujuan agar suplai arus menuju mikrokontroler optimal. Namun, sinyal PWM yang digunakan untuk menggerakkan motor dc servo berasal dari mikrokontroler. Untuk mensiasati keadaan tersebut, maka digunakan suatu rangkaian optocoupler tambahan sehingga akan diperoleh sinyal PWM baru yang mendekati bentuk

sinyal PWM yang dihasilkan oleh mikrokontroler pada pin TIOCA. Rangkaian *optocoupler* yang dirancang ditunjukkan pada gambar 3.4.



Gambar 3.4. Rancangan Rangkaian Optocoupler

Prinsip kerja rangkaian *optocoupler* menyerupai *switch* yang *on/off*-nya diatur oleh ada tidaknya tegangan pada *base* fototransistor. Saat terdapat tegangan masukan pada *photodiode*, maka pada *base* fototransistor akan terdapat masukan tegangan. Hal ini menyebabkan hubung singkat pada transistor sehingga dapat diperoleh tegangan keluaran dari pin transistor. Sebaliknya, saat tidak terdapat tegangan masukan pada *photodiode*, maka pada *base* fototransistor mendapat masukan tegangan. Hal ini menyebabkan hubung terbuka pada transistor sehingga tidak terdapat tegangan keluaran dari pin transistor.

3.2.6. Perancangan Lengan Motor

Lengan yang digunakan untuk menggerakkan motor berupa batangan aluminium. Panjang aluminium yang digunakan sebagai lengan motor sepanjang ± 15 cm.

3.2.7. Hasil Perancangan Perangkat Keras

Setelah melakukan perancangan terhadap masing-masing bagian perangkat keras, dilakukan penggabungan dan instalasi alat menjadi satu bagian. Rangkaian hasil perancangan perangkat keras untuk pengendalian bilateral robot ditunjukkan pada gambar 3.5.



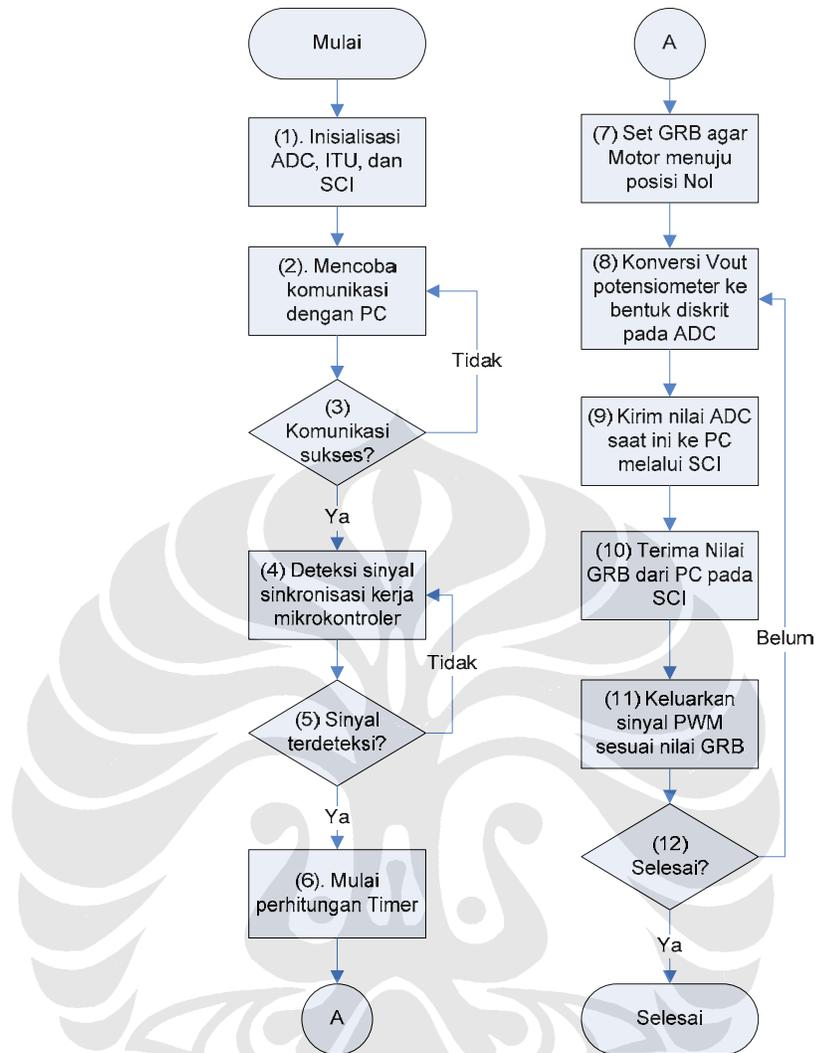
Gambar 3.5. Rangkaian hasil perancangan perangkat keras untuk pengendalian bilateral robot

3.3. Perancangan Perangkat Lunak

Perangkat lunak yang digunakan dalam skripsi ini diantaranya sistem operasi Linux Redhat 9.0, serta program pada sisi PC dan mikrokontroler. Pada skripsi ini, mikrokontroler berfungsi sebagai unit data akuisisi, sedangkan PC berperan sebagai unit pengendali gerakan lengan robot.

3.3.1. Program pada Sisi Mikrokontroler

Agar mikrokontroler dapat berperan sebagai unit data akuisisi, maka pada mikrokontroler harus dimasukkan suatu program dengan algoritma yang sesuai. Karena pada skripsi ini, mikrokontroler hanya berfungsi sebagai unit data akuisisi, maka kedua program yang dimasukkan pada mikrokontroler dibuat sama. Diagram alir algoritma program pada sisi mikrokontroler ditunjukkan gambar 3.6.



Gambar 3.6. Diagram Alir Algoritma Program pada Sisi Mikrokontroler

Pekerjaan yang dilakukan mikrokontroler dimulai dengan inisialisasi fungsi-fungsi mikrokontroler yang digunakan pada skripsi ini. Kemudian, dilanjutkan dengan mencoba komunikasi serial antara PC dan mikrokontroler. Pengujian komunikasi dilakukan dengan mengirimkan suatu nilai byte tertentu ke komputer dan membaca nilai yang dikirimkan kembali oleh PC ke mikrokontroler. Jika byte yang terbaca pada mikrokontroler sesuai, artinya komunikasi antar PC dan mikrokontroler berlangsung sukses.

Setelah komunikasi antar mikrokontroler dan PC terbukti sukses, mikrokontroler akan menunggu sinyal dari PC yang menandakan mulainya kerja

dan *timer* pada mikrokontroler. Tujuan adanya sinyal ini adalah agar kedua perangkat PC dan mikrokontroler dapat mulai bekerja secara bersamaan.

Agar pengolahan data pada PC dapat berjalan dengan baik, maka perlu dilakukan kalibrasi terhadap keluaran potensiometer. Untuk itu, sudut motor harus dinolkan dengan cara mengatur nilai GRB pada mikrokontroler agar diperoleh PWM yang sesuai.

Setelah melakukan proses kalibrasi, maka mikrokontroler akan memasuki looping program yang hanya dapat dimatikan dengan *me-reset* atau mematikan sumber tegangan masukan ke mikrokontroler. Prinsip kerja looping program yang dibuat yaitu, mengkonversi tegangan analog keluaran potensiometer menjadi bentuk digital oleh ADC, kemudian mengirimkannya ke PC melalui SCI. Setelah itu, mikrokontroler akan membaca nilai hasil olahan PC berupa GRB pada SCI. Nilai GRB ini kemudian akan digunakan untuk menghasilkan sinyal PWM yang menggerakkan motor ke sudut tertentu sesuai dengan *duty cycle* pada PWM tersebut.

3.3.2. Program pada Sisi PC

Program yang dirancang pada kedua sisi komputer, memiliki algoritma dan prinsip kerja yang hampir sama.

Pada awal program, port serial (`/dev/ttyS0`) perlu dibuka dan diinisialisasi. Dalam hal ini, inisialisasi bertujuan untuk mengkonfigurasi format data, bit rate, parity, dan konfigurasi lainnya. Setelah inisialisasi port serial, dibuat socket UDP untuk melakukan komunikasi dengan PC lain melalui protokol *datagram* (UDP).

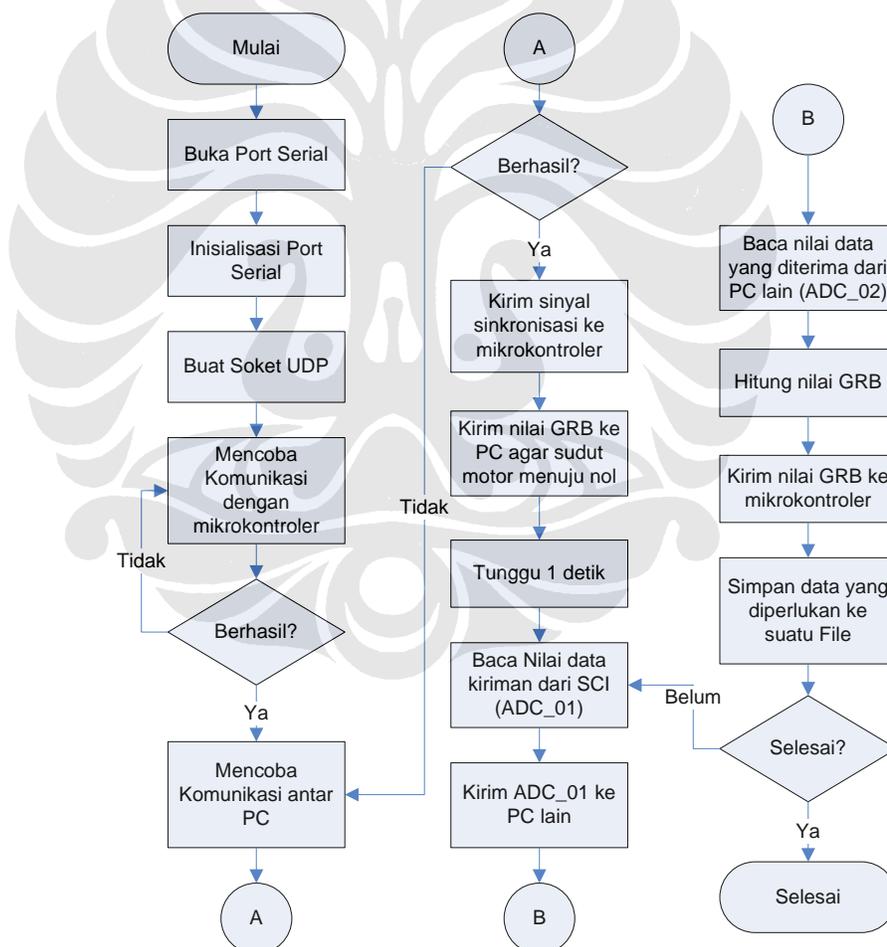
Kemudian, PC akan mencoba komunikasi dengan mikrokontroler melalui kabel serial hingga berhasil. Hal ini, dilanjutkan dengan pengujian komunikasi dengan PC lain melalui jaringan komputer yang ada hingga terbukti berhasil

Untuk keperluan kalibrasi potensiometer, maka PC akan mengirim nilai GRB yang nilainya tertentu ke mikrokontroler untuk menghasilkan PWM yang dapat menggerakkan motor menuju sudut nol. Karena motor memerlukan waktu untuk menuju sudut nol, maka diberikan waktu tunggu selama 1 detik pada PC.

Setelah melakukan proses kalibrasi, maka program C yang dibuat akan memasuki *looping* program. Proses *looping* ini dimulai dengan pembacaan nilai

keluaran ADC mikrokontroler (ADC_01) dan mengirimkan nilai tersebut ke PC lainnya. Setelah itu, perlu dilakukan pembacaan data (ADC_02) dari PC lain melalui soket UDP yang telah dibuat sebelumnya.

Perhitungan nilai GRB dilakukan dengan menggunakan data nilai keluaran ADC yang diterima dari mikrokontroler dan nilai keluaran ADC yang diterima dari jaringan komputer. Setelah diperoleh nilai GRB, maka nilai ini akan dikirim ke mikrokontroler untuk menghasilkan sinyal PWM. Kemudian, dilakukan pengambilan data untuk tiap *looping* program yang dijalankan dengan cara menyimpannya ke bentuk suatu file. Gambar 3.7. menunjukkan diagram alir algoritma program pada sisi PC.

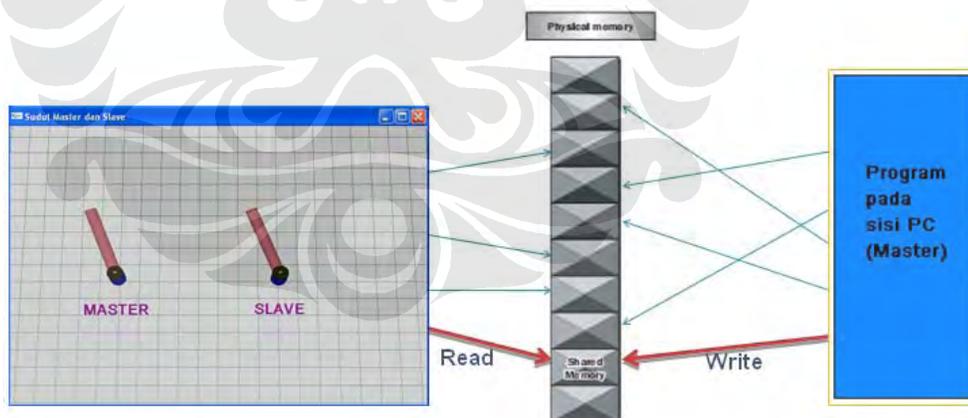


Gambar 3.7. Diagram Alir Algoritma Program Pada Sisi PC

3.4. Perancangan Program Grafis dengan OpenGL

Pada sistem telerobotik yang sebenarnya, lokasi manipulator *slave* biasanya cukup jauh dari keberadaan manipulator *master* dan operator. Hal ini menyebabkan operator sulit mengetahui keadaan dan posisi manipulator *slave*. Oleh karena itu, dibuat suatu aplikasi grafis tambahan pada komputer *master* dengan bantuan OpenGL. Variabel yang digunakan untuk menggerakkan lengan robot *master* dan *slave* pada program grafis yang dibuat adalah sudut lengan *master* dan *slave*.

Agar lengan *master* dan *slave* pada program grafis selalu bergerak sesuai dengan keadaan lengan yang sebenarnya, maka digunakan metode *share memory* pada variabel sudut antara program grafis OpenGL dan program utama yang terdapat pada PC *master*. Ketika PC *master* memperoleh data update-an berupa nilai hasil konversi A/D *master* dan *slave*, nilai ini dikonversi menjadi sudut *master* dan *slave*. Karena isi variabel sudut berubah, gambar lengan pada program grafis juga berubah mengikuti perubahan yang sudut *master* dan *slave* yang terjadi. Ilustrasi kerja program grafis OpenGL yang dibuat ditunjukkan pada gambar 3.8.



Gambar 3.8. Ilustrasi Kerja Program Grafis dengan OpenGL

3.5. Kalibrasi Nilai Keluaran Konverter A/D dan GRB

Besarnya jangkauan sudut motor pada skripsi ini dibatasi hingga 150° saja. Kalibrasi GRB dilakukan dengan bantuan busur derajat yang diletakkan pada

poros sendi lengan robot. Nilai GRB akan dicari ketika lengan robot berada pada sudut 0° dan 150° untuk nilai GRA yang tetap.

Kalibrasi nilai keluaran konverter A/D dilakukan untuk mendapatkan jangkauan nilai keluaran tegangan potensiometer yang terhubung dengan motor, namun dalam bentuk digital. Kalibrasi dilakukan dengan menghubungkan pin keluaran tegangan potensiometer dengan pin masukan ADC pada mikrokontroler H8/3052F dan dilakukan konversi A/D. Selanjutnya, sudut putaran motor diatur agar menuju sudut 0° dan 150° dengan memanfaatkan nilai-nilai GRB yang diperoleh melalui konversi sebelumnya. Kemudian, hasil konversi A/D ditampilkan pada LCD yang terdapat pada mikrokontroler H8/3052F kit.

Hasil kalibrasi nilai keluaran konverter A/D dan GRB ditunjukkan pada tabel 3.2.

Tabel 3.2 Hasil Kalibrasi Nilai Keluaran Konverter A/D dan GRB

Sudut	Nilai GRB		Nilai Keluaran A/D	
	Master	Slave	Master	Slave
0	3700	3700	88	88
150	5900	5900	198	198
Beda	2200	2200	110	110

BAB 4 ANALISA

4.1. Pengujian Respon Aktuator Slave

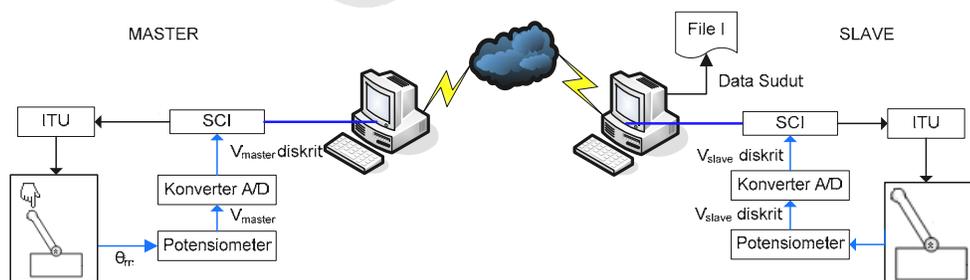
Pada bagian ini, dilakukan pengujian respon aktuator *slave* terhadap pergerakan aktuator *master* yang diberikan oleh operator.

4.1.1. Prosedur Pengujian Respon Aktuator Slave

Prosedur pengujian respon motor *slave* dilakukan dengan menggerakkan lengan yang dihubungkan terhadap motor *master*. Pergerakan lengan ini dilakukan secara manual oleh operator.

Adanya perubahan posisi sudut pada manipulator *master* akan memicu perubahan tegangan keluaran pada potensiometer yang dihubungkan dengan motor *master*. Tegangan keluaran ini dikonversi menjadi bentuk digital oleh konverter A/D sebelum dikirim ke PC *master* melalui koneksi serial. Setelah PC *master* menerima data hasil ADC dari mikrokontroler, data tersebut akan dikirimkan ke PC *slave* melalui jaringan komputer yang menghubungkan kedua PC yang digunakan. Data yang diterima PC *slave* dari PC *master* akan dijadikan referensi masukan untuk menghasilkan PWM yang sesuai, sehingga motor *slave* akan mengikuti pergerakan motor *master*.

PC pada sisi *slave* akan digunakan untuk mencatat posisi sudut motor *master* dan *slave* pada setiap waktu. Ilustrasi proses pengujian respon aktuator *slave* terhadap pergerakan sudut aktuator *master* ditunjukkan gambar 4.1

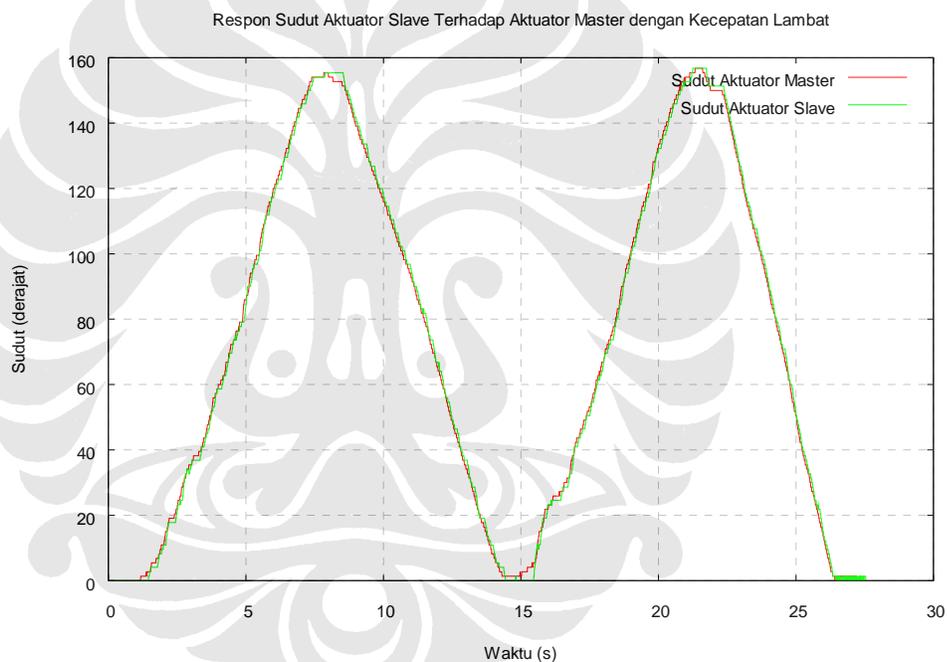


Gambar 4.1. Ilustrasi proses pengujian respon aktuator slave

Pengujian respon aktuator slave, dilakukan dengan memberikan beberapa perlakuan yang berbeda pada aktuator *master*. Beberapa perlakuan yang diberikan, dilakukan dengan cara membedakan kecepatan perubahan sudut masukan pada manipulator master oleh operator.

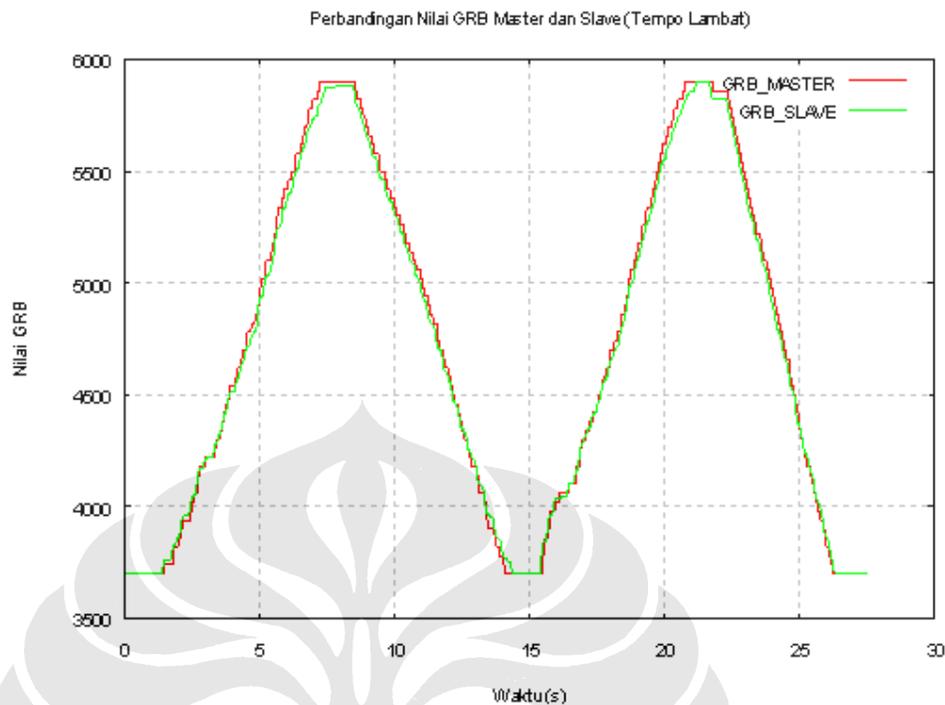
4.1.2. Hasil Pengujian Respon Aktuator Slave

Untuk pengujian respon aktuator *slave* yang pertama, dilakukan pengujian respon terhadap perubahan sudut aktuator *master* dengan tempo lambat. Hasil pengujian respon aktuator *slave* terhadap perubahan sudut aktuator *master* dengan kecepatan yang lambat ditunjukkan pada gambar 4.1.



Gambar 4.1 Respon Aktuator Slave terhadap Perubahan Sudut pada Aktuator Master dengan Tempo Lambat

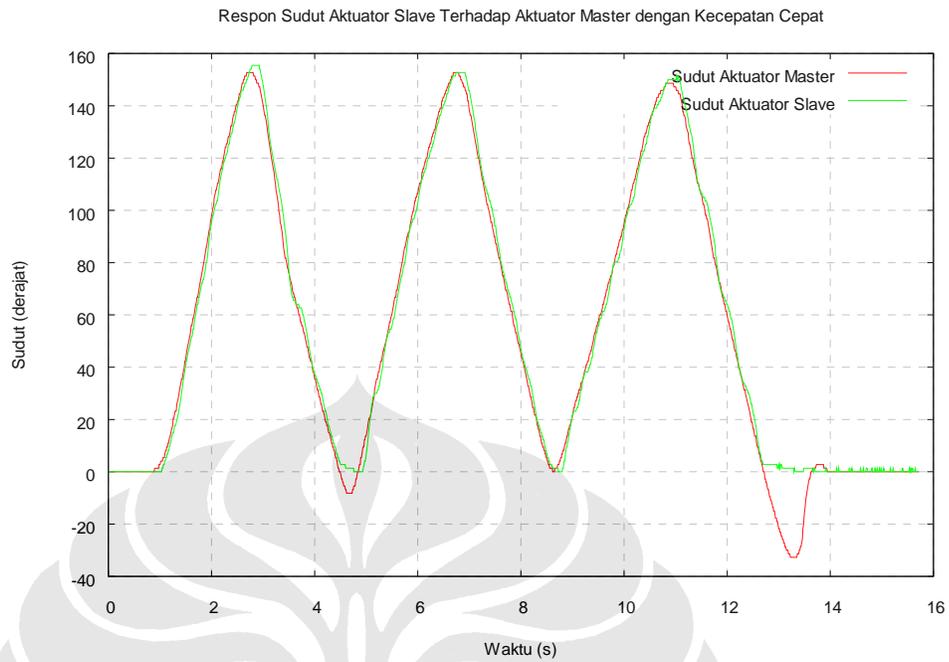
Sedangkan nilai GRB pada sisi master dan slave untuk tempo lambat ditunjukkan pada gambar 4.2.



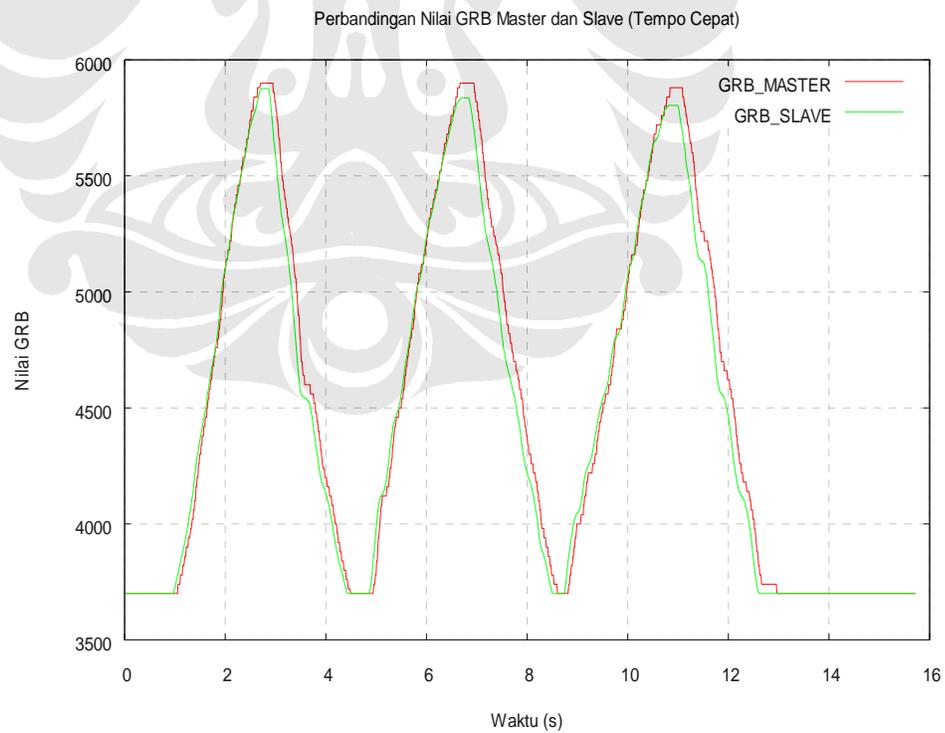
Gambar 4.2. Perbandingan Nilai GRB Master dan Slave untuk Tempo Lambat

Pada pengujian yang dilakukan, kecepatan rata-rata perubahan sudut pada aktuator master untuk tempo lambat adalah sebesar $\pm 46,15^\circ/\text{detik} \approx 0,815$ rad/detik. Dari grafik respon aktuator dengan tempo lambat, diketahui bahwa respon aktuator *slave* terhadap pergerakan aktuator *master* baik. Hal ini terlihat dari pergerakan aktuator *slave* yang dapat mengikuti pergerakan aktuator *master* sebagai referensi masukan dengan waktu jeda yang dapat ditoleransi.

Untuk pengujian respon aktuator slave yang kedua, dilakukan pengujian respon terhadap perubahan sudut aktuator *master* dengan tempo cepat. Hasil pengujian respon aktuator *slave* terhadap perubahan sudut aktuator *master* dengan kecepatan yang cepat ditunjukkan pada gambar 4.3. Sedangkan nilai GRB pada sisi master dan slave untuk tempo cepat ditunjukkan pada gambar 4.4.



Gambar 4.3 Respon Aktuator Slave terhadap Perubahan Sudut pada Aktuator Master dengan Tempo Cepat

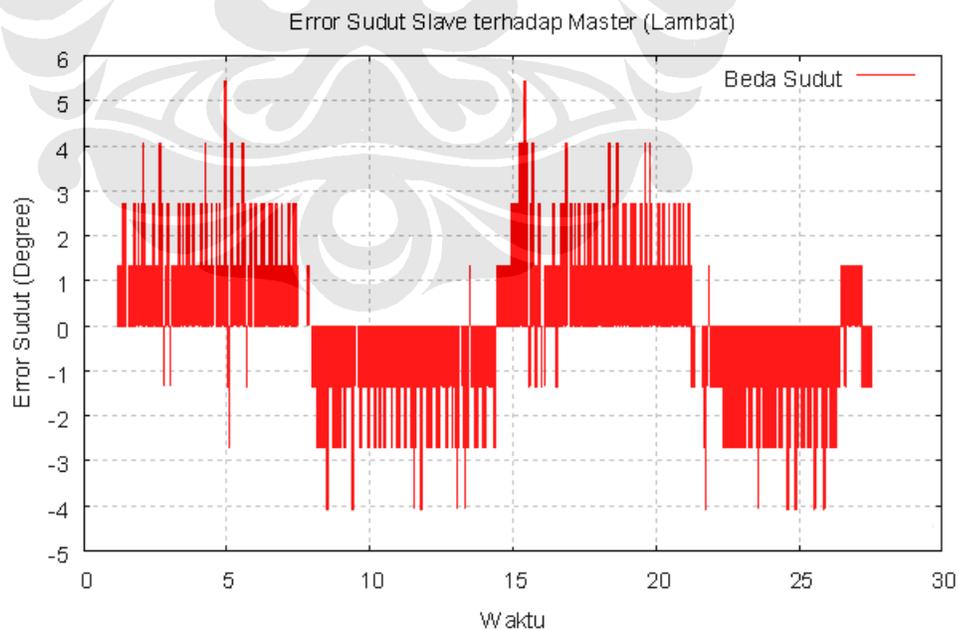


Gambar 4.4. Perbandingan Nilai GRB Master dan Slave untuk Tempo Cepat

Pada pengujian yang dilakukan, kecepatan rata-rata perubahan sudut pada aktuator master untuk tempo lambat adalah sebesar $\pm 75^\circ/\text{detik} \approx 1,308 \text{ rad/detik}$. Dari grafik respon aktuator dengan tempo cepat, diketahui bahwa respon aktuator *slave* terhadap pergerakan aktuator *master* cukup baik. Hal ini terlihat dari pergerakan aktuator *slave* yang dapat mengikuti pergerakan aktuator *master* sebagai referensi masukan dengan waktu jeda yang dapat ditoleransi. Pada grafik pun terlihat adanya overshoot pada puncak sudut *master* yaitu saat perubahan sudut *master* berubah dari membesar menjadi mengecil. Pada gambar juga terlihat bahwa pada saat sudut aktuator *master* digerakkan operator menjadi lebih kecil dari 0° , maka sudut aktuator *slave* hanya akan mengikuti pergerakan aktuator *master* hingga 0° . Hal ini dikarenakan pada program yang dibuat pada skripsi ini, besar sudut yang dapat dijangkau motor hanya berkisar antara $0^\circ - 150^\circ$. Oleh karena itu, nilai minimum sudut aktuator *slave* adalah 0° .

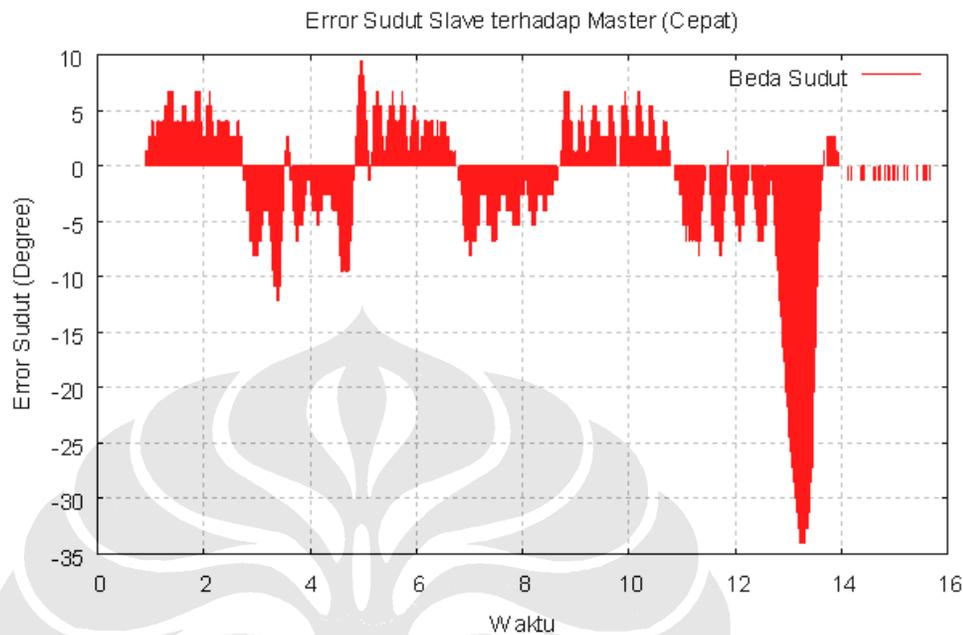
4.1.3. Error Respon Aktuator Slave Terhadap Master

Dari hasil pengukuran respon aktuator *slave* terhadap *master*, dapat diperoleh *error* posisi sudut *slave* terhadap *master*. Besarnya *error* posisi sudut *slave* terhadap *master* untuk tempo lambat ditunjukkan pada gambar 4.5



Gambar 4.5. Error Posisi Sudut Slave Terhadap Master untuk Tempo Lambat

Sedangkan besarnya error posisi sudut *slave* terhadap *master* untuk tempo lambat ditunjukkan pada gambar 4.6.



Gambar 4.6. Error Posisi Sudut Slave Terhadap Master untuk Tempo Cepat

Dari grafik *error* yang ditampilkan, diperoleh besarnya *error* maksimum respon *slave* terhadap *master* untuk tempo lambat ($46,15^\circ/\text{detik}$) adalah $\pm 5,45^\circ$. Sedangkan besarnya *error* maksimum respon *slave* terhadap *master* untuk tempo lambat ($75^\circ/\text{detik}$) adalah $\pm 12,55^\circ$. Perhitungan *error* untuk respon *slave* pada tempo cepat dilakukan hanya hingga detik ke-13. Hal ini dikarenakan, pada detik yang selanjutnya, pergerakan sudut *master* yang diberikan operator melewati sudut daerah kerja motor yang ditentukan.

4.2. Pengujian Pengaruh Objek yang Menahan Pergerakan Manipulator Slave

Pada bagian ini, dilakukan pengujian pengaruh adanya objek yang menahan pergerakan manipulator slave, sehingga manipulator slave menjadi tidak mampu mengikuti pergerakan manipulator master yang diberikan oleh operator.

4.2.1. Prosedur Pengujian Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave

Pada bagian ini, pengujian dilakukan dengan cara meletakkan suatu objek pada daerah jangkauan kerja motor. Untuk membuktikan, bahwa manipulator

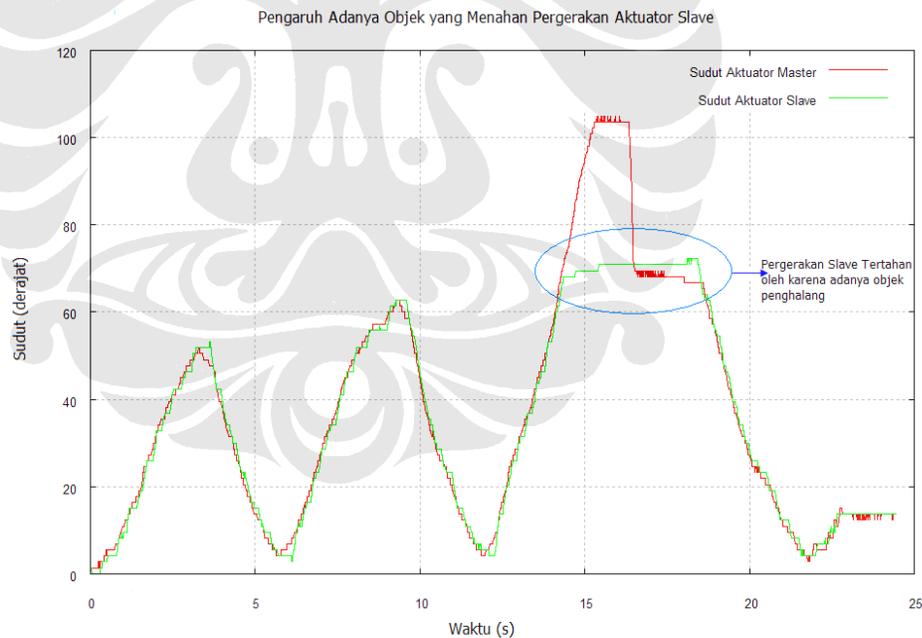
slave mampu mengikuti pergerakan manipulator master, pada keadaan awal, manipulator master digerakkan beberapa saat. Untuk itu, pergerakan manipulator slave perlu dijaga agar tidak menyentuh objek.

Setelah itu, operator mencoba menggerakkan manipulator *master* hingga manipulator *slave* menabrak objek. Manipulator *master* kemudian dilepaskan oleh operator untuk membuktikan bahwa pergerakan manipulator *master* dapat mengikuti pergerakan *slave*, jika pergerakan *slave* tersebut tertahan.

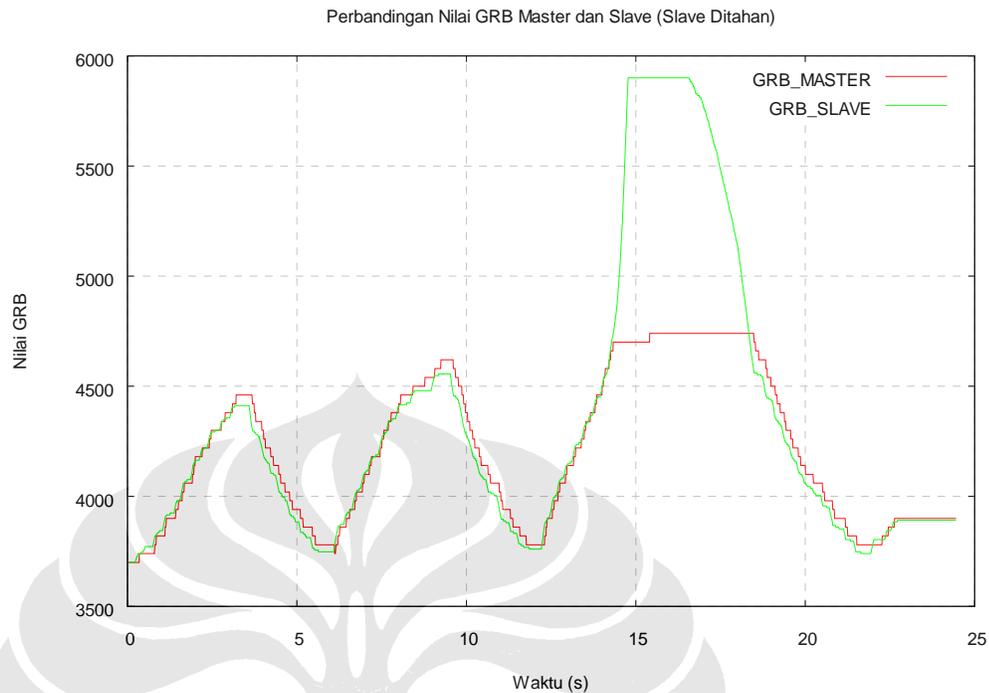
Kemudian, manipulator *master* akan kembali digerakkan sesaat untuk membuktikan manipulator *slave* masih mampu mengikuti pergerakan *master*.

4.2.2. Hasil Pengujian Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave

Hasil pengujian pengaruh adanya objek yang menahan pergerakan manipulator slave ditunjukkan pada gambar 4.7



Gambar 4.7 Pengaruh Adanya Objek yang Menahan Pergerakan Manipulator Slave Sedangkan grafik GRB master dan slave untuk pengujian adanya objek yang menahan pergerakan slave ditunjukkan pada gambar 4.8.



Gambar 4.8. Perbandingan Nilai GRB Master dan Slave Pada Pengujian Adanya Objek yang Menahan Pergerakan Slave

Pada awal grafik, terlihat bahwa manipulator *slave* mampu mengikuti pergerakan manipulator *master* dengan baik. Ketika pergerakan manipulator *slave* tertahan, manipulator *master* terlihat masih dapat digerakkan untuk melebihi sudut *slave* dengan cukup mudah. Padahal, jika memperhatikan grafik GRB pada gambar 4.8, terlihat bahwa nilai GRB *master* sudah mencapai nilai yang konstan. Seharusnya, motor *master* sudah tidak dapat digerakkan lagi karena duty cycle PWM yang dihasilkan tidak berubah. Hal ini dikarenakan, torsi tolak pada motor yang digunakan jauh lebih kecil dibandingkan torsi yang diberikan oleh operator. Pada sistem bilateral robot yang baik, seharusnya operator pada sisi *master* harus dapat merasakan adanya objek pada sisi *slave* yang diberikan oleh adanya torsi tolak lebih pada lengan oleh motor.

Saat pergerakan *slave* tertahan, pada grafik terlihat nilai GRB *slave* yang terus naik hingga nilai tertentu. Hal ini disebabkan motor *slave* berusaha mengikuti pergerakan *master* yang tidak pernah sampai dengan cara menaikkan nilai GRBnya. Karena nilai GRB pada kedua sisi dibatasi agar sudut motor tidak

berada di luar sudut $0^\circ - 150^\circ$, maka GRB *slave* akan tertahan pada nilai maksimum (5900).

Pada saat manipulator motor dilepaskan oleh operator, pada grafik terlihat bahwa manipulator *master* secara cepat akan berusaha mengikuti pergerakan manipulator *slave*. Hal ini menunjukkan bahwa, pengendalian sudut manipulator *master* dan *slave* sudah berjalan dengan cukup baik. Setelah tidak terdapat objek penahan gerakan manipulator *slave*, maka pergerakan manipulator *slave* akan kembali mengikuti pergerakan manipulator *master*.



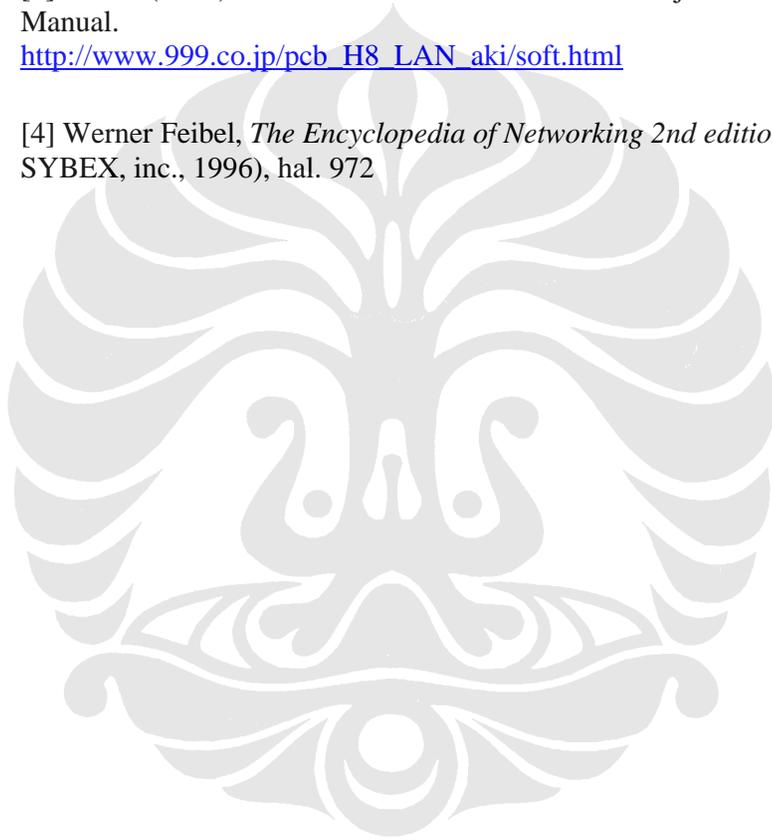
BAB 5 KESIMPULAN

Setelah melakukan pembahasan dan pengujian terhadap aplikasi yang dibuat, serta menganalisa hasil pengujian untuk beberapa kriteria prosedur yang dibahas pada bab sebelumnya dapat ditarik kesimpulan:

1. Sistem bilateral robot yang dibuat sudah cukup baik dalam hal respon, dimana respon pergerakan aktuator *slave* sudah mampu mengikuti pergerakan aktuator *master* untuk berbagai tempo kecepatan yang diberikan pada aktuator *master*.
2. Pada aplikasi yang dibuat, lengan master mampu mengikuti pergerakan slave jika pada lengan master tidak diberikan torsi masukan yang melebihi torsi motor.
3. Pada skripsi ini, mikrokontroler digunakan sebagai unit data akuisisi, sedangkan PC digunakan sebagai unit pengontrol gerakan motor. Adapun hubungan antara PC dan mikrokontroler menggunakan koneksi serial, dan hubungan antara PC *master* dan *slave* melalui jaringan komputer.
4. Pada pengujian respon aktuator slave dari aplikasi yang dibuat, didapat *error* maksimum posisi *slave* terhadap *master* untuk kecepatan $46,15^\circ$ adalah sebesar $5,45^\circ$. Sedangkan *error* maksimum posisi *slave* terhadap *master* untuk kecepatan 75° adalah sebesar $12,55^\circ$.
5. Setelah menambahkan lengan sepanjang 15 cm pada motor dc servo, torsi yang diberikan operator menjadi lebih besar daripada torsi yang dihasilkan oleh motor. Sehingga, motor *master* tidak mampu menahan pergerakan yang diberikan operator saat pergerakan motor *slave* ditahan.
6. Kelebihan sistem yang dibuat pada skripsi ini adalah letak pengontrolan pada sisi komputer, sehingga mempermudah pengembangan selanjutnya dalam penambahan kontroler untuk memperbaiki respon motor. Selain itu, proses komputasi juga dapat lebih cepat dan presisi.

DAFTAR ACUAN

- [1] Alireza Alfi, Mohammad Farrokhi, *Bilateral Control to Achieve Transparent Teleoperation with Perturbation of Static Time Delay*, Iran University of Science and Technology
- [2] Hitachi, *Hitachi Single-Chip Microcomputer, H8/3052 F-ZTAT Hardware Manual*, Hitachi, Ltd.
- [3] Maum (2002). *AKI-H8-LAN document*. diakses 25 januari 2008 dari H8-AKI Manual.
http://www.999.co.jp/pcb_H8_LAN_aki/soft.html
- [4] Werner Feibel, *The Encyclopedia of Networking 2nd edition*, (San Francisco: SYBEX, inc., 1996), hal. 972



DAFTAR PUSTAKA

_____, *H8 Family Tutorial Course*, Renesas, terakhir diakses 12 februari 2008.

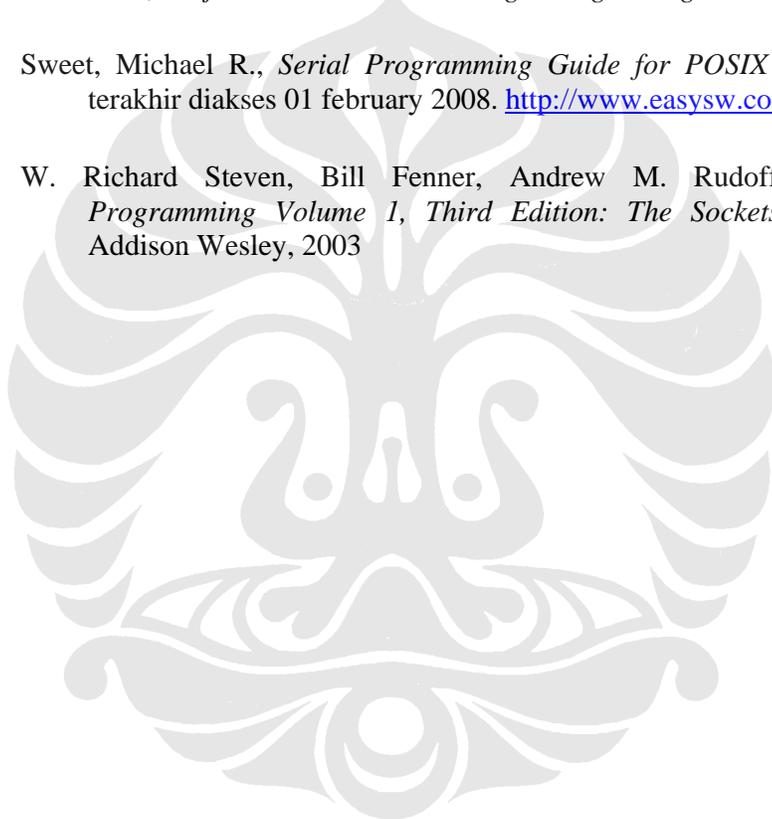
http://resource.renesas.com/lib/eng/e_learnig/h8_300henglish/index.html

Abdul Muis, Atsushi Kato, dkk, *RTLinux based Real-time Control Cooperation through Internet Protocol*. Journal.

Brian Hall, *Beej's Guide to Network Programing: Using Internet Sockets*, 2001

Sweet, Michael R., *Serial Programming Guide for POSIX Operating System*, terakhir diakses 01 february 2008. <http://www.easysw.com/~mike/serial>

W. Richard Steven, Bill Fenner, Andrew M. Rudoff, *UNIX Network Programming Volume 1, Third Edition: The Sockets Networking API*, Addison Wesley, 2003



LAMPIRAN

I. Rangkaian Hasil Perakitan Perangkat Keras

