TABLE OF CONTENTS

ACKNOWLEDGEMENT

Every project work has an imprint of many people and hereby we take this excellent opportunity to acknowledge all the help, guidance and support that we have received for completion of this project.

With extreme sincerity and deep sense of appreciation, we express thanks to our guide and mentor Prof. Prasad S. Joshi (Head of Electronics Department), for his co-operation, suggestion and guidance. We appreciate his role of mentoring us for this project and thank him whole heartedly for all the support he has given us throughout this year.

We also express our gratitude to workshop in-charges Deepak Telwane, Mangesh Devrukhkar, Vijay Chavan and lab in-charges Suresh Darde, Ravi Khapne, Karuna Bhosle, Mayuresh Vartak, Kinnari Salvi for the help and co-operation offered by them along every stage of the project. We would also like to thank all our friends and colleagues who co-operated with us throughout the period with their innovative and new ideas which were also implemented in our project.

Lastly, we would like to thank our parents for extending their support towards the completion of this project and for the unconditional encouragement.

.

ABSTRACT

This thesis extends the theory of passive dynamic bipedal robots to the case of bipedal robots which walk by means of simple forms of actuation. Beginning with the study of the simplest possible mechanisms which could be described as actuated walking robots we build up to more advanced walking machines developing the tools along the way which are required to describe and control the possible gait behavior of a super articulated torso driven biped robot. A design for a torso driven biped robot is proposed which involves the use of just a single motor to enhance balance and improve efficiency of the walking motion.

The actuation of the controlled joint is planned such that the motion of the uncontrolled degrees of freedom sustains the walking behavior. A control strategy is devised which stabilizes the equilibrium walking behavior of the torso driven biped.

Many researchers have been encouraged to investigate the design, posture and stability of biped robots in order to replicate the anthropoid gait. This paper addresses the design and development of a bipedal robot. It presents a combination of the design considerations and simplicity of design to provide a test bed for autonomous biped robots. Overall, a low cost, open system biped robot is the underlying objective on which new gait algorithms and controllers will be developed to further the research in the field of humanoid robots.

A few of the intended features of this project will include the ability to remotely control the robot from a computer, both cabled and wirelessly, basic obstacle avoiding mechanism, performing a pre-learnt motion and safety system to avoid falling. The structure is capable of forward, backward and sideways motion. It has provisions for attaching motors for more degrees of freedom, as well as having a multi-purpose utility arm for industrial applications. Apart from anthropoid gait motion for walking, the hardware is capable of various other motions like kicking and one-leg balancing. It is capable of various other motion patterns which it can be programmed to perform.

Chapter 1

Overview

1.1. Introduction :

Past three decades witnessed a growing interest in biped walking robots because of their possible advantageous use in the human environments. Many research labs and Universities started to work on biped and humanoid projects, in order to develop their own robots. Nowadays we can see some companies' robots that are commercialized and put on to market, like Honda, Sony and Fujitsu. Probably the ultimate goal when developing these machines is to replace the humans with robots, that are able to work with or instead of humans having some of the human skills, in the factories, malls, restaurants, our homes and even in the soccer fields. There are numerous places that humanoids might be used to ease our life. This will be one of the biggest revolutions after the step on moon that technology will accomplish in the human history. Biped robot research is one of the bricks to realize the humanoid technology.

Despite their functionality, their control is challenging because of their many DOFs and nonlinearities in their dynamics. Offline trajectory generation and the so-called open loop walking is one of the control approaches in the literature for bipeds. There are various problems involved in this approach, the most pronounced one being the difficulty in tuning the gait parameters.

1.2. Project Specifications:

- Height of our Biped Robot: 30 cm
- Weight of our Biped Robot: 1.2kg excluding Lion Batteries & GPB
- Total 8 Servomotors are used to give 16 DOF in all
- 4 servomotors each used in one leg. Each servo motor gives max 2 DOF.
- We have used aluminum as our material which is very light weighted
- We used Atmega16 Micro controller to drive the motors.
- Project Concept based on Gait Algorithm & Walking Methodology of Human Legs.
- Total Current our Biped Robot takes is 2A maximum & Total Voltage it takes is 5V.

- The Step Size of our Biped Robot is 7cm for half a step completion of one leg.
- The Speed of our Biped Robot currently is 0.7cm/sec for entire one step of walking pattern.

1.3. Project Concept :

In recent years some very good work has been done with passive dynamic walking machines - machines that utilize gravity, and the physical dynamics of legs with passive joints to produce very impressive walking gaits with very minimal control and excellent mechanical efficiency. We would like to build a machine which extends the key elements of the passive dynamic walking machine with powered joints which are able to either transition into a passive mode, or actively sense torque loading and react to minimize load in order to simulate a passive joint. A biped of this sort, combined with intelligent control could be capable of functioning in human environments with a reasonable level of efficiency.

We propose a humanoid robot of child size (perhaps 30cm tall) big enough to reach typical household items, created using extremely lightweight materials and using intelligent networked servo control and a micro controller.

Fig. a. Actual Picture Of Robot

1.4. Literature Survey :

To compare and contrast existing literature with the contents of this thesis, a few of the more dominant trends in bipedal locomotion will now be examined. This survey is not intended to be exhaustive, but rather to provide a representative cross section showing both the breadth and the depth of ongoing projects in bipedal locomotion, emphasizing a correlation between robot morphologies and control tools.

Three classes of research in bipedal locomotion will be briefly reviewed: analytical approaches to locomotion, the ZMP (zero moment point) criterion, and passive dynamic walkers. Boundaries between these groups are often blurred, but they nevertheless represent a few of the dominant approaches driving research in robotic locomotion. The first group, the camp of formal stability theory, focuses on the use of rigorous mathematical methods in the procedures of gait design, controller derivation, and stability proof. Analytically proving the stability of dynamic walking and running Motions can be relatively difficult, stemming from the multi-phase, hybrid nature of the problem and the mathematical precision involved in the formulation of relevant theorems and proofs.

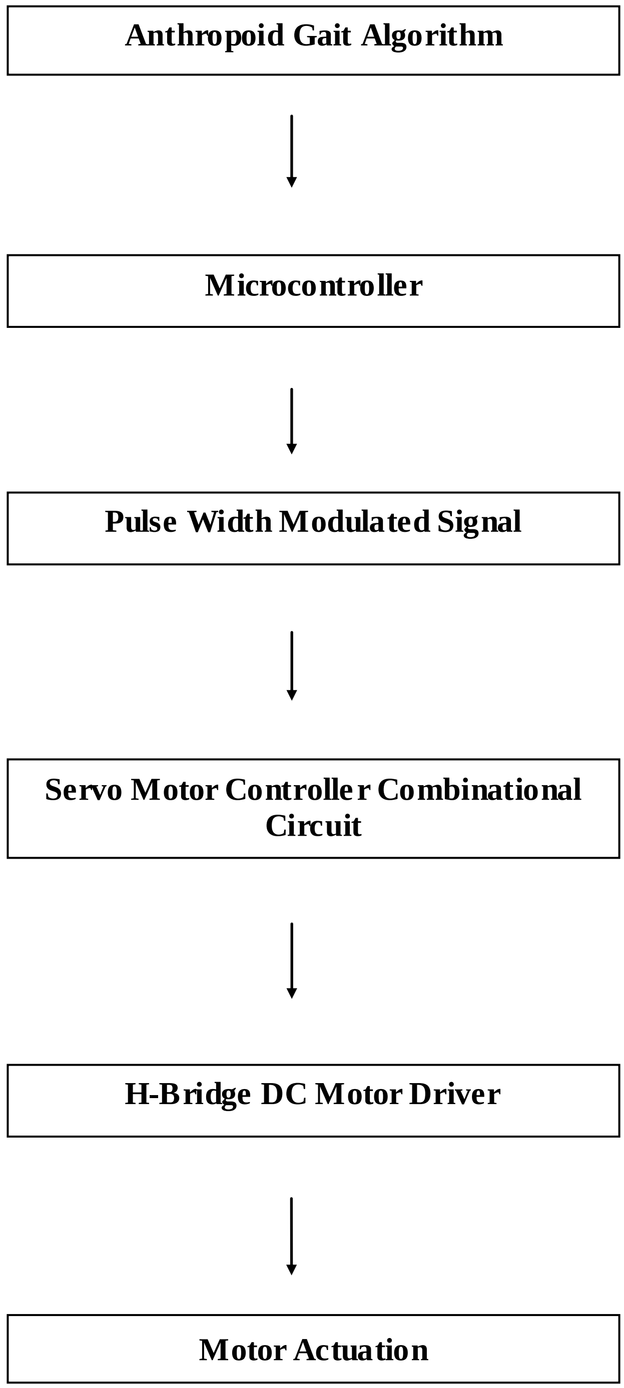
For this reason many researchers choose to study static or quasi-static walking using the ZMP criterion, forming a second major trend in bipedal locomotion research. Here trajectory tracking controllers are coupled with online gait modification schemes to achieve quasi-static walking gaits that keep the robot upright, but often at the cost of producing a slow, crouching motion. A third group of researchers follows in the footsteps of Tad McGee, studying robots that require no actuation other than gravity to walk stably down a slope. With no active control whatsoever, passive dynamic walkers produce elegant, human-like gaits with maximal efficiency, but with minimal versatility of locomotion behavior.

The following sections examine these three methodologies in greater detail, highlighting research philosophies, common tools, and explaining a few of the notable experimental successes of each group. Because the work of this thesis is so closely tied to the context of hybrid zero dynamics and provable stability, more emphasis will be placed on reviewing this area than the other two.

Chapter 2

System Block Diagram

2.1. System Block Diagram:



2.2. Servo Motors:

Servomotors are also called control motors as they are involved in control system. The motor gets its name from the word **servo mechanism**.

These motors are used in feedback control systems as output actuator. A feedback control system is a control system which tends to maintain a prescribed relationship between a controlled quantity and a reference quantity by comparing their functions and using the difference as a means of control. The essential contents of such a control system are: error detecting device, an amplifier, and error correcting device / controller. Each element serves a functional purpose in matching the controlled or regulated quantity to the reference quantity. The error detecting device determines when the regulated quantity is different from the reference quantity. It then sends an error signal to the amplifier which supplies power to error correcting device.

There are two fundamental characteristics of any servo motor. These are:

- The motor output torque is proportional to the voltage applied to it (i.e. control voltage developed by amplifier in response to an error signal).
- The instantaneous polarity of control voltage governs the direction of torque developed by servomotors.

Working principle of DC servomotor:

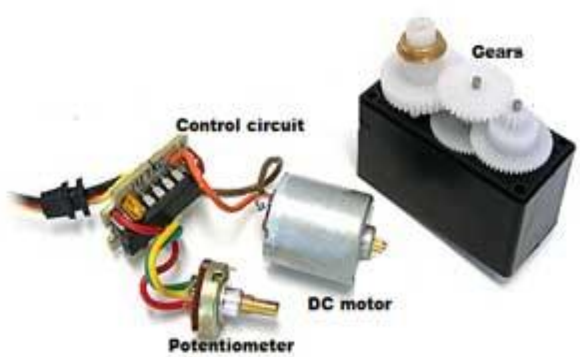


Fig. a. Servo Motor Circuitry

In DC operation, servomotors are usually responds to error signal abruptly and accelerate the load quickly. A DC servo motor is actually an assembly of four separate components, namely:

- **a DC motor**
- **a gear assembly**
- **a position-sensing device**
- **a control circuit.**

In case of field controlled dc motor, the field is excited by the amplified error signal mentioned earlier. The armature winding is energized from a constant current source. Torque developed is proportional to field current up to saturation level. This method is applied in small servomotors. It has longer time constant owing to highly inductive field circuit so dynamic response is slower than armature controlled dc motor.

But in armature controlled dc motor, the motor armature is energized by amplified error signal and field is supplied from a constant current source. High field flux density also increases torque sensitivity of motor (torque proportional to $\varphi\ I_a$). Here dynamic response is faster because it has shorter time constant of the resistive circuit.

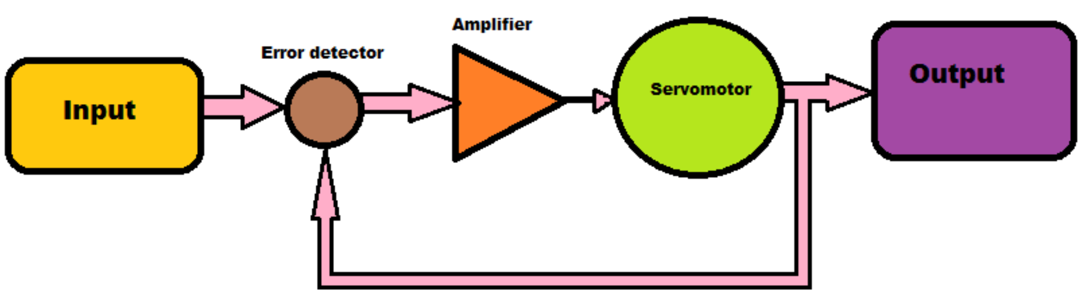


Fig. b. Flow Chart Showing Feed backed Servo Control

As you can see in the diagram, the output signal is sampled and thereafter the actuating device rectifies the errors. After that, the servomotor gear assembly is used for controlling the position.

How is the servo controlled?

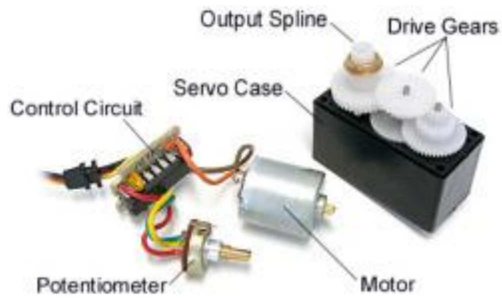
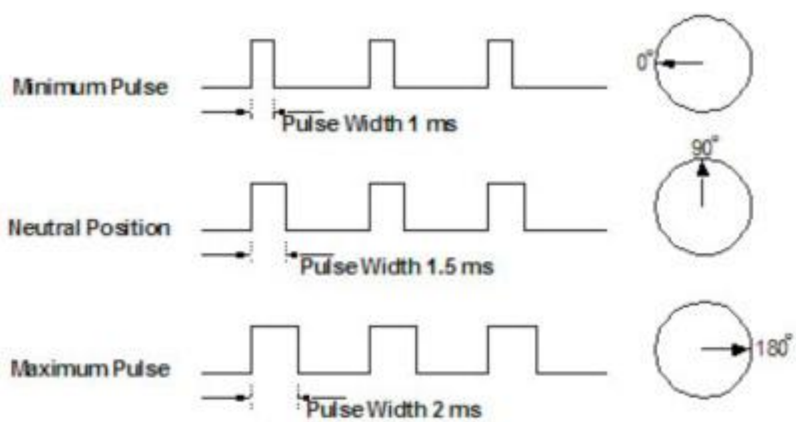


Fig. c. Servo Motor Circuitry

Servos are controlled by sending an electrical pulse of variable width, or **pulse width modulation** (PWM), through the control wire. There is a minimum pulse, a maximum pulse, and a repetition rate. Servo motors can usually only turn 90 degrees in either direction for a total of 180 degree movement. The motor's neutral position is defined as the position where the servo has the same amount of potential rotation in the both the clockwise or counter-clockwise direction. The PWM sent to the motor determines position of the shaft, and based on the duration of the pulse sent via the control wire; the rotor will turn to the desired position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns. For example, a 1.5ms pulse will make the motor turn to the 90-degree position. Shorter than 1.5ms moves it to 0 degrees, and any longer than 1.5ms will turn the servo to 180 degrees, as diagramed below



Variable Pulse width control servo position

When these servos are commanded to move, they will move to the position and hold that position. If an external force pushes against the servo while the servo is holding a position, the servo will resist from moving out of that position. The maximum amount of force the servo can exert is called the **torque rating** of the servo. Servos will not hold their position forever though; the position pulse must be repeated to instruct the servo to stay in position.

2.3. Atmega16 Microcontroller:

Features

• High-performance, Low-power Atmel® AVR® 8-bit Microcontroller

• Advanced RISC Architecture

– 131 Powerful Instructions – Most Single-clock Cycle Execution
– 32 x 8 General Purpose Working Registers
– Fully Static Operation
– Up to 16 MIPS Throughput at 16 MHz
– On-chip 2-cycle Multiplier

• High Endurance Non-volatile Memory segments

– 16 Kbytes of In-System Self-programmable Flash program memory
– 512 Bytes EEPROM
– 1 Kbyte Internal SRAM
– Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
– Data retention: 20 years at 85°C/100 years at 25°C(1)
– Optional Boot Code Section with Independent Lock Bits
In-System Programming by On-chip Boot Program
True Read-While-Write Operation
– Programming Lock for Software Security

• JTAG (IEEE std. 1149.1 Compliant) Interface

– Boundary-scan Capabilities According to the JTAG Standard

– Extensive On-chip Debug Support

– Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface

• Peripheral Features

– Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes

– One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode

– Real Time Counter with Separate Oscillator

– Four PWM Channels

– 8-channel, 10-bit ADC

8 Single-ended Channels

7 Differential Channels in TQFP Package Only

2 Differential Channels with Programmable Gain at 1x, 10x, or 200x

– Byte-oriented Two-wire Serial Interface

– Programmable Serial USART

– Master/Slave SPI Serial Interface

– Programmable Watchdog Timer with Separate On-chip Oscillator

– On-chip Analog Comparator

• Special Microcontroller Features

– Power-on Reset and Programmable Brown-out Detection

– Internal Calibrated RC Oscillator

– External and Internal Interrupt Sources

– Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby

• I/O and Packages

– 32 Programmable I/O Lines
– 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF

• Operating Voltages

– 2.7V - 5.5V for ATmega16L
– 4.5V - 5.5V for ATmega16

• Speed Grades

– 0 - 8 MHz for ATmega16L
– 0 - 16 MHz for ATmega16

• Power Consumption @ 1 MHz, 3V, and 25°C for ATmega16L

– Active: 1.1 mA
– Idle Mode: 0.35 mA
– Power-down Mode: < 1 µA

Chapter 3

System Implementation

3.1 Basics:

3.1.1 Why Legs?

- Potentially less weight.
- Better handling of rough terrains.
    - Only about a half of the world's land mass is accessible by current man-built vehicles.
- Do less damage to terrains (environmentally conscious)
- More energy-efficient.
- More maneuverability.
    - Use of isolated footholds that optimize support and traction.
    - (I.e. ladder)
- Active suspension
    - Decouples the path of body from the path of feet.

3.1.2 Why Bipeds?

- Why 2 legs? 4 or 6 legs give more stability, don't they?
    - A biped robot body can be made shorter along the walking direction and can turn around in small areas
    - Light weight
    - More efficient due to less number of actuators needed
- Everything around us is built to be comfortable for use by human form
- Social interaction with robots and our perception (HRI perspective)
    - Form will become as important as functionality in the future
- Our instinctive desire to create a replica of ourselves (maybe?)

3.1.3 Joints in a Leg:

- At least 2 DOF (degrees of freedom) needed to move a leg
    - A lift motion + a swing motion
- A human leg has 30 DOF

  - Hip joint = 3 DOF
  - Knee joint = 1 ~ 2 DOF (almost a hinge)
  - Ankle joint = 1 DOF (hinge)
  - 24 DOF for the foot!
- In many cases, a robot leg has 3 DOF
  - Control becomes increasingly complex with added DOF
- With 4 DOF, ankle joint can be added
- Reasonably walking biped robots have been built with as few as 4 DOF.

3.1.4 Stability:

- Stability means the capability to maintain the body posture given the control patterns
- Statically stable walking implies that the posture can be achieved even if the legs are frozen / the motion is stopped at any time, without loss of stability
- Dynamic stability implies that stability can only be achieved through active control of the leg motion
- Statically stable systems can be controlled using kinematical models.
- Dynamic walking requires use of dynamical models.

3.1.5 Gaits:

- Gaits determine the sequence of configurations of the legs
  - A sequence of lift and release events of individual legs
- Gaits can be divided into 2 main classes
  - Periodic gaits → repeat the same sequence of movements
  - Non-periodic or free gaits → no periodicity in the control and could be controlled by the layout of environment
- The number of possible events N for a walking machine with k legs is:
  - $N = (2k - 1)!$
- For a biped robot (k = 2), there are 3! = 6 possible events
  - Lift left leg, lift right leg, release left leg, release right leg, lift both legs, and release both legs.

3.1.6 Gait and Stability:

- People, and humanoid robots, are not statically stable
- Standing up and walking appear effortless to us, but we are actually using **active control of our balance**
    - We use muscles and tendons
    - Robots use motors
- In order to remain stable, the robot's Center of Gravity must fall under its polygon of support
    - The polygon is basically the projection between all of its support points onto the surface
    - In a biped robot, the polygon is really a line
        - The center of gravity cannot be aligned in a stable way with a point on that line to keep the robot upright.

3.2 Control of a Walking Robot:

- 3 things that control must consider for walking:
    - Gait: the sequence of leg movements
    - Foot placement
    - Body movement for supporting legs
- Leg control patterns
    - Legs have 2 major states:
        - Stance: On the ground
        - Fly: In the air moving to a new position
    - Fly state has 3 major components:
        - Lift phase: leaving the ground
        - Transfer: moving to a new position
        - Landing: smooth placement on the ground
- More DOF for the legs means
    - Smoother movement, but
    - Increasingly complex controls

Biped Walking = Rolling

- Rolling is quite efficient
- Biped walking is similar to rolling a polygon
    - Polygon side length = step length
    - As step length gets shorter, more like rolling a circle

Fig. a. Polygon Law Of Motion

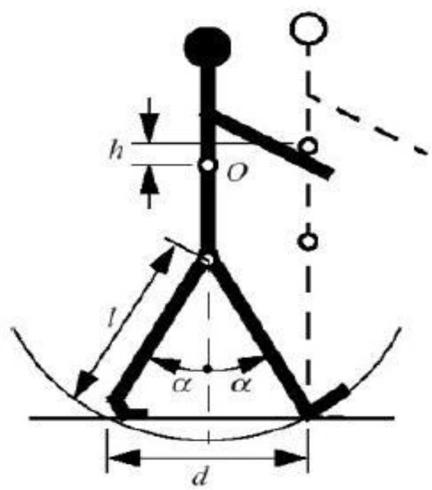
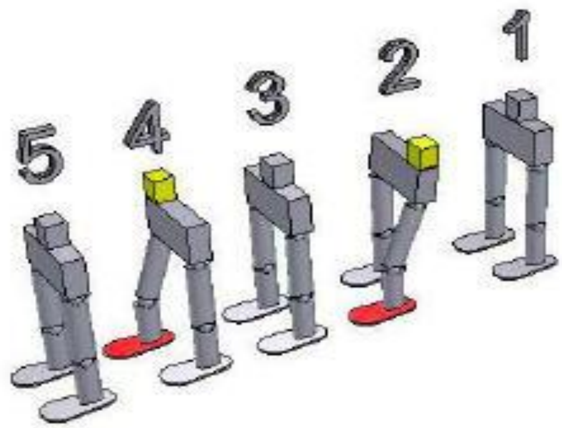


Fig. b. Walking Trajectory

3.3 Implementation of walking state methodology:

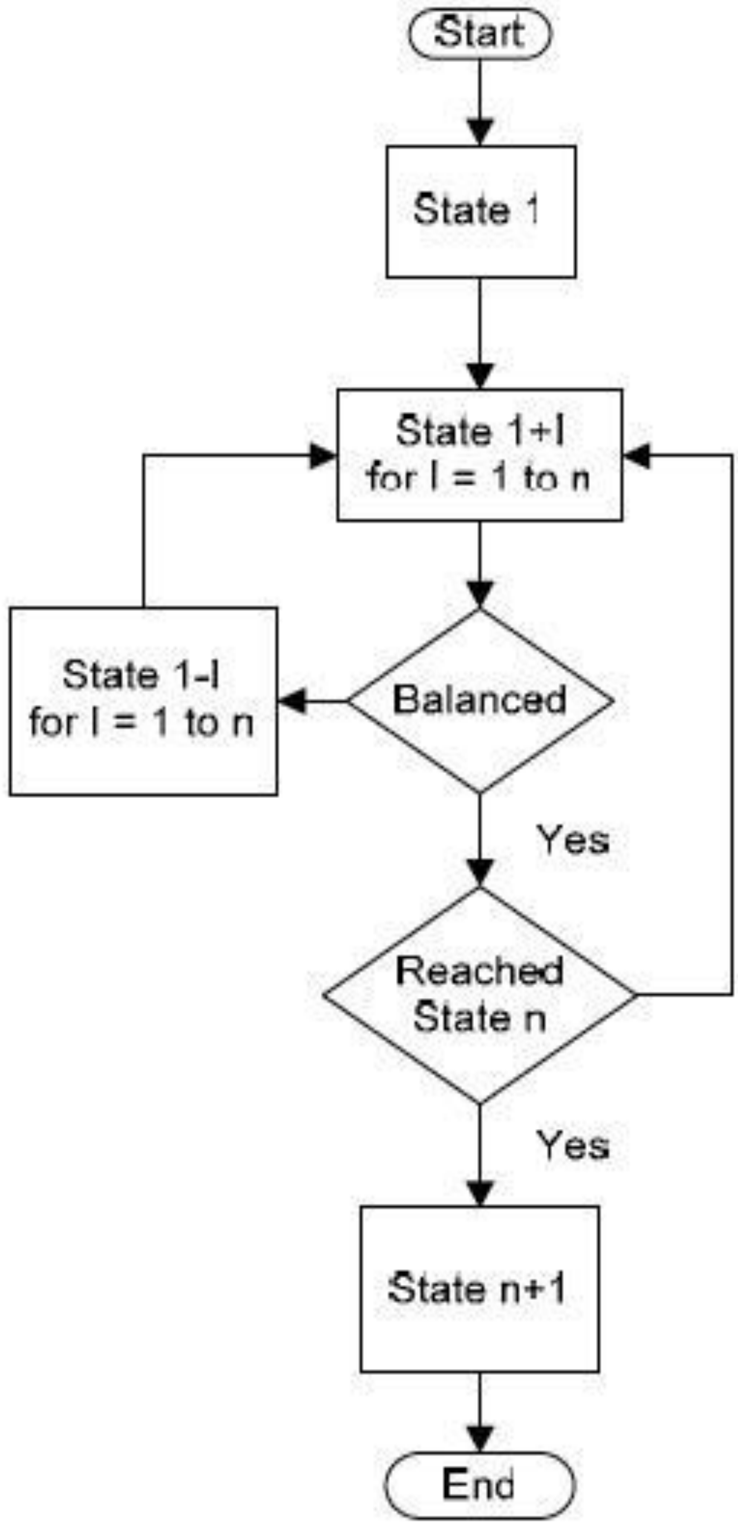
- Walking algorithm for biped robots often derived from classical control theory
    - Uses a reference trajectory for the robot to follow
    - Reference trajectories can rarely be defined to work in the real world
        - Irregular terrains and encountering different obstacles, etc.

- Uses static balance poses to define points of tending to balance during a gait
- The point that a biped robot tends to balance is called a state
- The walking states are chosen as the maximum and minimum tending to balance stance equilibrium positions where little or no torque needs to be applied to maintain the state.
- Marching gait example
- 5 states where the robots tends to either balance or tend to topple
- The center of gravity tends to shift as shown by the cube on top of the robot



Hypothetical marching gait example

- While advancing to new states during the actual walking locomotion, an autonomous robot's software should ideally extrapolate the gait from balanced state to the next.
- In states 2 and 4, we can interpret the robot as tending to an out of balance point. If the leg that is bent continues in the same direction, then the robot will topple.
- The control algorithm should not counter the tending to topple position by bending the other knee on the other leg or shifting the original leg back to its initial position.
- The control algorithm should continue with the balance control state, expecting that to prevent a fall, the robot has to counter balance by shifting the center of gravity to either the neutral position or to the next tending to out of balance point on the opposite side.



General autonomous robot gait algorithm

Chapter 4

Locomotion Algorithm

4.1 Significance Of Center Of Gravity:

- It is the point at which the entire mass of an object is assumed to be acting from.
- For any object to be stable, its center of gravity, projected towards the base, must lie within the base area.
- This is extremely important for a dynamic articulating robot, as with every motion its center of gravity changes.
- The key to the bipedal robot's stability is keeping its center of gravity stable at all times.
- The most important consideration in walking is transferring weight of the complete body on one leg and subsequently the other, center of gravity and its manipulation and balance form the essence of the walking algorithm

4.2 Significance Of Structural Symmetry:

- The center of gravity of any object can be known by suspending it from two different angles and checking where the lines normal to the ground, taken from the point of suspension, intersect.
- Manually performing this exercise for every position of the bipedal robot is tedious and complex.
- Calculations needed to be performed for an awkward center of gravity, which keeps shifting with every movement of the motors, is also very complex.
- A way to circumvent this problem is by using a symmetrical design. While the two legs are perfectly symmetric, the top and bottom halves of the legs should also be equal. By doing this, the center of gravity comes at half the height of the robot.
- It is possible to achieve this symmetry as the number of motors is even in number (4) in each leg, and there are 2 motors in each leg for vertical and transverse motions.
- For perfect symmetry, we keep the motors performing transverse motion at the top and bottom, and the motors performing front-back motions in the middle, as shown in the structural diagram.
- This structural symmetry has another significant advantage – it makes calculation for the height of the robot and the torque required by the motors much easier.

4.3 Text Notifications:

- **Motor notations:** For sake of simplicity and uniformity, we shall refer to the motors in this bipedal robot throughout this text as follows:
  - *Hip-transverse:* This is the first motor from top. It performs the motion of moving a single leg sideways.
  - *Hip:* This is the second motor from top. It performs front-back motion of lifting and placing the leg.
  - *Knee:* This is the third motor from top. It performs front-back motion of lifting and placing the leg, in antagonism to the hip motor. It also performs functions of the foot and ankle (front-back) motions, due to which it also bends backwards unlike a human knee joint.
  - *Ankle:* This is the bottom motor. It performs the transverse motion used to transfer the weight of the robot on a particular leg.
- Due to structural symmetry, these motors are present in both legs, and will be preceded by "left" or "right" to specify the leg henceforth.
- **Direction note:** "Left" and "Right" are relative as to whether we are looking at the robot, or with it. This text assumes that we are looking with the robot. As an illustration, if the robot is moving towards us the notifications will be as follows:

```
           —
          | |
R         | |         L
         --  --
```

- The initial position of the robot is assumed to be as shown in the structural representation. The forward walking algorithm initializes the motors to that position and proceeds.

4.4 Forward Walking Gait Algorithm:

- For continuous walking, this algorithm runs in an infinite loop. However, before starting the loop, certain initial steps need to be taken once. Each step will specify the motors involved during that step. The other motors hold their state. All motors involved in a step move simultaneously.

- The walking algorithm implemented moves the robot's left leg forward first, and the algorithm here too is described similarly. However, if need be the right leg may be moved first, keeping the logic same.

- *Initial Step 1:*

    - We move the transverse motion motors such that the center of gravity shifts and rests completely on the right leg.

    - For stability, it is important to keep the feet of the robot flat to the ground. Thus, in this step, all 4 transverse motion motors are moved by the same angle.

    - This angle is specific to the structural design. It that angle, by which if the robot is tilted on one side, the center of gravity comes to rest on the right leg. Due to symmetry, this angle will remain same to transfer the weight on the left leg too.

    - On completion of this step, the structure would look like a parallelogram, if the initial position looked like a rectangle.

    - Motors involved: Left & Right Hip-transverse and Left & Right Ankle motors

- *Initial Step 2:*

    - Now, as the center of gravity has been shifted to the right leg, the entire weight of the robot is on the right leg.

    - However, at this stage both legs are on the ground. In order to lift the left leg a bit from the ground, it is moved away from the body.

- This is done by first moving the Right Ankle motor to move the center of gravity further to the right, since moving the left leg away would result in the center of gravity moving towards the left.
- Subsequently, the left leg is moved away by using the Left Hip-Transverse motor. For the leg to be parallel to the ground, the Left Ankle motor is also moved by the same angle.
- Motors involved: Right Ankle and Left Hip-Transverse & Ankle motors

- *Loop Step 1:*
  - Now we lift the robot's Left leg. This is done my moving the Left Hip motor such that the left leg moves forward and the Left Knee moves simultaneously in perfect antagonism to keep the foot parallel to the ground.
  - Motors involved: Left Hip & Knee motors
- *Loop Step 2:*
  - For human walking, if we lift our left leg, we use our right foot and hip motions to place the left leg back on the ground, and take a "step."
  - In absence of these, the robot uses the Right Hip & Knee motors to place the left leg back on the ground.
  - Just as the Left leg was bent forward to lift it from the ground, the Right leg is bent backwards.
  - This moves the "body" of the robot forward and brings the left leg close to the ground.
  - However, the left leg does not touch the ground yet, as it is away from the body.
  - Motors involved: Right Hip & Knee motors
- *Loop Step 3:*

- We now need to place the left leg on the ground.
- Thus, it is moved closer to the body using the Left Hip-Transverse motor and the foot is kept parallel to the ground using the Left Ankle motor simultaneously in perfect antagonism with the Left Hip-Transverse motor.
- Subsequently, the Right Ankle motor, which was moved further to keep the center of gravity comfortably on the right leg is moved back.
- At the end of this step, we again form a parallelogram-like state similar to Initial Step 1, with the difference that the Left leg is bent forward, and the Right leg is bent backwards.
- While the Left leg is now on the ground, the weight is still on the Right leg due to the tilt.
- Motors involved: Left Hip-Transverse & Ankle and Right Ankle motors

- *Loop Step 4:*
  - At this stage, we have completed half a step. To complete the step, we need to move the Right leg forward.
  - For this, like before, we need to transfer the center of gravity to the left leg.
  - Thus, like previously, we move the 4 transverse motors such that the center of gravity moves towards the left.
  - At the start of this step, the weight is on the Right leg since the 4 transverse motors are at an angle on one side of their initial positions. To transfer weight on the Left leg, we move the motors simultaneously such that they are, by the same amount, on the other side of their initial values.
  - At the end of this step, we again get a parallelogram state but this time tilting on the left side.
  - Motors involved: Left & Right Hip-transverse and Left & Right Ankle motors

- *Loop Step 5:*
    - At this stage, the Left leg is in front and the Right leg is behind. To move the Right leg front, we need to lift it.
    - This is done by straightening the Left leg. Since the weight is on the left leg, by straightening it the body of the robot moves forward and the Right leg is lifted off the ground.
    - Motors involved: Left Hip & Knee motors
- *Loop Step 6:*
    - Then the Right Hip & Knee motors then move simultaneously in perfect antagonism such that the Right Hip motor moves the Right leg forward.
    - To prevent the Right leg from touching the ground when it is to be moved forward, we move the Right leg away from the body, just as we had done with the Left leg.
    - Just like the previous case, to do this we need to first move the center of gravity comfortably in the left foot, and to do this we tilt the Left Ankle motor further.
    - The Right Ankle motor is used to keep the foot parallel to the ground at all times.
    - At the end of this step, the Right leg is lifted, and the weight is on the left leg. This is just like Initial Step 1, except that the weight is on the left leg and the right leg is moving forward.
    - Motors involved: Right Hip-Transverse, Hip, Knee and Ankle motors
- *Loop Step 7:*
    - This step is similar to Loop Step 2.

- Just as the Right leg was bent forward to lift it from the ground, the Left leg is bent backwards.
- This moves the "body" of the robot forward and brings the Right leg close to the ground.
- However, the Right leg does not touch the ground yet, as it is away from the body.
- Motors involved: Left Hip & Knee motors

- *Loop Step 8:*
  - This is similar to Loop Step 3.
  - We now need to place the Right leg on the ground.
  - Thus, it is moved closer to the body using the Right Hip-Transverse motor and the foot is kept parallel to the ground using the Right Ankle motor simultaneously in perfect antagonism with the Right Hip-Transverse motor.
  - Subsequently, the Left Ankle motor, which was moved further to keep the center of gravity comfortably on the Left leg is moved back.
  - At the end of this step, we again get a parallelogram state but this time tilting on the left side.
  - While the Right leg is now on the ground, the weight is still on the Left leg due to the tilt.
  - Motors involved: Right Hip-Transverse & Ankle and Left Ankle motors
- *Loop Step 9:*
  - We now need to transfer the center of gravity to the Right leg so that we can prepare a state from where the Left leg can again be moved forward.
  - Thus, like previously, we move the 4 transverse motors such that the center of gravity moves towards the left.

- At the start of this step, the weight is on the Left leg since the 4 transverse motors are at an angle on one side of their initial positions. To transfer weight on the Right leg, we move the motors simultaneously such that they are, by the same amount, on the other side of their initial values.
- At the end of this step, we again get a parallelogram state but this time tilting on the right side.
- Motors involved: Left & Right Hip-transverse and Left & Right Ankle motors

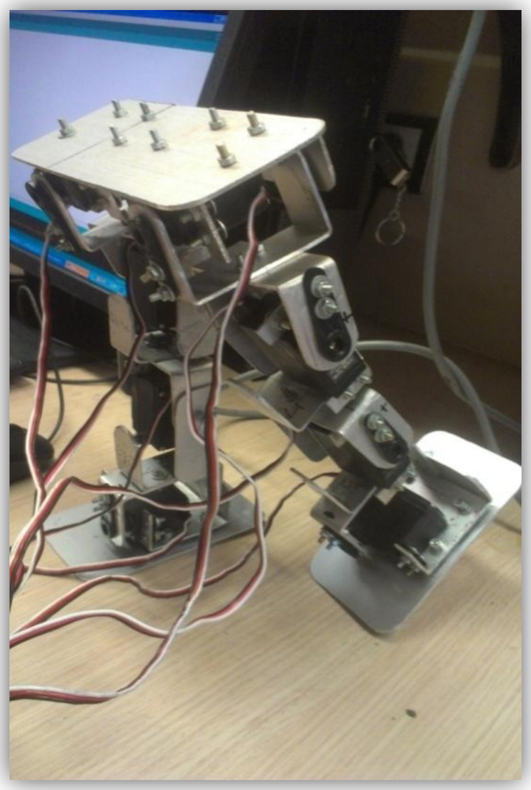
- *Loop Step 10:*
  - At this stage, the Right leg is in front and the Left leg is behind. To move the Left leg front, we need to lift it.
  - This is done by straightening the Right leg. Since the weight is on the Right leg, by straightening it the body of the robot moves forward and the Left leg is lifted off the ground.
  - Motors involved: Right Hip & Knee motors

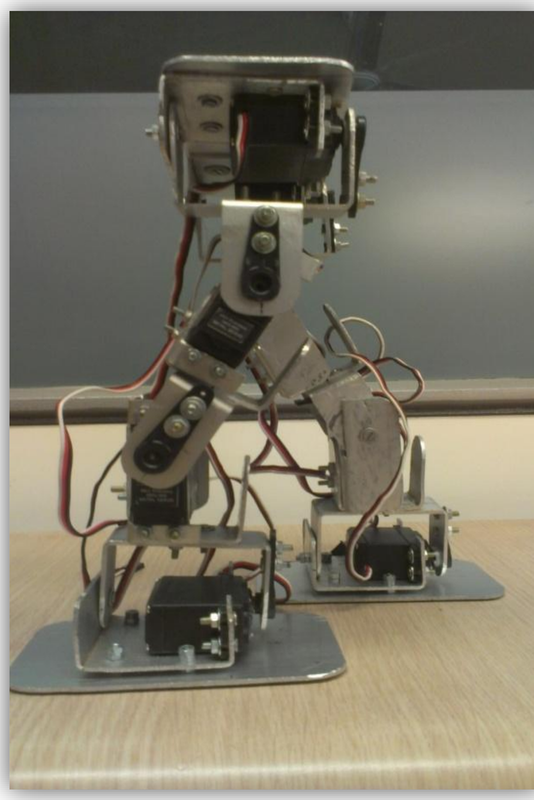
- *Loop Step 11:*
  - At the end of this step, we should be where we were after Initial Step 2 to loop the motion.
  - Since during the Initial Steps the legs were straight, we need this half-step to straighten the Left leg so that we can continue the loop seamlessly.
  - The Left Hip & Knee motors then move simultaneously in perfect antagonism such that the Left Hip motor moves the Left leg forward. However, we stop this step just as the Left leg straightens out so as to continue the loop.
  - To prevent the Left leg from touching the ground when it is to be moved forward, we move the Left leg away from the body, just as we had done with the Right leg.

- Just like the previous case, to do this we need to first move the center of gravity comfortably in the Right foot, and to do this we tilt the Right Ankle motor further.
- The Left Ankle motor is used to keep the foot parallel to the ground at all times.
- At the end of this step, the Left leg is straightened, and we reach the position we were at after Initial Step 2.
- Motors involved: Left Hip-Transverse, Hip, Knee and Ankle motors

- *Again the same steps are called sequentially from Loop Step 1.*

Kicking & Walking Positions:





Chapter 5

Hardware Design

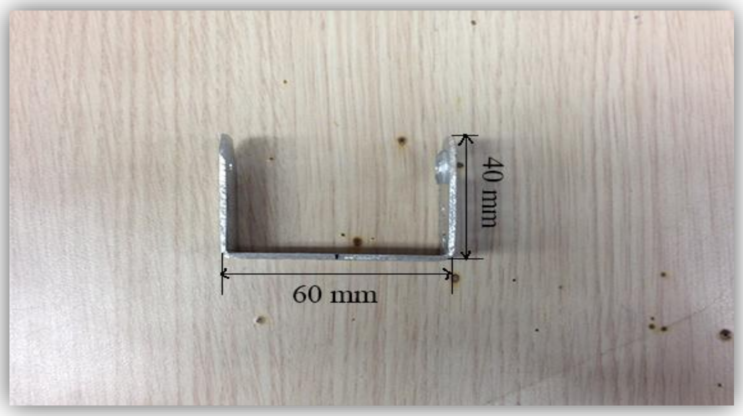


Fig. a. C- Clamp

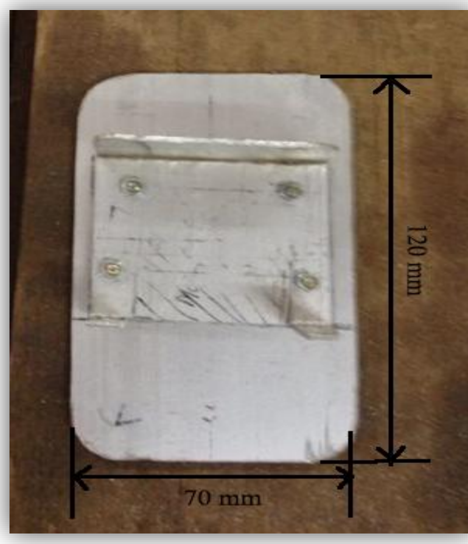


Fig. b. Foot Plate

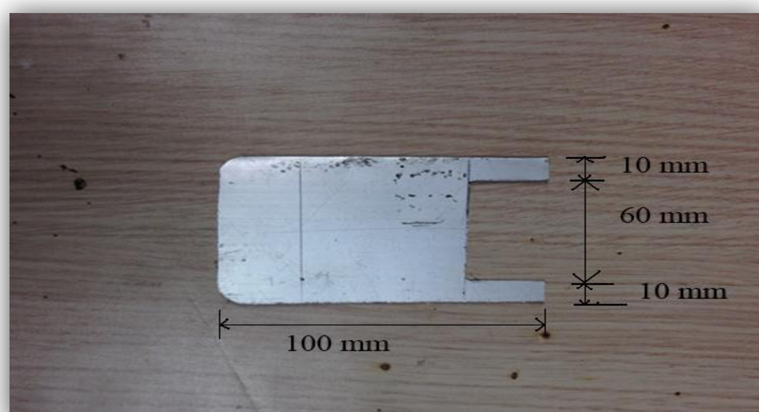


Fig. c. Multipurpose Clamp

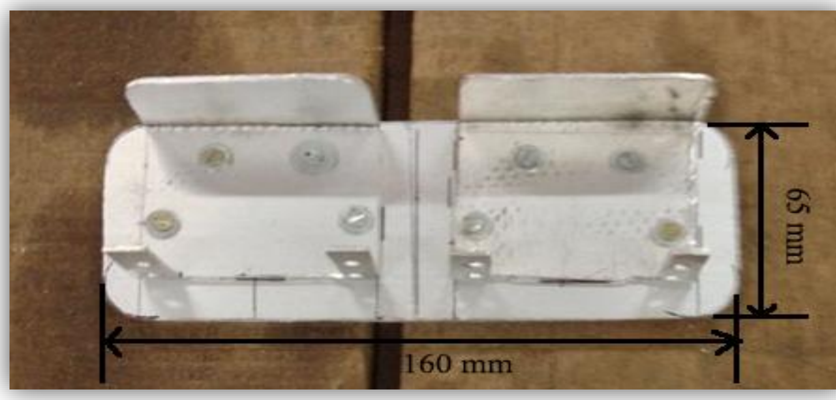


Fig. d. Hip Plate With Multipurpose Clamps

Fig. e. All Parts Required In The Making

BRAT Assembly Guide

To assemble the BRAT robot, construct a right leg (following the pictures on the right), and a left leg (by following the pictures on the left)! Keep in mind that the pictures marked "Right Leg" are the robot's right leg. After the legs are constructed, the images will no longer be split. For part sizes refer to the pictures shown earlier.

Image of completed robot.

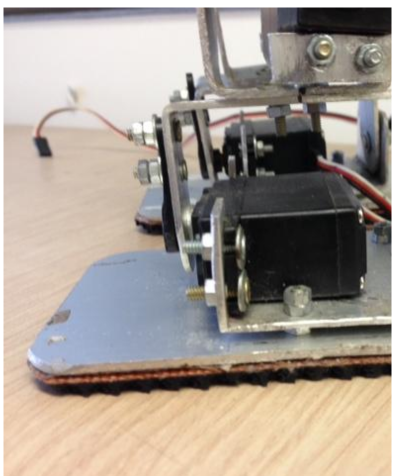


Figure 1. (Left Leg)

Step 1.
Attach a multi-purpose bracket to the foot as shown, using four 3mm screws and corresponding nuts.

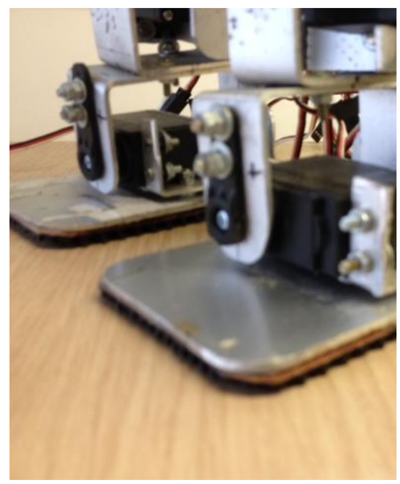
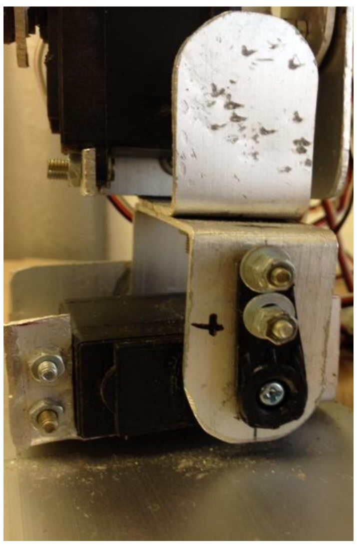


Figure 1. (Right Leg)

Step 2.
Attach the small "U" bracket to a short "C" bracket as shown, using two 3mm screws and 3mm nuts each.

Figure 2. (Left Leg)

Figure 2. (Right Leg)



Step 3.
Attach a multi-purpose bracket to the small "U" bracket as shown, using two 3mm screws and 3mm nuts each.

Figure 3. (Right Leg)

Figure 3. (Left Leg)

Step 4.
Attach the assembly from Step 3 to the multi-purpose bracket on the foot. See figure 4-1 for detailed information.

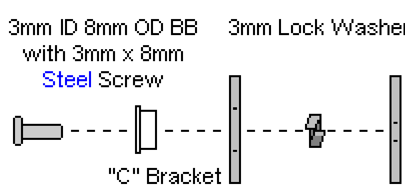


Figure 4-1.

Figure 4. (Left Leg)

Step 5.
Connect two short "C" brackets as shown, using two 2-56 x .250" screws and 2-56 nuts each.

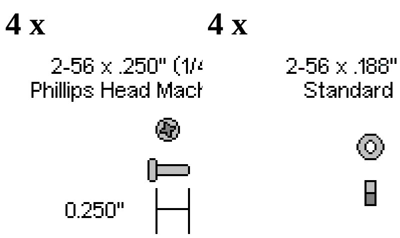


Figure 4. (Left Leg)

Figure 4. (Right Leg)

Step 6.
Attach the "C" brackets to the leg assemblies as shown. See figure 6-1 for detailed information.

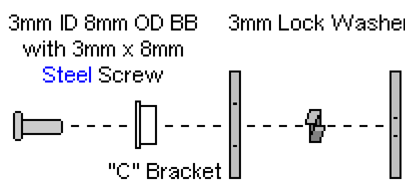


Figure 6-1.

Figure 6. (Left Leg)

Figure 6. (Right Leg)

Step 7.
Continue the similar arrangement for attaching one more set of same small “U” and “C” clamp with the multi-purpose clamp.

Step 7.
Attach a multi-purpose bracket to the "C" bracket as shown. See figure 7-1 for detailed information.

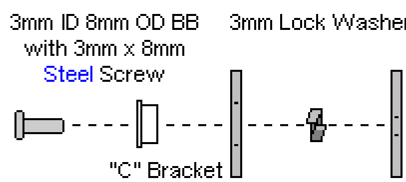


Figure 7-1.

Figure 7. (Left Leg)

Figure 7. (Right Leg)

Step 8.
Attach the leg assembly from step 7 to the 160 x 65mm rectangular plate as shown, using four 2-56 x .250" screws and 2-56 nuts on each side.

6 x **6 x**

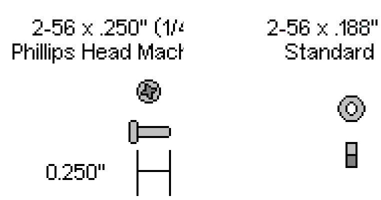


Figure 8. (Left Leg)

Figure 8. (Right Leg)

Step 9.
Your assembly should look like the image so far. Note, in the image the robot is face down. Note that since we now have a single object to work with, we will be proceeding with a single image from here on.

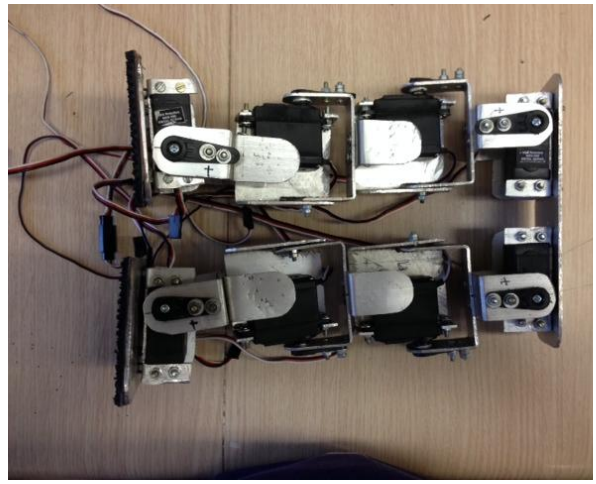
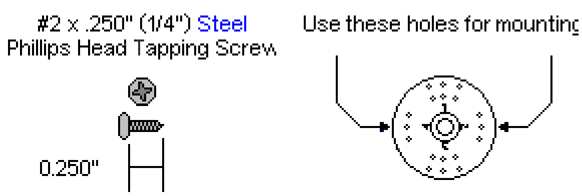


Figure 9.

Step 10.
Install the two ankle servos as shown, using the included 3mm hardware, two #2 tapping screws, and the diagram below. Note, your servos may be a little off. We will fix this in software later.

4 x



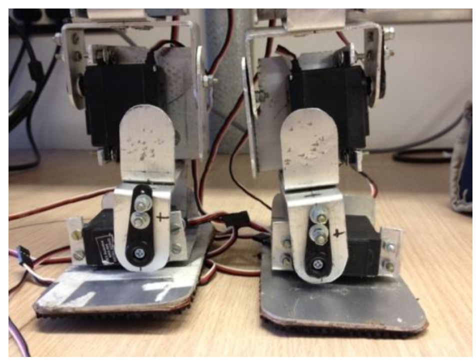
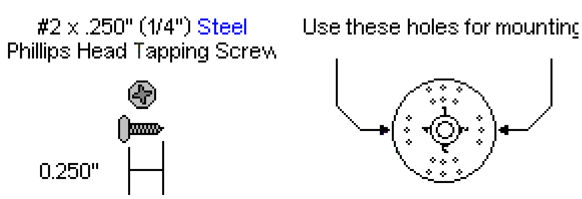


Figure 10.

Step 11.
Install the knee and hip servos as shown. Use the #2 tapping screws and the diagram below.

8 x



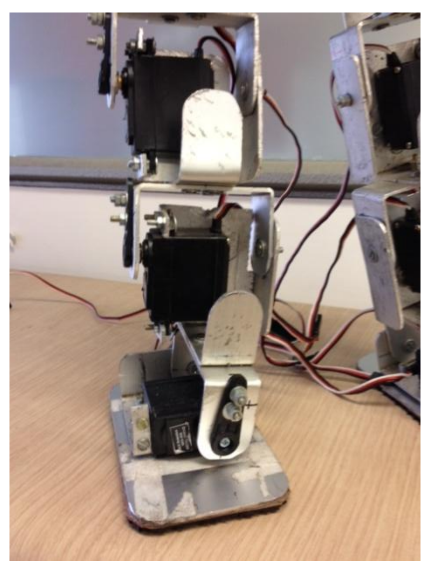


Figure 11.

Step 12.
To prevent the wires from tangling, you will want to secure them as shown. This can be done with wire ties or similar, not included. Make sure that the ankle servo is positioned as shown when securing the wires to ensure the full range of motion is available.

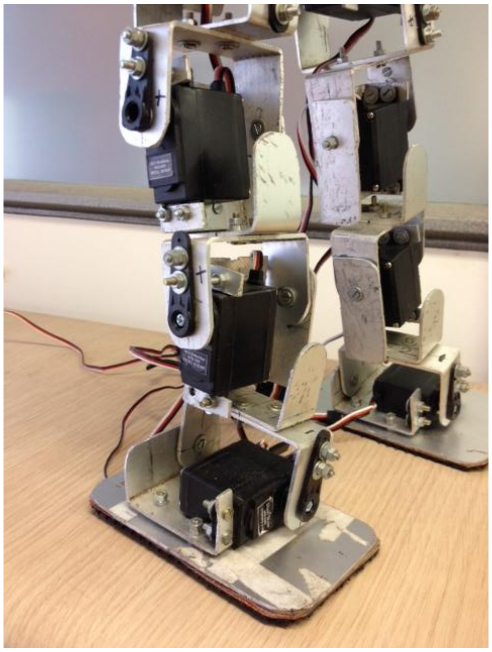


Figure 15.

Step 13.
Attach the power/reset switch to the electronics carrier as shown or you can even use the controller reset switch.

Step 14.
The BRAT hardware is ready to be programmed.

Chapter 6

Results And Discussions

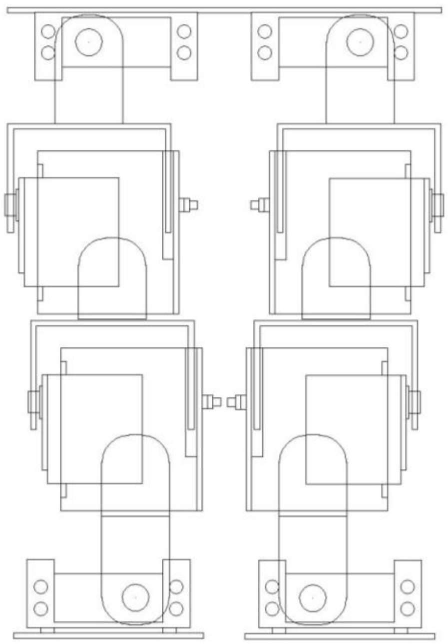


Fig. a. AutoCAD Design of Robot Before Actual Construction

Fig. b. Final Robot after Construction

- The hardware the Biped Robot was completed successfully with 90% to 95% accuracy with reference to the AutoCAD design.
- Currently, we have succeeded in programming the hardware for a stable anthropoid gait motion for walking in forward, reverse and sideways directions.
- The mechanism for remotely controlling the robot is ready, but is yet to be integrated with the current bipedal structure.
- With this platform, we have achieved stability in ideal conditions, which is flat walking. A step size of 7cm on a prototype of 25cm height and foot lift comfortable for kicking motion has been achieved.
- Further, this structure can be used to climb stairs by a slight modification in the algorithm.
- The principal factor for the success of a bipedal system is its stability in ideal conditions. This can be observed by the “lift” of the foot, the current consumed by the motors and step size.

Chapter 7

Conclusion & Future Scope

1. Applications and Future Scope :

- This is a bipedal platform on which application specific features can be added.
- An Inertia Measuring Unit (IMU) can be added for self-stability. First, the ideal values from the IMU are known relative to the motion of the bipedal robot in ideal conditions. For self-stability, we check the difference between the ideal-case values and actual values and take corrective action accordingly.
- The human foot has the maximum number of degrees of freedom (24). Due to this, it is possible for humans to walk on any terrain. Adding degrees of freedom to the robot's foot will help smoothen its motion and make it more terrain-flexible.
- Replacing the hip and ankle motors with spherical mechanisms can help save power, reduce weight and improve structural versatility.
- For up-scaling the prototype, the Servo motors can be replaced with high-torque DC motors. This has two advantages – firstly the cost reduction, and secondly the feedback needed to drive the DC motor to a specific position, in the form of variable resistance or encoders, is needed for the self-stability algorithm. This helps in further simplifying the structure.
- In the industrial scenario, soft-automation is increasingly preferred over hard-automation. This structure is versatile and by programming different motions the same structure can perform various functions. For example, in industries this structure can climb stairs to transfer material from one level to another. Also, it can use its squatting motion to act as a stop-go lift.
- This structure can very easily be remotely controlled. It can have a few pre-encoded motions which it can perform on receiving the respective signals. Else, it can also use a higher processor which directly feeds the angles of every motor to the robot's controller and precise execution of any motion can take place remotely.
- A camera can be added to the structure and image processing can be used for decision making, either on the robot itself or remotely.
- A robotic gripper/arm capable of lifting weight can be added. This arm has to be balanced against the center of gravity of the bipedal structure. With this synchronization, it is possible to use the same structure with lesser motion of its transverse motors.
- With self-stability, remote control, robot vision and lifting capabilities, such a robot can be ideal for military applications. It is particularly effective to weed out mines and save precious human life.
- Currently, research is going on about using the human brain directly to control objects. This is done by capturing the "brainstorms" an individual has while the person is performing a particular action. By capturing such "brainstorms" of

physically challenged persons and actuating those motions, it is possible to give them artificial limbs which can help them walk near-naturally.

- Based on the analysis and study, the output of this type of robots can be used for developing artificial limbs for the physically challenged person.

2. Conclusion:

An extensive Literature Survey conducted for the project gave profound insight on the requirements for building the robot. Based on the Literature survey, the inputs for designing the robot have been decided and Software model has been created. After creating the software model it is fabricated and tested.