TWO-LEGGED ROBOT
SIMULATION AND DEVELOPMENT

By

Nirav A. Patel
(05MCE011)

DEPARTMENT OF CCOMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY
NIRMA UNIVERSITY OF SCIENCE AND TECHNOLOGY

Ahmedabad 382481
May 2007

TWO-LEGGED ROBOT SIMULATION AND DEVELOPMENT

Major Project

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science and Engineering

By

Nirav A. Patel
(05MCE011)

Guide
Dr. S N. Pradhan
Prof. K D. Shah

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY
NIRMA UNIVERSITY OF SCIENCE AND TECHNOLOGY

Ahmedabad 382481
May 2007

This is to certify that Dissertation entitled

TWO LEGGED ROBOT

SIMULATION AND DEVELOPMENT

Presented by

Niravkumar A. Patel

has been accepted toward fulfillment of the requirement
for the degree of
Master of Technology in Computer Science & Engineering

Prof. S. N. Pradhan
Professor In Charge

Prof. D. J. Patel
Head of The Department

Prof. A. B. Patel
Director, Institute of Technology

CERTIFICATE

This is to certify that the Major Project entitled "Two Legged Robot Simulation and Development" submitted by Mr. Niravkumar A. Patel (05MCE011), towards the partial fulfillment of the requirements for the degree of Master of Technology in Compute Science & Engineering of Nirma University of Science and Technology, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Project Guide,
Dr. (Prof.) S. N. Pradhan.
P. G. Coordinator,
Department of Computer Engineering,
Institute of Technology,
Nirma University,
Ahmedabad

Date:-

Project Guide,
Prof. K. D. Shah.
Department of Instrumentation and Control Engineering,
Institute of Technology,
Nirma University,
Ahmedabad

Date:-

ACKNOWLEDGEMENT

It gives me great pleasure in expressing thanks and profound gratitude to **Dr. S. N. Pradhan**, M-Tech In-charge, Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad for his valuable guidance and continual encouragement throughout the Major project. I heartily thankful to him for his time to time suggestion and the clarity of the concepts of the topic that helped me a lot during this study.

I like to give my special thanks to **Prof. K.D. Shah**, Professor, Department of Instrumentation and Control Engineering, Institute of Technology, Nirma University, Ahmedabad, without whose suggestions and help this project would not have seen light of the day. I am very thankful to him for his continual kind words of encouragement and motivation throughout the Major Project.

I like to give special thanks to **Prof. B.B. Kadam** for his keen interest in my project and help for approval of my project proposal.

I would like thank lab in-charges of laboratories of Electrical Engineering Department, Instrumentation and Control Engineering Department, Mechanical engineering department and Computer Engineering Department for helping me in all possible ways.

I am also thankful to **Prof. A. B. Patel**, Director, Institute of Technology for his kind support in all respect during my study.

The blessings of God and my family members makes the way for completion of major project. I am very much grateful to them.

The friends, who always bear and motivate me throughout this course, I am thankful to them.

Niravkumar A. Patel
[05mce011]

ABSTRACT

Robot are becoming more and more ubiquitous in our life, but still Humanoids (robots like humans) are in their development phase and much of the research is being carried out by many companies and instituted for enhancing various capabilities of humanoids.

Walking is the most difficult task to be carried out by a humanoid because it has to balance it self during walking. So developing a two legged robot is a beginning which would ultimately lead to development of a humanoid. In this project a two-legged robot has been designed. The robot has 10-Degrees of Freedom (DOF) in total, for each Leg 2 DOF in Ankle Joint, 1 DOF in Knee joint and 2 DOF in Hip Joint. The height of the robot is approximately 600 cm and weight is approximately 6.85 Kg. Robot has open loop control system since stepper motors are used as actuators. Simulator has also been developed which simulates all the joint movements of the robot under development. Simulator also implements forward kinematics and the analysis of torque requirements of all the joints. The torque required by each joint for carrying out various tasks can also be computed. It also shows the projection of the Centre of Gravity along with top view of the robot which can be used to view the stability of the robot. Simulator also includes an **A**pplication **S**pecific **C**ompiler (ASC) which allows the user to write program in high level language which is compiled to low level (Assembly) code for the target microcontroller (89S52) and high level code is also simulated by the simulator.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATION NOTATION AND NOMENCLATURE

ASC	-	Application Specific Compiler
CAD	-	Computer Added Design
CG	-	Center of Gravity
DOF	-	Degrees of Freedom
GUI	-	Graphical User Interface
PCB	-	Printed Circuit Board
RNN	-	Recurrent Neural Network

1. INTRODUCTION

Robotics is one of the most growing fields these days. When some one hears a word "*robot*" thinks of a machine which looks like human being but actually robot is any machine which repeatedly follows instruction given to it. Broadly speaking robot can be wheeled or legged, autonomous or human controlled battery operated or externally energized.

Most of the present robots are wheeled or assembly line robots which carry out repetitive tasks given to them. For last few decades robots are used in the industry for making the manufacturing process automatic, which also reduces the time to manufacturer.

Even after robotics being quite old field, legged-robots are in their primitive stage and are able to perform very primitive tasks like walking or transferring some predefined thing from one place to another place. Capabilities of robots are depended on Degrees of Freedom (DOF) and number of sensory inputs and increasing DOF increases the cost of development and manufacturing. Most challenging task to be carried out legged robot is to balance it self while walking and this is limited by capability and response of actuators. Before developing the physical model it is economical to carry out simulation to verify the design.

Balancing is the most difficult task for any two-legged robot so for planning walking patterns many algorithmic approaches are proposed. Some approaches are derived by studying the human walk patterns and some are neural network based which generates control signals by learning from previous faults.

This thesis discusses various design challenges involved in designing a two-legged robot along with simulation results. It focuses on some very important aspects of designing a two-legged robot. It also focuses on parameters affecting the design of the robot and also gives solution to overcome those problems.

1.1 HISTORY OF TWO-LEGGED ROBOTS FROM HONDA

In 1986, Honda engineers set out to create a walking robot. Early models (E1, E2, and E3) focused on developing legs that could simulate the walk of a human. The next series of models (E4, E5, E6) were focused on walk stabilization and stair climbing. Next, a head, body and arms were added to the robot to improve balance and add functionality. Honda's first humanoid robot, P1 was rather rugged at 6′ 2″ tall, and 386 lbs. P2 improved with a more friendly design, improved walking, stair climbing/descending, and wireless automatic movements. The P3 model was even more compact, standing 5′ 2″ tall and weighing 287 lbs.[8]

Figure 1.1 Robot E0 from Honda

E0 (1986)

- First robot developed at Honda to examine the principles of Two-Legged Locomotion. In Figure 1.1[9] E0 is shown.

E1, E2, E3 (1987-1991)

- **Goal:** Realizing Rapid Two-Legged Walking by analyzing human walking
- **E1:**Walked at static pace of 0.25 km/h
- **E2:**First dynamic movement at 1.2 km/h mimicking the human walk
- **E3:**Walked at normal human speed of 3 km/h

E4, E5, E6 (1991-1993)

- **Goal:** Completing the basic functions of Two-Legged walking and Establishing technology for stable walking
- **E4:**Knee length increased to 40 cm to achieve speed of 4.7 km/h
- **E5:**First autonomous locomotion
- **E6:**Autonomous control of balancing when going up and down the stairs or slopes or stepping over an obstacle

P1, P2, P3 (1993-1997)

- **Goal:** Research on completely Independent Humanoid Robots
- **P1:**First prototype of a man-like model with upper limbs and the body
- **P2:**Worlds first self regulating humanoid walking robot
- **P3:**The first completely independent humanoid walking robot

1.2 ASIMO

ASIMO stands for Advanced Step in Innovative Mobility. Its world's most advanced humanoid robot. ASIMO is the culmination of nearly two decades of humanoid robotics research by Honda scientists and engineers. ASIMO can walk on uneven slopes and surfaces, turn smoothly, climb stairs, reach for and grasp objects, switch lights on and off, and open and close doors. Now, ASIMO can also comprehend and respond to simple voice commands. ASIMO has the ability to recognize the face of a select group of individuals. Using its camera eyes, ASIMO can map its environment and register stationary objects. ASIMO can also yield to pedestrians in its path until its path is cleared. In Figure 1.2[9] ASIMO is shown.

Figure 1.2 ASIMO

1.2.1 Features of ASIMO

ASIMO is of normal human height and weight and is capable of carrying out many tasks in the same way as human. It can walk, run and climb the stairs.

ASIMO has below mentioned features.

- Its Advanced Step in Innovative Mobility
- Compact and Light weight
- Sophisticated walk technology
- Greater freedom of upper limb movements
- Greater ease of manipulation
- More human friendly design

1.2.2 Specifications Of ASIMO

ASIMO is designed to work in human friendly environment. So its dimensions are same as normal human. It can work both in office and in the home. Its height and various objects' height in office and home are shown in Figure 1.3[8].

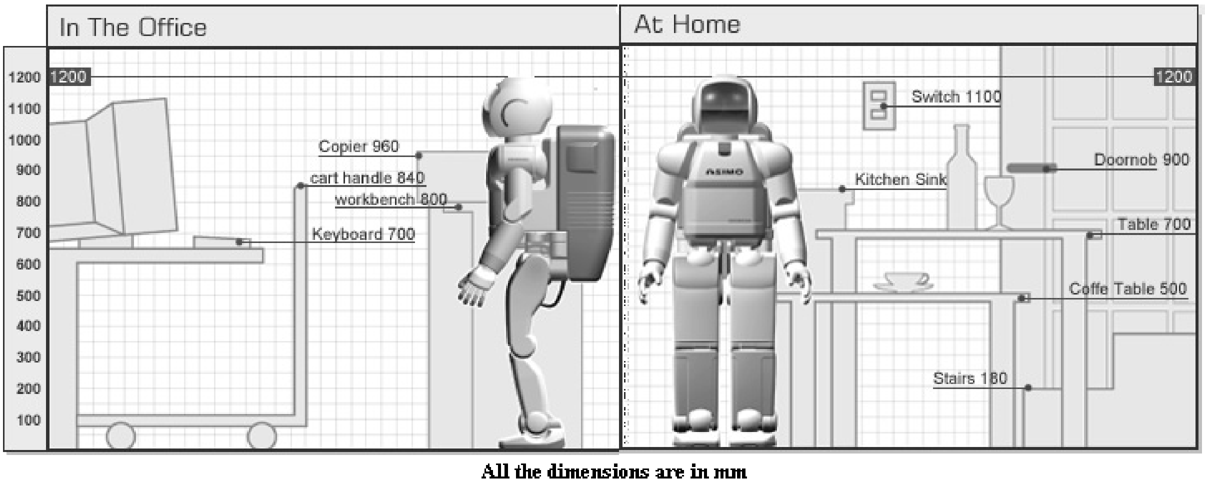

Figure 1.3 ASIMO at home and office

Table 1.1 Specifications of ASIMO

Parameters	"original ASIMO" (2000)	"next generation ASIMO" (2004)	"new ASIMO, version 2.0" (2005)
Weight	52 kg	54 kg	54 kg
Height	120 cm	130 cm	130 cm
Width	45 cm	45 cm	45 cm
Depth	44 cm	44 cm	37 cm
Walking speed	1.6 km/h	2.5 km/h	2.7 km/h 1.6 km/h (carrying 1 kg)
Running speed	-	3 km/h	6 km/h (straight) 5 km/h (circling)
Airborne time	-	0.05 seconds	0.08 seconds
Battery	Nickel metal hydride / 38.4 V / 10 Ah / 7.7 kg / 4 hours to fully charge		
Continuous operating time	30 minutes	1 hour	40 minutes (walking)
Degrees of Freedom	26	34	34

1.3 CHALLENGES OF TWO-LEGGED ROBOT DEVELOPMENT

Development of legged robots is complex task since it requires knowledge of multiple disciplines like electronics engineering, mechanical engineering and computer engineering. Developing a two-legged robot is more difficult then multi leg robots because balancing the two-legged robot when either leg is in the air is very complex problem of control system engineering. Performance of the robot is restricted by the response and capacity of actuating device.

Since two-legged robot has to lift the weight of whole body while walking joints requires very high torque. Small motors can not provide very high torque and hence are combined with gearbox to increase the torque. To reduce the torque requirement of any joint the height of the robot should be decreased which makes the robot unusable in daily uses in home. So selection of actuating device is the most difficult task for developing the two-legged robot. Actuators may be hydraulic, pneumatic or electrical. Now a day stepper motors for open loop control and servo motors for closed loop control are used.

Since two-legged robot have high Degrees of Freedom(DOF) it's control becomes difficult, so microcontrollers and microprocessors are used to control the actuating devices like motors. As DOF increases the cost of robot increases so cost minimization is also a challenging job to be carried out to bring robot from lab to factory for manufacturing.

The performance of the human sized humanoid robots is strictly limited by the performance of the motors, because the progress in the motors have not been remarkable compared with the progress in the electronics. The ratio of motor power and motor weight is saturating, or at least the great progress cannot be expected. In order to overcome the limitation of performance, the gravity must be compensated [1].

1.4 ARCHITECTURE OF TWO-LEGGED ROBOTS

Two-legged robot can be divided in to three sub systems, mechanical subsystem, electronics subsystem and software subsystem. Mechanical subsystem consists of physical model of the robot and calculation of torque required by each joint. It also takes care of kinematics of robot to carry out different tasks. Electronics subsystem consists of microcontroller and motor controller. Microcontroller is responsible for generating control signals for rotating each motor as per the command given to it. Motor controller is responsible for providing required voltage and current to motor on reception of signal from microcontroller.

In the Figure 1.4 basic structure and working principle of any autonomous robot with open loop control is shown.

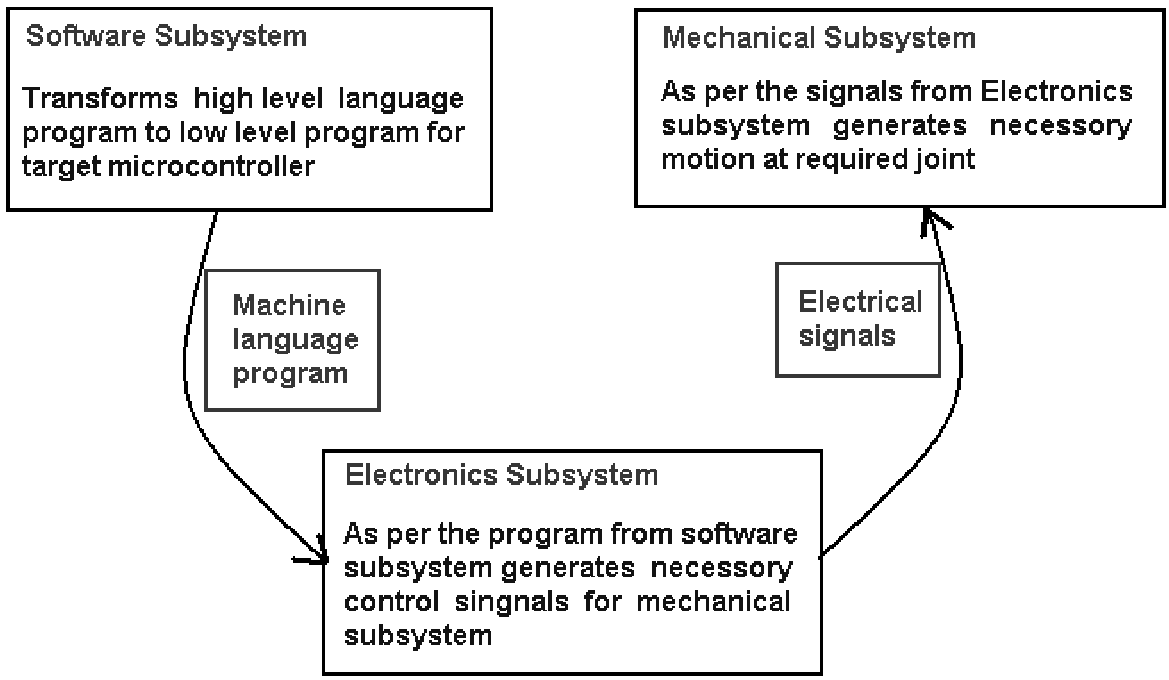

Figure 1.4 Architecture of autonomous robot

As shown in the Figure 1.4 high language program is transformed to low level machine code for target microcontroller or microprocessor. After that software subsystem (computer) is detached and microcontroller takes command. As per the program microcontroller generates necessary signals for required actuation and these signals are converts in to motion at appropriate joint by mechanical subsystem. This process is repeated until required functionality is not performed by robot satisfactorily.

1.5 TWO-LEGGED ROBOT SIMULATOR

Simulators are very useful since they provide easy way to carry out feasibility study and also are helpful for reducing price of development. To predict the physical humanoid motion, it is necessary to accurately formulate and solve the kinematics and dynamic equations of the mechanisms. The humanoid is a system with multiple bodies and many degrees of freedom. The analytical complexity of non-linear algebraic equations of kinematics and nonlinear differential equations of dynamics makes it practically impossible to obtain closed-form solutions. On the other hand, commercial dynamic software packages such as ADAMS, 3D Working Model, DADS can also be used. The usefulness of such software packages has been confirmed in many areas. These software packages can automatically formulate the equations of kinematics and dynamics, solve the nonlinear equations, and provide computer graphics output of the simulation results [3].

Most of the Simulators have below mentioned features

- Simulates movements of different parts of body
- Can be used to analyze movements and their effects on centre of gravity
- Shows 3D model on computer screen
- Provides Graphical User Interface (GUI) with buttons for applying movements to different parts.
- Provides facility to supply multiple instructions to be executed in parallel or sequentially.

2. LITERATURE SURVEY

In aging societies, the need of robots to assist human activities in daily environments, such as offices, homes and hospitals, is growing rapidly. Such robots must communicate with humans and accomplish tasks in the human's daily environments. In the personal or service robotics domain a very close interaction between humans and robots is crucial [2].Our daily environments are constructed so as to accommodate human bodies. Therefore, a humanoid with the anthropoid shape (two arms and two legs) is a uniquely appropriate form [3].

Humanoid robots are expected to have variety of motions that enables good interaction with real human environment [4]. A simple bipedal locomotion is considered to study the learning processes in a self-organized fashion. There are two different concepts, self-learning and self-training. The former is a step by step learning to a better performance starting from a bad moving style. The latter is concerned with how to keep the performance after arriving at the best one [5].

Since a biped humanoid inherently suffers from in-stability and always risks to tipping over, stable and reliable biped walking is the most important goal. [3].For walking stable motions are required to be generated. Making a program for generating several stable motions using the standard programming language such as C is not only time consuming but also hard to understand and tune. For this, a suitable recurrent neural network (RNN) language inspired from neurobiology can be used. A simple method of motion generation based on polynomials generated by RNN can be used. All motions can be generated using a basic RNN circuit of a first order polynomial. [4]

The design of a simple biped robot primarily involves the control of balance. Controlling the direction of balance for a two legged walking robot typically means mimicking the human form and its walking locomotion. Even though the human locomotion approach is taken as the ultimate reference, gaits can be developed using less sophisticated methods. The ultimate aim is to maintain an upright torso while advancing one leg in front of the other continuously. The mechanical structure then forces us to consider a combination of factors such as weight of each leg mechanism, and its appropriate dynamics such as velocity, acceleration, and degrees of freedom. Without developing other established principles, an adequate biped robot should have at least two degrees of freedom. Two degrees of freedom will allow the robot to swing or swivel one leg in front of the other for a step. With increasing degrees of freedom the robotic gait becomes smoother but extremely complex with respect to control. For a simple biped robot, the hardware needed is mechanical links, servos, servo controller, and a power source. The required components can be arranged in different configurations, but for the true autonomous biped robot a self contain package is ideal [6].

The simulator is a significant tool to generate stable and reliable walking pattern. Using the simulator, we can predict the humanoid's physical capability subject t o the constraints of actuators, and clarify the required specifications of actuators to execute a desired task [3].

2.1 EXISTING TOW-LEGGED ROBOTS

There exists very few numbers of two-legged robots which can walk and run smoothly. Some two-legged robots studied are discussed in sub sections.

2.1.1 ASIMO

ASIMO was created by Honda engineers with 34 Degrees of Freedom that help it walk and perform tasks much like a human. Specifications of various versions of ASIMO are shown in Table 1.1. One Degree of Freedom is the ability to move right and left or up and down. These degrees of freedom act much like human joints for optimum movement and flexibility. Lightweight materials, like a magnesium alloy structure, combined with powerful computers and 34 servo motors throughout its body help ASIMO move smoothly with ease. In Table 2.1 locations and number of DOF in ASIMO is shown. ASIMO is designed to operate in the real world, where people need to reach for things, pick things up, navigate along floors, sidewalks, and even climb stairs. ASIMO's abilities to run, walk smoothly, climb stairs, communicate, and recognize people's voices and faces enables ASIMO to easily function in our world and truly assist humans.

Table 2.1 Degrees of Freedom of ASIMO

Head	Neck Joint(U/D,RT)*1	2 DOF
Arm	Shoulder Joint(F/B, U/B, RT)	3 DOF
	Elbow Joint(F/B)	1 DOF
	Wrist Joint(RT)	1 DOF
		5 DOF X 2 arms = 10 DOF
Hand	5 Fingers(Grasping)	1 DOF
		1 DOF X 2 hands = 2 DOF
Leg	Hip Joint(F/B, L/R, RT)	3 DOF
	Knee Joint(F/B)	1 DOF
	Ankle Joint(F/B, L/R)	2 DOF
		6 DOF X 2 Legs = 12 DOF

*1 F/B: Forward/Backward U/D: Up/Down L/R: Left/Roght RT: Rotation

Through proactive control of ASIMO's posture while both feet are off the ground, the running speed can achieve nearly 4 mph (6km/hour). ASIMO is able to run in a circular pattern at high speeds because the center of gravity of ASIMO's body is tilted inside of the circle to maintain balance with the amount of centrifugal force experienced.

A 6-axis force sensor in each foot detects stable foot placement on each step. Combined with data from its torso-mounted gyroscope and accelerometer, ASIMO utilizes unique mathematical algorithms that allow it to negotiate stairs, successfully climbing and descending with ease.

ASIMO can comprehend and carry out tasks based on simple voice commands given in English that have been preprogrammed into its onboard memory. The number of commands that can be programmed is basically unlimited. Individual voices can also be registered to increase the performance of the voice recognition function.

ASIMO incorporates the capability to interface with an IC Communication card so ASIMO is no longer limited to relying only on front vision obtained via a visual sensor. ASIMO can exchange information with an IC card, enabling ASIMO to recognize the position of the person with the card in a 360-degree angle. By having a person hold or wear this card, ASIMO can interact better since the card can be programmed to include a name, affiliation and ID number.

2.1.1.1 Earlier Walking Technology

In 1980's Honda developed first two-legged robot which was able to walk. Walking by putting one leg before the other was successfully achieved. However, taking nearly five seconds between steps, it walked very slowly in a straight line. To increase walking speed, or to allow walking on uneven surfaces or slopes, fast walking must be realized. While walking the projection of Center of Gravity (CG) should belong to the area of resting foot. As shown in the Figure 2.1[8] when robot walks at a lower speed it is able to maintain the CG so that its projection remains in side the area of resting foot, but when the speed is increased CG moves outside the support area so robot can not remain stable.

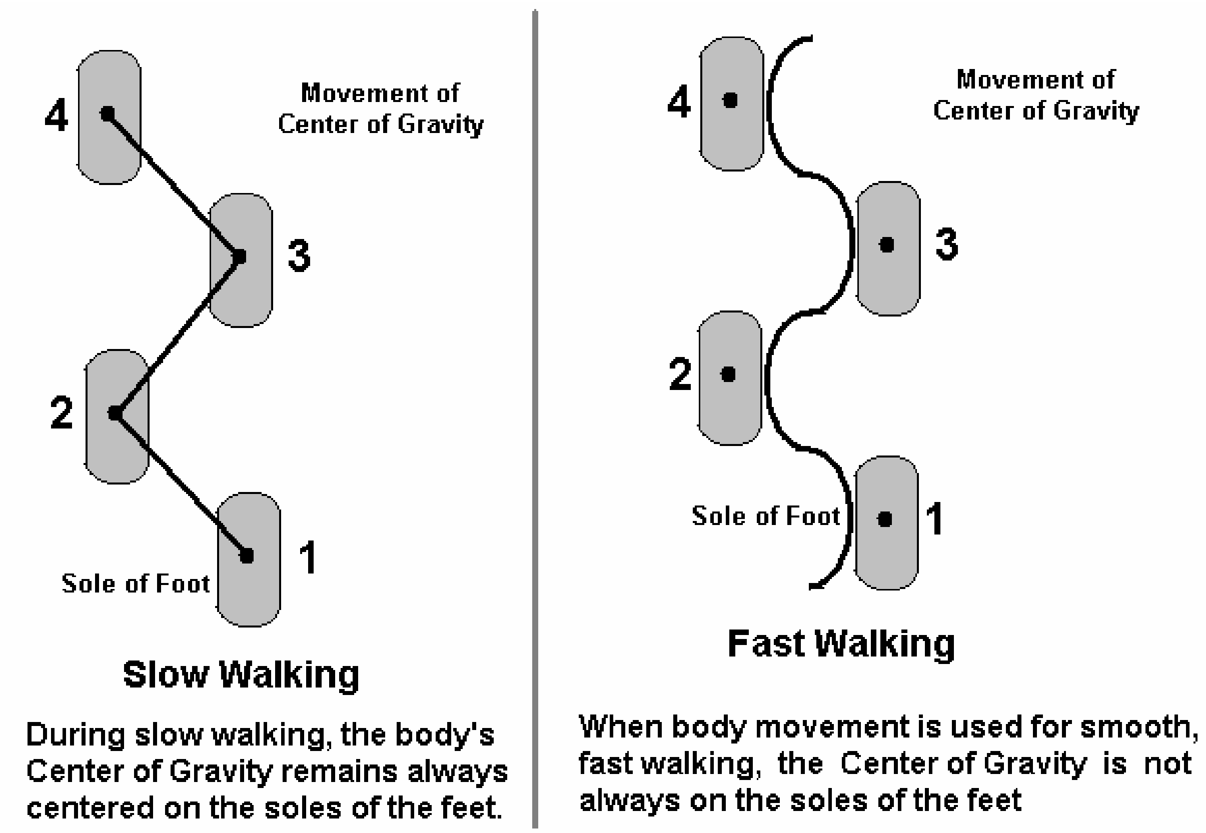

Figure 2.1 Fast and slow walking and its effect on CG

2.1.1.2 Advanced Walking Technology

New ASIMO Walking Technology uses a predictive movement control over the earlier walking control technology. This new two-legged walking technology permits more flexible walking. As a result, ASIMO now walks more smoothly and more naturally. When human beings walk straight ahead and start to turn a corner, they shift their center of gravity toward the inside of the turn. By using New ASIMO Walking Technology, ASIMO can predict its next movement in real time and shift its center of gravity in anticipation.

Because continuous flexible walking is possible, ASIMO can move and walk rapidly and smoothly at all times. Robots up to the P3 turned according to combinations of stored walking patterns. ASIMO creates walking patterns in real time and can change foot placement and turning angle at will. As a result, it can walk smoothly in many directions. In addition, because stride (time per step) can also be changed freely, ASIMO's movements are much more natural. By using New ASIMO Walking Technology, ASIMO can change its walking smoothly and continuously at any time. New ASIMO Walking Technology allows robots to exist more easily in the human living environment. This technological development will allow robots of the future to work in harmony with people while avoiding obstructions on their own.

2.1.2 Saika-4

In order to meet the demand of a research platform for intelligent robotics, a humanoid robot named Saika-4 has been developed at Department of Aerospace Engineering, Tohoku University, Japan. The Saika-4 is a self-sustained biped robot equipped with two arms and a robotics head. The Saika-4 has been designed so as to the light weight, low cost and human size. The weight is about 47 [kg], and the height is about 1300 [mm]. The Saika-4 has 30 DOF in total, which is *6* DOF for each leg, 7 DOF for each arm, 1 DOF for each hand and 2 DOF for robotics head [4]. Locations of DOFs are shown in the Figure 2.2[4].

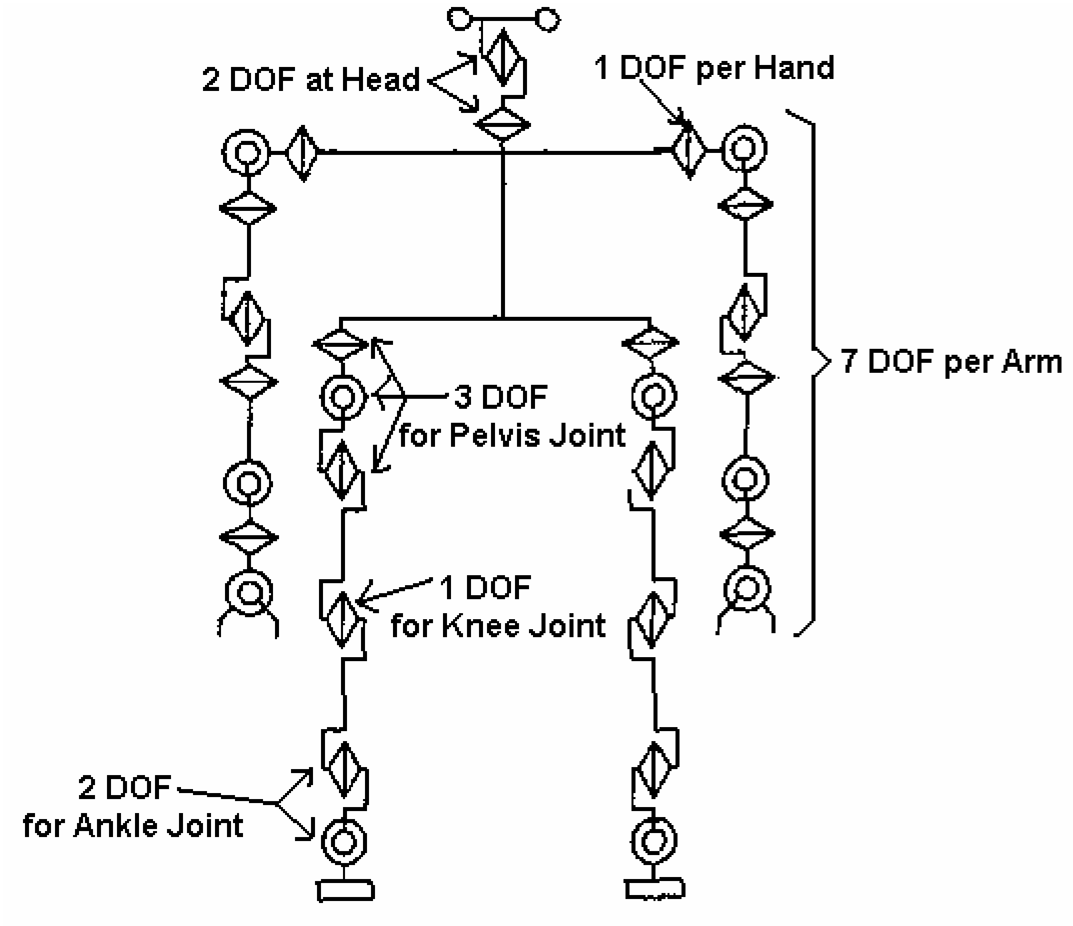

Figure 2.2 Locations of DOFs in Saika-4

The Saika-4 is equipped with a PC, batteries, a wireless Ethernet modem, a gyroscope, and motor driven in the body. Furthermore, the Saika-4 is equipped with a gravity compensation mechanism using springs in the legs.

Each joint consists of a DC servo motor, a harmonic drive reduction gear, and a rotary optical encoder. Specifications of motor drivers are shown in Table 2.2[4].

Table 2.2 Specifications of motor driver.

Maximum Voltage[V]	Maximum Current [A]	Maximum Power [W]	Weight [gram]	Length [mm]	Width [mm]	Depth [mm]
12	7	84	15	45	27	10

2.1.2.1 Gravity Compensation Mechanism

In a humanoid robot, almost actuator power of the leg is used for supporting the upper body. If the weight of the upper body is compensated by some kind of method, mobility and energy efficiency will be improved. At the arms, some gravity compensation method are suggested, legs can be reequipped with a gravity compensation mechanism by springs. But in the case of the leg, equipping with a gravity compensation mechanism is difficult, because size and weight are restricted and base part moves. In the case of common method using spring, as shown in Figure. 2.3(a), compensation ratio depends on a joint angle. And so they (at Department of Aerospace Engineering, Tohoku University, Japan) carry out a using stainless-wire, and contrive spring layout, as shown in Figure. 2.3(b) [4]. Therefore, term depending on a joint angle is canceled, and this mechanism can compensate without depending on a joint angle, follows:

$$M_1 = mgL\sin\left(\frac{\theta}{2}\right) - 2K_1 a^2 \sin\left(\frac{\theta}{2}\right)$$
$$= \mu mgL\sin\left(\frac{\theta}{2}\right) \quad \text{------------ (1)}$$

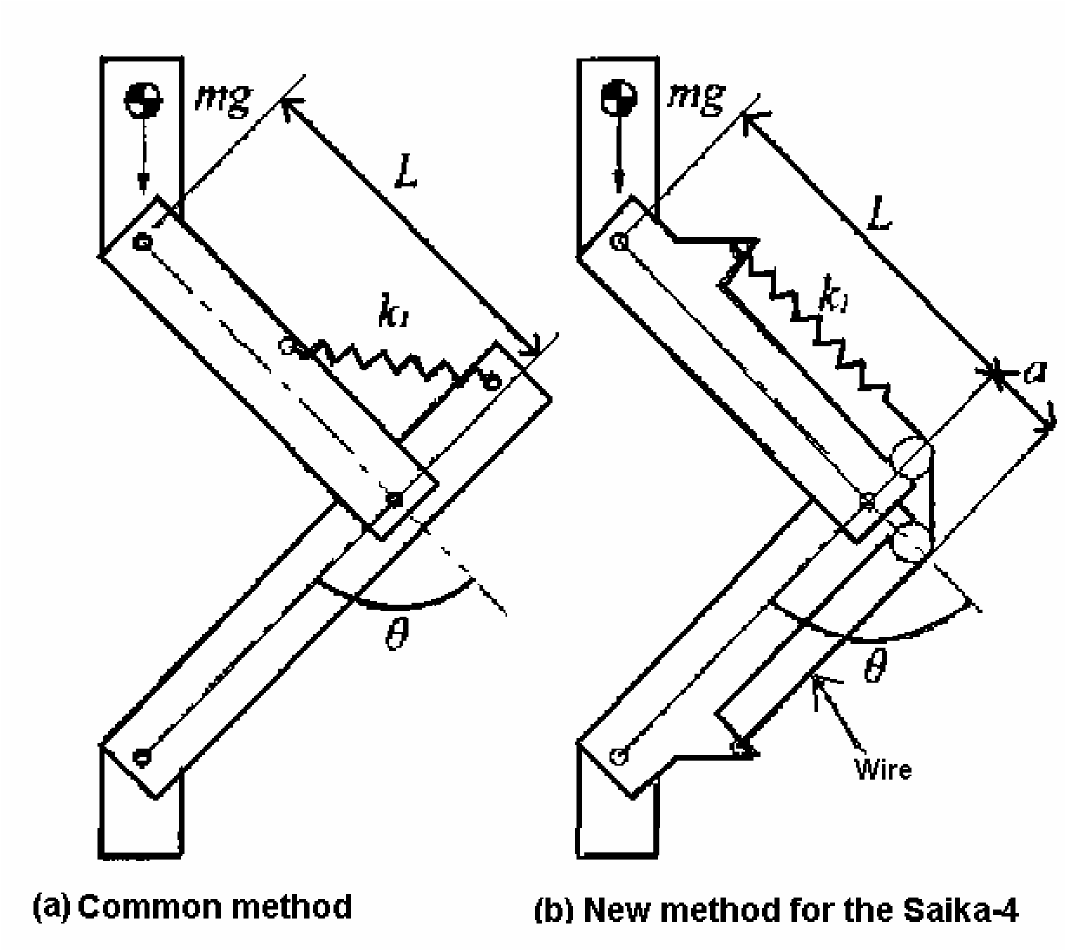

Figure 2.3 Gravity compensation mechanisms at knee

M1 is needed torque in the stationary condition for the extension-flexion joints at the knee. And μ is the compensation ratio. Next, they considered the ankle and the hip. Even in this case, they use wire as shown in Figure. 2.4, and have the following relationship.

$$M_2 = \tau - K_2 r^2 \sin\varphi \text{ - - - - - - - - - - - - - } (2)$$

τ is gravitational torque. M_2 is needed torque for the extension-flexion joints at the ankle and the hip, rating φ equal to θ/2 at knee-bend and the gravity torque equal to 0.5mgLsin (θ/2). Therefore, term depending on a joint angle is canceled and equation (2) turns to following:

$$\begin{aligned} M_2 &= 0.5 mgL \sin\varphi - k_2 r^2 \sin\varphi \\ &= 0.5 \mu mgL \sin\varphi \text{ - - - - - - - - - - - - - } (3) \end{aligned}$$

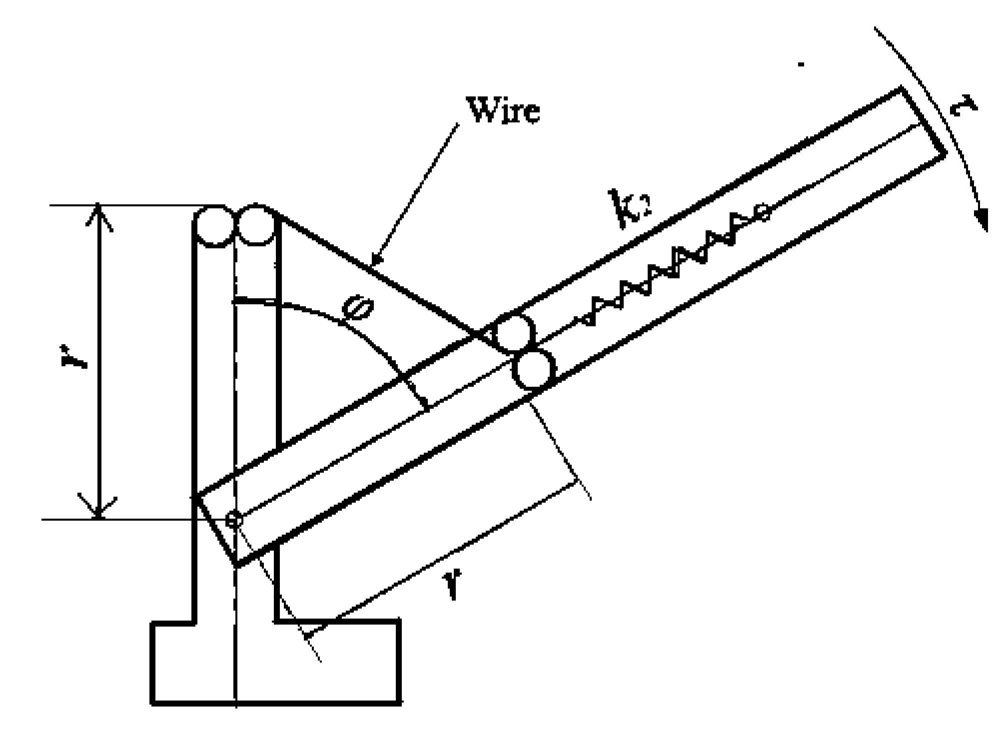

Figure 2.4 Gravity compensation mechanisms for ankle and hip

These mechanisms can be equipped in legs without changing appearances of the legs, therefore it is suited to a humanoid robot that is restricted to size and weight. The spring constants *(k1* and *K2)* are determined so that the compensation ratio *(μ)* is 0.5.And this mechanism aim that load torque of the motor is flattened and the leg return to initial position by spring force at the swinging phase [4].

2.2 APPLICATIONS OF TWO-LEGGED ROBOTS

In the application research, five applications of humanoid robots, i.e. maintenance tasks of industrial plants, security services of home and office, human care, tele-operations of construction machines, and cooperative works in the open air When we consider how to use humanoid, it is important to deeply consider how the human shape can be utilized practically. Generally speaking, there may not be many cases where human shape robot will be required and almost cases will require the robot to have only the functions enough to achieve the desired tasks [7]. When we consider the practical importance of the human shape, some merits the human shape robot produces can be recognized as follows:

(1) The behaviors of the human shape robot produce emotional feelings useful for friendly.

(2) Human shape is one of the best shapes for robot remotely controlled.

(3) There are many cases which need human shape devices to replace human worker with a robot and machines.

3. SUBSYSTEMS OF TWO-LEGGED ROBOT

Any autonomous robot can be divided in to three subsystems, Mechanical subsystem, electronics subsystem and software subsystem. For correct functioning of the robot all three subsystems should work in collaboration with each other.

3.1 MECHANICAL SUBSYSTEM

Mechanical subsystem focuses on motor and gearbox selection, torque requirements of each joint and kinematics of the robot. Stepper motors are used for actuation and one stepper motor for each joint is used. Each stepper motor gets signal from electronics subsystem.

3.2 ELECTRONICS SUBSYSTEM

Electronics subsystem consists of microcontroller and stepper motor controller for each joint. Microcontroller gets program from the computer and generates step signals for stepper motor controllers.

3.3 SOFTWARE SUBSYSTEM

Software subsystem consists of tree components, two-legged robot simulator, **A**pplication **S**pecific **C**ompiler (ASC) and torque analyzer. Simulator simulates various movements of robot while ASC generates assembly language code from high level program written by using simple natural language like commands. Torque analyzer calculates torque required by each joint and also shows the plot for each joint.

4. MECHANICAL SUBSYSTEM

Mechanical subsystem consists of motor, gearbox, and links between joints. Motor and gearbox selection is the most difficult task since performance of the robot is completely depended on it. For making the development process faster and easier Computer Added Design (CAD) model has been developed showing location of each component of the robot.

4.1 MOTOR SELECTION

In most of the robots either servo motors or stepper motors are used. Using servo motor provides closed loop control but increases complexity of control system while stepper motor are easy to control and provides good open loop control. Also cost of servo motors is very high compared to stepper motor. So keeping all parameters in mind stepper motors has been selected as actuators.

4.1.1 Stepper Motors

A stepper motor is an electromechanical device which converts electrical pulses into discrete mechanical movements. The shaft or spindle of a stepper motor rotates in discrete step increments when electrical command pulses are applied to it in the proper sequence. The motors rotation has several direct relationships to these applied input pulses. The sequence of the applied pulses is directly related to the direction of motor shafts rotation. The speed of the motor shafts rotation is directly related to the frequency of the input pulses and the length of rotation is directly related to the number of input pulses applied. Stepper motors are commonly used in open loop systems since they provide reliable and accurate motion controlled by electric pulse given to it. There are two types of commonly used stepper motors

(1)Unipolar and (2) bipolar stepper motors.

Unipolar stepper motors are easy to control since all the windings are just turned on and off one after another to rotate the motor step by step, but they don’t provide higher torque compared to bipolar stepper motors.

Bipolar stepper motors provides high torque compared to simple DC motor and Unipolar stepper motor, but control system is more complex since for rotation of the motor winding polarities are required to be changed.

4.1.2 Specifications Of Stepper Motor

After comparing performance and cost of stepper motors from various manufacturers, stepper motors from Precision Motors have been selected. In the Table 4.1 values of parameters of the selected motor along with other motors are depicted. Column with gray background shows the specifications of selected motor. In Figure 4.1[10] torque characteristics and in Figure 4.2[10] dimensional specifications of the motor are shown.

Table 4.1 Specifications of stepper motors from precision motors

Specifications		**D-48-42-B20**	**D-48-42-B25**	**D-48-42-B28**
	Units	Bipolar Medium Torque	Bipolar High Torque	Bipolar High Torque
Operating Voltage	V	6	12	24
Resistance per phase		2.7	6	15
Inductance per phase	mH	3	6	12
Holding Torque	mNm(oz-in)	94.9 (13.47)	148 (21.1)	164 (23.3)
Detent Torque	mNm(oz-in)	18.5 (2.67)	18.5 (2.67)	18.5 (2.67)
Rotor Inertia	g-m2	25.6 x 10-4	25.6 x 10-4	25.6 x 10-4
Weight	gms (oz)	185 (6.52)	185 (6.52)	185 (6.52)
Step Angle	degrees	7.5	7.5	7.5
Step angle accuracy	o	+/- 0.5o	+/- 0.5o	+/- 0.5o
Max. operating temperature	oC	100	100	100
Dielectric strength	-	1000 VAC for 1 min.	1000 VAC for 1 min.	1000 VAC for 1 min.
End play	mm (in)	0.2 (0.008)	0.2 (0.008)	0.2 (0.008)

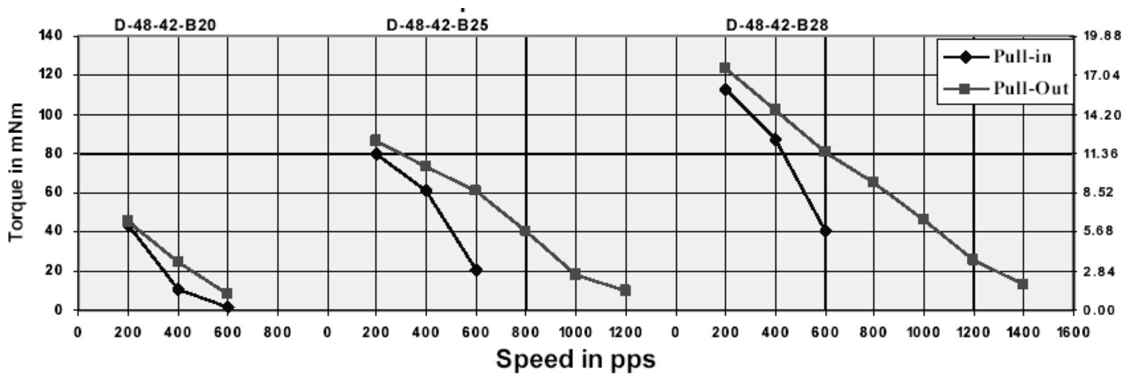

Figure 4.1 Torque characteristics of stepper motor D-48-42-B28

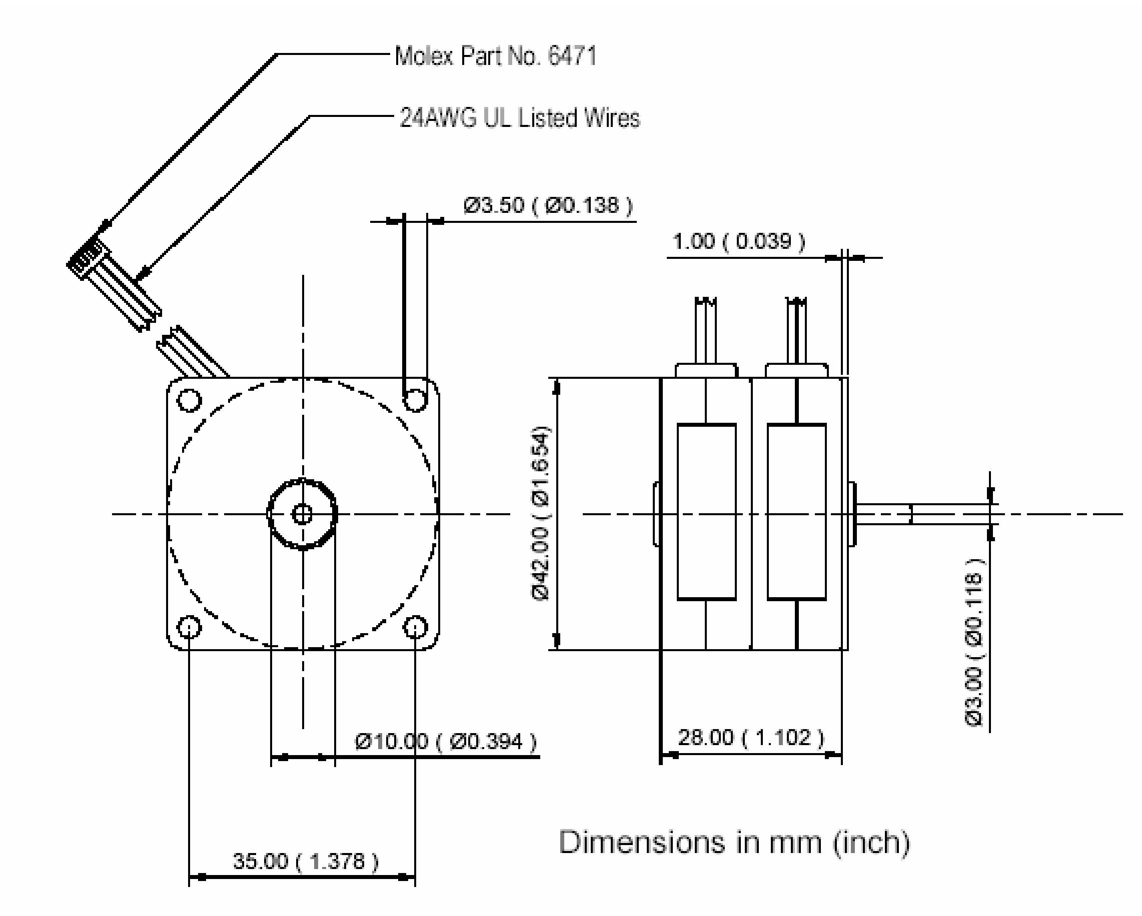

Figure 4.2 Dimensional specifications of stepper motor D-48-42-B28

4.2 GEARBOX SELECTION

Usually torque provided by stepper motor is enough for small applications where weight of components doesn't affect the design but here we need very high torque which can not be produces by such small stepper motor, so we need to combine it with Gearbox. In this section selected gearbox with its specifications and dimensional drawing is discussed. Selected gearbox is from an Indian manufacturer Mechtex and model no is GBH4

Gear head GB4 contains heavily loaded steel gear wheels. The spur gears rotate on fixed steel spindles which are hardened and polished to a mirror finish. The thick output shaft rotates in robust sintered bushings. It can also be mounted on a ball bearing which can be provided in the output bush of the gear box. All the gears are housed between two metal plates with a plastic frame. All bearings are permanently lubricated and therefore require no maintenance. This gear box can also be combined with small to medium sizes of DC and stepper motors. Specifications are shown in the Table 4.2[11] and dimensional drawing is shown in Figure 4.3[11].

Table 4.2 Specifications of gearbox GBH4

Gear Torque	kg-cm	100 (Gear Torque can be increased to 200 kg-cm on request)
Combination with Mechtex motors		small DC and stepper motors
Mounting		any position
Weight	gm	350
Axial thrust	kg	10
Radial torque	kg	80
Lateral torque	kg-cm	80
Output bearing		Sintered Bronze; Ball Bearing on request
Output shafts		dia. 8 x 18 mm (with a flat); others on request
Ambient temperature operation	˚C	-15 ... +55
Transmission Ratios		25/3, 10, 25/2, 50/3, 20, 25, 30, 100/3, 40, 125/3, 50, 125/2, 200/3, 250/3, 100, 120, 125, 200, 250, 300, 350.

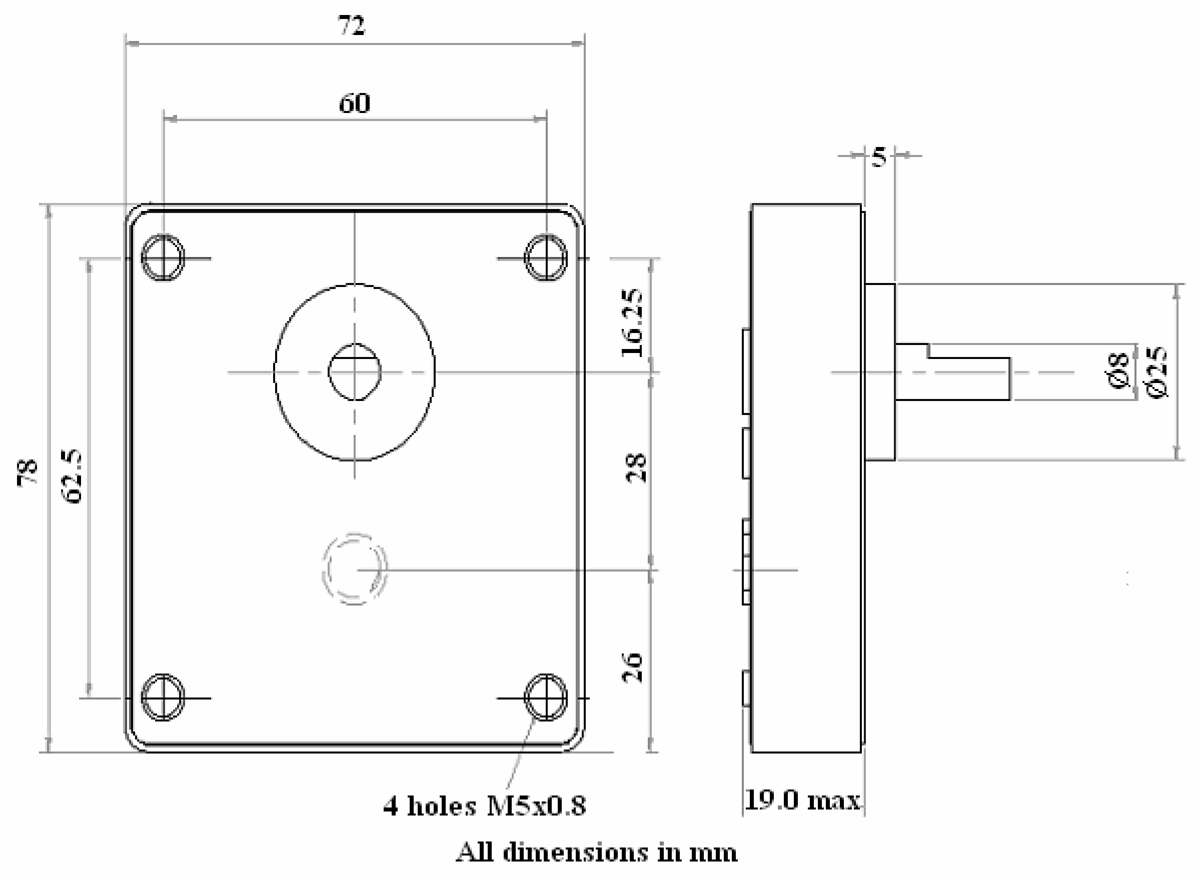

Figure 4.3 Dimensional drawing of gearbox GBH4

4.3 TORQUE CALCULATION

Two legged robot developed has 10 motors and each motor is responsible for movements at different joints. This section provides calculations of required torque by different motors. In the below Figure locations of different motors are shown. In Figure 4.4 different motors are responsible for different joints and movements. In the Figure 4.5 dimensional specifications of robot are shown.

For Left Leg:

- ML1: Up-Down movement of Ankle joint.
- ML2: Clock-Anti clock Rotation of Ankle joint.
- ML3: Movement of Knee joint.
- ML4: Up-Down movement of Pelvis joint.
- ML5: Clock-Anti clock rotation of pelvis joint.

For Right Leg:

- MR1: Up-Down movement of Ankle joint.
- MR2: Clock–Anti clock Rotation of Ankle joint.
- MR3: Movement of Knee joint.
- MR4: Up-Down movement of Pelvis joint.
- MR5: Clock-Anti clock rotation of pelvis joint

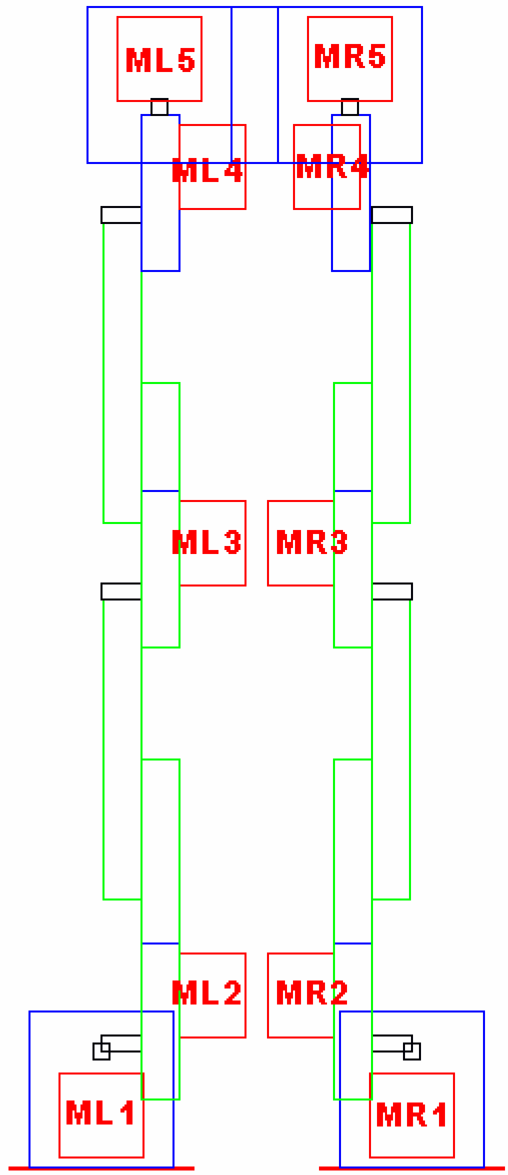

Figure 4.4 Locations of motors in robot

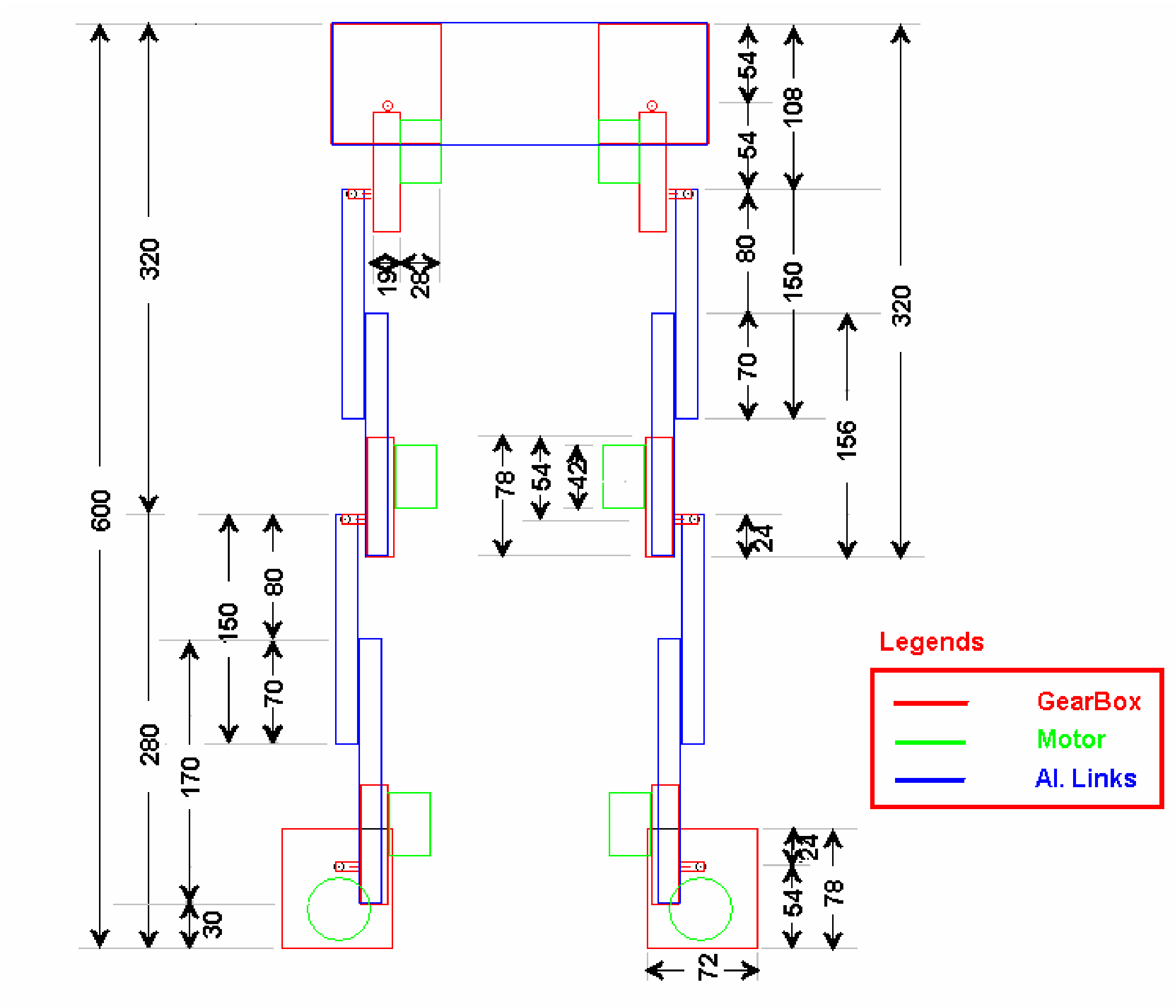

Figure 4.5 Dimensional specifications of robot

From Figure 4.5 we can see that total height is about 60 cm so distance of CG from any motor will not be more than 30 cm.

Torque calculations for each motor of left leg are shown below. Here from the specifications of selected motor and gearbox below mentioned parameters are taken for calculations.

- Motor Weight = 230 gram
- Gearbox Weight = 350 gram
- Controller Weight = 50 gram
- Other material(Aluminum links) weight = 1 kg

Now we calculate torque required by each motor.

For ML1: As shown in the Figure 4.6 this is the motor requiring maximum torque when it has to lift rest of the body to maintain Center of Gravity. It has to lift 9 motors, gearbox and all the aluminum links.

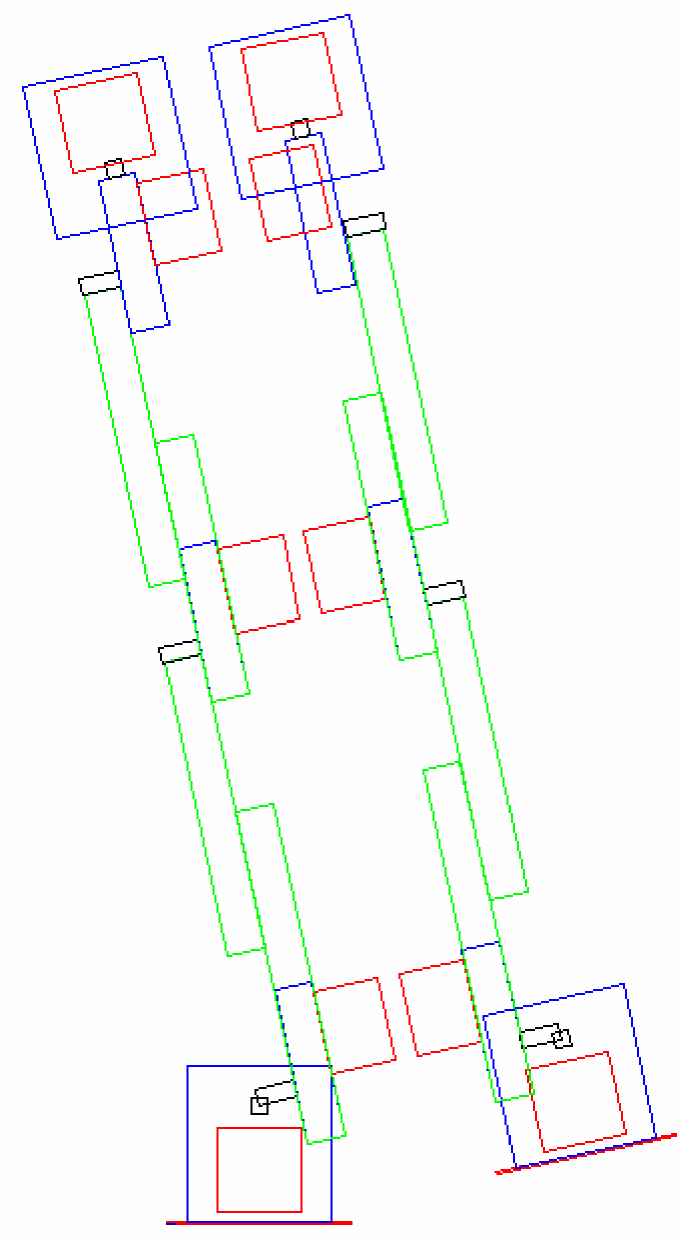

Figure 4.6 One leg up situation

So weight required to be lifted by this motor is

W = No of Motors *(Motor Weight + Controller Weight Gearbox weight) + Other Material Weight. (in gram)

= 9 * (185 + 50 + 350) + 1000

= 6265 gram

= 6.265 Kg

Now maximum torque required by this motor is

T = W * 30

= 6.265 * 30

=187.95 Kg-cm

For ML2: This motor has to lift 8 other motors, gearboxes and 4 aluminum links. So, weight required to be lifted by this motor is

W = No of Motors *(Motor Weight + Controller Weight Gearbox weight) + Other Material Weight. (in gram)

= 8 * (185 + 50 + 350) + 1000

=5680 gram

=5.68 Kg

Now maximum torque required by this motor is

T = W * 30

= 5.68 * 30

=170.4 Kg-cm

For ML3: This motor has to lift 7 other motors, gearboxes and 3 aluminum links. So, weight required to be lifted by this motor is

W = No of Motors *(Motor Weight + Controller Weight Gearbox weight) + Other Material Weight. (in gram)

= 7 * (185 + 50 + 350) + 1000

=5095 gram

=5.095 Kg

Now maximum torque required by this motor is

T = W * 30

= 5.095 * 30

=152.85 Kg-cm

For ML4: This motor has to lift 6 other motors, gearboxes and 2 aluminum links. So, weight required to be lifted by this motor is

W = No of Motors *(Motor Weight + Controller Weight Gearbox weight) + Other Material Weight. (in gram)

= 6 * (185 + 50 + 350) + 1000

=4510 gram

=4.51 Kg

Now maximum torque required by this motor is

T = W * 30

= 4.51 * 30

=135.3 Kg-cm

For ML5: This motor has to lift 5 other motors, gearboxes and 2 aluminum links. So, Weight required to be lifted by this motor is

W = No of Motors *(Motor Weight + Controller Weight Gearbox weight) + Other Material Weight. (in gram)

= 5 * (185 + 50 + 350) + 1000

= 3925 gram

= 3.925 Kg

Now maximum torque required by this motor is

T = W * 30

= 3.925 * 30

= 117.75 Kg-cm

Now let us calculate available torque by selected motor and gearbox combination. From the specifications of stepper motor and gearbox available specifications are as below.

For stepper motor

- Minimum motor torque = 1.24 Kg-cm
- Step angle = 7.5 degree

For gearbox

- Maximum gearbox torque = 200 Kg-cm
- Gear efficiency = 0.6

From above parameters we can see that stepper motor can't directly be used so we have to combine it with gearbox. Now suppose we assume that

- N is Gear Ratio
- To is Output torque at gearbox shaft
- Ti is input torque to gearbox
- Tm is torque produced by motor
- Ge is gearbox efficiency

Then,

To = Ti * Ge * N or **To = Tm * Ge * N**because **Tm=Ti.**

Here Ti = Tm = 1.24 Kg-cm

Ge = 0.6

To = 187.95 Kg-cm

=>187.95 = 1.24 * 0.6 * N

=>N = 252.62

So we can conclude that for using this stepper motor we need gearbox with gear ratio of at least 252.62.And hence gearbox GBH4 with gear ratio 250 has been selected.

4.4 MECHANICAL CAD MODEL

Before starting the development of the robot we need to develop a Computer Added Drawing which allows us to carry out the feasibility study and also helps in the development phase to decide the location of each component. It reduces the time required to develop the system once all the components and CAD drawing is available. CAD drawing includes positioning of different components like

- Stepper Motor
- Gear boxes
- Links

In Figure 4.7 depicts the simple CAD drawing of robot to be developed. This CAD drawing shows positions of Motors and Gearboxes and their connection with each other.

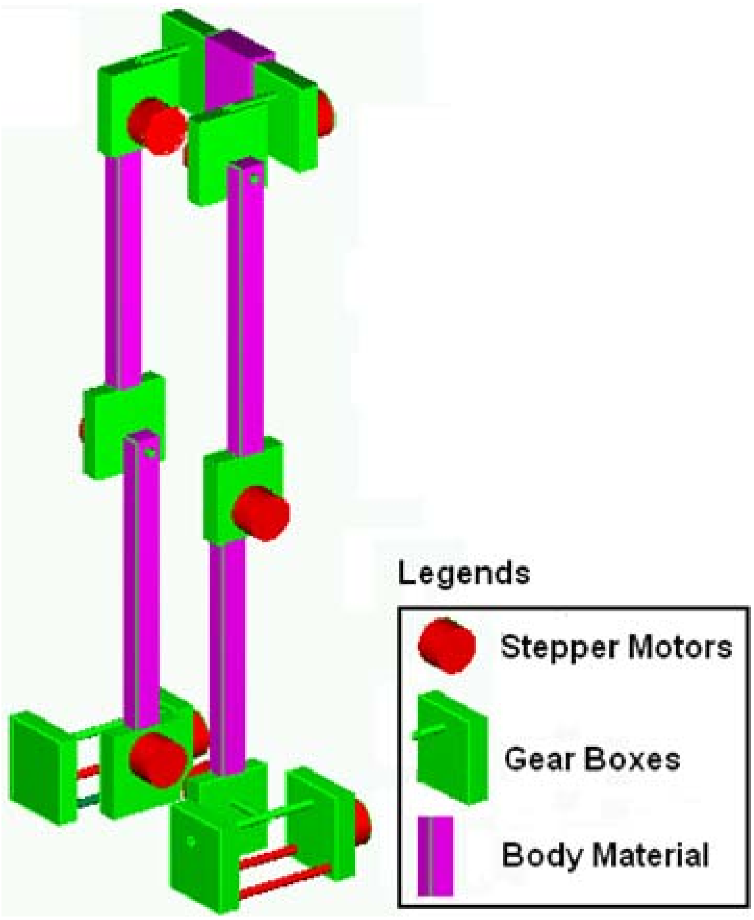

Figure 4.7 Mechanical CAD drawing of robot

4.5 FACTORS AFFECTING THE DESIGN

Since two legged robot have various parts where movements occurs, each requires different torque to provide movement. So the weight of different components affects the design and dimensions of the robot. Factors affecting the design are as below

- Weight of motors.
- Weight of Gearbox
- Torque of Motors

- **Weight of motors:** By studying various motors available from so many manufacturers it has been observed that weight of the stepper motor ranges from few hundred grams to few kilograms. But since two-legged robot have to lift weight of almost all the motors while rotating ankle joint so motor weight should be kept as low as possible. Next coming section gives information of parameters from one of the manufacturer.

- **Weight of Gearbox:** Generally weight of Gearbox is more then the motor weight but these should be minimized to decrease the torque required to be produced by the motor.
- **Torque of Motor:** Stepper Motors doesn't provide enough torque so that they can directly be used so we need to combine them with Gearbox with suitable gear ratio to increase the torque. In the previous section calculation of required torque and available torque of selected motor are discussed.

5. ELECTRONICS SUBSYSTEM

Electronics subsystem consists of microcontroller and stepper motor controller. It focuses on controlling the stepper motors to run them at required speed and required degree of rotation. Program is loaded from computer which is in form of binary hex file. Here in this project development board using 89S52 microcontroller and stepper motor controllers from Allegro Microsystems with model no A3982 SLB-T are used.

5.1 MICROCONTROLLER DEVELOPMENT BOARD

Microcontrollers are very easy to use and provide enough number of output ports. So, for controlling the stepper motors from most widely used microcontroller family 8051, Atmel 89S52 has been selected. In the next section architecture and features of 89S52 are discussed.

5.1.1 AT89S52 Architecture and Features

The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Figure 5.1 pin diagram and in Figure 5.2 architecture of AT89S52 is shown.

❖ **Features of AT89S52**

- 8K Bytes of In-System Programmable (ISP) Flash Memory
    - o Endurance: 1000 Write/Erase Cycles
- 4.0V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)
- Green (Pb/Halide-free) Packaging Option

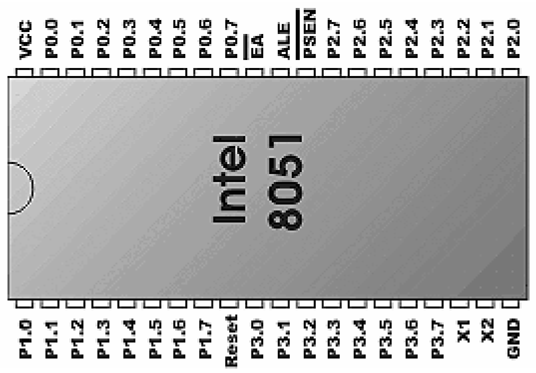

Figure 5.1 Pin Configuration of AT89S52

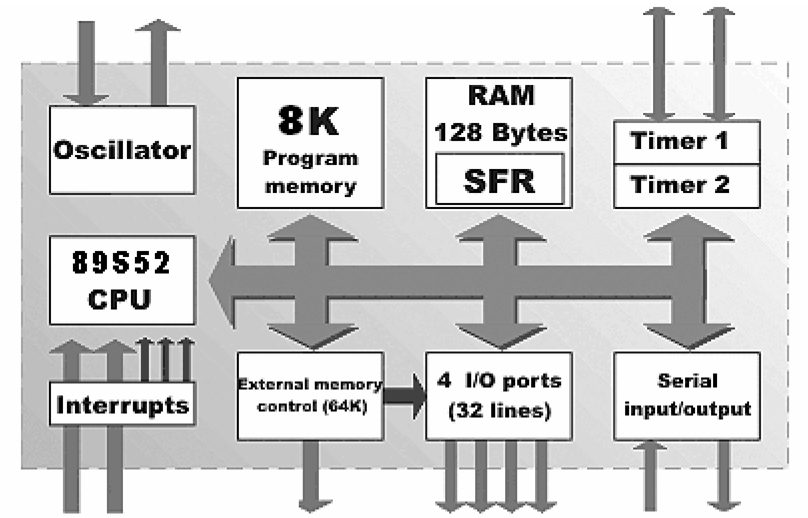

Figure 5.2 AT89S52 Architecture

5.1.2 Printed Circuit Board

For reducing the complexity of connections Printed Circuit Board (PCB) for microcontroller 89S52 has been developed. PCB has serial communication capability, in system programming capability by using which without removing the microcontroller from the board it can be programmed via parallel port of the computer by using programming application. In the Figure 5.3 layout of the PCB is shown while in the Figure 5.4 actual microcontroller board is shown.

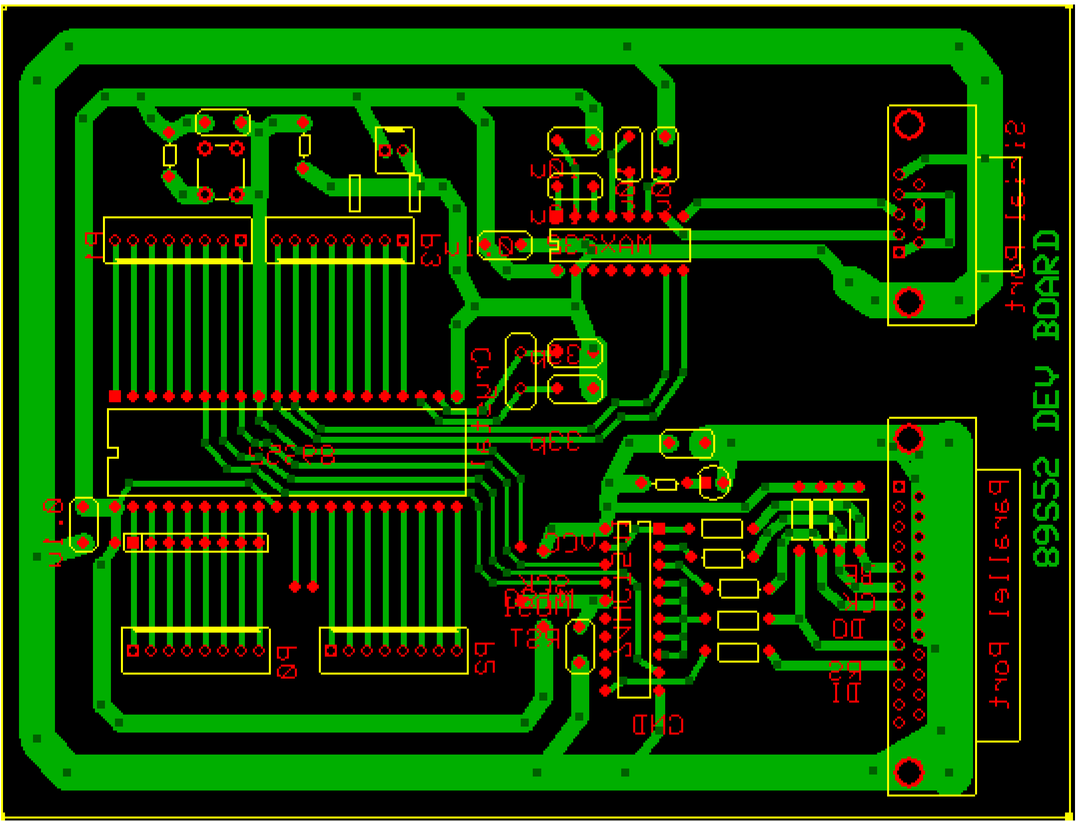

Figure 5.3 PCB for AT89S52

5.1.3 89S52 Microcontroller Board

For easier connection of output pins of the microcontroller connector for each port and power supply port are mounted on the PCB as shown in the Figure 5.3. For in system programming parallel port and for serial communication capability DB9 connector is mounted. All the ports are highlighted in the Figure 5.4 and all other components are also shown.

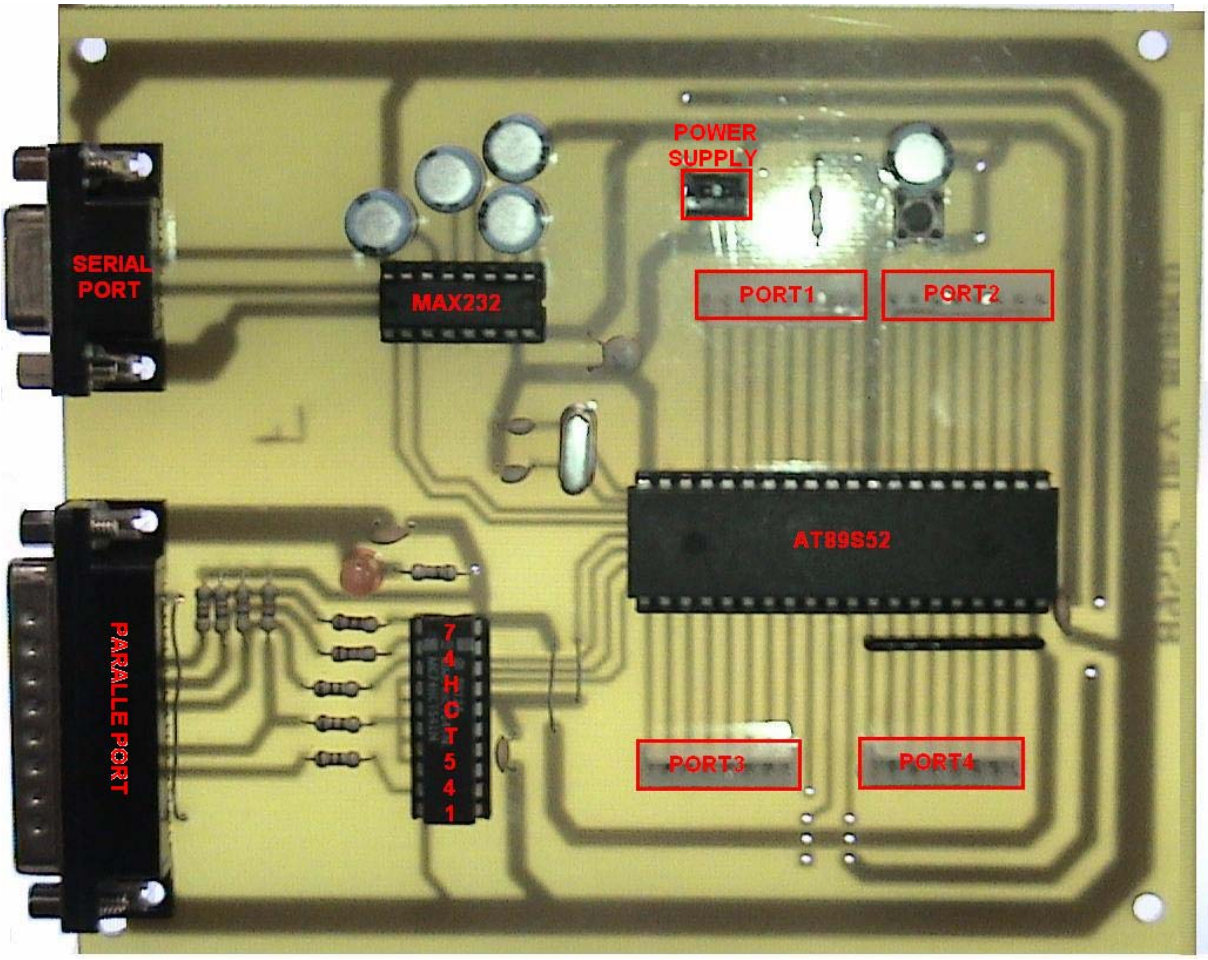

Figure 5.4 AT89S52 microcontroller board

5.2 STEPPER MOTOR CONTROLLER

For controlling the bipolar stepper motor without using any controller requires two control signals per winding to control the polarity of the inputs. In this project 10 stepper motors are used so just by using micro controller all the motors can not be controlled. So, stepper motor controller has been used to reduce required pin count.

Stepper motor controller

- Can control stepper motor just by 2 signals instead of two for each coil which ranges from 4 to 8
- Two signals per motor
  - Step (CLOCK)
  - Direction (CW/ACW)
- Reduces programming complexity

5.2.1 Allegro A3982 SLB-T

The A3982 is a complete stepper motor driver with built-in translator for easy operation. It is designed to operate bipolar stepper motors in full- and half-step modes, with an output drive capacity of up to 35 V and ±2 A. The A3982 includes a fixed off-time current regulator which has the ability to operate in Slow or Mixed decay modes. The translator is the key to the easy implementation of the A3982. Simply inputting one pulse on the STEP input drives the motor one step. There are no phase sequence tables, high frequency control lines, or complex interfaces to program. The A3982 interface is an ideal fit for applications where a complex microprocessor is unavailable or is overburdened.

The chopping control in the A3982 automatically selects the current decay mode (Slow or Mixed). When a signal occurs at the STEP input pin, the A3982 determines if that step results in a higher or lower current in each of the motor phases. If the change is to a higher current, then the decay mode is set to Slow decay. If the change is to a lower current, then the current decay is set to Mixed (set initially to a fast decay for a period amounting to 31.25% of the fixed off-time, then to a slow decay for the remainder of the off-time). This current decay control scheme results in reduced audible motor noise, increased step accuracy, and reduced power dissipation.

Internal synchronous rectification control circuitry is provided to improve power dissipation during PWM operation. Internal circuit protection includes: thermal shutdown with hysteresis, under voltage lockout (UVLO), and crossover-current protection. Special power-on sequencing is not required. The A3982 is supplied in a 24-pin wide-body SOIC (package *LB*) with internally-fused power ground leads. It is also available in a lead (Pb) free version (suffix –*T*), with 100% matte tin plated lead frames. In the Figure 5.5 functional block diagram of Allegro A3982 SLB-T is shown.

Some of the features of A-3982 are as below

- Low RDS(ON) outputs
- Automatic current decay mode detection/selection
- Mixed and Slow current decay modes

- Synchronous rectification for low power dissipation
- Internal UVLO and thermal shutdown circuitry
- Crossover-current protection

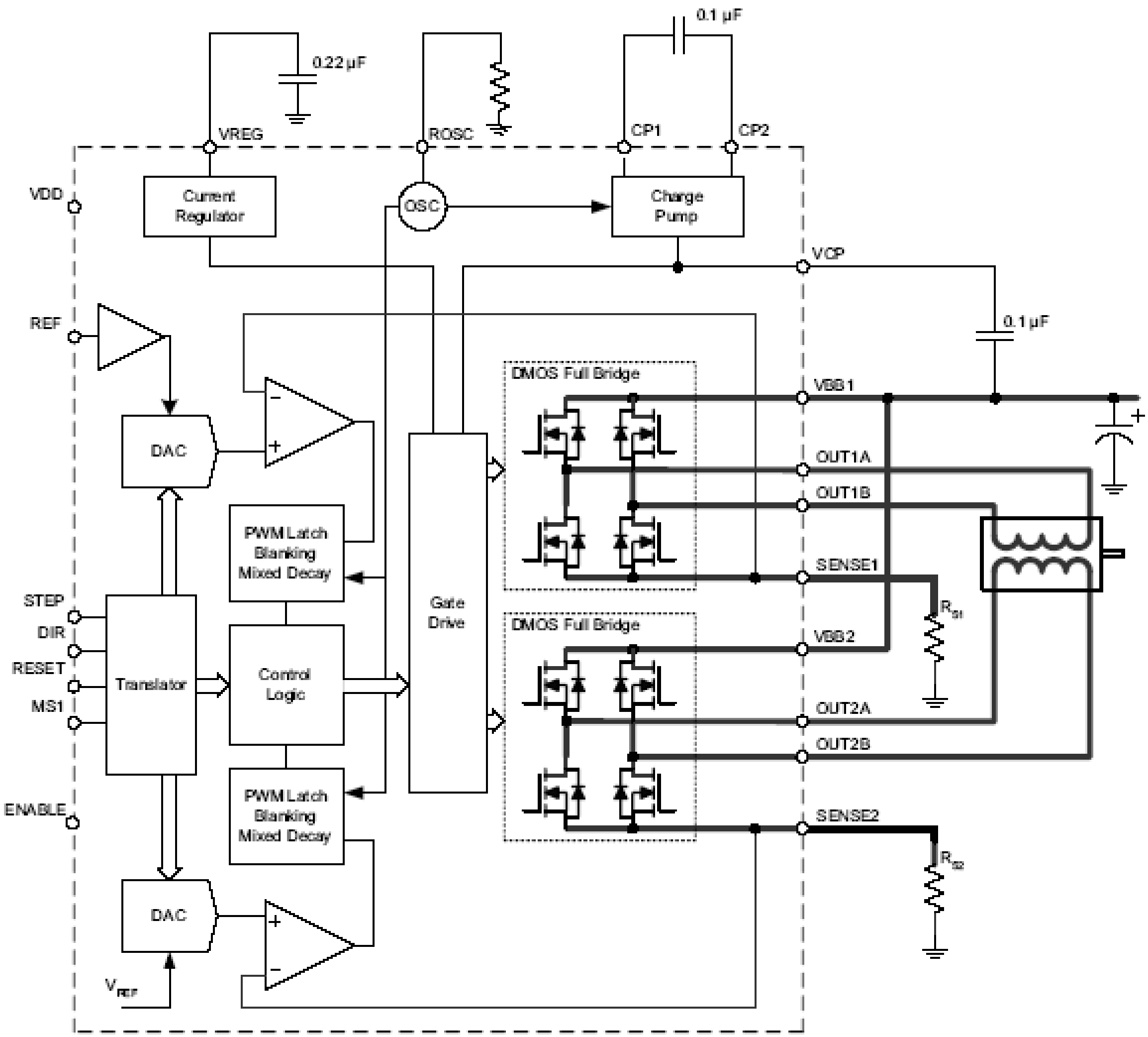

Figure 5.5 Functional block diagram of A-3982 SLB-T

Here in above circuit diagram some important signals are as below.

- STEP : Signal responsible for step rotation of the motor
- DIR: State of this signal decides the direction of rotation of motor.
- VBB1 : Voltage applied to controller and these are used to drive the motor
- OUT1A and OUT1 B : Outputs connected to Motor winding1
- OUT2A and OUT2 B : Outputs connected to Motor winding2

Block representation of the stepper motor controller is as shown in the Figure 5.6.

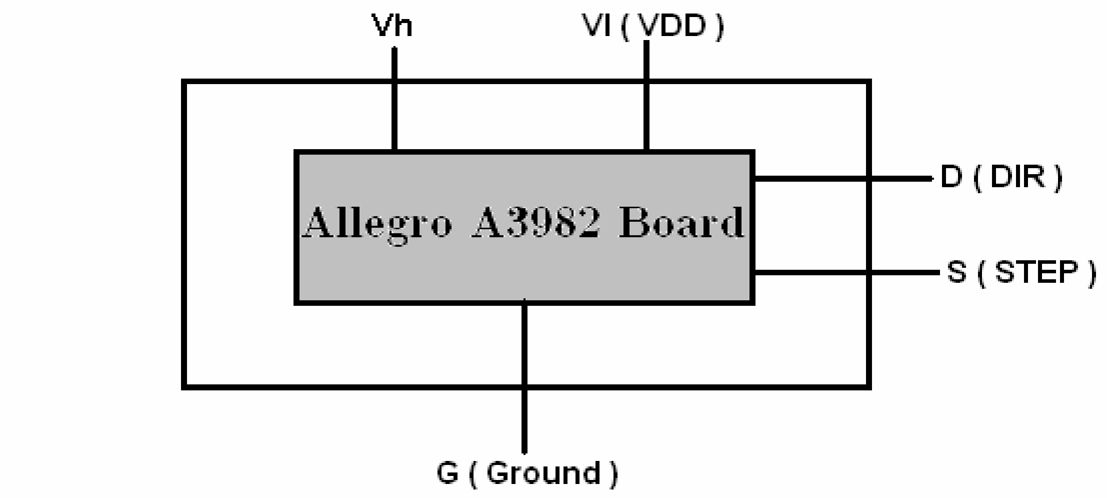

Figure 5.6 Block diagram of the stepper motor controller

Above shown block representation is used in the integration diagram of the complete system shown in the Figure 5.7.

5.2.2 Integration of Microcontroller and Stepper Motor Controller

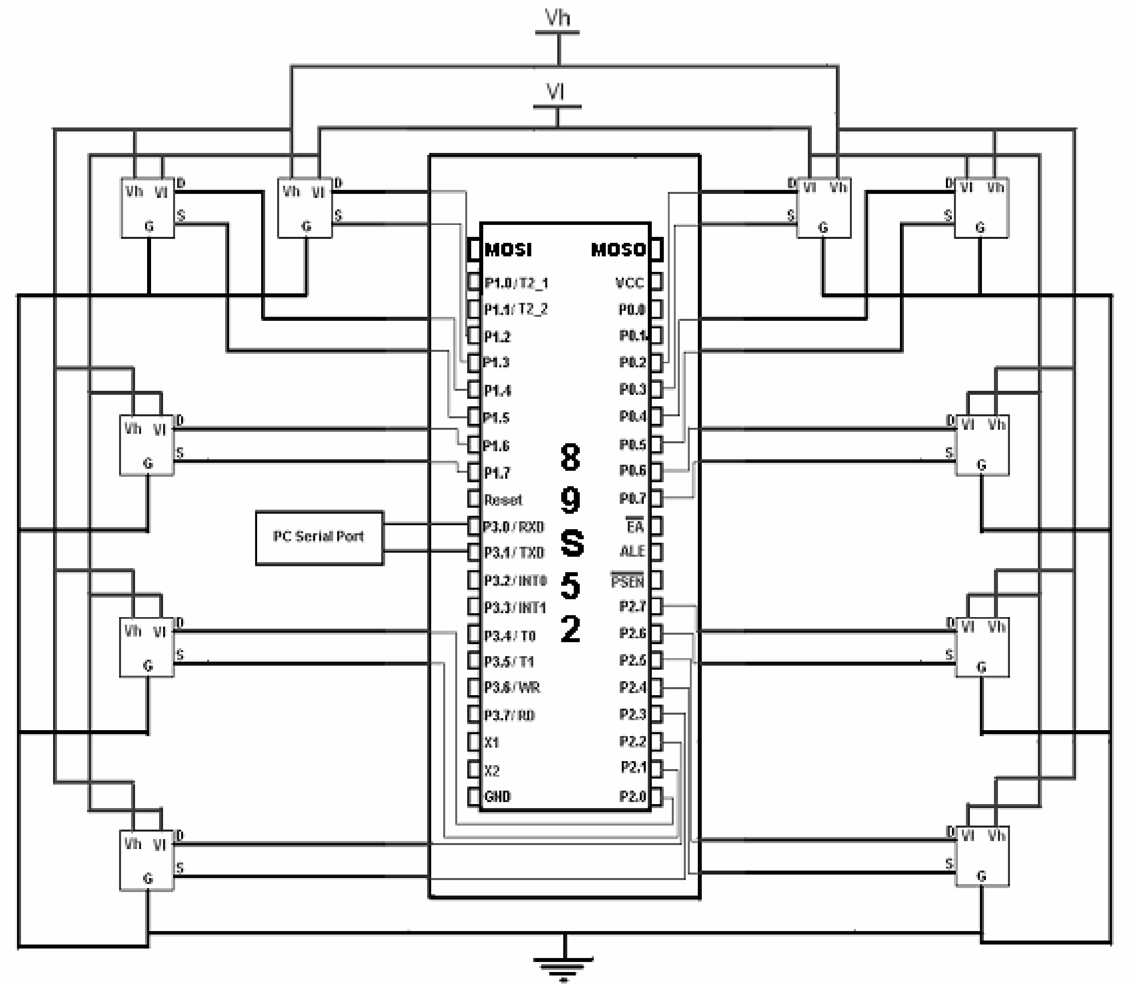

Figure 5.7 Integration diagram of the electronics subsystem

In the Figure 5.7 connections of all stepper motor controllers with microcontroller are shown. Each stepper motor controller gets two signals from microcontrollers which are STEP and DIR(Direction).On the microcontroller pins labeled P*.* are output port pins where stepper motor controllers are connected. As per the program loaded in the microcontroller it generates direction and pulse signal for each motor.

6. SOFTWARE SUBSYSTEM

Software subsystem consists of various modules like two-legged robot simulator, Application specific compiler, torque analyzer and stepper motor control module. Each of these modules is described in the following sections.

6.1 TWO-LEGGED ROBOT SIMULATOR

Simulators are very useful since they provide easy way to carry out feasibility study and also are helpful for reducing price of development. Developed simulator has very rich Graphical User Interface with buttons to control each joint of the robot.

Developed Simulators have below mentioned features

- Simulates movements of different parts of body
- Can be used to analyze movements and their effects on centre of gravity
- Shows 3D image of the robot on computer screen as per the dimensions in the CAD model.
- Provides GUI with buttons for applying movements to different parts.
- Provides facility to supply multiple instructions to be executed in parallel or sequentially.

6.1.1 Class Hierarchy of the Simulator.

Robot simulator is developed in JAVA which is a platform independent, Object Oriented Programming (OOP) language. In the Figure 6.1 class hierarchy is shown and each class's functionality is described in the next section.

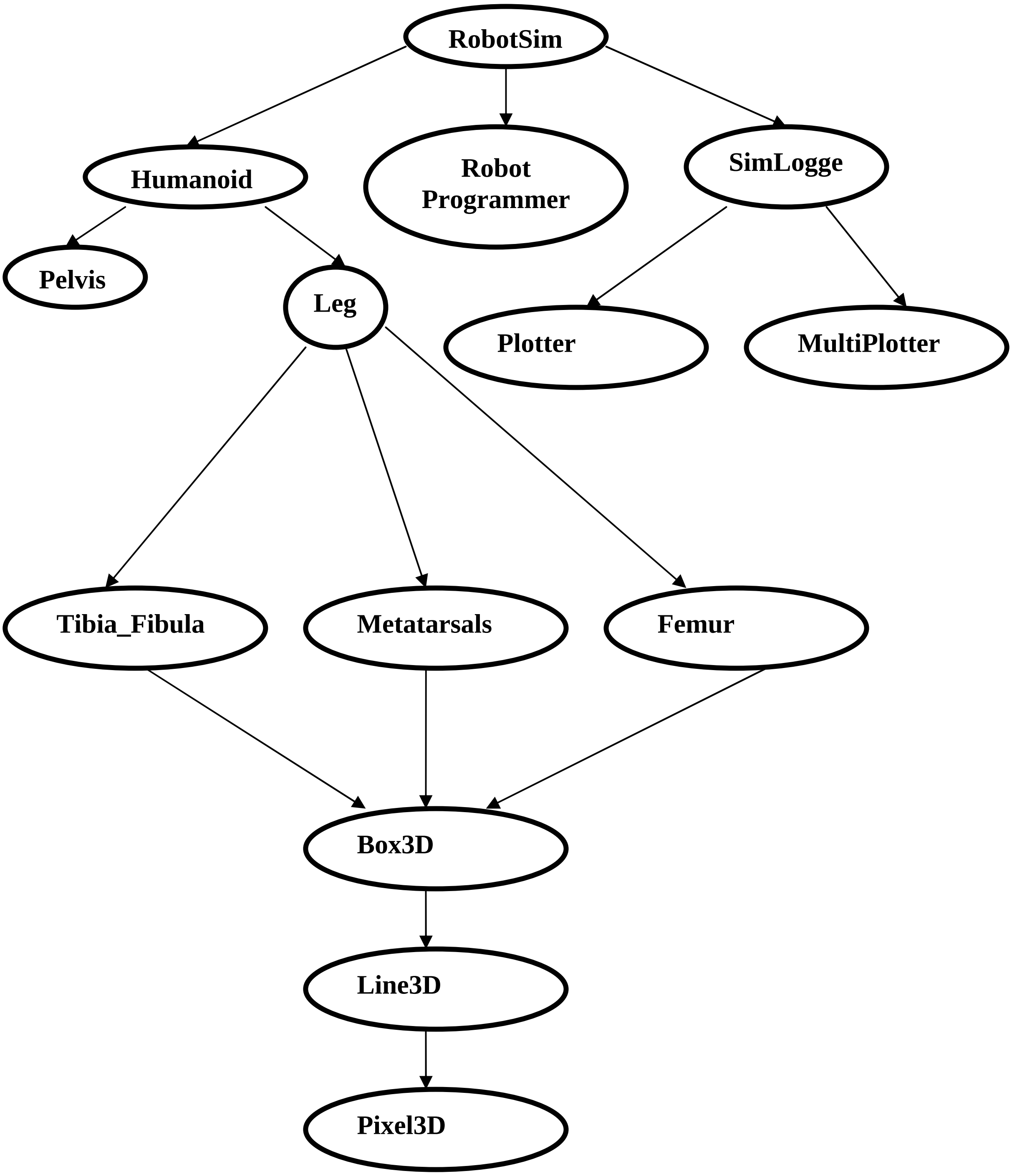

Figure 6.1 Class hierarchy of robot simulator

6.1.2 Functionalities Provided By Classes Of Simulator

Each class in the Simulator is responsible for providing different functionalities. All the classes can be divided in to four categories.

(1) Classes for Graphical User Interface

(2) Classes for Graphical Robot display

(3) Classes for Application specific compiler
(4) Classes for Handling and displaying simulation data

Each class's functionalities as below.

(1) Classes for Graphical User Interface

- Class Name: RobotSim
  - Functionalities: This class is responsible for providing main screen of Graphical User Interface which consists of buttons for controlling the robot, another tab for Application Specific Compiler and one tab for torque analyzer. This class initializes robot and all other classes.

(2) Classes for Graphical Robot display

- Class Name: Humanoid
  - Functionalities: This class is responsible for creating the wire-frame model of the robot. It has functions to rotate, translate and display the robot on the screen.
- Class Name: Leg
  - This class creates graphical representation of the Leg of the robot as per the size specified, this class also has functionality to rotate, translate, move up-down the leg.
- Class Name: Pelvis
  - This class creates graphical representation of pelvis joint of the robot as per the size specified.
- Class Name: Femur
  - This class creates graphical representation of the Femur link of the robot as per the size specified.
- Class Name: Tibia_Fibula
  - This class creates graphical representation of the Tibia Fibula link of the robot as per the size specified.
- Class Name: Metatarsals
  - This class creates graphical representation of the Metatarsals part of the robot as per the size specified.
- Class Name: Box3D
  - This is the basic class used by all other classes of this category to create graphical representation. This class provides functionalities to

translate and rotate the 3-Dimensinal box about X-Axes, Y-Axes, Z-Axes and about any arbitrary line.

- Class Name: Line3D
  - This is the basic class used by Box3D class to create graphical representation of 3-Dimensional box. This class provides functionalities to create the line in 3-Dimensinal space. It also provides facility to translate and rotate the line about X-Axes, Y-Axes, Z-Axes and about any arbitrary line.
- Class Name: Pixel3D
  - This is the basic class used by Line3D class to create graphical representation of line in 3-Dimensional space. This class provides functionalities to create the pixel in 3-Dimensinal space. It also provides facility to translate and rotate the pixel about X-Axes, Y-Axes, Z-Axes and about any arbitrary line.

(3) Classes for Application Specific Compiler

- Class Name: RobotProgrammer
  - This class is responsible for compiling the program written in high level language to assembly language of target microcontroller. It also generates intermediate code for simulation of the same code, which is used by RobotSim class.

(4)Classes for Handling and displaying simulation data

- Class Name: SimLogger
  - This class stores the simulation data like torque for each joint during simulation. It uses classes plotter and multiplotter to display the stored data in form of graphs.
- Class Name: Plotter
  - This class creates graph representation of provided data. It can plot only one series of data.
- Class Name: MultiPlotter
  - This class creates graph representation of provided data. It can plot multiple series of data.

6.1.3 Using Simulator

In the Figure 6.2 snapshot of the robot simulator is shown. As shown in the Figure, buttons on the left side are used to apply movements on the robot, while in the centre wire-frame model of the robot is shown. On the right side buttons to control the movements of right leg are shown. On the top side of window RESET ROBOT button is shown which is used to reinitialize the robot while another button is used to change the view of robot. At the bottom side of the window four buttons are shown which are used to translate and rotate the robot about x-axes, y-axes and z-axes.

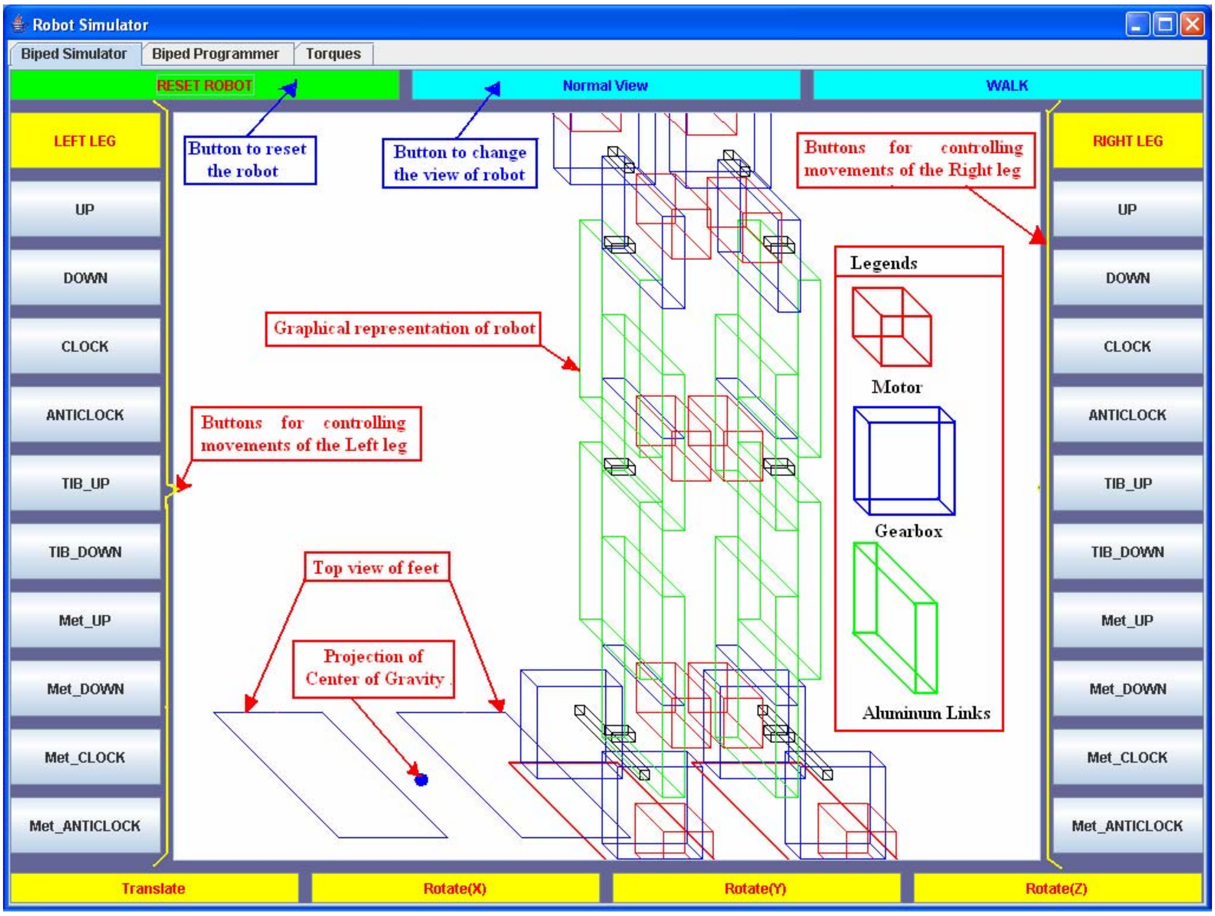

Figure 6.2 Snapshot of robot simulator

Developed two-legged robot simulator provides the facility to change the current view of robot being shown on the computer screen. It supports four different views of the robot as below

(1) Isometric view (2) Front view

(3) Side view (4) Top view

In the Figure 6.3 isometric view of robot is shown. In the isometric view the relative size of all the components are clearly visible. This view can be used to observe pelvis joint.

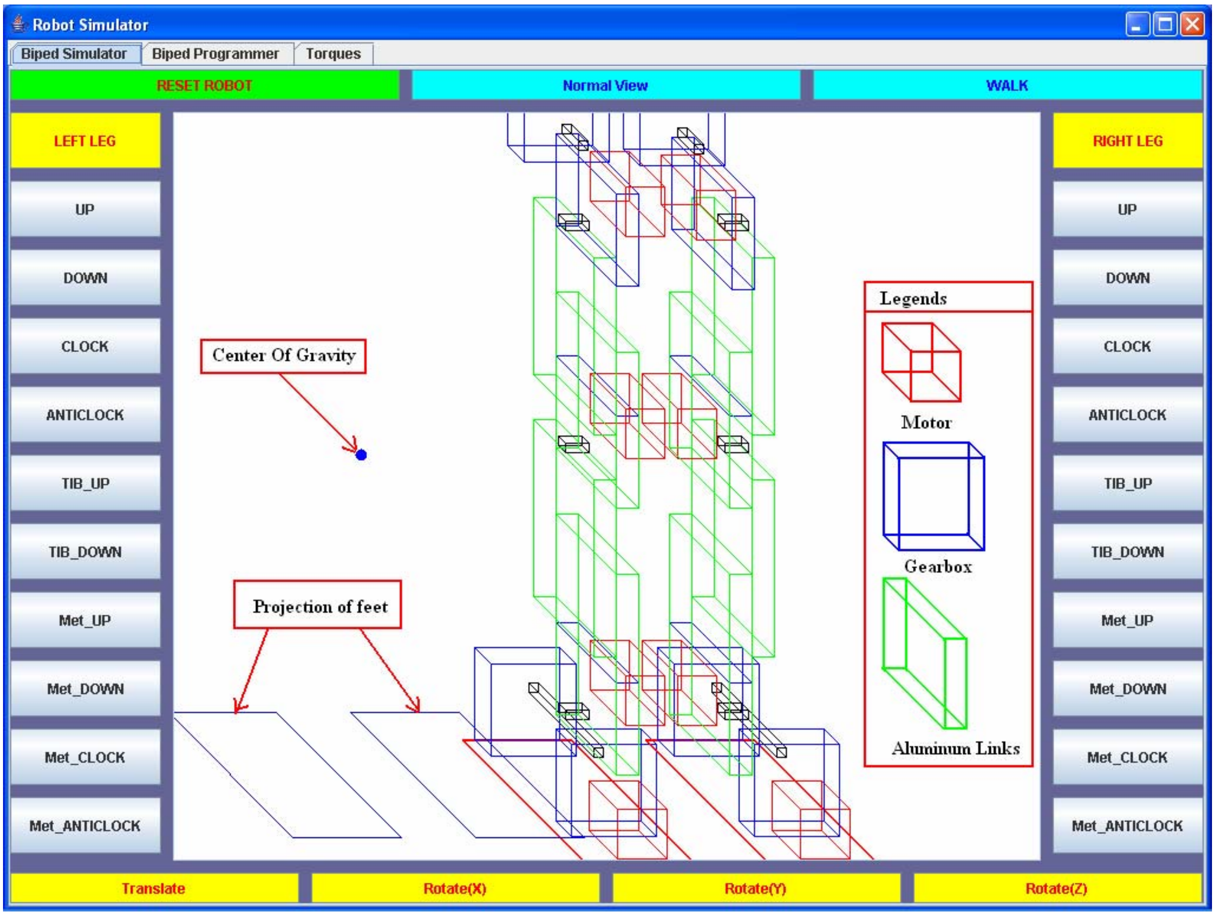

Figure 6.3 Isometric view of robot

In the Figure 6.4 front view of the robot is shown. In this view all the motors and gearboxes are clearly visible and also all the joint links are clearly visible.

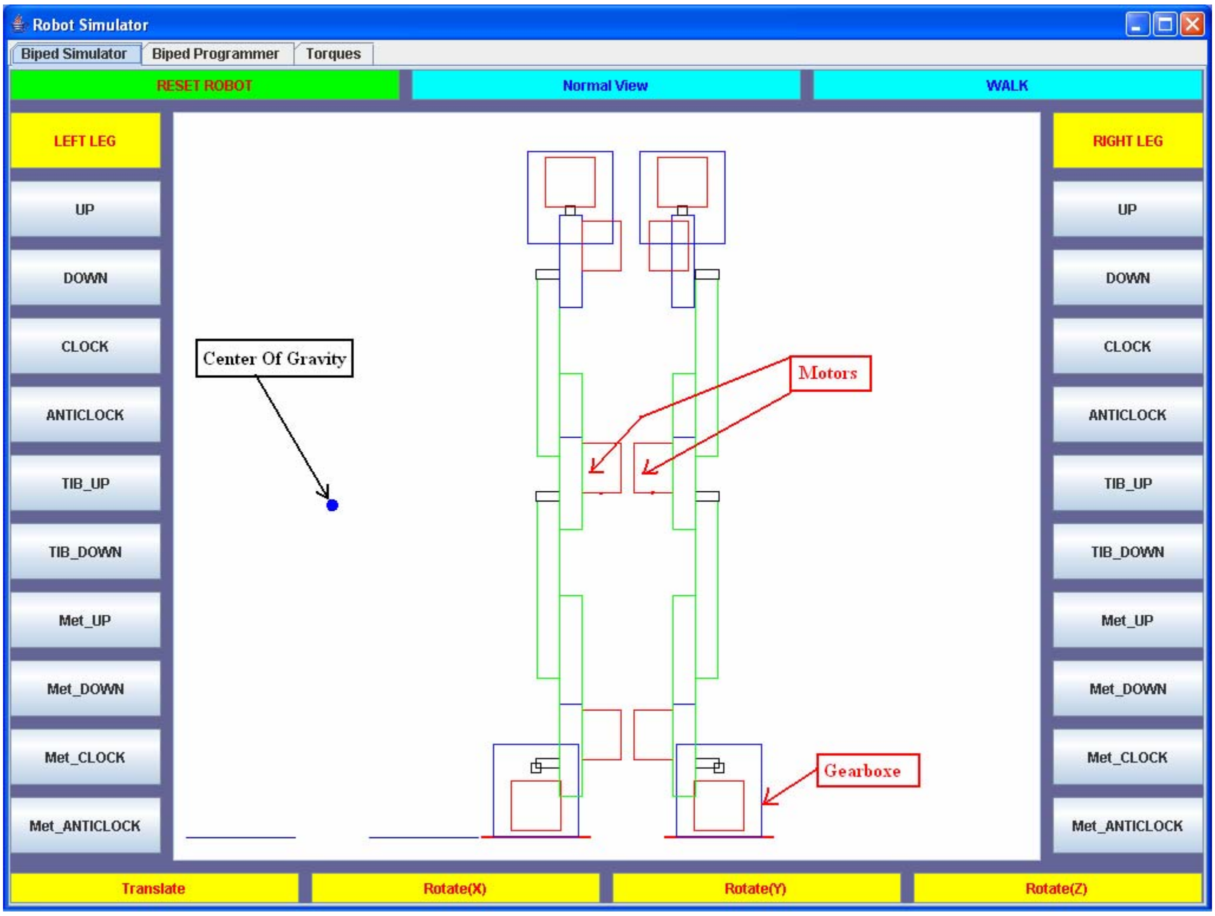

Figure 6.4 Front view of robot

In the Figure 6.5 side view of the robot is shown. This view can be used to observe the situations of knee joint, ankle joint and pelvis joint.

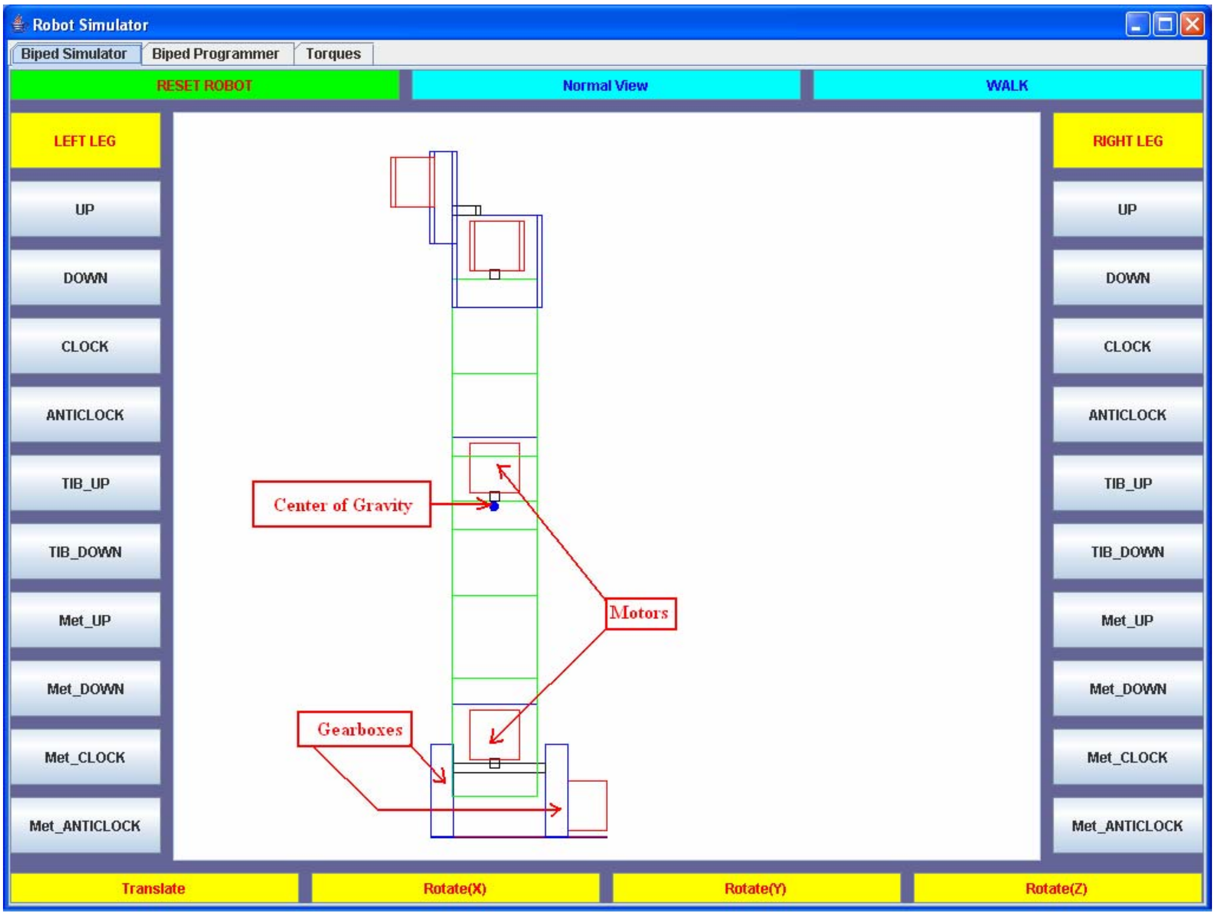

Figure 6.5 Side view of robot

In the Figure 6.6 top view of the robot is shown. This view can be used to observe relative locations of projection of Center of Gravity and projection of feet support area.

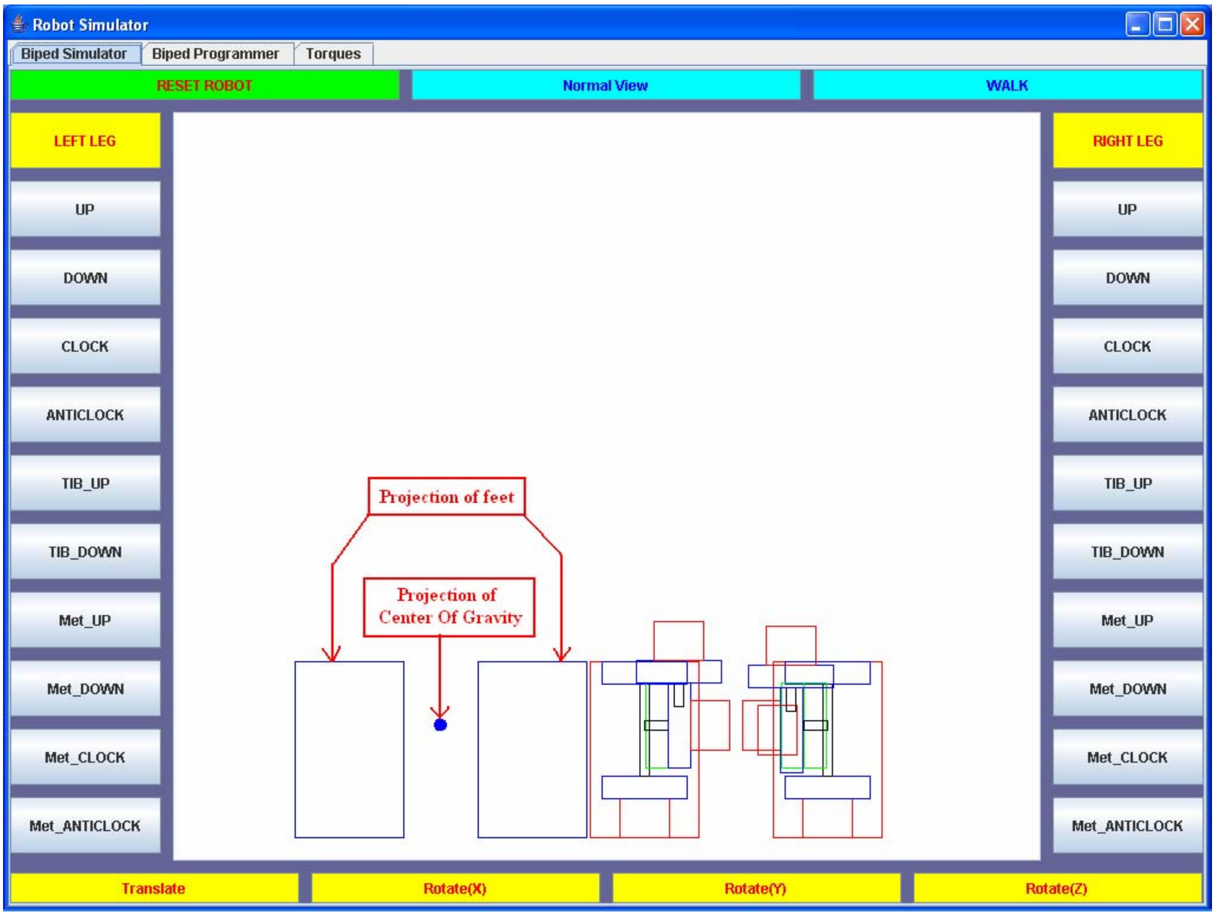

Figure 6.6 Top view of robot

6.2 APPLICATION SPECIFIC COMPILER

Compiler is a program which translates the code written in one language to another language. Usually input to compiler is high level language code like C , C++ or JAVA code and out put is object code or executable code for the target operating system or processor. Application Specific Compiler (ASC) is a special type of compiler which has input in form of program written using special instruction for specific system. ASC generates very low level (assembly) language code from the high level language code.

In this project an ASC for two legged robot has been developed. This compiler accepts program written with specific instructions for movement of different joints of robot. In the Table 6.1 instructions supported by the ASC and the joint actuated by the instruction is shown. ASC translates the input program to assembly language program for target microcontroller (89S52). Its has capability to execute more than one program instruction simultaneously. All the program lines are separated with ‘;’ while each instruction of the same line is separated with ‘,’. All the instructions in the same line are executed simultaneously. In the Figure 6.7 flow chart of the ASC is shown.

Table 6.1 Instruction set of ASC

Instruction	Joint Actuated
ROTATE LEFT ANKLE CLOCKWISE BY	Left Ankle(Rotate)
ROTATE LEFT ANKLE ANTI-CLOCKWISE BY	Left Ankle(Rotate)
MOVE LEFT ANKLE UP BY	Left Ankle(Move)
MOVE LEFT ANKLE DOWN BY	Left Ankle(Move)
MOVE LEFT KNEE UP BY	Left Knee
MOVE LEFT KNEE DOWN BY	Left Knee
ROTATE LEFT LEG CLOCKWISE BY	Left Leg(Rotate)
ROTATE LEFT LEG ANTI-CLOCKWISE BY	Left Leg(Rotate)
MOVE LEFT LEG UP BY	Left Leg(Move)
MOVE LEFT LEG DOWN BY	Left Leg(Move)
ROTATE RIGHT ANKLE CLOCKWISE BY	Right Ankle(Rotate)
ROTATE RIGHT ANKLE ANTI-CLOCKWISE BY	Right Ankle(Rotate)
MOVE RIGHT ANKLE UP BY	Right Ankle(Move)
MOVE RIGHT ANKLE DOWN BY	Right Ankle(Move)
MOVE RIGHT KNEE UP BY	Right Knee
MOVE RIGHT KNEE DOWN BY	Right Knee
ROTATE RIGHT LEG CLOCKWISE BY	Right Leg(Rotate)
ROTATE RIGHT LEG ANTI-CLOCKWISE BY	Right Leg(Rotate)
MOVE RIGHT LEG UP BY	Right Leg(Move)
MOVE RIGHT LEG DOWN BY	Right Leg(Move)

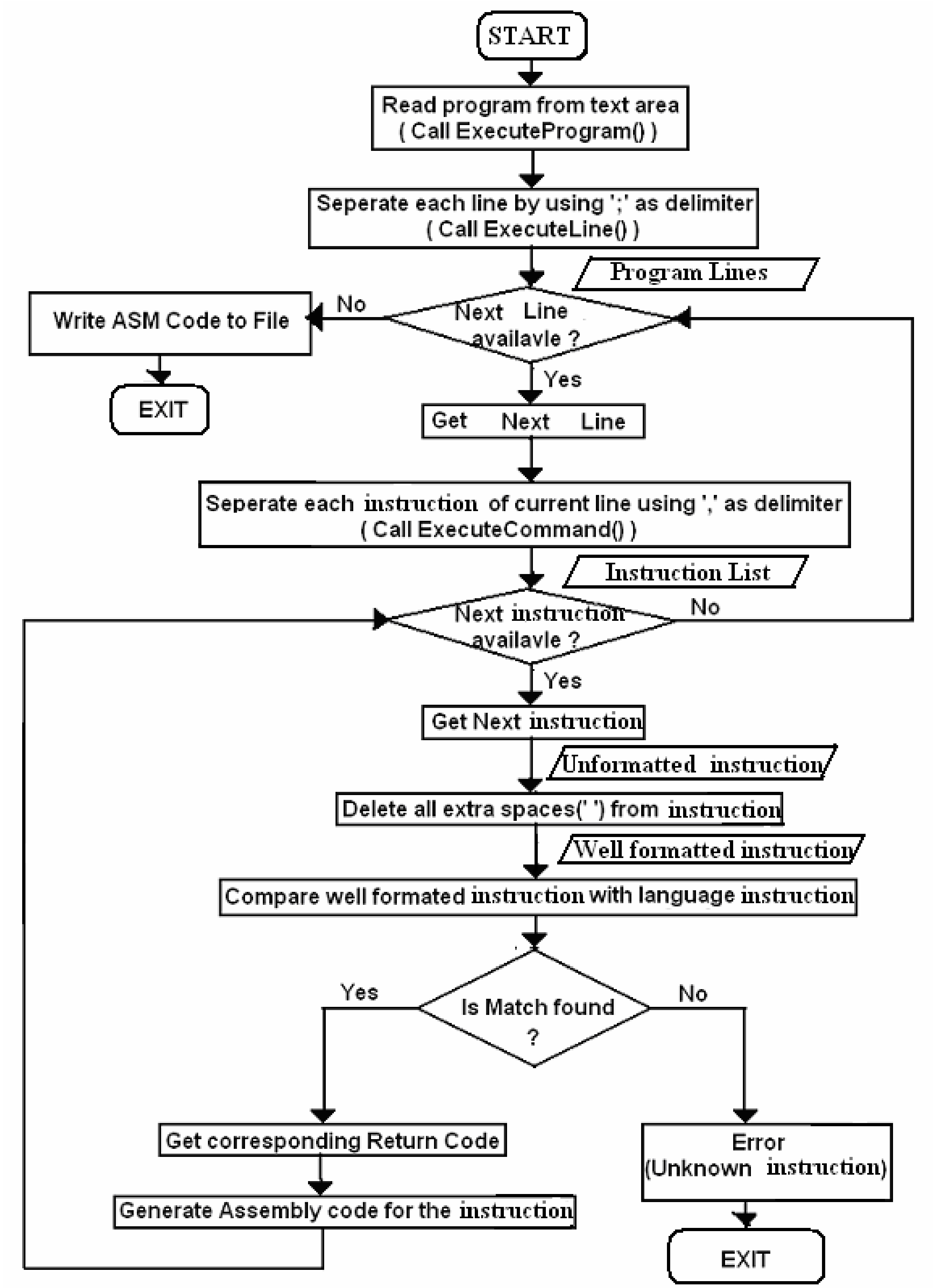

Figure 6.7 Flow chart of Application Specific Compiler

In the Figure 6.8 snapshot of the Application Specific Compiler GUI is shown. As shown in the Figure in the text area program using special instructions for two legged robot are written which is compiled by pressing Compile button.

Generated assembly language code is written to file specified in the text field at the top side of the window.

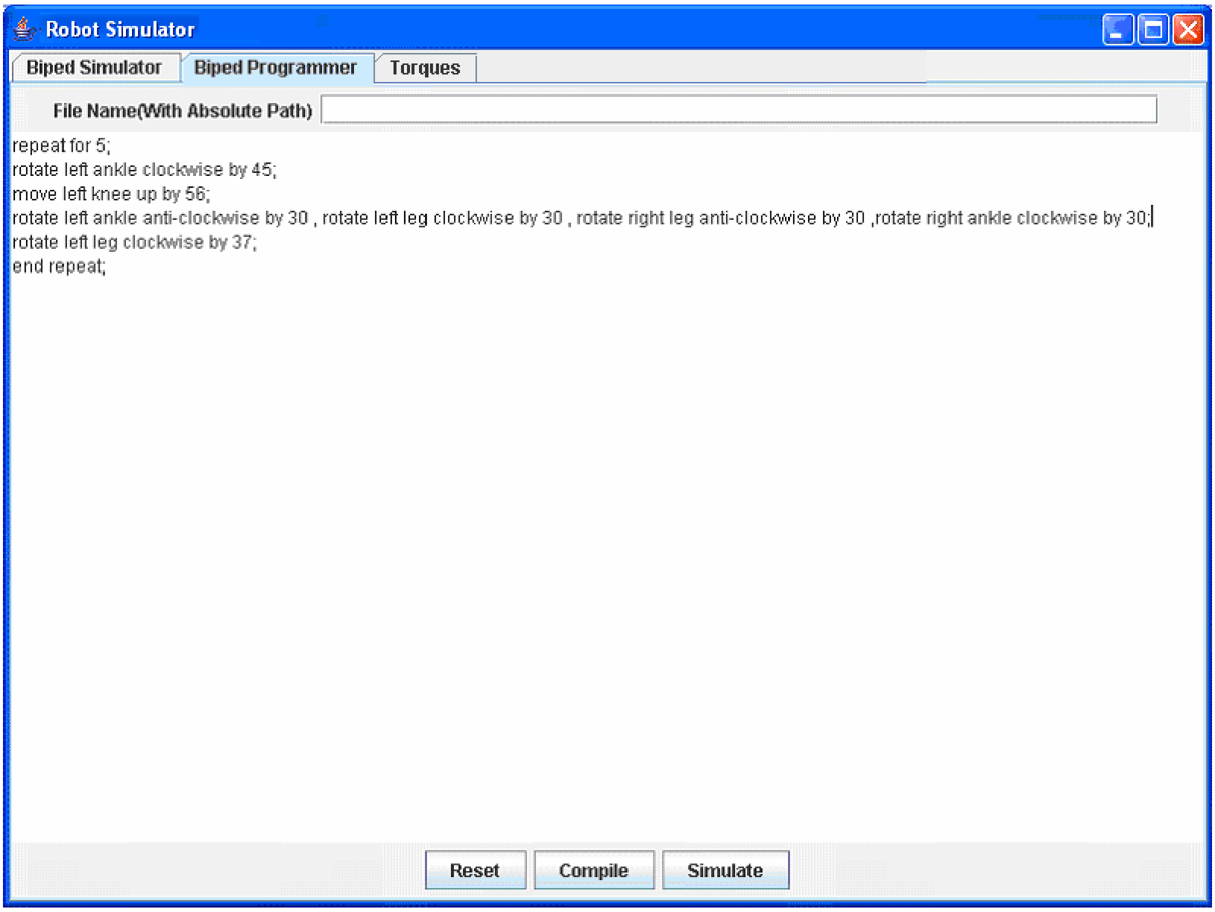

Figure 6.8 Snapshot of Application Specific Compiler

As shown in the Figure 6.8 at the bottom there is a Simulate button on pressing this button code written in the text area is simulated and program switches to simulator view.

6.3 Torque Analyzer

Simulator also contains utility to record torque required by each joint while simulating any program in ASC. It shows the stored torque data in form of graph for each joint. It can be used to analyze the torque required by each joint for different programs. Torque required for executing below mentioned program for left ankle and for all the joints is shown in the Figure 6.9 and 6.10 respectively.

➢ Program Executed by ASC:

rotate left ankle anti-clockwise by 35 , rotate left leg clockwise by 35 , rotate right leg anti-clockwise by 35 , rotate right ankle clockwise by 35;

move right leg up by 60 , move right knee down by 30;move right leg up by 32 , move left leg down by 32 , move right knee down by 32 , move left ankle down by 40;

move right knee up by 32 , move right ankle down by 40;

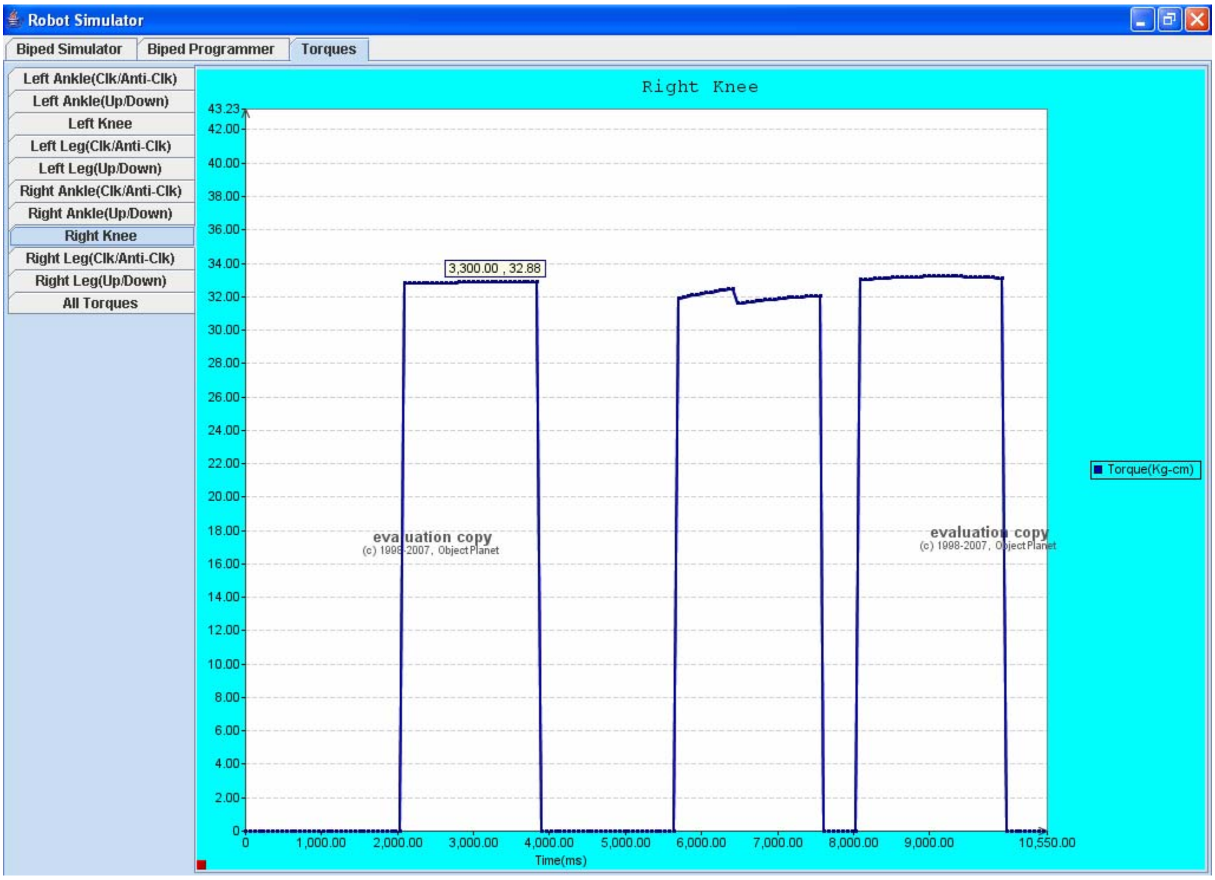

Figure 6.9 Torque required at right knee joint

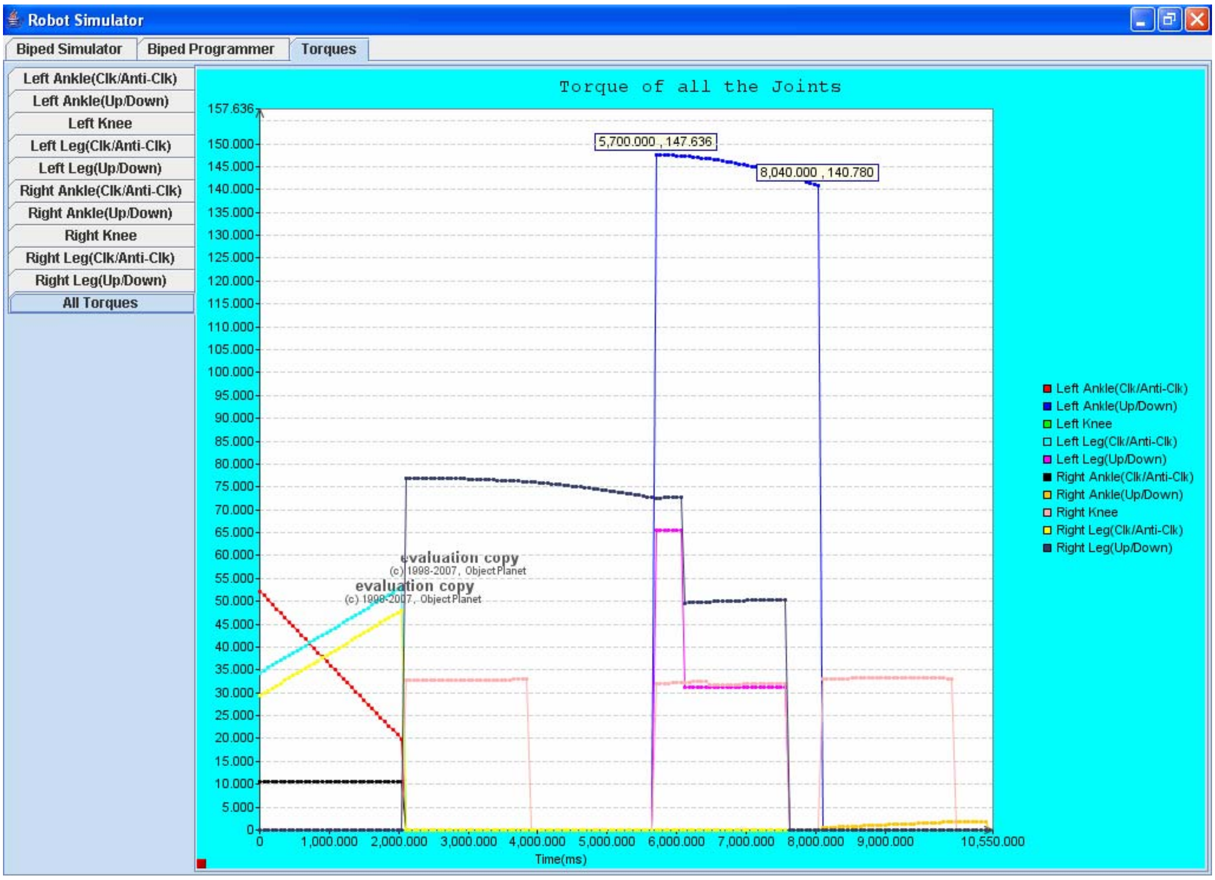

Figure 6.10 Torque required by all the joints

7. SUMMARY AND CONCLUSION

7.1 SUMMARY

Robotics is very broad field and even after decades legged robots are not yet capable of taking decisions on their own. Legged robots are able to walk, run and can perform some primitive tasks like pick and place. Developing a two-legged robot costs very high so, before starting the development it is good idea to simulate the robot and get various parameter values like torque requirement of each joint. Simulator helps in reducing the development cost and also helps in manufacturing process by optimizing various parameters of the system. In this project a two-legged robot simulator, Application specific compiler and torque analyzer has been developed. Simulator simulates various movements of two-legged robot. Simulator has very rich GUI with buttons to control each joint of the robot. Application specific compiler is a special program developed to convert the very high level language code to very low level (assembly) language code for target microprocessor or microcontroller. In this project ASC for two-legged robot has been developed which generates assembly language code for target microcontroller AT89S52. Along with simulator and ASC another module called torque analyzer has been developed to analyze the torque requirement of each joint of the robot while performing given task. Torque analyzer shows the torque data in form of graph for each joint.

On the basis of simulation data it is confirmed that the design of the robot is proper and robot should be able to walk. As per the simulation data and CAD model robot is being developed.

7.2 CONCLUSION

For two-legged robots torque requirements are very high compared to multi legged robots and wheeled robots. Balancing is one of the most difficult tasks for two-legged robots. In the Table 7.1 maximum torque required by robot without including dynamics of the robot are given for taking one step forward from rest condition. From the table we can conclude that knee joint requires highest torque.

Table 7.1 Maximum torque required by joints of robot

Joint Name	Minimum Torque Kg-cm
Left Ankle(Up/Down)	145
Left Ankle(Clock/Anti-clock)	43
Left Knee	0
Left Pelvis(Up/Down)	70
Left Pelvis(Clock/Anti-clock)	53
Right Ankle(Up/Down)	2
Right Ankle(Clock/Anti-clock)	43
Right Knee	34
Right Pelvis(Up/Down)	77
Right Pelvis(Clock/Anti-clock)	46

The selected stepper motor and gearbox combination is able to provide maximum torque of 188 Kg-cm which higher than torque required by any joints. So it can be concluded that robot should be able to walk.

7.3 FUTURE WORK

In this project a two legged robot has been simulated and robot's physical model is being developed. Once the physical model has been developed its can be enhanced to include dynamics of the robot by using feedback of the feet soles by putting pressure sensors at the sole. By reading the values of pressure sensors it can be derived that on which direction robot will fall down when its not balanced and appropriate actions can immediately be take to avoid fall down. For controlling all the stepper motors 8-bit microcontroller AT89S52 has been used which does not has capability to perform floating point calculations, so on top of that a Digital Signal Processor (DSP) like ARM can be used which can use data acquired from pressure sensors mounted on the bottom part of feet soles to calculate the forces acting on the each foot, and if robot is found to be unbalanced it can generate necessary movements of each joint to balance the robot. Once such high power DSP is used some algorithms using artificial intelligence or neural network can also be evaluated on the robot.

On the other side mechanical structure can be enhanced by putting other parts of the body like hands and head to convert the two-legged robot to humanoid robot.

REFERENCES

[1] Satoru Shirata, Atsushi Konno, and Masaru Uchiyama, "Design and Development of a Light-Weight Biped Humanoid Robot Saika-4", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, September 28 - October 2, 2004, pp. 148-153.

[2] Rainer Bischoff and Tamhant Jain,"Natural Communication and Interaction with Humanoid Robots", *Second International Symposium on Humanoid Robots*, Tokyo, Japan, October 1999.

[3] Qiang Huang, Yoshihiko Nakamura, Hirohiko Arai, and Kazuo Tanie, "Development of a Biped Humanoid Simulator", *Proceedings of the lEEE/RSJ International Conference on intelligent Robots and Systems* , Takamatsu, Japan, Vol. 3, 2000, pp. 1936-1942.

[4] Riadh Zaier, "Motion Generation of Humanoid Robot based on Polynomials Generated by Recurrent Neural Network", *Proceedings of the First Asia International Symposium on Mechatronics*, Xi'an, China, September 27-30, 2004.

[5] Tetsuro Kitazoe, "Unsupervised Learning of Two Legged Robot", *IEEE International Workshop on Robot and Human Communication*, Nagoya, Japan , 18-20 Jul, 1994, pp. 351-355.

[6] Andre Senior, and Sabri Tosunoglu, "Design of a Biped Robot", *Florida Conference on Recent Advances in Robotics*, Miami, Florida, May 25-26, 2006.

[7] Kazuo Tanie, "Humanoid Robot and its Application Possibility", *IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems*, 30 July-1 Aug, 2003, pp. 213 – 214.

[8] "ASIMO Technical Information", American Honda Motor Co. Inc. Corporate Affairs and Communications, January 2003. ,http://asimo.honda.com/downloads/pdf/asimo-technical-information.pdf

[9] "History of ASIMO", http://asimo.honda.com/AsimoHistory.aspx

[10] "Stepper Motor Specifications", Precision Motors, http://www.pmpl.co.in/d4842bi.pdf

[11] "Gearbox Specifications", Mech-Tex Manufacturing Co., http://www.mechtex.com/PDF/gb4.pdf

USEFUL WEBSITES

[1] "Circuit Diagram of the ISP Programmer Interface",
http://www.kmitl.ac.th/~kswichit%20/IspPgm30a/ISPPgm30a.html

[2]"8051 Development Tools", http://bit.kuas.edu.tw/~8051

[3]"Datasheet of Allegro A-3892 Bipolar stepper motor controller",
http://www.allegromicro.com/en/Products/Part_Numbers/3982/3982.pdf