

**PERANCANGAN PENGENDALIAN ROBOT  
BILATERAL BERBASIS MIKROKONTROLER**

**SKRIPSI**

Oleh

**ANGGI PURWANTO**

**04 04 03 010 5**



**SKRIPSI INI DIAJUKAN UNTUK MELENGKAPI SEBAGIAN  
PERSYARATAN MENJADI SARJANA TEKNIK**

**PROGRAM STUDI TEKNIK ELEKTRO  
DEPARTEMEN ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS INDONESIA  
GENAP 2007/2008**

## **PERNYATAAN KEASLIAN SEMINAR**

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul :

### **PERANCANGAN PENGENDALIAN ROBOT BILATERAL BERBASIS MIKROKONTROLER**

yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, 14 Juli 2008

Anggi Purwanto

NPM 04 04 03 010 5

# PENGESAHAN

Skripsi dengan judul:

## **PERANCANGAN PENGENDALIAN ROBOT BILATERAL BERBASIS MIKROKONTROLER**

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Studi Teknik Elektro Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia dan disetujui untuk diajukan dalam presentasi skripsi.

Depok, 14 Juli 2008

Dosen Pembimbing,

Abdul Muis, ST., M.Eng, Ph.D

NIP. 132 323 210

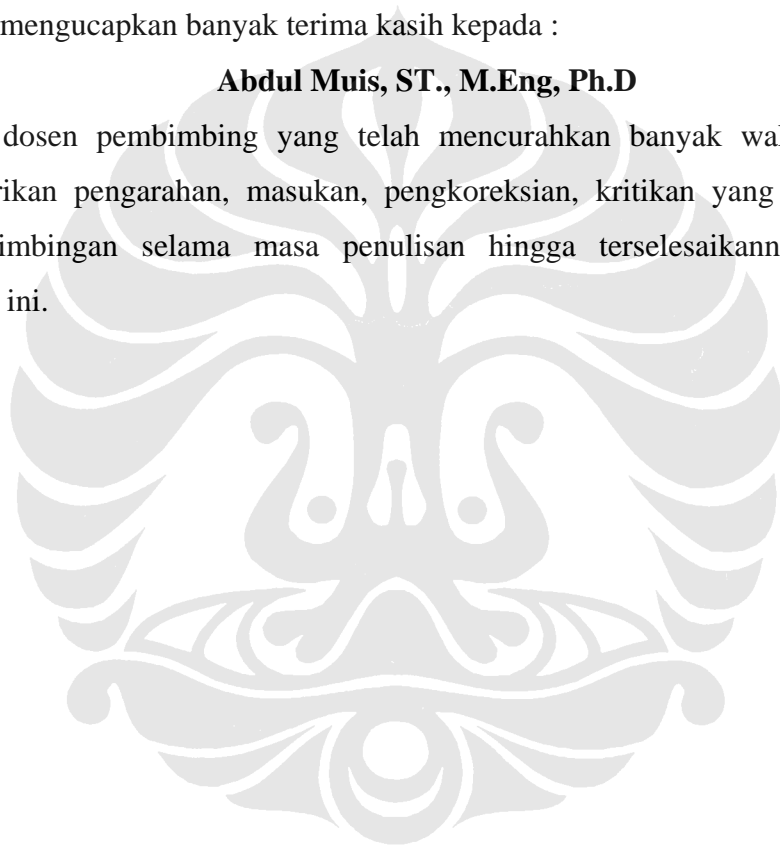
## UCAPAN TERIMA KASIH

Segala puji bagi Allah SWT atas segala limpahan rahmat dan kasih sayang-Nya sehingga penulisan seminar ini dapat terselesaikan dengan baik. Sholawat serta salam semoga tetap tercurah kepada Nabi Muhammad SAW.

Penulis mengucapkan banyak terima kasih kepada :

**Abdul Muis, ST., M.Eng, Ph.D**

Selaku dosen pembimbing yang telah mencurahkan banyak waktunya untuk memberikan pengarahan, masukan, pengkoreksian, kritikan yang membangun, serta bimbingan selama masa penulisan hingga terselesaikannya penulisan seminar ini.



Anggi Purwanto  
NPM 04 04 03 0105  
Departemen Teknik Elektro

Dosen Pembimbing  
Abdul Muis, ST., M.Eng, Ph.D

## PERANCANGAN PENGENDALIAN ROBOT BILATERAL BERBASIS MIKROKONTROLER

### ABSTRAK

Pengendalian suatu alat dari jarak jauh banyak mendapat perhatian pada saat ini. Ada yang menggunakan kabel ataupun tanpa kabel. Pengendalian ini pun harus dilakukan secara *real time*, apa yang diperintahkan untuk dikerjakan harus selalu di-*update* setiap waktu pada benda yang dikendalikan, dan diperlukan sebuah proses *feedback* yang digunakan untuk mengetahui apakah yang dikerjakan sesuai dengan yang diperintahkan dan akan dikirimkan melalui jarak yang sama dengan pengiriman, secara *real time* juga. Kemudian diperlukan pula suatu sistem dimana operator juga dapat merasakan sensasi sentuhan manipulator slave saat menyentuh objek sehingga operator seolah dapat berinteraksi langsung dengan objek. Pengendalian robot bilateral adalah konfigurasi dua manipulator robot yang saling mempengaruhi satu sama lain dan terkoneksi melalui sebuah jaringan yang dapat memenuhi ketiga spesifikasi di atas.

Dalam skripsi ini dibahas mengenai perancangan pengendalian sudut robot bilateral berbasis mikrokontroler dengan PC sebagai pengukur dan penyimpan data.

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 : Pengendalian bilateral, Mikrokontroler H8/3052F.**

Anggi Purwanto  
NPM 04 04 03 0105  
Electrical Departement Engineering

Counsellor  
Abdul Muis, ST., M.Eng, Ph.D

**DESIGNING CONTROL OF BILATERAL ROBOT BASED ON  
MICROCONTROLLER**

**ABSTRACT**

Tele-operation has been a great issue recently. Some application of this technology use wires and the rest use wireless connection. The controlled equipment has to be executed in real-time, all the tasks must be updated every time. This controlled equipment also has to send feedback that is used to sense if the task has been done rightly and it has to be sent in real-time across the same distance with sending process. Next, a system is definitely needed to make an operator sense the touch sense of slave manipulator when it touch an object, so the operator may interact with the object as if he/she done direct interaction. Bilateral robot is two-robot-manipulator configuration that influence each other and is connected by a network that qualifies three specifications above.

This bachelor thesis discusses designing control of bilateral robot based on microcontroller with PC as data acquiring and storage.

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.

**Kata kunci : Bilateral control, Mikrokontroller H8/3052F.**

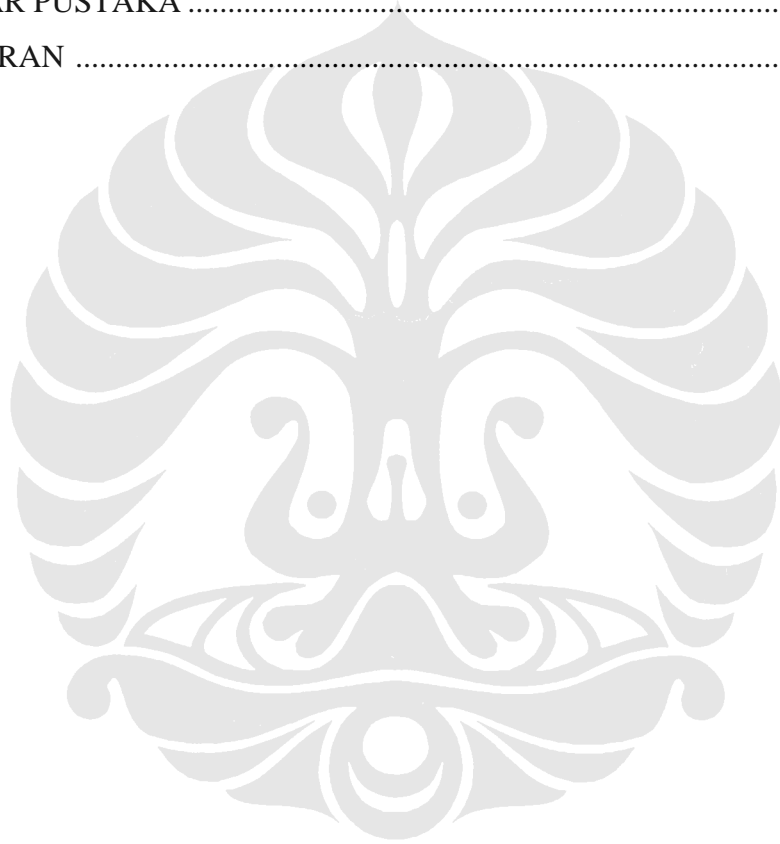
# DAFTAR ISI

	Halaman
PERNYATAAN KEASLIAN SKRIPSI.....	ii
PENGESAHAN .....	iii
UCAPAN TERIMA KASIH.....	iv
ABSTRAK .....	v
ABSTRACT .....	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR .....	x
DAFTAR TABEL.....	xii
DAFTAR LAMPIRAN.....	xiii
DAFTAR SINGKATAN .....	xiv
DAFTAR ISTILAH .....	xv
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG .....	1
1.2 TUJUAN PENULISAN.....	2
1.3 PEMBATASAN MASALAH.....	2
1.4 METODE PENELITIAN.....	3
1.5 SISTEMATIKA PENULISAN.....	3
BAB II DASAR TEORI .....	4
2.1 PRINSIP DASAR PENGENDALIAN BILATERAL.....	4
2.2 MIKROKONTROLER HITACHI H8/3052F [2].....	5
2.2.1 Gambaran umum H8/3052F .....	5
2.2.2 Konversi sinyal analog ke digital (ADC).....	5
2.2.2.1 Gambaran umum ADC .....	5
2.2.2.2 Deskripsi register .....	6
2.2.2.3 Operasi konversi A/D .....	8
2.2.2.4 Pemrograman ADC dalam bentuk bahasa C .....	9
2.2.3 16-Bit Integrated Timer Unit (ITU).....	10
2.2.3.1 Gambaran umum ITU .....	10
2.2.3.2 Register-register ITU .....	10

2.2.3.3 Operasi mode PWM.....	13
2.2.3.4 Pemrograman ITU dalam bentuk bahasa C .....	15
2.2.4 Pin Input/output .....	15
2.2.4.1 Gambaran Umum I/O .....	15
2.2.4.2 Port 3 .....	15
2.2.4.3 Konfigurasi Register .....	16
2.2.5 Serial Communication Interface (SCI) .....	17
2.2.5.1 Konfigurasi Pin dan Register .....	17
2.2.5.2 Operasi SCI pada Mode Asinkron pada Mikrokontroler H8/3052F.....	21
2.2.5.3 Pemrograman SCI dalam bentuk bahasa C.....	25
<b>BAB III PERANCANGAN ROBOT BILATERAL .....</b>	<b>26</b>
3.1 PERANCANGAN PERANGKAT LUNAK .....	26
3.1.1 Blok diagram pengendalian bilateral .....	26
3.1.2 Algoritma pengendalian bilateral.....	28
3.1.3 Algoritma pemrograman penerimaan data oleh PC.....	28
3.1.3.1 Inisialisasi komunikasi serial di PC .....	29
3.1.3.2 Algoritma komunikasi serial di PC.....	29
3.1.4 Algoritma pemrograman penggambaran 3D dengan Open GL .....	30
3.2 PERANCANGAN PERANGKAT KERAS .....	30
3.2.1 Motor servo Hitech HSR 5995T .....	31
3.2.2 Potensiometer RDC503 .....	32
3.2.3 Evaluation Kit AKI-H8/3052-LAN .....	32
3.2.4 Perancangan rangkaian <i>voltage regulator</i> .....	33
3.2.5 Perancangan rangkaian optocoupler .....	34
3.2.6 Perancangan filter <i>low pass</i> konverter analog ke digital .....	34
3.2.7 Perangkat keras komunikasi serial.....	35
3.2.7.1 Komunikasi serial antar mikrokontroler .....	35
3.2.7.2 Komunikasi serial dengan PC.....	35
3.2.8 Rangkaian pengambilan data kecepatan motor.....	37
3.3 KALIBRASI NILAI GRB DAN ADC .....	37



BAB IV PENGUJIAN DAN ANALISA .....	38
4.1 PENGUJIAN RESPON MANIPULATOR <i>SLAVE</i> .....	38
4.2 HASIL PENGUJIAN RESPON MANIPULATOR <i>SLAVE</i> .....	38
4.3 PENGUJIAN KESESUAIAN FUNGSI MANIPULATOR <i>MASTER</i> .....	39
4.4 HASIL PENGUJIAN KESESUAIAN FUNGSI MANIPULATOR <i>MASTER</i> .....	40
BAB V KESIMPULAN .....	42
DAFTAR ACUAN .....	43
DAFTAR PUSTAKA .....	44
LAMPIRAN .....	45



## DAFTAR GAMBAR

	Halaman
<b>Gambar 1.1</b> Sistem kendali bilateral [1].....	1
<b>Gambar 1.2</b> Ilustrasi sistem robot bilateral yang akan dirancang .....	1
<b>Gambar 2.1</b> Blok diagram pengendalian bilateral.....	4
<b>Gambar 2.2</b> Struktur register data A/D .....	7
<b>Gambar 2.3</b> Diagram alir (a) operasi ADC dan (b) inisialisasi ADC pada mode <i>single</i> .....	9
<b>Gambar 2.4</b> Contoh prosedur mode PWM.....	14
<b>Gambar 2.5</b> Contoh operasi mode PWM .....	14
<b>Gambar 2.6</b> Konfigurasi pin port 3 .....	16
<b>Gambar 2.7</b> Struktur P3DDR .....	16
<b>Gambar 2.8</b> Struktur P3DR .....	17
<b>Gambar 2.9</b> Format komunikasi serial dengan 8-bit data dengan parity dan 2-bit stop .....	21
<b>Gambar 2.10</b> Flowchart proses inisialisasi serial pada operasi asinkron .....	22
<b>Gambar 2.11</b> Flowchart proses transmisi data serial pada operasi asinkron.....	23
<b>Gambar 2.12</b> Flowchart proses penerimaan data serial pada operasi asinkron.....	24
<b>Gambar 3.1</b> Blok diagram pengendalian bilateral yang dibuat .....	27
<b>Gambar 3.2</b> Algoritma komunikasi serial di PC .....	29
<b>Gambar 3.3</b> Tampilan penggambaran manipulator secara 3D.....	30
<b>Gambar 3.4</b> Gambar rangkaian perangkat keras pengendalian bilateral.....	31
<b>Gambar 3.5</b> Motor servo Hitech HSR 5995TG.....	31
<b>Gambar 3.6</b> Gambar Eval-kit AKI-H8/3052-LAN .....	33
<b>Gambar 3.7</b> Gambar rangkaian <i>voltage regulator</i> .....	33
<b>Gambar 3.8</b> Gambar rangkaian optocoupler .....	34
<b>Gambar 3.9</b> Rangkaian filter <i>low pass</i> konverter analog ke digital .....	34
<b>Gambar 3.10</b> Koneksi pin serial SCI <i>master</i> dan SCI <i>slave</i> .....	35
<b>Gambar 3.11</b> RS-232C DB-9 konektor tampak depan.....	35
<b>Gambar 3.12</b> Koneksi Kabel Pin Serial SCI dengan Port Serial Komputer.....	36
<b>Gambar 3.13</b> Blok diagram pengambilan data kecepatan motor.....	37

<b>Gambar 4.1</b>	Grafik perbandingan sudut <i>master</i> dan sudut <i>slave</i> terhadap waktu untuk tempo lambat .....	38
<b>Gambar 4.2</b>	Grafik perbandingan sudut <i>master</i> dan sudut <i>slave</i> terhadap waktu untuk tempo cepat.....	39
<b>Gambar 4.3</b>	Grafik perbandingan sudut <i>master</i> dan sudut <i>slave</i> terhadap waktu ketika manipulator <i>slave</i> tertahan oleh objek .....	40
<b>Gambar 4.4</b>	Grafik perbandingan sudut <i>master</i> dan GRB <i>master</i> terhadap waktu ketika manipulator <i>slave</i> tertahan oleh objek .....	41



## DAFTAR TABEL

	Halaman
<b>Tabel 2.1</b> Pin Konverter A/D .....	6
<b>Tabel 2.2</b> Register konverter A/D .....	6
<b>Tabel 2.3</b> Kanal Input Analog dan Register data A/D-nya .....	7
<b>Tabel 2.4</b> Pin-pin Register ADCSR .....	8
<b>Tabel 2.5</b> Pin-pin Register ADCR .....	8
<b>Tabel 2.6</b> Pin-pin Register TSTR.....	10
<b>Tabel 2.7</b> Pin-pin Register TMDR .....	11
<b>Tabel 2.8</b> Pin-pin Register TCR.....	12
<b>Tabel 2.9</b> Pin-pin Register TIOR .....	12
<b>Tabel 2.10</b> Pin-pin Register TSR .....	13
<b>Tabel 2.11</b> Pin-pin Register TIER.....	13
<b>Tabel 2.12</b> Pin-pin SCI.....	17
<b>Tabel 2.13</b> Pin-pin Register SMR .....	19
<b>Tabel 2.14</b> Pin-pin Register SCR .....	19
<b>Tabel 2.15</b> Pin-pin Register SSR .....	20
<b>Tabel 2.16</b> Pemilihan Nilai n Berdasarkan Bit CKS1 dan CKS0 .....	20
<b>Tabel 3.1</b> Karakteristik Servo HSR 5995TG .....	32
<b>Tabel 3.2</b> Karakteristik Potensiometer RDC503.....	32
<b>Tabel 3.3</b> Penjelasan Pin-pin RS-232C DB-9 Beserta Kegunaannya .....	36
<b>Tabel 3.4</b> Kalibrasi Nilai GRB dan ADC.....	37

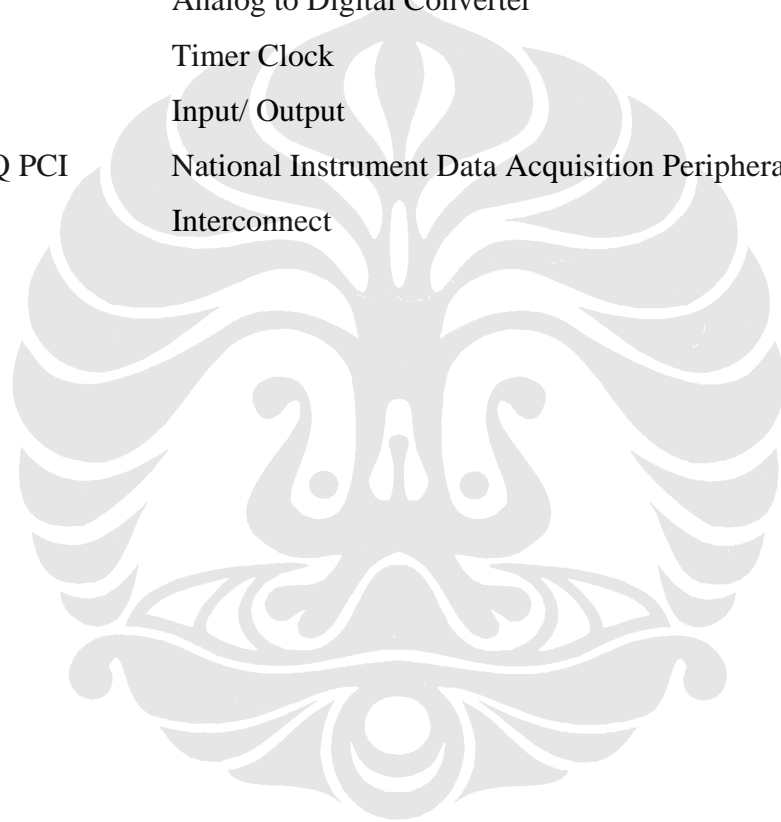
## DAFTAR LAMPIRAN

	Halaman
<b>Lampiran 1</b> Algoritma Pemrograman Pengendalian Bilateral Robot .....	45
<b>Lampiran 2</b> Blok Diagram Perangkat Keras Pengendalian Bilateral .....	46



## DAFTAR SINGKATAN

PC	Personal Computer
MC	Microcontroller
PWM	Pulse Width Modulation
MCU	Microcontroller Unit
CPU	Central Processing Unit
SCI	Serial Communication Interface
ADC	Analog to Digital Converter
TCLK	Timer Clock
I/O	Input/ Output
NI DAQ PCI	National Instrument Data Acquisition Peripheral Component Interconnect



## DAFTAR ISTILAH

<i>Bit</i>	digit dalam sistem bilangan biner, dapat bernilai 0 atau 1
<i>Byte</i>	seri 8-bit
<i>Degree of Freedom</i>	derajat kebebasan gerak
<i>Duty cycle</i>	perbandingan antara panjang sinyal on dengan panjang satu gelombang pada sinyal PWM
<i>General register</i>	memori penyimpanan bit
<i>Mikrokontroler</i>	<i>single purpose processing unit</i> yang didesain untuk melakukan program pengendalian kecil, kadang real time
<i>Most Significant Bit</i>	bit dengan nilai pangkat tertinggi dalam satuan bit
<i>Pulse Width Modulation</i>	tipe gelombang segi empat yang dimodulasi terhadap lebar sinyal

# BAB I

## PENDAHULUAN

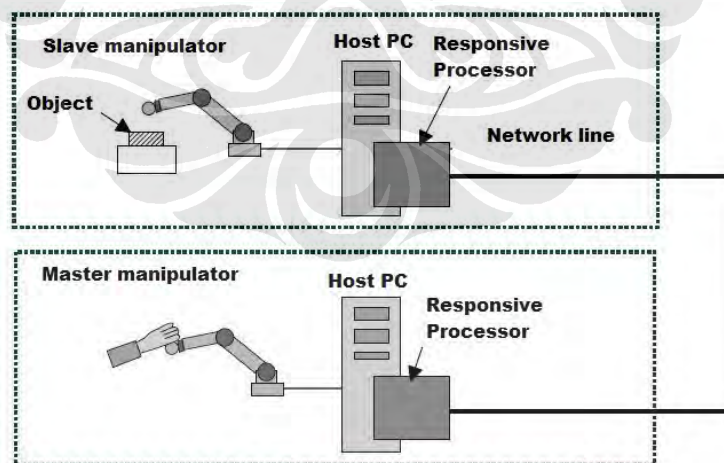
### 1.1 LATAR BELAKANG

Pengendalian suatu alat dari jarak jauh banyak mendapat perhatian pada saat ini. Ada yang menggunakan kabel ataupun tanpa kabel.

Pengendalian ini pun harus dilakukan secara *real time*, apa yang diperintahkan untuk dikerjakan harus selalu di-*update* setiap waktu pada alat yang dikendalikan, dan diperlukan sebuah proses *feedback* yang digunakan untuk mengetahui apakah yang dikerjakan sesuai dengan yang diperintahkan. Feedback ini akan dikirimkan melalui jarak yang sama dengan pengiriman, secara *real time* juga.

Kemudian diperlukan pula suatu sistem dimana operator juga dapat merasakan sensasi sentuhan manipulator *slave* saat menyentuh objek sehingga operator seolah dapat berinteraksi langsung dengan objek.

Robot bilateral adalah konfigurasi dua manipulator robot yang saling mempengaruhi satu sama lain dan terkoneksi melalui sebuah jaringan yang dapat memenuhi ketiga spesifikasi di atas [1]. Hal tersebut dapat digambarkan sebagai berikut:

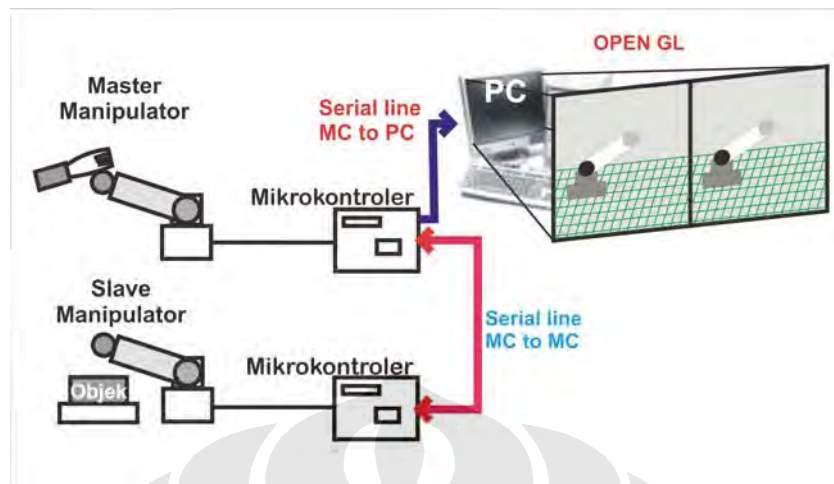


Gambar 1.1 Sistem kendali bilateral [1]

Sebagai pengganti *Personal Computer*, mikrokontroler pun dapat digunakan sebagai tempat pengendalian. Dua buah mikrokontroler digunakan sebagai *Responsive Processor* yang dihubungkan melalui komunikasi serial. Agar data



dapat dianalisa, maka data disimpan ke dalam *Personal Computer*. Hal tersebut dapat diilustrasikan sebagai berikut:



Gambar 1.2 Ilustrasi sistem robot bilateral yang akan dirancang

Pada pengendalian bilateral ini hal yang akan dikendalikan adalah sudut atau posisi dari manipulator.

## 1.2 TUJUAN PENULISAN

Tujuan penulisan skripsi ini, merancang suatu robot bilateral berbasis mikrokontroler dengan PC sebagai pengukur dan penyimpan data. Hal yang akan dikendalikan adalah sudut atau posisi manipulator.

## 1.3 PEMBATASAN MASALAH

Pembahasan dalam skripsi ini difokuskan pada kemampuan mikrokontroler Hitachi H8/3052F dalam pengendalian bilateral dua buah lengan robot satu DOF (sendi), yang meliputi ketepatan dan kecepatan *feedback* (manipulator *slave*) mengikuti referensi (manipulator *master*) serta kesesuaian fungsi manipulator *master* untuk men-*transfer* sensasi sentuhan yang dirasakan manipulator *slave* kepada operator. Skripsi ini tidak membahas penambahan pengendali seperti PI, PD, dan sebagainya.

## 1.4 METODE PENELITIAN

Metode penelitian yang digunakan dalam penyusunan skripsi ini meliputi:

1. Pendekatan tinjauan pustaka, yaitu dengan melakukan studi literatur dari buku-buku pustaka, atau *manual book* serta *reference book* dari suatu piranti.
2. Pendekatan diskusi dengan pembimbing skripsi yang berkaitan dengan topik bahasan skripsi.
3. Perancangan perangkat keras dan perangkat lunak robot bilateral.
4. Pengujian robot bilateral.

## 1.5 SISTEMATIKA PENULISAN

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

Bab pertama, Pendahuluan, memuat latar belakang masalah, tujuan penulisan, pembatasan masalah, metode penelitian, dan sistematika penulisan. Bab kedua, Dasar Teori Pengendalian Bilateral dan Mikrokontroler H8/3052F, menjelaskan tentang prinsip dasar pengendalian bilateral serta blok diagramnya. Serta menjelaskan dasar penggunaan Mikrokontroler H8/3052F yang meliputi penggunaan *Analog to Digital Converter*, penghasilan sinyal *Pulse Width Modulation*, penggunaan port input dan output, dan pengiriman serta penerimaan data serial. Bab ketiga, Perancangan Robot Bilateral Berbasis Mikrokontroler, menjelaskan tentang perancangan perangkat keras dan perangkat lunak robot bilateral dua buah manipulator menggunakan mikrokontroler, yang kemudian data pengendaliannya dikirimkan ke PC dan selanjutnya PC akan menggambarkan gerakan manipulator tersebut dalam bentuk 3D. Bab keempat, Pengujian dan Analisa Data, membahas pengujian dan penganalisaan data uji coba dari pengendalian bilateral yang telah dibuat. Bab kelima, Kesimpulan, merupakan kesimpulan dari penulisan skripsi ini.

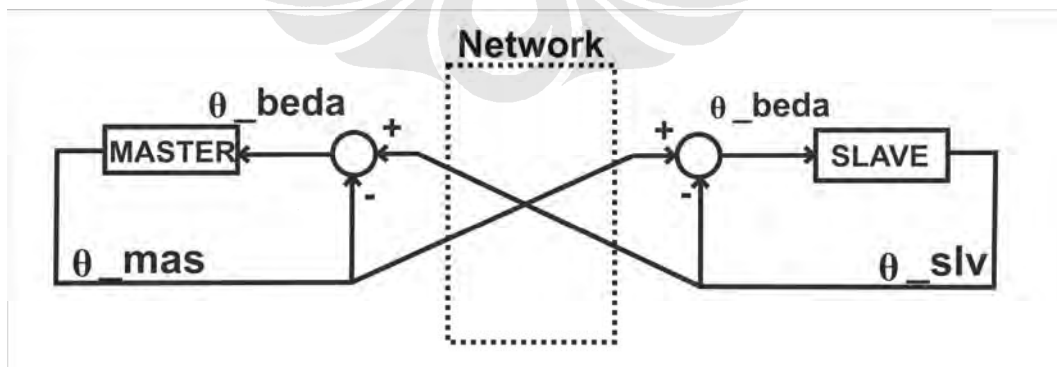
## BAB II

### DASAR TEORI

#### 2.1 PRINSIP DASAR PENGENDALIAN BILATERAL

Pengendalian bilateral adalah konfigurasi dua manipulator yang saling mempengaruhi satu sama lain dan terkoneksi melalui sebuah jaringan. Ketika operator menggerakkan manipulator *master*, manipulator *slave* akan bergerak mengikuti gerakan *master*. Operator juga dapat merasakan sensasi sentuhan manipulator *slave* saat menyentuh objek. Idealnya, sebuah sistem bilateral memungkinkan operator dapat merasakan seolah berinteraksi langsung dengan objek.

Pada pengendalian bilateral ini hal yang akan dikendalikan adalah posisi (sudut) *slave* atas perintah *master*. Serta membuat sisi *master* dapat merasakan apa yang dialami oleh sisi *slave*. Untuk itu kedua manipulator harus dibuat saling mempengaruhi satu sama lain. Dimana sudut *master* memberikan acuan pada manipulator *slave* dan sudut *slave* memberikan acuan pada manipulator *master*. Sudut yang akan masuk ke masing-masing manipulator adalah *error* atau beda dari sudut *slave* dan sudut *master*. Kedua manipulator terkoneksi melalui sebuah jaringan untuk saling mengirimkan referensi sudut. Komunikasi antar kedua manipulator dilakukan dengan menggunakan komunikasi data serial. Hal tersebut dapat diilustrasikan dengan gambar blok diagram berikut.



Gambar 2.1 Blok diagram pengendalian bilateral

## 2.2 MIKROKONTROLER HITACHI H8/3052F [2]

Adapun mikrokontroler yang digunakan pada skripsi ini adalah mikrokontroler H8/3052F, yaitu seri mikrokontroler (MCU) yang memiliki arsitektur Hitachi yang dapat mengintegrasikan fungsi sistem pendukung dengan inti CPU H8/300H.

### 2.2.1 Gambaran umum H8/3052F

H8/3052F mempunyai arsitektur internal 32 bit dengan enam belas *general register* 16 bit. Pewaktuan yang dapat dipilih, 25 MHz atau 16 Mhz. Memiliki dua buah *memory*, *flash memory* sebesar 512 kbyte dan RAM sebesar 8kbyte. Memiliki ruang alamat hingga 16 Mbyte.

Di dalamnya terdapat fungsi pendukung sistem termasuk ROM, RAM, Integrated Timer Unit 16 bit, Programmable Timing Patern Controller, konverter A/D, konverter D/A, port Input/Output, Serial Communication Interface, dan fungsi lainnya.

### 2.2.2 Konversi sinyal analog ke digital (ADC)

#### 2.2.2.1 Gambaran umum ADC

H8/3052F mempunyai 8 buah kanal input analog dengan resolusi 10 bit. Batasan konversi tegangan analog dapat diprogram dengan tegangan referensi analog pada pin  $V_{REF}$ . Waktu untuk konversinya mencapai 5,4  $\mu$ s per kanal (dengan clock sistem 25 MHz). H8/3052F mempunyai 2 mode konversi, *single mode* dan *scan mode*. *Single mode* digunakan untuk konversi analog digital satu kanal dan *scan mode* digunakan untuk konversi analog digital satu hingga empat kanal secara kontinu. Untuk menyimpan hasil konversi, H8/3052F mempunyai 4 buah register data 16-bit. Konversi analog digital dapat ditrigger secara eksternal. Pada akhir konversi A/D sebuah interrupt akhir konversi A/D dapat diminta.

Tabel 2.1 merangkum pin input konverter A/D. Delapan pin input analog dibagi ke dalam dua grup: grup 0 ( $AN_0$  sampai  $AN_3$ ) dan grup 1 ( $AN_4$  sampai  $AN_7$ ).  $AV_{CC}$  dan  $AV_{SS}$  adalah *power supply* untuk rangkaian analog dalam konverter A/D.

Tabel 2.1 Pin Konverter A/D

Nama Pin	Singkatan	I/O	Fungsi
Pin power supply analog	AV <sub>CC</sub>	Input	Power supply analog
Pin ground analog	AV <sub>SS</sub>	Input	Ground analog dan tegangan referensi
Pin tegangan referensi	V <sub>REF</sub>	Input	Tegangan referensi analog
Pin input analog 0	AN <sub>0</sub>	Input	Input analog grup 0
Pin input analog 1	AN <sub>1</sub>	Input	
Pin input analog 2	AN <sub>2</sub>	Input	
Pin input analog 3	AN <sub>3</sub>	Input	
Pin input analog 4	AN <sub>4</sub>	Input	Input analog grup 1
Pin input analog 5	AN <sub>5</sub>	Input	
Pin input analog 6	AN <sub>6</sub>	Input	
Pin input analog 7	AN <sub>7</sub>	Input	
Pin input trigger eksternal A/D	$\overline{ADTRG}$	Input	Input trigger eksternal untuk memulai konversi A/D

Tabel 2.2 merangkum register konverter A/D.

Tabel 2.2 Register Konverter A/D				
Alamat* <sup>1</sup>	Nama	Singkatan	R/W	Nilai awal
H'FFE0	Register data A/D A (high)	ADDRAH	R	H'00
H'FFE1	Register data A/D A (low)	ADDRAL	R	H'00
H'FFE2	Register data A/D B (high)	ADDRBH	R	H'00
H'FFE3	Register data A/D B (low)	ADDRBL	R	H'00
H'FFE4	Register data A/D C (high)	ADDRCH	R	H'00
H'FFE5	Register data A/D C (low)	ADDRCL	R	H'00
H'FFE6	Register data A/D D (high)	ADDRDH	R	H'00
H'FFE7	Register data A/D D (low)	ADDRDL	R	H'00
H'FFE8	Register kontrol/status A/D	ADCSR	R/(W)* <sup>2</sup>	H'00
H'FFE9	Register kontrol A/D	ADCR	R/W	H'7E

- Catatan:
1. Alamat 16 bit terbawah
  2. Hanya 0 yang dapat ditulis di bit ke-7, untuk meng-clear-kan flag

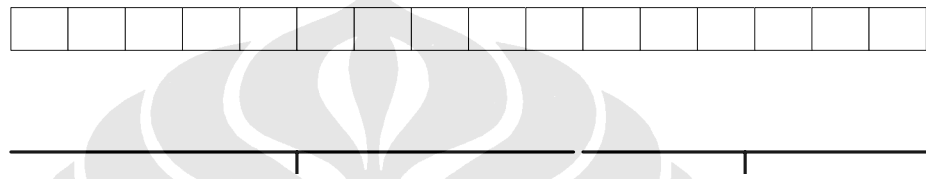
### 2.2.2.2 Deskripsi Register

Ada 3 jenis register untuk konverter A/D, yaitu register data (ADDR), register kontrol/status (ADCSR), dan register kontrol (ADCR).

### 1. Register Data A/D

Empat buah register data ADC (ADDRA sampai ADDRD) adalah register 16 bit yang menyimpan hasil konversi A/D.

Sebuah konversi A/D menghasilkan data 10 bit, yang disimpan pada register data A/D tergantung kanal yang dipilih. Delapan bit teratas disimpan pada byte atas dari register data A/D dan dua bit terbawah disimpan pada byte bawah. Bit 5 sampai 0 pada register data A/D akan selalu bernilai 0. Struktur dari register data A/D dapat dilihat pada gambar 2.2.



Gambar 2.2 Struktur register data A/D

Tabel 2.3 menunjukkan pembagian kanal input analog dan register data A/D-nya.

Tabel 2.3 Kanal Input Analog dan Register Data A/D-nya

**Analog Input Channel**

Grup 0	Grup 1	Register Data A/D						
AN <sub>0</sub>	AN <sub>4</sub>	15	ADDRA	14	13	12	11	10
AN <sub>1</sub>	AN <sub>5</sub>		ADDRB					
AN <sub>2</sub>	AN <sub>6</sub>		ADDRC					
AN <sub>3</sub>	AN <sub>7</sub>		ADDRD					

### 2. Register Kontrol/Status A/D (ADCSR)

ADCSR adalah register baca/tulis 8 bit memilih mode dan mengontrol konversi A/D. Tabel 2.4 merangkum pin-pin yang ada pada register ADCSR.

Bit	7	6	5	4	3	2	1	0
Nilai inisial	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

10 Bit Data Hasil Konve

\* n = A - D

Tabel 2.4 Pin-pin Register ADCSR

Bit	Nama	Singkatan	R/W	Fungsi
0	Channel select 0	CH0	R/W	bit ini memilih kanal input analog
1	Channel select 1	CH1	R/W	
2	Channel select 2	CH2	R/W	
3	Clock select	CKS	R/W	memilih pewaktuan konversi A/D
4	Scan mode	SCAN	R/W	memilih single mode atau scan mode
5	A/D start	ADST	R/W	start atau stop konversi A/D
6	A/D interrupt enable	ADIE	R/W	mengaktifkan atau menonaktifkan interrupt akhir A/D
7	A/D end flag	ADF	R/(W)*	menandakan akhir dari konversi A/D

Catatan : \*Hanya 0 yang dapat ditulis, untuk meng-clear-kan flag

### 3. Register Kontrol A/D (ADCR)

ADCR adalah register baca/tulis 8 bit yang mengaktifkan atau menonaktifkan trigger eksternal dari konversi A/D. Tabel 2.5 merangkum pin-pin yang ada pada register ADCR.

Tabel 2.5 Pin-pin Register ADCR

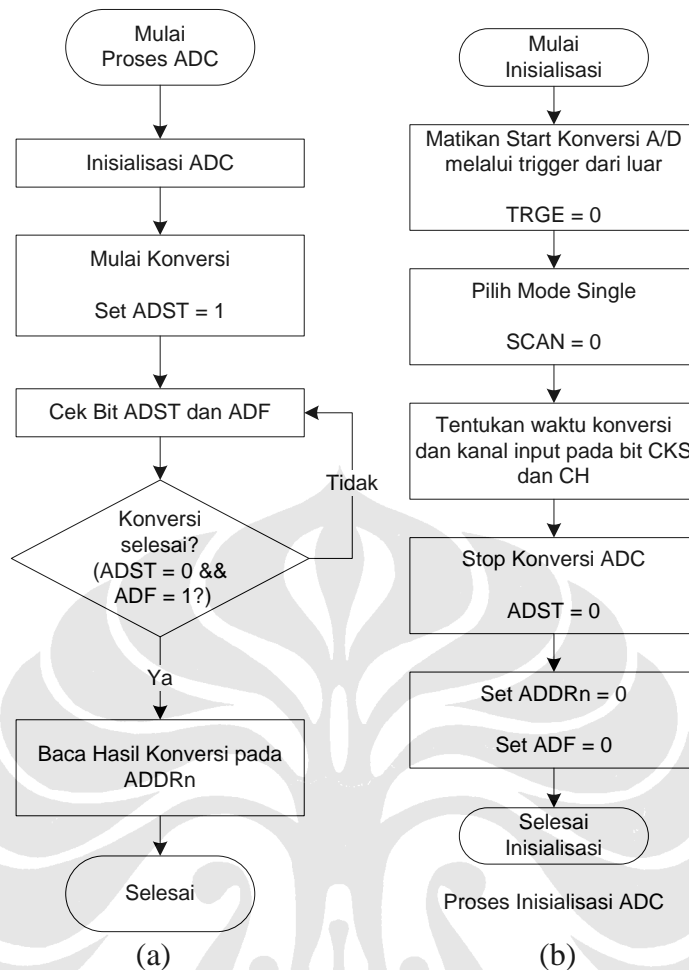
Bit	Nama	Singkatan	R/W	Fungsi
7	Trigger enable	TRGE	R/W	mengaktifkan atau menonaktifkan trigger eksternal konversi A/D

#### 2.2.2.3 Operasi Konversi A/D

Terdapat dua buah mode konversi A/D: *single mode* dan *scan mode*. Namun pada pembahasan skripsi ini hanya ditekankan pada *single mode* saja karena masing-masing mikrokontroler pada skripsi ini hanya melakukan konversi analog ke digital pada satu kanal saja.

Konversi A/D dimulai ketika bit ADST diset menjadi 1 oleh perangkat lunak atau masukan trigger eksternal. Bit ADST diset menjadi 1 selama proses konversi dan secara otomatis di-clear-kan menjadi 0 ketika konversi berakhir.

Ketika akhir konversi bit ADF diset menjadi 1. Jika bit ADIE juga diset menjadi 1 sebuah interrupt ADI dikeluarkan. Untuk meng-clear-kan ADF flag menjadi 0, pertama baca ADCSR kemudian tulis 0 ke ADF. Setelah bit ADST menjadi 0 dan ADF menjadi 1, maka hasil konversi A/D dapat dibaca pada register ADDRA – ADDRD. Gambar 2.3 menunjukkan diagram alir dari operasi ADC pada mode *single*.



Gambar 2.3 Diagram alir (a) operasi ADC dan (b) inisialisasi ADC pada mode *single*

#### 2.2.2.4 Pemrograman ADC dalam bentuk bahasa C

Berikut adalah contoh program penggunaan ADC Mikrokontroler H8/3052F pada mode single dalam bahasa C.

```

int nilai_ADC;

void initADCsingle(void) {
    MSTCR.BIT._AD = 0;           /* AD Operates normally */

    AD.ADCSR.BIT.ADST = 0;      /* Stop A/D Conversion */
    AD.ADCR.BIT.TRGE = 0;      /* Disable A/D conversion start by
                                external trigger input */

    AD.ADCSR.BIT.SCAN = 0;      /* Select Single mode */
    AD.ADCSR.BIT.CKS = 1;       /* Conv time = 134 states */
    AD.ADCSR.BIT.CH = 0;        /* Select Channel 0 as analog
                                input */

    AD.ADCSR.BIT.ADIE = 0;      /* Disable ADI */
    AD.ADCSR.BIT.ADF = 0;
    AD.ADDRA = 0;               /* Initialize ADDRA Register with
                                0 */
}

```



```

void main (void) {
    set_imask_ccr(1);          /* Disable all of the interrupts
                               */

    initADCsingle();
    set_imask_ccr(0);          /* Enable interrupts */
    AD.ADCSR.BIT.ADST = 1;     /*Start A/D Conversion */
    while((AD.ADCSR.BIT.ADST == 1) || (AD.ADCSR.BIT.ADF == 0));
    nilai_ADC = ((AD.ADDRA >> 8)&0xff);
}

```

## 2.2.3 16-Bit Integrated Timer Unit (ITU)

### 2.2.3.1 Gambaran umum ITU

H8/3052F mempunyai 5 kanal unit pewaktuan 16 bit yang dinamakan Integrated Timer Unit. ITU mampu memproses sampai 12 pulsa output atau 10 pulsa input. ITU mempunyai 10 *general register* (GRs, dua di setiap kanalnya) yang dapat berfungsi sebagai *output compare* atau *input capture*. Terdapat 8 jenis sumber pewaktuan di setiap kanalnya, yaitu pewaktuan internal ( $\phi$ ,  $\phi/2$ ,  $\phi/4$ ,  $\phi/8$ ) dan pewaktuan eksternal (TCLKA, TCLKB, TCLKC, TCLKD). Lima jenis operasi pada setiap kanalnya yaitu, *waveform output by compare match*, *input capture*, *counter clearing*, *synchronization*, dan mode PWM. Terdapat sebuah mode tambahan pada kanal 2 yaitu mode *Phase Counting* dan tiga buah mode tambahan pada kanal 3 dan 4, yaitu mode *Reset-synchronized PWM*, mode *Complementary PWM* dan *Buffering*. Serta ITU memiliki akses dengan kecepatan tinggi via bus internal 16 bit.

### 2.2.3.2 Register-register ITU

#### 1. Timer Start Register (TSTR)

TSTR adalah register 8 bit baca/tulis yang memulai atau menstop *timer counter* (TCNT) pada kanal 0 sampai 4. Tabel 2.6 merangkum pin-pin yang ada pada register TSTR.

Tabel 2.6 Pin-pin Register TSTR

Bit	Nama	Singkatan	R/W	Fungsi
0	Counter start 0	STR0	R/W	Bit ini memulai TCNT0
1	Counter start 1	STR1	R/W	Bit ini memulai TCNT1
2	Counter start 2	STR2	R/W	Bit ini memulai TCNT2
3	Counter start 3	STR3	R/W	Bit ini memulai TCNT3
4	Counter start 4	STR4	R/W	Bit ini memulai TCNT4

## 2. Timer Mode Register (TMDR)

TMDR adalah register 8 bit baca/tulis yang memilih mode PWM pada kanal 0 sampai 4 dan memilih mode *phase counting* pada kanal 2. Tabel 2.7 merangkum pin-pin yang ada pada register TMDR.

Tabel 2.7 Pin-pin Register TMDR

Bit	Nama	Singkatan	R/W	Fungsi
0	PWM mode 0	PWM0	R/W	memilih mode PWM untuk kanal 0
1	PWM mode 1	PWM1	R/W	memilih mode PWM untuk kanal 1
2	PWM mode 2	PWM2	R/W	memilih mode PWM untuk kanal 2
3	PWM mode 3	PWM3	R/W	memilih mode PWM untuk kanal 3
4	PWM mode 4	PWM4	R/W	memilih mode PWM untuk kanal 4
5	Flag direction	FDIR	R/W	memilih pengaturan kondisi overflow flag (OVF) pada TSR2
6	Phase counting mode flag	MDF	R/W	memilih mode phase counting untuk kanal 2

## 3. Timer Counter (TCNT)

TCNT adalah register baca/tulis 16 bit yang menghitung masukan pulsa dari sumber pewaktuannya. ITU mempunyai 5 TCNT, satu di setiap kanalnya.

TCNT dapat di-clear-kan menjadi H'0000 dengan mencocokkan nilai GRA atau GRB atau *input capture* ke GRA atau GRB pada kanal yang sama. TCNT terhubung dengan CPU oleh bus internal 16 bit dan dapat ditulis atau dibaca dengan akses word (16 bit) atau byte (8 bit).

## 4. General Register (GRA, GRB)

General register adalah register baca/tulis 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* nilainya akan dibandingkan dengan nilai TCNT. Ketika keduanya cocok, *flag* IMFA atau IMFB menjadi 1 pada TSR.

Ketika berfungsi sebagai register *input capture*, perubahan naik, turun atau gabungan keduanya terdeteksi pada sinyal *input capture* eksternal maka nilai TCNT akan disimpan pada *general register*, bersamaan dengan *flag* IMFA atau IMFB menjadi 1 pada TSR.

*General register* terhubung dengan CPU oleh *bus* internal 16 bit dan dapat ditulis atau dibaca dengan akses *word* (16 bit) atau *byte* (8 bit).

*General register* diinisialisasi menjadi fungsi *output compare* (tanpa sinyal output) oleh reset ketika mode *standby* .

#### 5. Timer Control Register (TCR)

TCR adalah register baca/tulis 8 bit yang memilih sumber pewaktuan, memilih bentuk perubahan sinyal dari sumber pewaktuan eksternal, dan memilih cara peng-clear-an counter. ITU mempunyai lima TCR, satu di setiap kanalnya. Tabel 2.8 merangkum pin-pin yang ada pada register TCR.

Tabel 2.8 Pin-pin Register 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	

#### 6. Timer I/O Control Register (TIOR)

TIOR adalah register baca/tulis 8 bit yang memilih fungsi *output compare* atau *input capture* untuk GRA atau GRB, dan menspesifikasikan fungsi pin TIOCA dan TIOCB. Jika fungsi *output compare* yang dipilih, TIOR juga memilih tipe dari output. Jika fungsi *input capture* yang dipilih, TIOR juga memilih bentuk perubahan sinyal dari sinyal *input capture*. ITU mempunyai lima TIOR, satu di setiap kanalnya. Tabel 2.9 merangkum pin-pin yang ada pada register TIOR.

Tabel 2.9 Pin-pin 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	

## 7. Timer Status Register (TSR)

TSR adalah register baca/tulis 8 bit yang mempunyai flag yang mengindikasikan TCNT *overflow* (dari H'FFFF ke H'0000) atau *underflow* (dari H'0000 ke H'FFFF) dan GRA atau GRB sebagai *output compare* atau *input capture*. Flag ini berfungsi sebagai sumber interupsi dan membangkitkan interupsi CPU jika diaktifkan oleh bit yang berhubungan di TIER. ITU mempunyai lima TSR, satu di setiap kanalnya. Tabel 2.10 merangkum pin-pin yang ada pada register TSR.

Tabel 2.10 Pin-pin Register TSR

Bit	Nama	Singkatan	R/W	Fungsi
0	Input capture/compare match flag A	IMFA	(R/W)*	status flag yang menandakan GRB <i>compare match</i> atau input <i>capture</i>
1	Input capture/compare match flag B	IMFB	(R/W)*	status flag yang menandakan GRA <i>compare match</i> atau input <i>capture</i>
2	Overflow flag	OVF	(R/W)*	status flag menandakan <i>overflow</i> atau <i>underflow</i>

## 8. Timer Interrupt Enable Register (TIER)

TIER adalah register baca/tulis 8 bit yang mengaktifkan atau menonaktifkan permintaan interupsi *overflow* dan permintaan interupsi *output compare* dan *input capture* dari general register. ITU mempunyai lima TIER, satu di setiap kanalnya. Tabel 2.11 merangkum pin-pin yang ada pada register TIER.

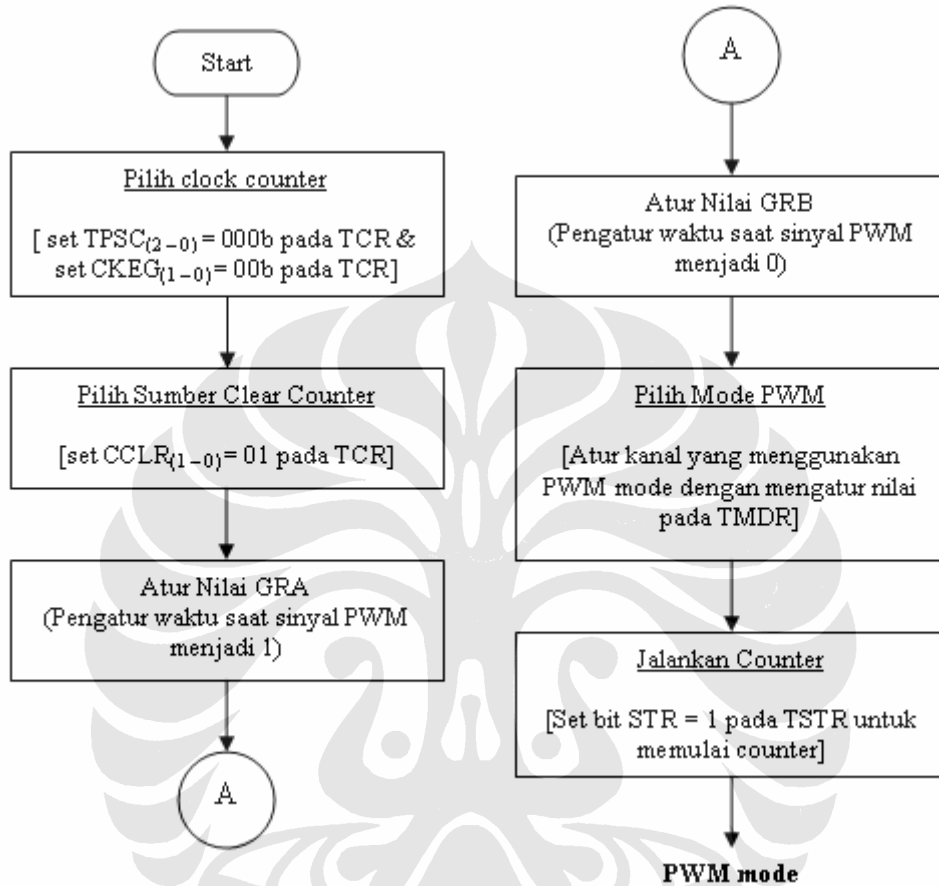
Tabel 2.11 Pin-pin Register TIER

Bit	Nama	Singkatan	R/W	Fungsi
0	Input capture/compare match interrupt enable A	IMIEA	R/W	mengaktifkan atau menonaktifkan interupsi IMFA
1	Input capture/compare match interrupt enable B	IMIEB	R/W	mengaktifkan atau menonaktifkan interupsi IMFB
2	Overflow interrupt enable	OVIE	R/W	mengaktifkan atau menonaktifkan interupsi OVF

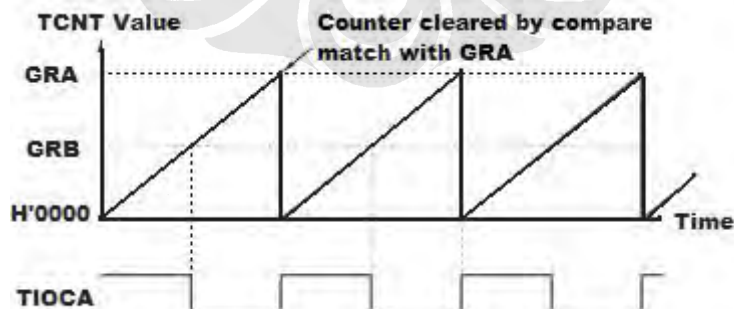
### 2.2.3.3 Operasi mode PWM

Dalam mode PWM, GRA dan GRB bekerja secara berpasangan dan menghasilkan bentuk gelombang PWM pada pin TIOCA. Saat *compare match* dengan GRA output PWM berubah menjadi 1. Saat *compare match* dengan GRB

output PWM berubah menjadi 0. Perbandingan nilai GRB dan GRA adalah *duty cycle* dari gelombang PWM yang dihasilkan. Mode PWM dapat dipilih pada semua kanal (0 sampai 4). Berikut adalah contoh prosedur dan contoh operasi dari pengaturan mode PWM:



Gambar 2.4 Contoh prosedur mode PWM



Gambar 2.5 Contoh operasi mode PWM

### 2.2.3.4 Pemrograman ITU dalam bentuk bahasa C

Berikut adalah contoh program penggunaan ITU Mikrokontroler H8/3052F pada mode PWM dalam bahasa C.

```
void initPWMSingle(void) {
    MSTCR.BIT._ITU = 0;      /* ITU operates normally */

    ITU.TOCR.BIT.XTGD = 1;  /* Disable external trigerring */
    ITU.TSTR.BIT.STR2 = 0;  /* Stop Counting TCNT in channel 2 */
    ITU2.TCR.BIT.TPSC = 3;  /*Select internal clock =
        (25Mhz/8) = 3.125 MHz */
    ITU2.TCR.BIT.CCLR = 1;  /* Clear TCNT by GRA output
        compare match */
    ITU2.GRA = 0xF424;      /* with pulse cycle approx 15 ms */
    ITU2.GRB = 3700;       /*Duty cycle approx 0.88ms at 0 deg
        */
    ITU2.TIOR.BIT.IOB = 1;  /* Output = 0 at GRB compare match */
    ITU2.TIOR.BIT.IOA = 2;  /* Output = 1 at GRA compare match */
    ITU.TMDR.BIT.PWM2 = 1;  /*Select channel 2 in PWM mode */
}

void main (void) {
    set_imask_ccr(1);      /* Disable all of the interrupts
        */
    initADCsingle();
    set_imask_ccr(0);      /* Enable interrupts */
    ITU.TSTR.BIT.STR2 = 1; /*Start counting TCNT2*/
}
}
```

## 2.2.4 Pin Input/output

### 2.2.4.1 Gambaran Umum I/O

H8/3052F mempunyai 10 port masukan/keluaran (port 1, 2, 3, 4, 5, 6, 7, 8, 9, A, dan B) dan satu port masukan (port 7).

Setiap port mempunyai sebuah data direction register (DDR) untuk pemilihan masukan atau keluaran, dan sebuah data register (DR) untuk menyimpan data keluaran. Pada pembahasan selanjutnya hanya mengarah pada port 3.

### 2.2.4.2 Port 3

Port 3 adalah port masukan/keluaran dengan konfigurasi pin yang ditunjukkan pada gambar 2.6. Port 3 adalah bus data dalam mode 1 sampai 6 (expanded mode) dan sebuah port masukan/keluaran umum pada mode 7 (single-chip mode).

		Port 3 pins	Modes 1 to 6	Mode 7
Port 3	↔	P3 <sub>7</sub> /D <sub>15</sub>	D <sub>15</sub> (input/output)	P3 <sub>7</sub> (input/output)
	↔	P3 <sub>6</sub> /D <sub>14</sub>	D <sub>14</sub> (input/output)	P3 <sub>6</sub> (input/output)
	↔	P3 <sub>5</sub> /D <sub>13</sub>	D <sub>13</sub> (input/output)	P3 <sub>5</sub> (input/output)
	↔	P3 <sub>4</sub> /D <sub>12</sub>	D <sub>12</sub> (input/output)	P3 <sub>4</sub> (input/output)
	↔	P3 <sub>3</sub> /D <sub>11</sub>	D <sub>11</sub> (input/output)	P3 <sub>3</sub> (input/output)
	↔	P3 <sub>2</sub> /D <sub>10</sub>	D <sub>10</sub> (input/output)	P3 <sub>2</sub> (input/output)
	↔	P3 <sub>1</sub> /D <sub>9</sub>	D <sub>9</sub> (input/output)	P3 <sub>1</sub> (input/output)
	↔	P3 <sub>0</sub> /D <sub>8</sub>	D <sub>8</sub> (input/output)	P3 <sub>0</sub> (input/output)

Gambar 2.6 Konfigurasi pin port 3

### 2.2.4.3 Konfigurasi Register

#### 1. Port 3 Data Direction Register (P3DDR)

P3DDR adalah register tulis 8 bit yang dapat memilih berfungsi sebagai pin masukan atau pin keluaran untuk setiap pin pada port 3. Struktur dari P3DDR dapat dilihat pada gambar 2.7.

Bit	7	6	5	4	3	2	1	0
	P3 <sub>7</sub> DDR	P3 <sub>6</sub> DDR	P3 <sub>5</sub> DDR	P3 <sub>4</sub> DDR	P3 <sub>3</sub> DDR	P3 <sub>2</sub> DDR	P3 <sub>1</sub> DDR	P3 <sub>0</sub> DDR
IV	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

**Port 3 data direction 7 to 0**  
Memilih fungsi masukan atau keluaran untuk pin port 3

Gambar 2.7 Struktur P3DDR

- Mode 1 sampai 6 (Expanded Modes)

Port 3 berfungsi sebagai bus data. P3DDR diabaikan.

- Mode 7 (Single-Chip Mode)

Port 3 berfungsi sebagai port masukan/keluaran. Sebuah pin di port 3 menjadi sebuah port keluaran jika bit P3DDR yang bersangkutan diset menjadi 1, dan menjadi port masukan jika bit ini di-clear-kan menjadi 0.

## 2. Port 3 Data Register (P3DR)

P3DR adalah register baca tulis 8 bit yang menyimpan data keluaran untuk pin P3. Struktur dari P3DR dapat dilihat pada gambar 2.8.

Bit	7	6	5	4	3	2	1	0
	P3 <sub>7</sub>	P3 <sub>6</sub>	P3 <sub>5</sub>	P3 <sub>4</sub>	P3 <sub>3</sub>	P3 <sub>2</sub>	P3 <sub>1</sub>	P3 <sub>0</sub>
IV	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
<b>Port 3 data 7 to 0</b> Menyimpan data untuk pin port 3								

Gambar 2.8 Struktur P3DR

### 2.2.5 Serial Communication Interface (SCI)

Mikrokontroler H8/3052F mempunyai dua kanal antarmuka komunikasi serial. Antarmuka komunikasi serial (SCI) pada mikrokontroler H8/3052F dapat berkomunikasi dengan mode sinkron dan asinkron. Komunikasi serial pada mikrokontroler H8/3052F berfungsi untuk pengiriman data antar mikrokontroler dan dari mikrokontroler ke PC.

#### 2.2.5.1 Konfigurasi Pin dan Register

Antarmuka komunikasi serial (SCI) pada mikrokontroler H8/3052F memiliki konfigurasi pin untuk tiap-tiap kanal seperti yang ditunjukkan tabel 2.12

Tabel 2.12 Pin-pin SCI

Kanal	Nama Pin	Abbreviation	I/O	Fungsi
0	pin pewaktuan serial	SCK0	Input/Output	Input/output pewaktuan SCI0
	pin penerimaan data	RxD0	Input/Output	Input penerimaan data SCI0
	pin pengiriman data	TxD0	Output	Output pengiriman data SCI0
1	pin pewaktuan serial	SCK1	Input/Output	Input/output pewaktuan SCI1
	pin penerimaan data	RxD1	Input/Output	Input penerimaan data SCI1
	pin pengiriman data	TxD1	Output	Output pengiriman data SCI1

Terdapat delapan jenis register pada SCI, register-register internal pada SCI berfungsi dalam pemilihan mode, format data, bit rate, dan pengontrolan bagian pengiriman dan penerimaan. Berikut adalah deskripsi dari kedelapan register tersebut :



### 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. 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 ia 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.

### 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.

### 4. Register Data Pengirim (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 ia 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.

### 5. Register Mode Serial (SMR)

SMR merupakan register 8-bit yang berfungsi untuk menentukan format komunikasi serial pada SCI dan memilih sumber pewaktuan untuk menghasilkan baud rate. Tabel 2.13 merangkum pin-pin yang ada pada register SMR.

Tabel 2.13 Pin-pin Register SMR

Bit	Nama	Singkatan	R/W	Fungsi
0	Clock select 0	CKS0	R/W	Memilih sumber pewaktuan dari penghasil baud rate
1	Clock select 0	CKS1	R/W	
2	Multiprocessor mode	MP	R/W	memilih fungsi multiprosesor
3	Stop bit length	STOP	R/W	memilih panjang bit stop
4	Parity mode	O/E	R/W	memilih parity genap atau ganjil
5	Parity enable	PE	R/W	mengaktifkan apakah parity ditambahkan
6	Character length	CHR	R/W	memilih panjang karakter dalam mode asinkron
7	Communication mode	C/A	R/W	memilih mode asinkron atau sinkron

### 6. Register Kontrol Serial (SCR)

SCR merupakan register 8-bit yang berperan dalam pengaktifan pengirim dan penerima SCI, pewaktuan keluaran pada mode asinkron, interupsi, dan sumber pewaktuan transmisi/penerimaan. Tabel 2.14 merangkum pin-pin yang ada pada register SCR.

Tabel 2.14 Pin-pin Register SCR

Bit	Nama	Singkatan	R/W	Fungsi
0	Clock enable 0	CKE0	R/W	memilih sumber clock SCI
1	Clock enable 1	CKE1	R/W	
2	Transmitte- end interrupt enable	TEIE	R/W	mengaktifkan atau menonaktifkan interrupt akhir pengiriman
3	Multiprocessor interrupt enable	MPIE	R/W	mengaktifkan atau menonaktifkan interrupt multiprosesor
4	Receive enable	RE	R/W	mengaktifkan atau menonaktifkan penerima
5	Transmitte enable	TEIE	R/W	mengaktifkan atau menonaktifkan pengirim
6	Receive interrupt enable	RIE	R/W	mengaktifkan atau menonaktifkan interrupt data diterima penuh dan interrupt error penerimaan
7	Transmitte interrupt enable	TIE	R/W	mengaktifkan atau menonaktifkan interrupt data dikirim kosong

### 7. Register Status Serial (SSR)

SSR merupakan register 8-bit yang mengandung nilai bit multiprosesor, dan status flag-flag yang mengindikasikan status operasi SCI. Tabel 2.15 merangkum pin-pin yang ada pada register SSR.

Tabel 2.15 Pin-pin Register SSR

Bit	Nama	Singkatan	R/W	Fungsi
0	Multiprocessor bit transfer	MPBT	R/W	nilai dari bit multiprosesor yang dikirimkan
1	Multiprocessor bit	MPB	R/W	menyimpan nilai bit multiprosesor pada data yang diterima
2	Transmit END	TEND	R/W	mengindikasikan akhir dari transmisi
3	Parity error	PER	R/W	mengindikasikan adanya error penerimaan parity
4	Framing error	FER	R/W	mengindikasikan adanya error framing
5	Overrun error	ORER	R/W	mengindikasikan adanya error overrun
6	Receive data register full	RDRF	R/W	mengindikasikan data telah diterima dan disimpan dalam RDR
7	transmit data register empty	TDRE	R/W	mengindikasikan data telah ditransfer dari TDR ke TSR

### 8. Register Bit Rate (BRR)

BRR merupakan register 8-bit yang bersama-sama dengan bit CKS1 dan CKS0 pada SMR bertugas untuk memilih sumber pewaktuan 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 persamaan 2.1 berikut:

$$N = \frac{\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1 \dots\dots\dots(2.1)$$

Dimana:

B = Bit rate (bit/s)

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

$\phi$  = Frekuensi sistem clock (MHz)

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

(Nilai dari n dapat dilihat berdasarkan tabel 2.16)

Tabel 2.16 Pemilihan Nilai n Berdasarkan Bit CKS1 dan CKS0

n	Sumber pewaktuan	Pengaturan SMR	
		CKS1	CKS0
0	$\phi$	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 2.2 berikut:

$$Error(\%) = \left\{ \frac{\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100 \dots\dots\dots(2.2)$$

Pada komunikasi serial yang dilakukan, digunakan sumber clock  $\phi$  (25MHz) dan bit rate 9600bps. Sehingga diperoleh nilai  $n = 0$  dan  $B = 9600$ , maka nilai setting BRR (N) dapat berdasarkan perhitungan:

$$N = \frac{25}{64 \times 2^{2(0)} \times 9600} \times 10^6 - 1 = 80,38 \cong 80 = 0x50$$

### 2.2.5.2 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 2.9 diperlihatkan contoh format komunikasi dan pengaturannya pada SMR dengan 8-bit data ditambah parity dan dua bit stop.

SMR Settings				Serial Communication Format and Frame Length											
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	1	0	1	S	8-bit data								P	STOP	STOP

Gambar 2.9 Format komunikasi serial dengan 8-bit data dengan parity dan 2 bit stop

Proses transmisi dan penerimaan pada SCI bersifat independen, sehingga memungkinkan komunikasi secara *full duplex* (mengirim dan menerima data serial dalam satu waktu bersamaan) dengan device lain. Selain itu baik pengirim maupun penerima pada SCI memiliki *buffer* sehingga memungkinkan proses penerimaan dan pengiriman data saat melakukan proses terhadap data lain.

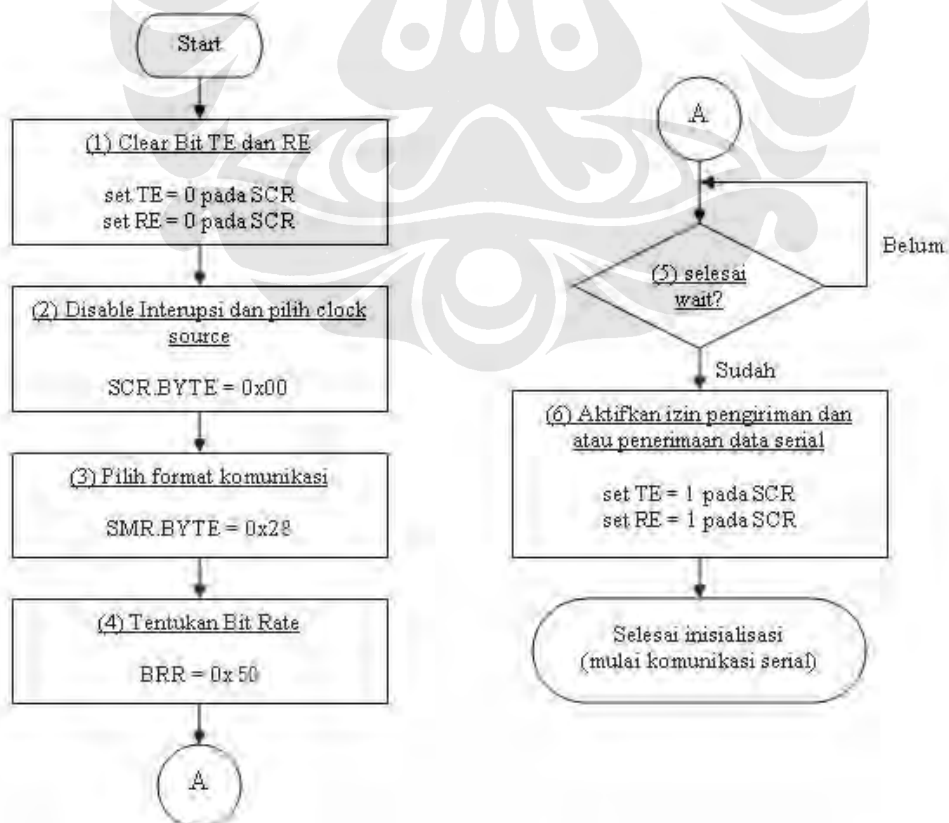
Pewaktuan pada SCI dapat dilakukan melalui dua pilihan yaitu secara internal dan eksternal yang pemilihannya berdasarkan konfigurasi bit  $C/\bar{A}$  pada SMR dan bit CKE1 dan CKE0 pada SCR. Pewaktuan secara internal dilakukan berdasarkan pewaktuan yang dihasilkan oleh baud rate generator yang terdapat 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 yaitu:

## 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.



Gambar 2.10 Flowchart proses inisialisasi serial pada operasi asinkron

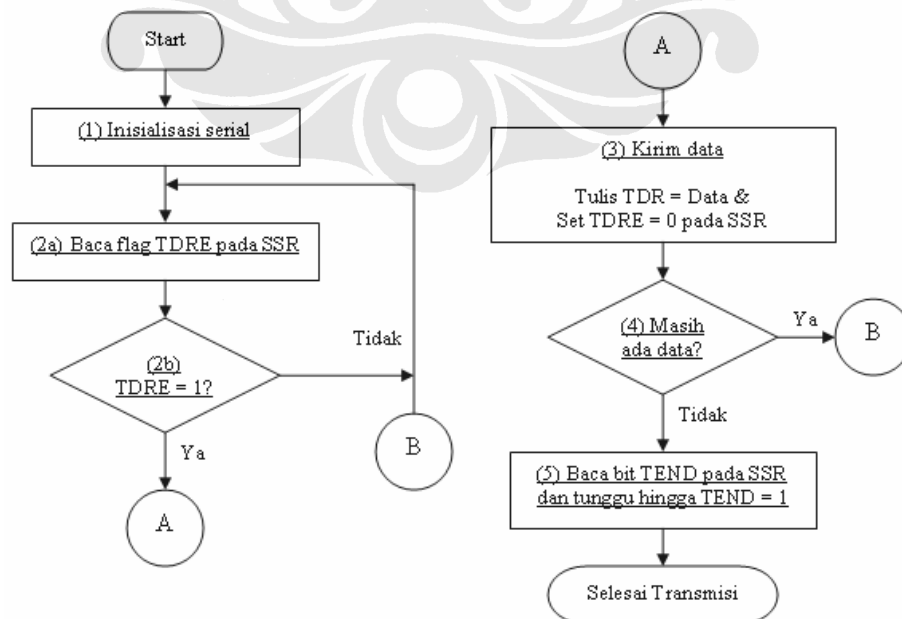
Pada gambar 2.10 ditunjukkan flowchart proses inisialisasi serial pada SCI untuk operasi asinkron. Format data komunikasi yang dipilih adalah 8-bit data dengan parity genap dan 2 bit STOP .

## 2. Transmisi Data Serial pada SCI

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

1. Melakukan proses inisialisasi 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 2.11 ditunjukkan flowchart proses transmisi data secara serial.



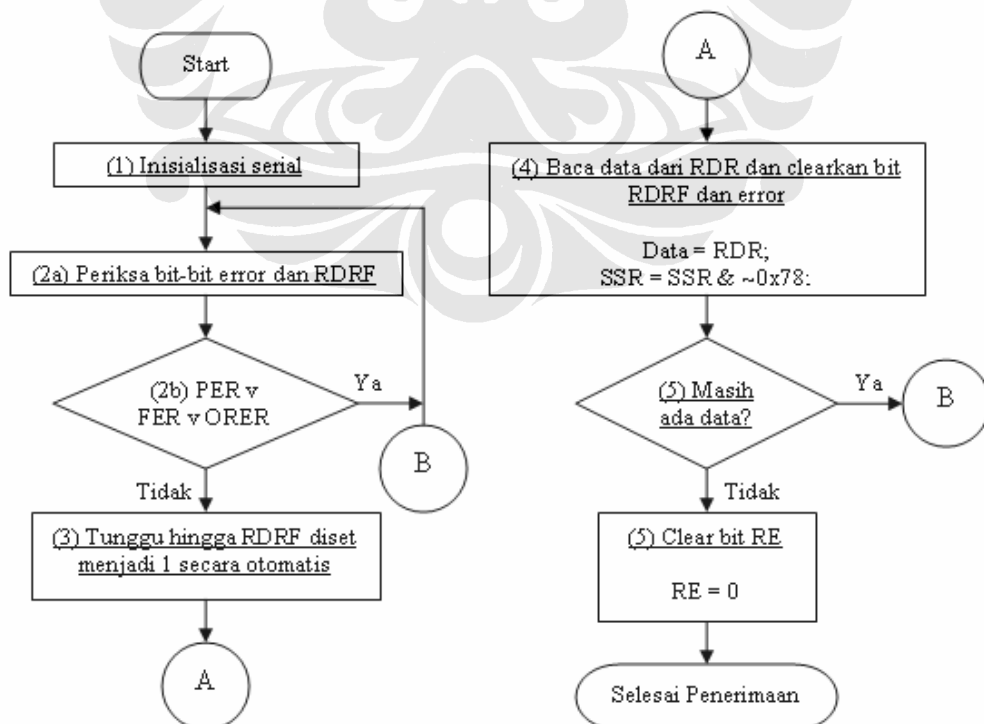
Gambar 2.11 Flowchart proses transmisi data serial pada operasi asinkron

### 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 (*receive enable*). 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 pada 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 2.12 ditunjukkan flowchart proses penerimaan data secara serial.



Gambar 2.12 Flowchart proses penerimaan data serial pada operasi asinkron

### 2.2.5.3 Pemrograman SCI dalam bentuk bahasa C

Berikut adalah contoh program penggunaan SCI Mikrokontroler H8/3052F dalam mengirimkan dan menerima data antar mikrokontroler dalam bahasa C.

```
int A;
int B = 10;

void initSCI(void) {

    MSTR.BIT._SCIO = 0;          /* Enable SCIO Module */

    SCIO.SCR.BYTE = 0x00;       /* Disable SCI Interrupts */
    SCIO.SMR.BYTE = 0x28;       /* Comm. format : Async, 8-bit
                                data, Even Parity added, 2 stop
                                bits */
                                /* Disable MP, On chip clock
                                source = 25 Mhz */
    SCIO.BRR = 0x50;           /* Baud rate initialized to
                                9600bps */

    SCIO.SCR.BIT.TE = 1;       /* Transmission Enabled */
    SCIO.SCR.BIT.RE = 1;       /* Reception Enabled */
}

void sci_write(unsigned int tulis ) {
    while(!(SCIO.SSR.BYTE & 0x80));
                                /* while transmit register empty,
                                wait */

    SCIO.TDR = tulis;          /* Transfer data into TDR
                                register */
    SCIO.SSR.BIT.TDRE = 0;
}

unsigned int sci_read( void ){
    int baca;
    while( !(SCIO.SSR.BYTE & 0x78) );
    baca = SCIO.RDR;
    SCIO.SSR.BYTE = SCIO.SSR.BYTE & ~0x78;
    return baca;
}

void main (void) {
    set_imask_ccr(1);          /* Disable all of the interrupts
                                */
    initSCI();
    set_imask_ccr(0);         /* Enable interrupts */
    A = sci_read();           /* transfer A */
    sci_write(B);             /* receive B */
}
```



## **BAB III**

### **PERANCANGAN ROBOT BILATERAL**

Manipulator untuk pengendalian bilateral yang dibuat berupa dua buah lengan dengan satu DOF. Gambar rancangan robot bilateral yang dibuat dapat dilihat pada gambar 1.2.

Komunikasi antar kedua mikrokontroler dilakukan dengan menggunakan kabel data serial. Pengiriman data dari mikrokontroler *master* ke PC juga dilakukan secara serial. PC yang digunakan sebagai penerima data, menggunakan Linux Redhat 9 sebagai *Operating System*-nya. Setelah data didapat oleh PC, data disimpan dan selanjutnya gerakan manipulator master dan slave akan digambarkan dalam bentuk 3D dengan software Open GL.

Perancangan robot bilateral dibagi ke dalam dua hal yaitu perangkat lunak dan perangkat keras. Serta dilakukan beberapa penyesuaian, seperti kalibrasi nilai ADC dan GRB ketika sudut  $0^\circ$  dan  $150^\circ$ .

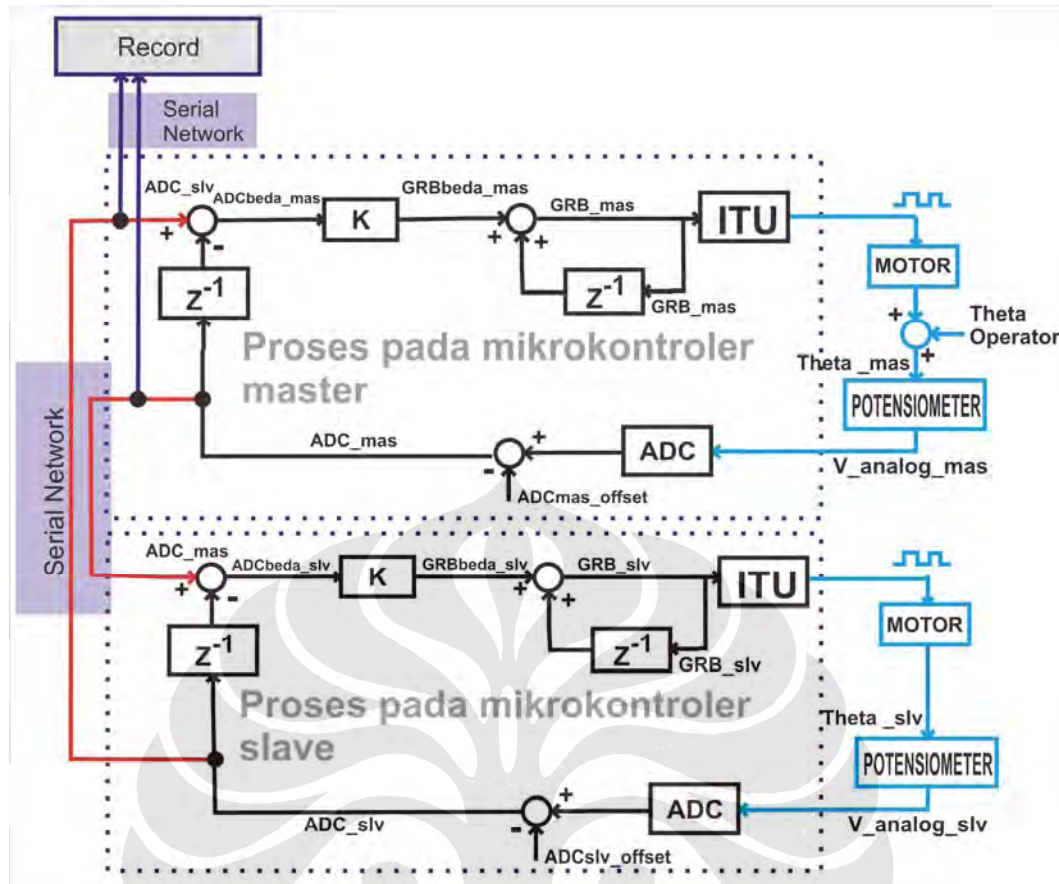
Perancangan perangkat lunak terbagi lagi menjadi perancangan blok diagram, algoritma pemrograman dengan menggunakan mikrokontroler H8/3052F dan algoritma penerimaan data di PC. Sedangkan perancangan perangkat keras terbagi menjadi perancangan setiap manipulatornya dan perangkat keras komunikasi serial yang menghubungkan kedua manipulator dan PC. Untuk mengambil data kecepatan motor digunakan NI DAQ PCI card yang terhubung dengan PC lain.

#### **3.1 PERANCANGAN PERANGKAT LUNAK**

Dalam merancang robot bilateral, blok diagram dan algoritma bilateral harus dirancang terlebih dahulu.

##### **3.1.1 Blok diagram pengendalian bilateral**

Berikut adalah blok diagram dari pengendalian bilateral yang akan dibuat.



Gambar 3.1 Blok diagram pengendalian bilateral yang dibuat

Di mana referensi sudut *master* adalah sudut *slave* sekarang dan feedback-nya adalah sudut *master* sebelumnya. Begitu juga dengan *slave*, referensi sudut *slave* adalah sudut *master* sekarang dan feedback-nya adalah sudut *slave* sebelumnya. Nilai beda sudut dari pengurangan referensi dan *feedback* (ADCbeda) akan dikonversi ke dalam bentuk beda GRB (GRBbeda). Beda GRB ini akan ditambahkan dengan nilai GRB sebelumnya untuk menjadi nilai GRB sekarang. Keluaran TIOCA pada mikrokontroler dalam bentuk PWM dimasukkan ke motor untuk menggerakkannya ke sudut tertentu. Gerakan motor ini akan menggerakkan potensiometer yang menyebabkan tegangan keluaran potensiometer pun berubah. Tegangan ini dikonversi menjadi bentuk digital oleh konverter A/D yang kemudian dikirim ke mikrokontroler lawannya sebagai nilai referensi. Nilai ADC offset adalah nilai ADC ketika sudut motor  $0^\circ$ . Sebelum nilai ADC saling dikirimkan, terlebih dahulu dikurangi dengan nilai ADC offset masing-masing. Nilai ADC *master* dan ADC *slave* akan dikirimkan ke PC untuk disimpan datanya.

### 3.1.2 Algoritma pengendalian bilateral

Dari blok diagram pengendalian bilateral dapat dibuat algoritma pengendalian bilateral yang menggunakan mikrokontroler H8/3052F sebagai pengendalinya. Algoritma pengendalian bilateral dapat dilihat pada lampiran 1.

Pemrograman dibuat seperti zig-zag, dimana setelah referensi sudut *master* dikirimkan ke *slave*, *slave* akan berjalan terlebih dahulu hingga menghasilkan nilai GRB *slave* baru dan sudut *slave* baru. Setelah itu barulah nilai sudut *slave* baru dikirimkan ke *master* hingga menghasilkan nilai GRB *master* baru dan selanjutnya mengikuti looping program.

Gerakan motor dibatasi antara sudut  $0^\circ$  dan  $150^\circ$ , hal ini dibuat dengan membatasi nilai GRB antara 3700 dan 5900 untuk motor Hitech yang digunakan.

### 3.1.3 Algoritma pemrograman penerimaan data oleh PC

Bahasa pemrograman yang digunakan untuk membuat program komunikasi serial pada komputer yaitu bahasa pemrograman C.

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.

Untuk membaca data *byte* dari port serial digunakan perintah `read()` yang sebelumnya port tersebut harus diinisialisasi dengan mengkonfigurasi port berdasarkan baud rate, flow control, dan format data yang akan dikomunikasikan. Prototipe fungsi dari perintah `read` yaitu:

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

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

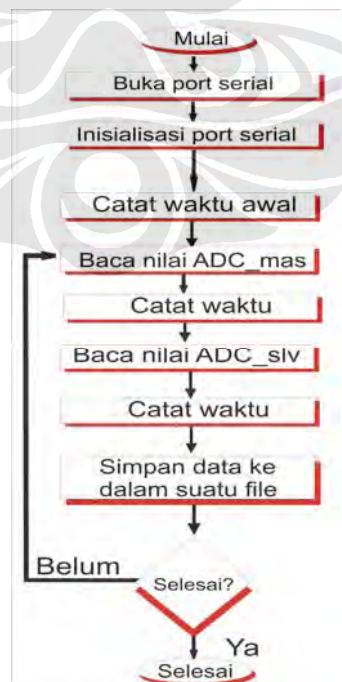
```
close(fd);
```

### 3.1.3.1 Inisialisasi komunikasi serial di PC

Inisialisasi dilakukan untuk mengkonfigurasi format data, bit rate, parity, dan konfigurasi lainnya. Format data diatur dengan 1 byte (8 bit) data ditambah 3 bit tambahan (1 bit start, 1 bit parity, 1 bit stop) dengan menggunakan bit rate yang didukung oleh mikrokontroler H8/3052F sebesar 9600bps.

### 3.1.3.2 Algoritma komunikasi serial di PC

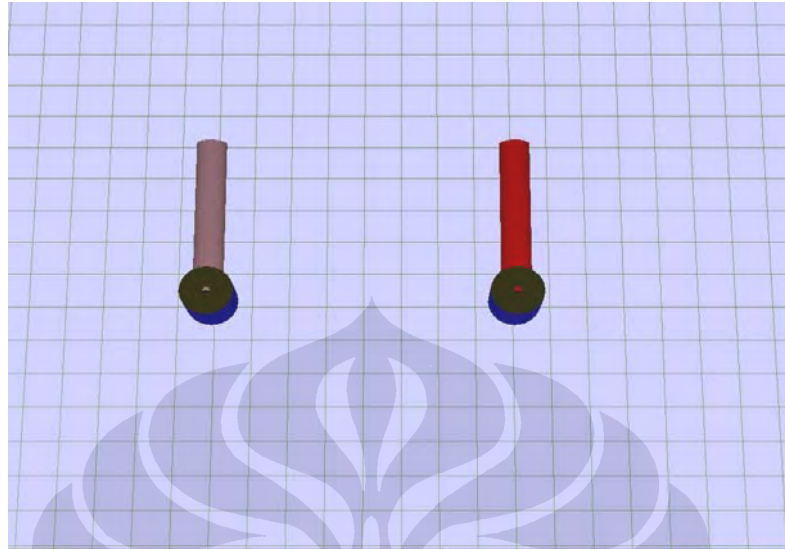
PC hanya mengambil dua buah nilai dari mikrokontroler, yaitu nilai ADC pada manipulator *master* dan nilai ADC pada manipulator *slave*. Sesaat setelah data didapat, PC akan mencatat waktu pengambilan data tersebut dengan perintah `gettimeofday()`. Data waktu dan nilai ADC selanjutnya disimpan ke dalam bentuk file. Berikut adalah algoritma komunikasi serial di PC.



Gambar 3.2 Algoritma komunikasi serial di PC

### 3.1.4 Algoritma pemrograman penggambaran 3D dengan Open GL

Berikut adalah tampilan penggambaran manipulator secara 3D dengan Open GL.

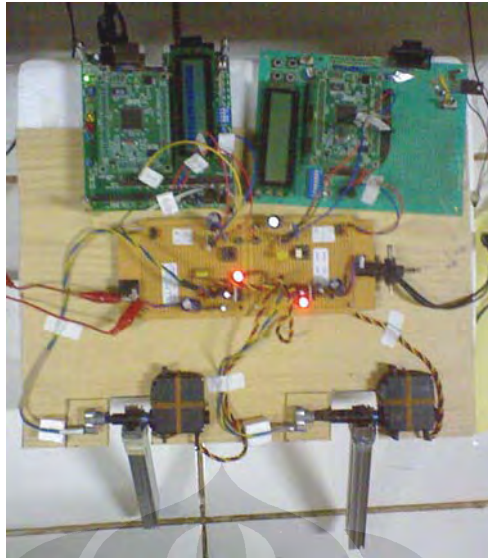


Gambar 3.3 Tampilan penggambaran manipulator secara 3D

Pemrograman ini hanya mengambil nilai ADC yang telah diterima PC dari mikrokontroler dan selanjutnya digambarkan dalam bentuk 3D.

### 3.2 PERANCANGAN PERANGKAT KERAS

Setelah algoritma dibuat, hal selanjutnya yang dilakukan adalah merancang perangkat keras robot bilateral, yaitu berupa dua buah manipulator yang setiap manipulatornya terdiri dari aktuator (motor), sensor (potensiometer), dan pengendali. Selain ketiga perangkat tersebut ada beberapa perangkat lain yang harus dibuat, yaitu rangkaian voltage regulator untuk memberikan sumber daya terpisah bagi motor, rangkaian optocoupler untuk menghasilkan sinyal PWM yang stabil serta rangkaian filter untuk mengurangi noise masukan konverter analog ke digital. Ditambah sebuah NI DAQ PCI card untuk mengambil data kecepatan motor. Gambar blok diagram rangkaian perangkat keras bilateral robot dapat dilihat pada lampiran 2. Berikut adalah gambar rangkaian jadi perangkat keras dari pengendalian bilateral yang dibuat.



Gambar 3.4 Gambar rangkaian perangkat keras pengendalian bilateral

Rangkaian perangkat keras ini disusun dengan menggunakan perangkat keras sebagai berikut:

1. Motor servo Hitech HSR 5995TG
2. Potensiometer RDC503
3. Evaluation Kit AKI-H8/3052-LAN
4. Rangkaian *voltage regulator*
5. Rangkaian optocoupler
6. Filter *low pass* konverter analog ke digital
7. Perangkat keras komunikasi serial
8. NI DAQ PCI card

### 3.2.1 Motor servo Hitech HSR 5995TG

Servo Hitech HSR 5995TG ini pada dasarnya diproduksi untuk aplikasi robot. Gearnya terbuat dari bahan titanium yang super kuat.



Gambar 3.5 Motor servo Hitech HSR 5995TG

Karakteristik servo HSR 5995TG dapat dilihat pada tabel 3.1.

Tabel 3.1 Karakteristik Servo HSR 5995TG

Karakteristik	Besar
torsi	24kg/cm (saat 6.0V) dan 30kg/cm (saat 7.4V)
kecepatan	0.15 detik/60° (saat 6.0V) dan 0.12 detik/60° (saat 7.4V)
tegangan operasi	4.8V - 7.4V
sudut operasi	180° (saat 1500µSec, ± 400µSec)
tegangan sinyal kendali	3.3V - 7.4V
konsumsi arus	300mA (saat 6.0V tanpa beban) dan 380mA (saat 7.4V tanpa beban)
	4.2A (saat 6.0V, terkunci) dan 5.2A (saat 7.2V, terkunci)
berat	62 gram

### 3.2.2 Potensiometer RDC503

Potensiometer yang digunakan ini memiliki batas maksimal hambatan sebesar 10 kΩ dan putaran yang lebih lembut dari potensiometer biasa.

Karakteristik dari Potensiometer RDC503 dapat dilihat pada tabel 3.2.

Tabel 3.2 Karakteristik Potensiometer RDC503

Karakteristik	Besar
rating	5VDC
sudut putaran maksimum	320°
torsi penggerak	2 mN.m maks
tegangan maksimum	16 VDC
batas pemutaran	1.000.000 putaran
toleransi	±2%

### 3.2.3 Evaluation Kit AKI-H8/3052-LAN

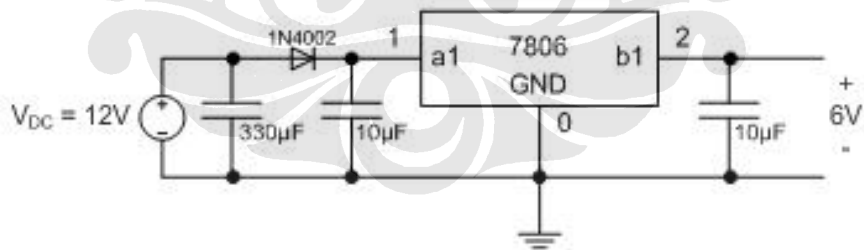
Perangkat pengendali yang digunakan adalah *evaluation kit* multifungsi untuk mikrokontroler Hitachi H8/3052F. Eval-kit ini memiliki dukungan beberapa perangkat keras tambahan, di antaranya LED 4 buah, LCD, kanal untuk servo, koneksi RS232, koneksi LAN, dan koneksi *power* USB. Berikut adalah gambar Eval-kit AKI-H8/3052-LAN.



Gambar 3.6 Gambar Eval-kit AKI-H8/3052-LAN

### 3.2.4 Perancangan rangkaian *voltage regulator*

Rangkaian *voltage regulator* dibuat agar motor dicatu oleh sumber daya lain. Tujuan penggunaan sumber daya yang terpisah ini adalah agar mikrokontroler dapat memperoleh suplai arus yang mencukupi kebutuhannya. Pada saat bergerak, motor memerlukan suplai arus yang 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. Gambar rangkaiannya adalah sebagai berikut.



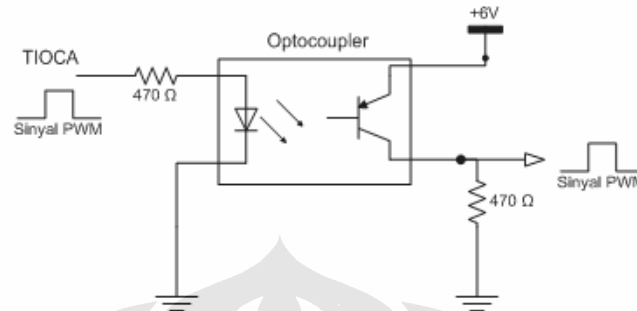
Gambar 3.7 Gambar rangkaian *voltage regulator*

*Voltage regulator* 7806 digunakan untuk menstabilkan tegangan masukan 12V menjadi tegangan 6V. Tegangan 6V ini akan menjadi tegangan masukan motor servo.



### 3.2.5 Perancangan rangkaian optocoupler

Rangkaian optocoupler dibuat agar sinyal PWM yang masuk ke motor lebih stabil karena sinyal PWM ini memiliki sumber daya yang berbeda dengan mikrokontroler. Gambar rangkaiannya adalah sebagai berikut.

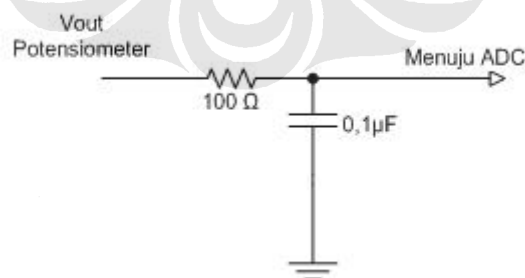


Gambar 3.8 Gambar rangkaian optocoupler

Sinyal PWM dari mikrokontroler akan digunakan sebagai masukan optocoupler TLP521, keluaran optocoupler ini akan serupa dengan sinyal PWM yang dimasukkan. Sumber VCC optocoupler ini adalah keluaran dari rangkaian voltage regulator. Sinyal PWM keluaran optocoupler ini akan menjadi masukan motor servo pada kabel sinyal.

### 3.2.6 Perancangan filter *low pass* konverter analog ke digital

Filter *low pass* digunakan untuk mengurangi derau tegangan yang terjadi pada tegangan masukan konverter dengan *cut off frequency* sebesar 15,916 kHz. Gambar rangkaiannya adalah sebagai berikut.

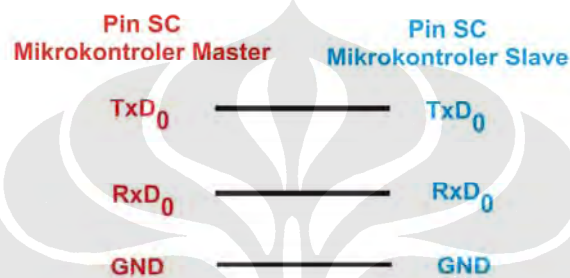


Gambar 3.9 Rangkaian filter *low pass* konverter analog ke digital

### 3.2.7 Perangkat keras komunikasi serial

#### 3.2.7.1 Komunikasi serial antar mikrokontroler

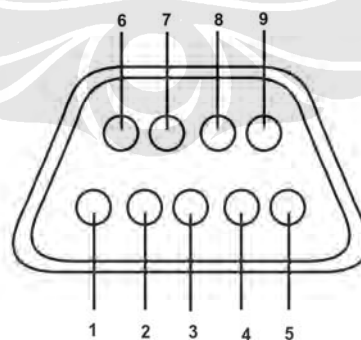
Untuk melakukan komunikasi serial antar mikrokontroler, mikrokontroler H8/3052F hanya memerlukan 3 buah pin yaitu  $TxD_0$  pada SCI *master* yang dihubungkan dengan  $RxD_0$  pada SCI *slave*,  $RxD_0$  pada SCI *master* yang dihubungkan pada  $TxD_0$  pada SCI *slave*, dan GND SCI *master* yang dihubungkan dengan GND pada SCI *slave*. Koneksi pin serial pada SCI *master* dan pin serial pada SCI *slave* ditunjukkan pada gambar 3.9.



Gambar 3.10 Koneksi pin serial SCI *master* dan SCI *slave*

#### 3.2.7.2 Komunikasi serial dengan PC

Komunikasi serial dengan PC membutuhkan konektor RS-232C DB-9. RS-232C DB-9 memiliki sembilan buah pin yang digunakan untuk komunikasi serial seperti yang ditunjukkan gambar 3.10.



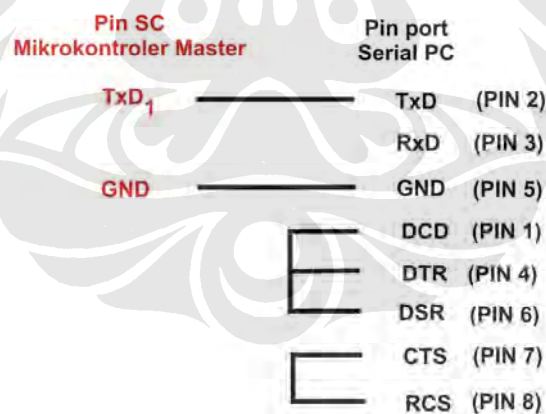
Gambar 3.11 RS-232C DB-9 konektor tampak depan

Sedangkan penjelasan pin-pin pada RS-232C ditunjukkan pada tabel 3.3.

Tabel 3.3 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

Untuk melakukan komunikasi serial dengan komputer, mikrokontroler H8/3052F hanya memerlukan 2 buah pin yaitu TxD<sub>1</sub> pada SCI yang dihubungkan dengan RxD pada sisi komputer dan GND SCI yang dihubungkan dengan GND pada sisi komputer. Pada konektor RS-232C pada 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 3.11.



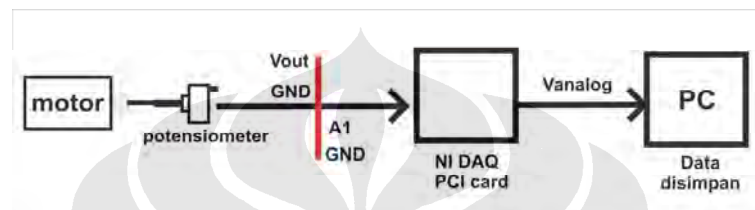
Gambar 3.12 Koneksi Kabel Pin Serial SCI dengan Port Serial Komputer

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 dihubungkan singkat. Sedangkan pin 7 dan 8 dihubungkan singkat untuk keperluan *flow control* dari komputer.

### 3.2.8 Rangkaian pengambilan data kecepatan motor

Pengambilan data kecepatan motor dilakukan dengan menghubungkan tegangan keluaran potensiometer yang mewakili sudut motor dengan pin analog input NI DAQ PCI card dan selanjutnya data disimpan pada PC. Berikut adalah blok diagram rangkaian tersebut.



Gambar 3.13 Blok diagram pengambilan data kecepatan motor

### 3.3 KALIBRASI NILAI GRB DAN ADC

Kalibrasi dilakukan dengan bantuan sebuah busur derajat yang telah diletakkan pada lengan robot. Selanjutnya besarnya GRB dicari ketika lengan robot menunjukkan sudut  $0^\circ$  dan  $150^\circ$  pada nilai GRA yang tetap. Setelah didapat nilai GRB, pin tegangan keluaran potensiometer dihubungkan dengan ADC mikrokontroler H8/3052F, dimana potensiometer terkoneksi pada rotor motor servo. Selanjutnya motor dibuat untuk menghasilkan sudut  $0^\circ$  kemudian hasil ADC-nya dilihat pada LCD mikrokontroler, begitu pula untuk sudut  $150^\circ$ . Hasilnya dicatat dan ditampilkan pada tabel berikut.

Tabel 3.4 Kalibrasi Nilai GRB dan ADC

Sudut	nilai GRB		nilai ADC	
	Master	Slave	Master	Slave
$0^\circ$	3700	3700	88	88
$150^\circ$	5900	5900	198	198
<b>beda</b>	2200	2200	110	110

## BAB IV

### PENGUJIAN DAN ANALISA

Pengujian performa robot bilateral dilakukan melalui dua pengujian, yaitu:

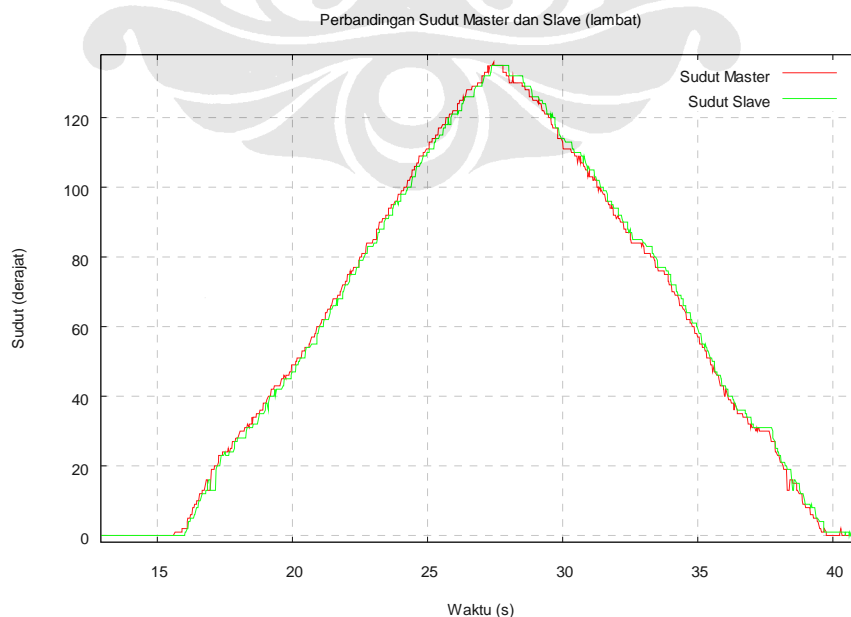
1. Pengujian respon gerakan manipulator *slave* mengikuti gerakan manipulator *master* dalam berbagai kecepatan gerak.
2. Pengujian kesesuaian fungsi manipulator *master* untuk men-*transfer* sensasi sentuhan yang dirasakan manipulator *slave* kepada operator.

#### 4.1 PENGUJIAN RESPON MANIPULATOR SLAVE

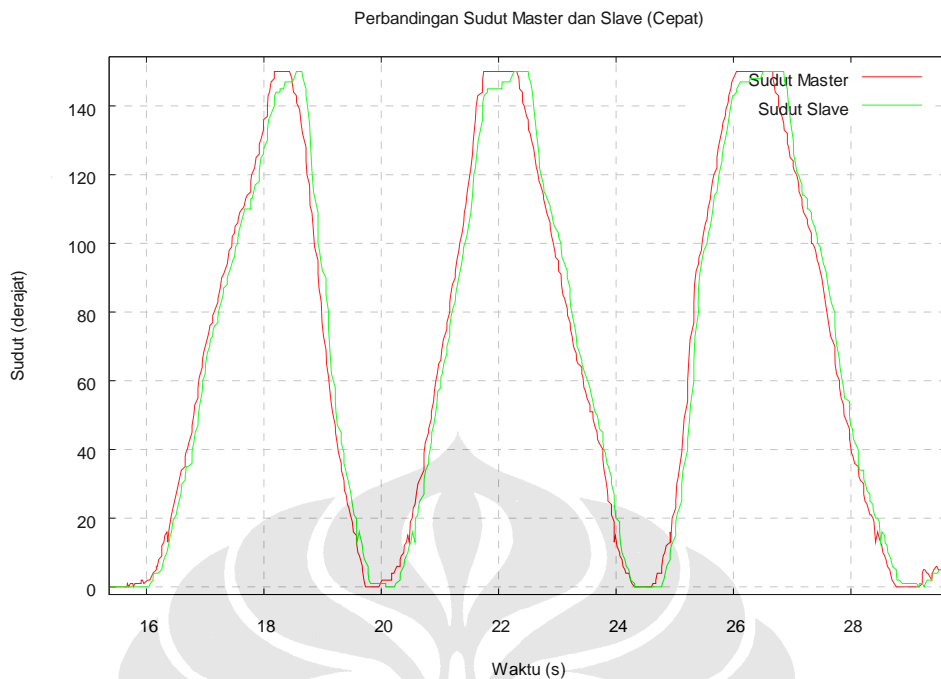
Pengujian ini dilakukan dengan cara menggerakkan manipulator *master* secara bebas dengan beberapa jenis kecepatan. Data kecepatan didapat dengan melihat data pada komputer yang terhubung dengan NI DAQ PCI card. Setelah percobaan dilakukan, data percobaan yang didapat kemudian dilihat hasilnya dan digambarkan ke dalam bentuk grafik.

#### 4.2 HASIL PENGUJIAN RESPON MANIPULATOR SLAVE

Hasil pengujian yang membandingkan sudut *master* dan sudut *slave* dalam dua jenis kecepatan dapat dilihat pada grafik 4.1 dan 4.2.



Gambar 4.1 Grafik perbandingan sudut *master* dan sudut *slave* terhadap waktu untuk tempo lambat



Gambar 4.2 Grafik perbandingan sudut *master* dan sudut *slave* terhadap waktu untuk tempo cepat

Dari grafik pengujian dengan tempo lambat (kecepatan rata-rata 11,28 deg/sec) diketahui untuk perubahan sudut *master* yang bersifat lambat, pergerakannya dapat diikuti oleh manipulator *slave* dengan cukup baik dan memiliki delay beberapa saat yang dapat ditoleransi untuk mencapai tujuannya (sudut referensi). Ketepatan sudut juga dapat terpenuhi. Pada saat awal pergerakan manipulator *master*, manipulator *slave* belum dapat mengikuti. Hal ini dikarenakan torsi awal untuk menggerakkan manipulator *slave* lebih besar dibandingkan ketika manipulator telah bergerak. Error maksimum yang terjadi pada kecepatan ini sebesar  $7^{\circ}$ .

Dari grafik pengujian dengan tempo cepat (kecepatan rata-rata 68,49 deg/sec) diketahui bahwa untuk perubahan dengan tempo cepat, manipulator *slave* juga dapat mengikuti gerakan putaran sudut *master* yang berubah-ubah secara cepat. Error maksimum yang terjadi pada kecepatan ini sebesar  $20^{\circ}$ .

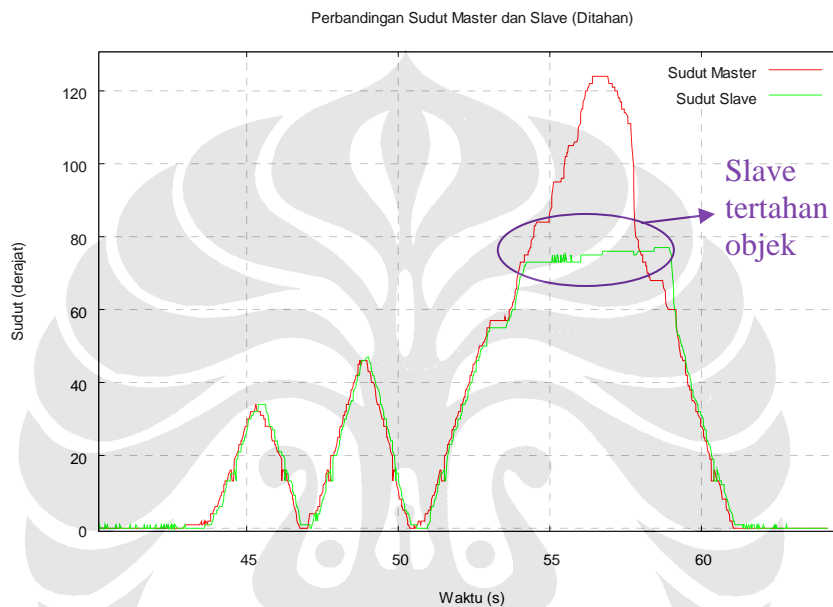
#### 4.3 PENGUJIAN KESESUAIAN FUNGSI MANIPULATOR *MASTER*

Pengujian ini dilakukan dengan cara menahan gerakan manipulator *slave* dengan menggunakan suatu objek. Kemudian manipulator *master* dicoba untuk terus digerakkan melebihi sudut manipulator *slave* yang tertahan objek. Setelah

percobaan dilakukan, data percobaan yang didapat kemudian dilihat hasilnya dan digambarkan ke dalam bentuk grafik.

#### 4.4 HASIL PENGUJIAN KESESUAIAN FUNGSI MANIPULATOR MASTER

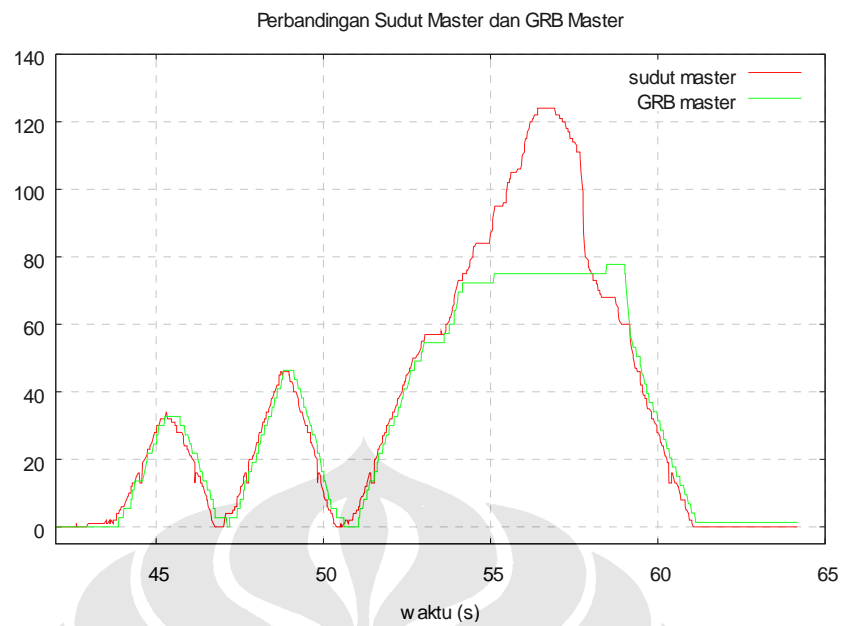
Hasil pengujian yang membandingkan sudut *master* dan sudut *slave* ketika manipulator *slave* tertahan oleh objek dapat dilihat pada gambar 4.3.



Gambar 4.3 Grafik perbandingan sudut *master* dan sudut *slave* terhadap waktu ketika manipulator *slave* tertahan oleh objek

Dari pengujian dan grafik dapat dilihat bahwa ketika manipulator *slave* tertahan oleh objek, manipulator *master* masih dapat digerakkan melebihi sudut manipulator *slave*. Namun setelah manipulator *master* berhenti digerakkan, ia akan kembali menyamai sudut *slave* dengan terdapat *delay* sesaat. Dan selanjutnya manipulator *slave* dapat kembali mengikuti gerakan manipulator *master*.

Pada gambar 4.4 dapat ditunjukkan bahwa ketika manipulator *slave* tertahan oleh objek, nilai GRB manipulator master besarnya tetap. Hal ini menunjukkan bahwa manipulator *master* seharusnya tetap pada posisi tersebut. Karena torsi operator lebih besar dari torsi yang menahan motor sehingga manipulator *master* masih dapat digerakkan.



Gambar 4.4 Grafik perbandingan sudut *master* dan GRB *master* terhadap waktu ketika manipulator *slave* tertahan oleh objek

Hal ini membuktikan bahwa torsi motor manipulator *master* terlalu kecil untuk menahan torsi operator walau dalam keadaan torsi motor maksimum.



## BAB V

### KESIMPULAN

Dari keseluruhan pembahasan dalam skripsi ini dapat disimpulkan beberapa hal, yaitu :

1. Pengendalian robot bilateral dapat direalisasikan dengan menggunakan mikrokontroler. Dimana robot bilateral ini berupa konfigurasi dua manipulator robot yang saling mempengaruhi satu sama lain dan terkoneksi melalui sebuah jaringan.
2. Pada skripsi ini, mikrokontroler digunakan sebagai unit pengendali. Kedua mikrokontroler dihubungkan dengan komunikasi serial. PC digunakan sebagai pengambil dan pengukur data, dihubungkan dengan mikrokontroler *master* melalui komunikasi serial.
3. Dalam pengendalian aktuator motor, diperlukan sumber daya terpisah agar suplai arus pada mikrokontroler tidak terganggu.
4. Pada pengujian respon manipulator *slave* dari aplikasi yang dibuat, didapat *error* posisi atau beda sudut lengan maksimum pada kecepatan 11,28 deg/sec sebesar  $7^{\circ}$  dan pada kecepatan 68,49 deg/sec sebesar  $20^{\circ}$ .
5. Torsi motor Servo Hitech HSR 5995TG tidak cukup untuk menahan torsi operator walau dalam keadaan torsi maksimum. Hal ini dibuktikan pada saat pergerakan lengan *slave* ditahan, manipulator *master* masih dapat digerakkan maju.

## DAFTAR ACUAN

- [1] Y, Uchimura. K, Ohnishi. 2006. "Real-Time Network based Robot bilateral Control System," ISARC, hal. 548.
- [2] \_\_\_\_\_. *Hitachi Single Chip Microcomputer H8/3052F-ZTAT Hardware Manual*. 2001



## DAFTAR PUSTAKA

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

\_\_\_\_\_, *H8 Family Tutorial Course*, Renesas, terakhir diakses 12 Februari 2008.

[http://resource.renesas.com/lib/eng/e\\_learnig/h8\\_300henglish/index.html](http://resource.renesas.com/lib/eng/e_learnig/h8_300henglish/index.html)

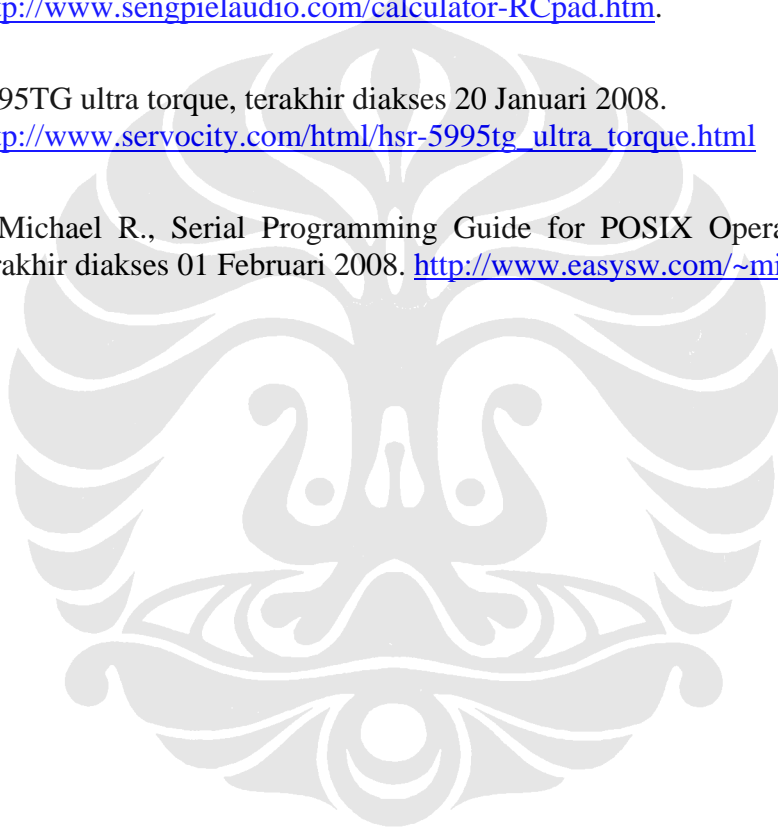
High Density Mounting Phototransistor Optically Coupled Isolators. 1997. terakhir diakses 24 Juni 2008.

<http://www.sengpielaudio.com/calculator-RCpad.htm>.

HSR 5995TG ultra torque, terakhir diakses 20 Januari 2008.

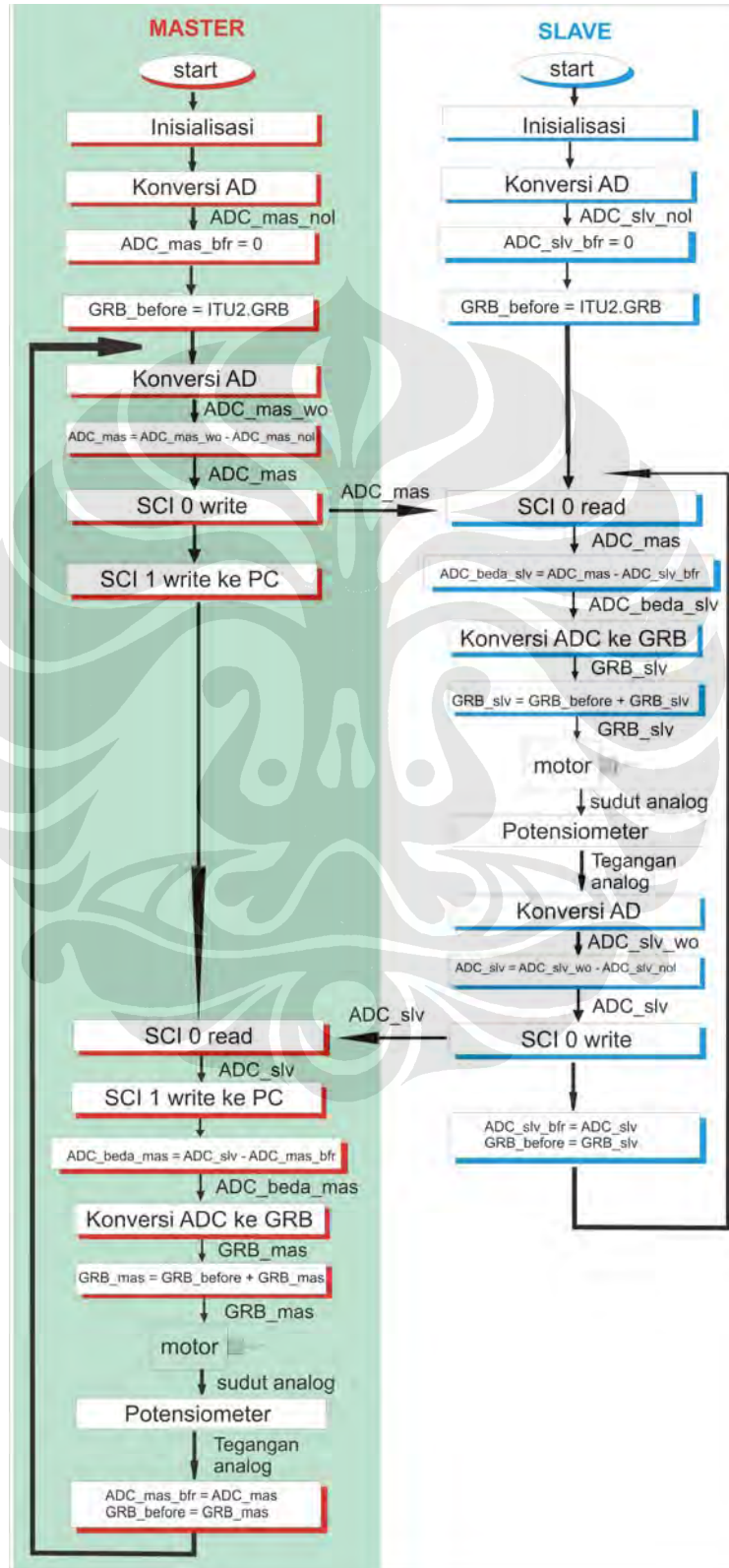
[http://www.servocity.com/html/hsr-5995tg\\_ultra\\_torque.html](http://www.servocity.com/html/hsr-5995tg_ultra_torque.html)

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



# LAMPIRAN

Lampiran 1 Algoritma Pemrograman Pengendalian Bilateral Robot



Lampiran 2 Blok Diagram Perangkat Keras Pengendalian Bilateral

