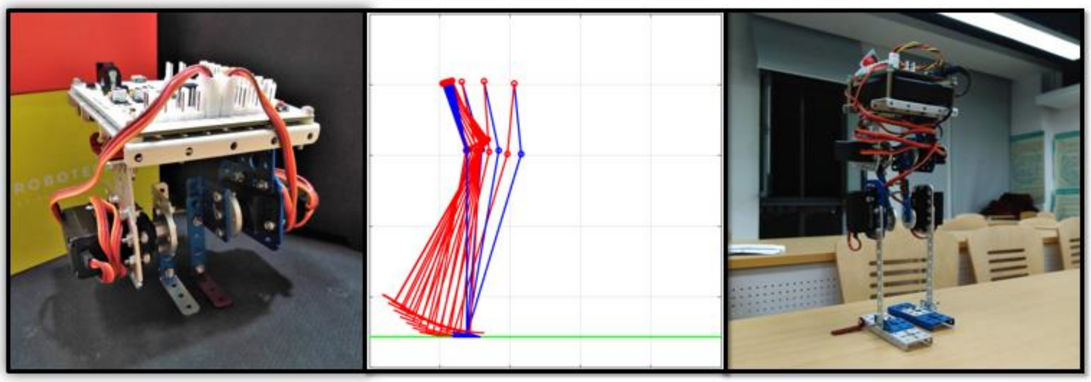
TONGJI UNIVERSITY

# Experiment Report for
# “Fundamentals of robotics and Artificial Intelligence”

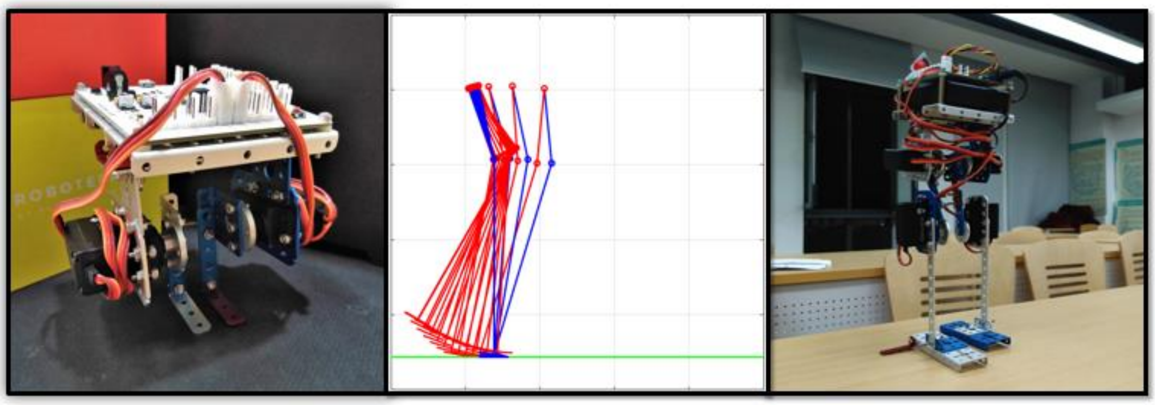


| | |
|---|---|
| ***Report Title*** | *A simple 2-DOF Biped and A 4-DOF Biped using tipping motion* |
| **School** | School of Mechanical Engineering |
| **Major** | Mechatronics |
| **Group Member** | 1551298 Kaixian QU |
| | 1551639 Zhe LIU |
| | 1551895 Yinglong XU |
| | 1551744 Yulin LI |
| **Advisor** | Prof. Dr.-Ing. Qirong Tang |

# A simple 2-DOF Biped and
# A 4-DOF Biped using tipping motion

Kaixian QU Zhe LIU Yinglong XU Yulin LI

Tongji University,
Jiading District, Shanghai, China



## Abstract

In this paper, we present one 2-DOF and one 4-DOF bipedal robots. We use 2 servos to control the 2-DOF biped, 4 servos for the 4-DOF biped, and modular components to construct their system structure.

With fundamental walking algorithm, our 2-DOF biped could walk in statically stable way, though neither agile nor robust. Besides, we add sensors in hope of making it more interactive.

Next, we propose a walking algorithm taking advantage of tipping motion for our 4-DOF bipedal robot. We conduct some experiments to test our theoretical assumptions, which, unfortunately, are proved tenuous for our 4-DOF biped. Nevertheless, we discuss some potential future works to be done on our 4-DOF biped, so that our walking algorithm will take effect.

# Contents

# 1. Introduction

Autonomous robotics has been an active research area for long time. Researchers are especially fascinated to the idea of creation of a versatile robotic platform able to perform highly dynamic tasks such as running and jumping autonomously in difficult terrain where wheeled robots cannot go. Thus, legged robots draw people's attention for its easy adaptation to uneven terrain. Furthermore, legged robots do not need a continuous path of support, but only isolated footholds. Bipedal robots like Atlas [1] can potentially assist humans in their daily tasks, bringing additional value to human society. Currently, bipedal robots, however, are not agile enough to perform many complex tasks that humans' legs routinely do, as their sensory system is too simple compared to ours. As locomotion is one of the most fundamental ability for most of us, we always overlook how many parts of our body are involved and how complex the interaction processes are even within a most basic motion. Therefore, it is a very hard task for robots to obtain this agility.

To study and test this advanced and challenged kind of robots, we have developed a simple version of bipedal robots. What we aim to do is to build bipedal robots who can walk on the flat ground. If successful, it can be used to build a humanoid robot.

The objectives of our robots are summarized as follows:

- The creation of a 2DOF bipedal robot and a 4DOF bipedal robot using the equipment provided in the lab.
- To attain the statically stability. In the field of legged robots, the stability is consisting of statically stable walking and dynamically stable walking. Dynamically walking have many methods, such as ZMP (Zero Moment Point) method [2] and HZD (Hybrid Zero Dynamics) method [3]. However, as we mentioned before, it can only be considered insofar as we have many kinds of accurate sensors that are not available in this experiment. Thus, we have to resort to achieving the former, statically stable walking.

## 2. Related Work


Figure 2.1: Some successful humanoid robots

Honda Robotics has developed a world-famous humanoid robot ASIMO [4]. The Humanoid Robotics research Group (HRG) in the Intelligent Systems Research Institute (ISRI) of AIST (National Institute of Advanced Industrial Science and Technology) has made several humanoid robots named HRP-2 [5], HRP-3 [6], and HRP-4 [7]. These robots are all controlled via the ZMP method and they achieved dynamically stable walking other than merely statically stable walking.

Figure 2.2: CASSIE from Agility Robotics

There are some bipedal robots which use the HZD method to achieve high robustness and high energy efficiency, such as CASSIE [8] from Agility Robotics. This method is becoming more and more popular these days.

However, the ZMP method is more fitted for Humanoid robots, and the HZD method is still mainly for Legged robots without torso. To use ZMP method, we must have sensor to detect the actual CoM , yet we do not have one. Therefore, we could just build one to walk on the given environment, which, in our case, is the flat ground.

Humanoid robots normally have 6 DOF for each leg, just like us. One DOF means one servo, which is to say, we need at least 12 servos to build a human-like bipedal robot. Nevertheless, we do not have so many servos, and too many servos mean more complicated control algorithm. As a result, we first attempt to build a 2-DOF robot, and then extend it into a 4-DOF robot.

Figure 2.3: DOF configuration of ASIMO

However, this by no means indicates that we are going to build some toys with giant feet, which could walk in any way without worrying about tipping over or falling down. We are trying to achieve statically stable walking not by increasing the support polygon [9] - the region formed by enclosing all the contact points between the robot and the ground as the picture indicated below.

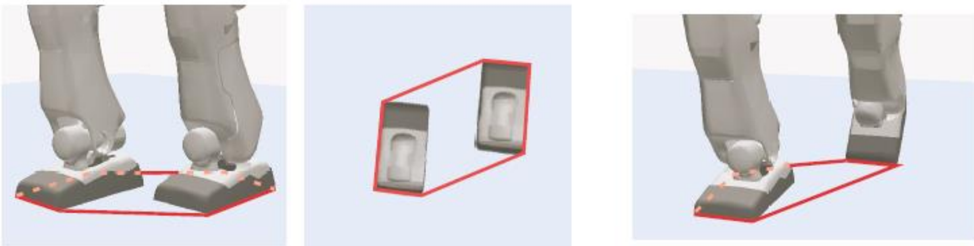


Figure 2.4: Definition of support polygon

# 3. 2-DOF Bipedal Robots

## 3.1 System Structure

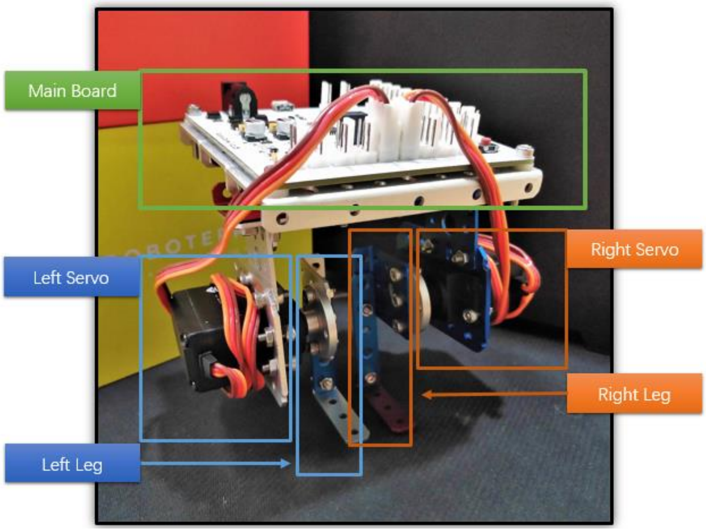



Figure 3.1: System structure of 2-DOF biped

Our 2-DOF biped has 5 main components: main board mounted on the top, the left servo, the left leg, and their counterparts on the other side. The relative position of Main Board and two servos are fixed. The servo could rotate its output shaft at the given speed to the given angle. The output shaft is connected with the leg rigidly, which makes the servo control the leg's position and angular speed. In the figure below, 2 degrees of freedom are highlighted.

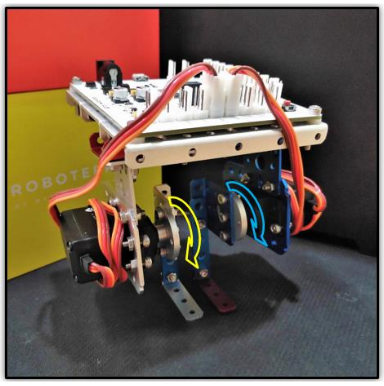


Figure 3.2: Highlighting of 2 degrees of freedom

## 3.2 Walking Algorithm

In order to make a 2-DOF robot move stably, we resort to one simple method after trying many novel ways. We just let the left leg and the right leg move in opposite direction to the opposite destination at the same absolute angular speed. The moment they reach one of their destination backward or forward, they move back to the other destination. The left and right legs will repeat those moves and this make the robot walk in forward direction. We show one series of picture of walking pattern about the previous robot so as to illustrate this point more clearly.

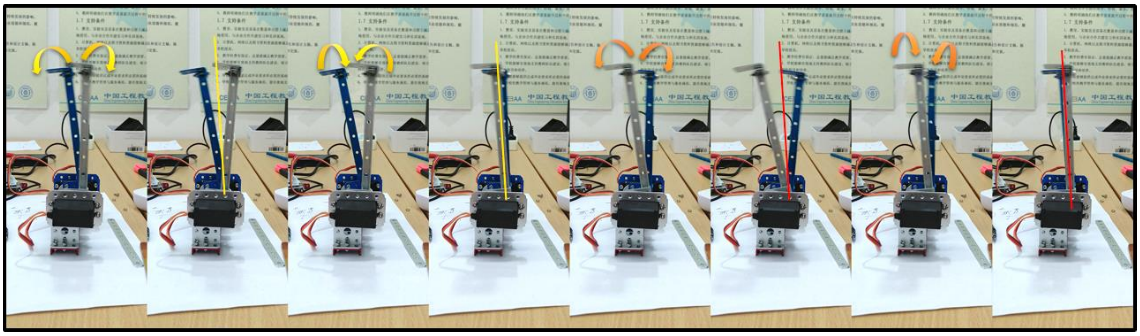


Figure 3.3: Series of picture showing the biped's walking pattern

### 3.3 Experimental Result

#### 3.3.1 Walking forward

When we first run the program on the main board, the result turns out to be unsatisfying. The robot rotates its legs as we expected, yet no forward velocity generated. Instead, our biped just remain there. We attribute this phenomenon to little friction.

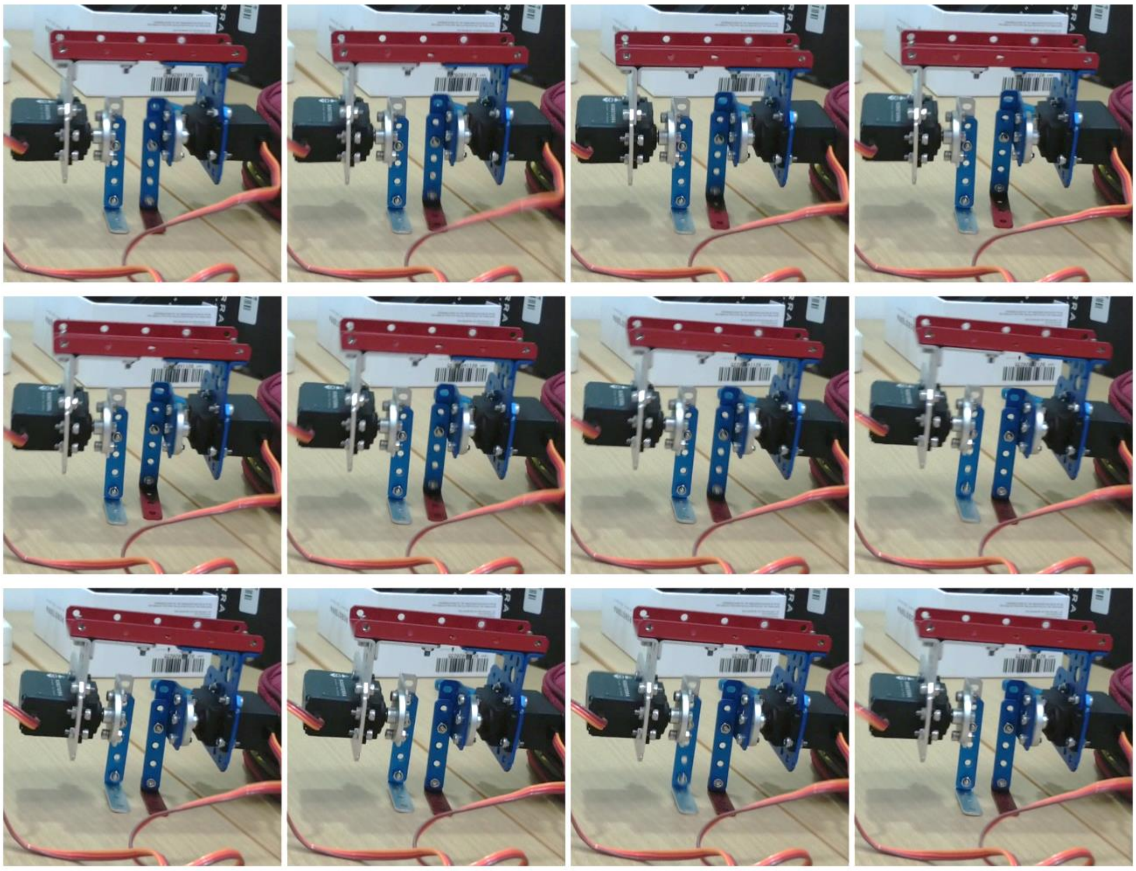


Figure 3.4**:** First try walking on the smooth surface

Consequently, we put this biped on a rubber pad to increase friction coefficients. Then, the robot starts to move in forward direction at a relatively low speed.

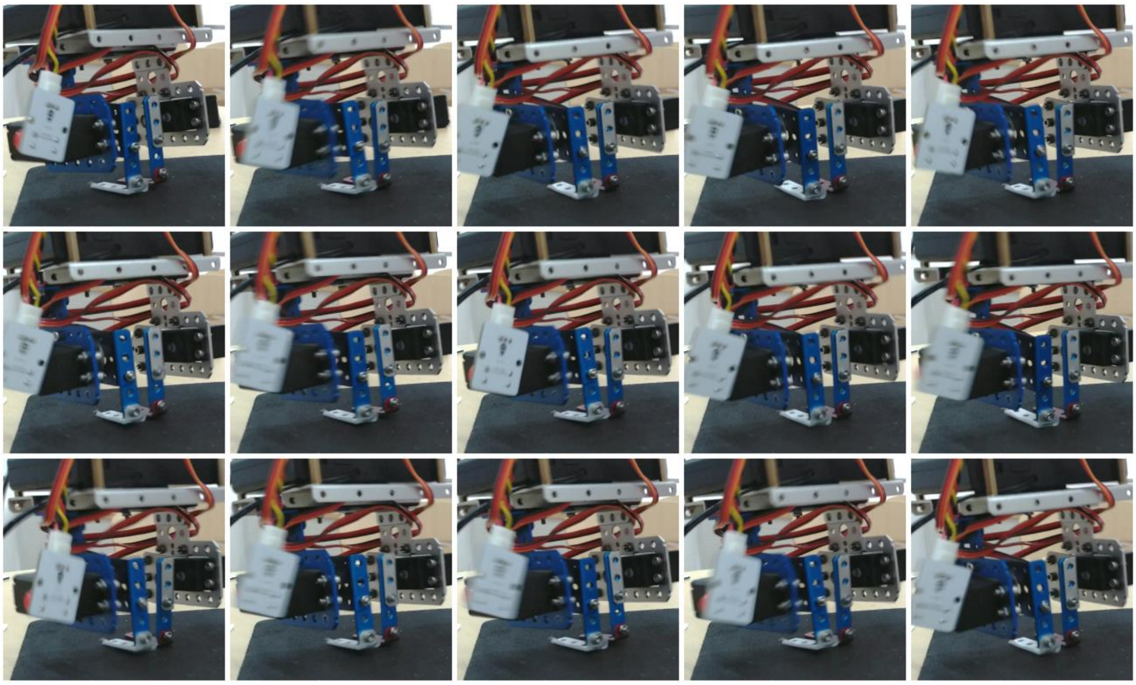


Figure 3.5: Successfully moving forward on a rubber pad

In order to show its movement, we choose to blend the first picture and the last picture above, which gave us Figure 3.6.

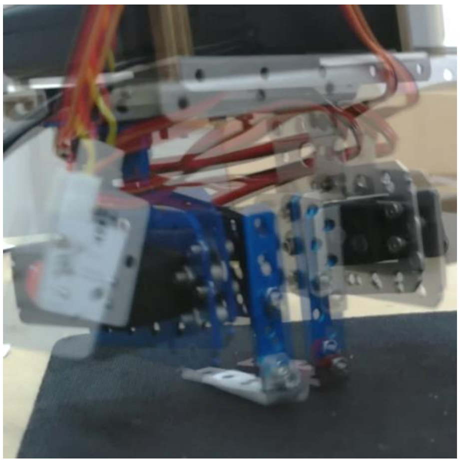


Figure 3.6: First and last pictures blended

### 3.3.2 Turning around

In addition to walking in straight line, the robot can also turn around. A series of photos showing its turning are shown below in figure 3.7. In this situation, our biped is turning to its left, and its right leg keeps touching the ground as a pivot. Each time biped swings its left leg forward towards the ground, the ground gives our robots a reaction force in the opposite direction. And this friction force has a moment effect on the right leg, which makes robot turning to its left.

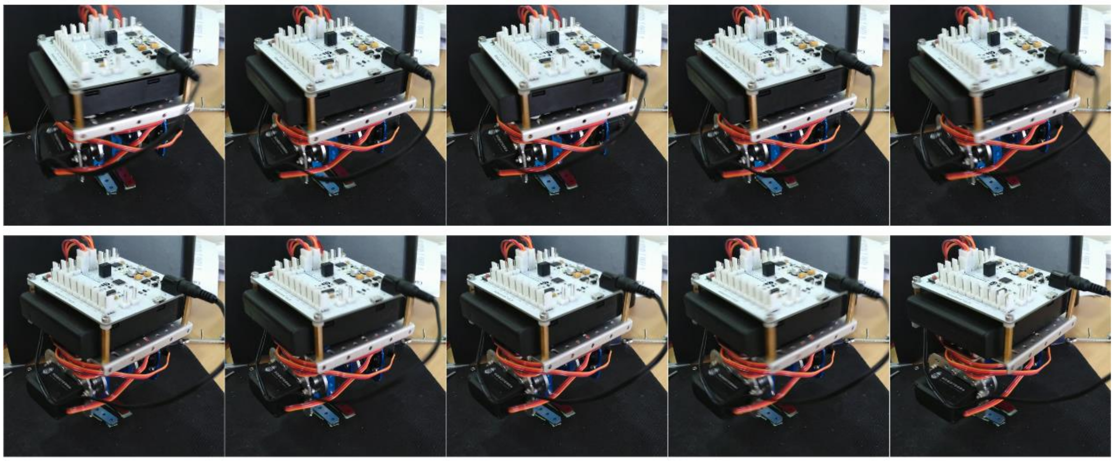


Figure 3.7: Our biped turning around

This movement, instead of walking normally, may be the consequence of the biped's leaning its center of mass on its right leg. We demonstrate its principle of turning in Figure 3.8

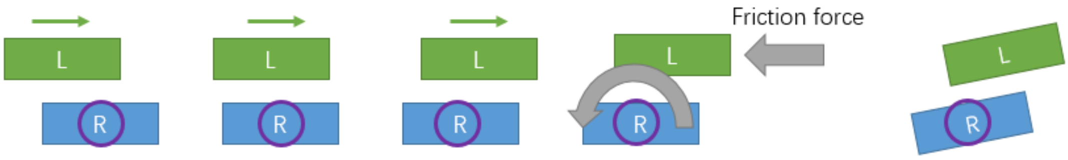



Figure 3.8: Analysis of the turning process

### 3.4 Sensory Systems

In addition to basic walking function, we add some sensors to make our biped more interactive.

#### 3.4.1 Sound control

When powered on, the robot will stand still waiting for a signal, which, in this situation, is sound. If you say something loud to the sound sensor, the robot will begin to move, and when your voice stops, it will stop its movement. If you keep saying 'GO', the robot will walk constantly. When you stop, our robots will stop as well. We added this function because it makes you feel like you are talking to the biped.

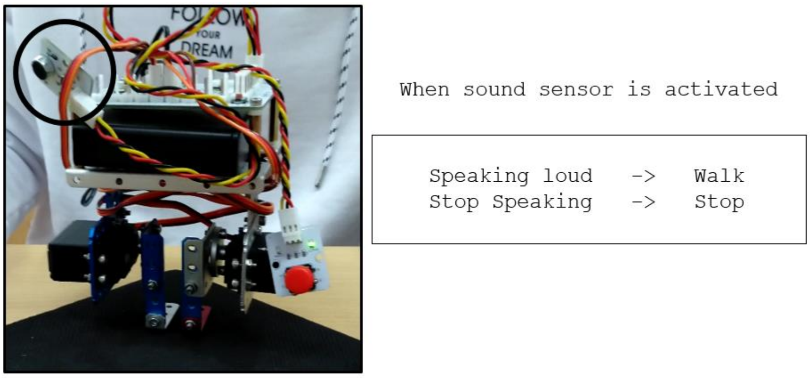



Figure 3.9: Pattern of sound control

### 3.4.2 Button control

However, you may also want the robots to move without this control signal. You can just press the red button, then the robot will go into automatic walking mode. It will just keep walking until you press that button again. Walking in this mode, it will deactivate sound sensor. When it stops walking, the sound sensor will be reactivated again waiting for the sound signal.

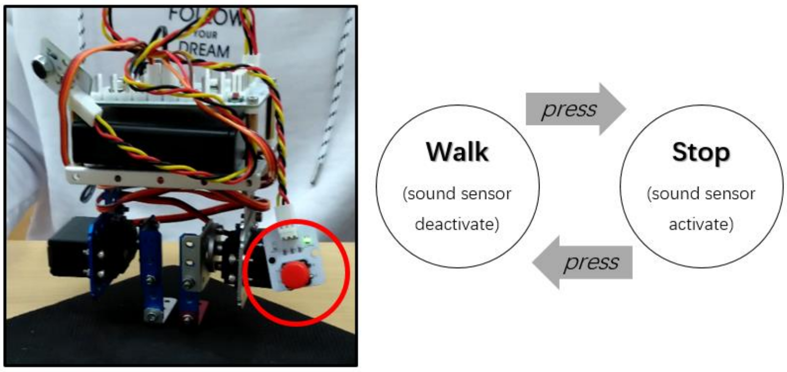



Figure 3.10: Pattern of sound control

### 3.4.3 Overall control

The overall sensory control systems could be summarized as Figure 3.11.

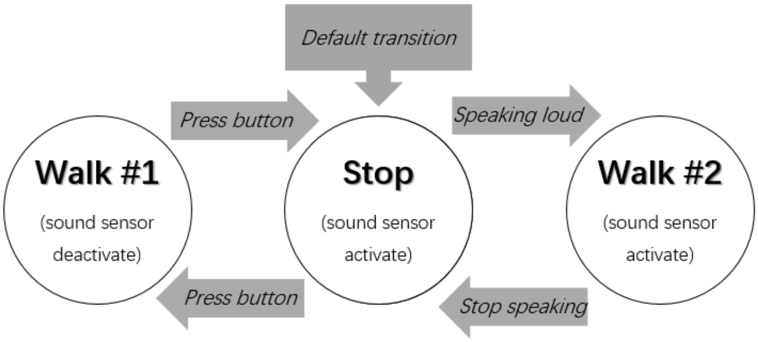



Figure 3.11: Pattern of button control coordinated with sound control

## 3.5 Structure Development

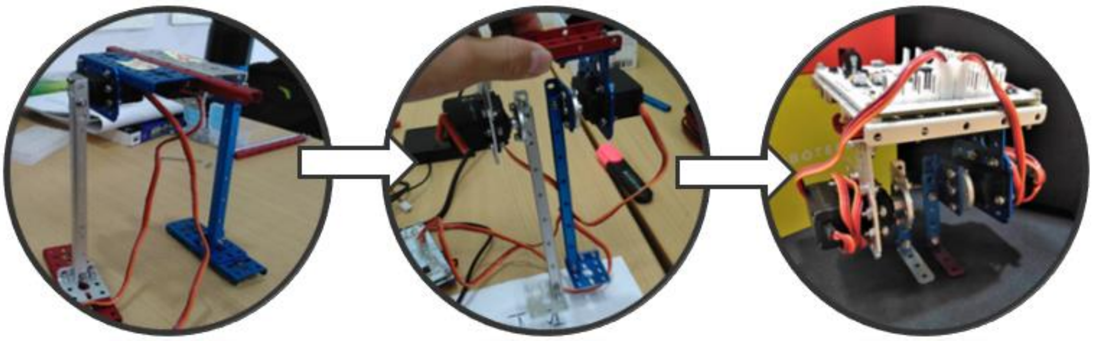


Figure 3.12: Iteration of the 2DOF biped

Before we built the final version of the 2-DOF robot described, we have iterated twice the structure of it. The iteration process is shown in Figure 3.12 as the arrows imply.

### 3.5.1 Version 1

In view of the limited experimental equipment, we firstly attempt to build a simple two-degree-of-freedom biped in Figure 3.13

Figure 3.13: 2-DOF Biped (Version 1)

The presumptive forward walking motion of the robot is disassembled as follows: the left servo motor is not rotating, and the right servo motor turns the appropriate angle, so that the right leg is raised forward relative to the waist. Then the CoM is transferred from the center of the original robot to the right leg. After that the left and right side servo motors turn the same angle but in different direction to make the waist back to the horizontal position. Finally, the left side servo motor turns the left leg forward aligning the right leg to complete the walk motion.

Experiment did not successfully complete the expected targets. Specific issues were embodied in two aspects. First, the length of supporting foot that contacts with the ground is too big. Since there is only one degree of freedom for each leg, foot interferes severely with the ground. Second, while the CoM moves forward to the right leg, it is too sideways to maintain standing.

### 3.5.2 Version 2

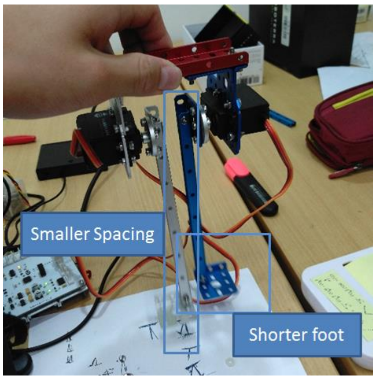



Figure 3.14: 2-DOF Biped (Version 2)

On the basis of the first experiment, we improved the structure of the robot aiming at the existing problems, mainly embodying in the following two points: the distance between two legs is reduced to reduce the lateral deviation of CoM, the foot length is shortened to reduce interference with ground. In addition, we also reduce the servo motor rotation angle for one step, hoping to solve the problem of side turn.

Make walking experiments again and we find that the interference and the lateral deviation of the CoM is reduced. Nonetheless, a new problem appears: the robot seems to be staying in the same place, not walking the ideal distance. With this problem to finished we develop the final experimental version biped shown in Figure 3.1.

### 3.6 Demonstrations

The 2DOF biped can walk forward in the following basic order ([2-DOF] Video 1: Automatic walking mode)

1) Power on the main board.
2) Initialize the left and right legs.
3) Move the left leg and the right leg in opposite direction to the opposite destination at the same absolute angular speed.
4) Once they reach the destination, move back to the other terminus.
5) Repeat this process and move forward.

The functions for turning around are as follows ([2-DOF] Video 2: Turning around):

1) Power on the main board.
2) Initialize the left and right legs.
3) The right leg keeps touching the ground as a pivot.
4) Swing the left leg forward towards the ground, the ground gives our robots a reaction force in the opposite direction. And this friction force has a moment effect on the right leg, which makes robot turning to its left.
5) Repeat this process and turning around.

Our robot can also perform walking and turning around under the control of sound and button ([2-DOF] Video 3: Sound control mode)

1) The Main Board powered on.
2) Initialize the button and sound sensor.
3) Initialize the left and right legs.
4) Press the button once, and enter walking autonomously mode.
5) Press the button twice, and enter sound control mode.
6) Keep saying “Go” to make our biped walk autonomously.
7) It stops as the sound disappear.
8) This process can be re-triggered by pressing the button again.

# 4. 4-DOF Bipedal Robot

## 4.1 System Structure

There are six degrees of freedom in each human's leg. If we want to build a biped to walk precisely as human, 12 servos, at least, would be required. However, we would like to imitate human walking roughly with less servos. In our experiment, we chose to use 4 servos, and the degrees of freedom of each leg to be controlled are pitch of hip joints and the pitch of knee joints, since they are more active compared to others. Therefore, we aim to control the thigh and thin properly so as to make the robots walk.

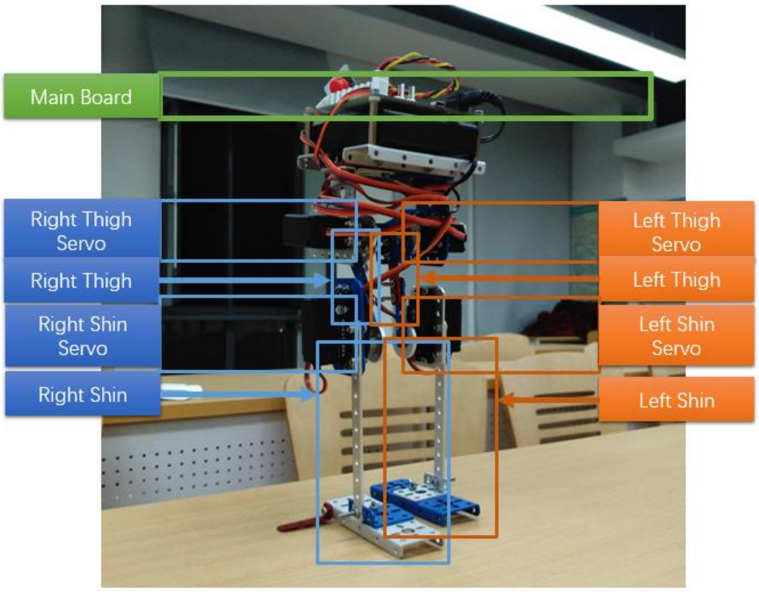



Figure 4.1: System structure of 4-DOF biped

As shown in picture Figure 4.1, Our final 4-DOF biped has 9 main components, main board still mounted on the top, the left thigh servo, the left thigh, the left shin servo, the left shin and their counterparts on the other side. The relative position of main board and two thigh servos are fixed, and so are the shin and the foot beneath. The servo could rotate its output shaft at the given speed to the given angle. The output shaft is connected with thigh rigidly, which makes the servo control the thigh's position and angular speed. In addition, the relative position of the shin servo and the shin are fixed, consequently, the thin servo could control the thin and the foot.

## 4.2 Walking Algorithm

### 4.2.1 Basic assumptions:

1) When one foot is in full contact with the ground, the other foot could swing back and forth freely unless the other foot is in its most forward position relative to the support leg.

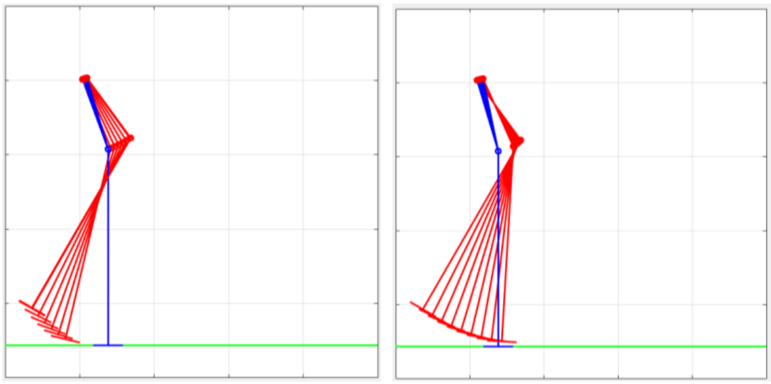


Figure 4.2: Nonsupport leg swings back and forth freely

2) When the other foot is in its most forward position relative to the support leg, the robot tips forward with respect to the toes of the support foot. Then the toes of both feet touch the ground without tipping over.

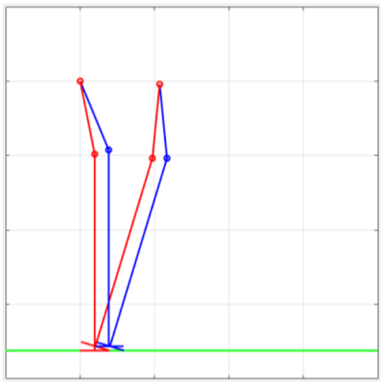


Figure 4.3: The biped tips forward

3) When two feet are both in partial contact with the ground, they will remain in partial contact until ones is in full contact.

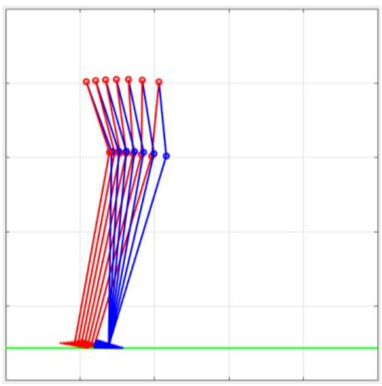


Figure 4.4: The biped remain in partial contact

4) When the robot tips it tips with respect to its toe of support leg. After the forward foot touch the ground, we assume its toe will not slip on the ground.

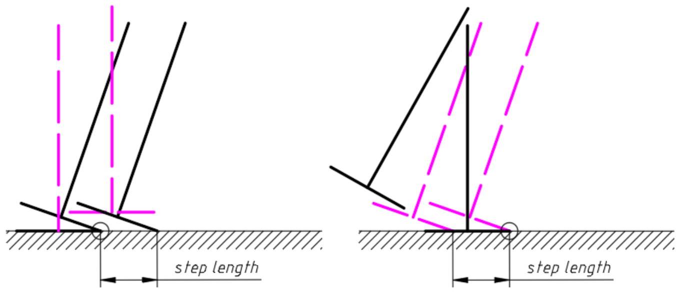



Figure 4.5: The robot tips forward to make a step (in real world)

There is another way to show its motion, which we use here. Since rotation is relative, we can rotate the ground with respect to the no-slip point when the foot is in not in full contact with the ground. Using this method, there is at least one horizontal foot, which make us program its motion more easily without loss in generalization.

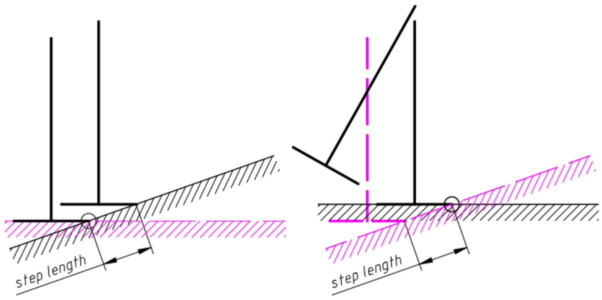



Figure 4.6: The robot tips forward to make a step (in real world)

Here after one gait cycle (right leg or left leg), the ground is raised to another level. To make this animation more real, we could also rotate the ground to the horizontal level with respect to the forward foot toe. In reality, robot tips forward with respect to the support foot, yet in our model, it tips forward with respect to the forward foot. There is no much difference in horizontal plane between 2 representations.

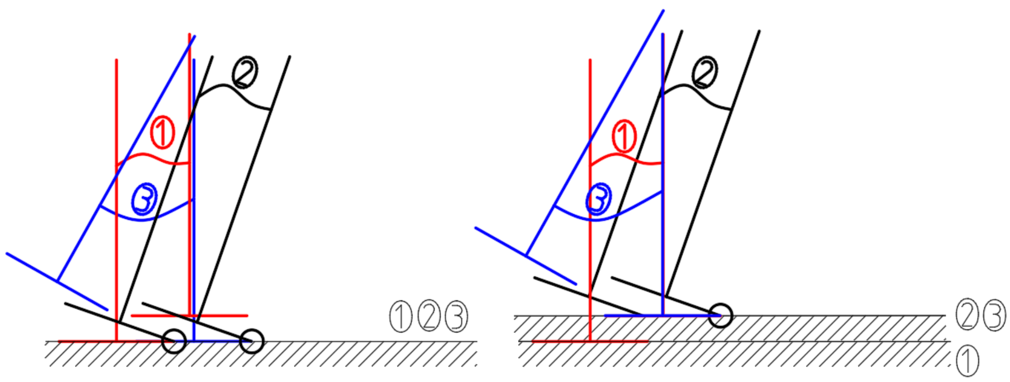



Figure 4.7: Reality Vs. Our representation

### 4.2.2 Walking Gait

From the assumptions above, it is not hard to figure out our bipedal walking gait.

- **State 1:**

The feet of both legs are in partial contact with the ground. The left leg is in front of the right leg.

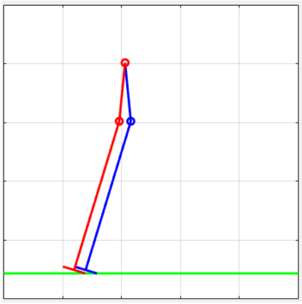


Figure 4.8: State 1

- **Transition between state 1 and state 2**

**Right leg motion:**

1) Rotate the right thigh forwards and right shin backwards.

   This shortens the distance between right toe and hip joint, which could clear the ground for the following forward action. Right foot first touches the ground until the left foot is in full contact with the ground.

2) Rotate the right thigh backwards and right shin forwards.

   The angle of right hip joint and right knee joint equals their counterpart of the left leg in state 1.

**Left leg motion:**

3) Rotate the thigh backwards.

   Since this is the main support leg, this makes the hip joint move forward.

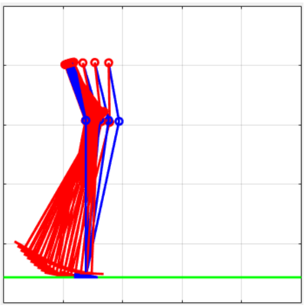


Figure 4.9: Transition between state 1 and state 2

- **State 2:**

Due to the gravity, this whole robot rotates with respect to the support leg until both toes touch the ground.

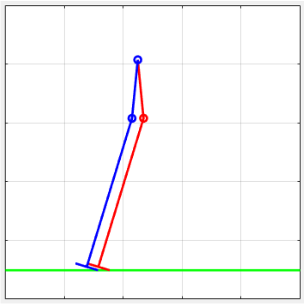
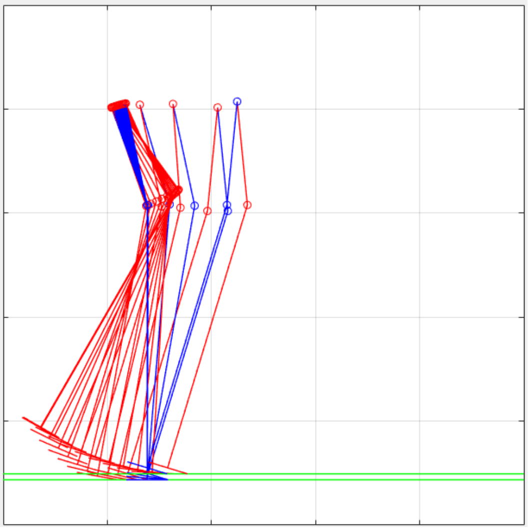
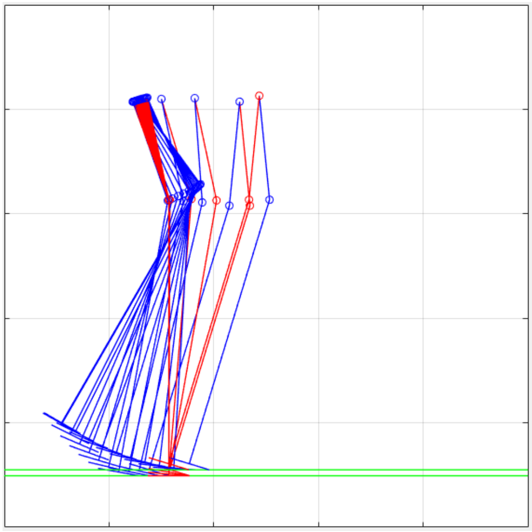


Figure 4.10: State 2

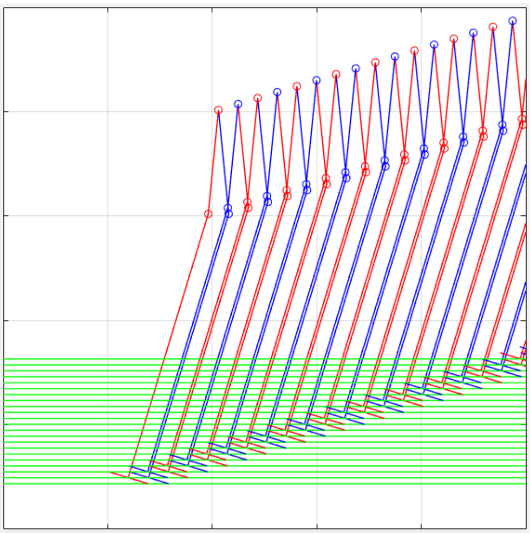
As you can see here, the left leg and right leg exchange their relative position. Besides this, everything else is the same but at another place. If we would like to go back to state 1, we just need to repeat what we do above with different legs. After this, our biped will have completed one cycle and successfully stepped forward. In the picture below, we have shown a sequence of biped's motion.



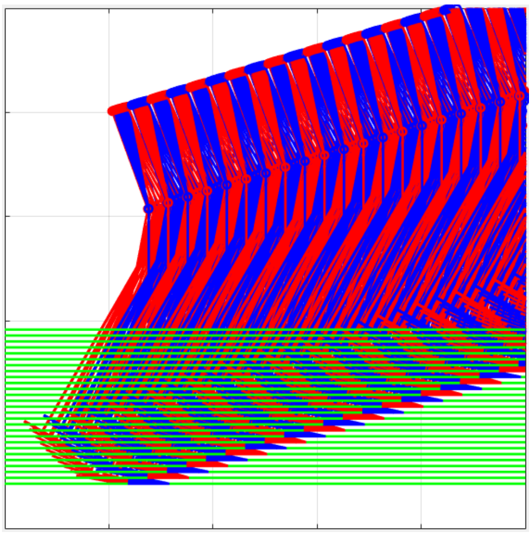
a) Transition from state 1 to state 2



b) Transition from state 2 to state 1



c) Sequences of state 1 and state 2



b) Sequences of state 1, state 2 and transitions

Figure 4.11: Demonstration of our biped's motion in MATLAB®

We could also show our walking algorithm using the model we built in Inventor. See Figure 4.12 below.

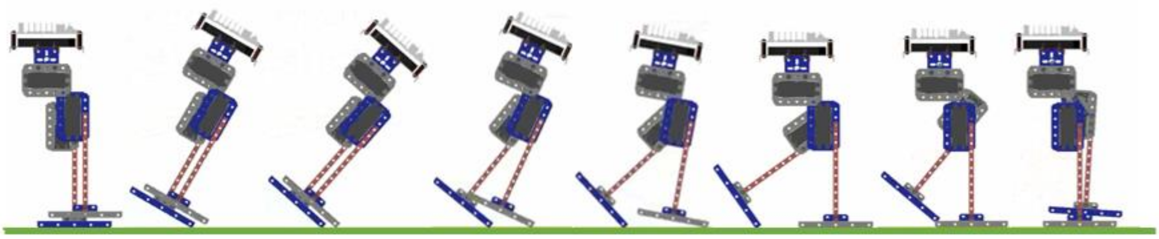


Figure 4.12: Demonstration of our biped's motion using Inventor®

**4.3 Experimental Result**

Our walking algorithm works if and only if our assumptions hold true. Therefore, what we need to do is to test validity of our assumptions.

**4.3.1 Test of assumption 1**

The major point of our assumptions 1 is that the robot can support itself on one foot. If the support leg is in full contact with the ground, the other leg should swing freely until it falls when reaching its most forward position.

We wrote an easy program in order to find the angle where biped could not support itself with one foot and falls to the ground. The left shin servo rotates an angle $\alpha$ at a certain speed backward. Meanwhile, the servo of the left thigh rotates the same angle $\alpha$ forward, thus ensuring left foot always parallel to the ground. At the same time, the servo of the right thigh also rotates at a suitable angle, so that the torso could moves forward. This movement is shown as the picture below. If successful, we could find the angle where biped falls and use this angle as the destination angle in our biped. Otherwise, our assumption would be tenuous.

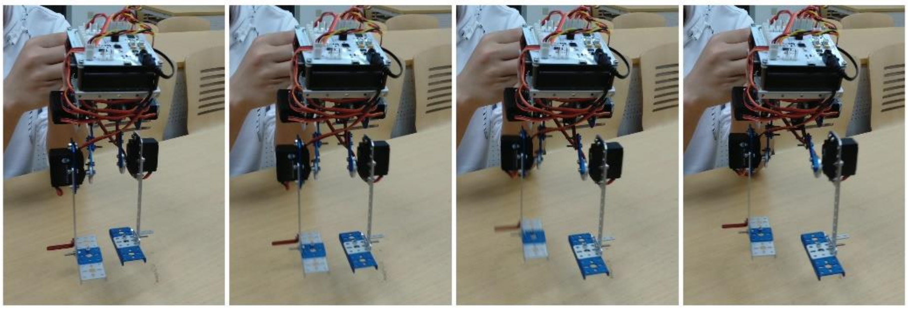


Figure 4.13: Motion programmed to test our assumption 1

We downloaded it to our main board and tested its performance. It turns out one foot is not enough to support its weight. The moment we raise the rare leg, the robot just tips to make the rare leg touch the ground. In other words, both legs touch the ground in the whole process as the figure 4.13 shows.

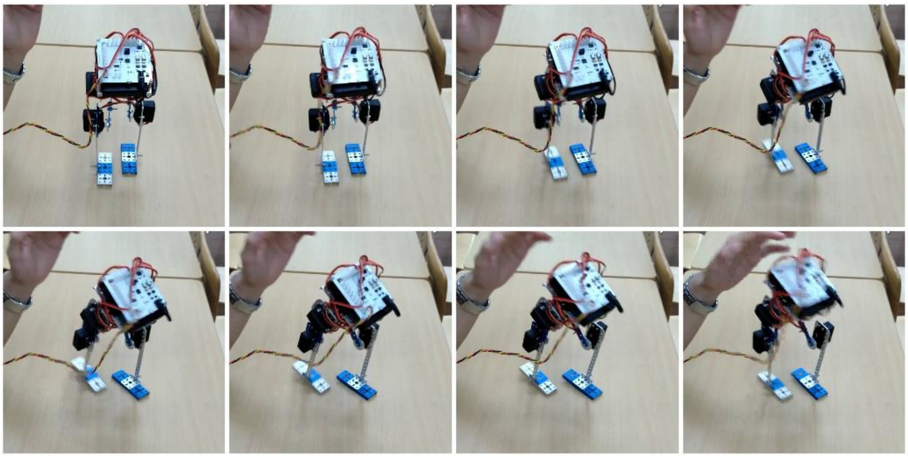


Figure 4.14: Tested performance

Our assumption 1 was not valid. We will present some potential future developments in section 4.5.1 to make our assumption 1 work.

**4.3.2 Test of assumption 2**

Holding the robot in our hand with its rare leg touching the ground and forward leg at its destination, we let the robot tip forward until both feet touches the ground. Our robot, however, could not stand still. It falls over to the ground instead. We fails to take into account that CoM will get outside support polygon when our biped tips forward, which will create a tipping moment to make the robot fall over immediately.

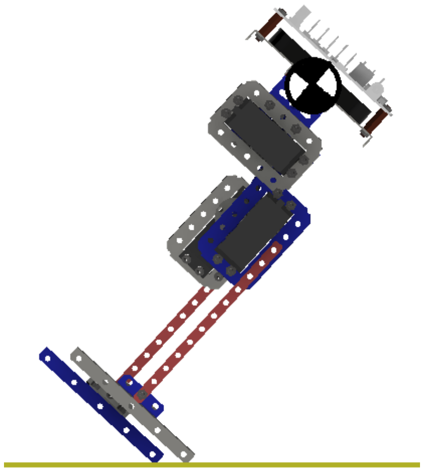


Figure 4.15: CoM outside the support polygon when tipping forward

Our assumption 2 was not valid. We will present some potential future developments in section 4.5.1 to make our assumption 2 work.

**4.3.3 Test of assumption 3**

When the ground projection of CoM is inside the support polygon, assumption 3 is proved to be valid. However, the robot will not maintain its balance when CoM outside the support polygon, let alone moving its legs. The assumption 3 will be sound if the assumption 2 works fine.

### 4.4 Structure Development

#### 4.4.1 Version 1

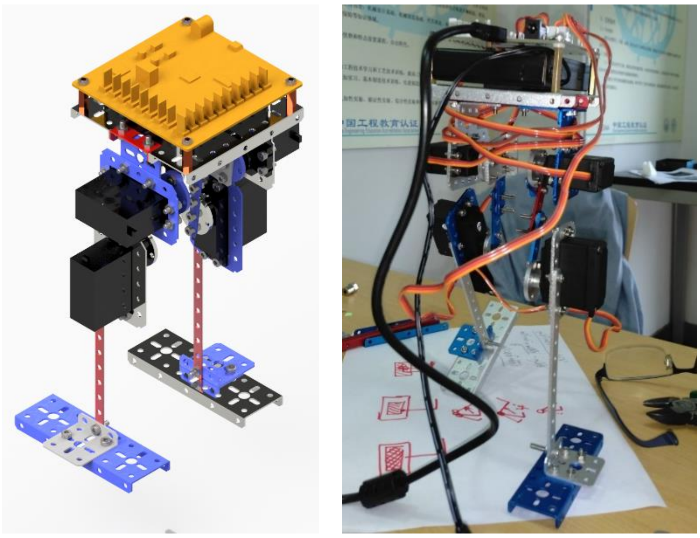


Figure 4.16: 4-DOF biped (Version 1)

Alpha 4-DOF biped robot did not achieve our target, because the moment robot stepped out its left leg, it just tips over towards the left foot immediately. When the left leg steps out, the CoM no longer remains in the support polygon, so that the torque of gravity force acting on the contact point makes it overturn. Because of the long distance between the legs, the robot turn over immediately before the left leg plays a supporting role again.

#### 4.4.2 Version 2

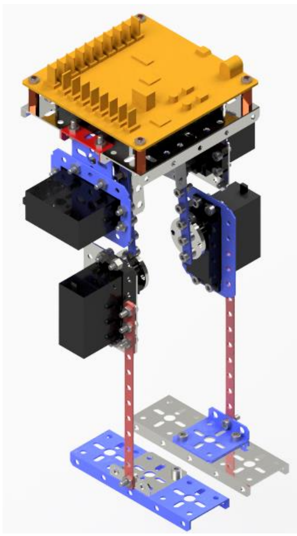


Figure 4.17: 4-DOF biped (Version 2)

The relatively far distance between the two legs will cause a severe tipping moment. Thus, we modifies our first version by shortening the distance between the two legs under the premise of no interference.

Our 4-DOF robot's static stability has been amended, however, the upsetting phenomena still existed, remained to be solved. The moment the left leg steps forward, our robot's CoM still fall on the left-rear corner of it, generating a tipping moment by the supporting force of the right side of the leg.

### 4.4.3 Version 3

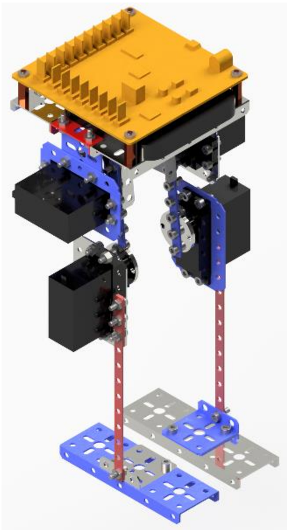


Figure 4.18: 4-DOF biped (Version 3)

When we found that our robot could not stand still when some disturbance exists, we put the battery box a little forward to keep CoM some distance away from edge of the support polygon. After that, the tipping moment generated in the sagittal plane has increased a lot when our biped tips forward. Now we are faced with a dilemma that the CoM should be relative forward to maintain statically stable, yet it shall be relatively backward when tipping forward to avoid toppling over.

## 4.5 Future potential developments

Though our robot cannot fulfill what we designed, it will be capable to do so with extra effort on controlling its CoM. In ZMP method, the major algorithm is how to use CoM to achieve desired ZMP trajectory. What we lack is one degree of freedom to rotate the CoM in the sagittal plane. Below is our discussion.

### 4.5.1 Assumption 1

In order to make our robot walk as our algorithm designed, we should always put the ground projection of CoM inside the support polygon. We could achieve this by making each foot cross the mid-point, just like the toy shown below. However, this is not we aim for.

Figure 4.19: Toys with giant feet making its CoM always stay in support polygan

Therefore, we resort to leaning our CoM to the support foot when only one foot is in contact. That would require another DOF, if we put a counterweight on top. Four degrees of freedom would be necessary if we just use the torso as the COM, just like the picture shown below. At the shifting phase between standing phase (both feet) and standing phase (right foot), we rotate the 2 DOF (brown arrows) above the knee for an angle leftwards and the 2 DOF beneath the knee rightwards for same absolute angle. This will help us lean CoM on the support foot without losing the parallel between the torso and the foot.

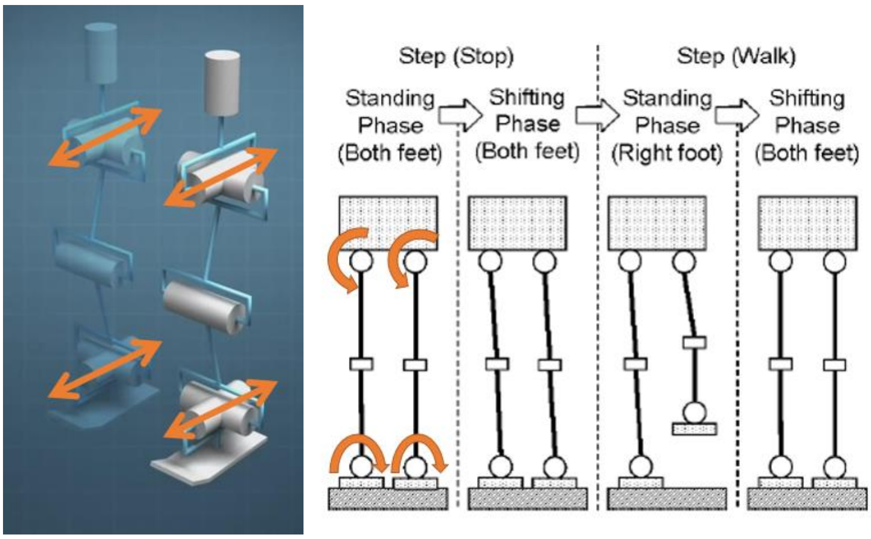



Figure 4.20: How to make the robot support itself with additional 4 DOF

To reduce the cost of our robot, we would like to add minimum degrees of freedom. So we can add one degree of freedom to rotate a counterweight leftwards or rightwards.

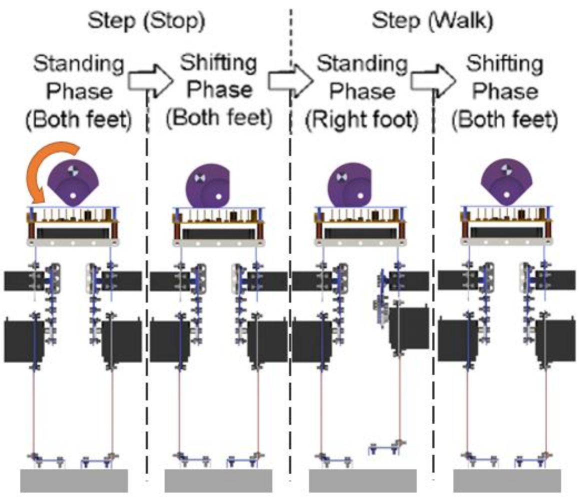



Figure 4.21: How to make the robot support itself using counterweight

### 4.5.2 Assumption 2

To make our biped tip forwards as we designed without toppling over, we would like to rotate the mainboard backwards when tipping forward. This could easily be done when both feet touch the ground. We just rotate left both thigh servos a same angle, the biped's torso could rotate this angle in the opposite direction, just like the picture below demonstrated.

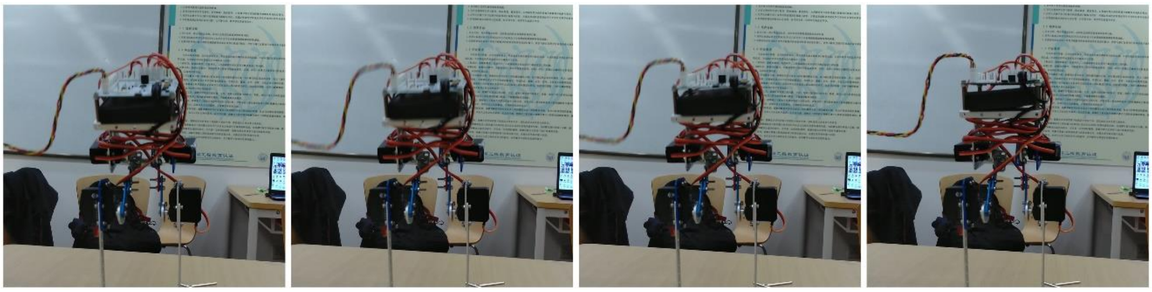


Figure 4.22: Rotating both thigh servos results in rotating torso

We should be able to achieve this when one leg is in contact with the ground. However, we cannot make the robot stand on one foot stably. If the assumption 1 is confirmed, we could just rotate thigh servo of support leg an suitable angle, the biped's CoM will be in support polygon. To find this time-variant angle, we need sensor to detect the position of the biped, and some advanced control law.

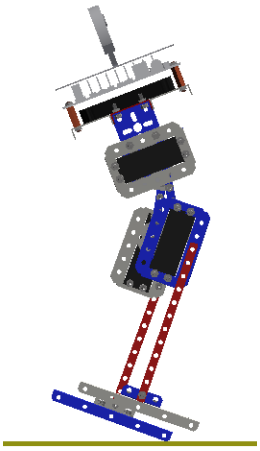


Figure 4.23: Rotating thigh servo of supporting leg results in rotating torso

**4.6 Demonstrations**

The 4-DOF biped robot can perform raising its head upon the press of the button ([4-DOF] Video 1: Raising head).

1) Power on the main board.
2) Initialize the button.
3) Press the button to make the 4-DOF biped robot raising its head.

We test our 4-DOF biped robot whether its one leg can swing freely when the support leg is in full contact with the ground ([4-DOF] Video 2: Testing assumption).

1) Power on the main board.
2) Initialize the button.
3) Initialize the left and right legs.
4) Press the button, and then both the left shin servo and the left thigh servo rotate an angle α at a certain speed to ensure the left foot is always parallel to the ground. We hope it could swing freely without touching the ground. Meanwhile, the servo of the robot's right thigh also rotates at a suitable angle.
5) However, it turns out that one support leg is not enough to maintain its balance. Our biped tips immediately and both feet touch the ground the whole time.

An animation made via Inventor to show our walking gait ([4-DOF] Video 3: Walking demonstration using Inventor).

# 5. Conclusion

We build one 2-DOF biped with two servos. Using a simple walking gait, this robot can walk forward or turning around in different situations. Aiming to make our robot more interactive, we add sound control mode and automatic walking mode via sound sensor and button. Our 2-DOF biped is relatively successful.

In addition, we propose a walking algorithm taking advantage of tipping motion. This tipping motion make our robot step forward. Some assumptions were made, and walking gait demonstrated. If our assumptions were valid, our walking gait will be effective with no doubt.

We build one 4-DOF biped to test our assumptions. However, the result turns out to be unsatisfying. We believe one additional degree of freedom to control a counterweight and an accurate sensor will be enough to satisfy our assumptions, and make our algorithm take effect.

## Acknowledgments

Our equipment is provided by RMB laboratory and Prof. Dr.-Ing. Qirong Tang. Mr. Zhenqiang DENG sacrificed his time to help us. Prof. Dr.-Ing. Qirong Tang and “Fundamentals of robotics and Artificial Intelligence” have helped us prepare for this report. All these supports are highly appreciated.

## References


[1] Feng S, Whitman E, Xinjilefu X, et al. Optimization based full body control for the atlas robot[C]// Ieee-Ras International Conference on Humanoid Robots. IEEE, 2014:120-127.

[2] Hirai K, Hirose M, Haikawa Y, et al. The development of Honda humanoid robot[C]// IEEE International Conference on Robotics and Automation, 1998. Proceedings. IEEE, 2002:1321-1326 vol.2.

[3] Eric R Westervelt, Jessy W Grizzle, Christine Chevallereau, et al. Feedback control of dynamic bipedal robot locomotion[M]. CRC Press, 2007.

[4] Cerny E, Gecsei J. The intelligent ASIMO: System overview and integration[C]// Ieee/rsj International Conference on Intelligent Robots and Systems. IEEE, 2002:2478-2483 vol.3.

[5] Kaneko K, Kanehiro F, Kajita S, et al. Humanoid robot HRP-2[C]// IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA. IEEE, 2004:1083-1090 Vol.2.

[6] Kaneko K, Harada K, Kanehiro F, et al. Humanoid robot HRP-3[C]// Ieee/rsj International Conference on Intelligent Robots and Systems. IEEE, 2008:2471-2478.

[7] Kaneko K, Kanehiro F, Morisawa M, et al. Humanoid robot HRP-4 - Humanoid robotics platform with lightweight and slim body[C]// Ieee/rsj International Conference on Intelligent Robots and Systems. IEEE, 2011:4400-4407.

[8] Xie Z, Berseth G, Clary P, et al. Feedback Control For Cassie With Deep Reinforcement Learning[J]. 2018.

[9] Kajita S, Hirukawa H, Harada K, et al. Introduction to Humanoid Robotics[M]. Springer Berlin Heidelberg, 2014.

[10] Raibert M H, Tello E R. Legged Robots That Balance[M]. MIT Press, 1986.