

PERANCANGAN SISTEM KENDALI ROBOT BERKAKI DUA

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Mesin**



Disusun Oleh :

Nama : Joko Sriyanto
No. Mahasiswa : 07525011

**JURUSAN TEKNIK MESIN
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2011

HALAMAN PERSEMBAHAN

Aku persembahkan untuk orang-orang yang menyayangiku, memberi support,
membantu, serta mendoakanku selama ini,

Untuk Bapak dan Ibu tercinta,

Yang slalu dan senantiasa menyayangi serta mendoakanku, berharap putranya
menjadi lebih baik, dewasa, dan memiliki kehidupan yang lebih baik,

Untuk Kakakku, dan Adik-adikku,

Yang slalu memberi motivasi, dan harapan melalui suara-suara candaan, tiap hati ini
sedang gundah atau gelisah,

Untuk Nyta Pratiwi, dan teman-temanku semua

Yang slalu membantu dan siap berkorban dalam senang maupun duka,

Dan Untuk Mereka

Yang menyimpan harapan besar dalam diriku,

Akan kulakukan yang terbaik.

HALAMAN MOTTO

“Setiap usaha dengan maksimal, tulus, dan niat yang kuat, sepantasnya mengharap hasil terbaik”.

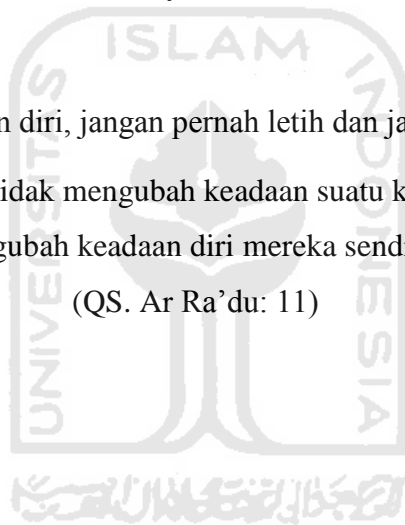
“Kadang keadaan tak sesuai dengan keinginan, rencana, maupun harapan sebelumnya,

Namun jika dihadapi, dijalani, dan diterima dengan sabar serta syukur, maka hasil akhirnya akan berasa manis”

“Jangan pernah melarikan diri, jangan pernah letih dan jangan pernah putus asa”.

”Sesungguhnya Allah tidak mengubah keadaan suatu kaum sehingga mereka mengubah keadaan diri mereka sendiri”.

(QS. Ar Ra’du: 11)



UCAPAN TERIMA KASIH



Assalamu'alaikum Warahmatullah Wabarakatuh

Alhamdulillah *rabbi'l'alamiin*, puji syukur kehadiran Allah SWT yang telah memberikan rahmat, hidayah, kesempatan, dan kemudahan dalam menjalankan amanah sehingga tugas akhir yang berjudul **“PERANCANGAN SISTEM MEKANIK ROBOT DUA KAKI”** ini dapat terselesaikan. Shalawat dan salam semoga selalu tercurah kepada Nabi Muhammad Shallallahu Alaihi wa Sallam, inspirasi akhlak dan pribadi mulia.

Tugas akhir ini merupakan syarat untuk mencapai jenjang Strata Satu (S1), pada Jurusan Teknik Mesin, Fakultas Teknologi Industri, Universitas Islam Indonesia. Masih terdapat banyak keterbatasan dalam penelitian dan penulisan tugas akhir ini, oleh karena itu penulis mohon maaf dan berharap akan ada pengembangan penelitian yang lebih baik dengan rekomendasi penelitian yang dikemukakan pada bagian akhir dari tugas akhir ini.

Penulis mengucapkan terima kasih tidak terhingga kepada pihak-pihak yang memberikan dukungan baik material maupun spiritual sehingga tugas akhir ini dapat terselesaikan, yaitu kepada:

1. Bapak Agung Nugroho Adi, ST., MT., selaku Dosen Pembimbing I dan Ketua Jurusan Teknik Mesin Fakultas Teknologi Industri Universitas Islam Indonesia.
2. Bapak Tri Setya Putra, ST., selaku Dosen Pembimbing II.
3. Bapak Slamet dan Ibu Warsiyem selaku orang tua saya yang telah mendoakan dan memberikan semangat.
4. Sukerman Endik dan Amrullah Mahardika, sebagai tim dan sahabat dalam lomba, dalam canda, dan keseharian.

5. Semua rekan-rekan di Jurusan Teknik Mesin Fakultas Teknologi Industri Universitas Islam Indonesia, khususnya angkatan 2007, serta semua pihak yang tidak dapat kami sebutkan satu persatu.

Penulis menyadari bahwa dalam penulisan laporan tugas akhir ini masih jauh dari apa yang diharapkan, maka penulis mengharapkan kritik dan saran dari para pembaca sehingga dapat berguna untuk perbaikan laporan ini di kemudian hari. Semoga penelitian ini dapat memberikan manfaat bagi kita semua.

Wassalamu'alaikum Warahmatullah Wabarakatuh



Yogyakarta, November 2011

Penyusun

Joko Sriyanto

ABSTRAK

Penelitian-penelitian tentang robotika kini semakin banyak dilakukan di Indonesia, salah satunya adalah robot Humanoid. Robot Humanoid merupakan robot yang menyerupai manusia baik secara fisik maupun beberapa fungsinya. Dari segi sistem geraknya robot Humanoid terbilang sulit dan kompleks, tak banyak pembuatan robot jenis ini dapat berjalan dengan baik. Penelitian ini bertujuan merancang sistem gerak dari robot Humanoid buatan sendiri dengan menggunakan metode praktis berupa pengujian-pengujian manual terhadap gerakan robot.

Dalam penelitian ini sedikit mengadaptasi dari prinsip inverse-kinematik yaitu mengetahui nilai posisi akhir gerakan robot. Proses pertama adalah mengkomunikasikan sistem kontroler dengan aktuator. Kemudian ditentukan langkah-langkah gerakan yang akan dilakukan oleh robot. Langkah-langkah tersebut kemudian diimplementasikan pada pengaturan tiap aktuator. Kemudian didapat nilai-nilai hasil pengujian yang menghasilkan gerakan robot yang terbaik. Hasil penelitian ini robot dapat melakukan gerakan maju, belok dan tendang dengan baik dan seimbang pada permukaan yang rata.

Kata kunci : Robot Humanoid, sistem gerak, inverse-kinematik, aktuator.

ABSTRACT

There were so many researchs in robotics technology in Indonesia, One of the most in research in Humanoid Robot. Humanoid Robot defined by a kind of robot that has an unique similarity with human in physical as well in its function . From its systematic- movement point of view, Humanoid robot still difficult and complex, so that many research in Humanoid robot resulted unsatisfying. The objective in this research was to design a movement-system of Humanoid robot “self-made” (non plug-play robot usually in company-made) using a simple method as trials in robot’s movement.

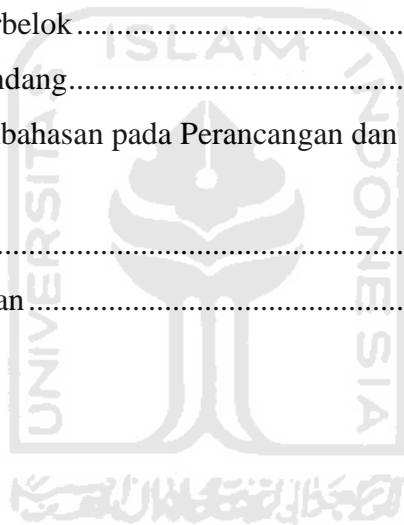
Also in this research, we adapt the main-principal of inverse-kinematic, it is to get value of servos after determining the exact position of robot. The first process was to comunicate system controller with actuator. Then determine the exact movement steps that robot would do. The steps will be implemented to control each of the aktuator. Then we get the values of the best result from many trials. The result of this research was the capability of robot to do some movement such as forward, turn, and kick, well and balanced though it still limited on flat floor.

Kata kunci : *Humanoid robot, movement-system, inverse-kinematic, actuator.*

DAFTAR ISI

Halaman Judul	i
Lembar Pengesahan Dosen Pembimbing	ii
Lembar Pengesahan Dosen Penguji	iii
Halaman Persembahan	iv
Halaman Motto	v
Ucapan Terima Kasih	vi
Abstrak	vii
Daftar Isi	x
Daftar Gambar	xii
Daftar Tabel	xiv
Daftar Grafik	xv
Bab I Pendahuluan	
1.1 Latar Belakang	1
1.2 Batasan Masalah	2
1.3 Rumusan Masalah	2
1.4 Tujuan Perancangan	2
1.5 Manfaat Perancangan	3
1.6 Sistematika Penulisan	3
Bab II Kajian Pustaka	
2.1 Robotika	4
2.2 Sistem Kendali	5
2.3 Aktuator	8
2.3.1 Motor DC	8
2.3.2 Motor Servo	8
2.4 Sistem Gerak	12
2.5 Sistem Kontroler	12

Bab III Metodologi Penelitian	
3.1	Pendahuluan 15
3.2	Perancangan Elektronik..... 18
3.3	Diagram Alir Perancangan Sistem Kendali..... 19
3.3.1	Inisialisasi Servo dan Kontroler 20
3.3.2	Perancangan Gerak Robot 22
Bab IV Hasil dan Pembahasan	
4.1	Hasil Perancangan 36
4.1.1	Gerakan Maju 37
4.1.2	Gerakan Berbelok..... 38
4.1.3	Gerakan Tendang..... 41
4.2	Kendala dan Pembahasan pada Perancangan dan Pengujian Robot..... 43
Bab V Penutup	
5.1	Kesimpulan..... 50
5.2	Penelitian Lanjutan..... 50
Daftar Pustaka	
LAMPIRAN	



DAFTAR GAMBAR

Gambar 2-1	Diagram Blok Sistem Kontrol Lup Terbuka	6
Gambar 2-2	Diagram Blok Sistem Kontrol Lup Tertutup.....	7
Gambar 2-3	Motor Servo Standar	10
Gambar 2-4	Hubungan Lebar Pulsa dengan Putaran Arah Motor Servo.....	11
Gambar 2-5	Susunan Pin Motor Servo.....	11
Gambar 2-6	Mikrokontroler	13
Gambar 3-1	Bentuk Konstruksi Mekanik Robot Berkaki Dua.....	15
Gambar 3-2	Konstruksi Robot dan Arah Gerak Tiap Motor Servo.....	16
Gambar 3-3	<i>Flowchart</i> Tahapan Proses Pembuatan Robot Berkaki	17
Gambar 3-4	Diagram Perancangan Elektronika Robot	19
Gambar 3-5	Diagram Alir Perancangan Sistem Kendali Robot.....	20
Gambar 3-6	Penomoran Motor Servo pada Robot	22
Gambar 3-7	Langkah Gerakan Maju pada Manusia.....	22
Gambar 3-8	Gerak Robot Maju 2 langkah	23
Gambar 3-9	Penomoran Servo pada Kaki Kanan Robot.....	26
Gambar 3-10	Contoh Program Gerak Maju 2 Langkah pada Robot	32
Gambar 3-11	Gerak Robot Mundur 2 Langkah.....	33
Gambar 3-12	Gerak Robot Belok Kanan 90°	34
Gambar 3-13	Gerak Robot Belok Kiri 90°	35
Gambar 4-1	Robot Hasil Perancangan Akhir	36
Gambar 4-2	Hasil Perancangan Gerakan Maju	37
Gambar 4-3	Hasil Perancangan Gerak Belok Kanan	38
Gambar 4-4	Hasil Perancangan Gerak Belok Kiri	39
Gambar 4-5	Hasil Perancangan Gerak Tendang (Kanan)	41
Gambar 4-6	Hasil Perancangan Gerak Tendang (Kiri)	41
Gambar 4-7	Robot pada Pengujian Pertama.....	43
Gambar 4-8	Pemisalan Robot Seperti Batang yang Lentur	44

Gambar 4-9	Bentuk Robot Setelah Modifikasi	45
Gambar 4-10	Kendala pada Servo	46
Gambar 4-11	Bentuk Part Penyangga.....	48
Gambar 4-12	Letak Part Penyangga dalam Robot	49
Gambar 4-13	Kaki Robot Tidak Goyang dan Lebih Kaku.....	49



DAFTAR TABEL

Tabel 2-1	Perbedaan Sistem Kontrol Terbuka dan Sistem Kontrol Tertutup	7
Tabel 3-1	Inisialisasi Kontroler Terhadap Servo pada Robot	21
Tabel 3-2	Nilai Servo pada Perancangan Gerak Robot Maju 2 Langkah	24
Tabel 3-3	Nilai Sudut Servo pada Perancangan Gerak Robot Maju 2 Langkah	25
Tabel 4-1	Data Servo Akhir Perancangan Gerak Maju	37
Tabel 4-2	Data Servo Akhir Perancangan Gerak Belok Kanan.....	39
Tabel 4-3	Data Servo Akhir Perancangan Gerak Belok Kiri	40
Tabel 4-4	Data Servo Akhir Perancangan Gerak Tendang (Kanan)	41
Tabel 4-5	Data Servo Akhir Perancangan Gerak Tendang (Kanan)	42
Tabel 4-6	Hasil Perancangan Ulang Gerak Robot.....	46
Tabel 4-7	Hasil Perancangan Ulang Gerak Robot (Nilai Sudut Servo)	47



DAFTAR GRAFIK

Grafik 3-1	Grafik Nilai Servo 1 pada Perancangan Gerak Robot Maju 2 Langkah ...	26
Grafik 3-2	Grafik Nilai Sudut Servo 1 pada Gerak Robot Maju 2 Langkah	26
Grafik 3-3	Grafik Nilai Servo 2 pada Gerak Robot Maju 2 Langkah	27
Grafik 3-4	Grafik Nilai Sudut Servo 2 pada Gerak Robot Maju 2 Langkah	27
Grafik 3-5	Grafik Nilai Servo 3 pada Gerak Robot Maju 2 Langkah	28
Grafik 3-6	Grafik Nilai Sudut Servo 3 pada Gerak Robot Maju 2 Langkah	28
Grafik 3-7	Grafik Nilai Servo 4 pada Gerak Robot Maju 2 Langkah	29
Grafik 3-8	Grafik Nilai Sudut Servo 4 pada Gerak Robot Maju 2 Langkah	29
Grafik 3-9	Grafik Nilai Servo 5 pada Gerak Robot Maju 2 Langkah	30
Grafik 3-10	Grafik Nilai Sudut Servo 5 pada Gerak Robot Maju 2 Langkah	30
Grafik 3-11	Nilai Servo pada Kaki Kanan Robot pada Gerak Maju Dua Langkah Robot.....	31
Grafik 3-12	Nilai Sudut Servo pada Kaki Kanan Robot pada Gerak Maju Dua Langkah Robot	31



LAMPIRAN

/*

*/

This program was produced by the
CodeWizardAVR V1.24.6 Professional
Automatic Program Generator
© Copyright 1998-2005 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com
e-mail:office@hpinfotech.com

Project :
Version :
Date : 5/23/2011
Author : F4CG
Company : F4CG
Comments:

Chip type : ATmega16
Program type : Application
Clock frequency : 11.059200 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256

*/

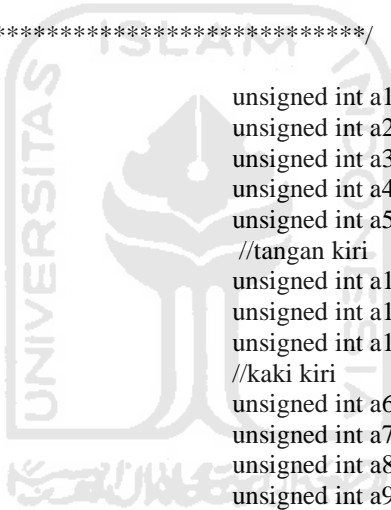
```
#include <mega16.h>
#include <stdio.h>
#include <delay.h>
char a,c;
int b=500;
int d=10;

//////////kaki kanan//////////
#define servo1_pin PORTD.6
#define servo2_pin PORTD.7
#define servo3_pin PORTC.6
#define servo4_pin PORTC.1
#define servo5_pin PORTC.0
//////////kaki kiri//////////
#define servo6_pin PORTA.4
#define servo7_pin PORTA.3
#define servo8_pin PORTA.2
#define servo9_pin PORTA.1
#define servo10_pin PORTA.0
//////////tangan kanan//////////
#define servo11_pin PORTD.5
#define servo12_pin PORTD.4
#define servo13_pin PORTC.7
//////////tangan kiri//////////
#define servo14_pin PORTA.5
#define servo15_pin PORTA.6
#define servo16_pin PORTA.7
#define servo17_pin PORTD.3

//////////kaki kanan//////////
unsigned int a1=1500;
unsigned int a2=2160;
unsigned int a3=1990;
unsigned int a4=1980;
unsigned int a5=2130;
//tangan kiri
unsigned int a14=2000;
unsigned int a15=1000;
unsigned int a16=2000;
//kaki kiri
unsigned int a6=2540;
unsigned int a7=1980;
unsigned int a8=1830;
unsigned int a9=1970;
unsigned int a10=1870;

//////////kaki kanan//////////
//tangan kanan
unsigned int pwm11=2000;
unsigned int pwm12=2900;
unsigned int pwm13=2000;
//kaki kanan
unsigned int pwm1=1500;
unsigned int pwm2=2160;
unsigned int pwm3=1990;
unsigned int pwm4=1980;
unsigned int pwm5=2130;

//////////kaki kiri//////////
//tangan kiri
unsigned int pwm14=2000;
unsigned int pwm15=1000;
unsigned int pwm16=2000;
//kaki kiri
```



```

unsigned int pwm6=2540;
unsigned int pwm7=1980;
unsigned int pwm8=1830;
unsigned int pwm9=1970;
unsigned int pwm10=1870;

unsigned char servo_index=0;

// Timer 1 output compare A interrupt
service routine
interrupt [TIM1_COMPA] void
timer1_compa_isr(void)
{
    if(servo_index==0)
    {
        servo1_pin=1;
        servo_index=1;
        if(a1>pwm1){pwm1+=d;}
        else if(a1<pwm1){pwm1-=d;}
        OCR1A=pwm1;
    }
    else if(servo_index==1)
    {
        servo1_pin=0;
        servo2_pin=1;
        servo_index=2;
        if(a2>pwm2){pwm2+=d;}
        else if(a2<pwm2){pwm2-=d;}
        OCR1A=pwm2;
    }
    else if(servo_index==2)
    {
        servo2_pin=0;
        servo3_pin=1;
        servo_index=3;
        if(a3>pwm3){pwm3+=d;}
        else if(a3<pwm3){pwm3-=d;}
        OCR1A=pwm3;
    }
    else if(servo_index==3)
    {
        servo3_pin=0;
        servo4_pin=1;
        servo_index=4;
        if(a4>pwm4){pwm4+=d;}
        else if(a4<pwm4){pwm4-=d;}
        OCR1A=pwm4;
    }
    else if(servo_index==4)
    {
        servo4_pin=0;
        servo5_pin=1;
        servo_index=5;
        if(a5>pwm5){pwm5+=d;}
        else if(a5<pwm5){pwm5-=d;}
        OCR1A=pwm5;
    }
    else if(servo_index==5)
    {
        servo5_pin=0;
        servo6_pin=1;
        servo_index=6;
        if(a6>pwm6){pwm6+=d;}
        else if(a6<pwm6){pwm6-=d;}
        OCR1A=pwm6;
    }
    else if(servo_index==6)
    {
        servo6_pin=0;
        servo7_pin=1;
        servo_index=7;
        if(a7>pwm7){pwm7+=d;}
        else if(a7<pwm7){pwm7-=d;}
        OCR1A=pwm7;
    }
    else if(servo_index==7)
    {
        servo7_pin=0;
        servo8_pin=1;
        servo_index=8;
        if(a8>pwm8){pwm8+=d;}
        else if(a8<pwm8){pwm8-=d;}
        OCR1A=pwm8;
    }
    else if(servo_index==8)
    {
        servo8_pin=0;
        servo9_pin=1;
        servo_index=9;
        if(a9>pwm9){pwm9+=d;}
        else if(a9<pwm9){pwm9-=d;}
        OCR1A=pwm9;
    }
    else if(servo_index==9)
    {
        servo9_pin=0;
        servo10_pin=1;
        servo_index=10;
        if(a10>pwm10){pwm10+=d;}
        else if(a10<pwm10){pwm10-=d;}
        OCR1A=pwm10;
    }
}
////////////////////

```



```

else if(servo_index==10)
{
    servo10_pin=0;
    servo11_pin=1;
    servo_index=11;
if(a11>pwm11){pwm11+=d;}
else if(a11<pwm11){pwm11-=d;}
    OCR1A=pwm11;
}
else if(servo_index==11)
{
    servo11_pin=0;
    servo12_pin=1;
    servo_index=12;
if(a12>pwm12){pwm12+=d;}
else if(a12<pwm12){pwm12-=d;}
    OCR1A=pwm12;
}
else if(servo_index==12)
{
    servo12_pin=0;
    servo13_pin=1;
    servo_index=13;
if(a13>pwm13){pwm13+=d;}
else if(a13<pwm13){pwm13-=d;}
    OCR1A=pwm13;
}
else if(servo_index==13)
{
    servo13_pin=0;
    servo14_pin=1;
    servo_index=14;
if(a14>pwm14){pwm14+=d;}
else if(a14<pwm14){pwm14-=d;}
    OCR1A=pwm14;
}
else if(servo_index==14)
{
    servo14_pin=0;
    servo15_pin=1;
    servo_index=15;
if(a15>pwm15){pwm15+=d;}
else if(a15<pwm15){pwm15-=d;}
    OCR1A=pwm15;
}
else if(servo_index==15)
{
    servo15_pin=0;
    servo16_pin=1;
    servo_index=16;
if(a16>pwm16){pwm16+=d;}
else if(a16<pwm16){pwm16-=d;}
    OCR1A=pwm16;
}
}
else if(servo_index==16)
{
    servo16_pin=0;
    OCR1A=2000;
    servo_index=0;
}
}

void program_1()
{
    c=getchar();
    ////////////tangan kanan//////////
    if (c=='1'){a13=a13+a;};
    if (c=='2'){a13=a13-a;};
    if (c=='3'){a12=a12+a;};
    if (c=='4'){a12=a12-a;};
    if (c=='5'){a11=a11+a;};
    if (c=='6'){a11=a11-a;};
    ////////////kaki kanan//////////
    if (c=='7'){a1=a1+a;};
    if (c=='8'){a1=a1-a;};
    if (c=='9'){a2=a2+a;};
    if (c=='0'){a2=a2-a;};
    if (c=='q'){a3=a3+a;};
    if (c=='w'){a3=a3-a;};
    if (c=='e'){a4=a4+a;};
    if (c=='r'){a4=a4-a;};
    if (c=='t'){a5=a5+a;};
    if (c=='y'){a5=a5-a;};
    ////////////tangan kiri//////////
    if (c=='u'){a16=a16+a;};
    if (c=='i'){a16=a16-a;};
    if (c=='o'){a15=a15+a;};
    if (c=='p'){a15=a15-a;};
    if (c=='a'){a14=a14+a;};
    if (c=='s'){a14=a14-a;};
    ////////////kaki kiri//////////
    if (c=='d'){a6=a6+a;};
    if (c=='f'){a6=a6-a;};
    if (c=='g'){a7=a7+a;};
    if (c=='h'){a7=a7-a;};
    if (c=='j'){a8=a8+a;};
    if (c=='k'){a8=a8-a;};
    if (c=='z'){a9=a9+a;};
    if (c=='x'){a9=a9-a;};
    if (c=='c'){a10=a10+a;};
    if (c=='v'){a10=a10-a;};
    ////////////
    if(a1>3000){a1=3000;};
    if(a2>2500){a2=2500;};
    if(a3>3000){a3=3000;};
    if(a4>3000){a4=3000;};
    if(a5>3000){a5=3000;};
}
}

```

```

if(a6>3000){a6=3000;}; //kaki kiri
if(a7>3000){a7=3000;}; a6=2540;
if(a8>3000){a8=3000;}; a7=2030;
if(a9>2500){a9=2500;}; a8=1830;
if(a10>3000){a10=3000;}; a9=1970;
if(a11>3000){a11=3000;}; a10=1870;
if(a12>3000){a12=3000;}; delay_ms(b);
if(a13>3000){a13=3000;}; ////////////////2
if(a14>3000){a14=3000;}; //tangan kanan
if(a15>3000){a15=3000;}; a11=2000;
if(a16>3000){a16=3000;}; a12=2900;
a13=2000;

if(a1<1000){a1=1000;}; //kaki kanan
if(a2<1500){a2=1500;}; a1=1400;
if(a3<1000){a3=1000;}; a2=2110;
if(a4<1000){a4=1000;}; a3=1990;
if(a5<1000){a5=1000;}; a4=1980;
if(a6<1000){a6=1000;}; a5=2130;
if(a7<1000){a7=1000;}; // tangan kiri
if(a8<1000){a8=1000;}; a14=2000;
if(a9<1500){a9=1500;}; a15=1000;
if(a10<1000){a10=1000;}; a16=2000;
if(a11<1000){a11=1000;}; //kaki kiri
if(a12<1000){a12=1000;}; a6=2440;
if(a13<1000){a13=1000;}; a7=2030;
if(a14<1000){a14=1000;}; a8=1830;
if(a15<1000){a15=1000;}; a9=1970;
if(a16<1000){a16=1000;}; a10=1870;
delay_ms(b);

printf("13=%i° 12=%i° 11=%i°--1=%i° ////////////////3
2=%i° 3=%i° 4=%i° 5=%i°--16=%i° //tangan kanan
15=%i° 14=%i°--6=%i° 7=%i° 8=%i° a11=2000;
9=%i° a12=2900;
10=%i°\r",a11,a12,a13,a1,a2,a3,a4,a5,a14,a a13=2000;
15,a16,a6,a7,a8,a9,a10); //Data yang //kaki kanan
ditampilkan pd HyperTerminal a1=1300;
//delay_ms(200); a2=2110;
} a3=1990;
a4=1980;
a5=2130;

void liat_kanan(){ // tangan kiri
b=250; a14=2000;
d=20; a15=1000;
////////////////tgak a16=2000;
//tangan kanan //kaki kiri
a11=2000; a6=2340;
a12=2900; a7=2030;
a13=2000; a8=1830;
//kaki kanan a9=1970;
a1=1500; a10=1870;
a2=2110; delay_ms(b) ;
a3=1990; ////////////////4
a4=1980; //tangan kanan
a5=2130; a11=2000;
// tangan kiri a12=2900;
a14=2000; a13=2000;
a15=1000; //kaki kanan
a16=2000; a1=1200;

```

```

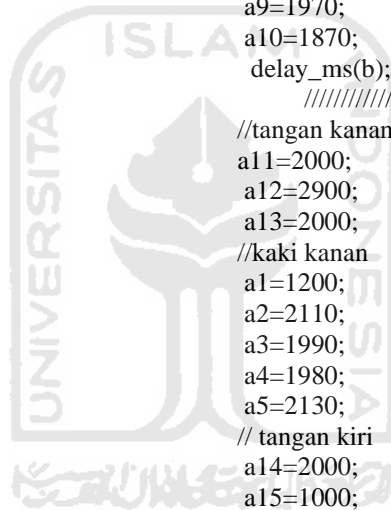
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2240;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////5
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1100;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2140;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////6
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1000;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2040;
a7=2030;
a8=1830;
a9=1970;
a10=1870;

```

```

  delay_ms(b);
  ////////////////5
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1100;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2140;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////4
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1200;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2240;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////3
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1300;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;

```



```

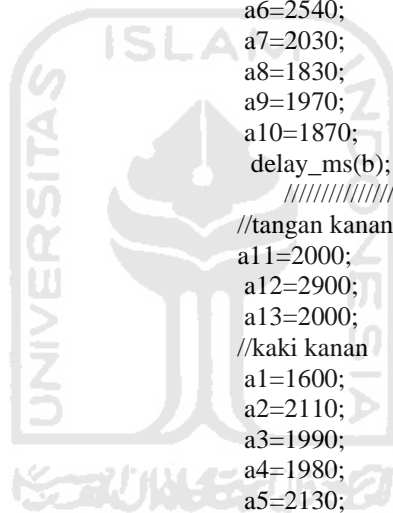
a15=1000;
a16=2000;
//kaki kiri
a6=2340;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////2
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1400;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2440;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b) ;
  ////////////////tgak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  ////////////////
PORTB.0=0;
PORTB.1=1;
  ////////////////
  delay_ms(b);
}

```

```

void liat_kiri(){
b=200;
d=20;
  ////////////////tgak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////2
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1600;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2640;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ////////////////3
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1700;
a2=2110;
a3=1990;
a4=1980;

```



```

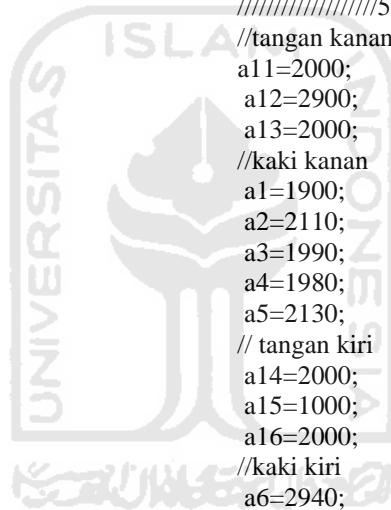
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2740;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b) ;
//////////4
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1800;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2840;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
//////////5
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1900;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2940;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
//////////6
//tangan kanan

```

```

a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=2000;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=3040;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
//////////5
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1900;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2940;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
//////////4
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1800;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri

```



```

a6=2840;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ///////////////////////////////////////////////////3
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1700;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2740;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ///////////////////////////////////////////////////2
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1600;
a2=2110;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2640;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b) ;
  ///////////////////////////////////////////////////tidak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2110;

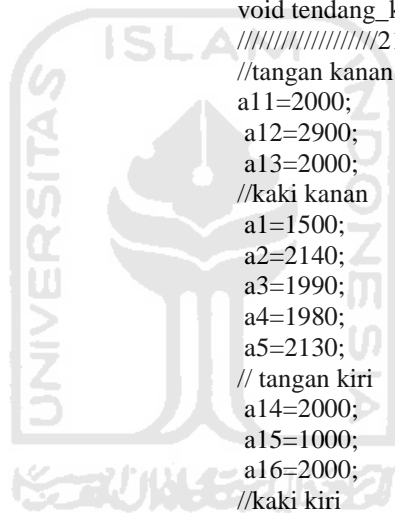
```

```

a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2030;
a8=1830;
a9=1970;
a10=1870;
  ///////////////////////////////////////////////////
  PORTB.0=1;
  PORTB.1=0;
  ///////////////////////////////////////////////////
  delay_ms(b);
}

void tendang_kiri(){
  ///////////////////////////////////////////////////21 tgak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2140;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2000;
a8=1830;
a9=1970;
a10=1870;
  delay_ms(b);
  ///////////////////////////////////////////////////22
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2140;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;

```



```

//kaki kiri
a6=2540;
a7=2000;
a8=1830;
a9=1970;
a10=2040;
delay_ms(b);
//////////////////23
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2140;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
pwm7=2000;
a8=1970;
a9=1970;
a10=1870;
delay_ms(b);
//////////////////24
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2140;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
pwm7=1700;
a8=1970;
a9=1970;
a10=1870;
delay_ms(b);
//////////////////21 tgak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;

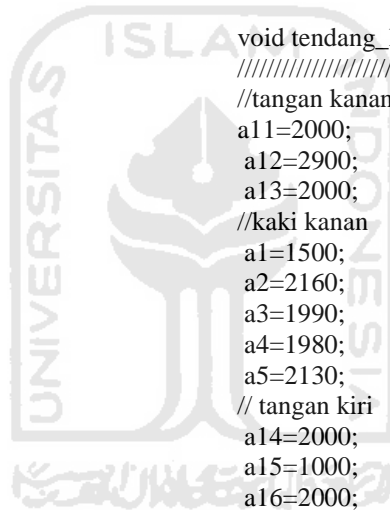
```

```

a2=2140;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2000;
a8=1830;
a9=1970;
a10=1870;
//////////////////
PORTB.1=1;
PORTB.2=0;
//////////////////
delay_ms(b);
}

void tendang_kanan(){
//////////////////11 tegak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2160;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1980;
a8=1830;
a9=1970;
a10=1870;
delay_ms(b);
//////////////////12
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2160;
a3=1990;
a4=1980;
a5=1940;
// tangan kiri
a14=2000;
a15=1000;

```



```

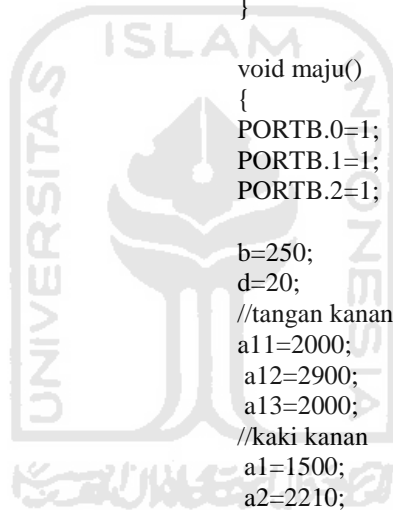
a16=2000;
//kaki kiri
a6=2540;
a7=1980;
a8=1830;
a9=1970;
a10=1800;
delay_ms(b);
////////////////////13
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
pwm2=2110;
a3=1850;
a4=1980;
a5=2070;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1980;
a8=1830;
a9=1970;
a10=1700;
delay_ms(b);
////////////////////14
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
pwm2=2500;
a3=1850;
a4=1980;
a5=2070;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2000;
a8=1830;
a9=1970;
a10=1700;
delay_ms(b);
////////////////////11 tegak
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan

```

```

a1=1500;
a2=2160;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1980;
a8=1830;
a9=1970;
a10=1870;
////////////////
PORTB.1=0;
PORTB.2=1;
////////////////
delay_ms(b);
}
void maju()
{
PORTB.0=1;
PORTB.1=1;
PORTB.2=1;
b=250;
d=20;
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2210;
a3=1990;
a4=1980;
a5=2080;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1930;
a8=1830;
a9=1970;
a10=1700;
delay_ms(b);
////////////////////2
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1650;

```




```

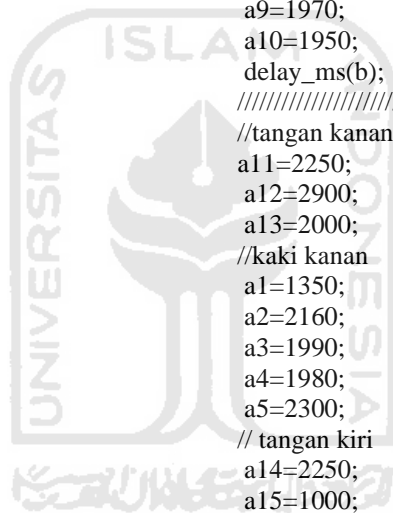
a2=2210;
a3=1990;
a4=1980;
a5=2080;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri
a6=2690;
a7=2000;
a8=1830;
a9=1970;
a10=1730;
delay_ms(b);
////////////////////3
//tangan kanan
a11=1500;
a12=2900;
a13=2000;
//kaki kanan
a1=1650;
a2=2210;
a3=1990;
a4=1980;
a5=2150;
// tangan kiri
a14=1500;
a15=1000;
a16=2000;
//kaki kiri
a6=2690;
a7=2000;
a8=1830;
a9=1970;
a10=1880;
delay_ms(b);
////////////////////4
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1650;
a2=2310;
a3=1990;
a4=1990;
a5=2290;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri
a6=2690;
a7=1960;
a8=1830;
a9=1970;
a10=1950;

```

```

delay_ms(b);
////////////////////5
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2210;
a3=1990;
a4=1980;
a5=2300;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1980;
a8=1830;
a9=1970;
a10=1950;
delay_ms(b);
////////////////////6
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1350;
a2=2160;
a3=1990;
a4=1980;
a5=2300;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2390;
a7=1990;
a8=1830;
a9=1970;
a10=1950;
delay_ms(b);
////////////////////7
//tangan kanan
a11=2500;
a12=2900;
a13=2000;
//kaki kanan
a1=1350;
a2=2160;
a3=1990;
a4=1980;
a5=2150;
// tangan kiri
a14=2500;

```



```

a15=1000;
a16=2000;
//kaki kiri
a6=2390;
a7=1990;
a8=1830;
a9=1970;
a10=1880;
delay_ms(b);
////////////////////8
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1350;
a2=2180;
a3=1990;
a4=1980;
a5=2080;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2390;
a7=1840;
a8=1830;
a9=1970;
a10=1730;
delay_ms(b);
}

void tegak()
{
PORTB.0=1;
PORTB.1=1;
PORTB.2=1;
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2140;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=2000;
a8=1830;
a9=1970;
a10=1870;

```

```

delay_ms(b);
}

void kanan()
{
b=250;
d=15;

PORTB.0=1;
PORTB.1=1;
PORTB.2=1;
////////////////////0
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1870;
delay_ms(b);
////////////////////1
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2160;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1720;
delay_ms(b);
////////////////////2
//tangan kanan
a11=1500;

```



```

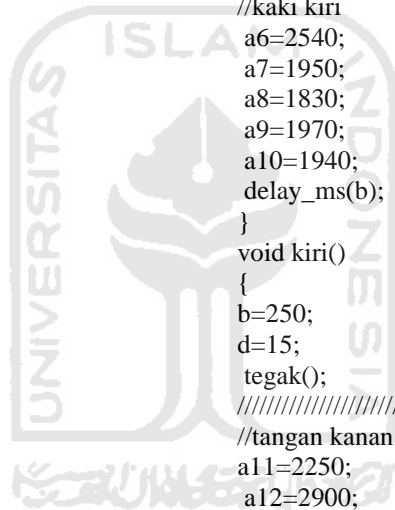
a12=2900;
a13=2000;
//kaki kanan
a1=1750;
a2=2190;
a3=1990;
a4=1980;
a5=2160;
// tangan kiri
a14=1500;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1720;
delay_ms(b);
////////////////////3
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1750;
a2=2190;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1870;
delay_ms(b);
////////////////////4
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1750;
a2=2350;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;

```

```

a7=1950;
a8=1830;
a9=1970;
a10=1940;
delay_ms(b);
////////////////////5
//tangan kanan
a11=2500;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2500;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1940;
delay_ms(b);
}
void kiri()
{
b=250;
d=15;
tegak();
////////////////////1
//tangan kanan
a11=2250;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2250;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1940;
delay_ms(b);
////////////////////2
//tangan kanan
a11=2500;

```



```

a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2150;
a3=1990;
a4=1980;
a5=2280;
// tangan kiri
a14=2500;
a15=1000;
a16=2000;
//kaki kiri
a6=2290;
a7=1950;
a8=1830;
a9=1970;
a10=1940;

delay_ms(b);
/////////////////////////////////3
//tangan kanan
a11=2000;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2150;
a3=1990;
a4=1980;
a5=2130;
// tangan kiri
a14=2000;
a15=1000;
a16=2000;
//kaki kiri
a6=2290;
a7=1950;
a8=1830;
a9=1970;
a10=1870;
delay_ms(b);
/////////////////////////////////4
//tangan kanan
a11=1750;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2060;
// tangan kiri
a14=1750;
a15=1000;
a16=2000;
//kaki kiri

```

```

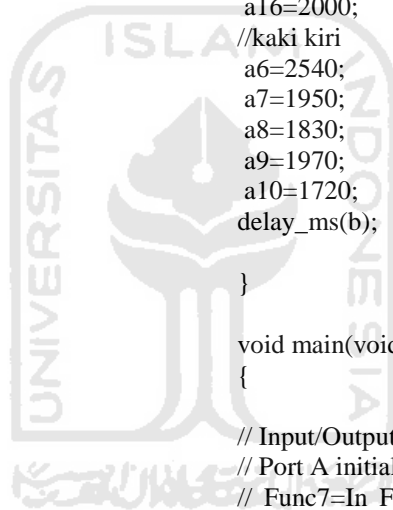
a6=2290;
a7=1840;
a8=1830;
a9=1970;
a10=1720;
delay_ms(b);
/////////////////////////////////5
//tangan kanan
a11=1500;
a12=2900;
a13=2000;
//kaki kanan
a1=1500;
a2=2190;
a3=1990;
a4=1980;
a5=2060;
// tangan kiri
a14=1500;
a15=1000;
a16=2000;
//kaki kiri
a6=2540;
a7=1950;
a8=1830;
a9=1970;
a10=1720;
delay_ms(b);
}

void main(void)
{
// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In
Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T
State3=T State2=T State1=T State0=T
PORTA=0x00;
DDRA=0xFF;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In
Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T
State3=T State2=T State1=T State0=T
PORTB=0x08;
DDRB=0x07;

// Port C initialization
// Func7=Out Func6=Out Func5=In
Func4=In Func3=In Func2=In Func1=Out
Func0=Out
// State7=0 State6=0 State5=P State4=P
State3=P State2=P State1=0 State0=0
PORTC=0x00;

```



```

DDRC=0x00; // Communication Parameters: 8 Data, 1
// Stop, No Parity
// Port D initialization // USART Receiver: On
// Func7=In Func6=In Func5=In Func4=In // USART Transmitter: On
// Func3=In Func2=In Func1=In Func0=In // USART Mode: Asynchronous
// State7=T State6=T State5=T State4=T // USART Baud rate: 9600
// State3=T State2=T State1=T State0=T UCSRA=0x00;
PORTD=0x0C; UCSRB=0x18;
DDR0=0xF0; UCSRC=0x86;
// Timer/Counter 0 initialization UBRRH=0x00;
// Clock source: System Clock UBRRL=0x47;
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh // Analog Comparator initialization
// OC0 output: Disconnected // Analog Comparator: Off
TCCR0=0x00; // Analog Comparator Input Capture by
TCNT0=0x00; Timer/Counter 1: Off
OCR0=0x00; ACSR=0x80;
SFIOR=0x00;

// Timer/Counter 1 initialization #asm("sei")
// Clock source: System Clock while (1)
// Clock value: 1382.400 kHz {
// Mode: CTC top=OCR1A if
// OC1A output: Discon. (PIND.3==1&&PIND.2==1&&PINB.3==1)
// OC1B output: Discon. {tegak();};
// Noise Canceler: Off if
// Input Capture on Falling Edge (PIND.3==0&&PIND.2==0&&PINB.3==1)
// Timer 1 Overflow Interrupt: Off {kanan();};
// Input Capture Interrupt: Off if
// Compare A Match Interrupt: On (PIND.3==1&&PIND.2==0&&PINB.3==0)
// Compare B Match Interrupt: Off {kiri();};
TCCR1A=0x00; if
TCCR1B=0x0A; (PIND.3==0&&PIND.2==0&&PINB.3==0)
OCR1A=5000; {maju();};
if
// Timer/Counter 2 initialization (PIND.3==1&&PIND.2==1&&PINB.3==0)
// Clock source: System Clock {liat_kiri();};
// Clock value: Timer 2 Stopped if
// Mode: Normal top=FFh (PIND.3==0&&PIND.2==1&&PINB.3==1)
// OC2 output: Disconnected {liat_kanan();};
ASSR=0x00; if
TCCR2=0x00; (PIND.3==0&&PIND.2==1&&PINB.3==0)
TCNT2=0x00; {tendang_kanan();};
OCR2=0x00; if
// External Interrupt(s) initialization (PIND.3==1&&PIND.2==0&&PINB.3==1)
// INT0: Off {tendang_kiri();};
// INT1: Off
// INT2: Off
MCUCR=0x00; };
MCUCSR=0x00; }

// Timer(s)/Counter(s) Interrupt(s)
initialization
TIMSK=0x10;

// USART initialization

```

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Perkembangan robotika saat ini berkembang sangat pesat. Robotika tidak hanya di pakai di pabrik-pabrik suatu industri atau di laboratorium, tetapi sekarang sudah mulai dijual secara umum dalam bentuk robot humanoid dengan kecerdasan yang dimilikinya. Untuk keperluan itu robot digunakan sebagai alat peraga karena robot dapat diprogram dengan bahasa pemrograman.

Untuk menunjang dan mengenalkan pentingnya pendidikan tentang robot, pemerintah gencar mengadakan lomba-lomba robot, baik tingkat SMA hingga tingkat Perguruan Tinggi, misalnya Kontes Robot Cerdas Indonesia dan Kontes Robot Indonesia (KRI dan KRCI) di tingkat perguruan tinggi, pemenangnya nantinya akan mewakili Indonesia di tingkat Internasional. Robot yang dilombakan bervariasi, salah satunya kini adalah *RoboSoccer* yang pesertanya menggunakan robot *Humanoid* (robot berkaki dua).

Robot berkaki pada dasarnya meniru sistem penggerak yang terdapat di alam. Ada robot dengan dua kaki, tiga, ataupun lebih seperti pada hewan. Kelebihan robot berkaki dibandingkan dengan robot beroda adalah kemampuan untuk dapat melewati lintasan yang tidak kontinyu, misalnya anak tangga atau jalan yang kasar. Kelebihan lain adalah adanya pemisahan lintasan antara robot dengan permukaan yang dilewati, karena kaki dapat sekaligus berfungsi sebagai penyerap getaran. Meski demikian dalam perancangannya robot berkaki jauh lebih sulit dibandingkan robot beroda ataupun tipe lainnya karena mekanisme gerakan yang lebih rumit.

Berdasarkan latar belakang diatas perlu dilakukan proses penelitian dengan fokus pada perancangan sistem kendali dari robot dua kaki.

1.2 BATASAN MASALAH

Batasan masalah dimaksudkan agar tidak menyimpang dan lebih terarah dari objek pembahasan maka yang difokuskan pada perancangan sistem kendali dari robot dua kaki ini, yaitu;

1. Aktuator yang digunakan berupa motor servo standar yang dapat bergerak 180° .
2. Penelitian difokuskan pada perancangan sistem kendali dari robot dua kaki dengan metode pengujian-pengujian.
3. Kontroler yang digunakan adalah *Minimum System* ATMEGA 16
4. Robot berjalan pada lantai yang permukaannya rata.

1.3 RUMUSAN MASALAH

Dengan memperhatikan latar belakang masalah dan batasan masalah tersebut maka dapat ditentukan rumusan masalah pada perancangan robot dua kaki, sebagai berikut:

1. Bagaimanakah sistem kontrol yang efektif dan efisien sehingga robot dapat berjalan dengan baik ?
2. Bagaimanakah pengaturan setiap motor servo dalam penyesuaiannya dengan pergerakan robot ?

1.4 TUJUAN TUGAS AKHIR

Tujuan tugas akhir ini adalah untuk merancang dan membuat sistem kendali robot dua kaki sehingga mampu berjalan dengan seimbang dan melakukan beberapa gerakan.

1.5 MANFAAT TUGAS AKHIR

Tugas akhir ini diharapkan dapat memberikan kontribusi pada pengembangan ilmu pengetahuan dan teknologi robot dua kaki yang akan sangat berguna bagi perkembangan teknologi robot khususnya di tingkat universitas pada masa yang akan datang.

1.6 SISTEMATIKA PENULISAN

Sistematika penulisan tugas akhir ini terdiri dari masing-masing bab yaitu ;

1. Bagian pendahuluan berisikan halaman judul, lembar pengesahan dosen pembimbing, lembar pengesahan dosen penguji, halaman motto, kata pengantar, abstraksi, daftar isi, daftar tabel dan juga daftar gambar.
2. Bab I Pendahuluan berisikan tentang latar belakang masalah yang akan diteliti, rumusan masalah, batasan masalah, tujuan dan manfaat dari tugas akhir dan sistematika penulisan laporan tugas akhir ini.
3. Bab II Kajian Pustaka berisikan teori-teori yang melandasi penyusunan tugas akhir ini.
4. Bab III Metodologi Penelitian berisikan penjelasan tentang proses, langkah-langkah, dan metode perancangan sistem kendali robot.
5. Bab IV Hasil dan Pembahasan menjelaskan tentang apa saja menjadi kendala ketika perancangan termasuk pemberian solusi dari kendala yang ada.
6. Bab V Penutup berisikan kesimpulan dan juga dilengkapi dengan apa saja yang bisa dilakukan untuk penelitian berikutnya. Bagian akhir ini memuat daftar pustaka, yaitu sumber acuan yang digunakan dalam pelaksanaan dan pembuatan laporan tugas akhir dan lampiran.

BAB II

KAJIAN PUSTAKA

2.1 Robotika

Robot mempunyai berbagai versi mengenai awal kemunculannya. Pada tahun 1770, Pierre Jacquet Droz, seorang pembuat jam berkebangsaan swiss membuat 3 boneka mekanis. Uniknya, boneka tersebut dapat melakukan fungsi spesifik, yaitu dapat menulis, yang lainnya dapat memainkan musik dan organ, dan yang ketiga dapat menggambar.

Pada tahun 1898, Nikola Tesla membuat sebuah *boat* yang dikontrol melalui *radio remote control*, dan didemokan di Madison Square Garden. Namun usaha untuk membuat *autonomus boat* tersebut gagal karena masalah dana.

Kata Robot sendiri berasal dari bahasa *Czech*, *robota* yang berarti pekerja. Kata Robot diperkenalkan dalam bahasa Inggris pada tahun 1921 oleh *Wright Karel Capek*. Selanjutnya Jacques de Vaucanson yang membuat bebek mekanik pada 1738 yang mampu mengunyah dan memakan biji-bijian. Hisashine Tanaga pria berkebangsaan Jepang pada 1796 membuat sejenis mainan tetapi mainan itu dapat membuat minuman teh. Gakutensoku robot pertama yang dibangun di Jepang, telah dibuat di Osaka pada tahun 1929. Robot ini dirancang dan diproduksi oleh ahli biologi Makoto Nishimura. Robot ini dapat mengubah ekspresi wajah dan menggerakkan kepala dan tangan melalui mekanisme tekanan udara. Namun robot ini hilang saat dipamerkan di Jerman pada 1939.

Bahkan di zaman sekarang sudah banyak robot yang dapat memainkan drama seperti halnya manusia yang memainkan drama yang dapat mengeluarkan air mata, mengedipkan mata dan memiliki suatu ekspresi pada wajahnya. Bahkan ada Lomba maraton antar robot yang pertama di dunia diadakan di kota Osaka, Jepang.

Menurut The American Heritage® Dictionary of the English Language, Fourth Edition, 2006 (Halim dan Tjen,2007), kata “Robot” dapat didefinisikan sebagai berikut :

- 1) “A mechanical device that sometimes resembles a human and is capable of performing a variety of often complex human tasks on command or by being programmed in advance. “
- 2) “A machine or device that operates automatically or by remote control.”
- 3) “A person who works mechanically without original thought, especially one who responds automa-tically to the commands of others.”

Dari penjelasan tersebut mereka menyimpulkan bahwa Robot adalah sebuah alat multi fungsi yang terdiri dari beberapa *joint* yang dirancang untuk melakukan suatu tugas, baik memindahkan barang atau melakukan tugas lainnya sesuai dengan program yang diberikan.

Definisi suatu 'Robot' Menurut Robot Institute of America, robot adalah "Sebuah manipulator, reprogrammable multifungsi yang dirancang untuk memindahkan material, bagian, peralatan, atau perangkat khusus melalui berbagai gerakan yang diprogram untuk kinerja berbagai tugas".

Definisi yang lebih inspiratif dapat ditemukan di Webster. Menurut Webster robot adalah "Sebuah perangkat otomatis yang melakukan fungsi normal dianggap berasal dari manusia atau mesin dalam bentuk manusia."

Robot *Humanoid* adalah robot yang menyerupai manusia baik secara fisik maupun beberapa fungsinya. Robot *Humanoid* mempunyai dua kaki, dua tangan, kepala dan dapat berjalan layaknya manusia. Sedangkan untuk beberapa fungsi seperti berbicara, mendengar, ataupun melihat sampai saat ini di setiap negara masih dalam pengembangan.

2.2 Sistem Kendali

Pengertian sistem kontrol itu sendiri adalah proses pengaturan / pengendalian terhadap satu atau beberapa besaran (variabel, parameter) sehingga berada pada suatu harga atau dalam suatu rangkuman harga (*range*) tertentu (anonim,2011).

Menurut rangkaian sistemnya, sistem kendali dibagi menjadi dua, yaitu kontrol lup terbuka (*Open Loop Control*) dan kontrol lup tertutup (*Closed Loop Control*).

Sistem kendali lup terbuka atau umpan maju (*feedforward control*) umumnya mempergunakan pengatur (*controller*) serta aktuator kendali (*control actuator*) yang berguna untuk memperoleh respon sistem yang baik. Sistem kendali ini keluarannya tidak diperhitungkan ulang oleh controller. Suatu keadaan apakah plant benar-benar telah mencapai target seperti yang dikehendaki masukan atau referensi, tidak dapat mempengaruhi kinerja kontroler.

Singkatnya, sistem kendali lup terbuka adalah sistem kontrol yang keluarannya tidak mempengaruhi terhadap aksi pengontrolannya.

• Open Loop

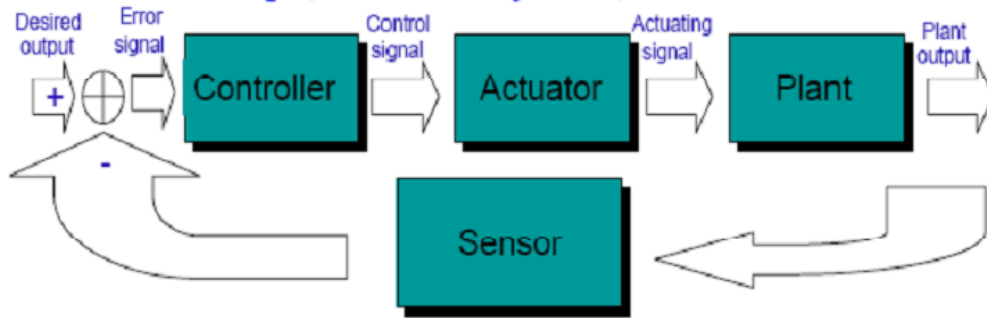


Gambar 2.1 Diagram Blok Sistem Kontrol Lup Terbuka

Sedangkan sistem kendali lup tertutup (*closed loop system*) memanfaatkan variabel yang sebanding dengan selisih respon yang terjadi terhadap respon yang diinginkan. Sistem seperti ini juga sering dikenal dengan sistem kendali umpan balik.

Closed loop System dapat juga diartikan sebagai sistem kontrol yang keluarannya mempengaruhi langsung terhadap aksi pengontrolannya

• Closed-Loop (Feedback system)



Gambar 2.2 Diagram Blok Sistem Kontrol Lup Tertutup

Kedua jenis sistem tersebut merupakan sistem dasar yang menjadi acuan dalam pembuatan berbagai macam sistem kendali yang lebih rumit dan kompleks. Masing-masing sistem kendali tersebut memiliki karakteristik yang berbeda, digambarkan pada tabel 2.1 tentang perbedaan sistem kontrol terbuka dan sistem kontrol tertutup.

Tabel 2.1 Perbedaan Sistem Kontrol Terbuka dan Sistem Kontrol Tertutup.

No	<i>Open Loop System</i>	<i>Close Loop System</i>
1	Aksi kontrolnya tidak terpengaruh <i>output</i> sistem	Aksi kontrolnya mempengaruhi <i>output</i> sistem (melalui <i>feedback</i>)
2	Tidak dapat memberikan koreksi jika ada gangguan (<i>Disturbance</i>)	Dapat memberikan koreksi jika ada gangguan (<i>Disturbance</i>) (ditunjukkan pada gambar 2.2)
3	Sederhana dan lebih murah	Kompleks dan mahal, karena komponennya lebih banyak

2.3 Aktuator

Aktuator adalah perangkat elektromekanik yang menghasilkan energi mekanik berupa gerakan. Dapat dibuat dari sistem motor listrik (motor DC magnet permanen, *brushless*, *shunt*, *series*, motor DC servo, motor stepper, dan solenoid), sistem pneumatis (perangkat kompresi berbasis udara atau gas nitrogen) dan perangkat hidrolis (berbasis bahan cair seperti oli). Untuk meningkatkan tenaga mekanik aktuator atau torsi gerakan dapat dipasang gearbox, baik system *direct gear* atau *sprocket gear* (Pitowarno, 2006).

2.3.1 Motor DC

Motor DC banyak digunakan pada peralatan yang menggunakan putaran sebagai penggerak seperti *walkman*, *handycam*, mainan anak-anak dan printer. Robot juga banyak yang menggunakan motor DC yang dilengkapi dengan sistem *gear* yang mereduksi putaran motor DC dan meningkatkan torsi. Sesuai dengan namanya, motor DC beroperasi dengan tegangan DC (arus searah). Dengan demikian putaran motor DC dapat berbalik arah jika polaritas tegangan yang diberikan juga dirubah. Motor DC juga datang dengan tegangan kerja yang bervariasi. Ada yang memiliki tegangan kerja 3 Volt, 6 Volt dan 12 Volt (Pitowarno, 2006).

2.3.2 Motor Servo

Aktuator yang digunakan pada setiap kaki maupun tangan robot dua kaki adalah motor servo. Motor servo merupakan motor yang banyak dipakai pada peralatan robot, banyak juga pada beberapa piranti elektronika yang dikontrol menggunakan radio control (RC). Motor servo ini yang bertanggung jawab untuk membuat mobil mainan dan sebuah rangkaian mekanik dapat berbelok dan bergerak sesuai yang dikehendaki. Berbeda dengan motor DC dan motor stepper yang didisain menggunakan sistem *open feedback* (tanpa umpan balik), maka

motor servo merupakan motor yang didisain bersistem *closed feedback* (dengan umpan balik).

Motor servo memiliki banyak jenis dan ukuran. Ada yang berukuran kecil yang biasanya dipakai untuk mainan (pesawat atau mobil) sampai yang besar yang dapat dipakai untuk peralatan industri. Dalam banyak hal, motor servo banyak dipakai untuk rangkaian robot. Hal yang unik dari motor servo adalah bahwa motor servo diatur/dikontrol menggunakan pulsa. Namun dengan menggunakan komputer atau rangkaian mikrokontroler, akan dapat dengan mudah mengontrol motor servo.

Motor servo jika dibuka dari badannya maka akan didapat part-part penyusunnya yaitu, sebuah motor yang merupakan jantung dari motor servo, sekumpulan *gear-gear* yang berguna untuk mengurangi kecepatan putar motor, sebuah potensiometer dan PCB yang berisi rangkaian kontrol. Motor servo termasuk jenis *closed loop*. Karena itu digunakan potensiometer dan rangkaian kontrol akan membentuk rangkaian umpan balik. Rangkaian kontrol dan motor tersebut diberi daya dengan tegangan DC sebesar 4.8 volt.

Motor servo memiliki 3 posisi utama yaitu posisi 0 derajat, posisi 90 derajat dan posisi 180 derajat. Poros motor servo biasanya dihubungkan dengan suatu mekanisme sehingga dapat membuat/mengontrol pergerakan roda depan pada suatu mobil mainan. Pada saat poros motor servo pada posisi 0 derajat, maka roda depan mobil mainan akan belok ke kiri, jika posisi poros pada 90 derajat, maka roda depan mobil mainan akan lurus. Sedangkan jika posisinya 180 derajat, maka roda depan mobil mainan akan berbelok ke kanan. Karena ada 3 posisi utama seperti dijelaskan di atas, maka dibuatlah sebuah cara khusus untuk mengatur motor Servo tersebut.

Cara yang digunakan adalah dengan memberikan pulsa digital dengan lebar yang berbeda-beda. Jika diberikan pulsa dengan lebar 1.5 ms maka motor servo akan berputar menuju 90 derajat (posisi tengah = middle). Pulsa dengan lebar 2.0 ms akan membuat poros motor servo menuju 180 derajat (posisi kanan), sedangkan pulsa dengan lebar 1.0 ms akan membuat motor Servo menuju 0 derajat (posisi kiri). Pulsa

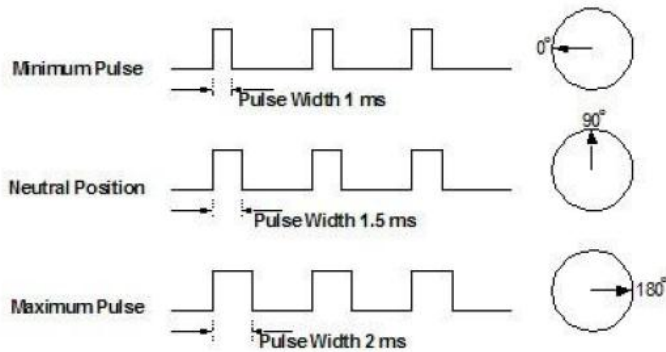
tersebut dikirimkan sebanyak kira-kira 50 kali per detik / 50 Hz. Contoh dari motor servo dapat dilihat pada gambar 2.3.



Gambar 2.3 Motor Servo Standar

Motor servo merupakan sejenis motor DC, motor servo beroperasi secara *closed loop*. Poros motor dihubungkan dengan rangkaian kendali, sehingga jika putaran poros belum sampai posisi yang diperintahkan maka rangkaian kendali akan terus mengoreksi posisi hingga mencapai posisi yang diperintahkan.

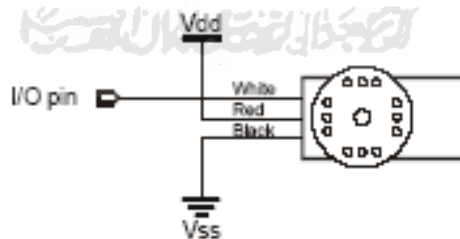
Motor servo terdiri dari dua jenis yaitu motor servo standar yang hanya dapat bergerak pada sudut tertentu, biasanya 180° atau 270° , dan motor servo continuous yang dapat berputar secara kontinu. Motor servo mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan dapat dikendalikan hanya dengan memberikan pengaturan lebar pulsa pada bagian pin kontrolernya. Lebar pulsa yang diperlukan antara 1 ms hingga 2 ms, contohnya Jika diberikan pulsa dengan lebar 1.5 ms maka motor servo akan berputar menuju 90 derajat (posisi tengah = *middle*). Pulsa dengan lebar 2.0 ms akan membuat poros motor servo menuju 180 derajat (posisi kanan), sedangkan pulsa dengan lebar 1.0 ms akan membuat motor servo menuju 0 derajat (posisi kiri). Untuk bekerja secara baik pulsa tersebut dikirimkan dengan frekuensi 50 kali per detik / 50 Hz. Hubungan lebar pulsa dengan putaran arah motor servo dapat dilihat pada gambar 2.4.



Gambar 2.4 Hubungan Lebar Pulsa dengan Putaran

Arah Motor Servo

Motor servo mempunyai putaran yang lambat, dimana biasanya ditunjukkan oleh rute putaran yang lambat, namun demikian memiliki torsi yang kuat karena internal gearnya. Pada motor servo terdapat tiga pin yaitu pin sinyal, pin tegangan catu daya positif dan negative, susunan ketiga pin tersebut dapat dilihat pada gambar 2.5. Untuk beroperasi dengan baik motor servo sendiri memerlukan tegangan catu daya berkisar 4,8 hingga 6 V.



Gambar 2.5 Susunan Pin Motor Servo

2.4 Sistem Gerak Robot Dua Kaki

Sistem gerak yang dimaksud di sini adalah hasil pengaturan gerakan berurutan motor servo robot secara keseluruhan sehingga dapat memanipulasi posisi dari sebuah titik satu ke titik yang lain.

Robot dua kaki yang dibuat menggunakan aktuator berupa 17 buah motor servo, yang terdiri dari dua kaki masing-masing 5 servo, dua tangan masing-masing 3 servo, dan sebuah servo sebagai penggerak kepala. Dalam pengaturan posisinya diupayakan sesuai dengan analogi tubuh manusia yaitu tempat-tempat sendi baik pada kaki maupun tangan manusia sesungguhnya.

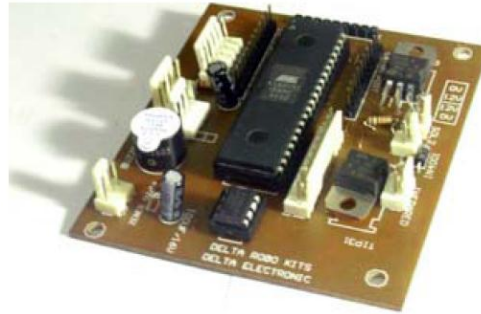
Pada dasarnya sistem kaki adalah gerakan roda yang didesain sedemikian rupa hingga memiliki kemampuan gerak seperti makhluk hidup. Robot berjalan dengan sistem dua kaki atau biped robot memiliki struktur seperti kaki manusia setidaknya-tidaknya memiliki sendi-sendi gerak yang mewakili pergelangan kaki lutut dan pinggul. Demikian pula halnya pergelangan kaki, idealnya adalah juga memiliki kemampuan gerakan polar. Untuk robot binatang atau animaloid seperti serangga, jumlah kaki dapat didesain lebih dari empat.

2.5 Sistem Kontroller

Sistem kontroler adalah rangkaian elektronik yang setidaknya-tidaknya terdiri dari rangkaian prosesor (Pitowarno. 2006). Bila diperlukan Sistem kontroler dapat dilengkapi dengan sistem monitor seperti *seven segment*, LCD (*liquid crystal display*) ataupun CRT (*cathode ray-tube*).

Sedangkan prosesor itu sendiri dapat berupa *microcontroller*. *Microcontroller* adalah sebuah komputer mini yang keseluruhan sistemnya berada dalam satu *chip*. Komponen *Microcontroller* terdiri dari CPU (*Central Processing Unit*) yang bertugas untuk memproses data, RAM (*Random Access Memory*) sebagai tempat menyimpan instruksi yang dijalankan dan data-data yang diperlukan, ROM (*Read-Only Memory*) sebagai tempat dimana program yang sebenarnya disimpan, jalur I/O (*Input/Output*) adalah jalur yang digunakan *microcontroller* untuk berkomunikasi dengan peralatan lain seperti sensor, LED, toggle switch dan lain-lain, timer untuk *interrupt* berdasarkan waktu, jalur antar muka (serial maupun paralel) dan terkadang terdapat ADC (*Analog-to-Digital Converter*) atau DAC (*Digital-to-Analog Converter*). *Microcontroller* banyak digunakan pada sistem kontroler karena mudah diaplikasikan ke peralatan-peralatan yang dimana ruang penempatannya sangat terbatas seperti

robot. Jenis *Microcontroller* yang digunakan dalam sistem kontroler untuk penelitian ini adalah ATMEGA8535. Bentuk dari sistem kontroler tersebut dapat dilihat pada gambar 2.6 di bawah ini.



Gambar 2.6 Mikrokontroler

Dalam penelitian ini menggunakan dua mikrokontroler, salah satu mikrokontroler lebih difungsikan untuk mengkomunikasikan data kemudian salah satunya lagi untuk mengendalikan aktuator sebagai sistem pengendali. Untuk dapat berkomunikasi dengan piranti lain contohnya mikrokontroler dengan komputer, mikrokontroler dengan mikrokontroler atau dengan yang lain. Mikrokontroler dilengkapi dengan fasilitas komunikasi, ada 2 jenis fasilitas komunikasi yang dikenal, yaitu komunikasi paralel dan komunikasi serial. Sesuai dengan namanya pada komunikasi paralel transfer data dilakukan secara serempak/bersamaan, sedangkan pada komunikasi serial data dikirim secara bergantian. Komunikasi secara paralel memiliki kelebihan pada kecepatan transfer data, tetapi kualitas suatu komunikasi tidak hanya ditentukan oleh kecepatannya saja, ada faktor lain yang perlu diperhatikan yaitu jarak dan efisiensi. Komunikasi paralel memerlukan jalur data yang lebih banyak, yang berarti pengkabelan (*wiring*) juga akan semakin banyak. Pada komunikasi serial biasanya hanya dibutuhkan 2 sampai 3 kabel saja, jadi bisa dikatakan komunikasi serial lebih praktis dibanding paralel apalagi jika komunikasi dilakukan dengan jarak yang jauh.

Agar komunikasi serial dapat berjalan dengan baik dibutuhkan suatu protocol/aturan komunikasi. Pada sistem kontroler berbasis ATMEGA8535 yang

digunakan pada penelitian ini terdapat beberapa protocol komunikasi serial, yaitu USART, SPI, dan I2C. Protocol komunikasi serial ATMEGA8535 yang digunakan dalam penelitian ini adalah USART (*universal synchronous and asynchronous serial receiver and transmitter*), karena protokol tersebut dapat menunjang pemakaian sensor kamera yang digunakan dalam penelitian ini.

USART adalah piranti komunikasi serial yang mempunyai kefleksibelan tinggi dan mempunyai fitur sebagai berikut :

1. Operasi *full duplex* (register pengiriman dan penerimaan data serial yang independen).
2. Mode sinkron dan asinkron.
3. Operasi sinkron dengan clock dari master atau slave.
4. *Baud rate* generator resolusi tinggi.
5. Mendukung *frame* serial dengan 5,6,7,8, or 9 bit data dan 1 atau 2 bit stop.
6. Pembangkitan parity ganjil atau genap dan pengecek *bit* parity oleh perangkat keras.
7. Deteksi data *over run*
8. Deteksi *framing error*.
9. Peredaman *noise* termasuk pendeteksi kesalahan bit start dan digital low pass filter.
10. 3 buah interupsi yang terpisah pada TX complete, TX data register empty dan RX complete.
11. Mode komunikasi multi-processor.
12. Mode komunikasi asinkron kecepatan ganda.

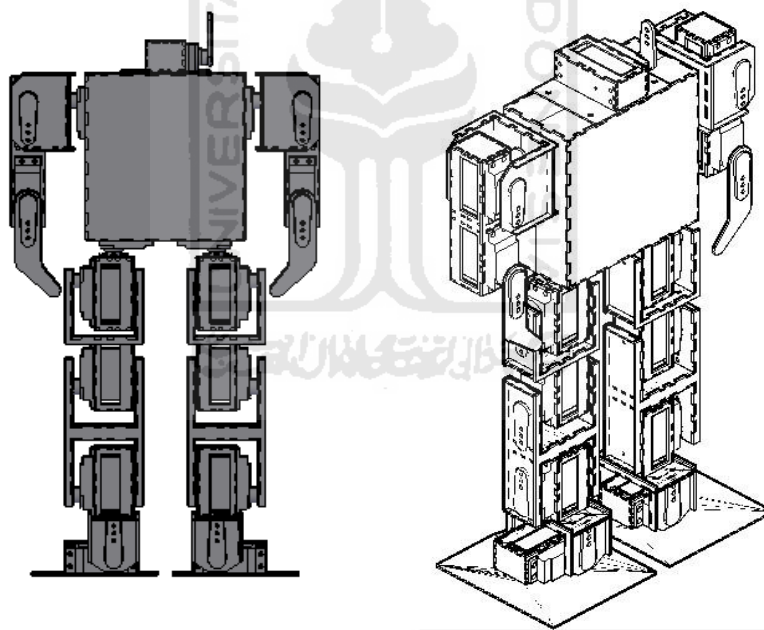
Beberapa fitur di atas dapat digunakan oleh sistem kontroler untuk berkomunikasi dengan sensor, Sehingga dengan komunikasi tersebut, akan didapatkan data-data yang diperlukan oleh sistem kontroler untuk menjalankan seluruh sistem yang ada pada robot agar berjalan dengan baik sesuai perintah yang telah diberikan.

BAB III

METODOLOGI PENELITIAN

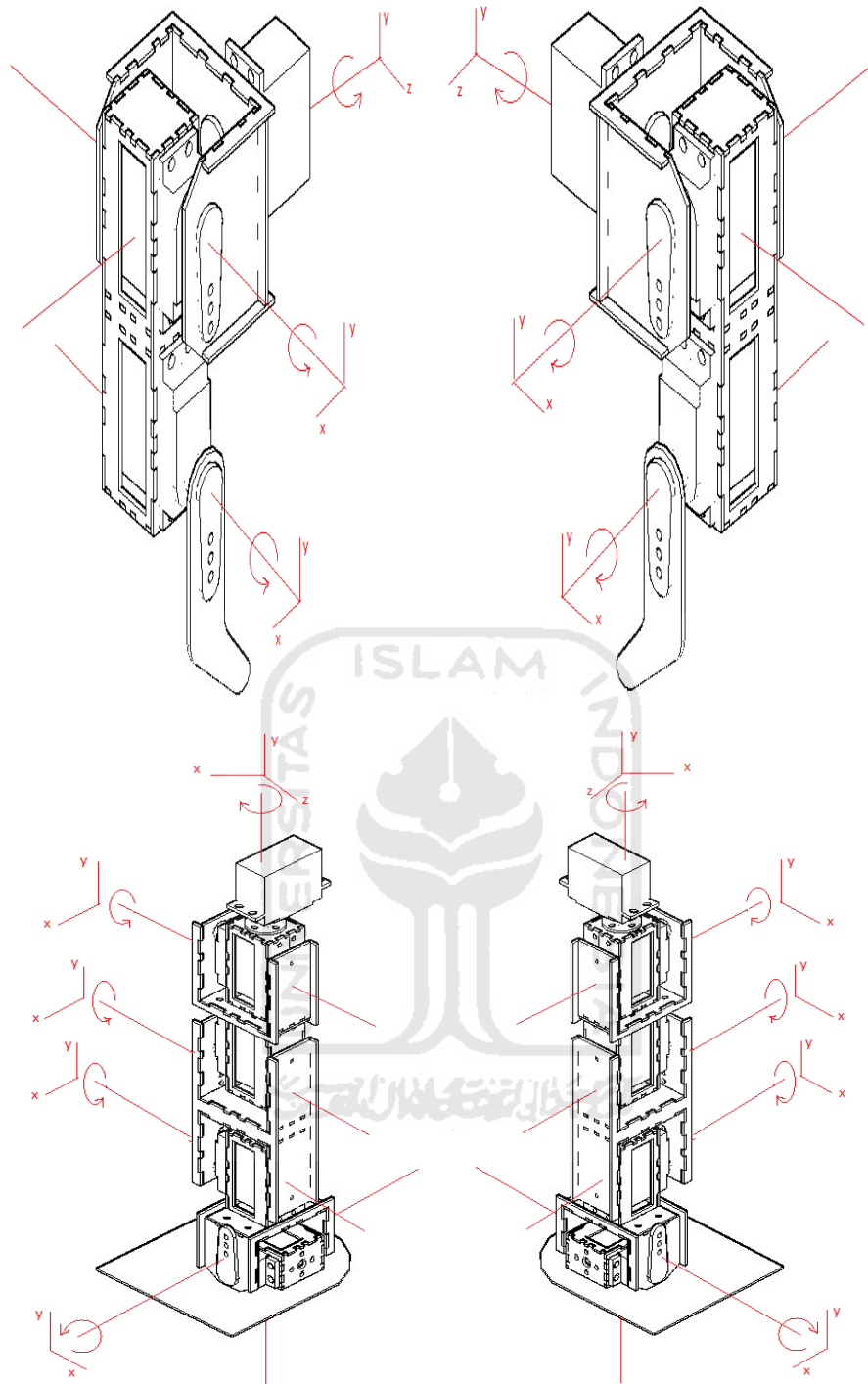
3.1 Pendahuluan

Sistem gerakan pada robot berkaki ini diharapkan mampu atau setidaknya mendekati sistem berjalan / gerak pada manusia sesungguhnya, dari setiap gerakan sendi tangan maupun kaki. Konstruksi mekanik robot didasarkan pada perancangan dan bentuk mekanik dari masing-masing servo (Endik, 2011). Penggunaan empat buah motor servo pada setiap kaki, diibaratkan layaknya sendi pada kaki manusia, tiga buah servo pada masing-masing tangan, sebuah servo pada bagian kepala, sehingga seluruh servo yang digunakan berjumlah 17 buah, seperti ditunjukkan pada gambar 3.1 dan gambar 3.2 di bawah ini.



Gambar 3.1 Bentuk Konstruksi Mekanik Robot Berkaki Dua

Sumber: Endik (2011)

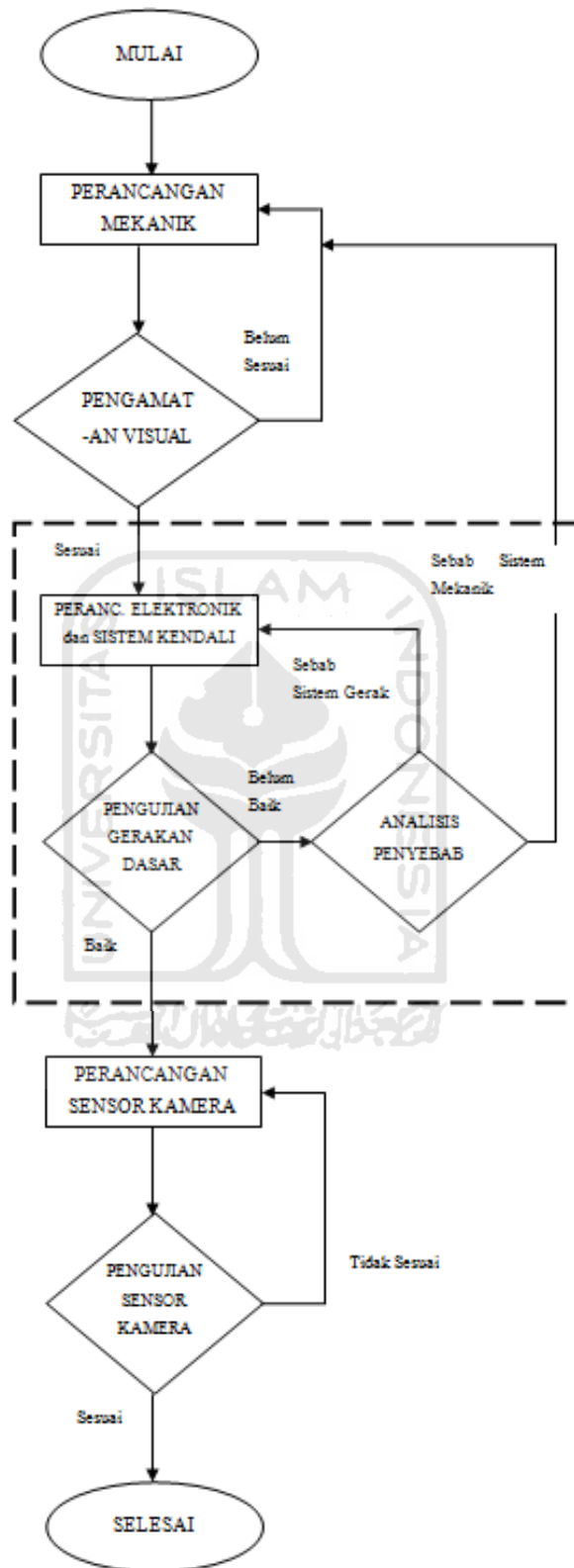


Gambar 3.2 Konstruksi Robot dan Arah Gerak Tiap Motor Servo

Sumber: Endik (2011)

Dalam perancangan sebuah robot humanoid terdiri dari beberapa tahapan sehingga memudahkan dalam menganalisis dan melakukan perbaikan baik pada sistem mekanik,

perangkat lunak maupun pada sistem sensor robot. Gambar 3.3 menunjukkan *flowchart* / diagram alir dari perancangan robot berkaki.

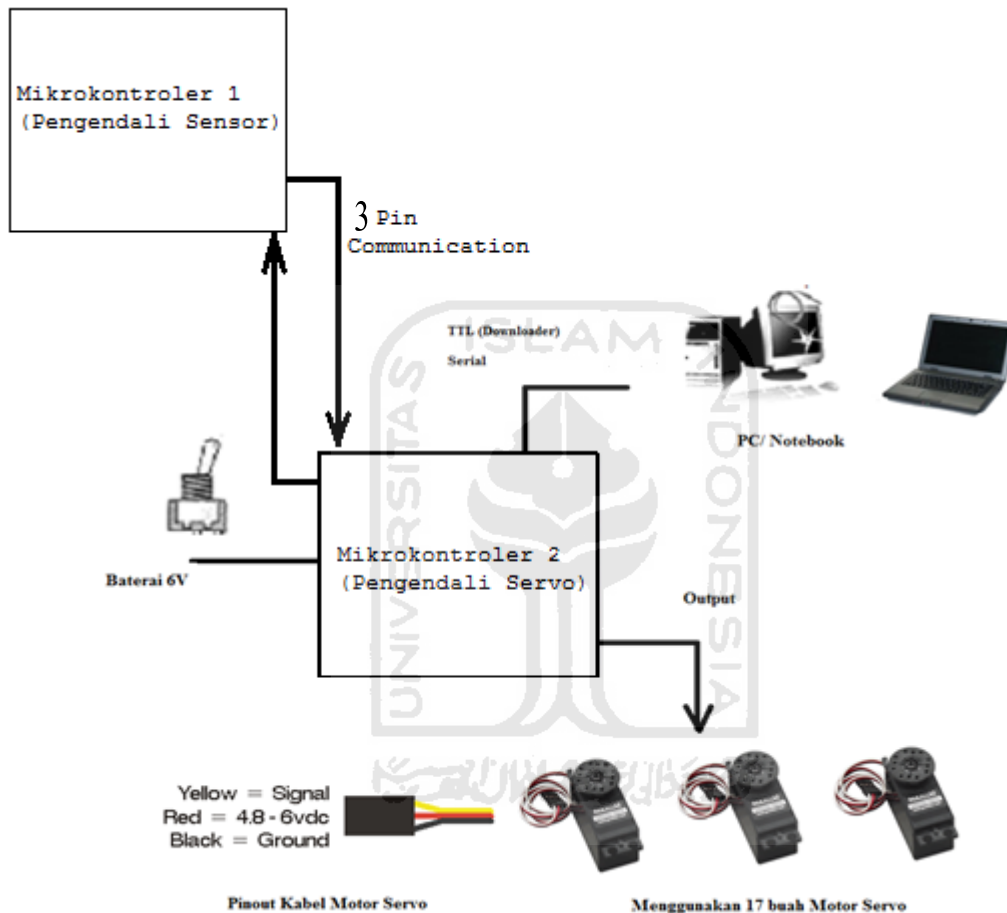


Gambar 3.3 *Flowchart* Tahapan Proses Pembuatan Robot Berkaki

Kemudian dalam pembahasan ini difokuskan pada proses perancangan sistem kendali untuk robot berkaki dua dengan sepuluh derajat kebebasan (gambar 3.3).

3.2 Perancangan Elektronik

Secara sederhana, perancangan sistem elektrik pada robot di tunjukkan pada gambar 3.4 berikut.



Gambar 3.4 Diagram Perancangan Elektronika Robot

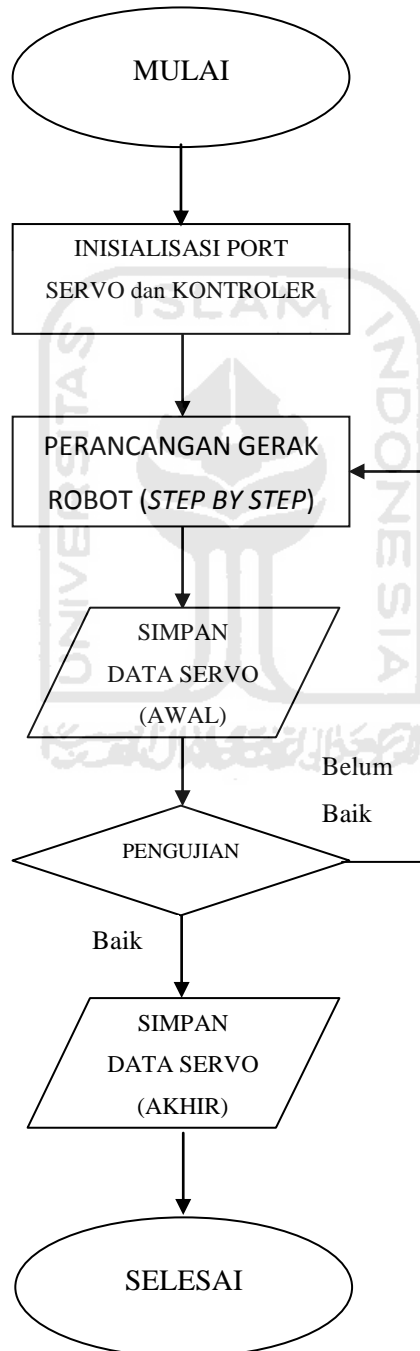
Sesuai dengan gambar diagram, mikrokontroler memberi sinyal pada output yaitu motor servo pada setiap bagian robot. Robot dua kaki ini menggunakan dua mikrokontroler, masing-masing untuk mengendalikan motor servo dan sensor kamera. Pemisahannya ditujukan agar mikrokontroler dapat lebih optimal dan menghemat memori sehingga eksekusi program dapat lebih cepat.

Mikrokontroler ATMEGA 16 mempunyai Port I/O berjumlah 32. Pada rangkaian ini menggunakan 17 port sesuai jumlah servo yang digunakan. Selain itu, rangkaian menggunakan dua komunikasi dengan komputer, yaitu sebagai *downloader* program ke

mikrokontroler melalui SPI Port (Mosi, Miso, SCK) dan komunikasi serial (USART) sebagai tampilan nilai motor servo melalui port serial mikrokontroler (Rx-Tx).

3.3 Diagram Alir Perancangan Sistem Kendali

Perancangan sistem kendali bertujuan agar robot dapat melakukan gerakan-gerakan dasar seperti maju, mundur, belok kiri, belok kanan dengan baik. Diagram Alir dari perancangan sistem kendali robot kaki dua ditunjukkan pada gambar 3.5 di bawah ini.



Gambar 3.5 Diagram Alir Perancangan Sistem Kendali Robot

3.3.1 Inisialisasi Servo dan Kontroler

Inisialisasi port servo dan kontroler merupakan proses inisialisasi kontroler terhadap port servo pada mikrokontroler. Hal ini diperlukan selain agar proses perancangan gerak lebih cepat, juga bertujuan mempermudah dan lebih teratur.

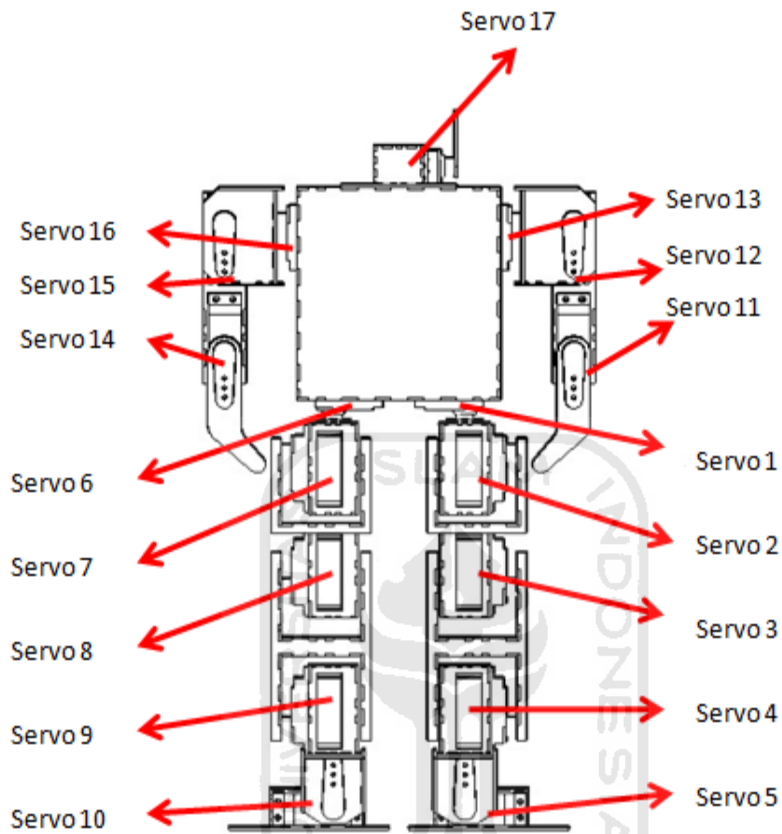
Kontroler yang digunakan adalah mikrokontroler ATMEGA 16/8535, dan sebuah pengendali buatan berupa pad *keyboard* komputer/ notebook sebagai pengontrol putaran motor servo, yaitu tombol-tombol yang tidak berfungsi khusus sebagai *shortcut* pada umumnya.

Untuk pengendali ini, masing-masing servo dikendalikan dengan dua tombol, pada arah jarum jam dan sebaliknya. Tabel 3.1 menunjukkan inisialisasi kontroler terhadap servo pada robot. Sedangkan gambar 3.6 menunjukkan penomoran motor servo pada bagian robot.

Tabel 3.1 Inisialisasi Kontroler (serta *Keyboard*) Terhadap Servo pada Robot.

No	Nomor Servo	Bagian pada Robot	Pin pada Mikrokontroler	Tombol Pengendali pada <i>Keyboard</i>
1	Servo 1	Kaki Kanan	PORTD.6	1 dan 2
2	Servo 2		PORTD.7	3 dan 4
3	Servo 3		PORTC.6	5 dan 6
4	Servo 4		PORTC.1	7 dan 8
5	Servo 5		PORTC.0	9 dan 0
6	Servo 6	Kaki Kiri	PORTA.4	Q dan W
7	Servo 7		PORTA.3	E dan R
8	Servo 8		PORTA.2	T dan Y
9	Servo 9		PORTA.1	U dan I
10	Servo 10		PORTA.0	O dan P
11	Servo 11	Tangan Kanan	PORTD.5	A dan S
12	Servo 12		PORTD.4	D dan F
13	Servo 13		PORTC.7	G dan H
14	Servo 14	Tangan Kiri	PORTA.5	J dan K

15	Servo 15		PORTA.6	Z dan X
16	Servo 16		PORTA.7	C dan V
17	Servo 17	Kepala	PORTD.3	N dan M

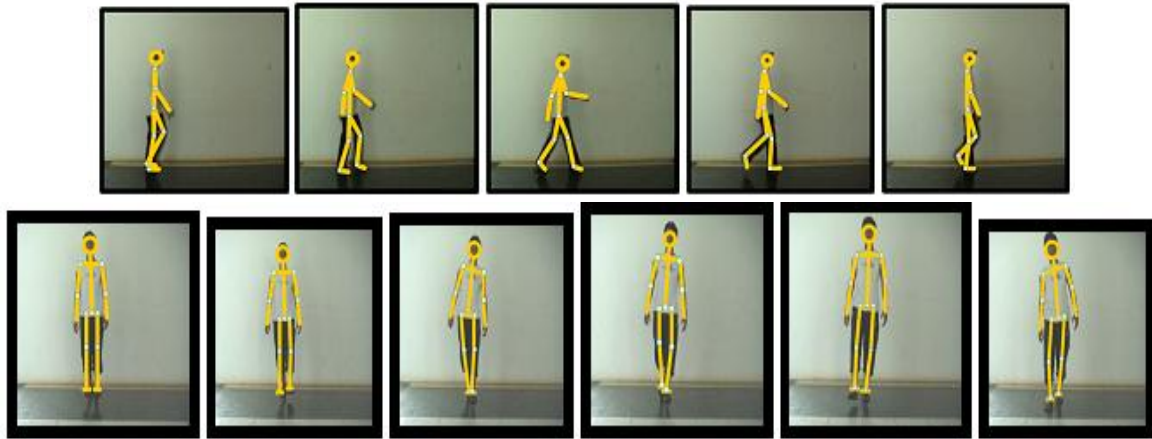


Gambar 3.6 Penomoran Motor Servo pada Robot

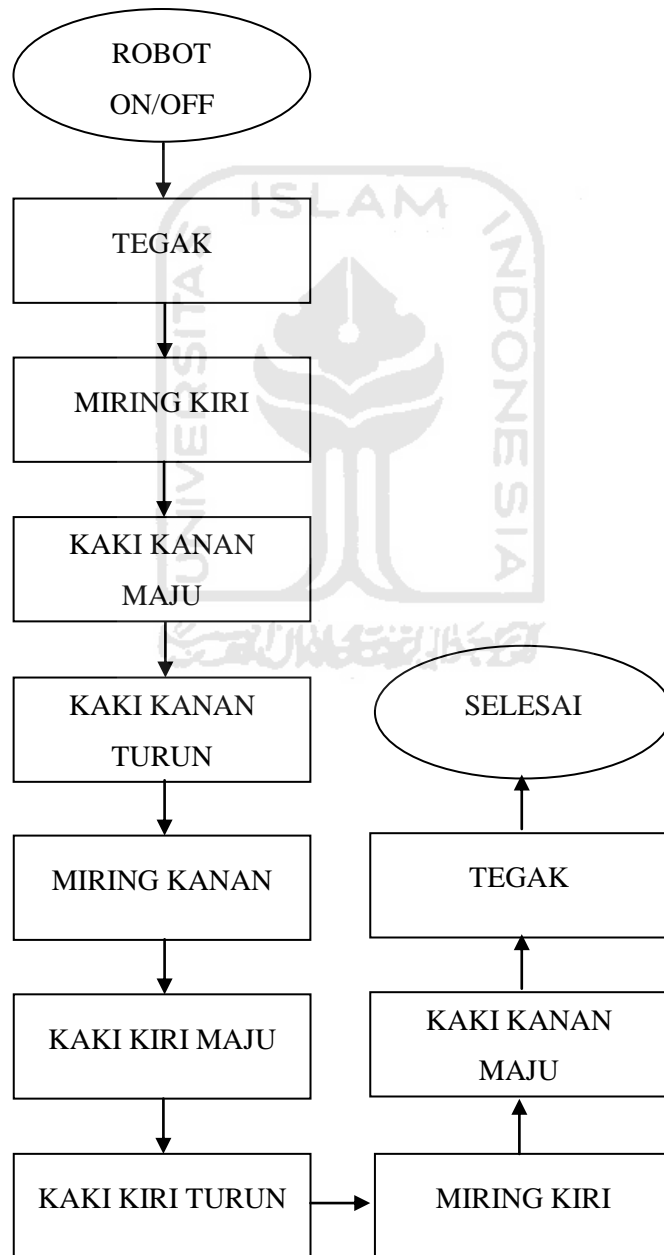
3.3.2 Perancangan Gerak Robot

Perancangan gerak robot yang dimaksud adalah proses pengendalian tiap motor servo, baik urutannya, seberapa lama, maupun posisi tiap motor servo sehingga menghasilkan gerakan robot maju-mundur, kiri-kanan, berputar, dan geser.

Perancangan gerak robot disesuaikan sedemikian sehingga mendekati atau sama dengan pergerakan pada manusia pada umumnya. Setelah sebelumnya mengamati cara berjalan maju, mundur dan belok pada manusia.



Gambar 3.7 Langkah Gerakan Maju pada Manusia (Tampak Samping dan Depan).



Gambar 3.8 Gerak Robot Maju 2 langkah

Pada diagram alir tersebut merupakan bahasa sehari-hari dari urutan gerakan manusia saat berjalan maju, bahasa tersebut merupakan bahasa sederhana yang dapat dimengerti.

Dari data tersebut kemudian diterjemahkan ke dalam bahasa pemrograman pada mikrokontroler dengan mengatur posisi motor servo pada setiap langkah gerakan robot. Posisi tiap motor servo pada setiap langkah gerakannya saat masa pengujian akan disimpan sementara, sebagai acuan untuk pengaturan selanjutnya.

Kemudian hasil pengujian yang telah baik, dalam hal ini parameternya adalah robot dapat melakukan gerakan keseluruhan tanpa jatuh (seimbang), akan menjadi data servo akhir yang merupakan hasil dari perancangan gerak robot. Kemudian diambil contoh hasil perancangan pada gerak robot maju dua langkah karena gerakan maju merupakan gerakan yang paling dasar dalam perancangan sebuah robot.

Tabel 3.2 Nilai Servo pada Perancangan Gerak Robot Maju 2 Langkah

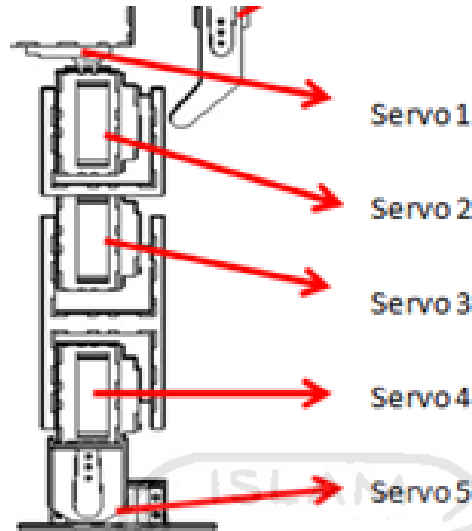
No	Nomor Servo	Bagian pada Robot	Nilai Motor Servo									
			Tegak	Miring Kiri	Kaki Kanan Maju	Kaki Kiri Turun	Miring Kanan	Kaki Kiri Maju	Kaki Kiri Turun	Miring Kiri	Kaki Kanan Maju	Tegak
1	Servo 1	Kaki Kanan	1520	1520	1520	1520	1520	1520	1520	1520	1520	1520
2	Servo 2		1840	1840	1880	1880	1880	1880	1880	1880	1900	1840
3	Servo 3		2280	2280	2280	2280	2280	2280	2280	2280	2280	2280
4	Servo 4		2180	2180	2270	2270	2270	2270	2270	2270	2260	2180
5	Servo 5		2120	1970	1970	2120	2270	2270	2120	1970	1970	2120
6	Servo 6	Kaki Kiri	2580	2580	2580	2580	2580	2580	2580	2580	2580	2580
7	Servo 7		2110	2110	2110	2110	2110	2070	2070	2070	2070	2110
8	Servo 8		1680	1680	1680	1680	1680	1680	1680	1680	1680	1680
9	Servo 9		1900	1900	1900	1900	1900	1850	1850	1850	1850	1900
10	Servo 10		1870	1720	1720	1870	2020	2020	1870	1720	1720	1870
11	Servo 11	Tangan Kanan	2010	2010	2010	2010	2010	2010	2010	2010	2010	2010
12	Servo 12		2910	2910	2910	2910	2910	2910	2910	2910	2910	2910
13	Servo 13		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000

15	Servo 15		1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000

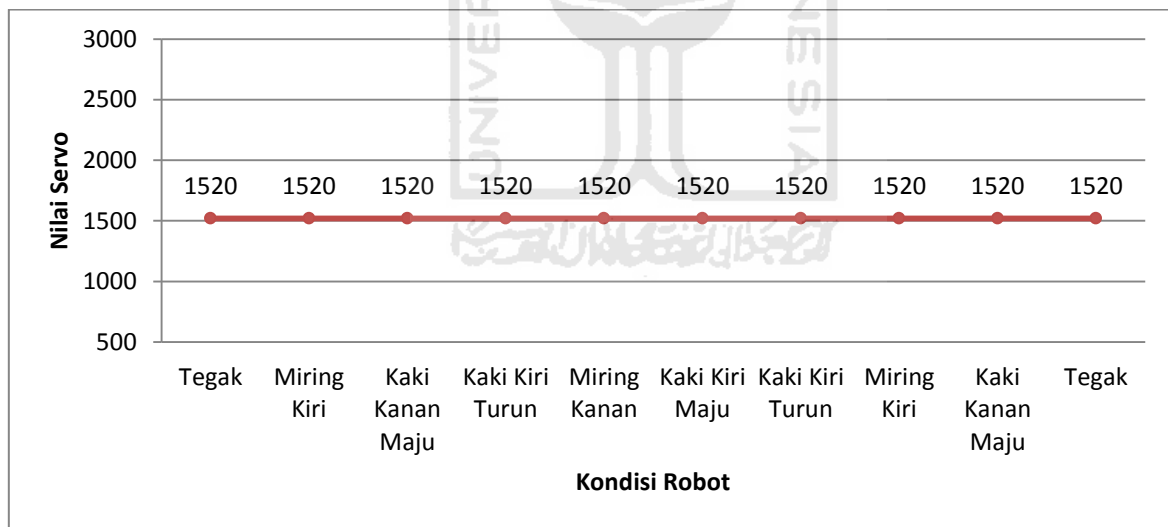
Tabel 3.3 Nilai Sudut Servo pada Perancangan Gerak Robot Maju 2 Langkah

No	Nomor Servo	Bagian pada Robot	Nilai Motor Servo									
			Tegak	Miring Kiri	Kaki Kanan Maju	Kaki Kiri Turun	Miring Kanan	Kaki Kiri Maju	Kaki Kiri Turun	Miring Kiri	Kaki Kanan Maju	Tegak
1	Servo 1	Kaki Kanan	47	47	47	47	47	47	47	47	47	47
2	Servo 2		76	76	79	79	79	79	79	79	81	76
3	Servo 3		115	115	115	115	115	115	115	115	115	115
4	Servo 4		106	106	114	114	114	114	114	114	113	106
5	Servo 5		101	87	87	101	114	114	101	87	87	101
6	Servo 6	Kaki Kiri	142	142	142	142	142	142	142	142	142	142
7	Servo 7		100	100	100	100	100	96	96	96	96	100
8	Servo 8		61	61	61	61	61	61	61	61	61	61
9	Servo 9		81	81	81	81	81	77	77	77	77	81
10	Servo 10		78	65	65	78	92	92	78	65	65	78
11	Servo 11	Tangan Kanan	91	91	91	91	91	91	91	91	91	91
12	Servo 12		172	172	172	172	172	172	172	172	172	172
13	Servo 13		90	90	90	90	90	90	90	90	90	90
14	Servo 14	Tangan Kiri	90	90	90	90	90	90	90	90	90	90
15	Servo 15		0	0	0	0	0	0	0	0	0	0
16	Servo 16		90	90	90	90	90	90	90	90	90	90
17	Servo 17	Kepala	90	90	90	90	90	90	90	90	90	90

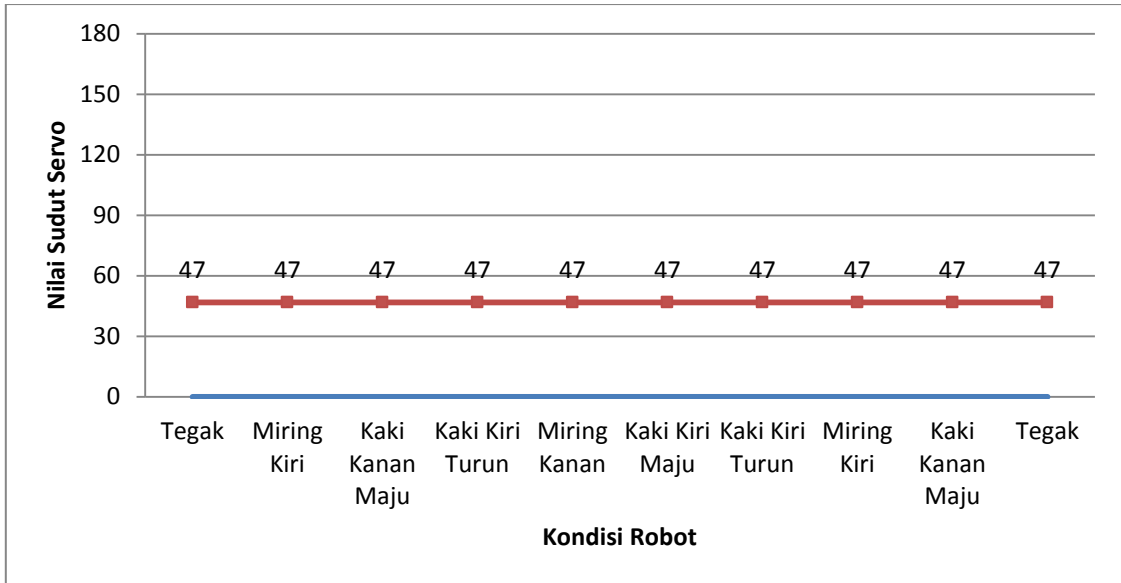
Kemudian dari data pada tabel 3.2 dan 3.3 dapat dibuat grafik yang menunjukkan perancangan gerak tiap motor servo pada kaki kanan robot sesuai perancangan gerak robot maju dua langkah.



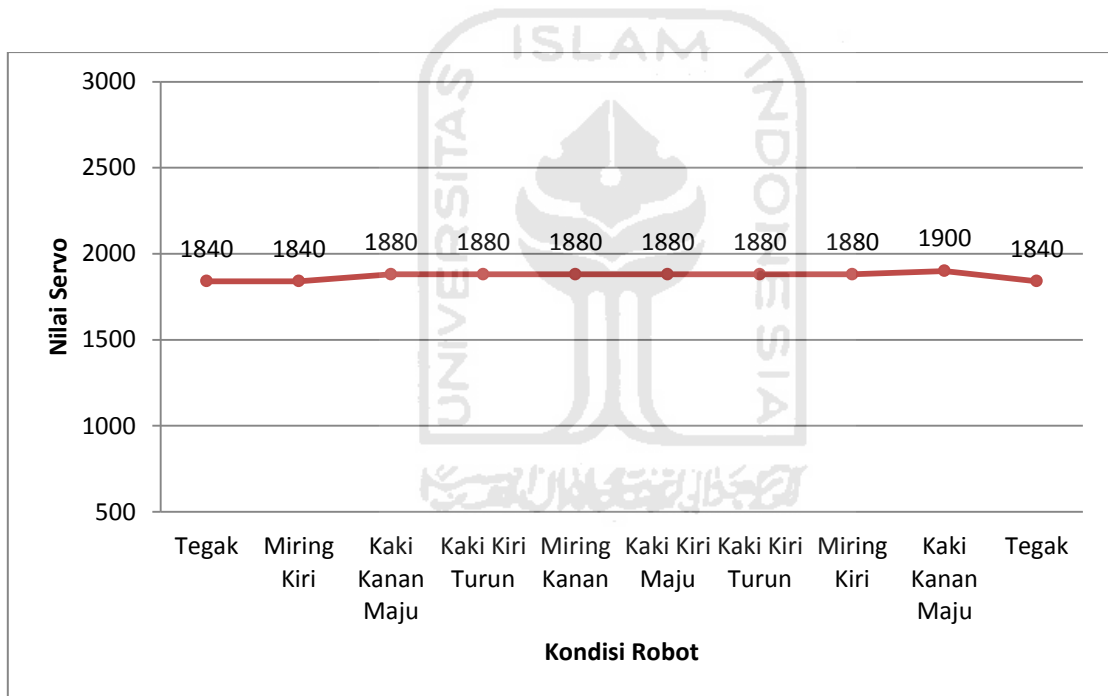
Gambar 3.9 Penomoran Servo pada Kaki Kanan Robot



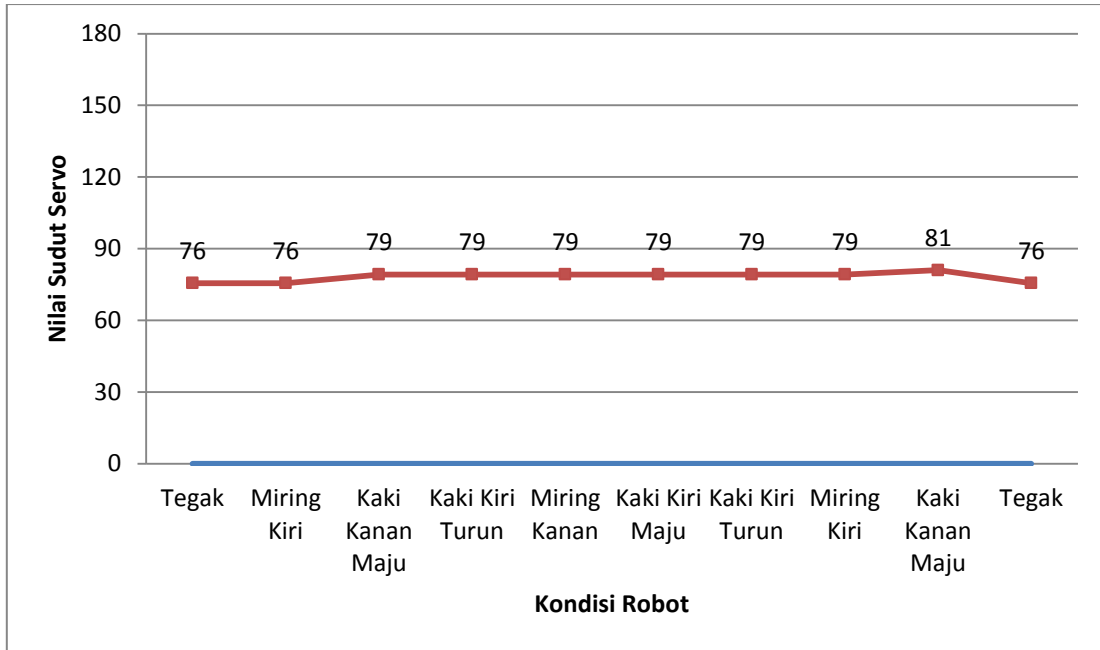
Grafik 3.1 Grafik Nilai Servo 1 pada Perancangan Gerak Robot Maju 2 Langkah



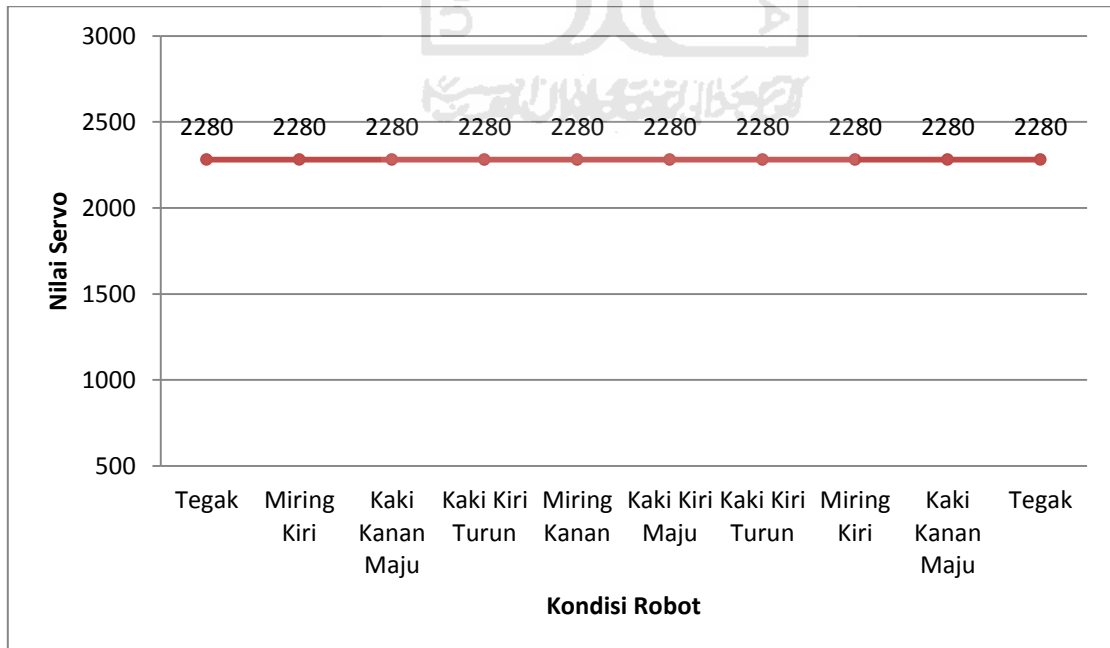
Grafik 3.2 Grafik Nilai Sudut Servo 1 pada Perancangan Gerak Robot Maju 2 Langkah



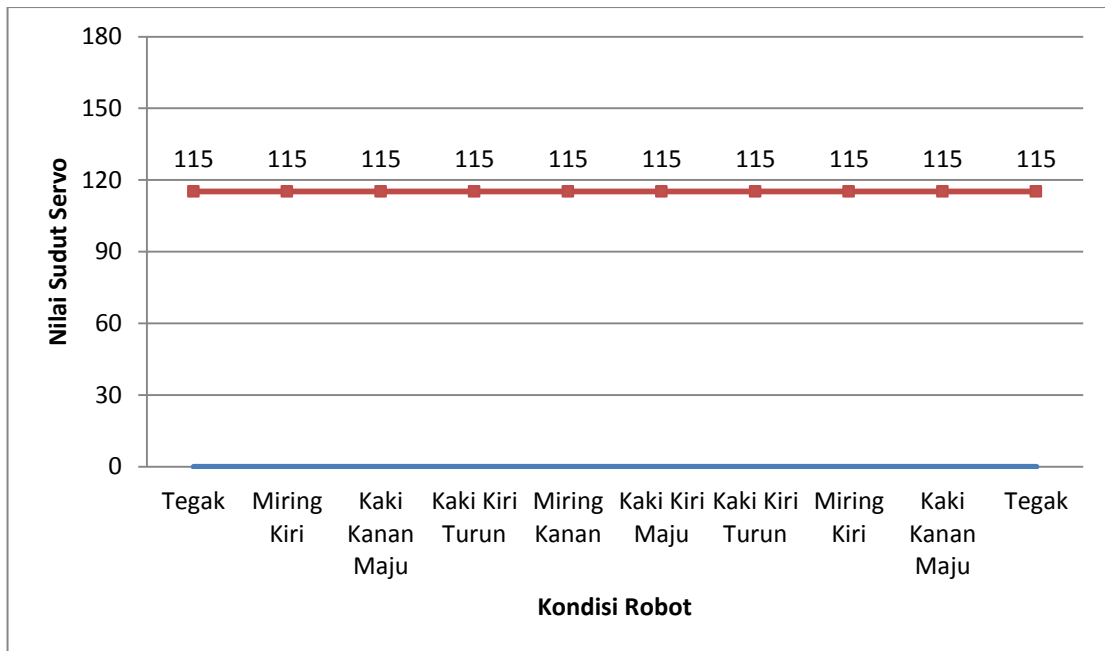
Grafik 3.3 Grafik Nilai Servo 2 pada Perancangan Gerak Robot Maju 2 Langkah



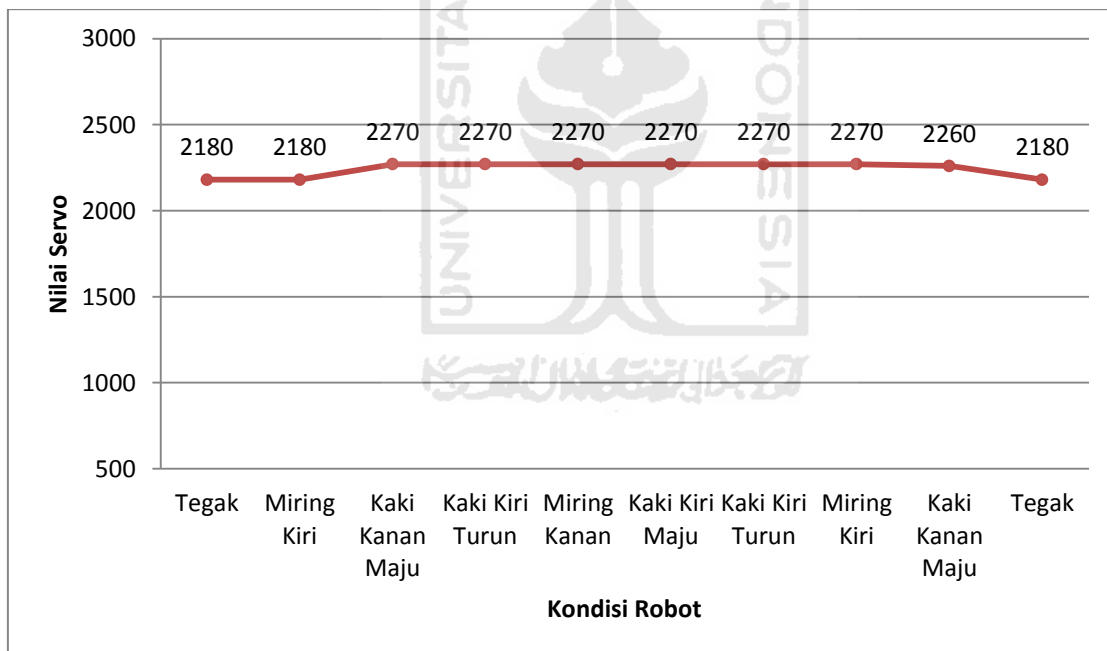
Grafik 3.4 Grafik Nilai Sudut Servo 2 pada Perancangan Gerak Robot Maju 2 Langkah



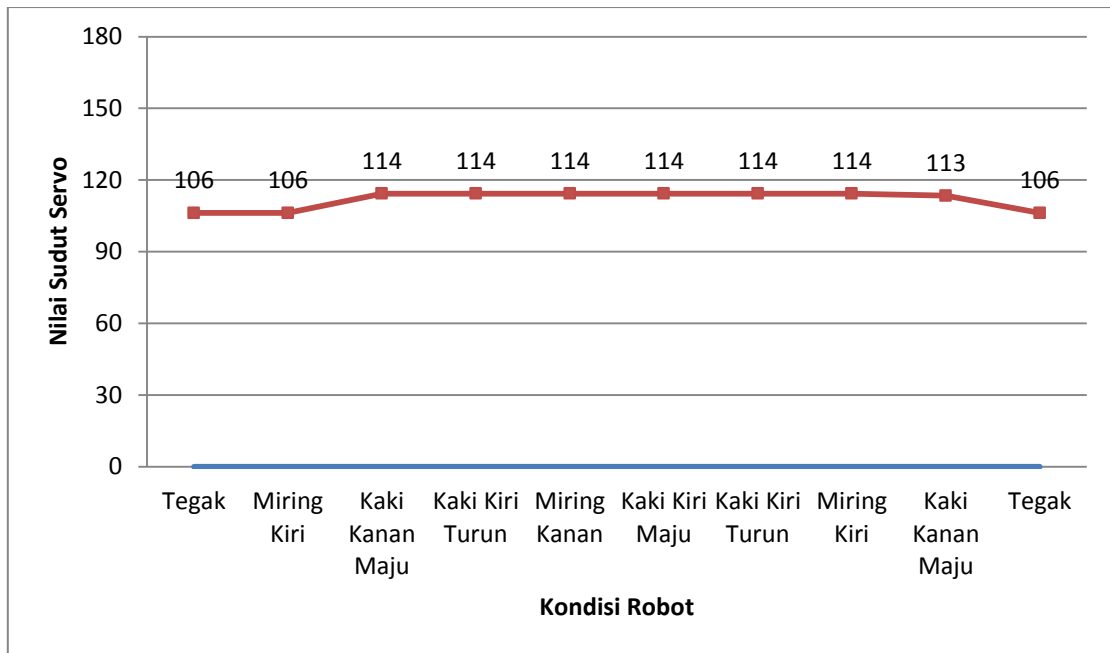
Grafik 3.5 Grafik Nilai Servo 3 pada Perancangan Gerak Robot Maju 2 Langkah



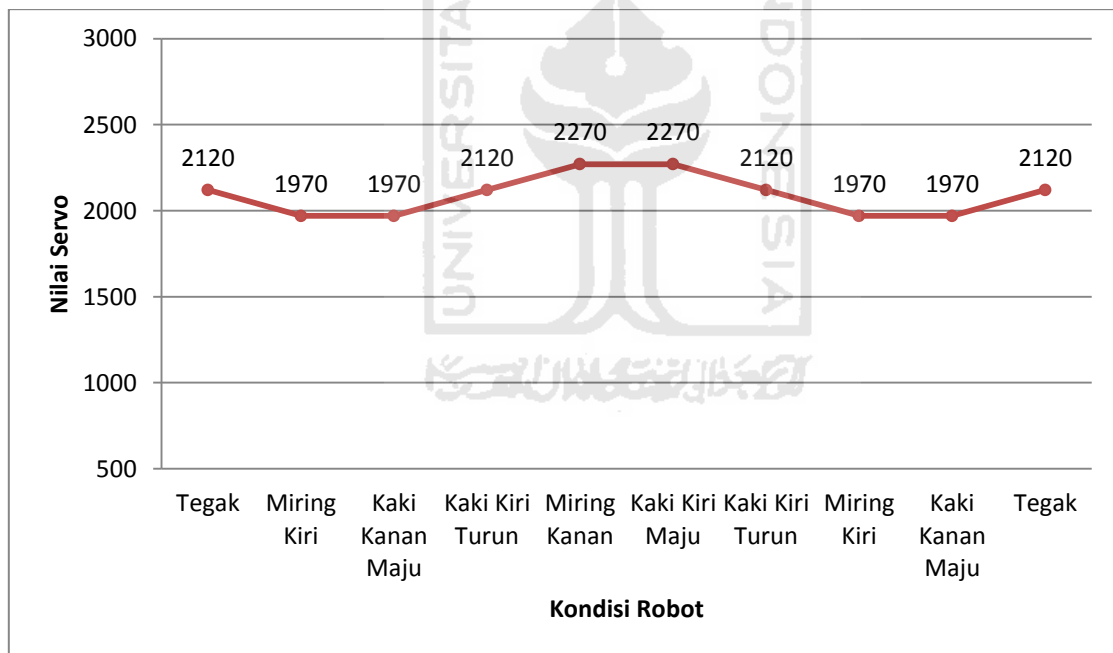
Grafik 3.6 Grafik Nilai Sudut Servo 3 pada Perancangan Gerak Robot Maju 2 Langkah



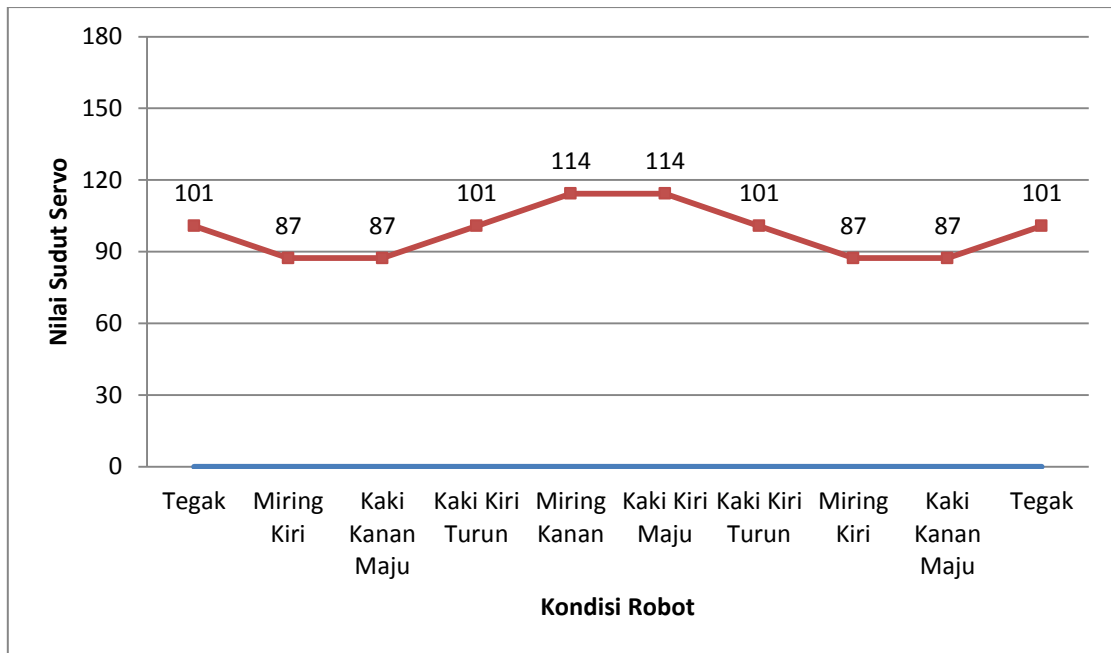
Grafik 3.7 Grafik Nilai Servo 4 pada Perancangan Gerak Robot Maju 2 Langkah



Grafik 3.8 Grafik Nilai Sudut Servo 4 pada Perancangan Gerak Robot Maju 2 Langkah

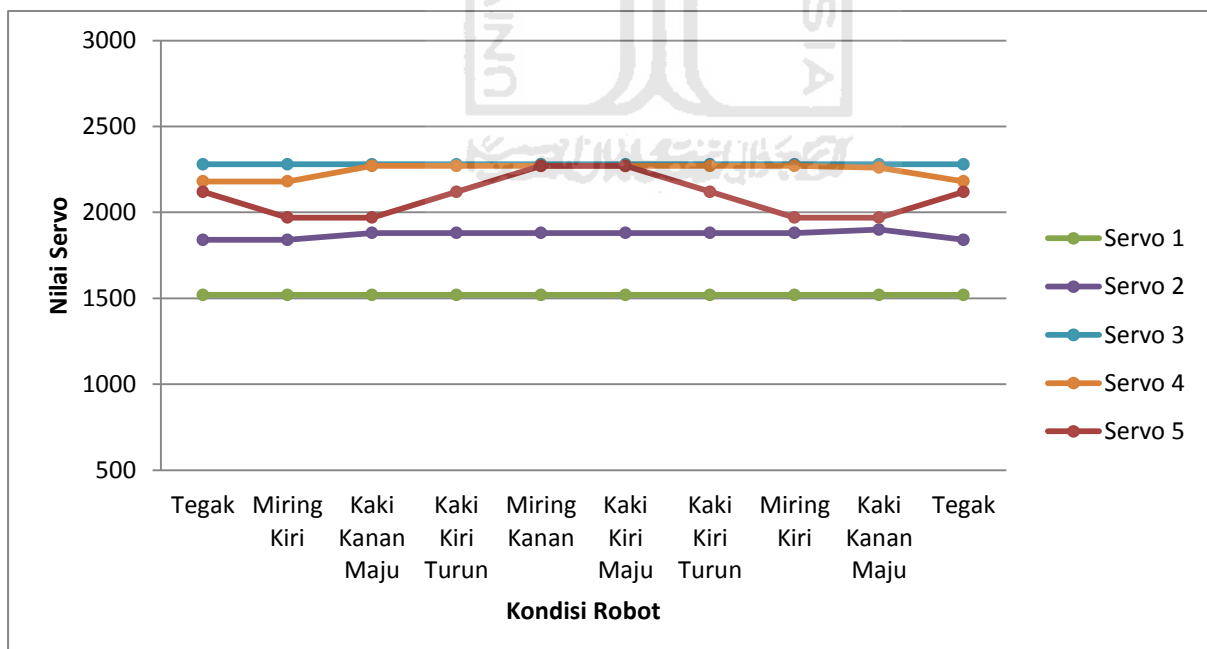


Grafik 3.9 Grafik Nilai Servo 5 pada Perancangan Gerak Robot Maju 2 Langkah

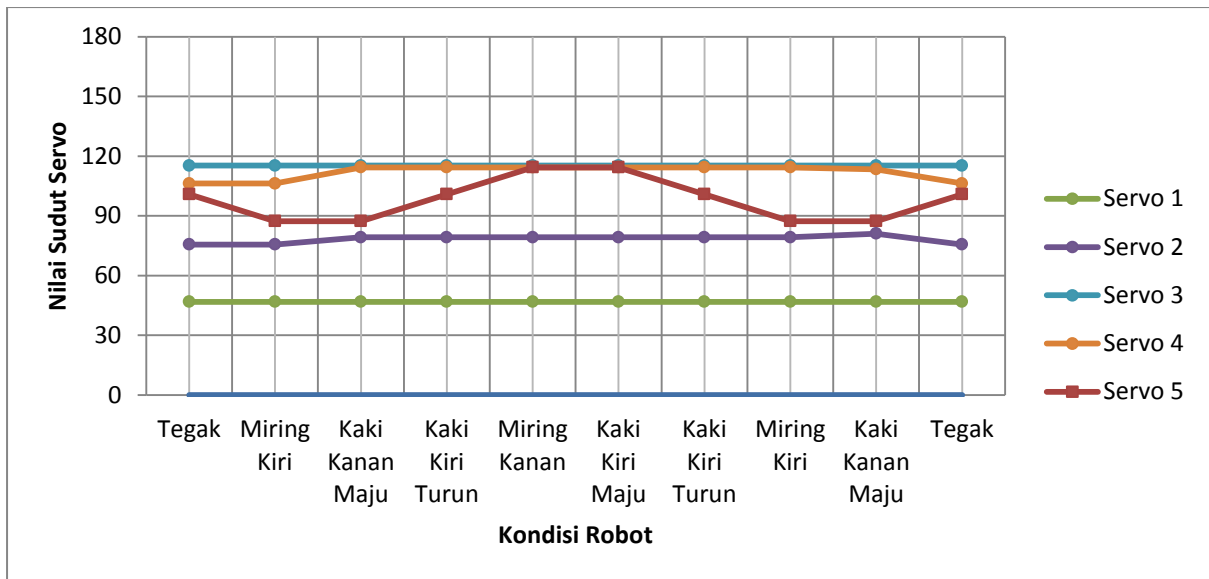


Grafik 3.10 Grafik Nilai Sudut Servo 5 pada Perancangan Gerak Robot Maju 2 Langkah

Dari grafik 3.1 – grafik 3.10 dapat disatukan menjadi grafik yang menggambarkan perubahan nilai servo dan nilai sudut servo pada kaki kanan robot pada perancangan gerak maju dua langkah robot.



Grafik 3.11 Nilai Servo pada Kaki Kanan Robot pada Gerak Maju Dua Langkah Robot



Grafik 3.12 Nilai Sudut Servo pada Kaki Kanan Robot pada Gerak Maju Dua Langkah Robot

```

81 void jln(){
82 //////////////////////////////////////////////////////////////////1////////////////////////////////////////////////////////////////
83 //tangan kanan
84 a11=2010;
85 a12=2910;
86 a13=2000;
87 //kaki kanan
88 a1=1520;
89 a2=1840;
90 a3=2280;
91 a4=2180;
92 a5=2120;
93
94 //tangan kiri
95 a14=2000;
96 a15=1000;
97 a16=2000;
98 //kaki kiri
99 a6=2580; |
100 a7=2110;
101 a8=1680;
102 a9=1900;
103 a10=1870;
104
105 a17=2000;
106 delay_ms(b);
107
108 //////////////////////////////////////////////////////////////////2////////////////////////////////////////////////////////////////
109 //tangan kanan
110 a11=2010;
111 a12=2910;
112 a13=2000;
113 //kaki kanan
114 a1=1500;
115 a2=1930;

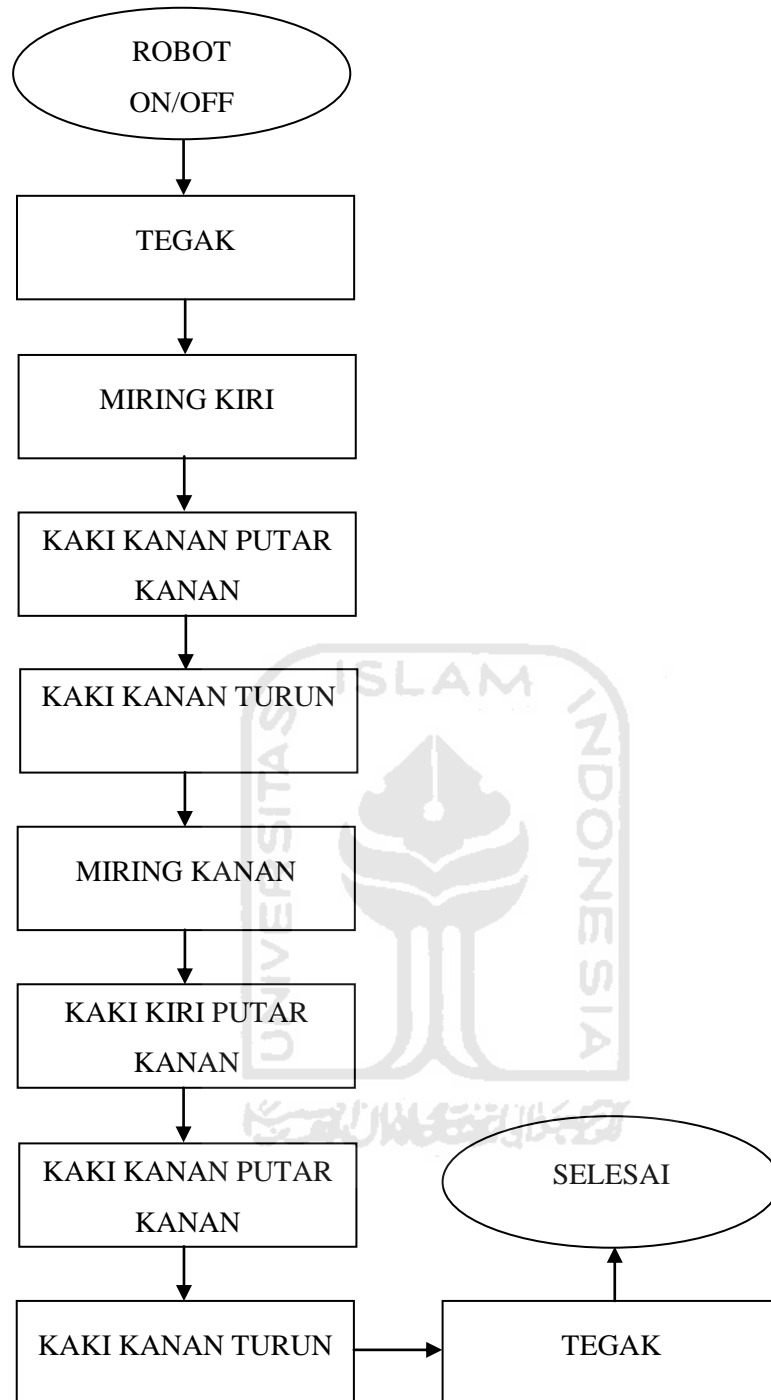
```

Gambar 3.10 Contoh Program Gerak Maju 2 Langkah pada Robot

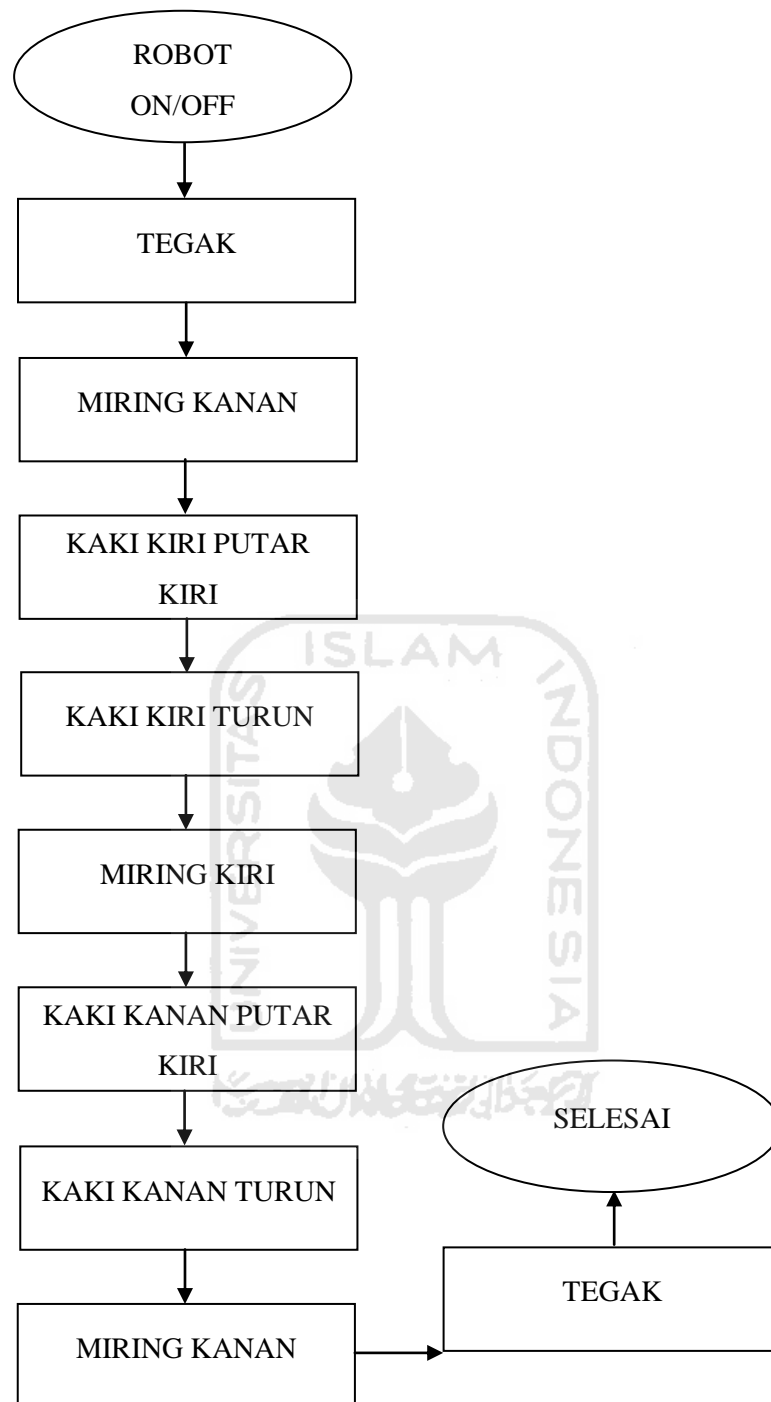
Sedangkan untuk diagram alir perancangan gerakan mundur, belok kanan, belok kiri, masing-masing ditunjukkan pada gambar 3.11, gambar 3.12, dan gambar 3.13.



Gambar 3.11 Gerak Robot Mundur 2 Langkah



Gambar 3.12 Gerak Robot Belok Kanan 90°



Gambar 3.12 Gerak Robot Belok Kiri 90°

Dari diagram-diagram alir tersebut, hasilnya dapat ditentukan seperti pada contoh untuk gerakan maju. Sehingga gerakan robot dua kaki akan lebih mudah dirancang, dipahami dan dikoreksi jika terdapat kesalahan.

BAB IV

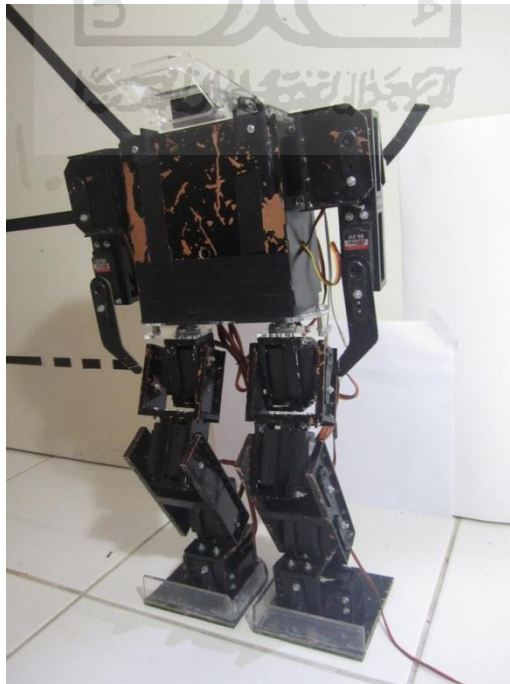
HASIL DAN PEMBAHASAN

Perancangan robot dilakukan secara bertahap, sesuai dengan tingkat keberhasilan perancangan sebelumnya yaitu perancangan mekanik robot. Perancangan sistem kendali robot dua kaki menggunakan metode praktis berupa pengujian-pengujian secara langsung dan pengendaliannya secara manual. Dalam penelitian ini didapat banyak permasalahan, baik dalam perancangan maupun pengujian gerak robot.

Hasil perancangan berupa nilai-nilai input pada mikrokontroler untuk memposisikan motor servo pada keadaan-keadaan tertentu sesuai langkah-langkah pada perancangan gerak robot. Nilai-nilai tersebut diidentifikasi sebagai data servo akhir.

4.1 Hasil Perancangan

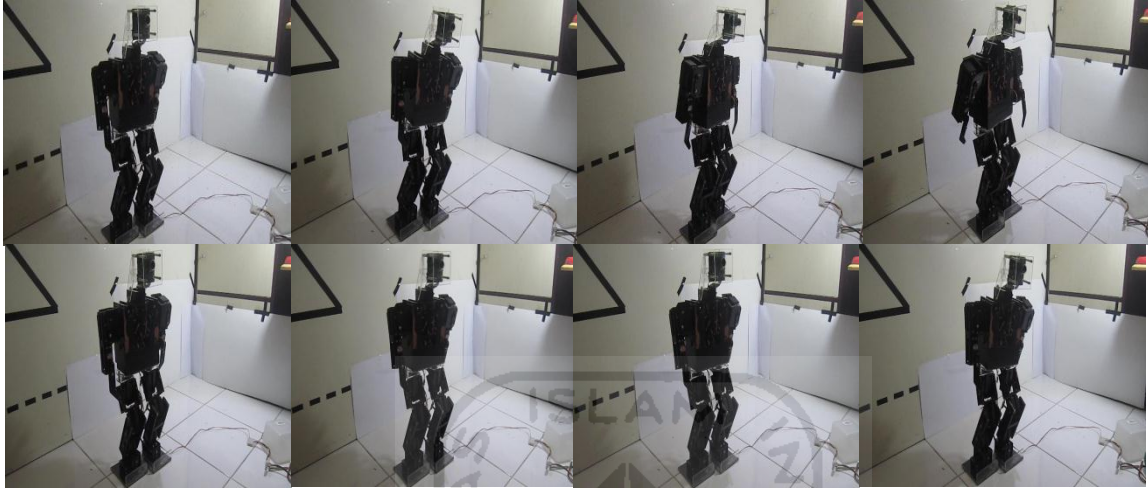
Dari hasil pengujian-pengujian dan kendala-kendala yang ditemukan, serta solusinya, robot telah sesuai dengan yang diharapkan, tentunya dengan terus dilakukan perbaikan-perbaikan ke depannya.



Gambar 4.1 Robot Hasil Perancangan Akhir

4.1.1 Gerakan Maju

Gerakan maju robot terdiri dari delapan langkah, dengan kecepatan langkah pada pengujian terukur 25 detik tiap satu keramik (pengujian pada lantai keramik yang rata). Selama melakukan gerakan jalan, robot seimbang (tidak jatuh).



Gambar 4.2 Hasil Perancangan Gerakan Maju

Tabel 4.1 Data Servo Akhir Perancangan Gerak Maju

No	Nomor Servo	Bagian pada Robot	Nilai Servo									
			Tegak	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Step 7	Step 8	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1650	1650	1650	1500	1350	1350	1350	1500
2	Servo 2		2110	2110	2110	2110	2210	2110	2060	2060	2080	2110
3	Servo 3		1990	1990	1990	1990	1990	1990	1990	1990	1990	1990
4	Servo 4		1980	1980	1980	1980	1980	1980	1980	1980	1980	1980
5	Servo 5		2130	2060	2060	2130	2280	2280	2280	2130	2060	2130
6	Servo 6	Kaki Kiri	2540	2540	2690	2690	2690	2540	2390	2390	2390	2540
7	Servo 7		2030	2030	2100	2100	2100	2080	2080	2080	2080	2030
8	Servo 8		1830	1830	1830	1830	1830	1830	1830	1830	1830	1830
9	Servo 9		1970	1970	1970	1970	1970	1970	1970	1970	1970	1970

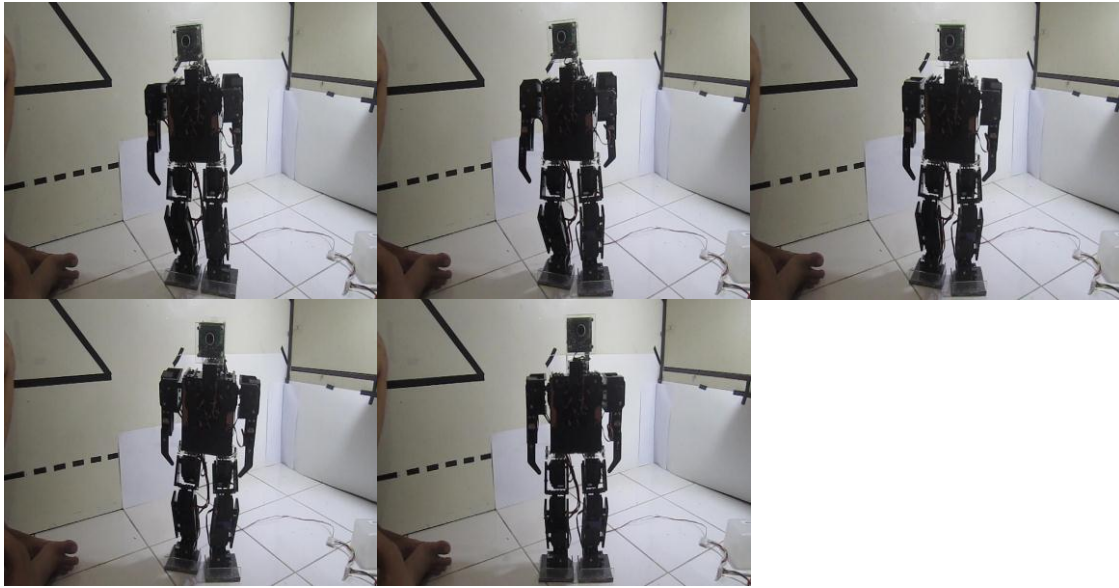
10	Servo 10		1870	1720	1720	1870	2020	1940	1940	1870	1720	1870
11	Servo 11	Tangan Kanan	2000	2000	1750	1500	1750	2000	2250	2500	2250	2000
12	Servo 12		2900	2900	2900	2900	2900	2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2000	1750	1500	1750	2000	2250	2500	2250	2000
15	Servo 15		1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000

4.1.2 Gerakan Berbelok

Gerakan berbelok baik belok kiri maupun belok kanan dari robot terdiri dari 5 langkah, dengan kecepatan belok terukur 20 detik untuk berputar 90°. Dalam gerak belok kanan maupun belok kiri, robot juga telah seimbang (tidak jatuh).



Gambar 4.3 Hasil Perancangan Gerak Belok Kanan



Gambar 4.4 Hasil Perancangan Gerak Belok Kiri

Tabel 4.2 Data Servo Akhir Perancangan Gerak Belok Kanan

No	Nomor Servo	Bagian pada Robot	Nilai Servo						
			Tegak	Step 1	Step 2	Step 3	Step 4	Step 5	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1750	1750	1750	1500	1500
2	Servo 2		2110	2190	2190	2190	2190	2190	2110
3	Servo 3		1990	1990	1990	1990	1990	1990	1990
4	Servo 4		1980	1980	1980	1980	1980	1980	1980
5	Servo 5		2130	2160	2160	2130	2130	2280	2130
6	Servo 6	Kaki Kiri	2540	2540	2540	2540	2540	2540	2540
7	Servo 7		2030	1950	1950	1950	1950	1950	2030
8	Servo 8		1830	1830	1830	1830	1830	1830	1830
9	Servo 9		1970	1970	1970	1970	1970	1970	1970
10	Servo 10		1870	1720	1720	1870	1940	1940	1870
11	Servo 11	Tangan Kanan	2000	1750	1500	2000	2250	2500	2000
12	Servo 12		2900	2900	2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000	2000	2000

14	Servo 14	Tangan Kiri	2000	1750	1500	2000	2250	2500	2000
15	Servo 15		1000	1000	1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000	2000	2000

Tabel 4.3 Data Servo Akhir Perancangan Gerak Belok Kiri

No	Nomor Servo	Bagian pada Robot	Nilai Servo						
			Tegak	Step 1	Step 2	Step 3	Step 4	Step 5	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1500	1500	1500	1500	1500
2	Servo 2		2110	2190	2150	2150	2190	2190	2110
3	Servo 3		1990	1990	1990	1990	1990	1990	1990
4	Servo 4		1980	1980	1980	1980	1980	1980	1980
5	Servo 5		2130	2280	2280	2130	2060	2060	2130
6	Servo 6	Kaki Kiri	2540	2540	2290	2290	2290	2540	2540
7	Servo 7		2030	1950	1950	1950	1840	1950	2030
8	Servo 8		1830	1830	1830	1830	1830	1830	1830
9	Servo 9		1970	1970	1970	1970	1970	1970	1970
10	Servo 10		1870	1940	1940	1870	1720	1720	1870
11	Servo 11	Tangan Kanan	2000	2250	2500	2000	1750	1500	2000
12	Servo 12		2900	2900	2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2250	2500	2000	1750	1500	2000
15	Servo 15		1000	1000	1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000	2000	2000

4.1.3 Gerakan Tendang

Gerakan menendang robot terdiri dari tiga langkah, dengan rentang jarak bola tertendang sejauh 20 cm hingga 35 cm, hal ini dipengaruhi ayunan kaki saat gerak menendang, kaki robot tidak boleh berayun terlalu ke depan, karena akan menyebabkan robot tidak stabil dan jatuh. Robot dapat menendang dengan tetap seimbang (tidak jatuh).



Gambar 4.5 Hasil Perancangan Gerak Tendang (Kanan)



Gambar 4.6 Hasil Perancangan Gerak Tendang (Kiri)

Tabel 4.4 Data Servo Akhir Perancangan Gerak Tendang (Kanan)

No	Nomor Servo	Bagian pada Robot	Nilai Servo				
			Tegak	Step 1	Step 2	Step 3	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1500	1500	1500
2	Servo 2		2110	2160	2110	2500	2110
3	Servo 3		1990	1990	1850	1850	1990
4	Servo 4		1980	1980	1980	1980	1980

5	Servo 5		2130	1940	2070	2070	2130
6	Servo 6	Kaki Kiri	2540	2540	2540	2540	2540
7	Servo 7		2030	1980	1980	2000	2030
8	Servo 8		1830	1830	1830	1830	1830
9	Servo 9		1970	1970	1970	1970	1970
10	Servo 10		1870	1800	1700	1700	1870
11	Servo 11	Tangan Kanan	2000	2000	2250	1750	2000
12	Servo 12		2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2000	2250	1750	2000
15	Servo 15		1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000

Tabel 4.5 Data Servo Akhir Perancangan Gerak Tendang (Kiri)

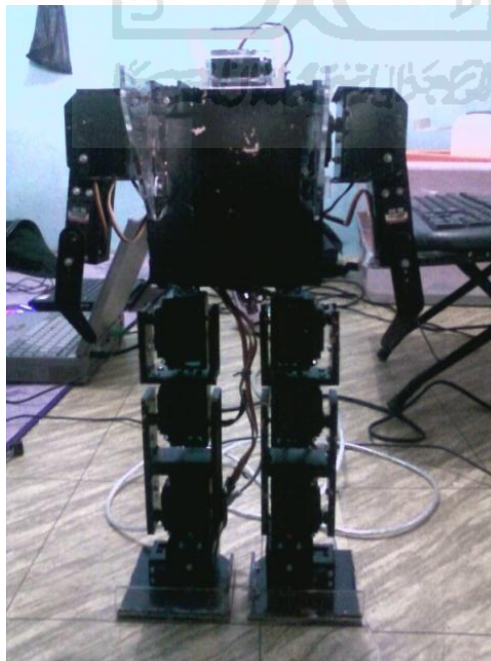
No	Nomor Servo	Bagian pada Robot	Nilai Servo				
			Tegak	Step 1	Step 2	Step 3	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1500	1500	1500
2	Servo 2		2110	2140	2140	2140	2110
3	Servo 3		1990	1990	1990	1990	1990
4	Servo 4		1980	1980	1980	1980	1980
5	Servo 5		2130	2280	2280	2280	2130
6	Servo 6	Kaki Kiri	2540	2540	2540	2540	2540
7	Servo 7		2030	2000	2000	1700	2030
8	Servo 8		1830	1830	1970	1970	1830
9	Servo 9		1970	1970	1970	1970	1970
10	Servo 10		1870	2040	1870	1870	1870

11	Servo 11	Tangan Kanan	2000	2000	1750	2250	2000
12	Servo 12		2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2000	1750	2250	2000
15	Servo 15		1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000

4.2 Kendala dan Pembahasan pada Perancangan dan Pengujian Robot

Dalam perancangan didapat kendala-kendala yang terlihat dalam hasil pengujian, kemudian dibahas solusi yang memungkinkan dan paling efisien. Kendala-kendala yang dibahas merupakan kendala yang berhubungan dengan sistem gerak robot, umumnya adalah masalah keseimbangan robot.

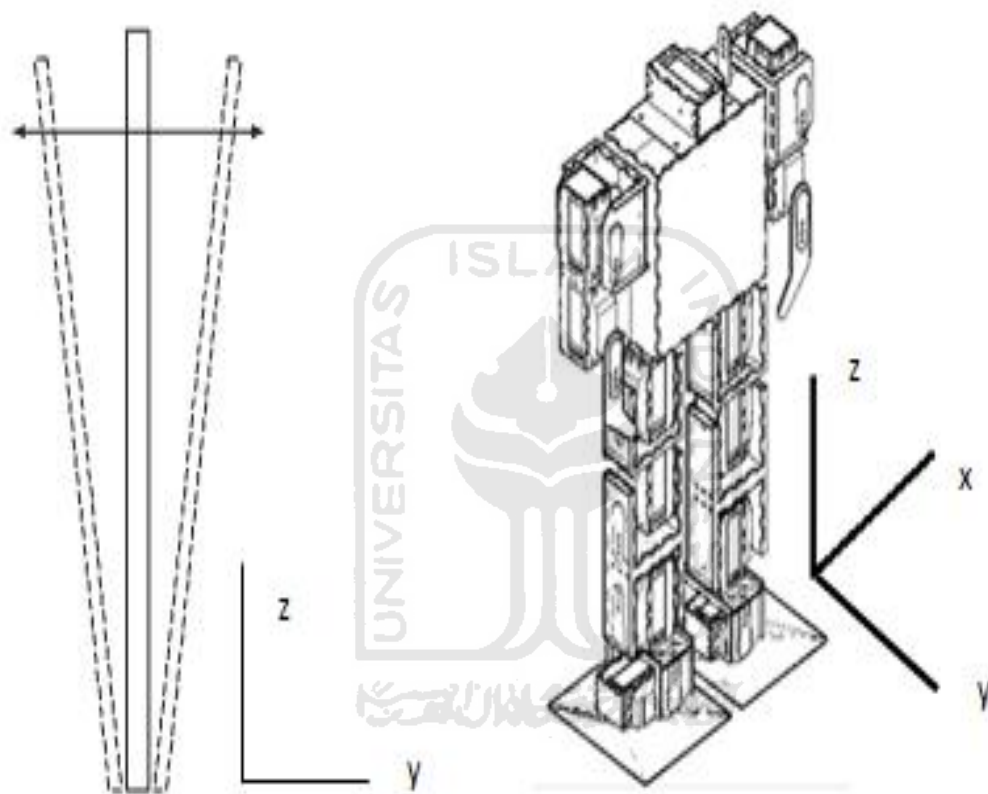
Dalam pencarian solusi lebih banyak digunakan metode pengujian-pengujian (*trial and error*) hingga ditemukan kombinasi yang sesuai dengan yang diharapkan. Gambar 4.7 menunjukkan bentuk robot dua kaki saat pengujian pertama.



Gambar 4.7 Robot pada Pengujian Pertama

Secara umum, robot dapat melakukan beberapa gerakan yaitu gerak maju, gerak belok, dan mundur. Namun kemudian ditemukan kendala yang tidak memungkinkan robot untuk berjalan dengan baik, yaitu robot bersifat sedikit lentur, dalam arti ketika fase akhir gerakan, robot terdorong jatuh akibat gaya sisa dari gerakan robot sebelumnya.

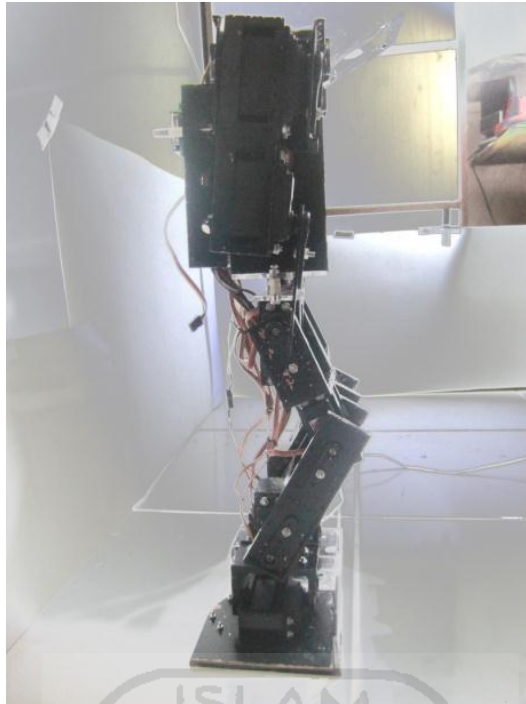
Dalam gerakan yang statis robot terlihat dapat melakukan gerakan-gerakan (*step-step*) dengan baik. Namun pada saat menggabungkan langkah-langkah tersebut (gerak dinamis), robot mengalami kesulitan untuk berhenti, dan selalu terlihat goyang saat bergerak.



Gambar 4.8 Pemisalan Robot Seperti Batang yang Lentur

Kemudian ditemukan solusi yaitu kaki robot akan dimodifikasi sedikit menekuk, yaitu bagian lutut (servo nomor 3 dan 8). Hal ini untuk mengurangi momen yang terjadi saat robot melakukan gerakan. Sehingga tidak lagi terdorong maju dan jatuh.

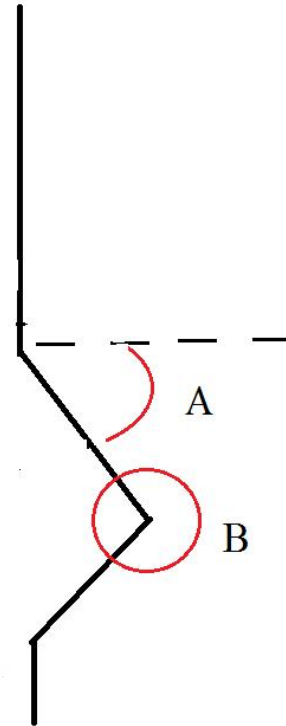
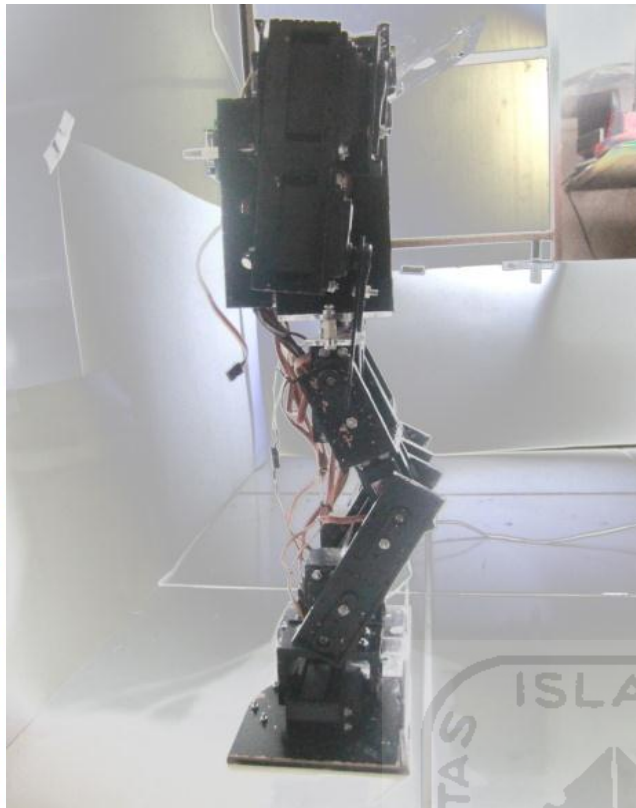
Bentuk ini kemudian dapat memberi solusi pada kendala sebelumnya. Dan robot tetap dapat melakukan gerakan seperti maju dan belok kanan belok kiri sesuai perancangan pada bab sebelumnya. Gambar 4.9 menunjukkan bentuk robot setelah modifikasi.



Gambar 4.9 Bentuk Robot Setelah Modifikasi

Namun dalam perancangan kedua ini juga didapat kendala, yaitu berkurangnya ruang gerak servo yang berhubungan dengan servo yang ditebuk (gambar 4.10-A) dan servo yang ditebuk menjadi kaku (gambar 4.10-B). Hal ini menyebabkan robot tidak dapat melakukan langkah yang panjang (langkah robot menyempit). Selain itu robot juga kesulitan melangkah ke belakang (berjalan mundur), dikarenakan servo yang ditebuk ke depan. Namun satu hal ini dapat diatasi dengan robot berputar 180° untuk berjalan mundur.

Kemudian dalam kurun waktu tertentu, didapat masalah baru yaitu servo menjadi lemah. Dalam arti, servo yang ditebuk lama-kelamaan tidak dapat menahan beban robot ketika melakukan gerakan. Robot cenderung jatuh saat ditopang oleh satu kaki, yang tidak terjadi sebelumnya. Selain itu servo-servo khususnya pada kaki robot menjadi goyah (kocak), mengakibatkan gerakan robot tidak sesuai lagi dengan perancangan gerakan robot sebelumnya.



Gambar 4.10 Kendala pada Servo

Solusi pada permasalahan tersebut adalah merancang ulang cara bergerak robot, dengan mempertimbangkan keadaan tiap motor servo khususnya pada kaki robot. Sedangkan untuk servo-servo yang goyah dapat diperbaiki dengan mengatur ulang baut pengunci sendi robot. Tabel 4.6 menunjukkan nilai servo pada robot saat melakukan gerakan maju setelah diperbaiki.

Tabel 4.6 Hasil Perancangan Ulang Gerak Robot

No	Nomor Servo	Bagian pada Robot	Nilai Motor Servo									
			Tegak	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Step 7	Step 8	Tegak
1	Servo 1	Kaki Kanan	1500	1500	1650	1650	1650	1500	1350	1350	1350	1500
2	Servo 2		2110	2110	2110	2110	2210	2110	2060	2060	2080	2110
3	Servo 3		1990	1990	1990	1990	1990	1990	1990	1990	1990	1990
4	Servo 4		1980	1980	1980	1980	1980	1980	1980	1980	1980	1980

5	Servo 5		2130	2060	2060	2130	2280	2280	2280	2130	2060	2130
6	Servo 6	Kaki Kiri	2540	2540	2690	2690	2690	2540	2390	2390	2390	2540
7	Servo 7		2030	2030	2100	2100	2100	2080	2080	2080	2080	2030
8	Servo 8		1830	1830	1830	1830	1830	1830	1830	1830	1830	1830
9	Servo 9		1970	1970	1970	1970	1970	1970	1970	1970	1970	1970
10	Servo 10		1870	1720	1720	1870	2020	1940	1940	1870	1720	1870
11	Servo 11	Tangan Kanan	2000	2000	1750	1500	1750	2000	2250	2500	2250	2000
12	Servo 12		2900	2900	2900	2900	2900	2900	2900	2900	2900	2900
13	Servo 13		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
14	Servo 14	Tangan Kiri	2000	2000	1750	1500	1750	2000	2250	2500	2250	2000
15	Servo 15		1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
16	Servo 16		2000	2000	2000	2000	2000	2000	2000	2000	2000	2000
17	Servo 17	Kepala	2000	2000	2000	2000	2000	2000	2000	2000	2000	2000

Tabel 4.7 Hasil Perancangan Ulang Gerak Robot (Nilai Sudut Servo)

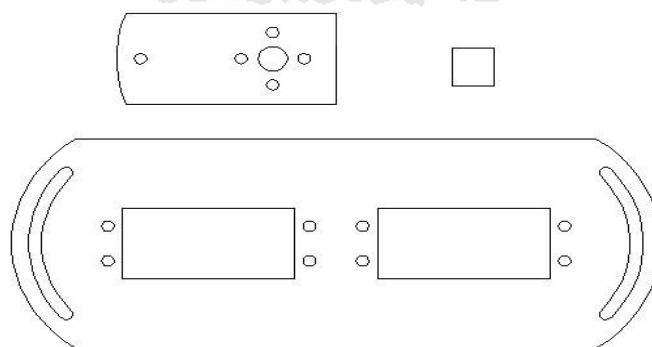
No	Nomor Servo	Bagian pada Robot	Nilai Motor Servo									
			Tegak	Step 1	Step 2	Step 3	Step 4	Step 5	Step 6	Step 7	Step 8	Tegak
1	Servo 1	Kaki Kanan	45	45	59	59	59	45	32	32	32	45
2	Servo 2		100	100	100	100	109	100	95	95	97	100
3	Servo 3		89	89	89	89	89	89	89	89	89	89
4	Servo 4		88	88	88	88	88	88	88	88	88	88
5	Servo 5		102	95	95	102	115	115	115	102	95	102
6	Servo 6	Kaki Kiri	139	139	152	152	152	139	125	125	125	139
7	Servo 7		93	93	99	99	99	97	97	97	97	93
8	Servo 8		75	75	75	75	75	75	75	75	75	75
9	Servo 9		87	87	87	87	87	87	87	87	87	87
10	Servo 10		78	65	65	78	92	85	85	78	65	78

11	Servo 11	Tangan Kanan	90	90	68	45	68	90	113	135	113	90
12	Servo 12		171	171	171	171	171	171	171	171	171	171
13	Servo 13		90	90	90	90	90	90	90	90	90	90
14	Servo 14	Tangan Kiri	90	90	68	45	68	90	113	135	113	90
15	Servo 15		0	0	0	0	0	0	0	0	0	0
16	Servo 16		90	90	90	90	90	90	90	90	90	90
17	Servo 17	Kepala	90	90	90	90	90	90	90	90	90	90

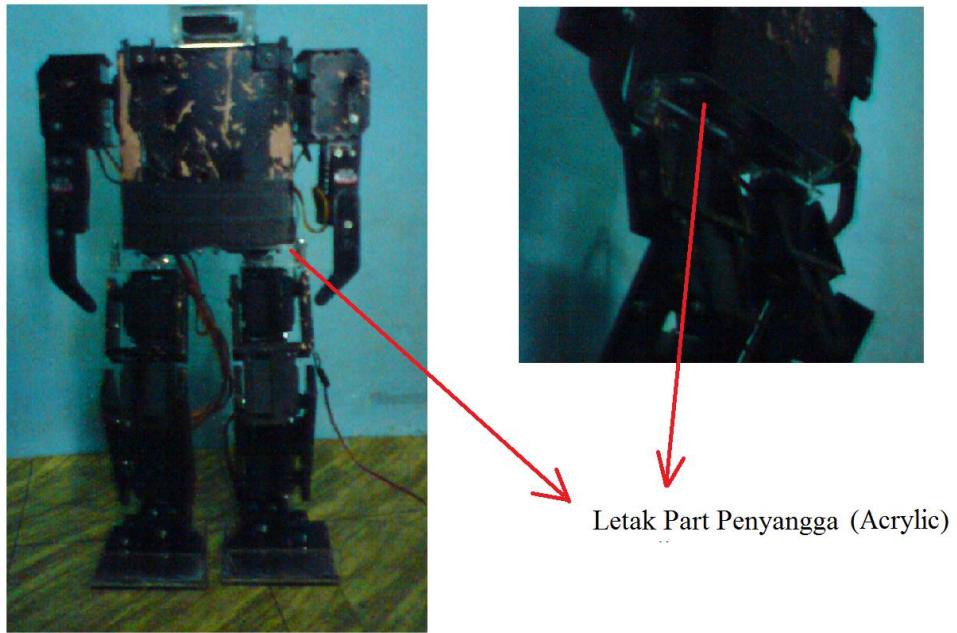
Dari tabel 4.6 dapat dilihat bahwa robot berjalan dengan servo pada sendi putar (servo nomor 1 dan 6). Hal ini memberi solusi pada permasalahan servo robot sebelumnya.

Dalam pengujian gerak robot, disamping kendala-kendala yang telah dibahas, didapat kendala paling utama dan sering muncul, kaitannya dengan sistem mekanik, namun berpengaruh tentunya pada gerak robot, yaitu posisi kaki robot yang menempel saat gerakan miring. Hal ini menyebabkan gerakan robot tidak sesuai dengan perancangan.

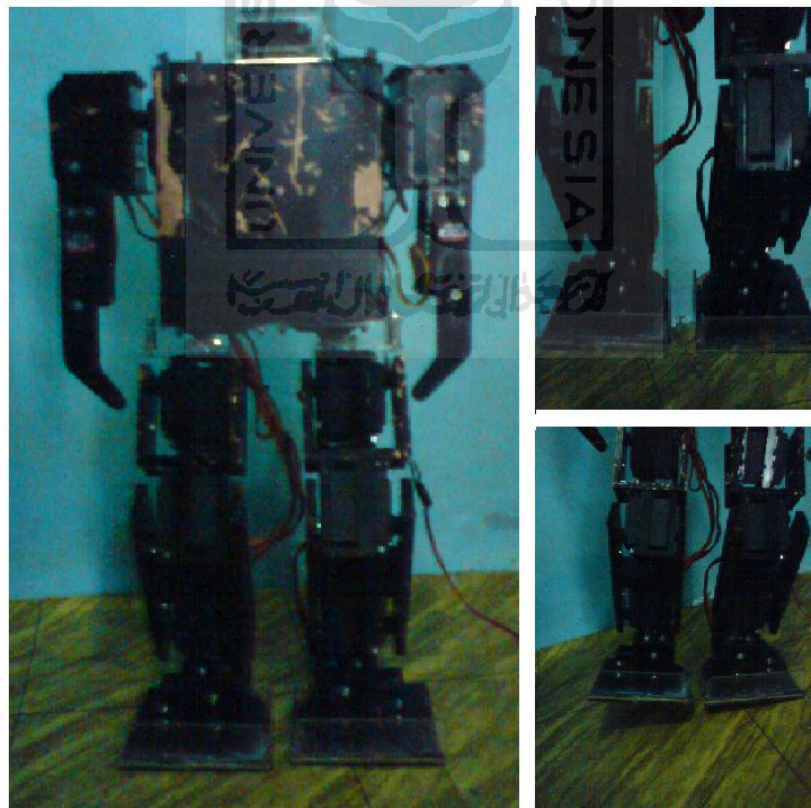
Solusinya dalam hal ini berupa penambahan part penyangga pada servo di bagian paha robot. Sehingga robot dapat melakukan gerakan sesuai yang diharapkan dalam perancangan. Gambar 4.11 menunjukkan bentuk batang hubung, letaknya pada robot, dan pengaruhnya pada gerakan robot.



Gambar 4.11 Bentuk Part Penyangga



Gambar 4.12 Letak Part Penyangga dalam Robot



Gambar 4.13 Kaki Robot Tidak Goyang dan Lebih Kaku

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil perancangan robot dua kaki yang dibuat, maka dapat diambil kesimpulan, yaitu;

1. Perancangan sistem kendali menggunakan metode pengujian-pengujian yaitu memposisikan robot pada gerakan yang diinginkan kemudian menyesuaikan data yang diperoleh dengan penggabungan gerakan robot lebih efektif dan efisien.
2. Penggunaan dua mikrokontroler dengan komunikasi 3 pin yakni untuk pengendali aktuator dan pengendali sensor lebih efektif dibanding penggunaan satu mikrokontroler untuk mengendalikan sensor dan aktuator.
3. Robot dapat melakukan gerakan maju dengan kecepatan terukur 25 detik/ keramik atau 1,2cm/detik.
4. Robot dapat melakukan gerakan belok atau memutar badan 90° dengan waktu tempuh terukur 20 detik/ 90° atau 4,5° /detik.
5. Robot dapat melakukan gerakan menendang bola dengan jauh bola tertendang pada pengujian terukur 20-35cm.

5.2 Penelitian Lanjutan

Dalam penelitian ini masih terdapat kekurangan, sehingga perlu diadakan penelitian-penelitian lanjutan, yaitu:

1. Pada gerak maju/ berjalan robot masih lambat, perlu diperhatikan sistem mekanik yang sesuai dan lebih ringan.
2. Pemasangan servo pada setiap sendi tangan perlu agar gerakannya tidak kaku dan dapat membantu keseimbangan robot.
3. Perlu adanya sensor keseimbangan pada robot agar robot dapat melakukan gerakan dengan lebih baik dan lebih wajar.
4. Jenis motor servo yang lebih baik dan pemrograman lanjut diperlukan agar gerakan robot lebih halus.

DAFTAR PUSTAKA

- Anonim. 2011. *Pengertian Sistem Kendali*. <http://www.wikipedia.com> (Diakses 3/11/2011)
- Asfour, T. (2006). *Humanoid Robots: Design Issues and Control*. Institute of Computer Science and Engineering (ITEC).
- Endik, S. (2011). Perancangan Sistem Mekanik Robot Berkaki Dua dengan Sepuluh Derajat Kebebasan. Skripsi. Tidak Diterbitkan. Jurusan Teknik Mesin Universitas Islam Indonesia Yogyakarta.
- Gandjar K, Filipus RK, Yohannes TT, Galih S., Akthur F. (2006). *Perancangan Bentuk Geometri dan Derajat Kebebasan dan Analisa Kestabilan Robot Humanoid "Makara 1"*. Departemen Teknik Mesin – Fakultas Teknik Universitas Indonesia.
- Halim, F.A., & Tjen, A. (2007). Perancangan Robot Humanoid Dengan Kemampuan Pergerakan Autonomous Mengikuti Objek. *Seminar Nasional Sistem dan Informatika 2007*. Bali, Indonesia.
- Ismaya, M.B., Mutijarsa, K. (2007). *Robot Pengikut Bola Menggunakan Sensor Kamera Berbasis Metoda Opencv Camshift*. Skripsi. Tidak diterbitkan. Jurusan Teknik Elektro Institut Teknologi Nasional Bandung.
- Jou, Y.T. (2003). *Human-Robot Interactive Control*. Disertasi. Faculty of the Frits J. And Dolores H. Russ College of Engineering and Technology of Ohio University.
- Maulana, R.H. *Penerapan Behavior Based Robotic Sistem Navigasi Dan Kontrol Robot Soccer*. Seminar Skripsi. Tidak Diterbitkan. Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember.

- Pardos, J.M., & Balaguer, C. (2001). *Humanoid Robot Kinematics Modeling Using Lie Groups*. Robotics Lab, Department of Systems Engineering and Automation, Uni-versity Carlos III of Madrid Spain.
- Pennestr`i, E., Cavacece, M., & Vita, L. (2005). On The Computation Of Degrees-Of-Freedom:A Didactic Perspective. *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Long Beach, California, USA.
- Pitowarno, E. (2006). *ROBOTIKA : Desain, Kontrol dan Kecerdasan*.
- Putra, T.S. (2008). *Perancangan Robot Dua Kaki dengan Tiga Derajat Kebebasan*. Skripsi. Tidak diterbitkan. Jurusan Teknik Mesin Universitas Islam Indonesia Yogyakarta.
- Salfikar, I., Sulistijono, I.A., & Setiaji, O. (2010). *Humanoid Soccer Robot : System Kendali Visual Menggunakan Metode Fuzzy Sebagai Penjejak Objek*. Skripsi. Tidak diterbitkan. Jurusan Teknik Elektronika Politeknik Elektronika Negeri Surabaya—*PART A: SYSTEMS AND HUMANS, VOL. 34, NO. 5*.
- Sardain, P. & Bessonnet, G. (2004). *Forces Acting on a Biped Robot. Center of Pressure — Zero Moment Point*. IEEE Transactions On Systems, Man, And Cybernetics.
- Vukobratovic, M. (2004). Zero-Moment Point — Thirty Five Years Of Its Life. *International Journal of Humanoid Robotics Vol. 1, No. 1. Page 157–173*