Foot Force Sensor Implementation and Analysis of ZMP Walking on 2D Bipedal Robot with Linear Actuators

Ferdi Perdana Kusumah

Master of Science
Space Engineering - Space Master

Luleå University of Technology
Department of Computer Science, Electrical and Space Engineering
Ferdi Perdana Kusumah

Foot Force Sensor Implementation and Analysis of ZMP Walking on 2D Bipedal Robot with Linear Actuators

School of Electrical Engineering
Department of Automation and Systems Technology
Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology

01.08.2011
Instructor: José-Luis Peralta, M.Sc.
Aalto University
School of Science and Technology

Supervisors: Professor Aarne Halme Professor Kalevi Hyyppä
Aalto University Luleå University of Technology
School of Science and Technology
Acknowledgements

Working on a bipedal robot was one of the most interesting projects that I have ever participated. It required knowledge in several subjects like electronics circuit and programming and without a doubt this project has increased my skills and experiences in the area of robotics.

There are many people that I want to give my gratitude regarding the completion of my thesis. Without any of them, I would never finish my thesis on time. My first gratitude goes to my supervisor José-Luis Peralta for his supports and advices and also for always being present whenever I needed help. I would also like to thank Professor Aarne Halme and Professor Kalevi Hyypää for their positive feedbacks, advices and corrections to increase the quality of my thesis. I will never forget when Professor Hyypää sat together with me debugging my circuit and gave me a lot of useful information. I also want to acknowledge William Martin who corrected my report and my presentation in term of its language and composition.

Many thanks also go to all the institutions and coordinators behind SpaceMaster program especially Department of Automation and Systems Technology (AST) of Aalto University and Lulea Institute of Technology (LTU). I would like to give my thanks to Tomi Ylikorpi for his valuable advices and supports and for taking care of me while I was working on my project, to Anja Hänninen and Annika Salama for coordinating my study and helping me with Aalto University official documents, to Victoria Barabash, Annette Snällfot-Brändström, MariaWinnebäck and Sven Molin from LTU. I would also like to thank Sami Kielosto and Tapio Leppänen for always helping me in building electronics circuits and mechanical constructions for my project.
The supports from my friends were also affected the continuation of my work throughout all semesters in SpaceMaster program, therefore I would like to thank Round 5 SpaceMaster and Erasmus students for their friendship.

Finally I would like to give my gratefulness to my family especially my mom. I know you are always praying for me and wish me good luck in everything I do. This work is for you.

Otaniemi, August 1, 2011

Ferdi Perdana Kusumah

עברית

iii
The objectives of this study were to implement force sensors on the feet of a bipedal robot and analyze their response at different conditions. The data will be used to design a control strategy for the robot. The powered joints of the robot are driven by linear motors. A force sensor circuit was made and calibrated with different kinds of weight. A trajectory generator and inverse kinematics calculator for the robot were made to control the robot walking movement in an open-loop manner. The force data were taken at a certain period of time when the robot was in a standing position. Experiments with external disturbances were also performed on the robot. The ZMP position and mass of the robot were calculated by using the data of force sensors. The force sensor circuit was reliable in taking and handling the data from the sensor although the noise from the motors of the robot was present.

**Keywords:** Bipedal robot, center of pressure, legged locomotion, sensor calibration, strain gage, trajectory generator, zero moment point
Contents

1 INTRODUCTION .................................................................................................................. 1
  1.1 Bipedal locomotion robots ......................................................................................... 2
  1.2 General bipedal walking terminology ...................................................................... 5

2 BIPEDAL WALKING PARADIGMS ........................................................................... 8
  2.1 Static Walking .......................................................................................................... 8
  2.2 ZMP Walking .......................................................................................................... 9
  2.3 Limit Cycle Walking ............................................................................................. 11

3 ZMP WALKING CONTROL APPROACH ..................................................................... 14
  3.1 Robot platform ....................................................................................................... 14
  3.2 Robot sensor .......................................................................................................... 16
  3.3 Walking trajectory generation ............................................................................... 19
  3.4 Walking movement generation ............................................................................. 22
  3.5 Center of Pressure calculation ............................................................................. 27
  3.6 Control system ...................................................................................................... 27

4 ZMP WALKING HARDWARE IMPLEMENTATION .................................................. 28
  4.1 Force sensor design ............................................................................................ 29
  4.2 Circuit schematics and electronics circuits ......................................................... 37
  4.3 Force sensor calibration ....................................................................................... 39
  4.4 Force sensor installation ..................................................................................... 50

5 ZMP WALKING SOFTWARE IMPLEMENTATION ........................................... 53
  5.1 Trajectory generator software ............................................................................ 53
  5.2 Inverse kinematics calculation ........................................................................... 55
  5.3 GIMNet integration ............................................................................................. 60
  5.4 Center of Pressure calculator ............................................................................. 61
6  ZMP DATA ACQUISITION AND ANALYSIS .................................................................63

6.1 Data throughput measurement .............................................................................63
6.2 Standing ZMP measurement ...............................................................................65
6.3 Standing ZMP measurement with disturbance .....................................................74
6.4 Bipedal robot weight measurement .....................................................................79

7  CONCLUSIONS ..................................................................................................89

7.1 ZMP hardware ....................................................................................................89
7.2 ZMP software .....................................................................................................90
7.3 ZMP data ...........................................................................................................90
7.4 Future work ..........................................................................................................91
List of Tables

Table 1. Mass measurement values ................................................................. 43
Table 2. Right leg link-joint parameters ......................................................... 57
Table 3. Left leg link-joint parameters ........................................................... 58
Table 4. Joints angle ranges and initial values of the robot ......................... 59
Table 5. Standing ZMP calculation when motors were turned-off ................. 66
Table 6. Standing ZMP calculation when motors were turned-on ................. 67
List of Figures

Figure 1. Legged robot with two legs (American Honda Motor Co. Inc. 2011) ........... 1
Figure 2. Robonaut R2 from NASA (NASA 2010b) ............................................. 4
Figure 3. Waseda University WL-5 (HRI) ............................................................. 9
Figure 4. Simplified bipedal robot model viewed in several angles (reproduced from Prahlad, Dip & Meng-Hwee 2008) ............................................................... 10
Figure 5. Compass gait bipedal robot model (reproduced from Siciliano & Khatib 2008) ............................................................................................................ 12
Figure 6. Two dimensional state space graph of compass gait model (reproduced from Siciliano & Khatib 2008) .................................................................... 12
Figure 7. The robot platform .................................................................................. 15
Figure 8. Robot platform dimensions ..................................................................... 16
Figure 9. Strain illustration (reproduced from National Instruments Corporation 1998) ........................................................................................................... 16
Figure 10. Strain gage structure (National Instruments Corporation 1998) .......... 17
Figure 11. A Wheatstone bridge circuit schematic ............................................... 18
Figure 12. A walking trajectory illustration ......................................................... 20
Figure 13. The events in human walking process (Vaughan, Davis & O’Connor 1999) ................................................................................................................ 22
Figure 14. Denavit-Hartenberg convention (reproduced from Craig 2005) ..... 23
Figure 15. Translation and rotation of coordinate system B with respect to coordinate system A (reproduced from Craig 2005) ......................................................... 24
Figure 16. Robot manipulator illustration ............................................................ 26
Figure 17. The block diagram of the force sensor hardware .............................. 28
Figure 18. Digital weighing scale force sensor schematics .................................. 30
Figure 19. Force sensor module structures and dimensions ............................ 30
Figure 20. A size comparison between force sensor module and one Euro coin 31
Figure 21. A normal human walking gait vertical force profile (Antonsson & Mann 1985) .............................................................................................................. 32
Figure 22. Frequency domain plot of human walking gait (Antonsson & Mann 1985) ................................................................. 33
Figure 23. An RC low-pass filter schematic .............................................................................................................................. 34
Figure 24. Offset-nulling circuit addition to the sensor bridge (reproduced from National Instruments Corporation 1998) ............................................................................................................................................ 35
Figure 25. Finished electronics boards ................................................................................................................................. 38
Figure 26. Shielded sensor cables .................................................................................................................................................. 38
Figure 27. First calibration platform prototype .......................................................................................................................... 39
Figure 28. Side view of the calibration platform bar with dimensions .................................................................................................. 39
Figure 29. Model of the lever from the calibration platform .......................................................................................................... 40
Figure 30. Exposed view of the first calibration platform problem ................................................................................................ 41
Figure 31. Supporting platform upgrade ....................................................................................................................................... 42
Figure 32. Bucket supporting mechanism and funnel addition ...................................................................................................... 42
Figure 33. Mass measurement comparison .................................................................................................................................. 43
Figure 34. SUCAZULLYDager, the force sensor calibration software .................................................................................................. 46
Figure 35. Calibration plot of force sensor number 1 ....................................................................................................................... 46
Figure 36. Calibration plot of force sensor number 2 ....................................................................................................................... 47
Figure 37. Calibration plot of force sensor number 3 ....................................................................................................................... 47
Figure 38. Calibration plot of force sensor number 4 ....................................................................................................................... 48
Figure 39. Calibration plot of force sensor number 5 ....................................................................................................................... 48
Figure 40. Calibration plot of force sensor number 6 ....................................................................................................................... 49
Figure 41. Calibration plot of force sensor number 7 ....................................................................................................................... 49
Figure 42. Calibration plot of force sensor number 8 ....................................................................................................................... 50
Figure 43. Top view of the foot and the position of each force sensor ............................................................................................... 51
Figure 44. Foot platform dimensions ................................................................................................................................................. 51
Figure 45. Finished product of the foot platforms .......................................................................................................................... 52
Figure 46. SUCAZULLYTractor, the trajectory generator software ..................................................................................................... 54
Figure 47. Robot model, right half of the model with attached coordinate systems and its link descriptions ................................................................................................................................. 56
Figure 48. Left half of the robot skeleton model with attached coordinate systems and its link descriptions ................................................................................................................................. 57
Figure 49. Basic construction of the robot powered joint ................................................................................................................. 60
Figure 50. Top view of the feet and their coordinate system placement .................................................................................................. 62
Figure 51. SUCAZULLYCeptor, the CoP calculator software ........................................................................................................... 62
Figure 79. Foot force and velocity calculations......................................................... 83
Figure 80. Weight measurement comparison between zero and 0.01 m/s disturbance
............................................................................................................................ 84
Figure 81. The movement of CoM in standing position with disturbance applied.... 85
Figure 82. Robot mass measurement when the second disturbance was applied...... 88
Figure 83. Frequency domain graph of the robot mass measurement with 0.04 m/s
disturbance ......................................................................................................... 88
Figure 84. Power supply circuit schematic .............................................................. 98
Figure 85. Low-pass filter circuit schematic ............................................................ 98
Figure 86. Instrumentation amplifier circuit schematic .......................................... 99
Figure 87. ADC and microcontroller circuit schematic ......................................... 100
Figure 88. Sensor holder sketch .......................................................................... 101
Symbols and Abbreviations

\( a_{i,1} \)  
Denavit-Hartenberg distance along x-axis

\( a_{i,1} \)  
Denavit-Hartenberg joint angle parameter about \( x_{i} \)-axis

\( d_{i} \)  
Denavit-Hartenberg distance along z-axis

\( D_X(a) \)  
Translation along x-axis with translation value of \( a \)

\( D_Z(d) \)  
Translation along z-axis with translation value of \( d \)

\( \varepsilon \)  
Strain

\( f_C \)  
Cut-off frequency

\( f_m \)  
Linear motor frequency

\( f_P \)  
Final position

\( f_V \)  
Final velocity

\( F_{x_i} \)  
The \( i \)-th force on x-axis

\( F_{y_i} \)  
The \( i \)-th force on y-axis

\( F_{Ri} \)  
Reaction force of force \( i \)

\( GF \)  
Gage factor

\( i_P \)  
Initial position

\( i_V \)  
Initial velocity

\( \{I\} \)  
Coordinate system of \( I \)

\( l_i \)  
Length of the \( i \)-th link

\( m \)  
Point mass

\( m_H \)  
Hip mass

\( \text{Mul} \)  
Multiplication value of a lever

\( ^{\text{I}}P \)  
A 3 x 1 position vector of the object with respect to \( \{I\} \)

\( P_O \)  
A 3x 1 position vector of the origin of a certain coordinate system

\( \text{R}_i \)  
The \( i \)-th electrical resistance

\( \text{R}_{\text{NULL}} \)  
Offset-nulling electrical resistance

\( \text{R}_{\text{POT}} \)  
Variable electrical resistance

\( R_{X}(\alpha) \)  
Rotation about x-axis with rotation value of \( \alpha \)

\( R_{Z}(\theta) \)  
Rotation about z-axis with rotation value of \( \theta \)
\( A^B R \) Rotation matrix of \( B \) relative to \( A \)

\( s_m \) Linear motor slider length

\( t_m \) Linear motor traveling time in one direction

\( A^B T \) Homogenous transformation matrix of \( B \) relative to \( A \)

\( \theta_i \) Denavit-Hartenberg joint angle parameter about \( z_i \)-axis

\( V_{CC} \) Input voltage

\( V_{Output \ i} \) Output voltage of

\( x_{zmp} \) Position of ZMP on \( x \)-axis

\( y_{zmp} \) Position of ZMP on \( y \)-axis

Aalto University School of Science and Technology

ADC Analog to Digital Converter

AST Automation and Systems Technology

CAN Controller Area Network

CoM Center of Mass

CoP Center of Pressure

DH Denavit-Hartenberg

DLS Damped Least Square

DoF Degree of Freedom

FFT Fast Fourier Transform

GUI Graphical User Interface

ISP In-system Programming

LCW Limit Cycle Walking

LPF Low-pass filter

NASA National Aeronautics and Space Administration

PC Personal Computer

USB Universal Serial Bus

ZMP Zero-Moment Point
Chapter 1

Introduction

Locomotion is the ability of manipulating a body with respect to its environment to move from one place to another (Westervelt et al. 2007). Most of the living beings are using a certain kind of locomotion system to change their position. In robotics field, a locomotion system is used in a mobile robot (a robot that can move freely in the environment (Siciliano et al. 2009)) to move from one point to another according to the commands that are given by the controller. One type of locomotion system is the legged locomotion. This kind of locomotion is characterized by a series of point contacts between the robot and the ground (Siegwart & Nourbakhsh 2004). An example of a legged robot is given in Figure 1.

Figure 1. Legged robot with two legs (American Honda Motor Co. Inc. 2011)
The robot in Figure 1 was developed by Honda Company in Japan and it can be classified as a bipedal robot, which is a subcategory of legged robots.

A bipedal robot is a legged robot that uses open kinematic chain consisting of two subchains called legs and often a subchain called torso as its locomotion system. All subchains are connected at a common point called hip (Westervelt et al. 2007).

1.1 Bipedal locomotion robots

1.1.1 Objective

There are two specific goals for performing this research work:

1. To study the implementation of strain gage sensor and its data acquisitions.
2. To study the Zero-Moment Point (ZMP) Walking paradigm implementation on the GIMBiped robot (bipedal robot with linear actuators) in sagittal plane. ZMP was chosen because it has not been applied practically on this platform. This is the first step before developing more advance control strategies to the robot and can also serve as a comparative study with another paradigm for energy consumption.

The main reasons for choosing bipedal configuration in this research are:

1. A good maneuverability in a rough terrain
   The robot only has a set of point contacts with the ground so the quality of the ground itself does not matter as long as the robot can maintain adequate ground clearance (Siegwart & Nourbakhsh 2004).

2. A good ability in avoiding an obstacle
   A walking robot can avoid an obstacle either by climbing or walking over it as long as the leg clearance of the robot is sufficient to perform the task (Siegwart & Nourbakhsh 2004). Bipedal locomotion can have a better mobility especially in smaller areas compared to multilegged locomotion (Prahlad, Dip and Meng-Hwee 2008).
A robot with legged locomotion can have better mobility in a more general environment and is not restricted to only a flat terrain.

However, there are some problems in implementing legged locomotion in a mobile robot. Power and mechanical complexity are the main issues because the robot legs must be able to sustain the total weight of the robot, and be able to lift or lower its body (Siegwart & Nourbakhsh 2004). Other problems are the system stability and robustness. Stability is related to how the robot can maintain its balance and avoid falling (Hobbelen & Wisse 2007). It can be achieved by applying a control system to handle the movement of joints and links of the legs. Robustness is a measure of sensitivity to parameter variations (Golnaraghi & Kuo 2010). By analyzing robustness, the performance change of the walking process with respect to changing in the system’s parameters and external disturbances can be measured.

1.1.2 Current development

At Aalto University (Aalto) School of Electrical Engineering, The GIMBiped team is designing a biped robot based on linear motors. Their research focuses on decreasing power consumption and increasing the mobility of the robot by using Limit Cycle Walking (LCW) paradigm. Linear motors are used in this case because they do not have gearboxes that can break the natural movements when they are not powered (Peralta et al.).

At Korea Advanced Institute of Science and Technology, Jong-Hwan Kim and his team have designed a compensation scheme for a humanoid robot with ZMP Walking paradigm. The compensation was utilized by applying a compensation angle to each joint involved in the walking motion. This technique was applied to a humanoid robot called HanSaRam (Kim et al. 2003).

In a space application, bipedal locomotion system can be used to create a space robot with good mobility on the surface of a planet. One example of such work is Project M by the National Aeronautics and Space Administration (NASA) in the United States. They built a humanoid robot with bipedal locomotion which is called Robonaut, to
work on the surface of the Moon for 1000 days (NASA 2010a). The picture of the latest version of Robonaut is given in Figure 2.

![Figure 2. Robonaut R2 from NASA (NASA 2010b)](image)

Figure 2 (above) shows only the upper part of the robot. The purpose of Project M is to create a robot as a study platform for a future human space mission. With a human-like body, the robot can provide performance data that can be used to improve design, enhancing crew safety and mission performance in outer space (NASA 2010a).

1.1.3 Future prospect

In future space application, a bipedal robot can be used to assist a human worker in space. Paolo Pierro and his team, from Universidad Carlos III de Madrid and the Hewlett-Packard Italy Innovation Center, have designed a humanoid teleoperation system for space environments. A space worker can use the humanoid robot through teleoperation to do dangerous, difficult or even impossible tasks for human. They use a humanoid robot because it is suitable to collaborate with humans and sharing the same working environments and tools (Pierro et al. 2009). NASA also announced that if the Project M is successful, they will use the robot to prepare future human sites in the Moon and act as a caretaker for their lunar facility (NASA 2010a).
1.2 General bipedal walking terminology

There are several general terminologies related to bipedal walking robot. These terminologies are used in the remaining chapters of this thesis.

1. **Gait**
   Gait is a repetitive pattern of foot placement (Todd 1985 p.55). In the context of this document, the movement belongs to a bipedal robot.

2. **Single support**
   This is a situation when the bipedal robot is supported only by one foot or when the bipedal robot has only one contact surface with the ground (Dekker 2009).

3. **Double support**
   This is a condition when the bipedal robot has two contact surfaces with the ground which means that it is supported by both feet on the ground (Dekker 2009).

4. **Walking**
   Walking is one of the main gaits of a human. It is alternating phases of single and double support, with the requirement that the horizontal displacement of the Center of Mass (CoM) of the robot is strictly monotonic (Westervelt et al. 2007 p.6). There are two basic elements in a walking process, which are a periodic movement of each foot and sufficient ground reaction forces on the stance leg to move the body forward (Vaughan, Davis & O’Connor 1999).

5. **Support polygon**
   This is the contact surface between foot or feet and the ground. In a single support, it is represented by the shape of the foot but in a double support case, the area is the convex hull between two feet (Siciliano & Khatib 2008), which can be obtained by connecting the outer parts of contact surfaces of the feet.
6. **Swing leg**
   A leg that performs a stepping movement in a walking process.

7. **Stance leg**
   A leg that supports the body when the other leg is performing a stepping movement in a walking process.

8. **Sagittal and frontal plane**
   Sagittal plane is the plane which divides the human body into left and right portions. Frontal plane divides the body into front and back portions (Westervelt et al. 2007 p.7).

9. **Statically stable gait**
   A gait where the CoM of the bipedal robot does not move out from its support polygon (Westervelt et al. 2007 p.7).

10. **Quasi-statically stable gait**
    A gait where the Center of Pressure (CoP) of the bipedal robot does not move out from its support polygon (Westervelt et al. 2007 p.7).

11. **Dynamically stable gait**
    A periodic gait where the CoP of the bipedal robot stays in the boundary of the support polygon for at least part of the cycle and yet the robot does not overturn (Westervelt et al. 2007 p.7).

12. **Open-loop control system**
    A control system in which the output is not measured for comparison with the input (Ogata 1997).

13. **Closed-loop control system**
    A control system that maintains a relation between the system’s output and the reference input by comparing them. It then uses the difference to reduce the output error and bring it to a desired value (Ogata 1997).
14. Fully actuated, underactuated and overactuated mechanical model

A mechanical model is defined as fully actuated if the number of independent actuator equals the number of its Degree of Freedom (DoF). If the number of actuators is less than the number of DoF, it is called underactuated. It is called overactuated if vice versa (Westervelt et al. 2007).
Chapter 2

Bipedal Walking Paradigms

In this chapter, three different walking paradigms which are Static Walking, ZMP Walking and LCW will be introduced. The walking paradigms are derived from the stability of a gait. Static Walking is a walking paradigm that uses statically stable gait concept (Westervelt et al. 2007). It uses a movable body so that the CoM can be maintained inside a support polygon (HRI). ZMP Walking uses quasi-statically stable gait and LCW uses dynamically stable gait (Westervelt et al. 2007). Both of them have less restriction of the position of CoM ground projection than Static Walking. The expected results of using the ZMP and LCW are to obtain a better movement and an energy effective walking process (Hobbelen & Wisse 2007). The walking paradigms are used by a human or a robot to walk in different environments and conditions.

2.1 Static Walking

Static Walking uses the robot CoM to maintain its stability. It uses a statically stable principle which means that if the robot motion is stopped at anytime, the robot will be in an indefinitely stable position. When using this paradigm, it is necessary to put the robot CoM projection on the ground inside the support polygon. This walking paradigm has several disadvantages. It requires large feet and strong ankle joints and can only be performed at a slow speed so inertial forces are negligible (Cuevas, Zaldívar & Rojas 2005).
Currently, Static Walking paradigm is still used in a bipedal robot to walk on a slippery surface or to carry a heavy object. It is also used on a robot that has more than two legs (Cappellini 2010; Garciaguirre, Adolph & Shrout 2007; Todd 1985). One of the robots that purely use this Static Walking concept is The Waseda University WL-5, which was developed from about 1970 to 1972. The picture is given below.

![Waseda University WL-5](image)

Figure 3. Waseda University WL-5 (HRI)

As it can be seen on Figure 3, the robot has a moveable body so it can move its CoM on a frontal plane to stabilize the robot (HRI).

### 2.2 ZMP Walking

This concept was introduced by Miomir Vukobratović in 1968. According to Vukobratović and Borovac (2004), ZMP is defined as the point on the ground at which the net moment of the inertial forces and the gravity forces have no component along the horizontal axes. The control of the robot will be based on placing this ZMP. The walking process is considered stable if the point is inside the support polygon at all stages of walking itself. The ZMP will coincide with CoP if only the reaction force at CoP balances all active forces acting on the robot during its movement. In the situation when system dynamics changes and make the ZMP approach support polygon edge, the robot start to rotate at its foot edge (Vukobratović & Borovac...
The ZMP position calculation when it coincides with CoP, can be performed by using simplified robot model as shown in Figure 4 below.

Figure 4. Simplified bipedal robot model viewed in several angles (reproduced from Prahlad, Dip & Meng-Hwee 2008)

From Figure 4, when the robot is lifting one foot for walking, the entire body can be modeled as an inverted pendulum with the robot mass concentrated at CoM (blue ball) connected by a link to the ankle joint of the stance leg (Prahlad, Dip & Meng-Hwee 2008). The reaction or normal forces that acted on the foot surface can be represented by $F_{x1}$, $F_{x2}$, $F_{y1}$, and $F_{y2}$. The position calculations of ZMP in the x-axis and y-axis are as follows,

$$F_{x1}(x_1 - x_{zmp}) - F_{x2}(x_2 + x_{zmp}) = 0$$  \hspace{1cm} (1)

$$x_{zmp} = \frac{F_{x1}x_1 - F_{x2}x_2}{F_{x1} + F_{x2}}$$  \hspace{1cm} (2)

$$F_{y1}(y_1 - y_{zmp}) - F_{y2}(y_2 + y_{zmp}) = 0$$  \hspace{1cm} (3)

$$y_{zmp} = \frac{F_{y1}y_1 - F_{y2}y_2}{F_{y1} + F_{y2}}$$  \hspace{1cm} (4)

The ZMP is the position of the center of reaction forces acting on the foot surface. By using the ZMP Walking paradigm, a walking robot can walk better in terms of speed.
and flexibility than a Static Walking robot because the CoM does not have to be inside the support polygon. But this paradigm still acts poorly in terms of efficiency, disturbance handling and natural walking appearance (Hobbelen & Wisse 2007). An example of a robot that uses this ZMP concept is the ASIMO humanoid robot from Honda (Siciliano & Khatib 2008) that was shown earlier in Figure 1.

2.3 Limit Cycle Walking

This method was first proposed by Hurmuzlu and Moskowitz in 1986. Limit Cycle Walking paradigm is design to improve speed, efficiency, disturbance rejection and versatility compared to the other existing walking paradigms. It imposes fewer artificial constraints to robot walking motion (Hobbelen & Wisse 2007). According to Hobbelen and Wisse (2007), Limit Cycle Walking is a nominally periodic sequence of steps that is stable as a whole but not locally stable at every instant in time.

The robot dynamic model of the paradigm is calculated by using nonlinear dynamics theory. The equations are plotted in a state space graph to analyze its behavior with different initial values. The plot is basically a map between two states that describes the changing state values during the walking cycle (Hobbelen & Wisse 2007). To illustrate this concept, a compass gait walker (a two links planar passive walker with prismatic legs (Westervelt et al. 2007)) is presented in the following example. The walker is illustrated in Figure 5.

The compass gait walker has telescopic knee joints for ground clearance when it walks. The simplified model of the walker is also shown in Figure 5. The total mass of the hip is represented by \( m_H \). The leg mass is represented by point mass \( m \) connected with massless link \( l_1 \) and \( l_2 \). The walker is walking down the slope with \( \gamma \) inclination angle. The derivation of its equation of motions is given in detail in (Goswami, Espiau & Keramane 1996; Siciliano & Khatib 2008). The state space graph of the dynamical system is presented in Figure 6.
Figure 5. Compass gait bipedal robot model (reproduced from Siciliano & Khatib 2008)

Figure 6. Two dimensional state space graph of compass gait model (reproduced from Siciliano & Khatib 2008)
In Figure 6, $\theta$ and $\dot{\theta}$ are the vectors of joint angles and joint angular velocities. The legs were drawn by different colors (blue and white) to distinguish them. The figure shows the state space graph of one leg (blue) which alternately becomes stance leg and swing leg. The graph illustrates the walking cycle of the walker and also it is used to analyze the stability of the walker. The cycle corresponds to a periodic movement and it contains two footsteps of the robot. The figure also shows the illustration of the gait at every stage represented by the model. Ground contact is denoted by a circle with a cross. The state trajectory begins at point 1 when the rear leg becomes the swing leg. At point 2 and point 4, the swing leg is about to touch the ground. Foot switching occurs between point 2 and 3 and point 4 and 1 (Goswami, Espiau & Keramane 1996; Siciliano & Khatib 2008). An example of a robot that is designed to use this paradigm is GIMBiped developed at Aalto School of Electrical Engineering.
Chapter 3

ZMP Walking Control Approach

In this chapter, the idea about how to use the ZMP Walking paradigm on the robot will be presented. GIMBiped was the robot platform used in this work. It was built in the Department of Automation and Systems Technology of Aalto. The robot leg consists of several joints and links connected together. The type of the walking control system that will be applied is the open-loop, which means the main computer will generate a walking movement of the robot without knowing whether the movement is stable or not (the controller does not have a walking data feedback). Linear actuators are used to rotate each active joint to a certain angle to move the foot to a desired position. The walking movement of the robot is pre-calculated by a trajectory generator software. The set of angle needed to make the robot follow the trajectory is calculated by using inverse kinematics. The robot controller will set the angles to each joint and move the robot to the desired position. The ZMP for each leg is calculated by using two sets of force sensors that are attached on the surface of the feet.

3.1 Robot platform

The robot used as a study platform to apply ZMP Walking paradigm has powered joints driven by linear motors. The motors do not have gearboxes, so they allow fluent movements in the walking cycle (Peralta et al.). Each motor is controlled by an independent PID controller. To move the motor, the controller needs a position command that is sent through Controller Area Network (CAN) communication. The
CAN bus was developed by BOSCH Company as a message broadcast system that transmit the data to the entire network (Corrigan 2008). There are two main computers that are responsible for taking the robot sensor data and controlling the robot movement. They are called sensor and motor hub. In this research, only motor hub was utilized. The computer for each hub is PC/104 with a Linux operating system installed. PC/104 is a computer dedicated for embedded application that supports a desktop Personal Computer (PC) architecture (PC/104 Embedded Consortium 2011). The motor hub can be connected to a server on a local network for remote operation. Figure 7 shows the appearance of the robot.

![Figure 7. The robot platform](image)

The name of the robot is GIMBiped. The robot body is attached to a movable platform which also acts as a supporting frame to prevent it from falling. This frame only allows the robot to move in sagittal plane (only forward and backward movement). There are only three powered joints on each robot leg which are hip, knee and ankle joints, so the total powered joints are six. The foot platform that houses the force sensors is attached as an extension without removing the original foot. The dimensions of the robot are given in Figure 8. The dimensions were measured directly on the robot using a measuring tape.
3.2 Robot sensor

The sensors that are used on the foot to collect pressure information are a set of force sensors. The force sensor used for this project is based on a strain gage and it was taken from a digital human weighing scale.

3.2.1 Strain gage

To understand how strain gage work, the concept of strain will be introduced first. Strain is the amount of deformation of a body due to an applied force. Figure 9 shows a body that is pulled by a force in one direction.
Strain is defined as the ratio between the amount of the size change and the original size of the body (National Instruments Corporation 1998). From Figure 9, strain can be written as

\[ \varepsilon = \frac{\Delta l}{l} \]  

(5)

with:
- \( l \): length of the body (m)
- \( \Delta l \): length of the deformation (m)
- \( \varepsilon \): strain of the body

Strain has no unit and it can be positive or negative according to whether the body is extended or compressed. In practice, the magnitude of measured strain is very small and often expressed as microstrain (National Instruments Corporation 1998).

Strain gage is a sensor that is used to measure a strain. The electrical resistance of this device varies in proportion to the amount of strain applied to the device itself. The most widely used gage is the bonded metallic strain gage. The metallic strain gage consists of a very fine wire or a metallic foil arranged in a grid pattern (National Instruments Corporation 1998). The representation of a strain gage is given below in Figure 10.

Figure 10. Strain gage structure (National Instruments Corporation 1998)
From Figure 10, if the carrier is bended, then the electrical resistance between the two outputs will change. The gage factor is a measure of gage sensitivity to a strain. This factor is defined as the ratio between a fractional change in electrical resistance and the strain (National Instruments Corporation 1998) or

\[
GF = \frac{\Delta R / R}{\varepsilon}
\]

With:

- \(GF\) : gage factor
- \(\Delta R\) : electrical resistance change \( (\Omega) \)
- \(R\) : electrical resistance \( (\Omega) \)
- \(\varepsilon\) : strain

\[\text{(6)}\]

### 3.2.2 Wheatstone bridge

A Wheatstone bridge is basically a pair of voltage divider. The two dividers are connected to each other at both ends forming a circuit with two outputs. Figure 11 shows the schematic of the circuit.

![Wheatstone bridge circuit schematic](image)

The output can be calculated using two equations:

\[
V_{Output_1} = \frac{R_2}{R_1 + R_2} V_{cc}
\]

\[\text{(7)}\]
From the equation (7) and (8), $R_2$ is proportional to $V_{Output1}$ and $R_4$ is proportional to $V_{Output2}$. If the value of $R_2$ or $R_4$ becomes smaller, then the $V_{Output1}$ or $V_{Output2}$ becomes smaller and vice versa. If $R_2$ is replaced by a variable resistor and the two outputs are compared, the change in the voltage difference will be proportional to the change in the value of $R_2$. This principle is used for measuring a force with a strain gage (the variable resistor is replaced by the gage).

### 3.3 Walking trajectory generation

In order to make the robot walk properly, a trajectory or a path is needed for the robot to follow. In this thesis, only the lower body part of the robot (from the hip to the foot) was used for conducting the experiment, therefore the trajectories for the upper body parts were not needed.

Three trajectories are needed in the walking process, one trajectory for the hip and two trajectories for the feet. The trajectories are generated by specifying walking characteristics such as:

1. Number of step
2. Time of one step
3. Sampling time
4. Step height
5. Step width
6. Double support percentage
7. Half distance between two legs
8. The height of the leg
9. Initial leg
10. Hip displacement on y-axis
11. Hip displacement on z-axis
12. Initial hip velocity on x-axis
13. Initial hip velocity on y-axis
14. Initial hip velocity on z-axis
15. Final hip velocity on x-axis
16. Final hip velocity on y-axis
17. Final hip velocity on z-axis
18. Leg displacement on y-axis
19. Initial foot velocity on x-axis
20. Initial foot velocity on y-axis
21. Initial foot velocity on z-axis
22. Final foot velocity on x-axis
23. Final foot velocity on y-axis
24. Final foot velocity on z-axis
Using the characteristics given previously, walking points can be specified on a coordinate system and through interpolation technique, each point is connected together to form walking trajectories. Those generated trajectories will become paths for the hip and the legs of the robot to follow. Figure 12 illustrates this process.

![Diagram of a walking trajectory illustration](image)

Figure 12. A walking trajectory illustration

The trajectory generator will generate hip and foot trajectories, which will then be converted to a series of corresponding angles that each joint must follow. The angles will be given to the robot control system to move the robot.

### 3.3.1 Interpolation method

The interpolation method used to generate a walking trajectory is the cubic spline interpolation. This method was chosen because of its advantages compared to the polynomial or linear interpolation method. The advantages are:

1. A smooth interpolation result
2. A third degree polynomial is enough to produce the smooth result

The spline method is basically a polynomial interpolation that is performed between each point specified by trajectory generator (Chevallereau et al. 2009). The
interpolation equation used in the trajectory generator program has the following form,

\[ H = i_p + i_v t + \left( \frac{(3(f_p - i_p) - T(2i_v + f_v))}{T^2} \right) t^2 + \left( \frac{(2(i_p - f_p) - T(i_v + f_v))}{T^3} \right) t^3 \]  

(9)

With:

- \( i_p \): Initial position
- \( f_p \): Final position
- \( i_v \): Initial velocity / gradient
- \( f_v \): Final velocity / gradient
- \( t \): Sampling time / Interpolation step
- \( T \): Maximum time / step
- \( H \): Position of the result in the coordinate system

In the beginning of the calculation, the initial and final position need to be specified in the equation and then by increasing the sampling time value gradually until it reaches maximum time, a series of point that form a curve are produced. The equation tailors the initial and final positions to form a complete trajectory path. A detailed explanation of the derivation of the cubic spline equation can be found in (Press et al. 1992).

### 3.3.2 Walking pattern

The walking pattern that the robot performs is based on human walking cycle. According to (Vaughan, Davis & O’Connor 1999), there are at least 8 events that form the cycle. Those events are shown in Figure 13.

A swing phase period is shorter than a stance phase. In the event of the stance leg, a Heel strike initiates the gait process, followed by a Foot-flat where the surface of the foot touches the ground. A Midstance happens when the swing foot passes the stance foot and at this event the CoM of the body is at its highest position. Heel-off and Toe-off give a body a push for a forward movement and terminate the stance phase. The swing phase is initiated with an Acceleration when the foot leaves the ground and accelerates the swing leg forward. This process is followed by the Midswing when the foot of the swing leg passes the body and the Deceleration to slow down the swing leg movement for stance leg phase preparation (Vaughan, Davis & O’Connor 1999 p.10-11).
The parameters of the events in the walking process will be used to determine the walking characteristics mentioned previously to produce the walking movement of the robot.

### 3.4 Walking movement generation

To make the leg of the robot follow the generated trajectory, a set of angles for each joint on the leg needs to be calculated. The angles will be sent to the robot controller to move each joint of the leg and make the foot reaches a certain position, in this case the trajectory. The angles can be obtained by inverse kinematics calculation. In this method, a homogenous transformation matrix is needed, which maps the position of the foot with respect to the position of a certain reference. The CoM of the robot is used as this reference.

#### 3.4.1 Denavit-Hartenberg convention

Denavit-Hartenberg (DH) convention is a set of rules to attach a coordinate system to each joint for a transformation calculation between two joints. To characterize a link
and joint pair on a robot manipulator, at least four parameters are needed. Two parameters describe the corresponding link, and the other two describe the connection of the link to a neighboring link (Craig 2005). The robot contains only revolute joints and the convention to calculate the transformation is given in Figure 14 below.

![Figure 14. Denavit-Hartenberg convention (reproduced from Craig 2005)](image)

The description of the convention is as follows (Craig 2005):

- $a_{i-1}$: The distance from $z_{i-1}$-axis to $z_i$-axis measured along $x_i$-axis.
- $a_{i-1}$: The angle from $z_{i-1}$-axis to $z_i$-axis measured about $x_{i-1}$-axis.
- $d_i$: The distance from $x_{i-1}$-axis to $x_i$-axis measured along $z_i$-axis.
- $\theta_i$: The angle from $x_{i-1}$-axis to $x_i$-axis measured about $z_i$-axis.

The convention above is used to construct a transformation matrix, which is called homogenous transformation that maps the CoM of the robot to its foot. The CoM was chosen as a reference to make the transformation calculation simpler for each foot, because the convention can be made identical for both feet.
3.4.2 Homogenous transformation

The homogenous transformation is a 4 x 4 matrix used to describe one coordinate system relative to the other. The matrix consists of rotation and translation matrix (Craig 2005). Figure 15 below shows the relation between two coordinate systems.

![Figure 15. Translation and rotation of coordinate system B with respect to coordinate system A (reproduced from Craig 2005)](image)

The descriptions of Figure 15 above are given below (Craig 2005):

- \( \{A\} \) : Coordinate system of A.
- \( \{B\} \) : Coordinate system of B.
- \( ^A P \) : A 3 x 1 vector that define the position of the object with respect to \( \{A\} \).
- \( ^B P \) : A 3 x 1 vector that define the position of the object with respect to \( \{B\} \).
- \( P_O \) : A 3 x 1 vector that define the position of the origin of \( \{B\} \) relative to \( \{A\} \).
- \( \alpha \) : The angle from \( Y_A \)-axis to \( Y_B \)-axis measured about \( Z_A \) or \( Z_B \)-axis.

The transformation equation between two coordinate systems is formed by vector and matrix equations. The form of the equation is given below.

\[
\begin{bmatrix}
^A P \\
1
\end{bmatrix} = \begin{bmatrix}
^A T
\end{bmatrix} \begin{bmatrix}
^B P \\
1
\end{bmatrix}
\]  
(10)
$$A^B_T = \begin{bmatrix} A^B_R & P_O \\ 0 & 0 & 0 & 1 \end{bmatrix}$$  \hspace{1cm} (11)

With:

$A^B_T$ : Homogenous transformation that maps rotation and translation of $\{B\}$ relative to $\{A\}$.

$A^B_R$ : A $3 \times 3$ rotation matrix that maps rotation of $\{B\}$ relative to $\{A\}$.

In robot manipulator case, the homogenous transformation matrix is formed using multiplications between translational and rotational operator. Translational operator is basically a homogenous transformation that does not have a rotation matrix component. The rotation matrix is replaced by identity matrix. Rotational operator is a homogenous transformation that has only its rotation matrix without a translation vector in the fourth column. The translation vector is replaced by zero values vector (Craig 2005). If the translational operator along x-axis with translation value of A is defined as $D_X(A)$ and rotational operator at x-axis with rotation angle value of B is defined as $R_X(B)$, then from the DH parameters given previously, four different operators can be constructed, which is:

$R_X(a_{i-1})$ : Rotation about x-axis

$D_X(a_{i-1})$ : Translation along x-axis

$R_Z(\theta_i)$ : Rotation about z-axis

$D_Z(d_i)$ : Translation along z-axis

By using the four operators above, the transformation of link $i$ with respect to link $i-1$ can be derived as follows:

$$i^{\text{-1}}T = R_X(\alpha_{i-1})D_X(a_{i-1})R_Z(\theta_i)D_Z(d_i)$$ \hspace{1cm} (12)

$$i^{\text{-1}}T = R_Z(\theta_i)D_Z(d_i)R_X(\alpha_{i-1})D_X(a_{i-1})$$ \hspace{1cm} (13)

The equations are based on Euler angle multiplication order. The detailed information on Euler angle can be found in Craig (2005). The equation (12) is used if the $i^{\text{th}}$ coordinate system is rotated on x-axis first and then z-axis afterwards with respect to ($i-1)^{\text{th}}$ coordinate system. The equation (13) is used if the rotation is vice versa (Craig 2005).
The transformation equation is calculated for each link on the robot from the robot CoM to the robot foot. The result of the calculation is used to compute the inverse kinematics.

3.4.3 Inverse kinematics

In a robot manipulator, an inverse kinematics is a method to calculate a set of angles of the joint of the robot to make the end-effector (the tip of the manipulator) arrives at certain position (Craig 2005). The illustration is given in Figure 16.

By using inverse kinematics, the angles for joint 0, 1 and 2 can be obtained to move the end-effector to a certain position. In a bipedal robot, this method is necessary to make the foot (end-effector) follow the desired trajectory. There are several techniques of inverse kinematics that can be used to produce the set of angles. One of them is Damped Least Square (DLS) method which was chosen for this thesis work.

The calculation of this method is done by incrementing each powered angle in the robot until the end-effector reaches (is close to) the target position. The increment angles are calculated through an equation involving a Jacobian matrix computation. This method can prevent oscillations in its result if the target is unreachable or the leg is fully extended (Buss 2009). Further analysis and derivation of the method can be
found in Buss (2009). To make the calculation of DLS faster, only powered joints are involved in the Jacobian matrix computation.

### 3.5 Center of Pressure calculation

The calculation of ZMP when the point coincides with Center of Pressure (CoP) is performed by computing the CoP itself from four force sensors installed on the foot. To calculate it, the equations (1 to 4) given previously in *ZMP Walking* section were used. The movement of this point due to the changes of the robot posture will be analyzed. The result will be further used to determine what kind of control system should be applied to maintain the stability of the robot.

### 3.6 Control system

The control system applied to the robot walking process is an open-loop control. The walking movement of the robot is generated by a computer program. The movement is based on a walking pattern explained previously. The software is only responsible of sending the data without analyzing the response or feedback from the robot. It also does not use the ZMP data and compare it with desired ZMP values to produce movement compensations to reduce the ZMP error. Therefore the stability of the walking movement is not guaranteed.
Chapter 4

ZMP Walking Hardware Implementation

This chapter will explain about the process of designing and making the electronics circuits to acquire data from the force sensors. A general block diagram of the hardware system is given in Figure 17 below.

From Figure 17, it can be seen that the force sensors outputs are amplified by the instrumentation amplifier because the changes in the signals are very small (in the order of millivolt). Instrumentation amplifier is a differential voltage amplifier that has high gain and low noise characteristics. This type of amplifier is generally used to amplify weak sensor signal (Whitaker 2005). A low-pass filter is used to remove high frequency noise coming from the circuit and the environment. Other noise removal strategies and sensor compensations are also presented in this section to improve the circuit performance. The analog signal from the filter is sent to Analog to Digital Converter (ADC). This ADC converts analog signal to digital data and sends the data.
to the microcontroller. The microcontroller transmits the data to a computer through a serial communication for further processing and analysis.

4.1 Force sensor design

The sensors used on the foot to collect foot pressure information are basically force sensors taken from a digital human weighing scale. The original scale is formed by four force sensors to calculate a human mass. The reason for taking the sensor directly from the scale was because it simplified the sensor design (the sensor did not have to be built from scratch). To build a fully working force sensor, sensor holders and electronics circuits are needed to collect its data. The AST Department of Aalto has the facilities to build the holder and the circuit.

4.1.1 Force sensor

The force sensor has three cables (red, white and black) as outputs. It was assumed that this sensor consists of a half Wheatstone bridge with one common connector. If the three outputs are measured with an ohmmeter, the results are:

- Red and white : 1 KΩ
- White and black : 1 KΩ
- Red and black : 2 KΩ

The measurement results verify the assumption that the sensor contains resistive materials connected in series with a common connection at white cable. The schematic of the sensor is given in Figure 18.

If a force is applied on the sensor, only the value of resistor made of piezoresistive material (in this case the $R_1$ value will decrease if the force applied to the sensor) will change. A piezoresistive material is a material that can change its electrical resistance if a strain is applied on it (National Instruments Corporation 1998). The other resistive material is for temperature compensation purpose because when the sensor is heated, the electrical resistance proportion between $R_1$ and $R_2$ remains the same.
4.1.2 Force sensor holder

To attach the sensors to the robot feet correctly, mechanical holders have been manufactured. The design was drawn and then given to the mechanical workshop in AST Department building for manufacturing. The drawing of the holder is given in Appendix section.

On the bottom surface of the foot, four sensors were placed for force measurement, therefore eight holders were needed for both feet. The holder dimensions and finished product are given in Figure 19 below.
The holder consists of three layers and one rubber protector (Figure 19). It was manufactured on aluminum material because of its strong and light characteristics. The mechanical workshop also has the tools to process the material. The dimensions were made as small as possible to remove unnecessary weight on the robot. Figure 20 gives a size comparison between the force sensor and a one Euro coin.

![Figure 20. A size comparison between force sensor module and one Euro coin](image)

### 4.1.3 Instrumentation amplifier

Instrumentation amplifier is used to multiply the output from the force sensors. It was needed to multiply the sensor output 1000 times in order to be able to detect a voltage difference between Wheatstone bridge outputs contained in the force sensor. There were eight force sensors that need to be connected to the robot’s feet, so eight instrumentation amplifiers were needed for the signal amplification. The INA2126 instrumentation amplifier IC from National Semiconductor was chosen as the amplifier because of its ease of use. The device has two amplifiers, so four of them were needed for the circuit. A detailed explanation of the IC can be found in Texas Instruments Inc. (2005).
4.1.4 Analog to Digital Converter and microcontroller

The Analog to Digital Converter (ADC) needs to be able to sample the foot force correctly. Therefore, a frequency analysis of the walking gait is necessary. According to the research by Antonsson & Mann (1985), the maximum frequency of the force on a human walking gait is 100 Hz. The walking vertical force profile and its FFT are given in Figure 21 and 22.

![Typical Vertical Forceplate Data](image)

Figure 21. A normal human walking gait vertical force profile (Antonsson & Mann 1985)

Shannon sampling theory (Shannon 1949) says that, in order to reproduce a signal with a maximum frequency of $F$, a sampling frequency greater than $2F$ is needed. In this case the ADC sampling frequency must exceed 200 Hz.

The ADC used here was the MCP3208 from Microchip. It is an 8 channels and 12-bit ADC with Serial Peripheral Interface (SPI) data communication capability. Its maximum sampling rate is 100 kilosamples (ksps) for 5V input supply voltage and 50 kcps for 2.7V input (Microchip 2002). The maximum sampling rate of this ADC at 2.7V is more than enough to sample foot force signal. The ADC will convert the
voltage level in the instrumentation amplifier output to a digital number representation and send it to a microcontroller through SPI interface.

![FFT of Forceplate Data](image)

Figure 22. Frequency domain plot of human walking gait (Antonsson & Mann 1985)

A Teensy++ 2.0 was used to collect data from the ADC. Teensy++ 2.0 is an electronics development board containing AT90USB1286 microcontroller from ATMEL. It was chosen because of its ease of use and capability of In-system Programming (ISP) through a Universal Serial Bus (USB) connector. ISP is a feature to program the microcontroller without removing it from the main electronic circuit. The microcontroller sends digital force data to a computer through a serial communication. Detailed descriptions for the ADC and microcontroller can be found in Microchip (2002) and PJRC. This microcontroller transmits the sensor digital data in a series of strings, starting from sensor 1 to sensor 8 separated by a ‘#’ mark. The minimum number of the transmitted character is 17 and the maximum number is 41.

### 4.1.5 Low-pass filter

A Low-pass filter (LPF) was designed to remove high frequency noise that occurred on the output of instrumentation amplifier. This circuit is needed because the noise is also amplified by the amplifier. The circuit is basically an RC filter (filter which is
composed by resistor and capacitor) and it will only pass the signal below a certain frequency. The schematic of the circuit is given in Figure 23.

![RC low-pass filter schematic](image)

*Figure 23. An RC low-pass filter schematic*

The instrumentation amplifier output is connected to $V_{\text{Input}}$ and $V_{\text{Output}}$ is connected to the ADC input pin.

To apply the filter to the circuit, the designer has to be aware of the signal frequency that wants to be measured. In this research, the corresponding signal is the walking vertical force output voltage from instrumentation amplifier. From Figure 22, it can be seen that the dominant frequency of the gait lies within the range of 0 Hz to about 6 Hz. According to this information, a cut-off frequency of the filter was calculated. It can be calculated by using

$$f_c = \frac{1}{2\pi RC}$$

(14)

With $f_c$ in *Hertz*, $R$ and $C$ are the values of resistor and capacitor in *Ohm* and *Farad* (Horowitz & Hill 1989). The cut-off frequency must be above significant frequencies of the gait. The value of $R$ and $C$ were chosen to be 270 KΩ and 0.1 μF to produce the cut-off frequency on 5.895 Hz, which can eliminate all unnecessary high frequency signals.

### 4.1.6 Sensor compensation

There were two types of compensation that were applied to the force sensor module. The first one is bridge compensation and the second one is temperature compensation. The bridge compensation was needed because the outputs of the Wheatstone bridge
are very unlikely to be equal when the sensor is not pressed (National Instruments Corporation 1998). This problem can be solved in hardware or in software. Hardware solution was chosen because it can also deal with negative offset value. Temperature compensation is needed because the sensor is very sensitive to temperature changes. The temperature fluctuation can affect the value of the resistance in the bridge and change the bridge output voltage. The compensations are needed to reduce the reading error of the sensor.

Offset-nulling circuit is used to compensate the bridge. The circuit consists of one resistor ($R_{NULL}$) and potentiometer ($R_{POT}$) connected to the sensor bridge (National Instruments Corporation 1998). The schematic of the circuit is given in Figure 24.

![Offset-nulling circuit addition to the sensor bridge](image)

Figure 24. Offset-nulling circuit addition to the sensor bridge (reproduced from National Instruments Corporation 1998)

By varying the value of $R_{POT}$, the $V_{OUTPUT1}$ can be tuned equally as $V_{OUTPUT2}$ when there is no force applied to the sensor. The $R_{NULL}$ value determines the range of the offset that the circuit can balance. This tuning was done to make the instrumentation amplifier output become zero at that condition (National Instruments Corporation 1998).

Temperature compensation was needed to balance the two outputs of the bridge. As was mentioned before, the sensor has another resistor or a gage for temperature
compensation but this is not enough. One sensor only constructs the half of the bridge so the other half of the bridge still needs to be compensated to make the whole bridges温度 changes equally. To solve this problem, another sensor was used for the other half. This other sensor is only used as a dummy sensor for temperature compensation, not as a force measurer.

Every resistor belonging to a certain instrumentation amplifier was also joined together to reduce temperature variation between them. The main goal was again to prevent the sensor reading error.

4.1.7 Other noise reduction techniques

There are other noise reduction techniques that were applied to the sensor main circuit to increase its performance besides applying the LPF. The techniques are listed below.

1. **Cable shielding**
   
The purpose of the shielding is to remove a capacitive coupling because of a stray capacitance that occurs between two conductors (i.e. cables). This coupling can introduce a noise voltage to the circuit connected to the affected cable. In this case the noise voltage is directly proportional to the resistance of the affected circuit to the ground and the stray capacitance value. By reducing those two values, the noise voltage can be reduced (Ott 2009).

   Cable shielding can remove the effect of stray capacitance if properly used. The cable should be shielded entirely which means the part that extends beyond the shield has to have as minimum length as possible. The shield must also be connected to a good ground (Ott 2009).

   In this project, the shield is applied to the cables that connect the sensor to the instrumentation amplifier.
2. **Twisted cables**
   This method is done to reduce the effect of inductive coupling. The coupling occurs because of a magnetic field that is generated when there is a current flowing through a conductor. This field will interact with a nearby circuit and can introduce noise voltage. By twisting the cables, the magnetic field produced by them will cancel each other (Ott 2009). All sensor cables in the circuit were twisted to prevent this inductive coupling.

3. **Ground plane**
   Transient ground currents are the main source of noise in a digital circuit. The currents are produced whenever a digital logic switches. To minimize the noise, a low impedance ground system is needed. This can be done by providing as many alternative ground paths as possible. The ground plane can solve this issue (Ott 2009). Every circuit board that was made in this project is equipped with the ground plane.

4.2 **Circuit schematics and electronics circuits**

The schematics of the circuit are given in Appendix section. The schematics and Printed Circuit Board (PCB) layouts were made by Altium Designer Software. A chemical etching method was used to manufacture the PCBs because the equipment is provided in AST Department. Etching is the process of selective removal of copper from all undesirable areas on the raw PCB board to achieve desired circuit pattern (Khandpur 2006). The ferric chloride was used as an etchant or a chemical substance to etch the copper. A detailed explanation of this manufacturing method can be found in (Khandpur 2006). The finished electronics boards are shown in Figure 25.

Four electronics boards were made for this project, which are a power supply (circuit A), an instrumentation amplifier (circuit B), a low-pass filter (circuit C) and a combined microcontroller and ADC board (circuit D). The cables that connect the circuit to the sensors had to be shielded to avoid noise signal addition to the sensor outputs. The shielding result can be seen in Figure 26.
The shielding material is aluminum foil that is commonly used for cooking. All sensor cables are shielded to reduce the noise.

Figure 25. Finished electronics boards

Figure 26. Shielded sensor cables
4.3 Force sensor calibration

The data received by the microcontroller from ADC are in the form of a count not a mass value. In order to know the mass representation of the data, each sensor has to be calibrated. The result of the calibration is a set equation that maps a count to a mass value.

4.3.1 Calibration platform

To apply different weights to the sensor, a mechanical platform was built. This mechanical platform needs to be able to carry a mass up to 30 kilograms. A lever system was chosen as the mechanical platform and it was manufactured in the workshop of the AST Department. The first design and its bar dimension of the platform are presented in Figure 27 and 28.
From Figure 27 and 28, point A is a connection between the metallic bar and the supporting platform, point B is a contact between the metallic bar and a force sensor module, point C is a place where different weights for calibration are placed. A bended wire was used at point C connected to a water bucket. Water was used as the variable weight to calibrate each sensor. The calibration platform will act as a lever to multiply the weight that is put at point C. The lever equation can be derived based on Figure 29 below.

\[
\begin{align*}
F_1 r_1 + F_2 r_2 &= rF \\
F &= \frac{r_1 F_1 + r_2 F_2}{r}
\end{align*}
\]

If point A is used as a zero reference, the equations are as follows:

\[
\begin{align*}
r_1 F_1 + r_2 F_2 &= rF \\
F &= \frac{r_1 F_1 + r_2 F_2}{r}
\end{align*}
\]

\(F_1\) represents the force applied at point A, \(F\) is the force at point B and \(F_2\) is the force at point C. The distance of the force from zero reference is represented by \(r_1\), \(r\) and \(r_2\). If the dimension values are inserted into the second equation \((r_1 = 0\, \text{cm}, \ r_2 = 50\, \text{cm}, \ r = 10\, \text{cm})\), the results are

\[
F = \frac{0 + 50F_2}{10}
\]

\[
F = 5F_2
\]

So, if a weight is hanged at the end of the lever (point C), the lever will multiply it by 5.
4.3.2 Calibration platform upgrade

During the experiment, there were some problems on the first calibration platform design. The supporting platform was not strong enough and could easily be bent. The consequence was the difficulty to make the bar parallel to the ground since the wooden block height was fixed.

The problem in Figure 30 made the force calculation at point B more complicated because \( \beta \) angle value was needed and unfortunately it was difficult to measure it. Another problem was a motion of the bucket. The bucket was left hanging on the wire connected to the metallic bar. This moving bucket made the data output from the sensor change according to the bucket swinging frequency. Finally, the last problem was the wave on the surface of the water when the water was poured into the bucket. This wave contributed to data output noise because of the amplification of the instrumentation amplifier. The large amplification makes the sensor very sensitive to external disturbance.

Several changes were made to the platform to reduce the effect of the problems mentioned previously. The fundamental changes on the platform are the following:

1. The supporting platform material was changed from a thin aluminum to a thick plastic (Figure 31). This technique prevented the bending and avoided wrong mass calculation.
2. The bucket was supported to prevent it from swinging (Figure 32). The supporter was taken from a chair.

3. To reduce the surface wave on the water inside the bucket when it was poured, a small rubber hose connected to a funnel was attached on the platform as a water transport mechanism (Figure 32).

Figure 31. Supporting platform upgrade

Figure 32. Bucket supporting mechanism and funnel addition
4.3.3 Actual calibration platform multiplication

The actual multiplication of the platform had to be checked because it was not made ideally. This is done by comparing a mass of an object when it is measured directly with a scale and if it is measured using a combination of the calibration platform and the scale. The measurement processes is shown in Figure 33 below.

Figure 33. Mass measurement comparison

Figure 33 illustrates the measurement with the calibration platform (the picture at the top) and with the scale only (the picture at the bottom). The measurements were done ten times and then the results were averaged. The objects that were used as the mass were three pieces of hexagonal nut. The measurement data are given in Table 1.

Table 1. Mass measurement values

<table>
<thead>
<tr>
<th>No</th>
<th>Three pieces of nut (gr)</th>
<th>Calibration platform (gr)</th>
<th>Three pieces of nut and calibration platform (gr)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>66</td>
<td>1175</td>
<td>1470</td>
</tr>
<tr>
<td>2</td>
<td>66</td>
<td>1175</td>
<td>1470</td>
</tr>
<tr>
<td>3</td>
<td>66</td>
<td>1180</td>
<td>1470</td>
</tr>
</tbody>
</table>
The multiplication value can be calculated using the following equation:

\[ \text{Mul} = \frac{\text{Three nuts mass and platform} - \text{Platform}}{\text{Three nuts mass}} \quad (19) \]

\[ \text{Mul} = \frac{1471.5 - 1178}{65.4} \quad (20) \]

\[ \text{Mul} = 4.488 \quad (21) \]

The multiplication value (Mul) in the equation (21) was used in the calibration process.

### 4.3.4 Calibration method

The calibration process was done by applying different weights to each force sensor module used under both feet. Water was used as a weight source because its mass can be changed by adding or removing the water itself. The maximum weight that can be applied to each sensor was determined from the maximum mass value that the digital weighing scale (where the sensors were taken) could hold. The maximum mass that the scale could handle was **150 kilogram** (kg). That means each sensor contributes to 150/4, which is **37.5 kg**. The bucket mass used to hold the water was measured 10 times with the same scale to measure the calibration platform multiplication. The average value of the bucket mass was **266 gr.** The maximum water mass was **7 kg**, therefore the maximum mass applied to the calibration platform was **7.266 kg** (the water and the bucket). The maximum mass applied to the sensor was **7.266 x 4.488 = 32.670 kg**.
The data were taken 500 times with the microcontroller data throughput as a sampling time. The masses applied to each sensor were 0 to 7 kg with 0.5 kg interval value, so there were 15 different values of weight. A 500 ml beaker was used to pour water into the bucket. The steps of the calibration method are as follows:

1. Put the bucket on the lever and lift the lever so it has a room for the sensor module.
2. Put the sensor module on the calibration platform and then support or hang the lever so it does not touch the module. Take the data 500 times and average them. This data correspond to 0 kg mass.
3. Put the lever on top of the module and then pour the water into the bucket through the funnel and rubber hose with 500 ml beaker.
4. Wait about 3 minutes until the water surface less wavy and then take the data 500 times and average them.
5. Repeat step 3 and 4 until the water inside the bucket reach 7 liter (7 kilogram).
6. Remove the mass by emptying the bucket and replace the sensor module with another module. Repeat step 2 to 6 until all sensor modules data are acquired.

The masses data were then added with the bucket mass for the calibration equations calculation.

4.3.5 Calibration software

A calibration software was made to average ADC data that were received by the computer from a microcontroller. The GUI is shown in Figure 34. The user can set the amount of data taken and time interval of taking and processing the data from computer buffer. Set mass option is only used to print the mass value to an output file. When the Start Averaging button is pressed, the software opens a connection to serial device and starts taking data that are sent from the microcontroller to computer buffer. After data acquisition process is finished, the data is averaged and displayed beside Average result label.
4.3.6 Calibration results

The acquired data from the calibration process were plotted and are presented in Figure 35 to 42 below.

![Figure 35. Calibration plot of force sensor number 1](image)

Figure 35. Calibration plot of force sensor number 1
Figure 36. Calibration plot of force sensor number 2

\[ y = 3E-07x^2 + 0.0092x - 0.2571 \]

Figure 37. Calibration plot of force sensor number 3

\[ y = 3E-07x^2 + 0.0091x - 0.2608 \]
Figure 38. Calibration plot of force sensor number 4

Figure 39. Calibration plot of force sensor number 5
Figure 40. Calibration plot of force sensor number 6

Figure 41. Calibration plot of force sensor number 7
The equation for each data series were obtained by using polynomial regression technique. The eight equations were used to map a force sensor data output to a certain mass.

4.4 Force sensor installation

In order to put the force sensors on the robot feet, a pair of foot platform had to be made. The platforms were manufactured by the machine workshop of AST Department in Aalto.

4.4.1 Foot platform

The foot platform was designed to be added to the existing robot foot, so another robot foot did not have to be made. The platform was made from aluminum material because of its strong and easy to manufacture properties. Every force sensor was labeled by a number from 1 to 8 for identification purpose according to the order of
transferred data from the microcontroller. The position of the sensor and the dimension of the platform are given in Figure 43 and 44.

Figure 43. Top view of the foot and the position of each force sensor

Figure 44. Foot platform dimensions
From Figure 44, the blue plates are the foot connectors, the grey plate is the foot base and the red blocks are the force sensors. The foot was designed as narrow as possible in width because the distance between the legs of the robot is small (only 24.2 cm). The sensor module is placed exactly at the edge of the platform. The finished foot product with the sensors attached to it can be seen on Figure 45 below.

Figure 45. Finished product of the foot platforms

From Figure 45, on the left side of the picture is the bottom side and on right side is the top side of the platform. The strain gages that are placed on top of the platform are used for temperature compensation.
Chapter 5

ZMP Walking Software Implementation

In this chapter, the software used in this ZMP Walking experiment will be explained. Three software were made for this project. The first one is used in sensor calibration to average force data and was explained in Force Sensor Calibration section in Chapter 4. The remaining are the trajectory and inverse kinematics calculator, and the CoP calculator. These two programs will be used side by side to generate a walking movement for the robot and record its CoP position.

5.1 Trajectory generator software

This software generates walking trajectories for the hip and both feet. The walking pattern that was introduced in Walking trajectory generation section in Chapter 3 needed to be changed because of the mechanical limitations of the robot.

5.1.1 Walking pattern modification

In ZMP Walking paradigm, the robot foot must always remain parallel to the ground, which means it is not a heel-to-toe walking process like in the LCW case. In the GIMBiped platform, the capability for the knees to support its entire body is limited. The knees motor can overheat in a short period of time if the load is too heavy (e.g. standing with knee bends position). The modification for the walking pattern was done in the swing leg phase. The maximum distance that the swing foot can travel in a forward direction on a sagittal plane is the position of the hip. The stance leg is the
only component responsible for the robot propulsion. The modifications decrease the complexity in generating a trajectory for the software and also decrease the load on the knees.

### 5.1.2 Main program interface

The software was made in Linux using qt from Nokia Corporation (2011). The software name is SUCAZULLLYTractor and its Graphical User Interface (GUI) is shown in Figure 46.

![Figure 46. SUCAZULLLYTractor, the trajectory generator software](image)

The interface contains all the walking parameters that can affect the generated trajectory. Four of the buttons are made unavailable in the beginning to avoid false execution. To generate a trajectory the user has to push `ReadConfig` button first to load default parameters from configuration file. If the command is successful, the `StartCalculation` button will be activated. If the `StartCalculation` button is pressed, the software will calculate a trajectory based on the parameters and also calculate the inverse kinematics of each point in the trajectory. The `Connect` button will then be available if the calculations were successful. The `Connect` button is used to connect
the software to GIMNet server. The *Test* and *SendCommand* button are used to send a test command and calculated data to the server.

### 5.1.3 Configuration file

This configuration file is an XML file that contains default parameters (i.e. DH parameters, walking parameters and robot dimensions parameter) used for trajectory and inverse kinematics calculation, including some parameters that were given in *Walking trajectory generation* section previously. The file also contains basic connection parameters for opening a communication to a GIMNet server. When the SUCAZULLYTractor program starts, it will ask the user to load this file.

### 5.2 Inverse kinematics calculation

To obtain the set of angles that the robot leg will follow, an inverse kinematics calculation has to be performed on each hip and foot trajectory point pair. The resulting angles will then be converted to stroke (linear motor position information) and sent to the GIMNet server to move the robot.

#### 5.2.1 Robot initial joints and links parameters

A set of angles and links parameters have to be set for the initial position of the robot. The rule of putting a coordinate system on a joint is based on DH convention that was explained previously in Chapter 3. The robot models with the attached coordinate systems are given in Figure 47 and 48.

These two models are made based on the general structure of a human leg with a reduced DoF on the hip joint (the hip only has 2 DoF). From Figure 47 and Figure 48, four link and joint parameters (to characterize the relation between each link-joint pair) are given in Table 2 and 3.
Figure 47. Robot model, right half of the model with attached coordinate systems and its link descriptions
Figure 48. Left half of the robot skeleton model with attached coordinate systems and its link descriptions

Table 2. Right leg link-joint parameters

<table>
<thead>
<tr>
<th>$i$ (joint number)</th>
<th>$a_{i-1}$ (meter)</th>
<th>$\alpha_{i-1}$ (degree)</th>
<th>$d_i$ (meter)</th>
<th>$\theta_i$ (degree)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>-90º</td>
<td>0</td>
<td>90º</td>
</tr>
<tr>
<td>2</td>
<td>$l_1$</td>
<td>0º</td>
<td>0</td>
<td>$\theta_2$</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>90º</td>
<td>0</td>
<td>$\theta_3$</td>
</tr>
<tr>
<td>4</td>
<td>$l_2$</td>
<td>0º</td>
<td>0</td>
<td>$\theta_4$</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>-90º</td>
<td>0</td>
<td>$\theta_5$</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0º</td>
<td>0</td>
<td>$\theta_6$</td>
</tr>
<tr>
<td>7</td>
<td>$l_3$</td>
<td>0º</td>
<td>0</td>
<td>$\theta_7$</td>
</tr>
<tr>
<td>8</td>
<td>$l_4$</td>
<td>0º</td>
<td>0</td>
<td>$\theta_8$</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>90º</td>
<td>0</td>
<td>$\theta_9$</td>
</tr>
</tbody>
</table>
Table 3. Left leg link-joint parameters

<table>
<thead>
<tr>
<th>i (joint number)</th>
<th>( a_{i-1} ) (meter)</th>
<th>( a_{i-1} ) (degree)</th>
<th>( d_{i} ) (meter)</th>
<th>( \theta_{i} ) (degree)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>-90º</td>
<td>0</td>
<td>90º</td>
</tr>
<tr>
<td>2</td>
<td>( l_{1} )</td>
<td>0º</td>
<td>0</td>
<td>( \theta_{2} )</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>90º</td>
<td>0</td>
<td>( \theta_{3} )</td>
</tr>
<tr>
<td>4</td>
<td>( l_{2} )</td>
<td>0º</td>
<td>0</td>
<td>( \theta_{4} )</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>-90º</td>
<td>0</td>
<td>( \theta_{5} )</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0º</td>
<td>0</td>
<td>( \theta_{6} )</td>
</tr>
<tr>
<td>7</td>
<td>( l_{3} )</td>
<td>0º</td>
<td>0</td>
<td>( \theta_{7} )</td>
</tr>
<tr>
<td>8</td>
<td>( l_{4} )</td>
<td>0º</td>
<td>0</td>
<td>( \theta_{8} )</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>90º</td>
<td>0</td>
<td>( \theta_{9} )</td>
</tr>
<tr>
<td>10</td>
<td>( l_{5} )</td>
<td>90º</td>
<td>0</td>
<td>90º</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>-90º</td>
<td>0</td>
<td>0º</td>
</tr>
</tbody>
</table>

The value for link parameters (\( l_{1} \) to \( l_{5} \)) can be seen on Figure 8 in Chapter 3. Beside the four parameters of Table 2 and 3, the angle range and initial value for \( \theta_{2} \) to \( \theta_{9} \) had to be defined. The angles range determines the minimum and maximum reachable angle’s value for each moving joint. Initial values for each joint were also needed to define the robot position in its initial walking process, which was the standing position in this case. The angles range and initial values for all joints are given in Table 4.

There are only three moving joints for each leg on the GIMBiped robot which are joint number 6, 7 and 8. The rest of the joints are passive or immovable, therefore the minimum and maximum values for them are \( \text{Any} \), which was set to zero in the calculation. The transformation matrices for both legs were calculated based on the information given in Table 2 and 3. The equation (12) in Chapter 3 was used to calculate all the transformation matrices for each joint except for joint number 5. Joint number 5 is a special case, because on the model, the coordinate system is rotated...
about z-axis first and then x-axis relative to joint number 4, therefore the equation (13) was used for this joint instead.

Table 4. Joints angle ranges and initial values of the robot

<table>
<thead>
<tr>
<th>i (joint number)</th>
<th>Right</th>
<th>Left</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Initial value</td>
<td>Minimum value</td>
</tr>
<tr>
<td>1</td>
<td>90º</td>
<td>Any</td>
</tr>
<tr>
<td>2</td>
<td>0º</td>
<td>Any</td>
</tr>
<tr>
<td>3</td>
<td>-90º</td>
<td>Any</td>
</tr>
<tr>
<td>4</td>
<td>0º</td>
<td>Any</td>
</tr>
<tr>
<td>5</td>
<td>90º</td>
<td>Any</td>
</tr>
<tr>
<td>6</td>
<td>-90º</td>
<td>-90º</td>
</tr>
<tr>
<td>7</td>
<td>0º</td>
<td>0º</td>
</tr>
<tr>
<td>8</td>
<td>0º</td>
<td>-90º</td>
</tr>
<tr>
<td>9</td>
<td>0º</td>
<td>Any</td>
</tr>
<tr>
<td>10</td>
<td>90º</td>
<td>Any</td>
</tr>
<tr>
<td>11</td>
<td>0º</td>
<td>Any</td>
</tr>
</tbody>
</table>

5.2.2 Angle to stroke conversion

The set of angle that were sent to the robot controller to move the leg according to the calculated trajectory had to be converted first to a set of strokes because the robot uses linear motors. The controller moves the joint by moving the slider of the motor forward or backward. Detailed descriptions of the mechanical and electrical characteristics of the motors can be found on Peralta et al. and NTI AG - LinMot & MagSpring (2011). Each powered joint in the robot has the same basic construction. The model and its descriptions can be seen in Figure 49.

It can be observed in Figure 49 that if the x bar extends or contracts, then the angle \( \beta \) will change. The main goal is to map the angle on the main joint to a stroke corresponding to the desired angle. The relationship between those two parameters can be calculated from the Law of Cosine. The derivation of the law itself can be
found in Wolfram Mathworld. If the law is applied to the triangle of A - (Bx) - C in Figure 49, then the equation will be

\[(B + x)^2 = A^2 + C^2 - 2AC \cos \delta \]  \hspace{1cm} (22)

\[B + x = \sqrt{A^2 + C^2 - 2AC \cos \delta} \]  \hspace{1cm} (23)

\[x = \sqrt{A^2 + C^2 - 2AC \cos \delta} - B \]  \hspace{1cm} (24)

The last equation was used in SUCAZULLYTractor software to convert the \( \delta \) angle to the stroke \( x \).

Figure 49. Basic construction of the robot powered joint

5.3 GIMNet integration

The SUCAZULLYTractor program is also a client program. This program can connect to a GIMNet server through AST Department network. GIMNet is a software infrastructure for controlling distributed system in real-time. It was developed in AST Department of Aalto. GIMNet was built over TCP/IP protocol and is a combination of Client-Server and Peer-to-Peer network. The advantages of using GIMNet are the robot can be controlled remotely in a real-time and the user can communicate with the
robot through a firewall because it has a VPN capability. The main library that the program uses to send and receive GIMNet data is GIM Interface or GIMI (Saarinen et al.).

### 5.3.1 Data format

The strokes from inverse kinematics calculation for each powered joint are sent to GIMNet server in the form of an array to command the robot. The order of the angles inside a payload is as follows:

\[
\text{payload} = [\text{left ankle, left knee, left hip, right ankle, right knee, right hip}]
\]

The SUCAZULLYTractor is capable of communicating with the GIMNet server because of the GIMI library that is included in its code. All libraries that support GIMNet communication and Client-Server program examples were made available by the AST Department.

### 5.4 Center of Pressure calculator

The CoP calculator is used to display the movement of CoP on each foot. This program takes the digital force data from microcontroller and calculate the CoP using equation (2) and (4) from Chapter 2. The coordinate system for the foot needs to be defined for this calculation. Figure 50 describes its position.

The sensors on the inner side of the foot are separated by 13.2 cm. The coordinate system is placed at the bottom left of the foot on the center of the force sensor. In double support case, only the coordinate system at sensor number 5 is used. The GUI of the program is presented in Figure 51.

The program takes the data from the microcontroller in real-time, extracts the sensor data from it and calculates the robot CoP. The CoP calculation results are displayed on two boxes on left and right sides of the GUI which represent left and right foots. It saves the data on a text file in its directory. It can also take the sensor data directly
without further processing and attach a time signature to each of them. This is useful for calculating the data rate of the microcontroller.

Figure 50. Top view of the feet and their coordinate system placement

Figure 51. SUCAZULLYCeptor, the CoP calculator software
Chapter 6

ZMP Data Acquisition and Analysis

This chapter presents the result and analysis of the ZMP experiment on the GIMbiped robot. The ZMP experiments were done in standing posture and standing with disturbance. The purpose is to analyze ZMP movement and determine a suitable control strategy for the robot to make it walk in a stable manner. The robot movement was restricted in a sagittal plane, which means the experiments were done for two dimensional standing and walking only.

Due to failures in mechanical construction of the robot and its supporting frame, the walking experiments were unsuccessful. The loose hip joint (left hip) of the robot made the robot underactuated, therefore the walking movement in 2 dimension could not be performed. The capability of the SUCAZULLYTractor software could also not be demonstrated. At that moment, there was not enough time to fix the robot.

6.1 Data throughput measurement

6.1.1 Microcontroller data rate

The sensor data rate that the microcontroller can transmit was measured by taking the data directly from the device without further processing. Every data was attached with a time signature for throughput calculation. The amount of the data was then plotted with respect to time intervals in second (Figure 52).
Figure 52. Microcontroller data throughput

The average rate of the data is 493.6 counts/second, therefore the interval of arrival between each data was 2.026 milliseconds. The standard deviation is 1.993 counts.

6.1.2 Program data rate

The software used for taking the data was SUCAZULLYCeptor. The taking of the data, the CoP calculation and the saving process affect the amount of the data that the software can handle at a certain period of time. The data throughput that the software could produce was determined by collecting the data with a time signature while calculating the CoP and then their amount were averaged with respect to their time period. The data were taken for 130 seconds and the amount was plotted with respect to the time in second. The graph is shown in Figure 53.

The minimum length of the data received was 17 characters and the maximum length was 41 characters. By averaging the number of data from Figure 53, the result is 14.792 counts/second which means that each data arrived in an interval of 1/14.792 seconds or 67.604 milliseconds. The standard deviation is 1.655 counts. The interval value was used to determine the time period of the other experiments.
By comparing Figure 52 and 53, it can be seen that the software calculation reduced the rate significantly. The rate of the CoP shown by SUCAZULLYCeptor was only 2.997% of the actual rate of the microcontroller. The reason for this reduction is inefficiency in the software computation. The program is using 2 time delays (one is used to calculate the CoP and the other one is used to display it in the GUI) in its process and the delays affect each other. The solution to solve this problem is either using 2 programs to calculate and display the CoP or using multithreading, so the 2 routines can run side by side.

6.2 Standing ZMP measurement

Once the motors were turned-on, noise was introduced to the output of the sensors, therefore the standing data were taken when the motors of the robot were in turned-off and turned-on conditions for comparison and further analysis. The data were taken for approximately 2 minutes. Because the robot is standing firmly, the ZMP coincides with the CoP of the robot. The force sensors data were mapped to their corresponding mass with the calibration equations and then plotted with respect to software data.
interval. The ZMP data when the motors were turned-off are presented in Figure 54 to 60.

![ZMP Position of the Left Foot in X-axis](image)

Figure 54. ZMP position on the left foot in x-axis when motors were turned-off (large scale)

It can be seen from Figure 54 that the sensor data are stable and not oscillating. The data are also not severely corrupted by noise. For further analysis, the axes scales of all figures in the experiments are magnified.

For both feet, the ZMP position was calculated inside the convex hull of the feet. The average and standard deviation values of the standing data are given in Table 5 and 6.

Table 5. Standing ZMP calculation when motors were turned-off

<table>
<thead>
<tr>
<th>ZMP</th>
<th>Left foot (mm)</th>
<th>Right foot (mm)</th>
<th>Both feet (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average in x-axis</td>
<td>109.790</td>
<td>2.929</td>
<td>122.360</td>
</tr>
<tr>
<td>Average in y-axis</td>
<td>126.547</td>
<td>124.582</td>
<td>125.564</td>
</tr>
<tr>
<td>Standard deviation in x-axis</td>
<td>0.058</td>
<td>0.075</td>
<td>0.045</td>
</tr>
<tr>
<td>Standard deviation in y-axis</td>
<td>0.176</td>
<td>0.163</td>
<td>0.113</td>
</tr>
</tbody>
</table>
Table 6. Standing ZMP calculation when motors were turned-on

<table>
<thead>
<tr>
<th>ZMP</th>
<th>Left foot (mm)</th>
<th>Right foot (mm)</th>
<th>Both feet (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average in x-axis</td>
<td>55.114</td>
<td>2.844</td>
<td>94.979</td>
</tr>
<tr>
<td>Average in y-axis</td>
<td>126.657</td>
<td>124.344</td>
<td>125.501</td>
</tr>
<tr>
<td>Standard deviation in x-axis</td>
<td>0.084</td>
<td>0.089</td>
<td>0.074</td>
</tr>
<tr>
<td>Standard deviation in y-axis</td>
<td>0.194</td>
<td>0.193</td>
<td>0.129</td>
</tr>
</tbody>
</table>

By comparing the standard deviations from Table 5 and 6, it can be concluded that the effect of the noise that the motor introduced was not severe and the data still valid. When the motors of the robot were not powered, there is a significant difference in the ZMP values for both feet on x-axis. This happened because the robot was not standing correctly. In Figure 67(a) and 67(b), the ZMP position of the left foot (blue), the right foot (red) and the feet (green) from Table 5 and 6 are illustrated. By referring to the figure, it can be seen that the ZMP on both feet is on the right foot, which means the CoM of the robot was staying on the right side of its body. When all the motors were powered, so the robot was standing on its own, the ZMP on the x-axis was moving further to the right. This can be seen on the Table 6. The reason on why the robot was not standing correctly when it was powered and un-powered is because there are differences between each leg parameters (joint position, link, robot motor and mechanical connector dimensions) when it was standing, so both feet did not support the robot mass equally (all the force sensors did not equally touch the ground). There are several assumptions for what could cause this problem. The first one is that the legs dimensions of the robot are not equal, the second one is that the frame that supports the robot did not support it correctly, and the last one is that the joints initial positions were not correct.
Figure 55. ZMP position on the left foot in x-axis when motors were turned-off

Figure 56. ZMP position on the left foot in y-axis when motors were turned-off
Figure 57. ZMP position on the right foot in x-axis when motors were turned-off

Figure 58. ZMP position on the right foot in y-axis when motors were turned-off
Figure 59. ZMP position on both feet in x-axis when motors were turned-off

Figure 60. ZMP position on both feet in y-axis when motors were turned-off
The ZMP data when the motors were turned-on are given in Figure 61 to 66:

Figure 61. ZMP position on the left foot in x-axis when motors were turned-on

Figure 62. ZMP position on the left foot in y-axis when motors were turned-on
Figure 63. ZMP position on the right foot in x-axis when motors were turned-on

Figure 64. ZMP position on the right foot in y-axis when motors were turned-on
Figure 65. ZMP position on both feet in x-axis when motors were turned-on

Figure 66. ZMP position on both feet in y-axis when motors were turned-on
Figure 67. Visual representation of ZMP position

6.3 Standing ZMP measurement with disturbance

To see the response of the ZMP position on the robot when an external disturbance is applied, a mechanism for disturbance simulation was made. The frame of the robot was connected to a powerful linear motor. The motor is produced by NTI AG - LinMot & MagSpring (2011). This motor introduced an oscillatory motion to the robot as external disturbance. The disturbance was only performed in a sagittal plan (forward and backward movement). The disturbance mechanism is given in Figure 68.

The moving speed, acceleration and deceleration of the motor can be specified. There were two disturbance experiments applied to the robot and they were carried out when the robot was in a powered standing condition.

6.3.1 Push disturbance with 0.01 m/s velocity

In the first experiment, the disturbance motor was set to perform push and pull movements continuously (oscillatory) with the maximum speed of 0.01 m/s and 0.25
m/s² acceleration and deceleration. The maximum displacement that the motor can push was 90 mm. The ZMP data under this condition were taken for 2 minutes. The results of the experiment are given in Figure 69 and 70.

From Figure 69, the ZMP in x-axis is changing its value although the movement was only done in sagittal plane. This means that the sensors in x-axis are also being pressed and un-pressed continuously. This can happen only if the robot did not stand correctly because of both legs did not share the force equally. This result also verifies the previous assumption about the difference in leg parameters. However the change is very small.

Figure 68. Robot external disturbance mechanism
Figure 69. ZMP position on both feet in x-axis when 0.01 m/s disturbance was applied

Figure 70. ZMP position on both feet in y-axis when 0.01 m/s disturbance was applied
6.3.2 Push disturbance with 0.04 m/s velocity

In the second experiment, the maximum speed was changed to 0.04 m/s but the acceleration and deceleration remained 0.25 m/s$^2$. The data are shown in Figure 71 and 72.

The responses from the disturbances are equal with the previous one in amplitude and pattern, the only difference is the frequency of occurrence. The second disturbance creates 4 times higher frequency than the first, since the maximum speed of the disturbance motor was multiplied by 4 in the experiment.

The disturbance experiments were applied to simulate a pushing disturbance when the robot is in a standing position. By analyzing the ZMP movement on the foot, a feedback control for the robot standing position can be utilized. One of the control strategies is specifying a stable region on the convex hull of the feet. The illustration is given in Figure 73.

![ZMP Position of The Feet with 0.04 m/s Disturbance](image)

Figure 71. ZMP position on both feet in x-axis when 0.04 m/s disturbance was applied
Figure 72. ZMP position on both feet in x-axis when 0.04 m/s disturbance was applied

Figure 73. Stable region of the robot

From Figure 73, the stable region (the gray area) is the area where the standing ZMP can lies and the robot remains stable. The blue area is the double support region. When the ZMP moves outside the boundary of the stable region because of an
external disturbance, the robot needs to add compensation angles to its powered joints to move the ZMP back to its initial position. For the disturbance in the sagittal plane, the hip, the knee and the ankle joints are involved in shifting the ZMP (Kim et al. 2003).

6.4 Bipedal robot weight measurement

The robot total mass can be calculated by inputting force sensor data to the mapping equations that were obtained through sensor calibration. All converted data were added together to form the total robot mass. The data were taken in a condition with and without disturbances when the robot was in a standing position. Both data were compared and examined for further control analysis. All the motors of the robot were turned on in this experiment.

6.4.1 Weight measurement without disturbance

The data were calculated using ZMP data in x and y-axis at a standing position when the motors of the robot were turned-on. The data are shown in Figure 74, 75 and 76.

![Figure 74. Robot mass measurement](image)
Figure 75. Robot weight measurement

Figure 76. Frequency domain graph of the robot weight measurement without disturbance
The average and standard deviation values of the robot mass without the disturbance (Figure 74) are 41.489 kg and 0.039. The weight was also plotted in Figure 75 by multiplying the mass data with the acceleration constant due to gravity or 9.80665 m/s$^2$ (Taylor & Thompson 2008). The average weight value and its standard deviation are 406.864 Newton and 0.385. It can be seen from the frequency domain graph of the weight that almost all frequencies are present there (Figure 76). The frequency response was calculated using Fast Fourier Transform (FFT) function in MATLAB software. FFT is an algorithm to transform signal in a time domain to a frequency domain. A more detailed explanation of the FFT can be found in Press et al. (1992). From Figure 76, the frequency spectrum belongs to the white noise in the circuit because it is composed of many frequencies (Ott 2009). The average and standard deviation values of this spectrum are 9.804 Newton and 5.142. The standard deviation value can be used to determine the smallest possible weight fluctuation that the sensors can measure. In this case, the sensors cannot measure the weight fluctuation in the range of 0 and 5.142 Newton.

### 6.4.2 Weight measurement with 0.01 m/s disturbance

The data was taken with oscillatory disturbance from the disturbance motor with maximum velocity of 0.01 m/s. The motor acceleration and deceleration were 0.25 m/s$^2$. The data are given in Figure 77 and 78.

On Figure 77, when the first disturbance was applied to the robot, it can be seen that the graph has two different peaks. One peak is slightly higher than the other. The high peak happened when the backward force was applied to the robot. The low peak relates to the forward force on the robot. This assumed to happen because the ankle joint of the robot is not located in the center of the foot. Figure 79 will be used to analyze the case.

In Figure 79, a simplified foot model is used to explain the issue. The distance between the toe (point A) and the heel (point C) to the ankle (point B) is denoted by $r_2$ and $r_1$. $F_1$ and $F_2$ are external forces with equal value applied to the tip of the links. It is assumed that the CoM concentrated at point B or ankle joint. In Figure 79 (a), when
the mass receives $F_1$ force, the torque will be introduced to the foot at point B. The torque will also affect the point A and introduce $F_{R1}$ force. The relationship between $F_1$ and $F_{R1}$ is

$$\tau = rF_1$$  \hspace{1cm} (25)\\
$$\tau = r_1 F_{R1}$$  \hspace{1cm} (26)\\
$$r_1 F_{R1} = r F_1$$  \hspace{1cm} (27)\\
$$F_{R1} = \frac{r}{r_1} F_1$$  \hspace{1cm} (28)\\

The same goes for in Figure 79 (b). The relationship between $F_2$ and $F_{R2}$ is

$$F_{R2} = \frac{r}{r_2} F_2$$  \hspace{1cm} (29)\\

Since the $r$ length is fixed and $r_2 < r_1$, it can be deduced from the equation (28) and (29) above that $F_{R2} > F_{R1}$. The equations verify the assumption of the ankle joint position the peak heights in Figure 77.

![Robot Weight Measurement with 0.01 m/s Disturbance](image)

Figure 77. Robot weight measurement when 0.01 m/s disturbance was applied
Figure 78. Frequency domain graph of the robot weight measurement with 0.01 m/s disturbance

Figure 79. Foot force and velocity calculations
Still from Figure 77, the valleys correspond to the transition period between maximum and minimum displacement of the slider of disturbance motor. The heights of the valleys are also different. The transition from minimum to maximum (min to max) displacement produced deeper valley than from maximum to minimum (max to min). Figure 80 compares the mass graphs when before and after the disturbance applied.

From Figure 80, the comparison of the mass measurement when the disturbance were introduced and removed can be seen. The black graph corresponds to the measurement with disturbance and the blue graph is the measurement result without disturbance. It is assumed that the CoM paths of the robot are not the same when the robot is pushed forward and pulled backward, therefore it creates different pressures between two transitions. Figure 81 is given to illustrate the movement.

![Robot Weight Measurement Comparison](image)

Figure 80. Weight measurement comparison between zero and 0.01 m/s disturbance
The CoM seemed to create unequal path that produced extra upward and downward forces (the paths assumed to be parabolic). These two forces affected the weight measurement of the robot when the disturbance was applied. The forward movement created an upward force that reduced the effect of the gravitational force which then reduced the pressure on the force sensors. The position of the maximum upward force is denoted by the minimum value in the min to max transition (Figure 80). In the other hand, when the robot was pulled backward, the CoM acceleration produced downward force. This force added to the gravitational force, so the transition from the max to min had higher pressure values than from the min to max. The frame that supports the robot is assumed to be the source of this problem.

From the frequency domain graph of Figure 78, the first dominant frequency is 0.042 Hz. This frequency is assumed to be the frequency of the CoM due to the disturbance. This can be verified by the disturbance motor frequency calculation. The calculations are as follows:

\[ t_m = \frac{s_m}{v_m} \]  
\[ t_m = \frac{0.09m}{0.01m/s} \]  
\[ t_m = 9s \]
$s_m$ is the length of the slider, $v_m$ is its movement velocity and $t_m$ is the time of the linear motor when travels at one direction. To calculate the time required when the motor is going forward and backward, the traveling time must be multiplied by 2 which is 18 s. Therefore the frequency of the motor will be

$$f_m = \frac{1}{2t_m} \quad (33)$$

$$f_m = \frac{1}{18s} \quad (34)$$

$$f_m = 0.056\ Hz \quad (35)$$

The 0.014 Hz difference in frequency was occurred because the motor was not ideal, its friction and its mass can slow down the moving parts and decrease its frequency of movement. The other dominant frequency components are assumed to be caused by the vibration of the mechanical components of the robot. The robot consists of several connections between joint and link and also the robot to the frame. Each joint in one leg has different dimensions and mass. Those differences can affect the total body vibration frequency because there can be a part in the robot that vibrates more than the other. The remaining three dominant frequency components are a multiplication of the first component (0.042 Hz) by 2.3, 3.3 and 4.3. The second dominant frequency indicates that there was a mass attached to the robot that vibrated 2.3 times faster than the robot’s vibration frequency. The last two dominant frequencies were also contributed by some parts of the robot that vibrated 3.3 and 4.3 times faster than the CoM frequency. The second dominant frequency has the highest amplitude. The amplitude is assumed to be formed by more than one vibration sources with the same frequency and phase so the constructive interference happened. To verify these assumptions, frequency domain analyses of the robot posture with different joint and link configurations are needed (e.g. by strengthening and loosening one or more joints). More examples on vibration diagnostics can be found in Piersol & Paez (2010).
6.4.3 Mass measurement with 0.04 m/s disturbance

The data was taken with oscillatory disturbance from the disturbance motor with maximum velocity of 0.04 m/s. The motor acceleration and deceleration were the same as the previous experiment. The graph representations of the data are given in Figure 82 and 83.

The results are the same as with the previous mass experiment with 0.01 m/s disturbance. The differences are the amplitude and the frequency of the pattern because the oscillatory disturbance in this experiment has a higher force and frequency. The frequency can be obtained using equation (30) and (33), which is 0.222 Hz. From Figure 83, the first dominant frequency is 0.182 Hz which is assumed to be related to the CoM frequency due to the disturbance. The other frequency components are assumed to be caused by the vibration of the mechanical components of the robot just like in the previous experiment. The other three dominant frequency components are a multiplication of the first component (0.182 Hz) by 2, 3 and 4. The identification of the parts can only be done by changing the robot parameters (e.g. by strengthening and loosening one or more joints) and comparing the frequency domain responses from each configuration. From the frequency analysis results, the amplitude and vibration frequency are related to the disturbance frequency applied to the robot. It can be inferred that the sensors can also be used as a tool for vibration analysis.
Figure 82. Robot mass measurement when the second disturbance was applied

Figure 83. Frequency domain graph of the robot mass measurement with 0.04 m/s disturbance
Chapter 7

Conclusions

This chapter summarizes all the work that have been done for this thesis. The research was performed using GIMBiped robot platform that is available in AST Department of Aalto. The overall goal of the thesis could not be fully fulfilled because of the limitations in research time and in the robot mechanical constructions. The force sensors for the feet of the robot were successfully utilized on the robot. The walking trajectory and inverse kinematics (SUCAZULLYTractor) software was developed to control the robot in an open-loop manner. The CoP of the robot was calculated and displayed by the center of pressure calculator (SUCAZULLYCeptor) software. The ZMP position analysis had only been done in a robot standing position. Disturbance experiments were also applied to the robot.

7.1 ZMP hardware

Several hardware were built in this thesis work, which are force sensors circuits, eight force sensors holders and a pair of foot platforms. The circuits are the power supply, the instrumentation amplifier, the LPF and the ADC and microcontroller circuit. They were designed using Altium Designer Software and the PCBs were manufactured by etching technique. All circuits were equipped with ground planes to minimize the effect of noise. Other noise reduction techniques (i.e. cable twisting and shielding) were also applied to reduce the effect of the capacitive and inductive coupling that can introduce noise to the circuits. The force sensors were compensated electrically to remove the effect of temperature changes. The sensors were calibrated using a lever
system that can multiply the mass applied to it. Water was used as the changing mass. The experiments results show a good performance of the circuit. The sensor data were reliable and did not oscillate. The noise reductions and compensation technique made the sensor reliable and allow it to take the data successfully.

7.2 ZMP software

The SUCAZULLYDager is the application for taking the sensor data from the microcontroller and average them. This application was used to calibrate each force sensor on the feet of the robot. Two other software were needed to generate a walking movement of the robot and display the position of the ZMP to ease its interpretation and analysis. The SUCAZULLYTractor is the software that generates a walking trajectory for the hip and the feet of the robot. This application also calculates the inverse kinematics of the trajectory to map the trajectory to a set of angles for the joints of the robot to follow. The ability to connect to GIMNet server was added to the software to control the robot remotely through a local network. The SUCAZULLYCeptor was made to collect and visualize the ZMP data on its GUI. The software can take and display the data in real-time.

7.3 ZMP data

Due to a mechanical limitation on the robot, the ZMP experiments were only performed in a standing position. The ZMP data were taken and plotted with respect to the processed data interval time. The reliability of the data was confirmed further by the standard deviation calculation. Disturbance experiments were performed to observe the movement of the ZMP under the influence of external force. This data can be used as a feedback to control the ankle joints torques of the robot in a closed-loop way. The force data were then collected by a computer and transformed using calibration equations to mass representations. Frequency domain analysis was used to analyze the frequency response of the robot by performing FFT to the collected data.
By inspecting the frequency domain graph, a vibration analysis of the robot could be implemented.

7.4 Future work

Although the ZMP hardware and software worked as they should, there are still some issues that can be considered in the future development of the robot to increase the performance of its entire system. These issues cover the hardware and the software of the robot.

7.4.1 Robot hardware

From the experiment results, both legs parameters of the robot were not equal, this can be related to different dimensions of the legs or the motors. Further inspection of the robot design needs to be conducted to fix this problem. The robot dimensions documentation was not available when the thesis was made. This can lead to a serious problem as the dimensions are necessary to calculate the transformation equation for the legs. Wrong dimensions input can cause the trajectory generator software to produce wrong walking trajectory for the robot. Documentation needs to be made to avoid the problem. The knees of the robot could easily overheat under the heavy load. This problem makes the standing experiments (if they are bended) or walking experiments difficult to perform for a long period of time. The problem can be fixed either by changing the knees motors with the more powerful one or applying a better cooling mechanism to the motor. A thick copper material can be used to dissipate heat better.

In force sensors circuits, the improvement can be made on the positioning of the instrumentation amplifier of the force sensors. A long sensor cables that connect the sensor to the amplifier are prone to capacitive and inductive couplings. The couplings can further introduce noise to the circuit. By placing the amplifier on the legs of the robot, the effect of the noise can be reduced. The performance can also be increased by separating its power supply from the digital circuit (i.e. the ADC and the
microcontroller circuits) supply and introducing bypass capacitors near the amplifier supply pins. This method not only can prevent the high frequency noise from the digital circuit entering the amplifier but also reduces the bad effect of sudden voltage change in the power supply on the circuit input. Generally, the power supply separation between the analog and the digital signal is necessary to reduce the noise. The force data were transmitted through a serial communication to a computer. To better organize the sensor data acquisition, it can be considered to send the force sensor data to the available sensor hub for further processing.

7.4.2 Robot software

In the area of the software, some parts can be improved. The communication capability of the SUCAZULLYTractor to the GIMNet can be upgraded by adding a joint position query feature. The feature will ensure that the data sent to the server is successfully executed. The SUCAZULLYCeptor takes the data and calculate the CoP of the robot at the same time. This can lead to a severe delay that was shown in data analysis section previously. To decrease the delay, the two routines need to be separated into two different processes by applying a multithreading technique.
References


National Aeronautics and Space Administration (NASA), 2010b, *Robonaut2, the Next Generation Dexterous Robot*, NASA, viewed 27 July 2011,


Nokia Corporation, 2011, *Qt*, Nokia Corporation, viewed 8 August 2011,


PJRC, Teensy USB Development Board, PJRC, viewed 29 July 2011,
<http://www.pjrc.com/teensy/>


Taylor, BN (ed.), Thompson, A (ed.), 2008, The International System of Units (SI), National Institute of Standards and Technology, Gaithersburg, viewed 13 August 2011,

Texas Instruments Inc., 2005, INA126, INA2126, Micropower Instrumentation Amplifier, Single and Dual Versions, Texas Instruments Inc., Dallas, viewed 13 August 2011,

Todd, DJ, 1985, WALKING MACHINES, AN INTRODUCTION TO LEGGED ROBOTS, Kogan Page Ltd, London.


Appendix A

Electronic Circuit Schematics

Figure 84. Power supply circuit schematic

Figure 85. Low-pass filter circuit schematic
Figure 86. Instrumentation amplifier circuit schematic
Figure 87. ADC and microcontroller circuit schematic
Appendix B

Sensor Holder Sketch

Figure 88. Sensor holder sketch