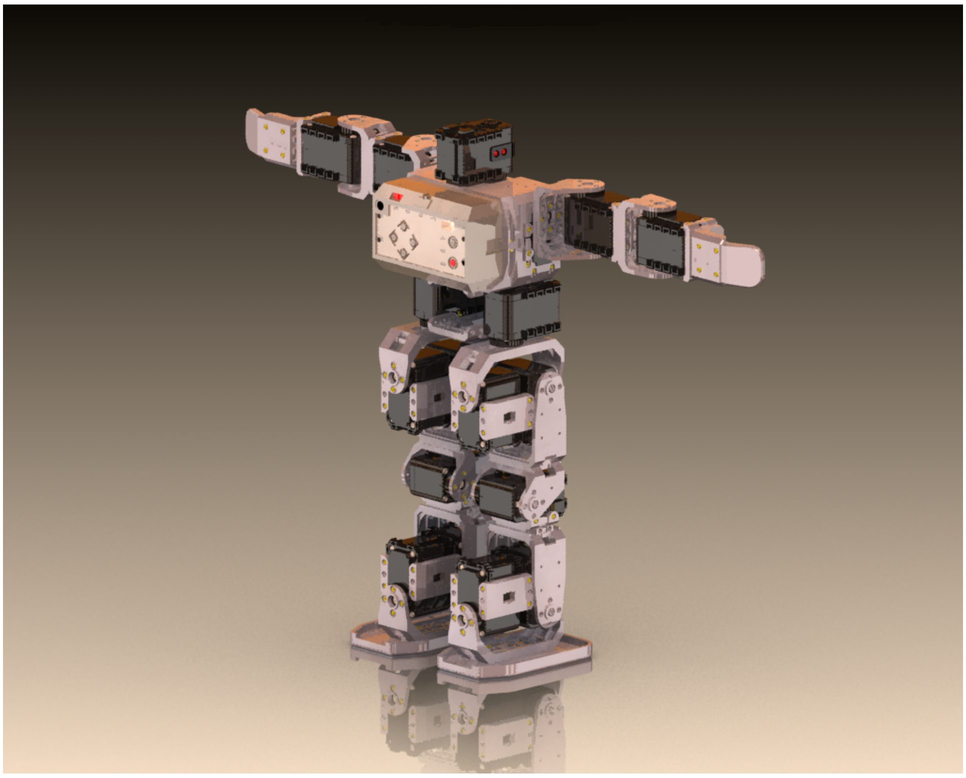
TECHNICAL UNIVERSITY OF CRETE

Department of Production Engineering
&
Management



MARIOS TYROPOULOS

"Modeling, Forward Kinematics, & Static Balance Equations for the Bioloid Humanoid Robot"

August 27, 2008

The Supervising Professor :
NIKOS VLASSIS

Examination Board:
NIKOS VLASSIS : Dept. of Production Engineering & Management
IOANNIS NIKOLOS : Dept. of Production Engineering & Management
DIMITRIS ROVAS : Dept. of Production Engineering & Management

Contents

Chapter 1

INTRODUCTION

1.1 Robotics Today

- **Industrial Applications**

  Modern industrial robots are true marvels of engineering. A robot the size of a person can easily carry a load over one hundred pounds and move it very quickly with a repeatability of +/-0.006 inches. Furthermore these robots can do that 24 hours a day for years on and with no failures whatsoever. Though they are re-programmable, in many applications (particularly those in the auto industry) they are programmed once and then repeat that exact same task for years.

  The Cartesian Robot

  Some original ideas have been used to construct original robots such as: The machine below that can be called a Cartesian robot, though calling this machine a robot is really stretching the definition of a robot. It is Cartesian because it allows x-y-z positioning. Three linear joints provide the three axes of motion and define the x, y and z planes. This robot is suited for pick and place applications where either there are no orientation requirements or the parts can be pre-oriented before the robot picks them up.

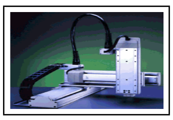


Figure 1.1: The Cartesian Robot

The Bioloid Plotter

As the humanoid we used in this thesis a similar robot with the same philosophy and also constructed with the Bioloid Kit can be used as a plotter.

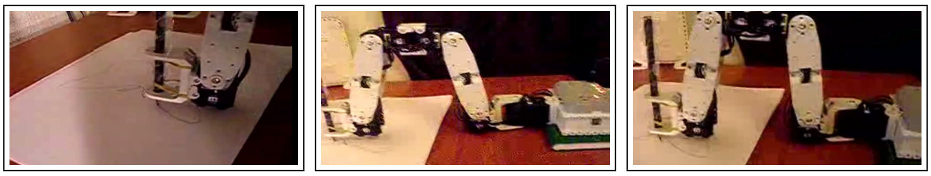


Figure 1.2: The Bioloid Plotter

Exoskeletons[9]

With the term "Exoskeleton" we reffer to robotic systems that are worn by the human operator as an orthotic device and used as a human-amplifier assistive device, haptic device or for automatic physiotherapy. Below we will present the potentials of the exoskeletons according to the purpose for which they are used.

The exoskeleton robot, serving as an assistive device, is worn by the human (orthotic) and functions as a human-amplifier. Its joints and links correspond to those of the human body, and its actuators share a portion of the external load with the operator. In this case the job of the coders of the actuators is being performed by a far more sophisticated method. One of the primary innovative ideas of the proposed research is to set the Human Machine Interface (HMI) at the neuromuscular level of the human physiological hierarchy using the body's own neural command signals as one of the primary command signals of the exoskeleton. These signals will be in the form of processed surface electromyography (sEMG) signals, detected by surface electrodes placed on the operator's skin. However the main principle stands. Firstly a method of measuring angle joints is performed and then the data are used so that the joints of the exoskeleton are moved by the same angle of the user's movement and in respect to every joint.

Figure 1.3: Exoskeleton

- **Education**

  Robotic devices enabling learn and perform delicate and dangerous tasks safely and effectively.

  SIMULATION-VIRTUAL ENVIRONMENTS [9]

  The Linear Haptic Display (LHD) - Excalibur is a 3 degree-of-freedom haptic device, built for maximum workspace, force output, and structural stiffness. It was designed and manufactured by Haptic Technologies, Inc. of Seattle, WA. Its hallmark is a patented steel cable transmission system which enables high forces and high rigidity in the three orthogonal axis of translational motion. The motors are mounted on the base of the mechanism, so that only light linkage elements move with the hand grip.

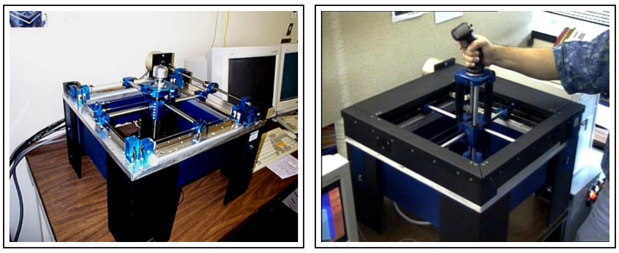


Figure 1.4: The Linear Haptic Display

Potential applications of Excalibur include virtual reality training, computer-aided design (CAD), telerobotic manipulation, and entertainment.

The Red Dragon[9]

The Red Dragon is a system that tracks the motion of two minimally invasive tools along with all the force and toques applied on the tools by the surgeon as he or she interacts with the simulated or animal models. The Red Dragon unique mechanism allows crossing the various training modalities while providing a standard interface. It follows the trainee as he or she progresses from a simulation environment to the reality (animal model) while providing quantitative information for objectively assessing the trainees technical skills.

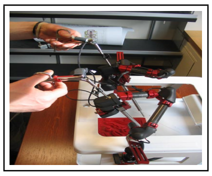


Figure 1.5: The Red Dragon

Research projects the kinematics and the dynamics of Minimally Invasive surgery - Objective assessment of surgical Performance Using Markov Models.

- **Non Human friendly environments**[7]

  Radio Active Environments

  The robot below was developed for the decontamination and dismantlement of nuclear weapons facilities. The radioactive fields makes this activity too hazardous for human workers so the use of robotics makes sense. The idea for this robot is that it can hold a part in one hand and use a cutting tool with the other; basically stripping apart the reactor layer by layer.

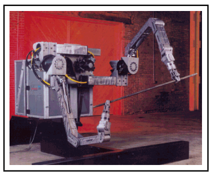


Figure 1.6: The Dawm Pipe

Research in Undersea Environments

Undersea operations are a great application for robotics to replace humans. Working underwater is both dangerous and difficult for humans. Schilling Robotics makes the system below called Quest. This system combines a remote operated vehicle with thrusters for maneuvering and two robot arms for manipulating. One of the arms is almost a grapper. It can grab something rigid, such as the base of an oil rig, to steady the vehicle while the other arm performs such tasks as welding and valve maintenance.

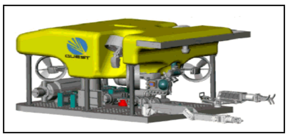


Figure 1.7: The Quest

Law Enforcement

Law enforcing is a dangerous activity. This makes it a good opportunity for robots. It is, however, highly-unstructured and that makes it very difficult for robots. Robots have found application in training, surveillance and bomb disposal. Bomb disposal by its nature is a highly risky task and the since it is dangerous for the physical integrity of a human being while the task itself is elementary, it was undertaken by robots. The robot below is called a Mini Andros and is made by the Remotec corporation. There is a sturdy manipulator on the front. The robot uses this manipulator to pick up the bomb and then uses the mobility platform to move the bomb to a remote location where it can be safely destroyed by secondary explosives. All of these activities are telecontrolled by a remote operator.

Figure 1.8: The miniAndros

- **Medical Applications**

  In this field using a change in scale, through forward kinematics a movement of a human being can be scaled down so that we can gain precision. This technic is widely used in the surgical field, where for example a surgeon may use micro-manipulator technology to conduct surgery on a microscopic level where the human precision is not enough and immediate decision making is essential. Also, telerobotic research is being done in the field of medical devices, and minimally invasive surgical systems. With a robot system a surgeon can work inside the body through tiny holes just big enough for the manipulator, with no need to open up the chest cavity to allow hands inside.

  THE HAND EXOSKELETON[10]

  The powered hand exoskeleton is an orthotic device that is worn by a human. Individual joints of the human hand can be moved. The actuation is performed by electric motors which transmit forces to the joints through Bowden cables. A real-time controller calculates the necessary control signals for the motor controllers based on the measured forces and joint angles. A separate interface computer allows the therapist to change the control modes, define new exercises, and supervise the rehabilitation. Following features are integrated to support movement and diagnosis:

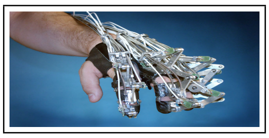


Figure 1.9: The TU Berlin HandExoskeleton

  The device was developed with the focus on the rehabilitation of hand injuries.

  Other possible application of hand exoskeleton devices are:

  - Haptic feedback for virtual reality
  - Telemanipulation
  - Assistive device

Robotic Jackets

Devices that help stroke victims recover from partial paralysis could be ready to wear in the near future.

Figure 1.10: The Robotic Jacket

The device-essentially a mesh jacket in form-uses sensors to detect the muscle movements in the patient's healthy arm and wrist, then uses artificial muscles to stimulate that same movement on the damaged side of the body. Researchers hope repeated therapy will bring back the regular functioning of the damaged limb.

1.2 Our Thesis

The main objective

In this diploma thesis we will present the modeling, the solution to the forward kinematics problem, and the static balance equations of the Bioloid Humanoid Robot by using the creation of a mathematical set of equations that will be able to describe sufficiently and accurately the movement of the BIOLOID humanoid.

Our approach

For the completion of the thesis we used a certain pattern for the placing of all of the coordinate systems on the CoM (Center of Mass) of each link. We also had to incorporate the use of the 3D designing program Solidworks since it could provide us with the necessary accuracy for the calculations of the geometrical features of parts of the Humanoid and also for the calculation of the mass properties involved in the static balance equations that describe the position of CoM of each link in relevance to the center of mass of previous ones. After we carried out the modeling and the complete definition of all of the geometrical features of the Humanoid - since they are prerequisites for static balance issues - we created the static balance equations and finally tested them with a fair number of experiments to ensure their validity. To do so we used a software provided with the bioloid kit with which we could create poses(2.4) in static equilibrium by manually setting each actuator to the desired position and then the program would return the values of each actuator from every posture. The next step was to apply these values in the equations and then to check the output of those equations in order to see whether their results are indeed lying within the support polygon.

The Modeling of the Humanoid

Before we proceeded to the main problems of the thesis we had to perform a accurate modeling of the system by using a convention quite similar to a commonly used convention for selecting frames of reference in robotics applications, the Denavint and Handergberg convention. As with the D-H convention we described the whole system using local coordinate systems and each time the position of the following would be described in relevance to the position of the base or so called the system origin's coordinate system. We examine joints with two links and we always have a follower and a base.

In order to follow our convention we had to define coordinate systems and place them on the center of mass of every link in contrast to the D-H convention placing the coordinate systems on every joint. Also the orientation we followed for the placement of every local coordinate system is following only one rule. That of minimization of calculations. We oriented the system of coordinates identically for a pose of the Humanoid that their actuators have 0° of turn.

The Use of Solidworks

The main problem that we had to overcome was the specifying of the position of the Centers of Mass within the space. To do so we had to know the exact geometrical features of each bioloid kits component and even if we had them or even measure each and every one of them, there would still be the issue of the very complex calculation of parts that do not have uniform density nor are symmetrical. Thanks to Solidworks not only were we able to define the exact position of the Center of Mass of each component but we did so rather accurately since the level of precision lavished by the 3d program reached 0.01mm. With this application, having given the mass properties of every component such as material and density the application returns the relative distances of the position of each Center of Mass in accordance to the assembly origin (4.1) used for the creation of each component. That way we could proceed to the static balance equations since all of the geometrical features could be calculated accurately enough with the use of Solidworks.

The forward kinematics

With the forward kinematics we described the interconnection of the components which implies that the motion of the components relative to each other is constrained but without regard to the constraining forces and define the motion of the interconnected bodies through space without regard to the generalized forces that cause a motion.

The static balance equations of the Bioloid Humanoid Robot

As far as the problem of static balance is concerned, we created a set of equations that is able to describe sufficiently and accurately the movement of the Bioloid humanoid. The equations have only one argument each and that is the present angle of the motor. Using this argument, each set of thriads (one equations for each dimension: X, Y, Z) describes the position of the link under examination according to the position of the previous link.

Once we had completed all of the 18 sets we used algebra to interpret the relative distances described by the equations in accordance to the system's origin defined as the Center of Mass of the right foot. That way once we had described every link's Center of Mass in accordance to the system's origin we used algebra to incorporate all of the 17 Center of Mass into a Center of Mass concerning the whole system. With the examination of the projection of this total Center of Mass onto the plane of the support polygon we are able to extract accurate conclusions as far as whether the system is in a state of static equilibrium just by checking if the projection of the total Center of Mass is found within the limits of the foot. All of the above while taking into consideration that static balance assumes that if the system's motion is stopped at any time, it will stay in a stable position indefinitely. That is, while the movement is slow enough, the system dynamics can be ignored.

The Experimental Verification

Once the calculation of the static balance equations had been performed it was of essential importance to perform a set of experiments to test the validity of the equations. To do so we used a software provided with the bioloid kit with which we could create poses(2.4) in static equilibrium by manually setting each actuator to the desired position and then the program returned as the values of each actuator from every posture.The next step was to put these values in the created equations and then to check the output of those equations in order to see whether their results are indeed lying within the support polygon.

Chapter 2

THE BIOLOID KIT

2.1 The concept

The Robotis BIOLOID kit includes a large number of brackets nuts and bolts which the user can easily assemble into a variety of robots that have biological features, anything from a cat, a puppy, a spider even a snake, up to a humanoid. The name Bioloid comes from the wordsBio + all + oid meaning that any living thing can be built in the form of a robot.

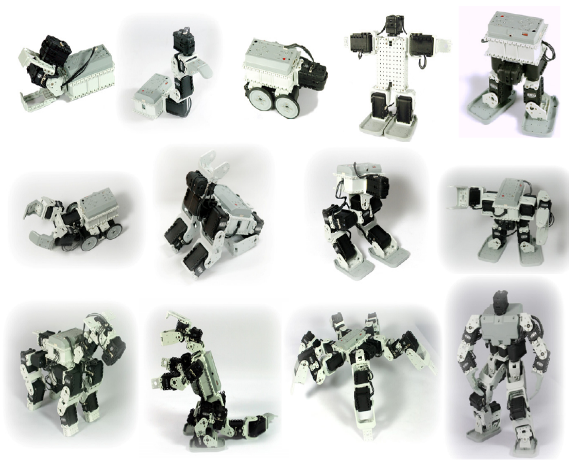


Figure 2.1: Robots constructed with the Bioloid Kit

The *Bioloid Expert Kit* concept can easily be compared to the concept *NXT Midstorm LEGOS* kits have. The kit supplies the components and the assembly is relatively unrestricted as the outcome is limited almost exclusively by the user's imagination. But unlike the *LEGO* sets, the robot is built with blocks that are actuated, so the joints can move.

2.2 The Bioloid Expert Kit

Contents: Within the aluminum case that includes the *Bioloid Expert Kit* components one can find the followings:

Figure 2.2: The Bioloid Kit Contents

2.3 The Hardware

The hardware of the Bioloid which consists of *three types*.

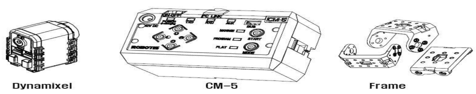


Figure 2.3: The main Hardware Components

- **Dynamixel**: This is the basic unit of the Bioloid which acts as a joint or a sensor. The AX-12 Dynamixel is an actuator that is used as a joint and it provides a torque to move this joint. *The Maximum Holding Torque* of the Ax 12 is ***12 kg f.cm at 7V and 16.5 kg f.cm at 10V***. Each Bioloid AX-12 Servo has its own Atmel MEGA8 micro controller. The servos are connected to a RS-485-like half-duplex serial bus. The Atmel MEGA 128 acts as the host and sends servo commands to the AX-12 servos at a data rate of 1 M bit/s.

Besides the ability to move the most important thing in making a robot autonomous is to give it the ability to sense and gather information. A device that can sense information is called a sensor. A sensor not only has the ability to sense objects, but also people or other robots. The process of a robot sensing outside information and reacting to it via outputting a movement is called robot interaction. The AX-S1 Dynamixel is a sensor unit that can sense distance, sound and the brightness of light.

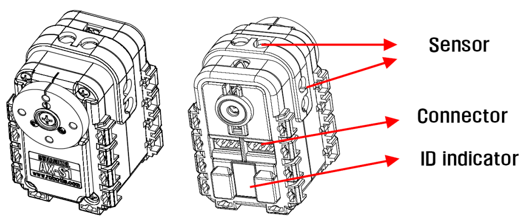


Figure 2.4: The AX-s1

In order for the user to operate Dynamixel AX-12, he or she has to understand the underlying principle of angle unit. The AX-12 can control 300 by 1024 unit steps. That means that the dynamixel AX-12 can rotate with a fairly accurate ***step of*** 0.2932°. When the groove of both horn and AX-12 correspond, the location value will be 512.

For the approach we used to form our equations for forward kinematics we decided the ***512 value of the actuator to be equal to*** 0° and that every value from 512-0 would correspond to negative angles-rotation from 0°to -150° respectively and every value from 512 up to 1023 would correspond to positive angles-rotation from 0° to + 150° respectively.

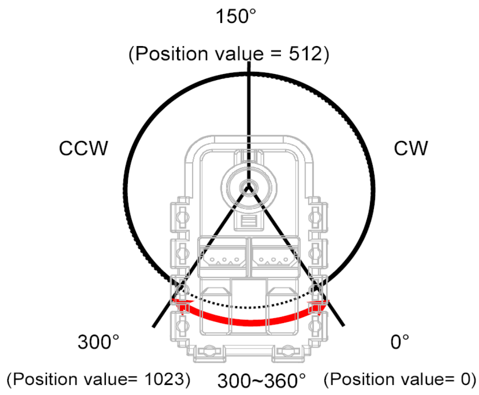


Figure 2.5: The Acceptable Actuator Values

Given that the a single joint of the human body cannot move more than maximum of 300° , the idea of using servos as actuators is obvious. A servo can provide *high torques* despite having a small size. In addition the positioning is very accurate being able to reach every angle exactly and to hold this position regardless which torque is acting on it. The power consumption of such servo is nearly the same as for a DC motor. Nevertheless, the maximum speed of a servo is limited by the gearbox and the speed of the motor inside and therefore they are not as fast as a motor.

AX-12 Continuous Turn Mode Control:

It can also be used as a wheel when set to endless turn mode, and additionally has functions of sensing both temperature and load. When the AX-12 is in continuous turn mode, it will be controlled in the motion speed, not desired position. If you input the value from 0 to 1023 in motion speed, AX-12 will rotate clockwise. Of course, the value of 0 will make the AX-12 stationary. However, if you put the value above 1024, it will rotate counter-clockwise corresponding to the inputted value. For example, if you input 600, it will rotate clockwise corresponding to the speed of 600, whereas, if you input 1624, it will rotate counter-clockwise, once again at the speed of 600(600+1024). Moreover, should a command that depicts in overload of the actuator occurs, then the temperature sensor will temporarelly turn the particular actuator off and its LED will start glowing indicating that something is wrong.

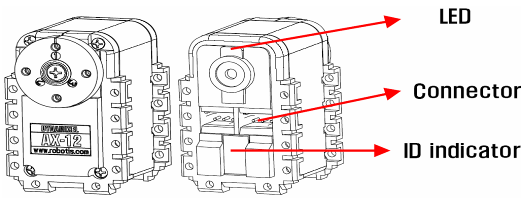


Figure 2.6: The Main Parts of the AX-12

- **CM-5**: The CM-5 is the main controller for the Bioloid. A robot is built by connecting the Dynamixels to the CM-5 as the central unit. Batteries that are placed here supply power to the connected Dynamixel. This unit includes an Atmega -128 processor a battery pack and the connector expansion board.

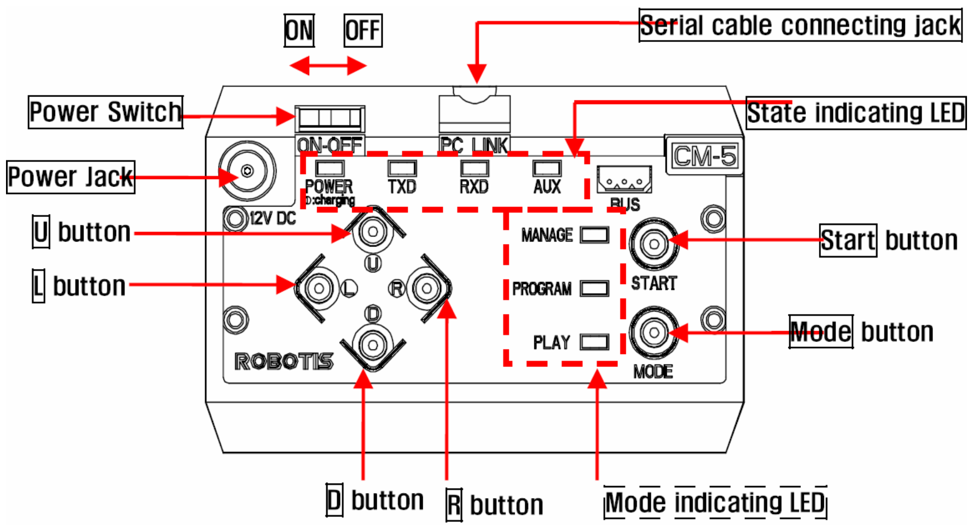


Figure 2.7: The Buttons of the CM-5

The main controller communicates with the Dynamixel units by sending and receiving data packets. There are two types of packets : the Instruction Packet (sent from the main controller to the Dynamixel actuators) and the Status Packet (sent from the Dynamixel actuators to the main controller.) If the main controller sends an instruction packet with the ID set to N, only the Dynamixel unit with this ID value will return its respective status packet and perform the required instruction. It is much like a communication chain. In order for the signal to reach the n actuator it has to be passed on from the unit to the next actuator to ne n-1 actuator and finally to the target actuator. Below we can see a picture describing the flow of information from the prossecor to the actuators.

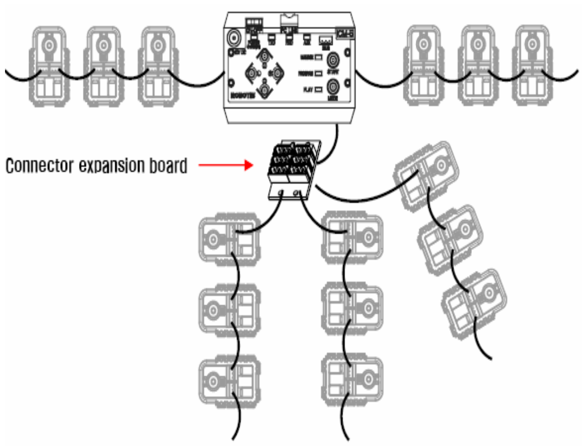


Figure 2.8: The Communication Chain between CM-5 and AXs

- **Frame**: The frame connects the robot units. The Dynamixel can be connected together with the use of the frame. Also, the frame connects the Dynamixel and the CM-5. The assembly of every frame part can be done easily using only a specific screwdriver. However the assembly of a sophisticated robot such as the humanoid can be very time consuming since there are about 300 bolts to be screwed. The total assembly can take 7 to 10 hours for a user that is quite familiar with this kind of constructions and this amount of time does not take into consideration the difficulties the assembler might encounter while trying to apply the manual's instructions. That is 7 to 10 hours assembly time.

2.4 The Software

The software of the *Bioloid Expert Kit* consists of *three types*:

- The Motion Editor: The Motion Editor is a package that allows the user, using a graphics based interface, to move the motors of a robot simply by increasing or decreasing the number that describes the motors current position. Motions are built up frame-by-frame - very similar to a story board in an animation sequence. This allows quite complicated ”animations” to be programmed and tested. Once a motion has been defined it can then be downloaded into the CM5 memory and called from the Behavior Control Program. Each pose stores a maximum of 30 joints information (position, velocity, and stop time).

That means that each pose can store the value the actuator can detect with its coders and thus the angle it has at the given pose. The acceleration with which the actuators will rotate in order to reach their goal position and finally the time -optional- for which the actuators will remain at the given position so as for the robot to maintain the given posture.

The user can also **define manually the pose of the robot** with a very easy procedure and then store it in one of the 7 poses of a motion page. The procedure for the manually setting of the actuators position is illustrated below :

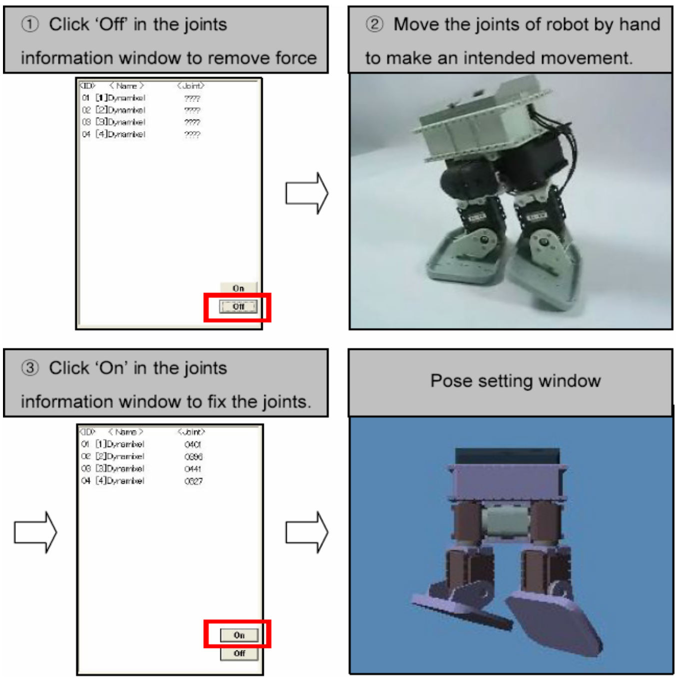


Figure 2.9: The Procedure for Manually Making Poses

The user can also connect or repeat edited motions. The numbers and poses that are shown on a motion editor screen are the information for a single motion page. A motion page is made up of 7 poses and 64 bytes of page information. The Bioloid has up to 127 motion pages. Several screen shots of the GUI can be found after the experimental verification in chapter 6.1

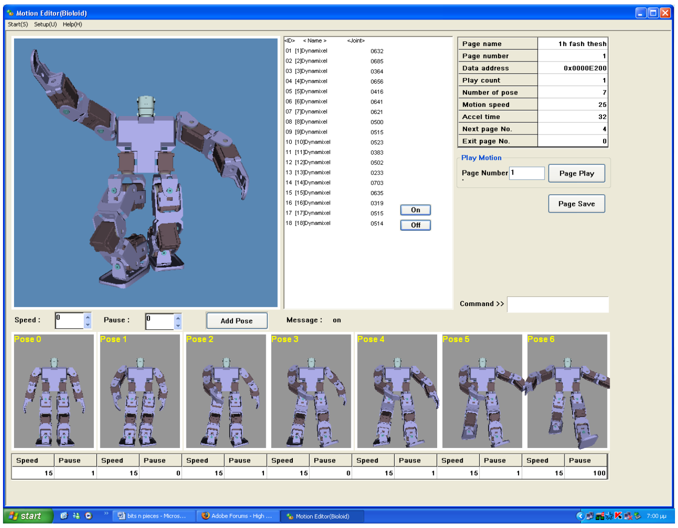


Figure 2.10: The Motion Editor GUI

- The Behaviour Control Programmer: The behavior control program is a series of rules programmed by the user, that define the action a robot should take for a given state. It also takes the form of a series of rules that mutually connect the input and output. With the AX-s1 dynamixel, the robot, given the appropriate set of rules can react to any external stimulate the sensor can detect, autonomously without the intervention of the user in real time. The result of a Behaviour Control Programmer file is similar to *a flow chart* with commands of behaviour according to external stimulates.

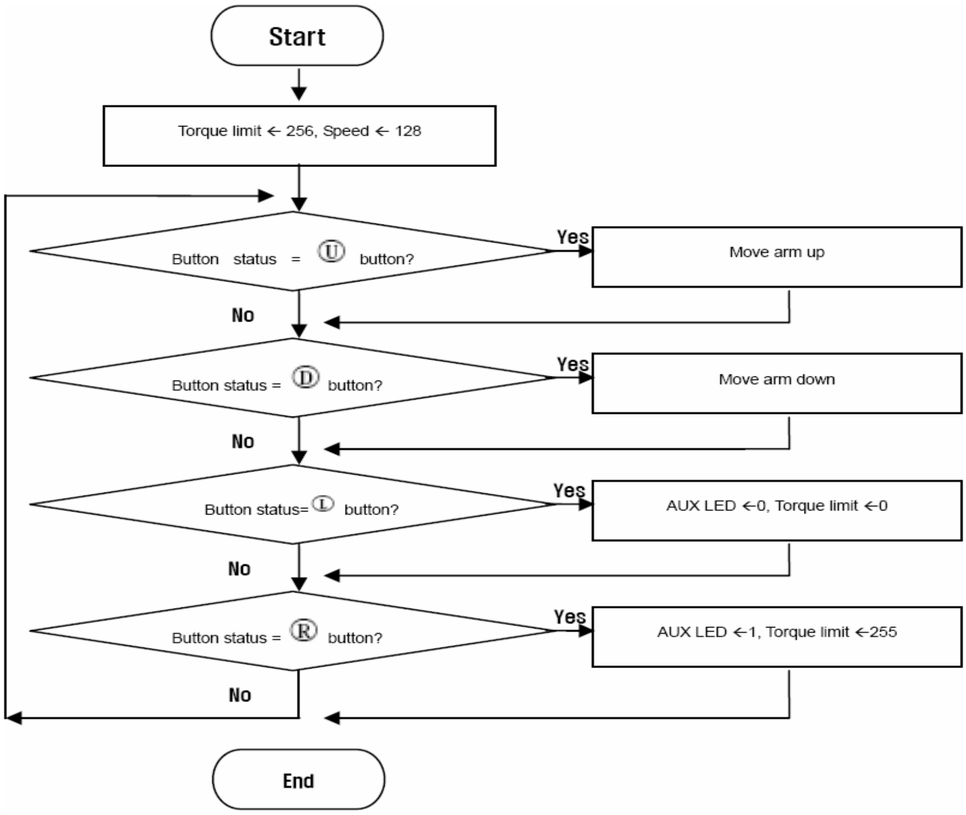


Figure 2.11: Typical Flow Chart Structure

The description of the different rules is done using a graphic based programming environment, in this way all motion defined using the Motion editor can be connected together depending on input from the sensors and programmatic logic.

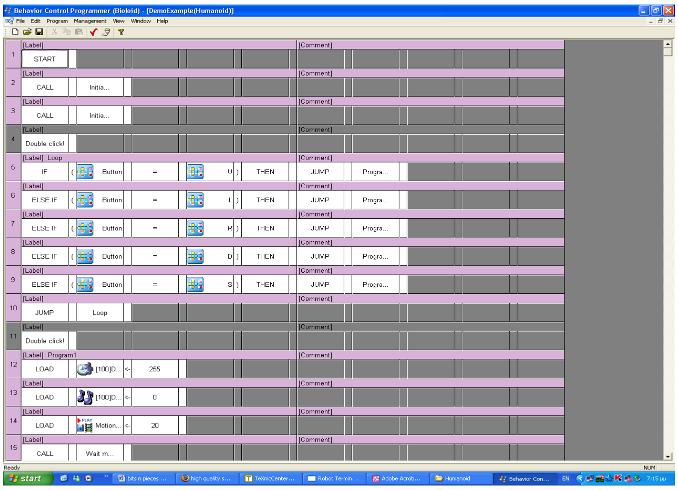


Figure 2.12: The Behaviour Control Programmer GUI

As we already mentioned, the The Behaviour Control Programmer acts as a a series of rules programmed by the user and is really much like flow chart. This program uses commands and Booleans in order to relate inputs with outputs according to the user's will. A table of the commands that the user can use to control the behaviour of the robot is the following :

Type	Command	Function
Operation	START	To indicate the beginning of a behavior control program.
	END	To indicate the end of a behavior control program.
	LOAD	To input various kinds of data.
	COMPUTE	To operate the four arithmetical operations and the logical operations.
Condition	IF	To operate the following command when the given condition is 'true'.
	ELSE IF	To come after IF and to operate the following command when the condition of IF is 'False' and the condition of ELSE IF is 'true'.
	ELSE	To come after IF or ELSE IF and to operate the command after ELSE when the both conditions of IF and ELSE IF are all 'False'.
	CONT IF	To distinguish the second IF from the first IF in the case that an IF command should be used again in another IF command.
Branch	JUMP	To move to a designated command.
	CALL	To call a designated command.
	RETURN	To return to the next row after CALL command after finishing the operation of command designated by CALL.

Figure 2.13: The Control Programmer Grammar

- The Robot Terminal: The Robot Terminal is a program that connects the CM-5 and the PC. The CM-5 does not have a screen or a keyboard, but the Robot Terminal program will allow you to input and output information using your the PC. The information outputted from the CM-5 will go to the PC through the serial cable and then printed on screen through the Robot Terminal. Also, information inputted in the Robot Terminal via the keyboard will be sent to the CM-5 through the serial cable.

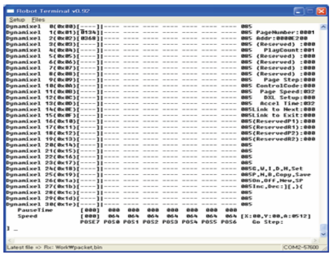


Figure 2.14: The Robot Terminal GUI

2.5 The Bioloid Humanoid

The Bioloid Humanoid is one of the most sophisticated of the robots that can be produced using the Bioloid Kit due to the large number of degrees of freedom it has. Also the option for interaction with its surroundings through the Ax-S1 sensor vastly increases the potentials of the Humanoid.

It's a rather complex robot to build as it has **a total of over 600 components** broken down to 20 cables 90 rigid frames and 508 nuts & bolts. A list with all of the components that are needed for assembling the Humanoid is shown below.

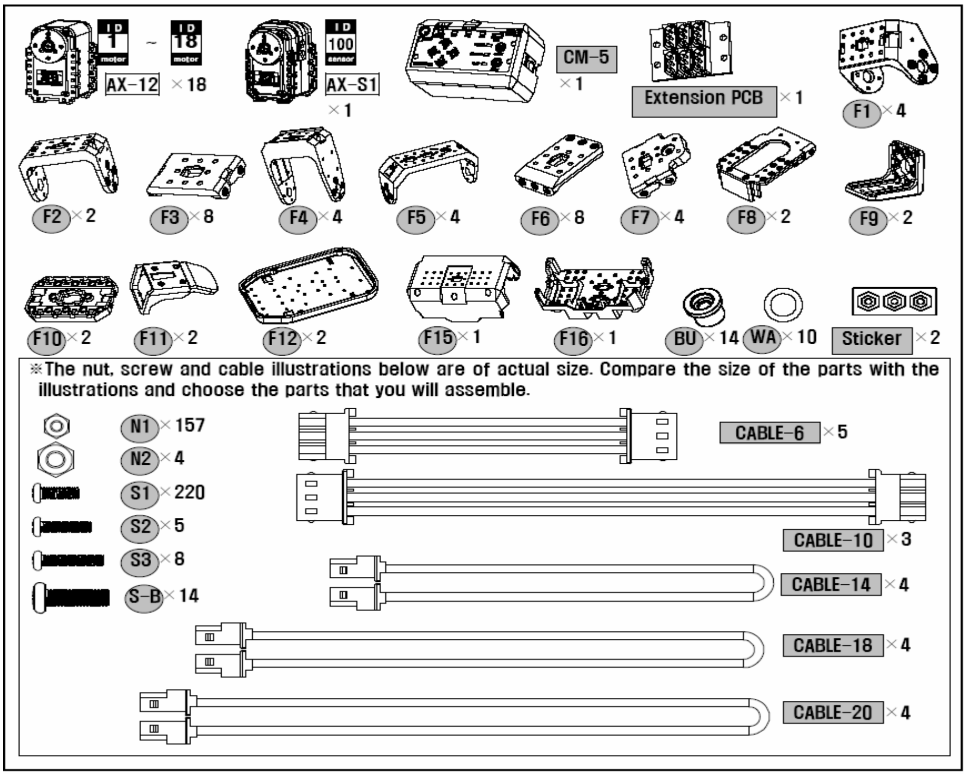


Figure 2.15: The HumanoidComponents

2.6 Robot designing With The Bioloid Kit

There are some factors that have to be taken into consideration during the assemble of a robot that can be a bit restrictive during the process of finding the exact shape of the robot. These factors mainly have to do with:

1. **The wiring.**

   Since the CM-5 unit can communicate with every actuator only in a serial way through every actuator that stands in between the processor and the target actuator the designing of a robot can be limited since the wires that come with the kit have only two fixed lengths.

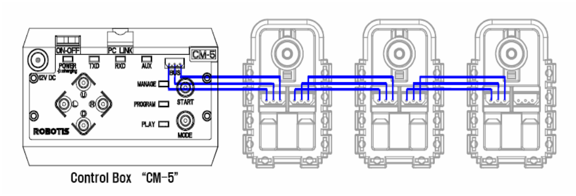


Figure 2.16: Serial Chain Connection of Cm-5 and AXs

2. **Stability issues.**

   When asseblying a robot that has to perform actions that require either static or dynamic balance, the designer has to take into consideration the volume of the unit that houses the processor as well as its weight since both the processor and the battery are within the same unit. Weight distribution can be of great importance in such cases and the geometrical features of the main unit can indeed be restrictive.

The concept of keeping it in the center:

The bioloid kit stands out when building robots that have a star-like structure. Star-like structure meaning that the main unit housing batteries and processor is located at the center of the robot and from which start a number of limbs according to its physical form. For example when eight limbs start from the center the structure which is the robot's main unit it is obvious that the robot will have the physical characteristics of a spider. By following *the concept of keeping the main unit in the center of the structure* appear a number of advantages -especially when dealing with animal-like robots-, some of which are:

- **Autonomous operation.**
  Every advanced non-fixed, portable robot constructed with the Bioloid kit should be able to operate without any external intervention from its user. By following the concept of keeping the main unit in the center of the structure there is no need to provide support to the main unit as the robot moves around.

- **Better Weight Distribution.**
  In order to apply this concept it is expected that the structure of the robot will be rather symmetrical. In that way the load that the actuators will have to deal with is distributed more evenly than in other

non-symmetrical structures, thus allowing the robot to have a more fluid motion. Moreover the available power-torque of the actuators is almost identical when comparing symmetrical joints so the implementation of every algorithm that has to do with movement is performed more successfully as the repeatability of routines is not affected by weight distribution. In that way the 'noise' produced by uneven weight distribution which affect the physical characteristics of the actuators (such as torque) is minimized.

Figure 2.17: SpiderBot and its Weight Distribution

2.7 Check assembly

Before we started operating the robot it was necessary to run a model check called "check assembly". With this procedure the user verify that every actuator is in the joint that it is meant to be according to the assembly instructions given for certain robots, in our case the humanoid. This is of crucial importance because:

- Every command that is stored in the CM-5 unit is directed to each actuator according to the actuator's unique ID number. As mentioned earlier if the main controller sends an instruction packet with the ID set to N, only the Dynamixel unit with this ID value will return its respective status packet and perform the required instruction. If multiple Dynamixel units have the same ID value, multiple packets sent simultaneously collide, resulting in communication problems. Thus, it is imperative that no Dynamixel units share the same ID in a network node.

- Matching the assembly's arrangement of actuators with the one given in the expert manual is almost essential since the powerful tool Motion Editor with the appropriate robot profile gives you the ability to visualize the humanoid's postures. Also when adjusting the values of the actuators, using the motion editor and not by switching of the actuator and doing it manually, the user can see in real time as he or she alters the values of each actuator how the humanoid changes its posture, which can be a great advantage when tampering with postures that are statically balanced and making slight modifications to explore the boundaries of the support polygon.

- Due to wrong assignment of offsets of the actuators the humanoid could not successfully perform the demo as it fell while performing a routine. This is not exclusively caused by wrong assignment of offsets, it could be caused by lack of firmness while assembling the humanoid due to insufficient screwing torgue applied at the bolts or by frames that were produced in the factory and not complying fully with the specifications given in the manual. Let us not forget that the components of the kit are all mass produced so it is possible that their dimensions may have divergences from the original model. So we reprogrammed the actuators giving them the proper offset as to achieve better static balance and after several experiments we found that a satisfying set of values for the initialization of the actuators is the following.

Motor ID	Actuator value
1	206
2	818
3	252
4	772
5	512
6	512
7	512
8	512
9	512
10	512
11	463
12	560
13	440
14	585
15	554
16	470
17	509
18	510

Figure 2.18: The Offset Table

How to Check Assembly

The Position with which the user can ‘run’ a provided by the ROBOTIS demo in the Behaviour control Programmer that can verify that the AXs are positioned according to the instructions given in the manual. That procedure is simply done by downloading the demo to the CM-5 and then, when executed, each actuator according to its ID activates its LED. Then every time the user presses a specific button of the CM-5 the next (according to its ID) Actuator’s LED is turned on. That way the user can locate the position of every actuator on the assembly and verify that it is placed according to the instructions of the manual.

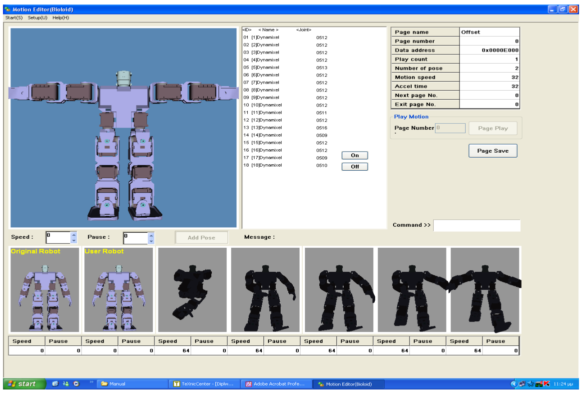


Figure 2.19: The check Assembly Mode

Having completed all of the above stages of preparation and once the initialization of the actuators has been performed the humanoid can be programmed and operated by its user.

Chapter 3

MODELLING OF THE BIOLOID HUMANOID

3.1 The Approach

A commonly used convention for selecting frames of reference in robotics applications is **the Denavit and Hartenberg (D-H) convention** introduced by Jaques Denavit and Richard S. Hartenberg. Denavit and Hartenberg used screw theory in the 1950's to show that the most compact representation of a general transformation between two robot joints required four parameters. In this convention, each homogeneous transformation is represented as a product of four basic transformations. These four parameters now known as the Denavit and Hartenberg parameters (D-H parameters) are the de-facto standard for describing a robot's geometry. The common normal between two lines was the main geometric concept that allowed Denavit and Hartenberg to find a minimal representation.

For the modeling of the Bioloid Humanoid we decided to follow an approach similar to the Denavit - Hartenberg convention.

The whole system would be described using local coordinate systems and each time the position of the following would be described in relevance to the position of the base or so called the system origin's coordinate system. As with the DH convention we examine joints with two links and we always have a follower and a base. The position of the CoM of the follower is described in an algebraic manner using homogeneous transformation matrices. Also after some matrix multiplications our goal is to describe the position of every link's CoM in relevance to the system origin (the right foot's CoM).

However there are some just a few minor differences when compared to the D-H approach. For the description of the system we placed the coordinate systems on the CoM of each link in an articulation, whereas the D-H convention places a coordinate system on every joint of the system under examination. Also the orientation of each coordinate system in the D-H convention is placed on the joint and oriented following certain rules.

The orientation we followed for the placement of every local coordinate system is following only one rule. That of minimization of calculations. As you will see at the right foot - whose center of mass we have stated as the origin of the base coordinate system, according to which the overall CoM is positioned - we have placed a coordinate system parallel to the sagital frontal and transversal plane of the Humanoid which is randomly positioned as far as the positive direction of the axes are concerned.

- The X- axis is parallel to the saggital plane and perpendicular to the coronal plane of the humanoid.
- The Y- axis parallel to the coronal plane and perpendicular to the saggital plane of the humanoid.
- The Z-axis parallel to the coronal plane and perpendicular to the transversal plane of the humanoid.

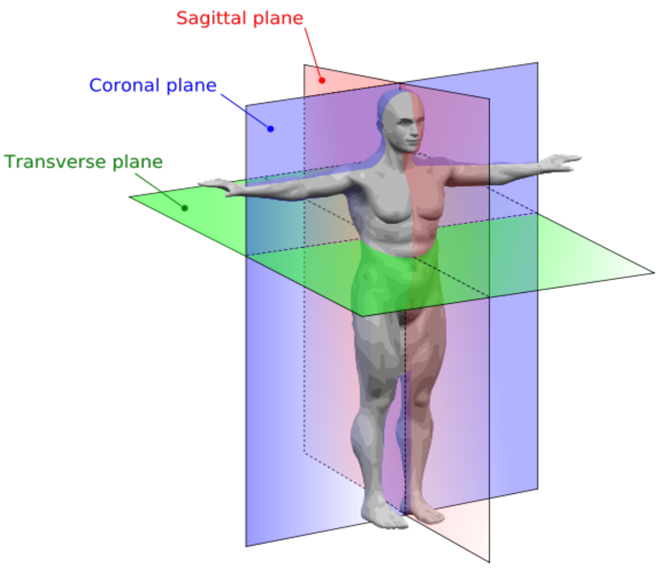


Figure 3.1: The Anatomy Planes In Robotics

We placed the robot at a pose where all of the actuators have 0° ,(identical to the check assembly pose 2.7) and then we created multiple copies of that particular coordinate system and place them on the CoM of every link. The previous procedure defined the placement and orientation of all of the coordinate systems of the humanoid.

- CALCULATE WITH GREAT ACCURACY THE CoM
- VISUALISE ITS POSITION WITHIN THE SPACE

At this point it might seem to the reader that the visualisation part is of minor importance, however the complexity with which changes take place and the interaction of factors that form these changes make the visualization if not necessary, at least a great asset to have. However not only the visualization of the CoMs was useful. The visualization of the assignment of a unique coordinate system in each link was also very helpful, as during the creation of sub assemblies it was much easier for the user to define the direction of the rotation of a joint as well as the rotation axis so as to clarify the rotation matrix that had to be used in the algebraic transformation in order to define the position of each coordinate system according to the previous one.

Moreover had we not used the software with all of its benefits, then THE LEVEL OF PRECISION would drammaticaly decrease and the validity of the equations would be jeopardized. Taking into consideration that since the position of each CoM is calculated in respect to the previous one and then all of the CoMs are calculated in respect to the coordinate system situated on the right foot -whose contact area with the ground has been defined as the support polygon-, we can understand that every deviation from the actual position of a CoM in the real model is carried through the whole set of equations. Once the calculation of a CoM according to the support polygon takes place then the deviation is added equal times to the number of links that interfere between the right foot and the link under examination. In other words every deviation constitutes a prosthetic error to the equation that describes the position of the link under examination in relevance to the right foot. The closest to the right foot the link with the deviation is the more the following equations are affected.

3.2 The Solidworks 3D Models

For the calculation of the CoMs we used a library of 3d models of every part of Bioloid's kit that was created at the University Of Lisbon IST as a part of a master thesis done by Pedro Teodoro[8]. The models used are of extreme detail and accuracy and by using the software's libraries they are assigned with the material properties the real components have. Also they are accurately diamensionalized according to the real components. In that way it is ensured that the 3d models have the same mass properties with the real components and therefore the center of mass of the models is situated within the space at the same point where the CoM of the real component is situated.

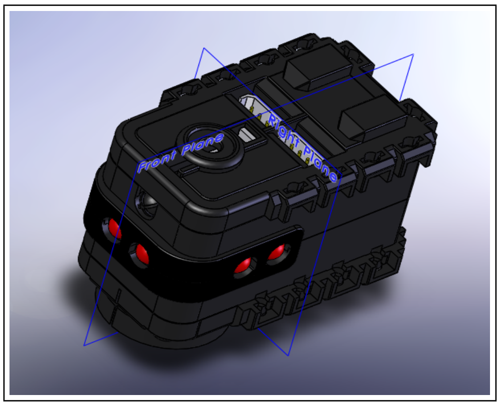


Figure 3.2: AxS1 Screenshot from Solidworks

3.3 Major Assemblies and Subassemblies

In order to proceed with the examination of the CoMs we had to perform the 3d assembly of certain components of the Bioloid using Solidworks. In order to examine two links that form a joint so as to create the equation that describes the motion of the second one in respect to the first one we had to know the geometrical features of each CoM of every one of the two links.That is, the X Y Z distances between: either the two CoMs, or certain fixed points (most usually the rotation axis of the joint under examination). So, even though all of the parts from the AXs1, the dynamixel AX-12, to the whole frame set down to the last nut and bolt were 'ready and cut', the assembly still had to be done and it was no trivial task.

The 3d models provided by the University of Lisbon were indeed the essential models a user would find within the bioloid kit. However **no assembly** of the 3d models had been done. For example the actuator model did not even have the rotating wheel assembled to it. We actually had to assemble the robot from scratch as if we were to build it in reality. *Right to the last bolt and nut.*

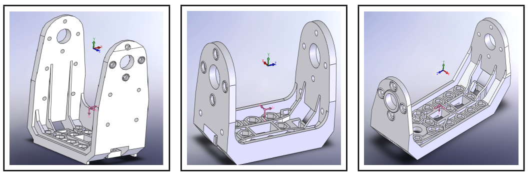


Figure 3.3: Most Commonly Used Types of Brackets

Every single component had to be oriented and placed within the virtual space of solidworks, a procedure that demanded at least three commands for every model (one for the positioning of the component in each dimension). Taking into consideration that the **complete assembly which inludes a total of nearly 500 pieces**, then roughly 3×500, 1500 commands and about 15 hours were needed only for the particular assembly. In picture 3.3 is an exploded view of the arm, an assembly with an average number of 50 components.

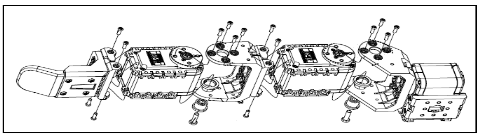


Figure 3.4: Hand Assembly

In picture 3.3 we can see a photorealistic representation of the total assembly with over 500 components.

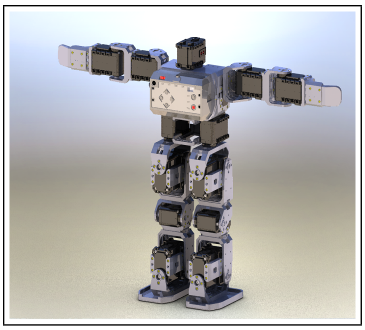


Figure 3.5: The Total Assembly Rendered in Solidworks

At this point we have to underline the fact that a large number of assemblies had to be made in order to:

- create the sub assemblies that would constitute the **links** and perform the calculation of the CoM of each one.

- create the **joints** made of two links each in order to visualize the relative position of the second center of mass according to the first one.

- create the mechanological drawings of both **links** and **joints** in solidworks so as to have a visualization of how the equations were constructed and also to specify and clearly show the relations of the mathematical figures with the dimensioning of the 3d models.[1]

For the above reasons we had to make a TOTAL OF 42 LARGE ASSEMBLIES. The number of each component required for the creation of each assembly can be found at the 'links' section, since there is BOM report (Bill Of Materials) at each mechanological design of every link. An example of one of the largest subassemblies is the torso of the humanoid with almost 80 components (we changed the transparency of the cover of the CM-5 so that the internal components can be visible):

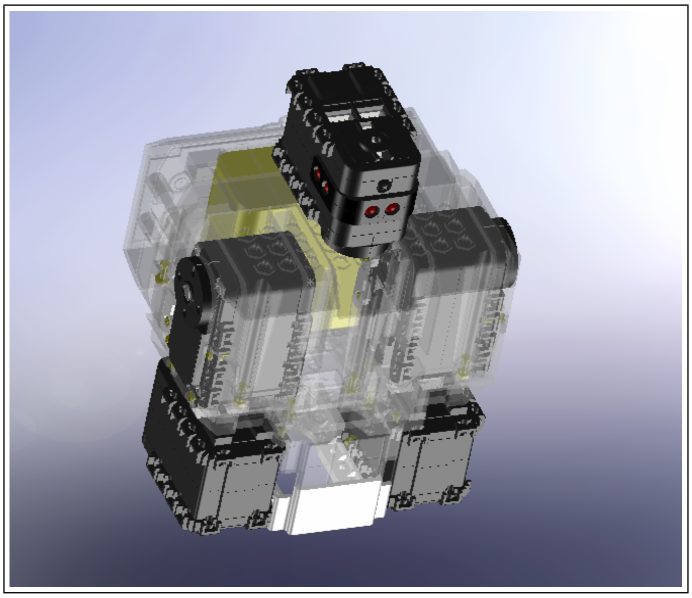


Figure 3.6: A Screenshot of the major Assembly 'Torso'

[1]that way the reader can have a intuitive understanding of what each figure of the equation represents and therefore what happens to the joint when there are angular differentions.

3.4 Marking The Position of CoMs

The procedure followed to mark the position of the CoM of each component was the following: After the calculation of the CoM in relevance to the assembly origin, we created 3 planes that were parallel to the ones of the assembly origin for every dimension and were placed in the x-, y-, and z-coordinates of the point defined by the COM calculation distances from the yz-plane, xz-plane, and xy-plane respectively. In that way the intersection of these three planes pinpointed the COM.

3.5 Point Of Origin

With this procedure we insured the marking of the CoM's position. However in order to continue with the assignment of the coordinate systems, it was essential to create a sketch in the 'sketch environment' so as to use it as a reference point for positioning the coordinate system.To do so, we selected one of the CoM planes that where described before, on which the point of reference would be. Then, having specified the one of the three dimensions, we defined that the point would be on the intersection of the two remaining planes. That way we had a fully specified point in the virtual space of solidworks that would later be used as a reference point for the positioning of every component's unique coordinate system.

3.6 Assignment and Orientation of Coordinate Systems

Once we had precisely defined a plane for every dimension we had to assign a unique coordinate system for each link that would be the point of reference for the description of the position and the displacement of the COM of the following link. Having fully defined a point in the virtual space which would be the beginning of the axes of the local 4.1.3 coordinate system the only thing remaining was the orientation of the coordinate system. The orientation of a coordinate system in Solidworks is done firstly by selecting an axis and then by defining the plane on which the selected axis will be perpendicular to. Thus, having defined the point of origin the procedure we described before had to be performed only for two out of the three axes in order to fully define the orientation of the local coordinate system.

3.7 The Drawings In Solidworks

For the making of the drawings, we used once again the Solidworks 2008 software. Utilizing the option 'Make drawing from assembly' a sheet is created on which the user can place certain views of an existing 3d model.

- Opening a drawing template and editing a sheet format This is the main layout of the drawing. Containing fixed cells in which the user can input information concerning the drawing. The name of drawing and model, copyrights, number of present sheet and remaining sheets, information concerning the model such as weight e.t.c. There are available preset layouts of the sheet from which the user can choose.

- Inserting standard views of a part model This is the visual part of the sheet, containing the views of the drawing (side, perspective e.t.c). There are available preset views to be imprinted on the drawing. For example the standard 3-view option gives the front side and floor plan of the model.The user can create his/her own views using the standard views and creating projections of them. The user can also rotate every view, that way almost every desired view can be produced. Also with the editing of a view the user can choose the quality of the drawing draft or high.

- Adding model and reference annotations We used this commands in order to fill the area that contains the information of the drawing. It includes the model's dimensions, visible planes (custom and default), coordinate systems, origins, axes of rotations and additional information related to the drawing such as notes. This area was the most important for our drawings not only due to the dimensioning of the model but also for the viewing of the local coordinate systems on every link and every joint and also for the viewing of the CoM of every link and the follower's relative position according to the previous - base link when referring to joints.

Chapter 4

FORWARD KINEMATICS

Kinematics is the study of the geometry of a mechanical system, where the motion of the system can be described in terms of the velocity and acceleration of all its components. The components can be connected through different types of joints, which limit how the components can move relative to each other. The interconnection of the components implies that the motion of the components relative to each other is constrained. Newtonian mechanics state that the forces acting on a system change the motion of the system. The forces can be divided into constraint forces, that limit (or constrain) the motion, and generalized forces, that cause the motion. Kinematics describe the interconnection of the components and the constraints without regard to the constraining forces and defines the motion of the interconnected bodies through space without regard to the generalized forces that cause a motion.

It is assumed here that each component of the mechanical system can be treated as a rigid body. A rigid body can be defined as a component for which the distance between any two points in the component is fixed, i.e. the points can not move relative each other.

The forward position kinematics problem can be stated as follows: given the different joint angles, what is the the position of the end-effector? Direct kinematics refers to the calculation of the end effector position -or any other point of interest-, orientation, velocity and acceleration when the corresponding joint values are known. Keeping that in mind, the answer to the statement of the problem is rather simple: The solution lies in the construction of the different transformation matrices and their combination - multiplication. The result being ${}^{i}T_0$, where 0 is generally the base frame of the robot manipulator and i the link under examination. Multiplication of these matrices leads to the complete transformation of any link's position in respect to the base frame, which solves the direct kinematics problem. For our case specifically it will be the link we have stated as the point of reference according to which all the other links' position will be described.

$$^{0}T_n = \prod_{i=1}^{n} {}^{i-1}T_i(\theta_i) \tag{4.1}$$

However in order to calculate through the multiplication of matrices the position of a given coordinate system of a random link of a robot in relevance to the coordinate system that we have assigned as reference or origin of the Humanoid, it is essential to evaluate each coordinate system according to its previous one. To do so we have to keep in mind that there are two kinds of "dislocations" that can take place: 1) Rotation, 2) distance normal to an axis or a plane.

An example of a matrix that evaluate a coordinate system's position according to its previous one is the following:

$$^{n-1}T_n = Trans_{z_{n-1}}(d_n) \cdot Rot_{z_{n-1}}(\theta_n) \cdot Trans_{x_n}(a_n) \cdot Rot_{x_n}(\alpha_n) \tag{4.2}$$

Regarding the matrices of the distances we observe that the equations that give the normal distance can be found in the last column of the matrix produced by all of the rotation and displacements of the second local coordinate system according to the first which is of similar type to the 4.3 equation result. An example of a matrix that gives the distance normal to an axis of the second coordinate system according to the first one is given below:

$$Trans_{z_{n-1}}(d_n) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_n \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

An example of a matrix that gives the rotation of the second coordinate system according to the first one is given below,although there is a detailed description in section 4.2:

$$Rot_{z_{n-1}}(\theta_n) = \begin{pmatrix} \cos\theta_n & -\sin\theta_n & 0 & 0 \\ \sin\theta_n & \cos\theta_n & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.1 Coordinate Systems

4.1.1 The need of Coordinate systems In Robotics

The generalized coordinates (or configuration coordinates) determine the geometric configuration of the mechanical system. This means that any point in the mechanism can be specified by giving the values of the generalized coordinates. The generalized coordinates can be chosen in different ways where the choice is dependent on, for example, which coordinates are of interest, for instance due to placement of sensors, or simplification of the equations. The minimum number of generalized coordinates that specify the configuration of the mechanism is called the geometric degree of freedom. A kinematic motion of the mechanism is determined by specifying all the generalized coordinates as function of one single variable, for example time, and thereby generating a curve for the motion of all points in the mechanism. The coordinates have to be given relative to a frame, usually a cartesian coordinate system, that is fixed relative to the earth[6].

4.1.2 Definition

The three dimensional Cartesian coordinate system provides the three physical dimensions of space - length, width, and height. The three Cartesian axes defining the system are perpendicular to each other. The point of intersection, where the axes meet, is called the *origin* normally labeled O. The x and y axes define a plane that is referred to as the xy plane. The relevant coordinates are of the form (x,y,z). The x-, y-, and z-coordinates of a point can also be taken as the distances from the yz-plane, xz-plane, and xy-plane respectively. The xy-, yz-, and xz-planes divide the three-dimensional space into eight subdivisions known as octants, similar to the quadrants of 2D space. While conventions have been established for the labeling of the four quadrants of the x-y plane, only the first octant of three dimensional space is labeled. It contains all of the points whose x, y, and z coordinates are positive.

4.1.3 Global & Local coordinate systems

In geometry and kinematics, coordinate systems are used not only to describe the (linear) position of points, but also to describe the angular position of axes, planes, and rigid bodies. In the latter case, the orientation of a second (typically referred to as 'local') coordinate system, fixed to the node, is defined based on the first (typically referred to as 'global' or 'world' coordinate system). For instance, the orientation of a rigid body can be represented by an orientation matrix, which includes, in its three columns, the Cartesian coordinates of three points. These points are used to define the orientation of the axes of the local system. They are the tips of three unit vectors aligned with those axes. In order to describe the rotations efficiently and without confusion it was essential to come up with a convention. The convention used for the rotation around the axes is the following :

- Roll for X - axis
- Pitch for Y - axis
- Yaw for Z - axis

The positive direction of the rotation around each axis can be emerged from the following figure showing the orientation that we used for ALL of the Humanoid's coordinate systems. The coordinate system that is shown in picture 4.1.3, is applied on the center of mass of each link oriented exactly as shown and applied to the particular pose of the Humanoid that is shown.

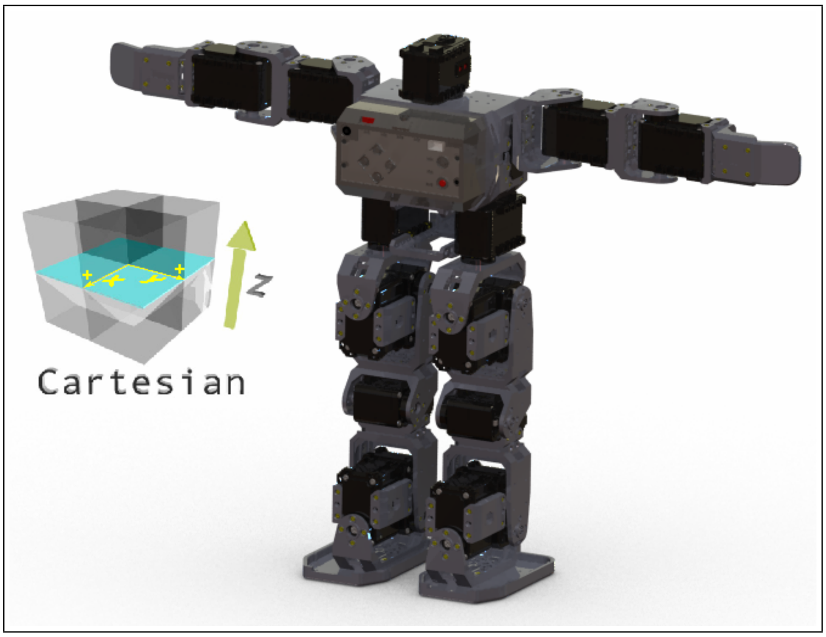


Figure 4.1: The Coordinate System Convention Used on all of the CoM at this Pose

A coordinate transformation is a conversion from one system to another, to describe the same space.With every bijection from the space to itself two coordinate transformations can be associated:

- such that the new coordinates of the image of each point are the same as the old coordinates of the original point (the formulas for the mapping are the inverse of those for the coordinate transformation)
- such that the old coordinates of the image of each point are the same as the new coordinates of the original point (the formulas for the mapping are the same as those for the coordinate transformation)

4.2 Rotation Matrices

By knowing the geometrical features of every part of the bioloid humanoid and using the 3d software Solidworks we were able to create a set of equations that, in a couple of links, describe the position of the CoM of the second one according to the coordinate system of the first one that is situated on the CoM of the first one. Having presented all of the equations that define the positions of all the centers of mass we had to incorporate them into an overall center of mass that would represent the whole system so as to be able to check if the projection of the total center of mass on the -xy plane is situated within the support polygon. In order to unify the 18 previous equations to a single one that would describe the overall center of mass and taking in consideration the rotation of each coordinate system we had to use some algebra.

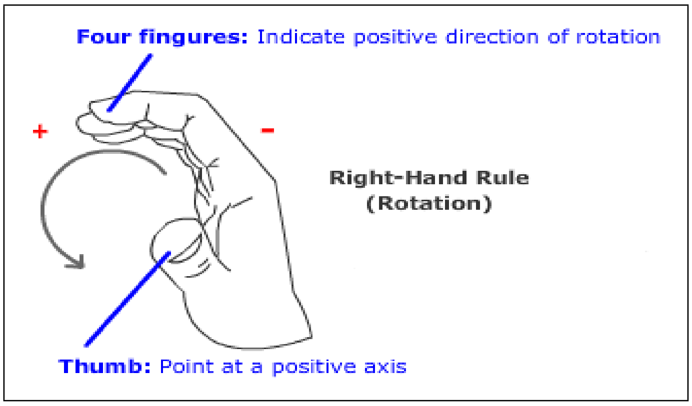


Figure 4.2: The Right hand Rule

A transformation matrix can be used to describe the location and orientation of a second coordinate system relative to a first coordinate system. We apply the transformation matrix to the origin and the endpoints of the unit vectors of the first coordinate system. This matrix multiplication produces the origin and the endpoints of the unit vectors of the second coordinate system.

A homogenous transformation matrix consists of *four* main sectors.

$$\mathbf{{}^{n}T_{n+1}} = \begin{pmatrix} & X = f_x(\theta) \\ Rot(3 \times 3) & Y = f_y(\theta) \\ & Z = f_z(\theta) \\ 0^T & 1 \end{pmatrix}$$

4.2.1 Rotation In Accordance to the RHR

The following matrices are used when the the increase of the actuator's value leads to a positive direction of the rotation axis according to the right hand rule. That is when the four fingers of the right hand excluding the thumb follow the direction of rotation when the thumb points at the positive direction of the axis of rotation.

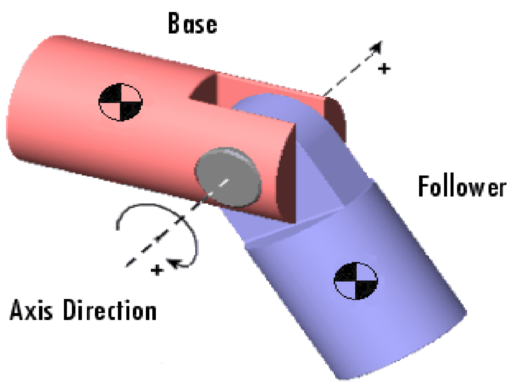


Figure 4.3: Rotation In Accordance to the RHR

$$\textbf{Rotation around X axis} = \begin{pmatrix} 1 & 0 & 0 & X = f_x(\theta) \\ 1 & cos(\theta) & -sin(\theta) & Y = f_y(\theta) \\ 1 & sin(\theta) & cos(\theta) & Z = f_z(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\textbf{Rotation around Y axis} = \begin{pmatrix} cos(\theta) & 0 & sin(\theta) & X = f(\theta) \\ 0 & 1 & 0 & Y = f(\theta) \\ -sin(\theta) & 0 & cos(\theta) & Z = f(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\textbf{Rotation around Z axis} = \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 & X = f(\theta) \\ sin(\theta) & cos(\theta) & 0 & Y = f(\theta) \\ 1 & 0 & 1 & Z = f(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.2.2 Rotation Opposed to the RHR

The following matrices are used when the the increase of the actuator's value leads to a positive direction of the rotation axis according to the right hand rule. That is when the four fingers of the right hand excluding the thumb follow the direction of rotation when the thumb points at the negative direction of the axis of rotation.

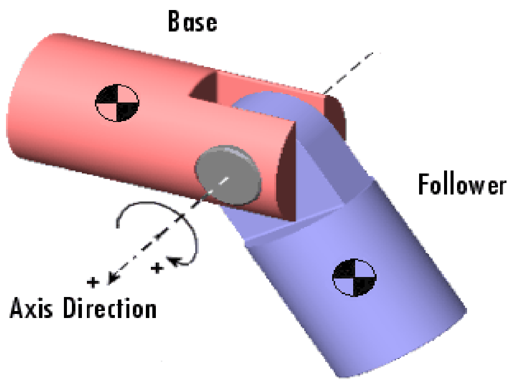


Figure 4.4: Rotation Opposed to the RHR

$$\textbf{Rotation around X axis} = \begin{pmatrix} 1 & 0 & 0 & X = f(\theta) \\ 1 & cos(\theta) & sin(\theta) & Y = f(\theta) \\ 1 & -sin(\theta) & cos(\theta) & Z = f(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\textbf{Rotation around Y axis} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & X = f(\theta) \\ 0 & 1 & 0 & Y = f(\theta) \\ sin(\theta) & 0 & cos(\theta) & Z = f(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\textbf{Rotation around Z axis} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = f(\theta) \\ -sin(\theta) & cos(\theta) & 0 & Y = f(\theta) \\ 1 & 0 & 1 & Z = f(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.3 THE EQUATIONS

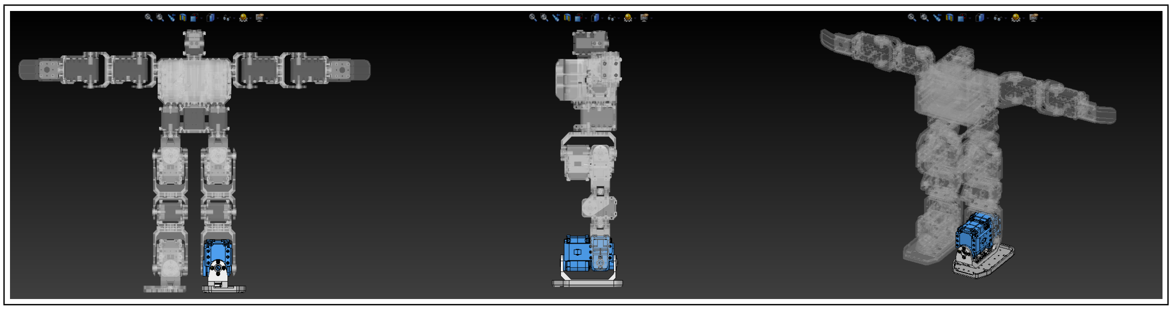
The equations used for the description of the position of each CoM in respect to the rotation of the actuators mounted on each joint derived from a very meticulous observation of the movement of the joints and links. There were four main *factors* that had to be taken into consideration in order to form the equations that describe the Bioloid's movement:

- The rotation of the actuator (clockwise or counterclockwise rotation)
- The axis around which the rotation takes place
- The unique coordinate system of the link under examination
- The initial position of the CoM under examination

4.4 Right Leg

4.4.1 JOINT 1

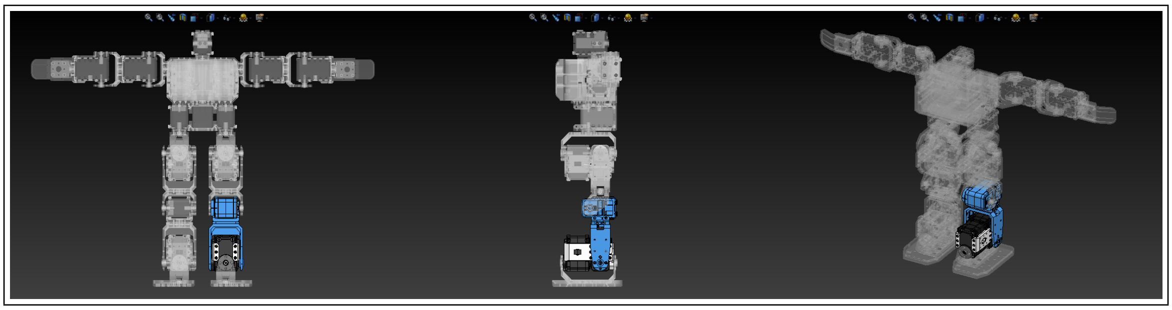
- Connecting Right Foot & Right Ankle -



$$
{}^{1}\mathbf{T_2} = \begin{pmatrix} 1 & 0 & 0 & X = -0.96 \\ 1 & cos(\theta) & sin(\theta) & Y = -4.37 + 12.9728 * sin(\theta \ - 1.1925°) \\ 1 & -sin(\theta) & cos(\theta) & Z = 24.63 + 12.9728 * cos(\theta \ - 1.19258°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.4.2 JOINT 2

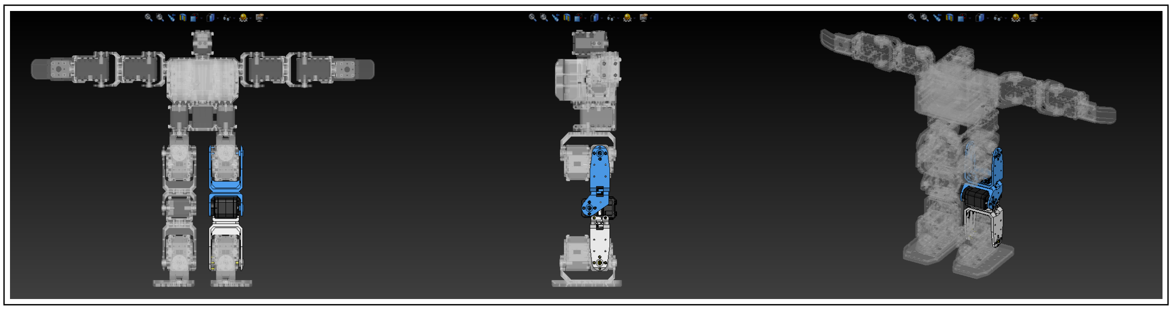
- Connecting Right ankle & Right Knee -



$$
{}^{2}\mathbf{T_3} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & X = -15.82 - 60.5479 * sin(\theta \ - 0.9274°) \\ 0 & 1 & 0 & Y = -0.06 \\ sin(\theta) & 0 & cos(\theta) & Z = -12.97 + 60.5479 * cos(\theta \ - 0.9274°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.4.3 JOINT 3

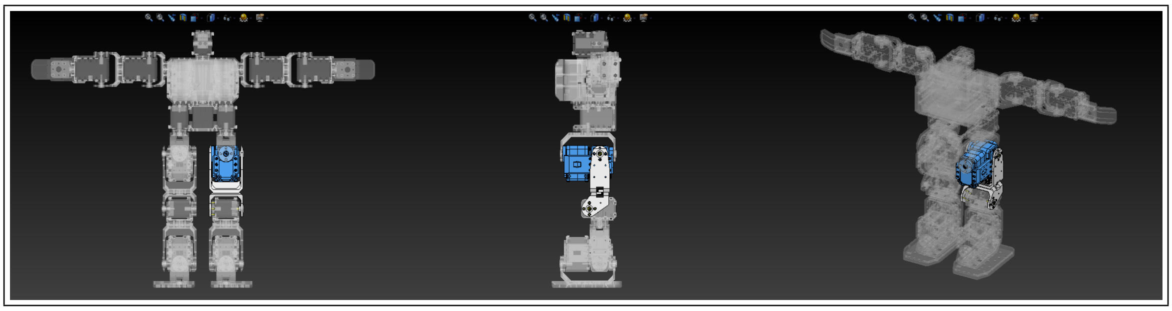
- Connecting Right Knee & Right Thigh -



$$
{}^{3}\mathbf{Y_4} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & X = 13.77 - 34.25 * sin(\theta \ + 22.686°) \\ 0 & 1 & 0 & Y = 0.69 \\ sin(\theta) & 0 & cos(\theta) & Z = 14.96 + 34.25 * cos(\theta \ + 22.686°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.4.4 JOINT 4

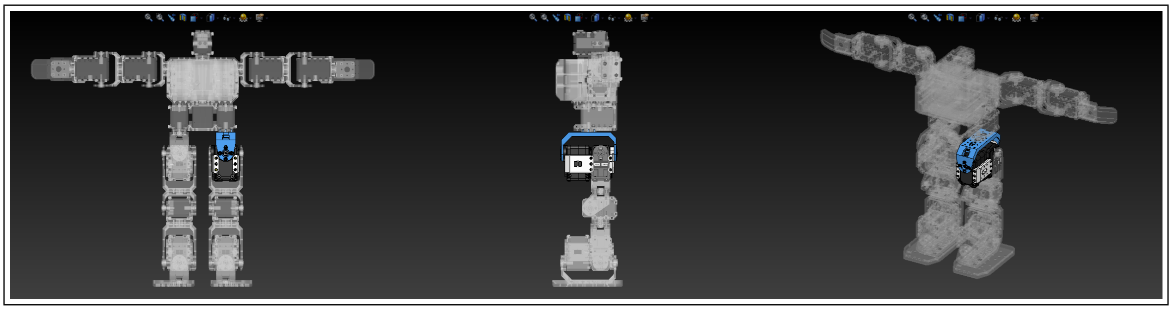
- Connecting Right Thigh & Right Hip -



$$
{}^{4}\mathbf{T_5} = \begin{pmatrix} cos(\theta) & 0 & sin(\theta) & X = -1.79 + 20.457 * cos(\theta \ + 39.346)°) \\ 0 & 1 & 0 & Y = -0.63 \\ -sin(\theta) & 0 & cos(\theta) & Z = 43.9 - 20.457 * sin(\theta \ + 39.346°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.4.5 JOINT 5

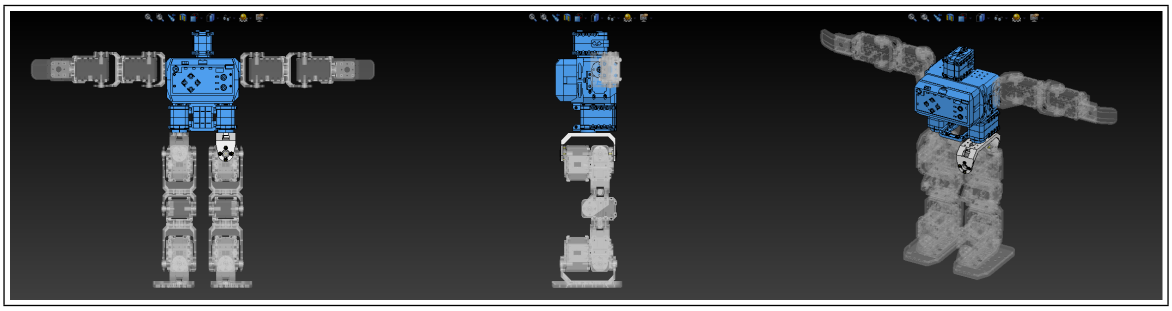
- Connecting Right Hip & Right Hip Support -



$$
{}^{5}\mathbf{T_6} = \begin{pmatrix} 1 & 0 & 0 & X = 0.11 \\ 1 & cos(\theta) & -sin(\theta) & Y = 0.26 - 17.72 * sin(\theta) \\ 1 & sin(\theta) & cos(\theta) & Z = 12.97 + 17.72 * cos(\theta) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.4.6 JOINT 6

- Connecting Right Hip Support & Torso -

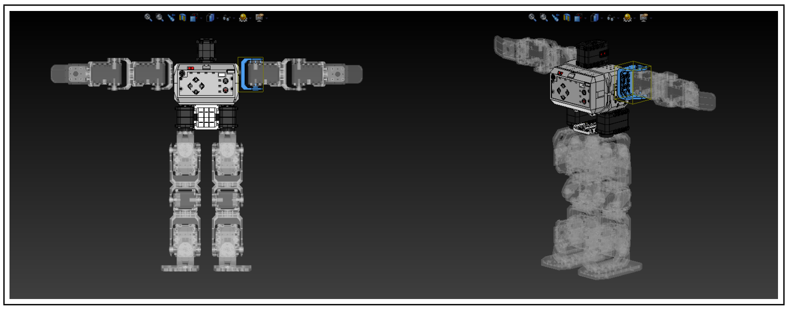


$$
{}^{6}\mathbf{T_7} = \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 & X = 1.07 + 32.447 * sin(\theta \ + 7.758°) \\ sin(\theta) & cos(\theta) & 0 & Y = -32.447 * cos(\theta \ + 7.758°) \\ 1 & 0 & 1 & Z = 77.29 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.5 Right Hand

4.5.1 JOINT 7

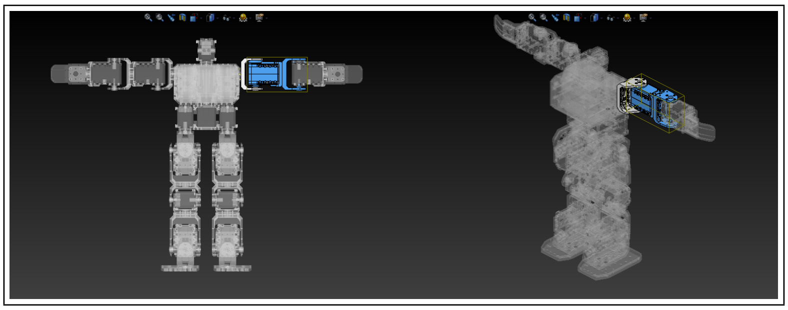
- Connecting Torso & Right Shoulder -



$$^{7}\mathbf{T_8} = \begin{pmatrix} cos(\theta) & 0 & sin(\theta) & X = -20.38 - 5,1089 * cos(\theta \ - 4.94°) \\ 0 & 1 & 0 & Y = 64.67 \\ -sin(\theta) & 0 & cos(\theta) & Z = 20.49 + 5.1089 * sin(\theta \ - 4.94°) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.5.2 JOINT 8

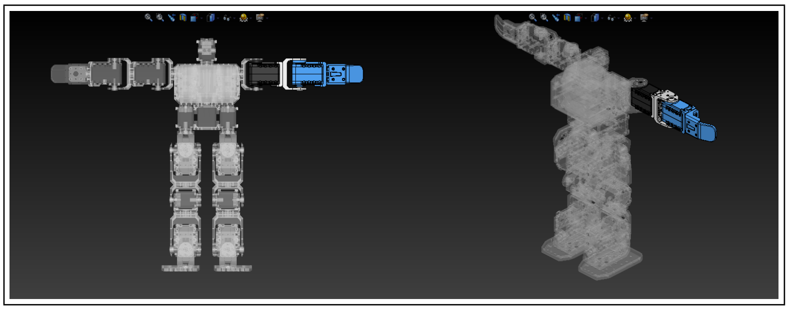
- Connecting Right Shoulder & Right Elbow -



$$^{8}\mathbf{T_9} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = -9.91 + 23.86 * sin(\theta) \\ -sin(\theta) & cos(\theta) & 0 & Y = 11.49 + 23.86 * cos(\theta) \\ 1 & 0 & 1 & Z = -0.79 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.5.3 JOINT 9

- Connecting Right Elbow & Right Hand -

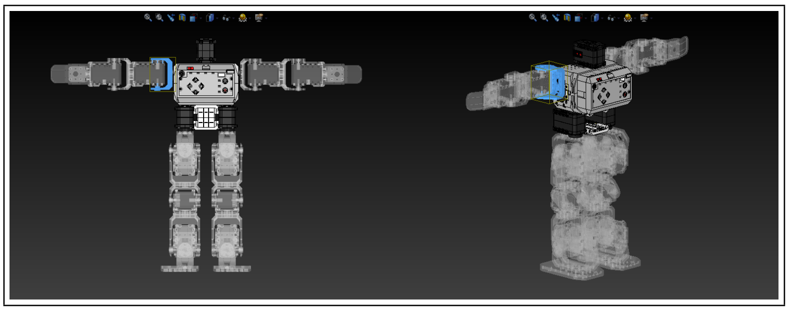


$$
{}^{9}\mathbf{T}_{10} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = 24.5 * sin(\theta \ + 2.9476°) \\ -sin(\theta) & cos(\theta) & 0 & Y = 44.14 + 24.5 * cos(\theta \ + 2.9476°) \\ 1 & 0 & 1 & Z = -0.1 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.6 Left Hand

4.6.1 JOINT 10

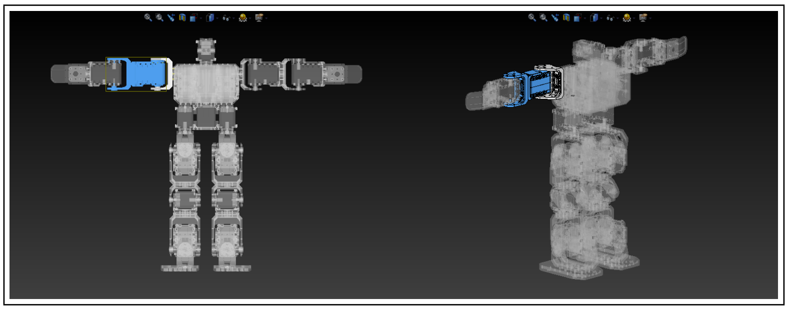
- Connecting Torso & Left Shoulder -



$$
{}^{10}\mathbf{T}_{11} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & X = -20.63 - 5.11 * cos(\theta \ + 5.02°) \\ 0 & 1 & 0 & Y = -65.36 \\ sin(\theta) & 0 & cos(\theta) & Z = 20,99 - 5.11 * sin(\theta \ + 5.02°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.6.2 JOINT 11

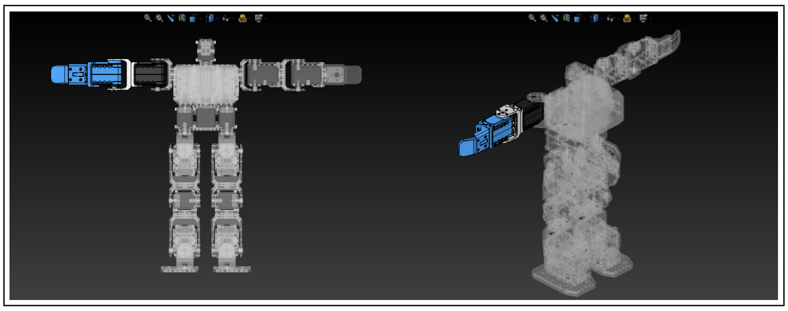
- Connecting Left Shoulder & Left Elbow -



$$
{}^{11}\mathbf{T}_{12} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = -9.91 - 23.86 * sin(\theta) \\ -sin(\theta) & cos(\theta) & 0 & Y = -11.49 - 23.86 * cos(\theta) \\ 1 & 0 & 1 & Z = 0.79 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.6.3 JOINT 12

- Connecting Left Elbow& Left Hand -

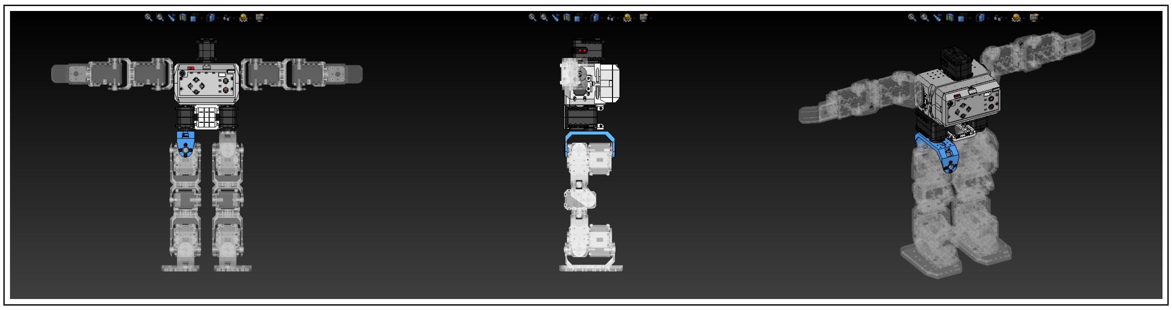


$$
{}^{12}\mathbf{T}_{13} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = -24.5 * sin(\theta \ - 2.94°) \\ -sin(\theta) & cos(\theta) & 0 & Y = -0.1 \\ 1 & 0 & 1 & Z = -44.14 - 24.5 * cos(\theta \ - 2.94°) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.7 Left Leg

4.7.1 JOINT 13

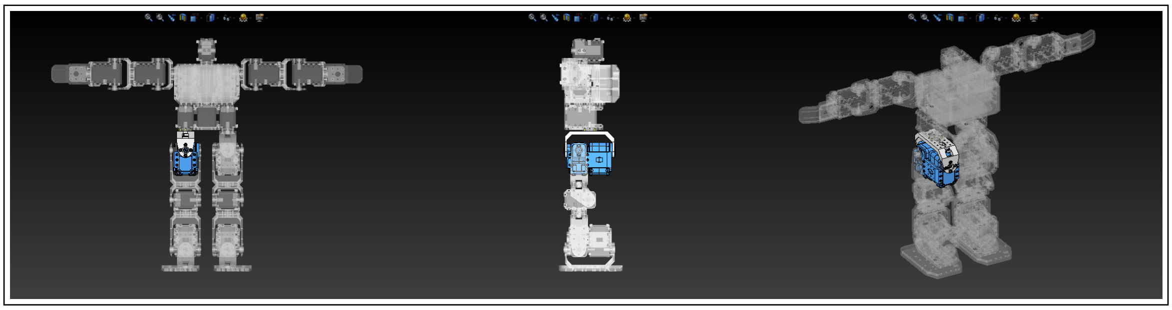
- Connecting Torso& Left Hip Support -



$$^{13}\mathbf{T}_{14} = \begin{pmatrix} cos(\theta) & sin(\theta) & 0 & X = -4.38 - 1.07cos(\theta) \\ -sin(\theta) & cos(\theta) & 0 & Y = -33.85 + 1.07 * sin(\theta) \\ 1 & 0 & 1 & Z = -77.29 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.7.2 JOINT 14

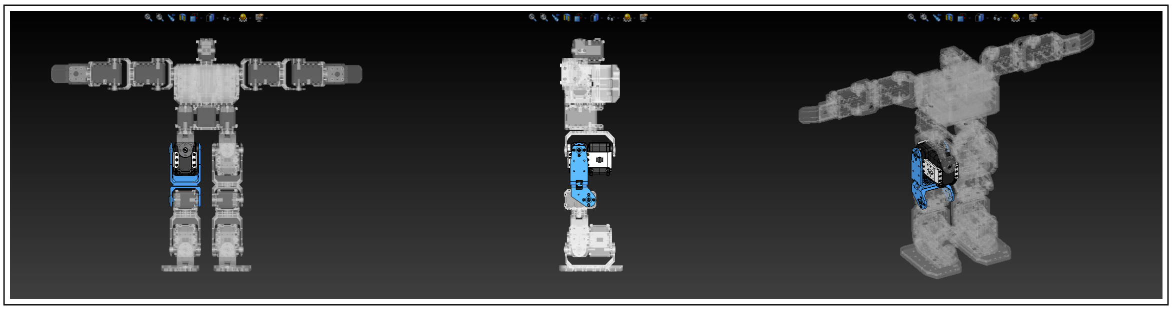
- Connecting Left Hip Support & Left Hip -



$$^{14}\mathbf{T}_{15} = \begin{pmatrix} 1 & 0 & 0 & X = -0.11 \\ 1 & cos(\theta) & sin(\theta) & Y = -12.97 * sin(\theta \ - 1.19°) \\ 1 & -sin(\theta) & cos(\theta) & Z = -17.72 - 12.97 * cos(\theta \ - 1.19°) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.7.3 JOINT 15

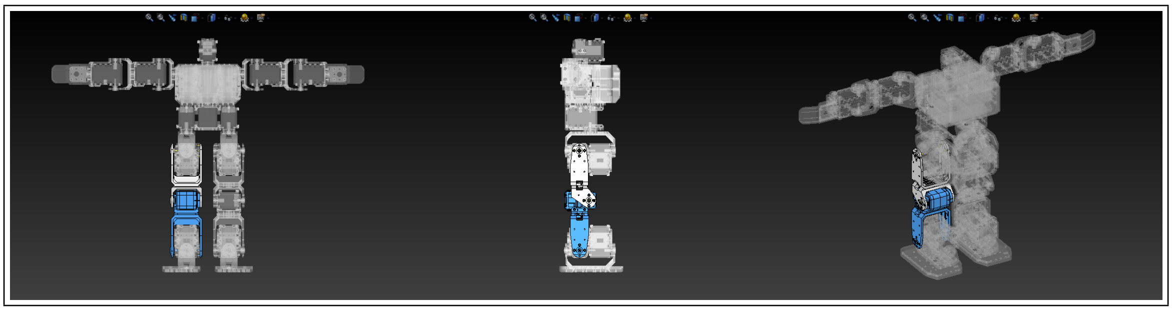
- Connecting Left Hip & Left Thigh -



$$
{}^{15}\mathbf{T}_{16} = \begin{pmatrix} cos(\theta) & 0 & sin(\theta) & = 15.82 + 43.93 * sin(\theta \ - 2.32^\circ) \\ 0 & 1 & 0 & Y = -0,63 \\ -sin(\theta) & 0 & cos(\theta) & = 12.97 + 43.93 * cos(\theta \ - 2.32^\circ) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.7.4 JOINT 16

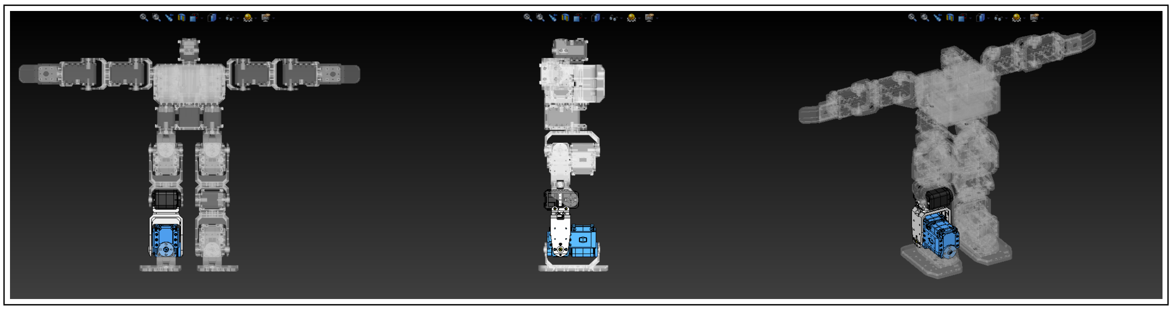
- Connecting Left Thigh & Left Knee -



$$
{}^{16}\mathbf{T}_{17} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & = 13.22 + 20.33 * sin(\theta \ - 42.63^\circ) \\ 0 & 1 & 0 & = 0,7 \\ sin(\theta) & 0 & cos(\theta) & Z = -31.6 - 14.96 * cos(\theta \ - 42.63^\circ) \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$

4.7.5 JOINT 17

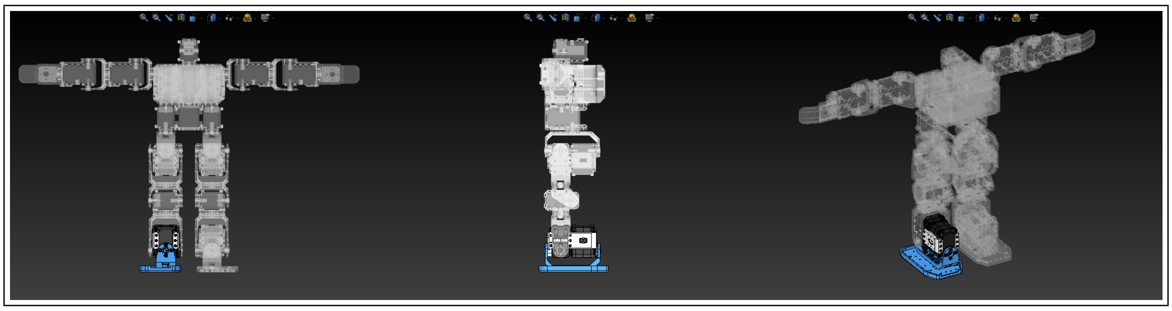
- Connecting Left Knee & Left Ankle -



$$^{17}\mathbf{T_{18}} = \begin{pmatrix} cos(\theta) & 0 & -sin(\theta) & X = -0,98 + 20,46 * cos(\theta \ + 39.35°) \\ 0 & 1 & 0 & Y = -0.06 \\ sin(\theta) & 0 & cos(\theta) & = -60,54 + 20,46 * sin(\theta \ + 39.35°) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.7.6 JOINT 18

- Connecting Left Ankle & Left Foot -



$$^{18}\mathbf{T_{19}} = \begin{pmatrix} 1 & 0 & 0 & X = 0.95 \\ 1 & cos(\theta) & sin(\theta) & Y = -0.26 - 25 * sin(\theta \ + 9.94°) \\ 1 & -sin(\theta) & cos(\theta) & = -12,97 - 25 * cos(\theta \ + 9.94°) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Chapter 5

STATIC BALANCE

5.1 Static Balance

5.1.1 Definition

A standard definition of static equilibrium is: When a system of forces acting on an object produces no motion, the system is said to be in static equilibrium. Static balance means the construction is at all time in balance.

McGhee, and Frank (1968) define static stability [5]as ‘An ideal legged locomotion machine is statically stable at time t if all the legs in contact with the support plane at the given time remain in contact with that plane when all the legs of the machine are fixed at their location at time t and the translational and rotational velocities of the resulting rigid body are simultaneously reduced to zero’. However the above definition of static stability by McGhee, and Frank (1968), is highly idealized as it assumes that the feet can provide unlimited forces, and does not treat external forces.[6]

5.1.2 The Problem of Static balance

During a static balanced movement such as walking the normal projection of the center of mass (NPCM) always stays in between the boundaries defined by the feet [1]. If both feet are on the ground, the NPCM has to be within the polygon determined by the outer corners of the biped feet. If only one foot is in contact with the ground, the NPCM has to be within the area of this foot. While the movement is slow enough, the system dynamics can be ignored. Static balance assumes that if the system’s motion is stopped at any time, it will stay in a stable position indefinitely.

5.2 Center Of Mass

5.2.1 Definition

In physics, the center of mass of a system of particles is a specific point at which, for many purposes, the system's mass behaves as if it were concentrated. The center of mass is a function only of the positions and masses of the particles that comprise the system. In the case of a rigid body, the position of its center of mass is fixed in relation to the object (but not necessarily in contact with it). It is is the balance point of a system, a point where you could consider the mass of the whole system to be concentrated.

In a system which has movable joints, i.e. the system can move or be moved into different configurations or positions, the location of the center of mass depends on the position-pose of the system. For example, when a figure skater lifts his or her arms during a spin, the mass of the arms is higher up in the system than when the arms were at the side of the skater. With more mass of the skater distributed in a higher position in the body, the center of mass of the skater rises to a higher position in the body.

Finally we present some properties of the center of mass:

1. The center of mass is the location where all of the mass of the system could be considered to be located.

2. For a solid body it is often possible to replace the entire mass of the body with a point mass equal to that of the body's mass. This point mass is located at the center of mass.

3. For homogeneous solid bodies that have a symmetrical shape, the center of mass is at the center of body's symmetry, its geometrical center.

4. The center of mass is the point about which a solid will freely rotate if it is not constrained.

5. For a solid body the center of mass is also the balance point. The body could be suspended from its center of mass and it would not rotate, i.e. not be out of balance.

6. The center of mass of a solid body does not have to lie within the body. The center of mass of a hula-hoop is at its center.

7. The center of mass for a system of independently moving particles still has meaning and is useful in analyzing the interactions between the particles in the system.

5.2.2 Calculation Of The CoM

The calculation of the CoM of an object can be approached in different ways depending on the geometrical features of the body and its mass properties. For example if an object has uniform density then its center of mass is the same as the centroid of its shape. If the body is symmetrical and has uniform density then its center of mass is the point of symmetry. However excluding these cases that can be found rather rarely in a real problem we will present the procedure of the calculation of CoM of a body that is not symmetrical and then we will present the most general case the one of a non symmetrical body with no uniform density.

The calculation of CoM of a body that is not symmetrical can be performed in two ways. The first one involves the partial examination of sections of the body that are symmetrical shapes and the second and more general that refers to the situation where the body can not be fully splitted into symmetrical shapes so the approach demands the use of integration.

For a continuous distribution with mass density p(r)and total mass M, the sum becomes an integral:

$$R = \frac{1}{M}\int r dm = \frac{1}{M}\int p(r) r dV = \frac{\int p(r) r dV}{\int p(r) dV} \tag{5.1}$$

Finally the calculation of the center of mass R of a system of n particles is defined as the average of their positions r_i, weighted by their masses m_i. Since in our case the particles are the components of the humanoid this is the formula we used for the calculation of the overall center of mass of the humanoid:

$$R = \frac{\sum_{i=1}^{n} m_i r_i}{\sum_{i=1}^{n} m_i} \tag{5.2}$$

5.3 Support Polygon

5.3.1 Definition

According to Timothy Bretl[2] on flat horizontal terrain, a necessary condition for balance is that the robots CoM lie above the base of its supports. This region is called the support polygon. The support polygon is a term used in robotics when dealing with matters of either static or dynamic equilibrium. As support polygon we characterize the area the foot of a biped robot (as in our case) that comes in contact with the ground when the robot is in an equilibrium state.

5.3.2 Dimensioning of the Support Polygon

When we have only one feet touching the ground then as support polygon we can define the surface that may come in contact with the ground. However this is not quite valid if both feet are on the ground. At this situation the support polygon's size is defined by the outer corners of the biped feet which seems to be different from the previous definition since the area in between the feet is not in contact with the ground in any way, but it still belongs to the surface of the support polygon.

Chapter 6

EXPERIMENTAL VERIFICATION

6.1 Real vs. Simulation Environment

6.1.1 The Figures

The equations used for the description of the position of each CoM in respect to the rotation of the actuators together with all of the calculations for the position of CoMs were done using figures that derived from ideal models created in absolute compliance with the manufacturer's specifications. However the reality as we know is by far different from an experimental environment where most parameters are controlled and ever number is an ideal representation of a figure.

6.1.2 Production Quality

The manufacturer's poor quality is indeed one of the main disadvantages of this kit. Even though the Bioloid kits software is very user friendly and provides great freedom to the user to easily create the desired routines and interaction of the robot with external stimulates, the quality of the fitting of the robot components is dissapointing. The fact that the fitting of the assembly's components depends on coupling made of industrial plastic makes the total assembly unstable. The joints even if they are screwed firmly still seem to be loose creating an overall lack of rigidness which makes the robot's stability while moving vulnerable to oscillations. So even if for example a walking algorithm is accurate and fast when it will be implemented in the robot it might not produce the expected results.

6.1.3 Power Supply

A crucial factor that might create serious problems while proceeding from the theoretical to the experimental stage is the power supplied to the dynamixels. The charging of the battery with which the humanoid is equipped demands approximately five hours, while the battery is drained in only under thirty minutes. From these thirty minutes the robot can operate providing the dynamixels with the power they demand in order to deliver the torque they can for only a fragment of this time. Lack of power means lack of control. If the actuators do not have the power to move the joints to the desired position as calculated by the control algorithm then the balance, static or dynamic is jeopardized.

For the proper interpretation of the experiments' result by the user we have to give the dimensions of the support polygon. That way the the reader will be able to see the verification of the results as all the poses are statically balanced and therefore the position of the total CoM is found at all times within the support polygon. With the demo the viewer can interpret that some of the poses shown in the pictures and the results of the position of the total CoM in every pose are found near the limits of the support polygon. Should the user try to place the Humanoid while in a pose in which the total CoM is near the limits of the support polygon he/she will understand it due to the micro-oscillations until the achievement of stabilisation as the system is ready to reach instability.

The Dimensions Of the Support Polygon :

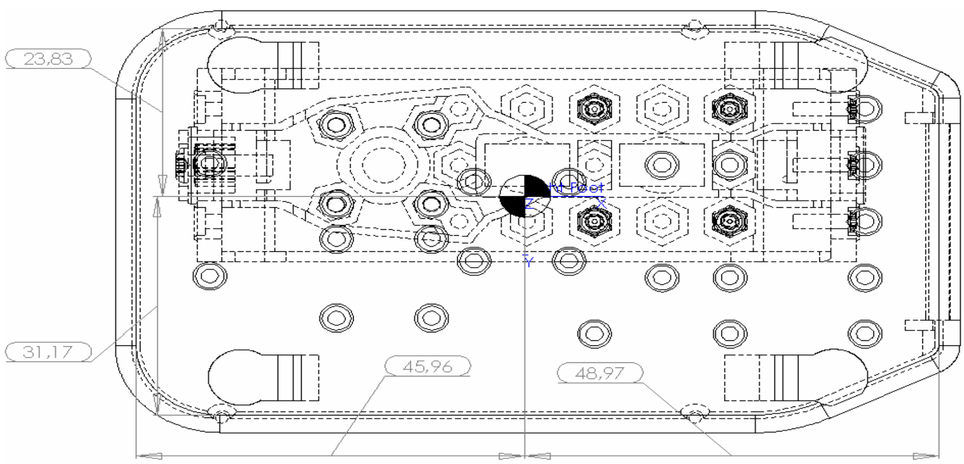
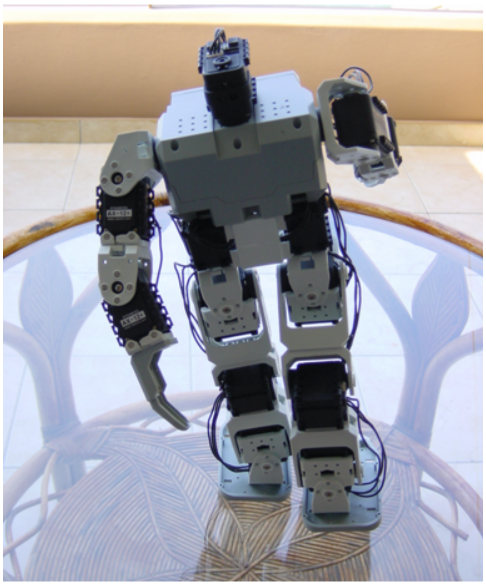


Figure 6.1: The Support Polygon's Dimensions

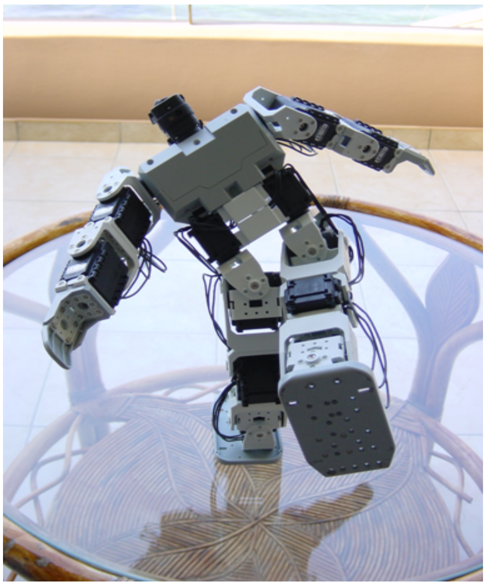
6.2 Crossed Hands

Position of total CoM according to system Origin		
X	Y	Z
-25,60492056	3,618325366	197,4468121



Joint Number	Motor ID	Joint Agle
Joint 1	17	$12,31°$
Joint 2	15	$10,26°$
Joint 3	13	$-14,95°$
Joint 4	11	$-13,48°$
Joint 5	9	$-7,33°$
Joint 6	7	$-0,29°$
Joint 7	1	$-49,53°$
Joint 8	3	$-80,00°$
Joint 9	5	$-24,62°$
Joint 10	2	$138,32°$
Joint 11	4	$72,09°$
Joint 12	6	$20,51°$
Joint 13	8	$6,45°$
Joint 14	10	$-0,59°$
Joint 15	12	$50,99°$
Joint 16	14	$42,20°$
Joint 17	16	$-2,93°$
Joint 18	18	$2,05°$

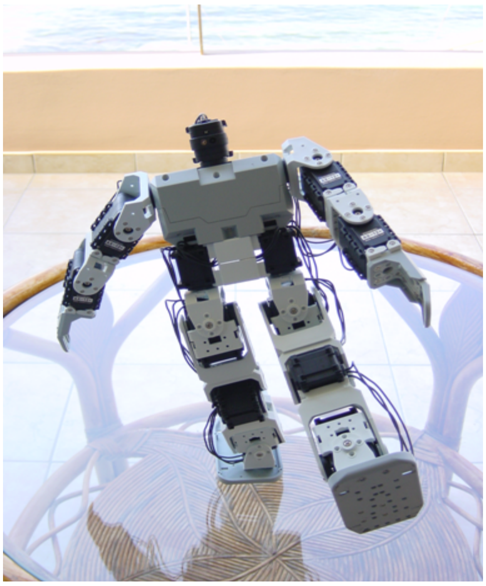
Position of total CoM according to system Origin		
X	Y	Z
-14,32552015	-20,85964158	200,6832611



Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$2,05°$
Joint 2	15	$20,81°$
Joint 3	13	$-32,82°$
Joint 4	11	$12,02°$
Joint 5	9	$-7,33°$
Joint 6	7	$-0,29°$
Joint 7	1	$-30,48°$
Joint 8	3	$-55,09°$
Joint 9	5	$-24,62°$
Joint 10	2	$130,70°$
Joint 11	4	$51,87°$
Joint 12	6	$20,51°$
Joint 13	8	$16,41°$
Joint 14	10	$-7,33°$
Joint 15	12	$64,76°$
Joint 16	14	$1,17°$
Joint 17	16	$-11,14°$
Joint 18	18	$-0,59°$

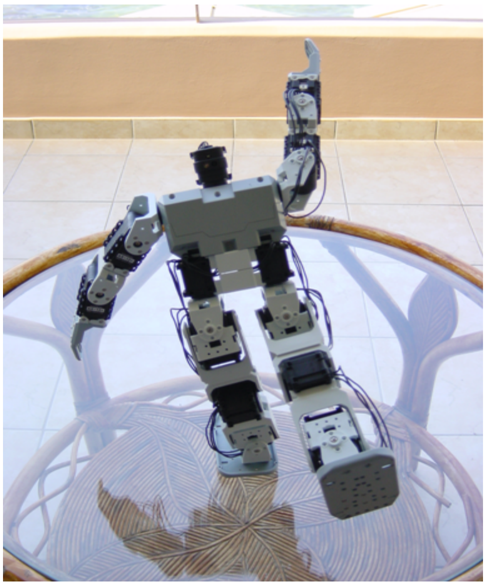
-

Position of total CoM according to system Origin		
X	Y	Z
-15,35914259	1,758325704	197,1620323



Joint Number	Motor ID	Joint Agle
Joint 1	17	$12,89°$
Joint 2	15	$17,00°$
Joint 3	13	$-27,55°$
Joint 4	11	$10,26°$
Joint 5	9	$-1,47°$
Joint 6	7	$-0,29°$
Joint 7	1	$-120,44°$
Joint 8	3	$-44,84°$
Joint 9	5	$-18,46°$
Joint 10	2	$43,66°$
Joint 11	4	$75,61°$
Joint 12	6	$0,00°$
Joint 13	8	$-1,47°$
Joint 14	10	$8,79°$
Joint 15	12	$52,46°$
Joint 16	14	$5,57°$
Joint 17	16	$-12,31°$
Joint 18	18	$-0,59°$

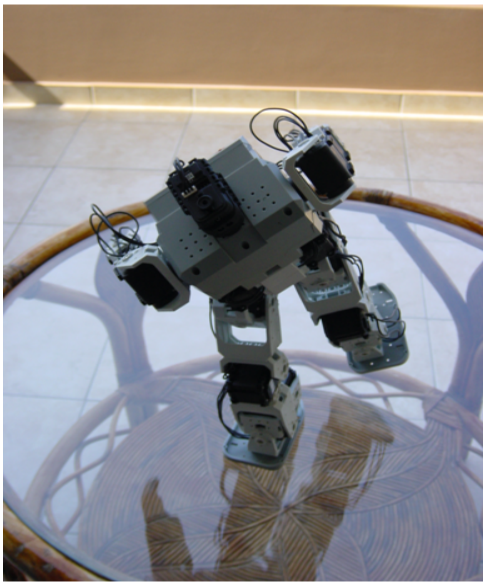
Position of total CoM according to system Origin		
X	Y	Z
-6,158049753	3,147890193	202,4270694



Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$12,89°$
Joint 2	15	$17,00°$
Joint 3	13	$-27,55°$
Joint 4	11	$10,26°$
Joint 5	9	$-1,47°$
Joint 6	7	$-0,29°$
Joint 7	1	$-120,44°$
Joint 8	3	$-44,84°$
Joint 9	5	$-18,46°$
Joint 10	2	$-120,44°$
Joint 11	4	$75,61°$
Joint 12	6	$0,00°$
Joint 13	8	$-1,47°$
Joint 14	10	$8,79°$
Joint 15	12	$52,46°$
Joint 16	14	$5,57°$
Joint 17	16	$-12,31°$
Joint 18	18	$-0,59°$

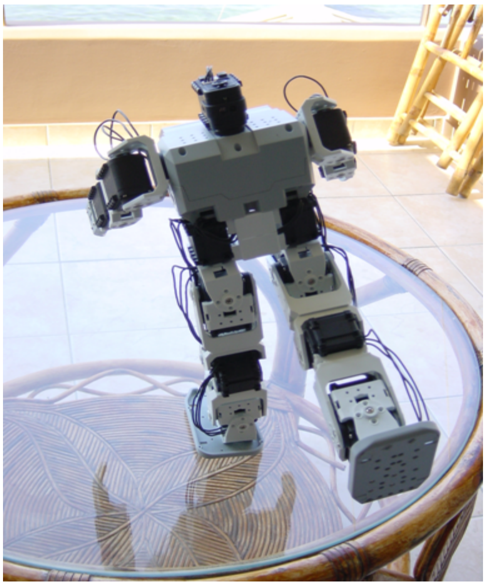
6.3 Hands Down Behind Torso

Position of total CoM according to system Origin		
X	Y	Z
-22,26914026	0,276380666	209,9611351



Joint Number	Motor ID	Joint Agle
Joint 1	17	$6,15^\circ$
Joint 2	15	$13,19^\circ$
Joint 3	13	$-21,10^\circ$
Joint 4	11	$5,44^\circ$
Joint 5	9	$-5,57^\circ$
Joint 6	7	$51,28^\circ$
Joint 7	1	$-142,42^\circ$
Joint 8	3	$-76,78^\circ$
Joint 9	5	$-0,29^\circ$
Joint 10	2	$136,27^\circ$
Joint 11	4	$82,64^\circ$
Joint 12	6	$0,00^\circ$
Joint 13	8	$25,20^\circ$
Joint 14	10	$-0,88^\circ$
Joint 15	12	$27,55^\circ$
Joint 16	14	$27,25^\circ$
Joint 17	16	$-11,14^\circ$
Joint 18	18	$-0,59^\circ$

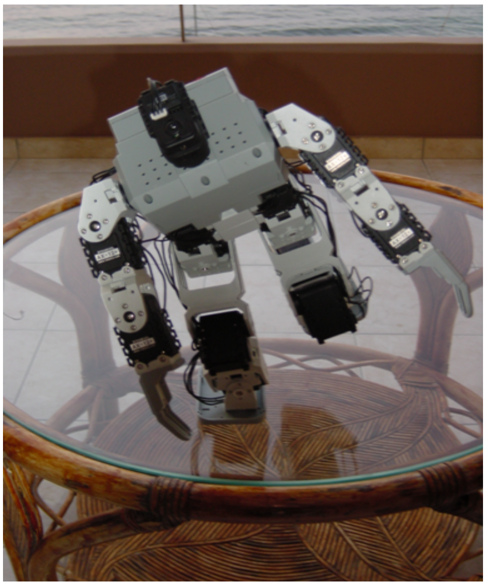
Position of total CoM according to system Origin		
X	Y	Z
-20,69595588	1,696564034	207,6409263



Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$9,96°$
Joint 2	15	$15,53°$
Joint 3	13	$-20,22°$
Joint 4	11	$0,88°$
Joint 5	9	$-12,31°$
Joint 6	7	$-0,29°$
Joint 7	1	$-149,46°$
Joint 8	3	$-76,78°$
Joint 9	5	$-0,29°$
Joint 10	2	$149,75°$
Joint 11	4	$75,61°$
Joint 12	6	$0,00°$
Joint 13	8	$0,59°$
Joint 14	10	$0,00°$
Joint 15	12	$68,87°$
Joint 16	14	$3,22°$
Joint 17	16	$-9,08°$
Joint 18	18	$-0,59°$

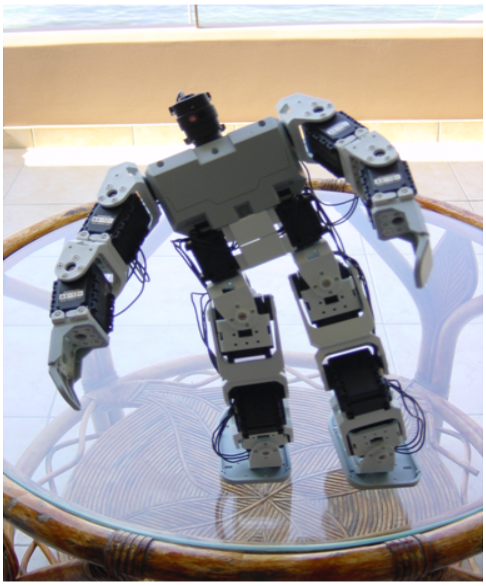
6.4 Hands Down In Front Of the Torso

Position of total CoM according to system Origin		
X	Y	Z
-27,96762543	-1,737139288	197,6508102



Joint Number	Motor ID	Joint Agle
Joint 1	17	$10,84°$
Joint 2	15	$26,37°$
Joint 3	13	$-60,66°$
Joint 4	11	$-66,23°$
Joint 5	9	$-13,19°$
Joint 6	7	$-21,39°$
Joint 7	1	$7,66°$
Joint 8	3	$-84,11°$
Joint 9	5	$-1,76°$
Joint 10	2	$10,68°$
Joint 11	4	$75,61°$
Joint 12	6	$0,00°$
Joint 13	8	$0,29°$
Joint 14	10	$-3,22°$
Joint 15	12	$78,83°$
Joint 16	14	$117,22°$
Joint 17	16	$-58,61°$
Joint 18	18	$1,17°$

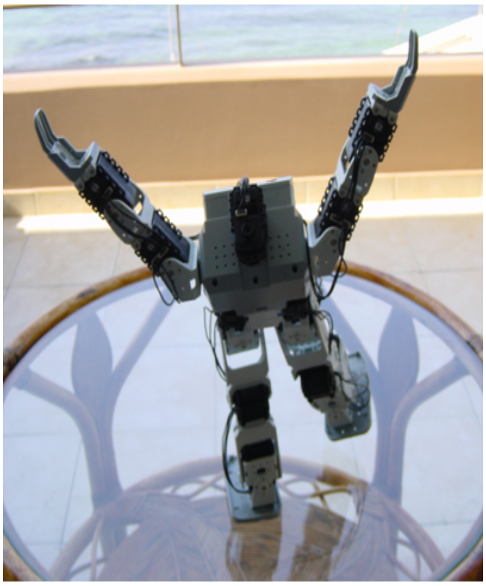
Position of total CoM according to system Origin		
X	Y	Z
-20,20116904	-9,928850823	196,2316922



Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$6,15°$
Joint 2	15	$28,72°$
Joint 3	13	$-40,15°$
Joint 4	11	$5,27°$
Joint 5	9	$-21,39°$
Joint 6	7	$2,34°$
Joint 7	1	$-42,20°$
Joint 8	3	$-89,97°$
Joint 9	5	$-22,86°$
Joint 10	2	$40,73°$
Joint 11	4	$74,14°$
Joint 12	6	$28,43°$
Joint 13	8	$4,40°$
Joint 14	10	$-5,27°$
Joint 15	12	$21,69°$
Joint 16	14	$52,75°$
Joint 17	16	$-7,91°$
Joint 18	18	$21,10°$

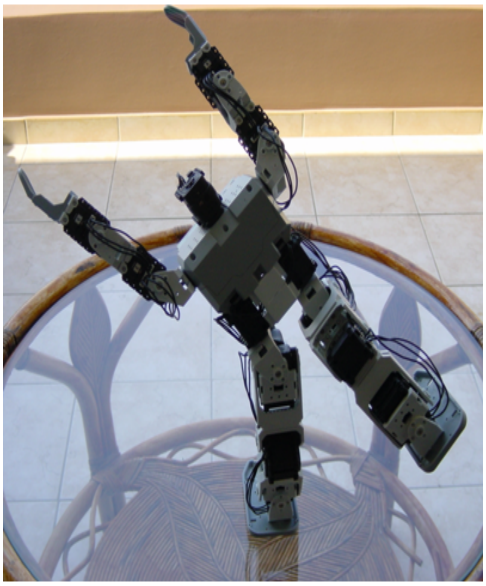
6.5 Hands Over The Head Behind the Torso

Position of total CoM according to system Origin		
X	Y	Z
-12,20203483	-8,300664892	208,4683706



Joint Number	Motor ID	Joint Agle
Joint 1	17	$8,21°$
Joint 2	15	$-9,38°$
Joint 3	13	$0,00°$
Joint 4	11	$-42,20°$
Joint 5	9	$-12,31°$
Joint 6	7	$-3,22°$
Joint 7	1	$128,06°$
Joint 8	3	$-53,92°$
Joint 9	5	$0,00°$
Joint 10	2	$-130,70°$
Joint 11	4	$58,02°$
Joint 12	6	$-0,29°$
Joint 13	8	$2,34°$
Joint 14	10	$17,88°$
Joint 15	12	$23,74°$
Joint 16	14	$25,20°$
Joint 17	16	$-18,17°$
Joint 18	18	$-0,88°$

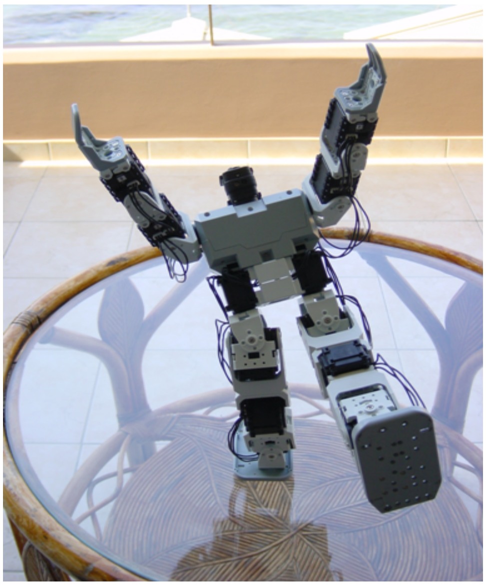
Position of total CoM according to system Origin		
X	Y	Z
-23,29943794	-8,240970463	226,0755479



Joint Number	Motor ID	Joint Agle
Joint 1	17	$-0,59°$
Joint 2	15	$25,50°$
Joint 3	13	$-33,41°$
Joint 4	11	$-3,52°$
Joint 5	9	$-34,87°$
Joint 6	7	$13,19°$
Joint 7	1	$109,89°$
Joint 8	3	$-76,78°$
Joint 9	5	$0,00°$
Joint 10	2	$-120,15°$
Joint 11	4	$87,33°$
Joint 12	6	$-0,29°$
Joint 13	8	$7,33°$
Joint 14	10	$13,48°$
Joint 15	12	$95,24°$
Joint 16	14	$71,50°$
Joint 17	16	$25,50°$
Joint 18	18	$-0,88°$

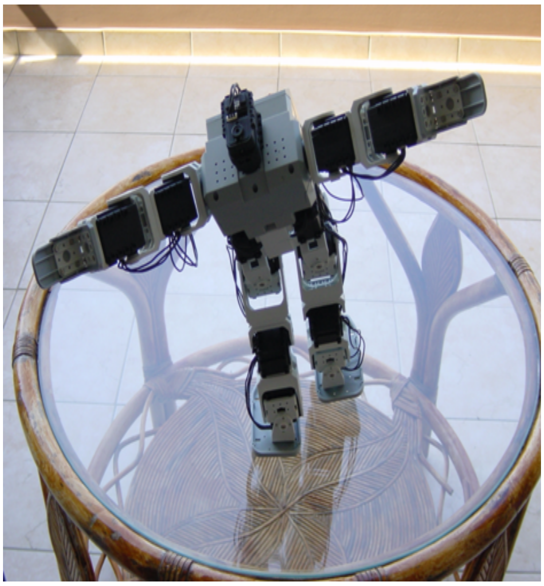
6.6 Hands Over Head In front of Torso

Position of total CoM according to system Origin		
X	Y	Z
-14,6323875	-1,981270173	215,4104715



Joint Number	Motor ID	Joint Agle
Joint 1	17	$7,62°$
Joint 2	15	$22,86°$
Joint 3	13	$-29,31°$
Joint 4	11	$23,44°$
Joint 5	9	$-17,29°$
Joint 6	7	$4,69°$
Joint 7	1	$39,56°$
Joint 8	3	$-75,90°$
Joint 9	5	$-0,29°$
Joint 10	2	$-31,94°$
Joint 11	4	$58,32°$
Joint 12	6	$0,00°$
Joint 13	8	$0,88°$
Joint 14	10	$-0,29°$
Joint 15	12	$58,90°$
Joint 16	14	$0,88°$
Joint 17	16	$-11,72°$
Joint 18	18	$-0,59°$

POSITION OF TOTAL COM ACCORDING TO SYSTEM ORIGIN		
X	Y	Z
-5,903161963	-0,730894118	209,4001087



Joint Number	Motor ID	Joint Agle
Joint 1	17	$8,21°$
Joint 2	15	$2,34°$
Joint 3	13	$-22,56°$
Joint 4	11	$-38,39°$
Joint 5	9	$-14,36°$
Joint 6	7	$4,40°$
Joint 7	1	$51,28°$
Joint 8	3	$-54,51°$
Joint 9	5	$2,34°$
Joint 10	2	$-54,51°$
Joint 11	4	$33,70°$
Joint 12	6	$-0,59°$
Joint 13	8	$1,17°$
Joint 14	10	$-9,08°$
Joint 15	12	$23,44°$
Joint 16	14	$22,27°$
Joint 17	16	$-12,02°$
Joint 18	18	$-0,88°$

Position of total CoM according to system Origin		
X	Y	Z
-10,03247325	-0,413294637	220,1327228

Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$7,03°$
Joint 2	15	$8,21°$
Joint 3	13	$-24,32°$
Joint 4	11	$-11,14°$
Joint 5	9	$-14,36°$
Joint 6	7	$-13,48°$
Joint 7	1	$59,49°$
Joint 8	3	$-47,77°$
Joint 9	5	$-0,29°$
Joint 10	2	$-62,13°$
Joint 11	4	$37,22°$
Joint 12	6	$0,00°$
Joint 13	8	$-4,98°$
Joint 14	10	$6,74°$
Joint 15	12	$84,98°$
Joint 16	14	$115,76°$
Joint 17	16	$-11,43°$
Joint 18	18	$0,00°$

6.7 Figures

Position of total CoM according to system Origin		
X	Y	Z
-14,6323875	-1,981270173	215,4104715

Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$2,34°$
Joint 2	15	$8,50°$
Joint 3	13	$-17,88°$
Joint 4	11	$-33,11°$
Joint 5	9	$-38,39°$
Joint 6	7	$13,19°$
Joint 7	1	$-90,26°$
Joint 8	3	$47,77°$
Joint 9	5	$-86,45°$
Joint 10	2	$87,33°$
Joint 11	4	$-47,18°$
Joint 12	6	$67,69°$
Joint 13	8	$-78,24°$
Joint 14	10	$9,23°$
Joint 15	12	$6,15°$
Joint 16	14	$26,96°$
Joint 17	16	$-14,07°$
Joint 18	18	$-0,59°$

Position of total CoM according to system Origin		
X	Y	Z
-5,464573893	-0,901236267	200,5883206

Joint Number	Motor ID	Joint Agle
Joint 1	17	$11,43°$
Joint 2	15	$-3,22°$
Joint 3	13	$-13,77°$
Joint 4	11	$-47,77°$
Joint 5	9	$-4,10°$
Joint 6	7	$-20,51°$
Joint 7	1	$-50,70°$
Joint 8	3	$-66,82°$
Joint 9	5	$45,42°$
Joint 10	2	$14,20°$
Joint 11	4	$24,91°$
Joint 12	6	$-39,85°$
Joint 13	8	$-23,74°$
Joint 14	10	$17,29°$
Joint 15	12	$-19,05°$
Joint 16	14	$67,99°$
Joint 17	16	$-21,69°$
Joint 18	18	$-0,88°$

Position of total CoM according to system Origin		
X	Y	Z
-6,266011261	-2,879114792	229,649973

Joint Number	Motor ID	Joint Agle
Joint 1	17	$8,21°$
Joint 2	15	$3,81°$
Joint 3	13	$-7,03°$
Joint 4	11	$-8,79°$
Joint 5	9	$-31,36°$
Joint 6	7	$11,14°$
Joint 7	1	$-84,98°$
Joint 8	3	$64,47°$
Joint 9	5	$-60,66°$
Joint 10	2	$97,00°$
Joint 11	4	$29,31°$
Joint 12	6	$-24,62°$
Joint 13	8	$7,47°$
Joint 14	10	$52,75°$
Joint 15	12	$38,10°$
Joint 16	14	$-2,93°$
Joint 17	16	$-55,68°$
Joint 18	18	$-0,59°$

Position of total CoM according to system Origin		
X	Y	Z
-0,508840825	-1,059151286	219,4620788

Joint Number	Motor ID	Joint Agle
Joint 1	17	$3,81°$
Joint 2	15	$-0,59°$
Joint 3	13	$-9,08°$
Joint 4	11	$-28,72°$
Joint 5	9	$-39,85°$
Joint 6	7	$2,93°$
Joint 7	1	$-148,28°$
Joint 8	3	$-31,36°$
Joint 9	5	$69,75°$
Joint 10	2	$38,68°$
Joint 11	4	$-13,48°$
Joint 12	6	$100,52°$
Joint 13	8	$-2,64°$
Joint 14	10	$13,48°$
Joint 15	12	$-1,17°$
Joint 16	14	$73,26°$
Joint 17	16	$-11,72°$
Joint 18	18	$-0,59°$

Position of total CoM according to system Origin		
X	Y	Z
-9,195606288	9,17886391	173,6751607

Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$1,47°$
Joint 2	15	$1,17°$
Joint 3	13	$-17,88°$
Joint 4	11	$-32,53°$
Joint 5	9	$-32,82°$
Joint 6	7	$-29,89°$
Joint 7	1	$-138,91°$
Joint 8	3	$-91,73°$
Joint 9	5	$-17,88°$
Joint 10	2	$48,06°$
Joint 11	4	$102,57°$
Joint 12	6	$48,06°$
Joint 13	8	$-58,61°$
Joint 14	10	$-0,29°$
Joint 15	12	$-7,33°$
Joint 16	14	$80,30°$
Joint 17	16	$-59,49°$
Joint 18	18	$-0,88°$

Position of total CoM according to system Origin		
X	Y	Z
-14,6323875	-1,981270173	215,4104715

Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$22,27°$
Joint 2	15	$24,32°$
Joint 3	13	$-99,64°$
Joint 4	11	$-76,19°$
Joint 5	9	$9,38°$
Joint 6	7	$-3,22°$
Joint 7	1	$-19,05°$
Joint 8	3	$-80,30°$
Joint 9	5	$-56,56°$
Joint 10	2	$19,34°$
Joint 11	4	$93,19°$
Joint 12	6	$51,58°$
Joint 13	8	$-4,10°$
Joint 14	10	$-8,50°$
Joint 15	12	$83,81°$
Joint 16	14	$-3,22°$
Joint 17	16	$-19,93°$
Joint 18	18	$12,02°$

Position of total CoM according to system Origin		
X	Y	Z
-14,18711542	-7,121221624	237,6236615

Joint Number	**Motor ID**	**Joint Agle**
Joint 1	17	$0,88°$
Joint 2	15	$10,26°$
Joint 3	13	$-22,27°$
Joint 4	11	$-16,70°$
Joint 5	9	$-47,47°$
Joint 6	7	$-7,62°$
Joint 7	1	$-96,41°$
Joint 8	3	$0,00°$
Joint 9	5	$-1,76°$
Joint 10	2	$89,38°$
Joint 11	4	$-6,45°$
Joint 12	6	$0,00°$
Joint 13	8	$-10,55°$
Joint 14	10	$84,40°$
Joint 15	12	$13,77°$
Joint 16	14	$12,89°$
Joint 17	16	$-12,89°$
Joint 18	18	$-0,88°$

6.8 Demo of Statically Balanced Poses

- "'No strings"' attached

  For the realization of the experiments besides the use of the motion editor we created a behaviour control program to manage the poses saved in the Humanoid's CM -5, so as to leave the humanoid perform the poses of static equilibrium without the extra weight of the cable connecting the serial port of the humanoid with the USB port of the controlling computer that would potentially affect the position of the overall CoM since it would be affected by external forces (gravitational forces acting on the cable that would eventually pull the system toward a direction). That is the reason we created a program for the demonstration of the Humanoid's poses.

- Interaction with the User

  We categorized the 20 different poses into groups of fives. For the triggering of each group the user must press one of the four buttons of the CM-5 (u, d, l, r). Once the choice of category has been made , then the robot will perform the number of pose that the user will indicate by giving sound signals(claps).For example U and 2 claps means that the Humanoid will perform the second pose of the first category of poses. Once the humanoid has reached the target pose it remains in static equilibrium for 2 seconds and then returns to the default upright position.

6.8.1 The Behaviour Control Program

The program was created with the use of the Bioloid's software "'Behaviour Control Programmer"'. Since there is no source code created and the whole programming is done in the graphical user interface of the software, the only way to show the the created program is with a serial presentation of screenshots taken from the GUI.

- The Main Program

Figure 6.2: The Main Program 1

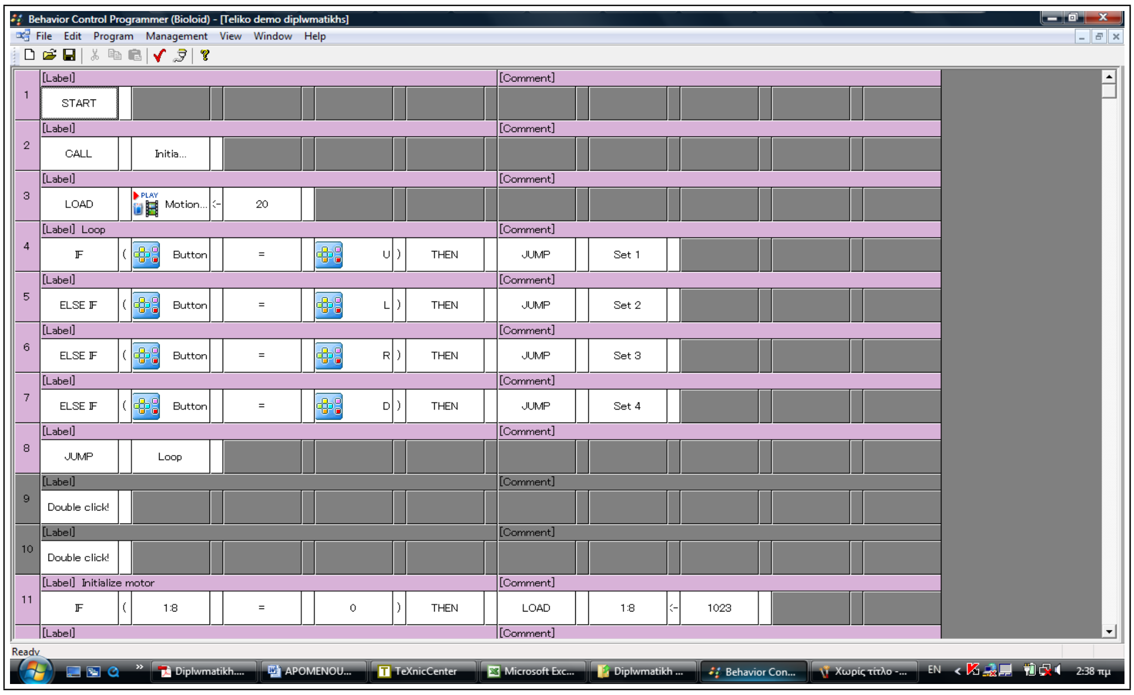


Figure 6.3: The Main Program 2

- Category 1

  In this routine there is the initialization of sound detector of the Ax S1. Then should a clap be detected , a counter is being used to note the number of claps the user gives. Finally the motion page that has the number of claps given by the user is executed. To ensure the execution of the pose see the complete motion execution routine.

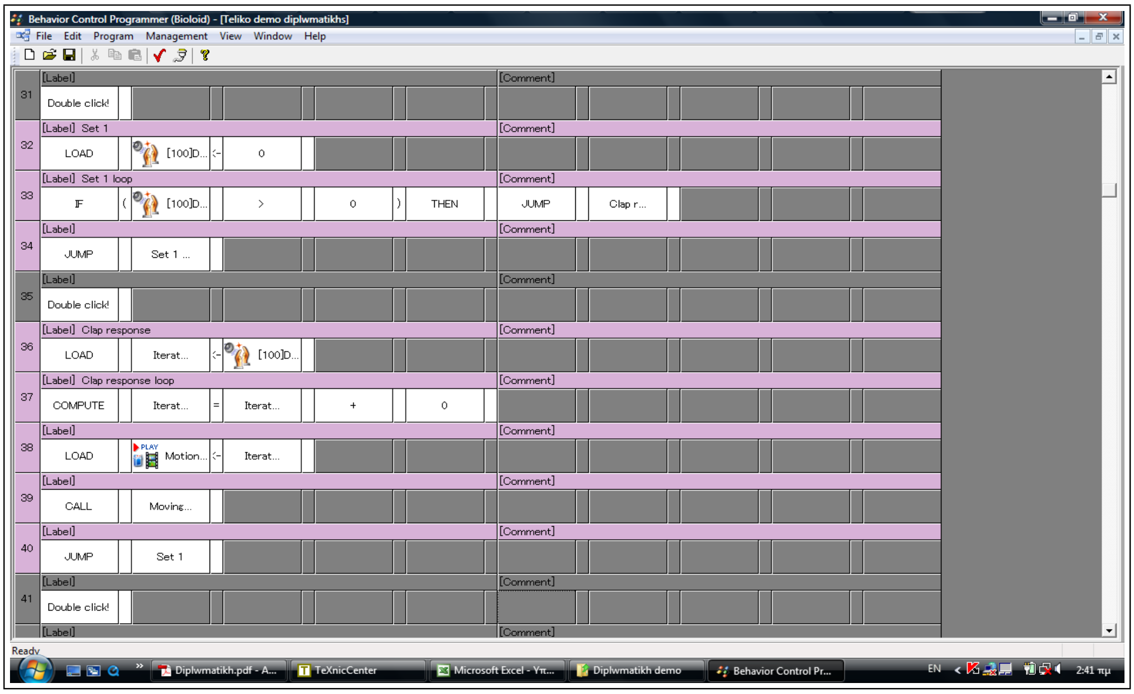


Figure 6.4: The Routine Set 1

- Category 2

  In this routine there is the initialization of sound detector of the Ax S1. For this category the number of claps is the one given by the user plus 5. Finally the motion page that has the number of claps given by the user for this category is executed. To ensure the execution of the pose see the complete motion execution routine.

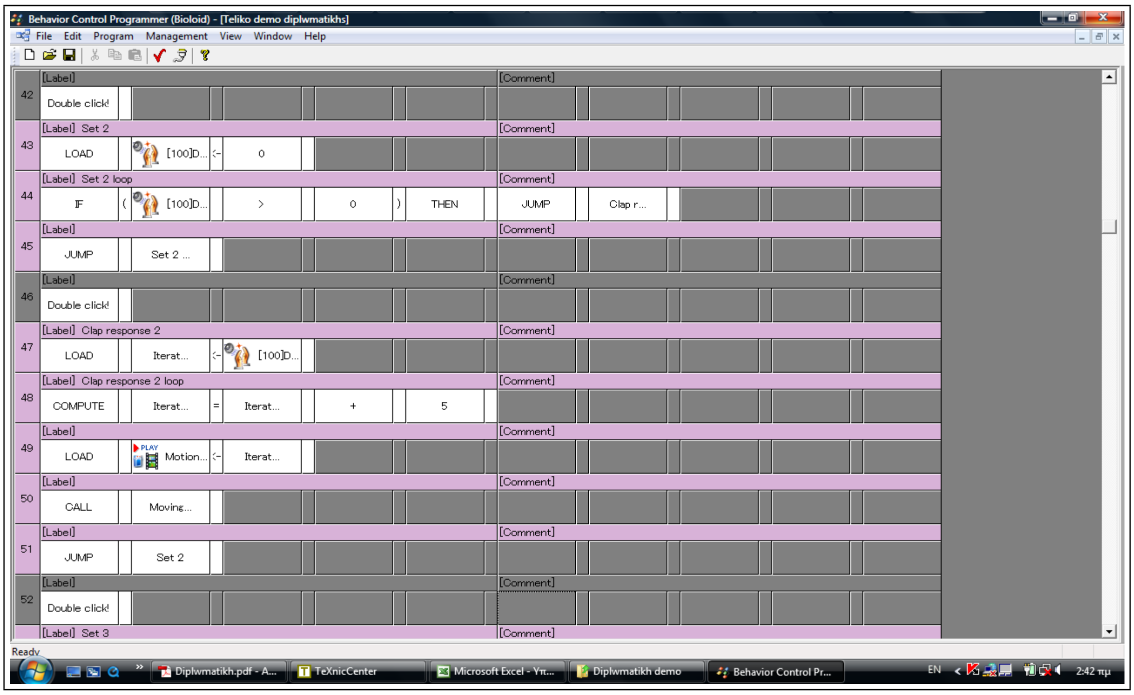


Figure 6.5: The Routine Set 2

- Category 3 In this routine there is the initialization of sound detector of the Ax S1. Then should a clap be detected , a counter is being used to note the number of claps the user gives. For this category the number of claps is the one given by the user plus 10. Finally the motion page that has the number of claps given by the user for this category is executed.

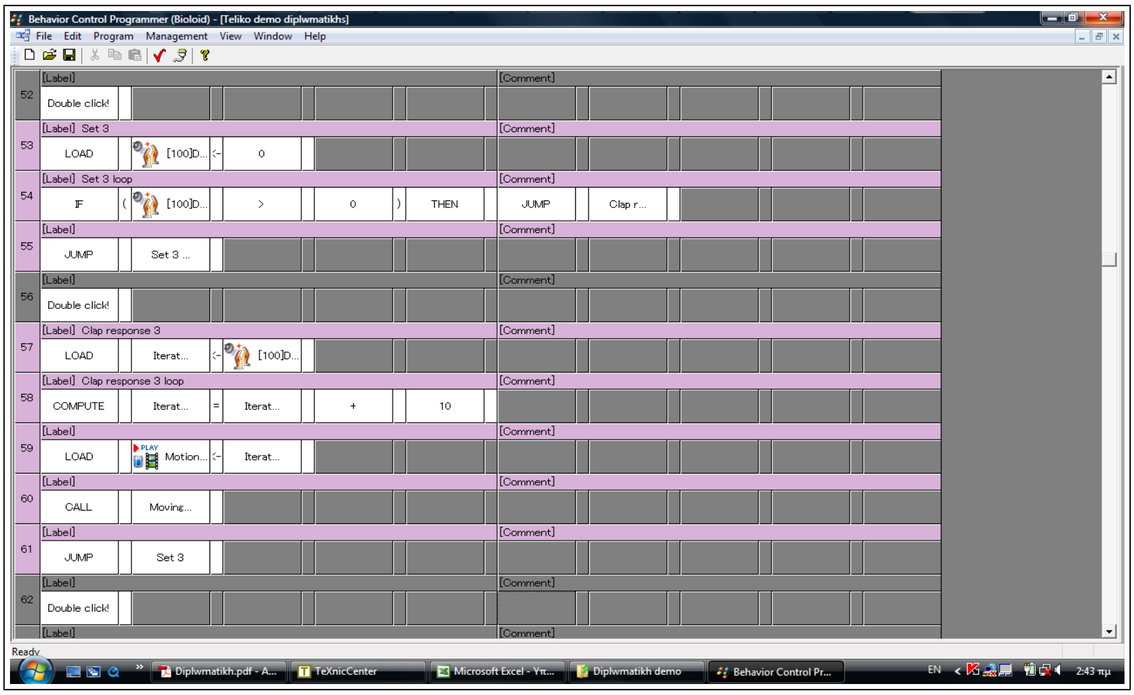


Figure 6.6: The Routine Set 3

- Category 4

  In the same motif in this routine there is the initialization of sound detector of the Ax S1. Then should a clap be detected , then a counter is being used to note the number of claps the user gives. For this category the number of claps is the one given by the user plus 15.

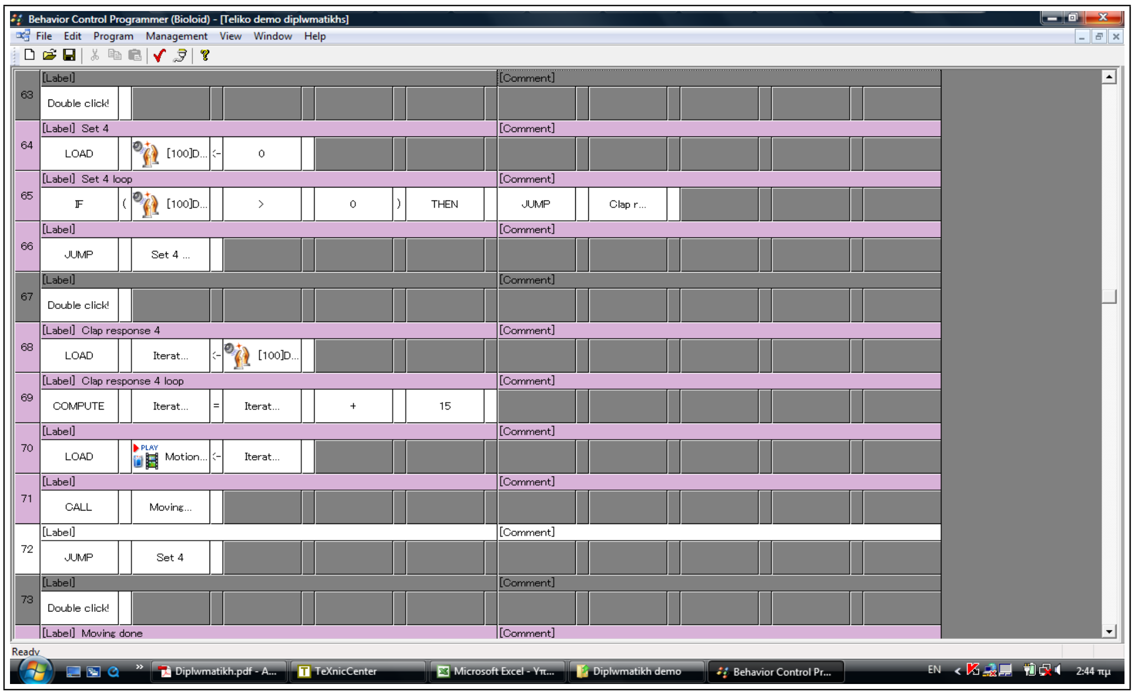


Figure 6.7: The Routine Set 4

- Initialization Of Motors

  Before the beginning of a program we initialize the values of the motors.

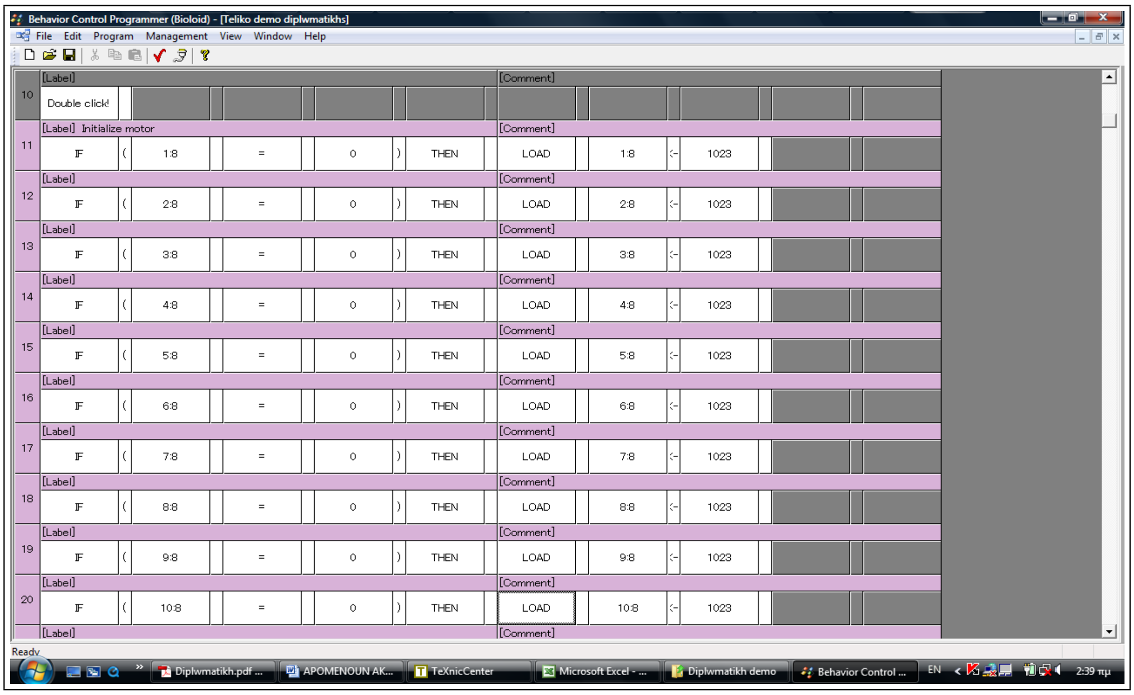


Figure 6.8: The Initialization of the actuators 1

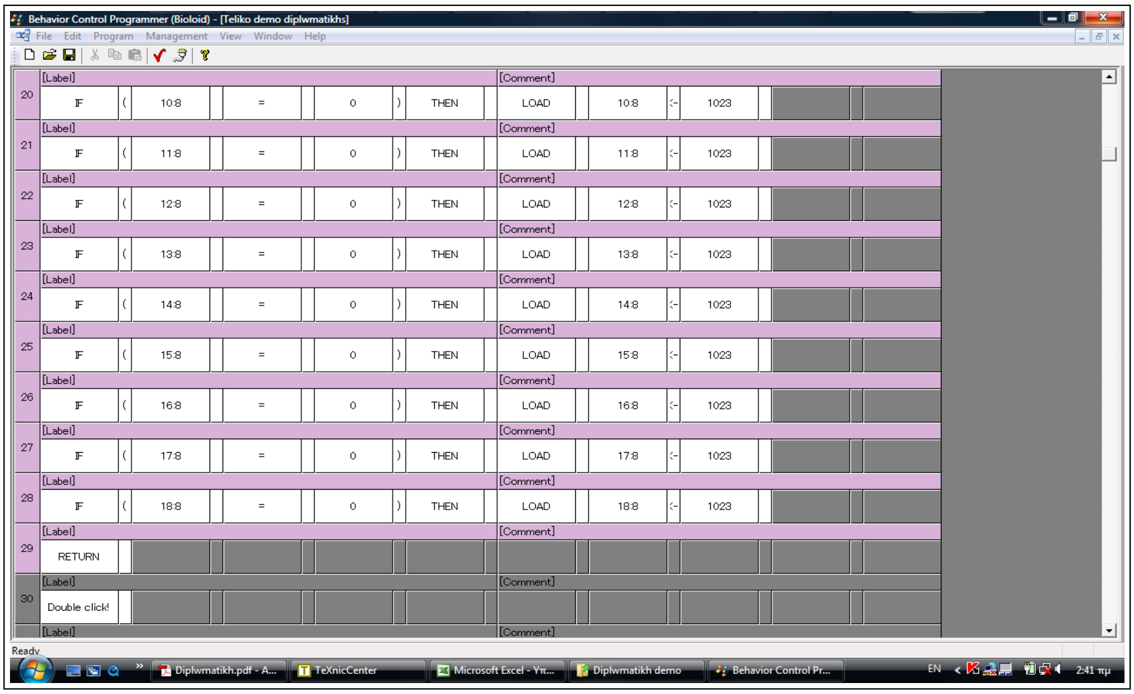


Figure 6.9: The Initialization of the actuators 2

- **Complete Motion Execution**

  For the completion of a motion called from the poses saved in the Humanoid's CM-5 with the use of the motion editor, we created a subroutine that checks whether any of the 18 actuators is moving or not. If motion is detected with the coders(the value of the actuator's angle is changing) then the routine calls its self, buying time so that the motion page can be executed not just "'passed"' while the program is being executed.

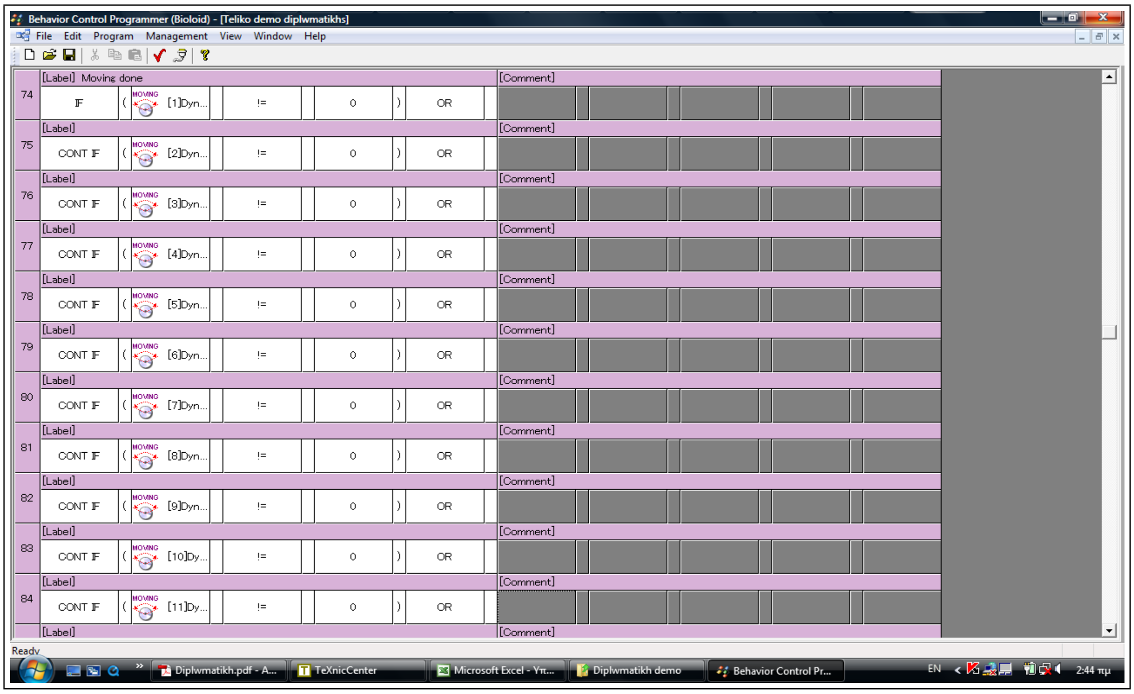


Figure 6.10: Sub-Routine 1

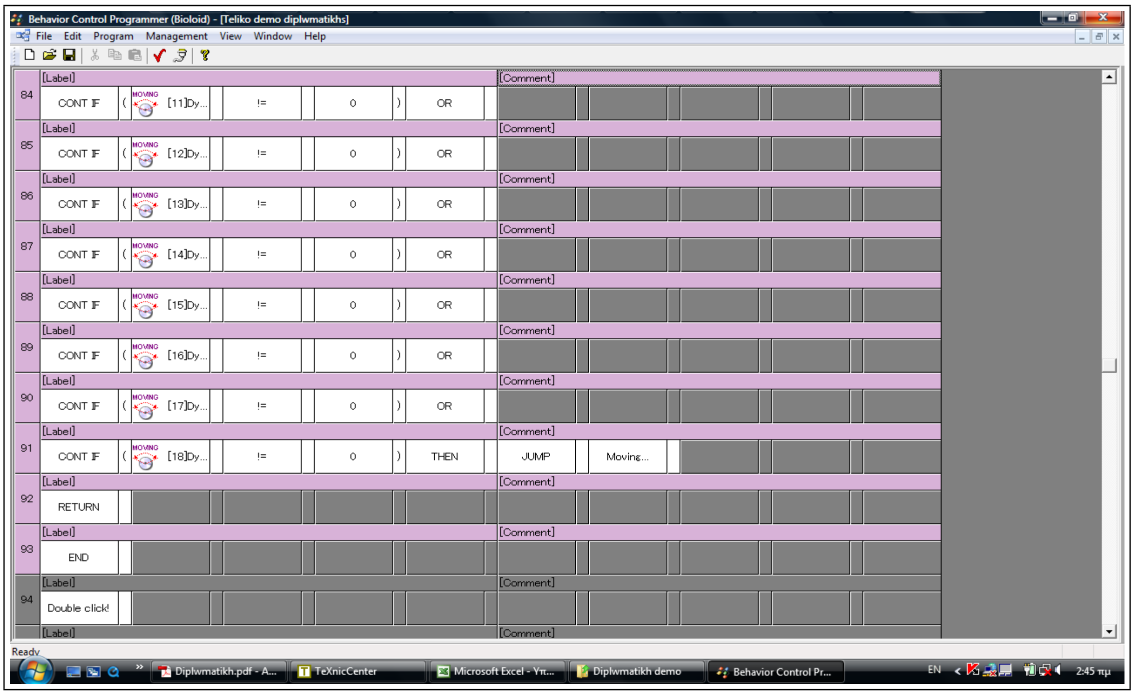


Figure 6.11: Sub-Routine 2

Chapter 7

SUMMARY AND FUTURE WORK

Having modeled the Humanoid Bioloid with the particular level of accuracy we have created the necessary foundations so that the programming part can be performed. With the use of the equations of the forward kinematics a programmer can create algorithmic solutions to almost any given problem that concerns static balance. Below we provide some pointers to future work to be done.

Taking Advantage of the Legs equations

Also the programming doesn't have to been constraint by the modeling of the Humanoid being supported by the right foot only. A method of generating random statically-stable postures (i.e. random point samples of C_{stable}) can be used . Although it is trivial to generate random configurations in C, it is not so easy to generate them in C_{stable}, since it encompasses a much smaller subset of the configuration space.

For configurations that involve balancing on only one leg, the set Q_{stable} can be populated for the left foot also, since most humanoid robots have left-right symmetry, if $Q_{rand} \epsilon C_{valid}$ in either or both cases, we can "mirror" Q_{rand} to generate stable postures for the opposite, left foot.[11]

Taken for granted that we have achieved the "mirrirong" of the equations that describe the centers of mass according to the left left foot, the Humanoid can be modeled in a way so that by incorporating the left foot, we would have the necessary means to create a statically stable walking algorithm .

Taking Advantage of the Arms & Legs equations

The next step would be the inverse kinematics solution. Since the inverse kinematics problem has been already been solved for 6 degrees of freedom (the legs have larger number of DOF than the arms and equal to six) the Humanoid can be modeled in a slightly different way so that we can incorporate the solution. Taking for granted that the inverse kinematics problem has been solved, as you understand a whole new plateau of applications is ready to be explored as the whole approach can reach the next level, from statically stable to dynamically stable. Incorporating ZMP algorithms motion planning e.t.c

Machine Learning algorithms can be implemented using the Humanoid. A procedure that does not have the prerequisite of dynamic approaches. Also statistical approaches can be used for the creation of randomly produced statically stable poses.

Lack of Motion Sensors & Solutions

The big drawback found in the lack of any kind of accelerometer, inclinometer, gyroscope or similar devices that can be of used for the achievement of balance, static or dynamic can be "compensated" if we could say, in a very primitive way since the Humanoid's actuators provide torque for motion that is proportional to the current given to them. However to remain in a pose, the actuators also have to provide torque. So while remaining in an equilibrium state the current provided to the actuators is stable. Should external forces be applied to the Humanoid in order to maintain its position the current supply will be altered so that different torque will be outputted by the actuators to compensate for the external forces.In very rough terms "current means torque". Even if this perspective does not seem too intriguing if we see this prospect from the exactly opposite point of view - that is "torque means current" -we will understand that by using a set of joints as "sensors" for positioning of the robot (for example the ankle joints) the Humanoid should we slowly change for example the inclination of its support polygon, will be able to use this pair of joints as an inclinometer and with the appropriate algorithm react to the change of position.

That way with a low budget humanoid we indeed have a wide range of options as far as programming is concerned.

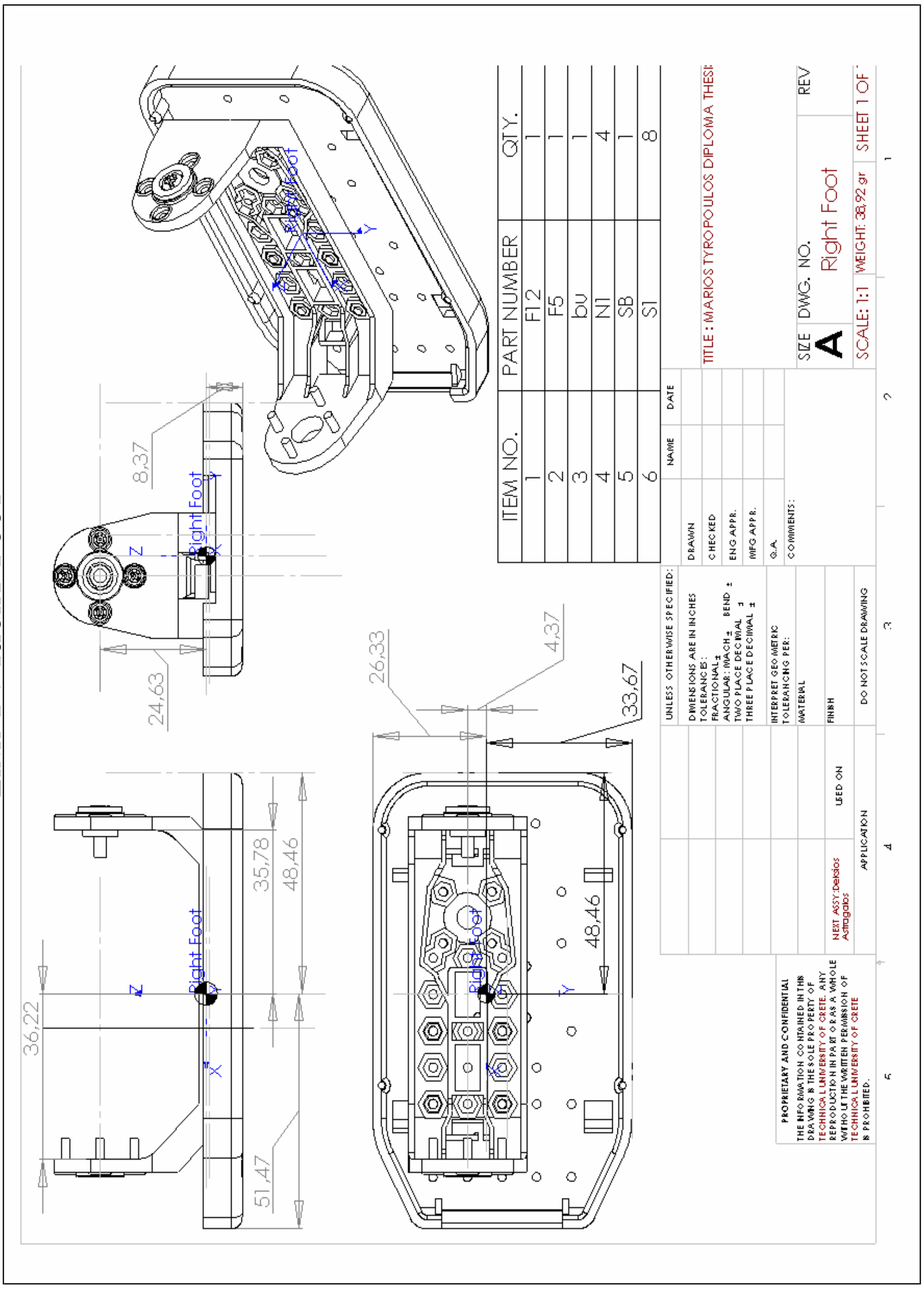
Part I

APPENDIX AND BIBLIOGRAPHY

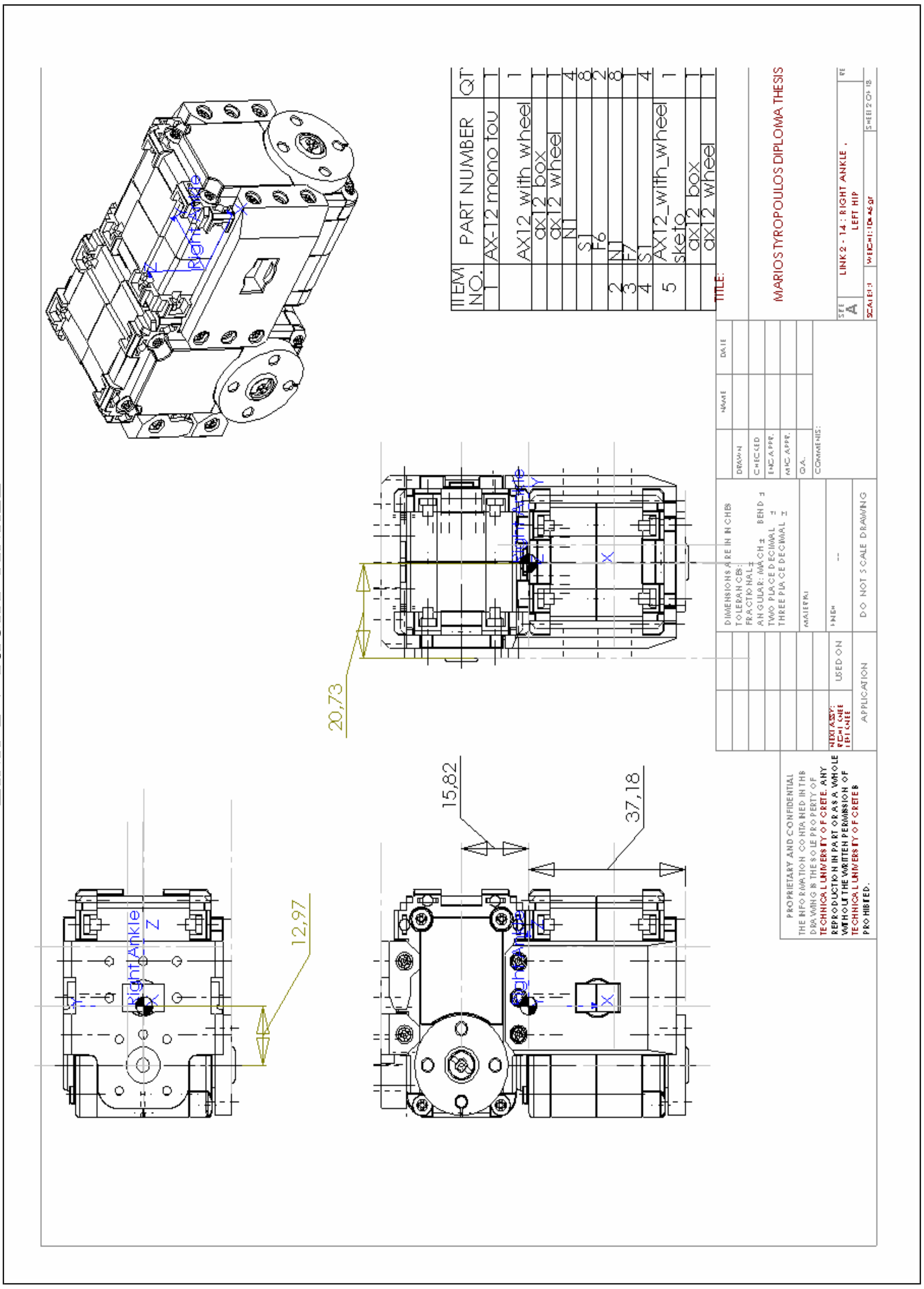
Chapter 8

LINKS

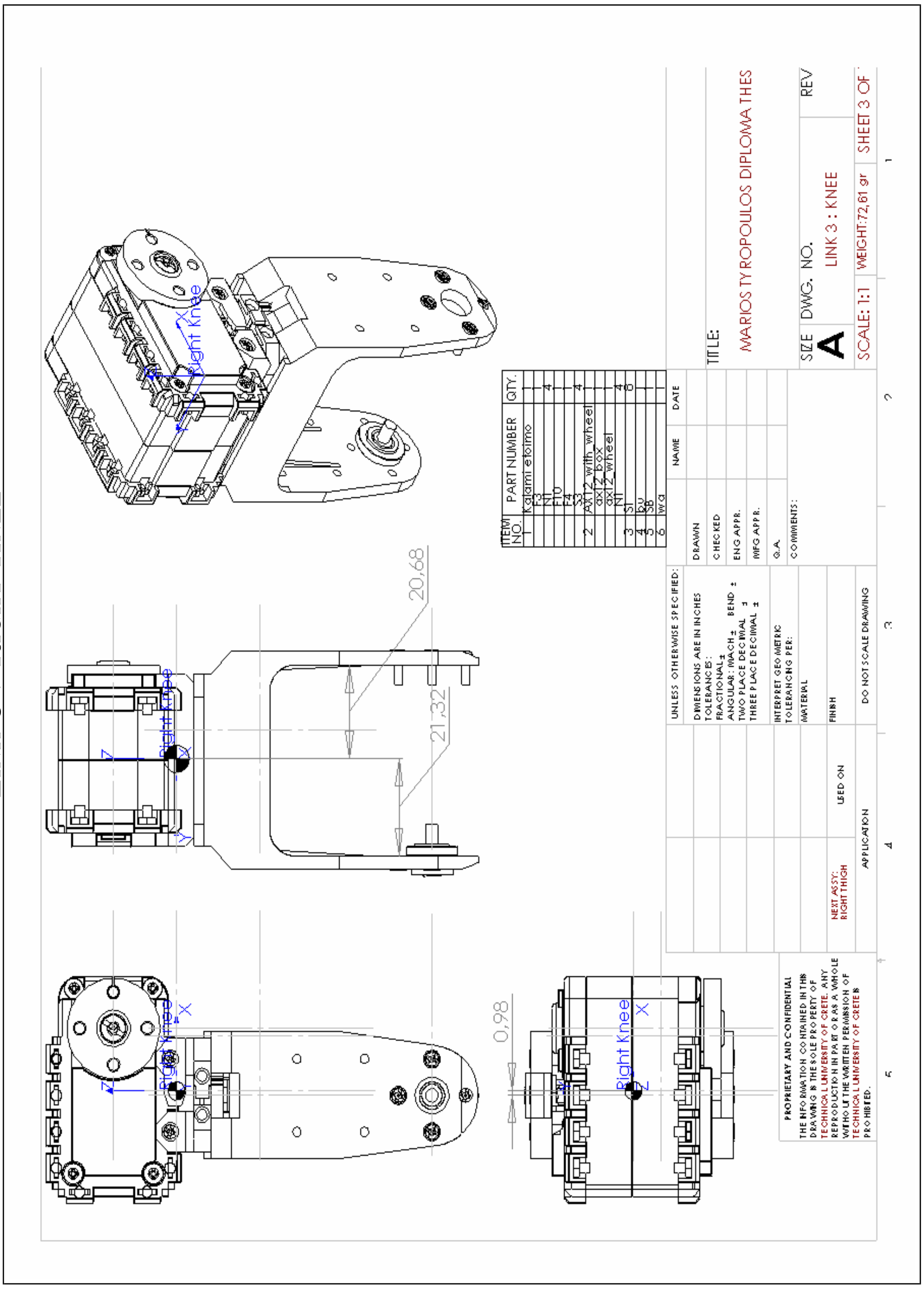
LINK 1 : RIGHT FOOT



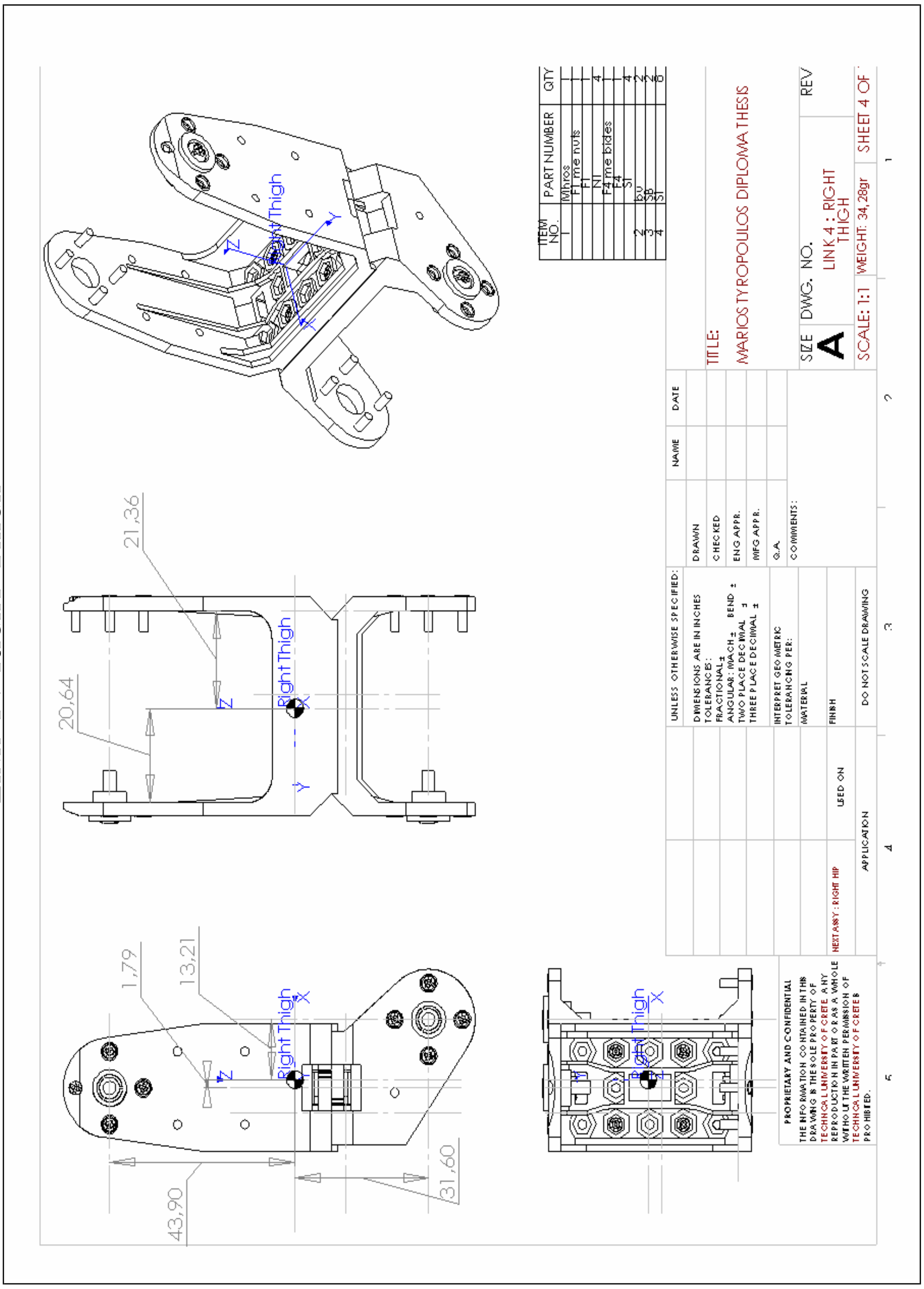
Link 2 : Right Ankle



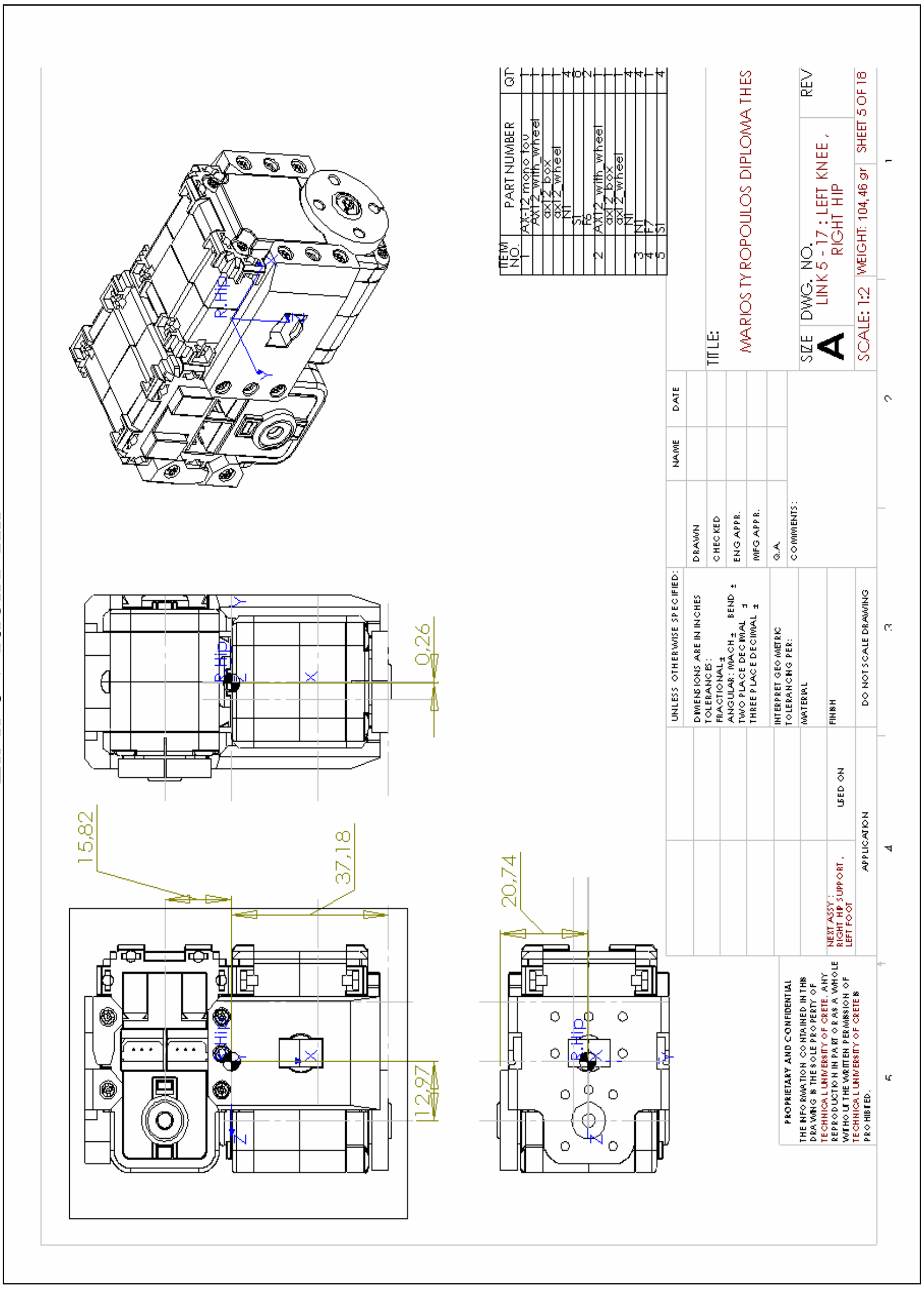
LINK 3 : RIGHT KNEE



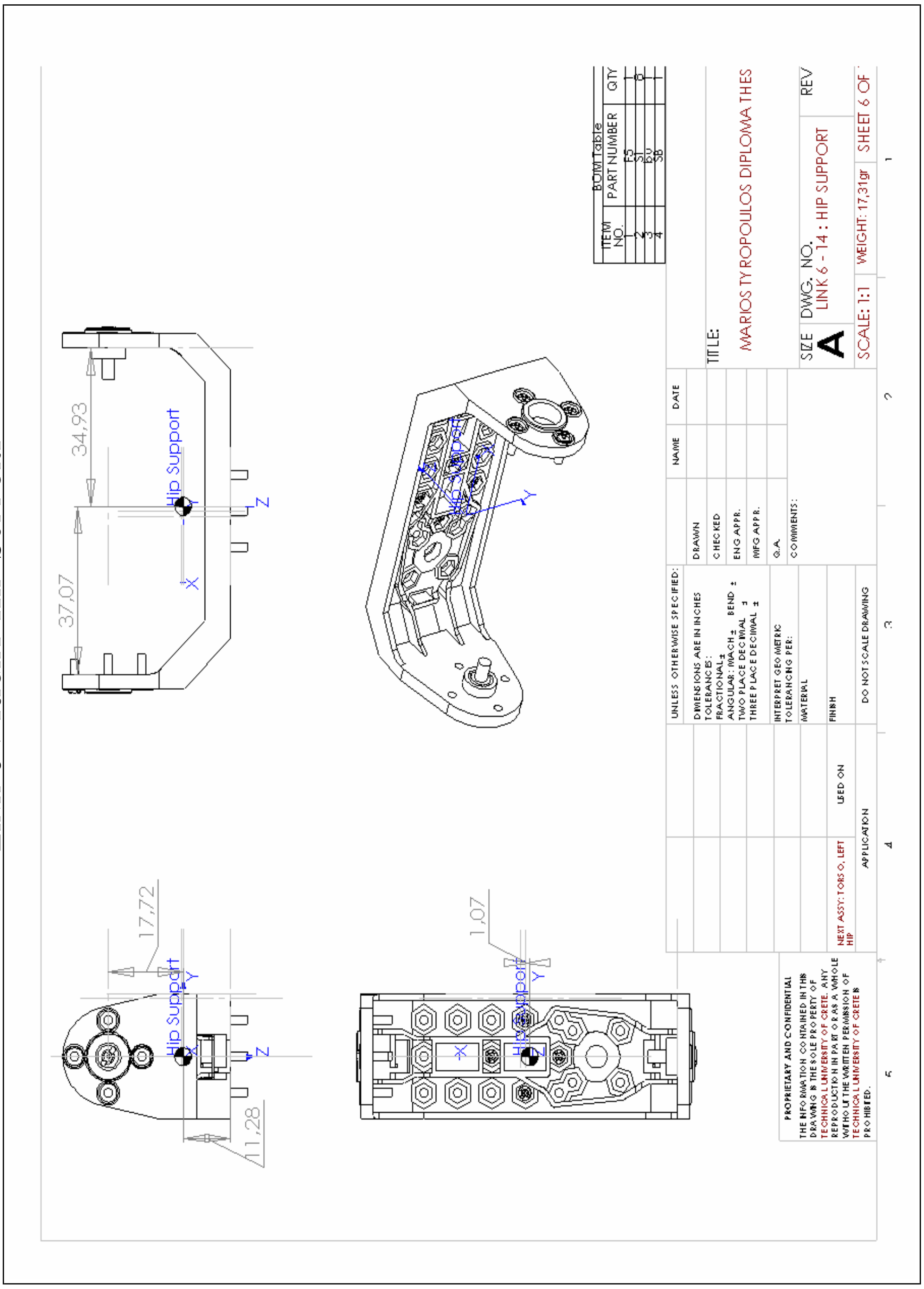
Link 4 : Right Thigh

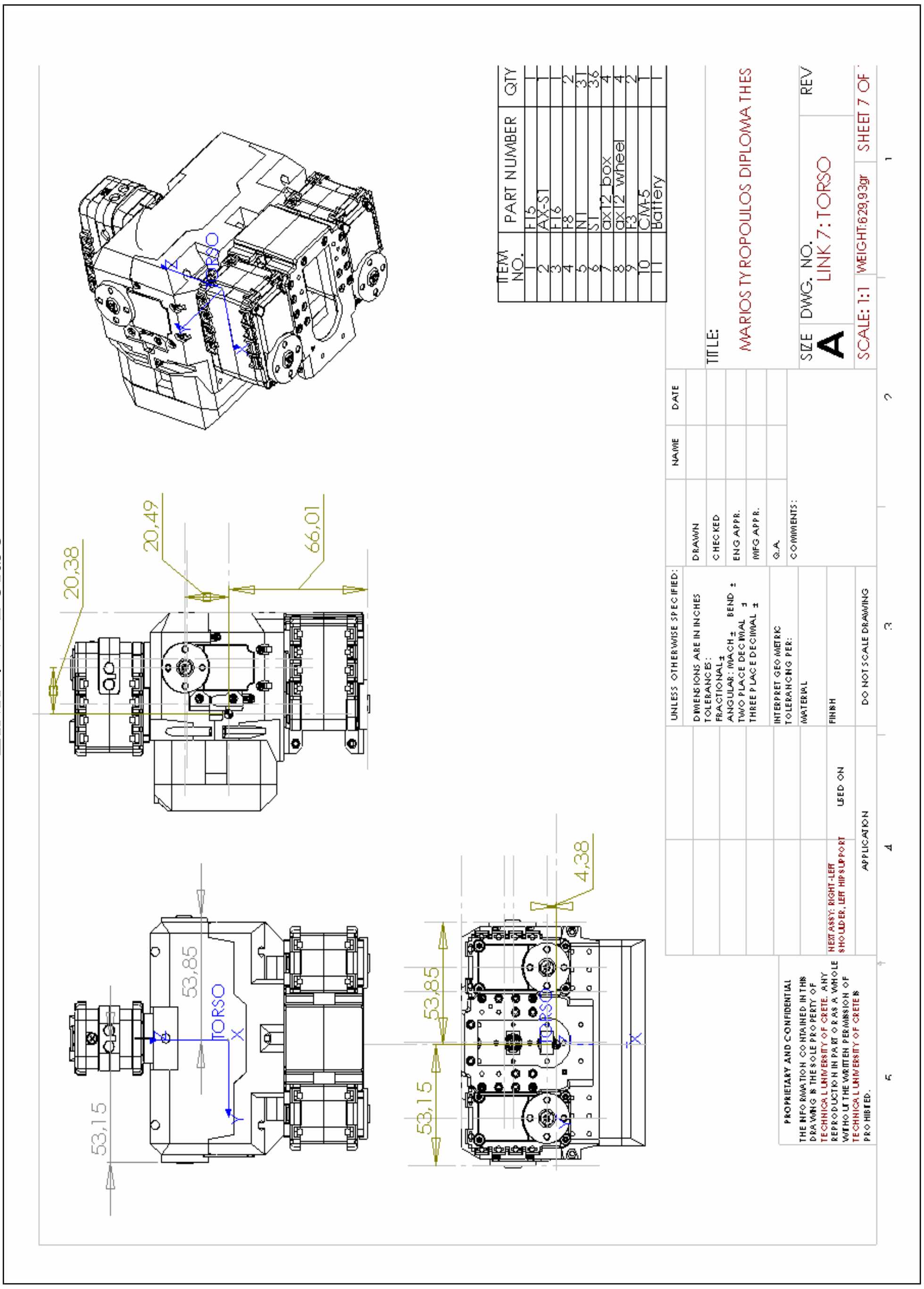


LINK 5 : RIGHT HIP



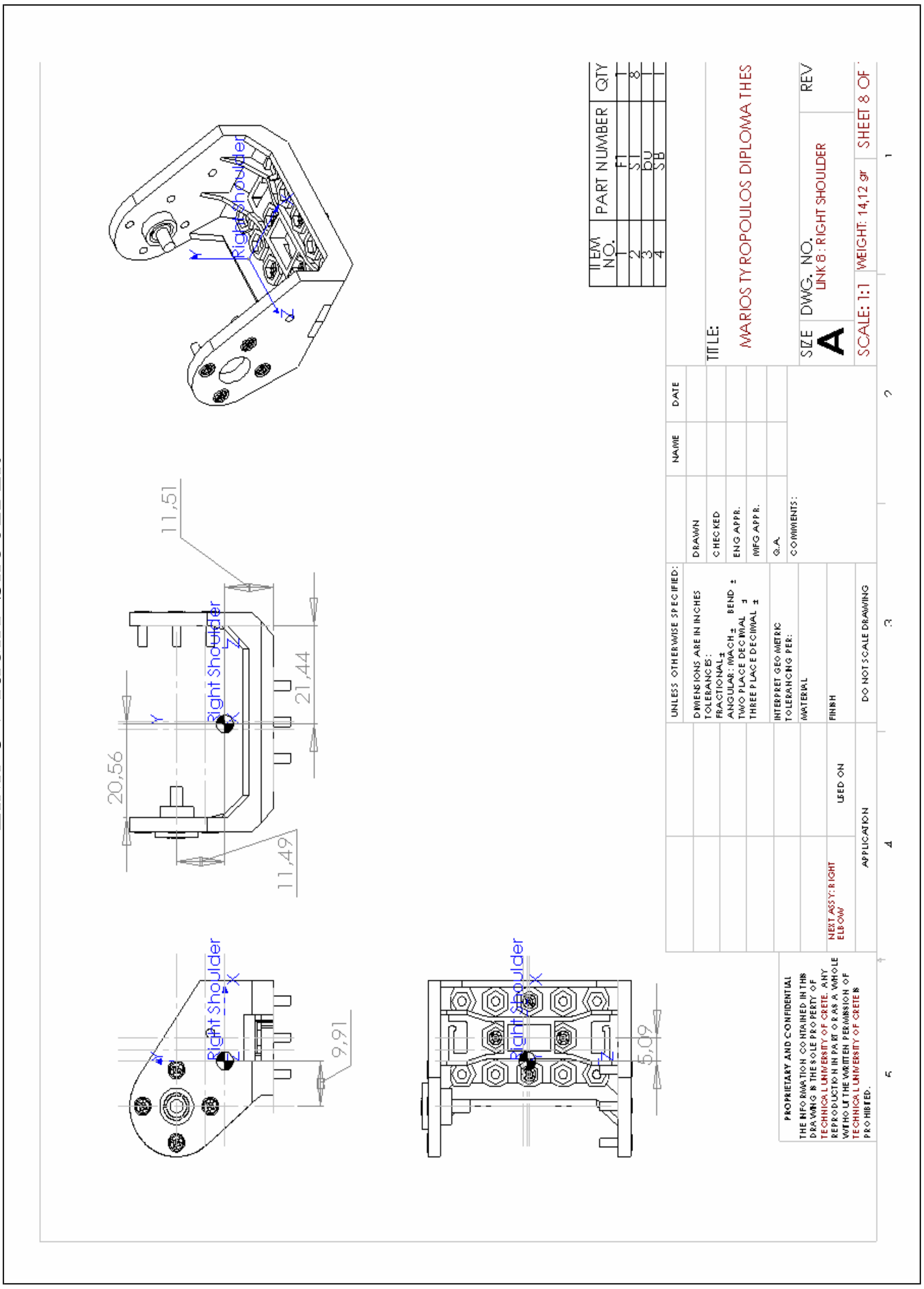
Link 6 : Right Hip Support



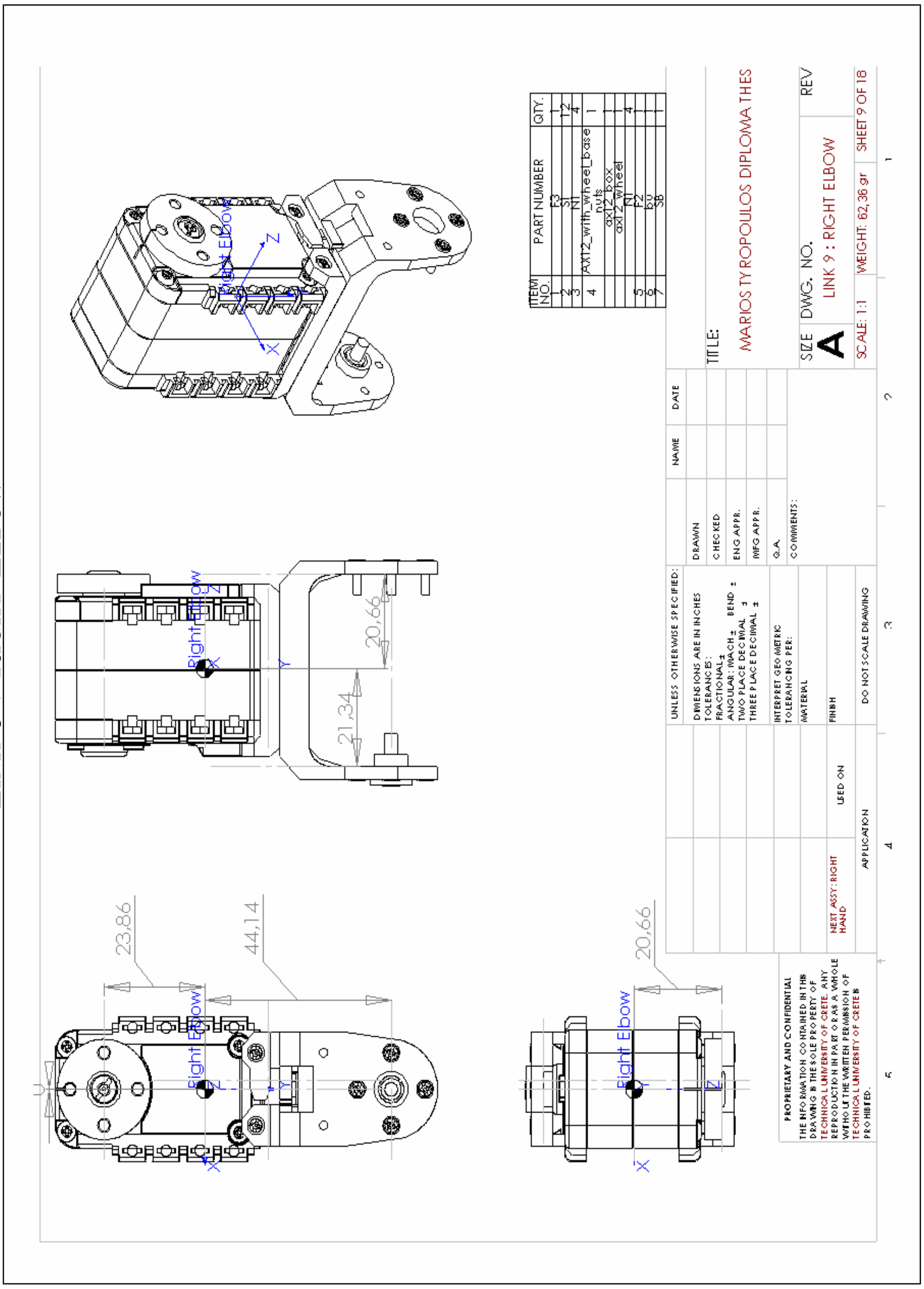

ITEM NO. | PART NUMBER | QTY
1 | F15 | 1
2 | AX-S1 | 1
3 | F16 | 1
4 | F8 | 1
5 | N1 | 2
6 | S1 | 31
7 | ax12_box | 36
8 | ax12_wheel | 4
9 | F3 | 4
10 | CM-5 | 2
11 | Battery | 1
20,38
20,49
66,01
53,85
53,15
4,38
TORSO
MARIOS TYROPOULOS DIPLOMA THES
LINK 7: TORSO
SCALE: 1:1
WEIGHT:629,93gr
SHEET 7 OF
DO NOT SCALE DRAWING
PROPRIETARY AND CONFIDENTIAL

Link 7 : Torso

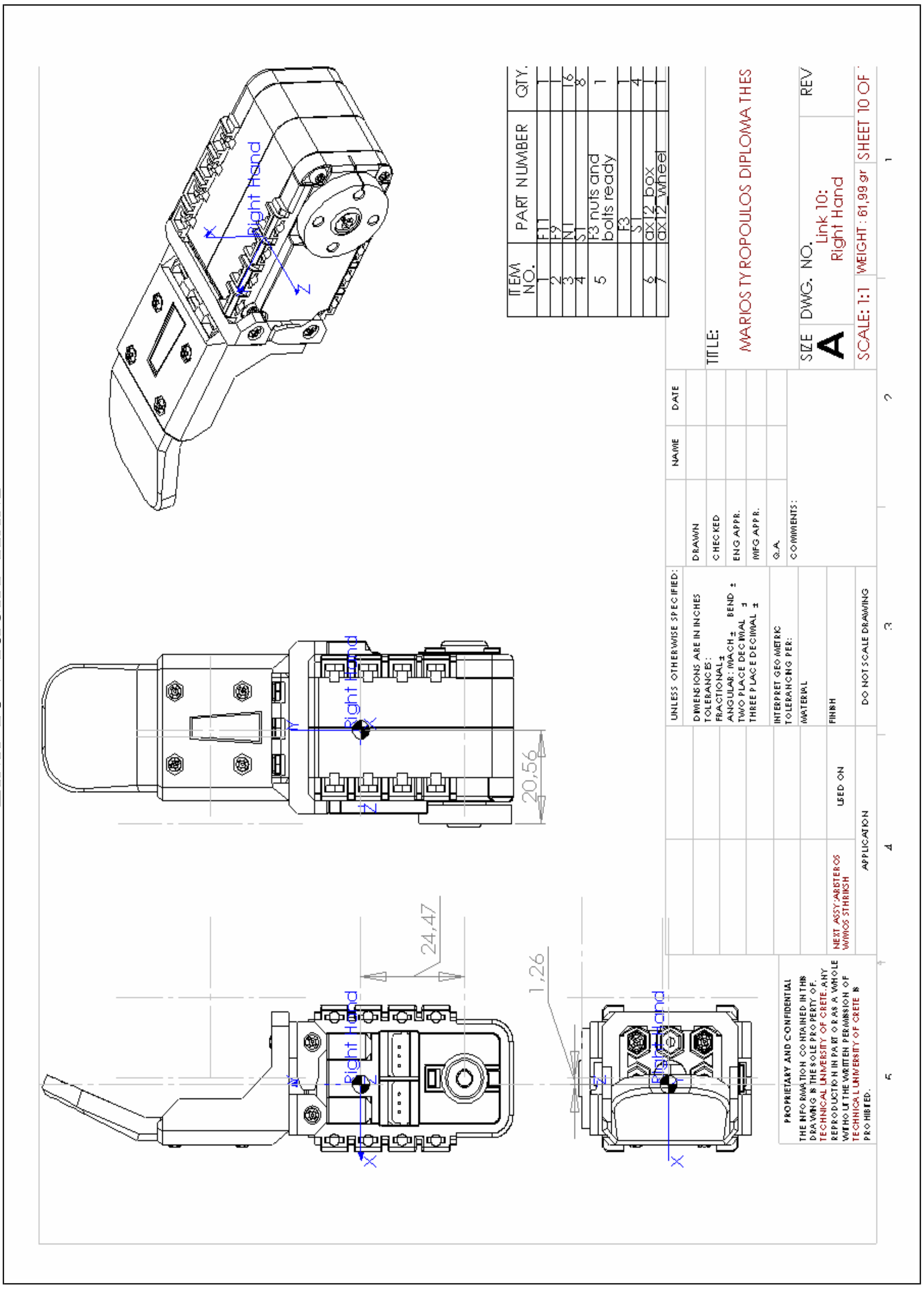
LINK 8 : RIGHT SHOULDER



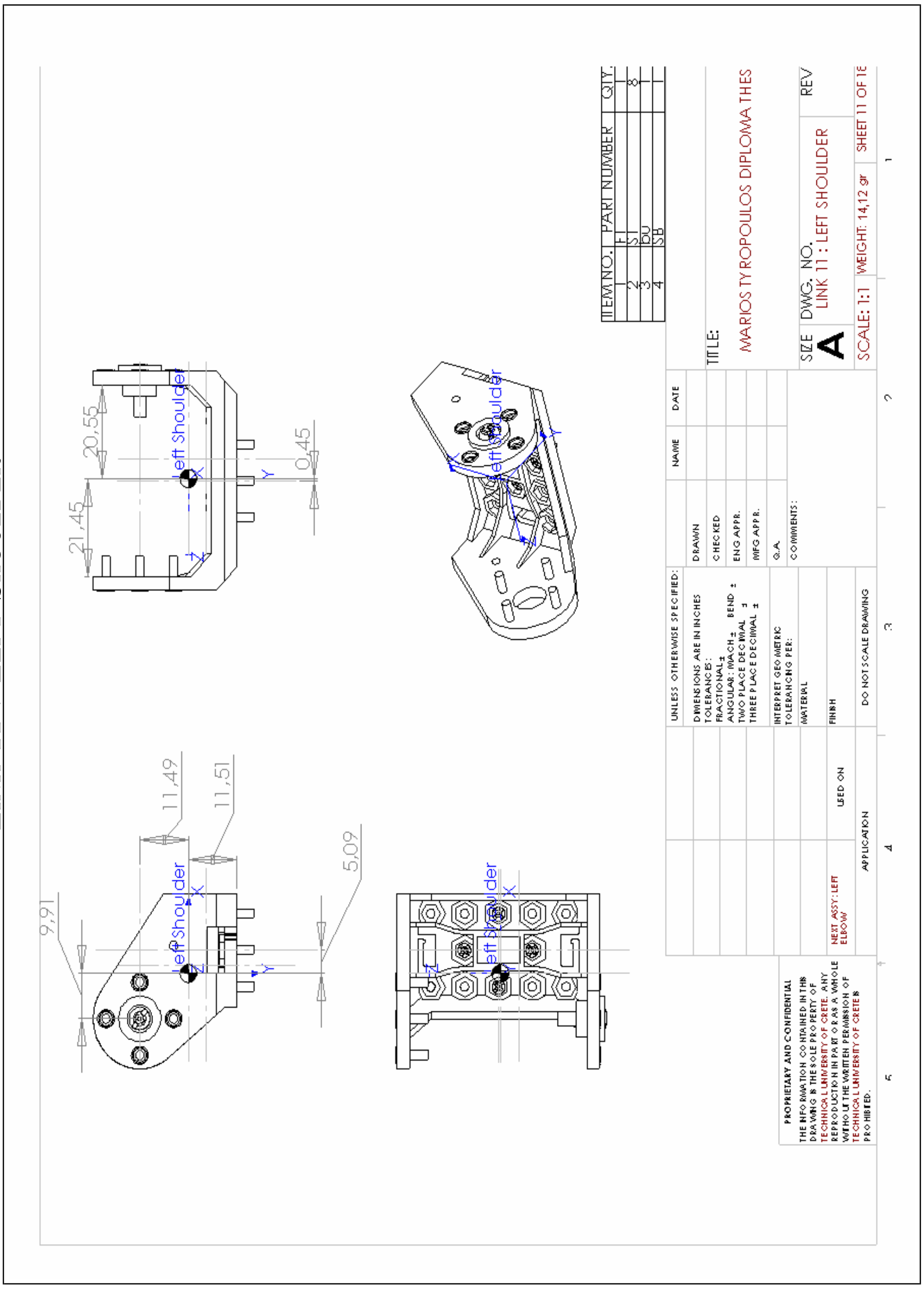
Link 9 : Right Elbow



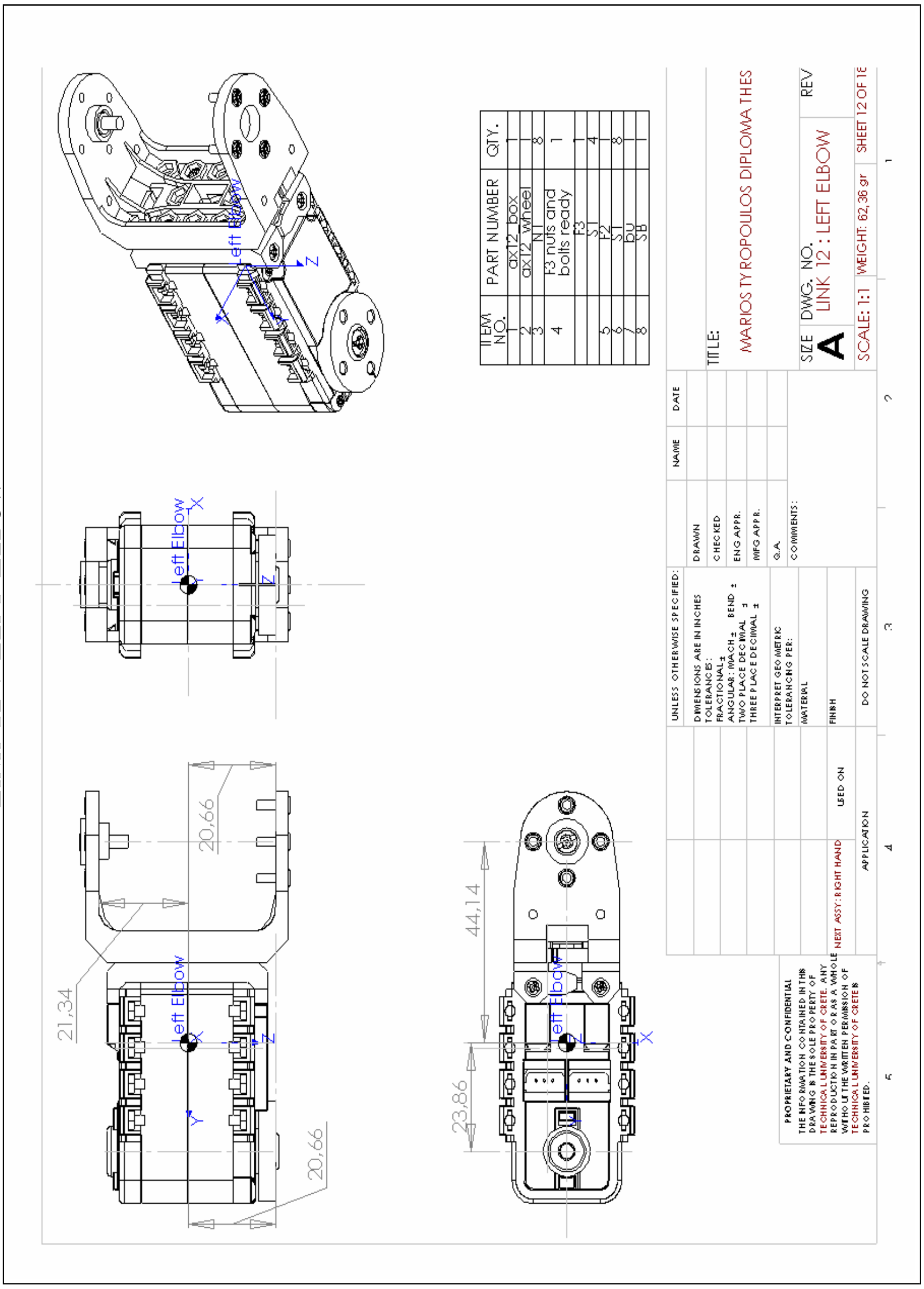
Link 10 : Right Hand



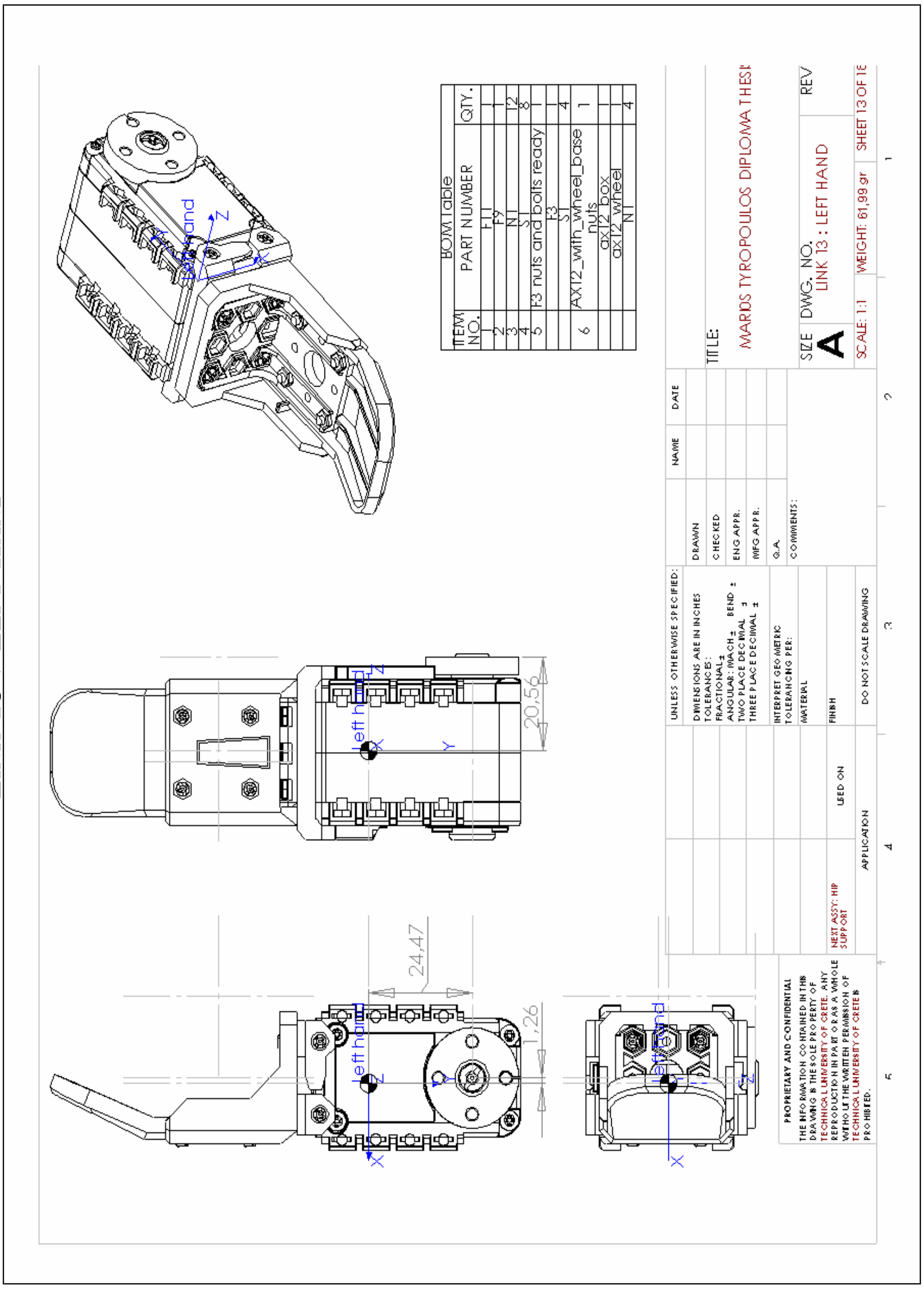
LINK 11 : LEFT SHOULDER



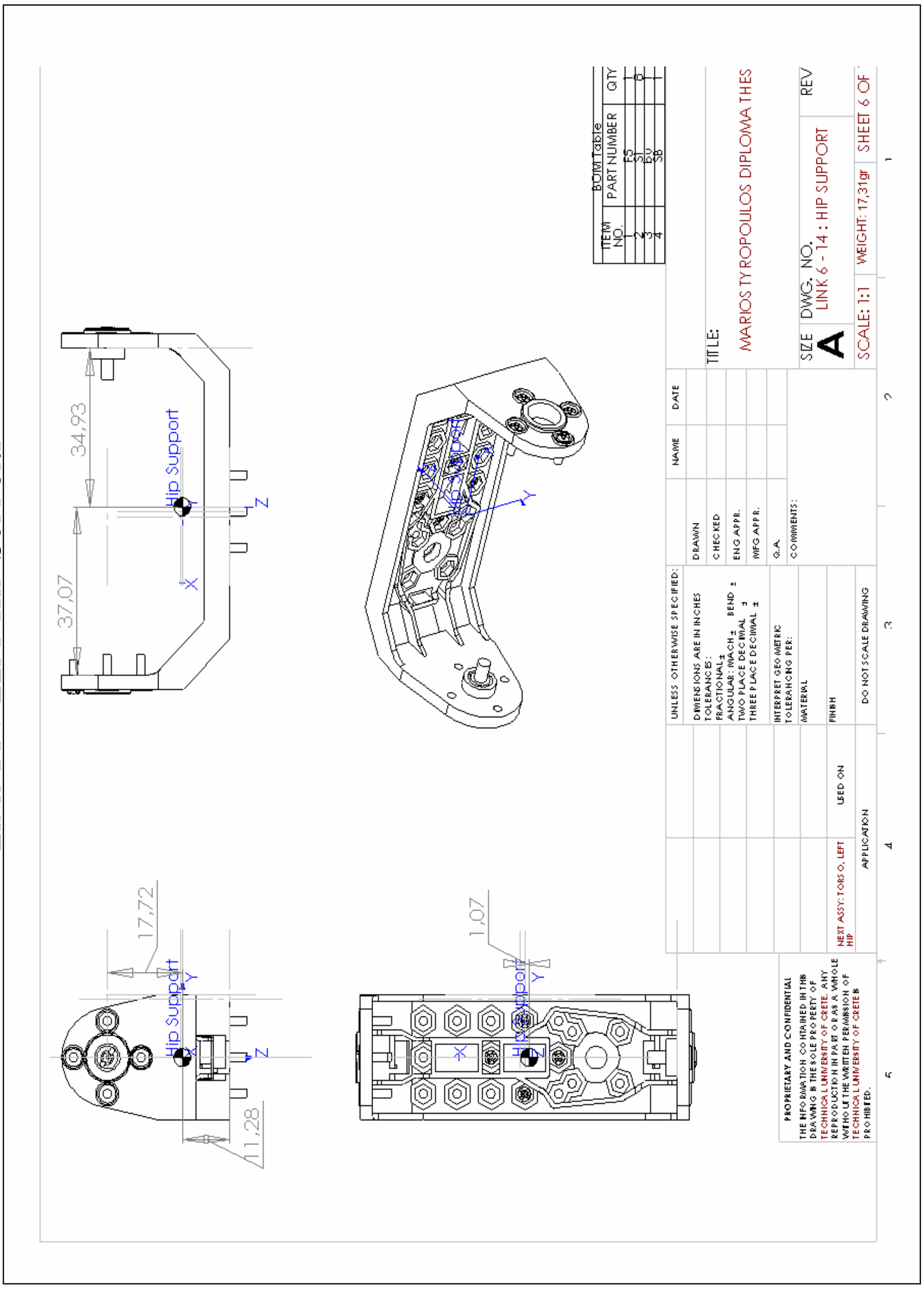
Link 12 : Left Elbow



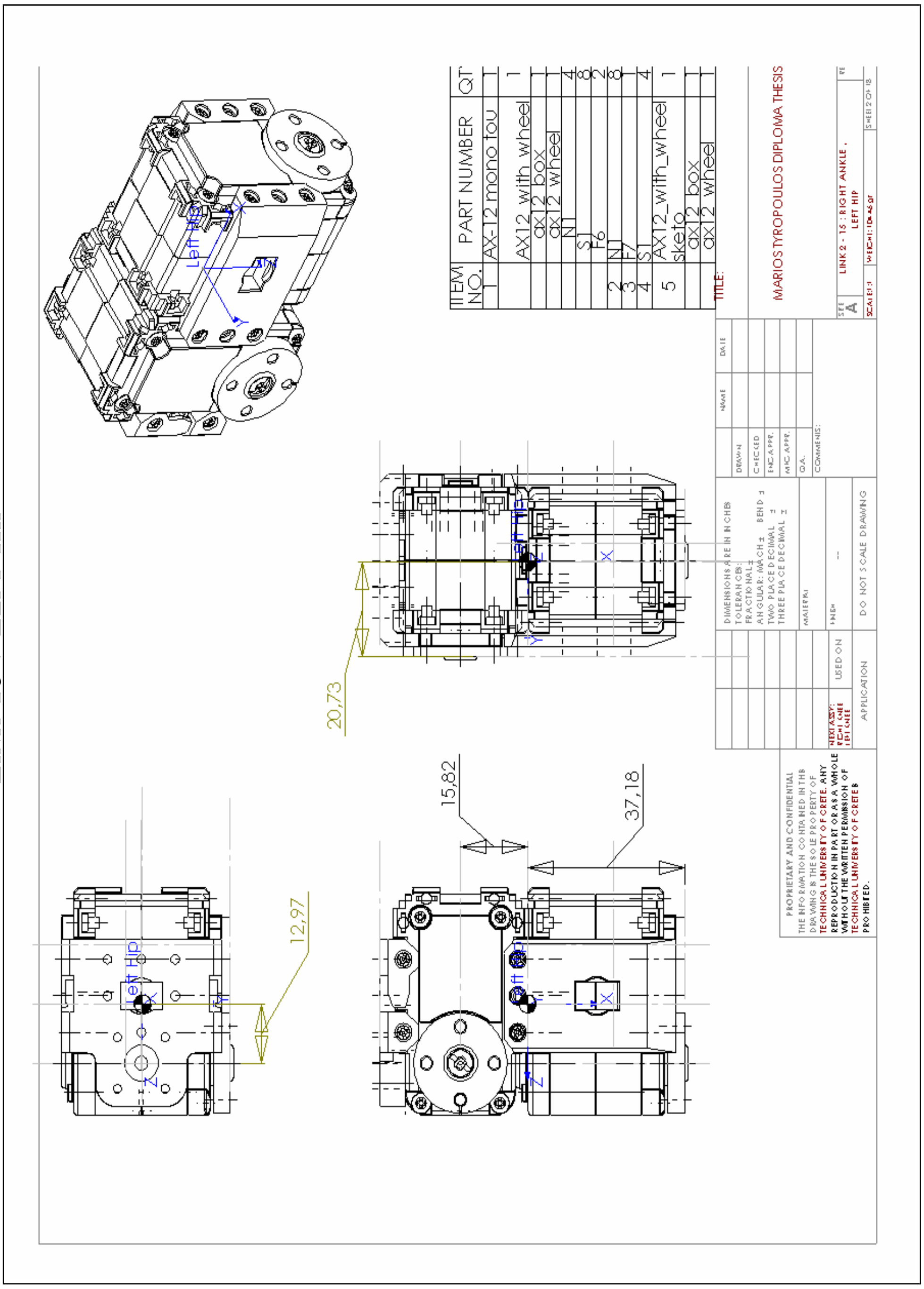
LINK 13 : LEFT HAND



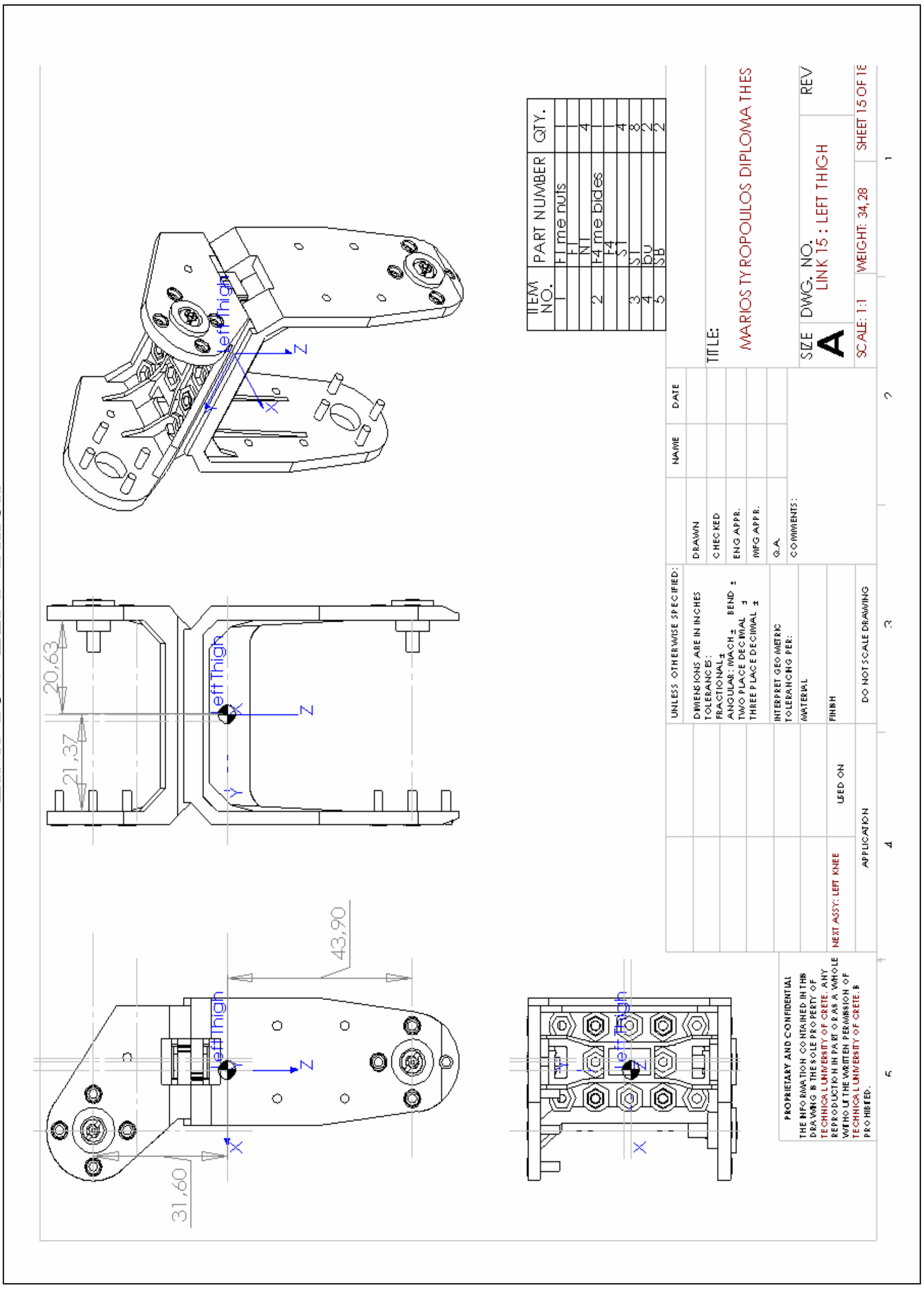
LINK 14 : LEFT HIP SUPPORT



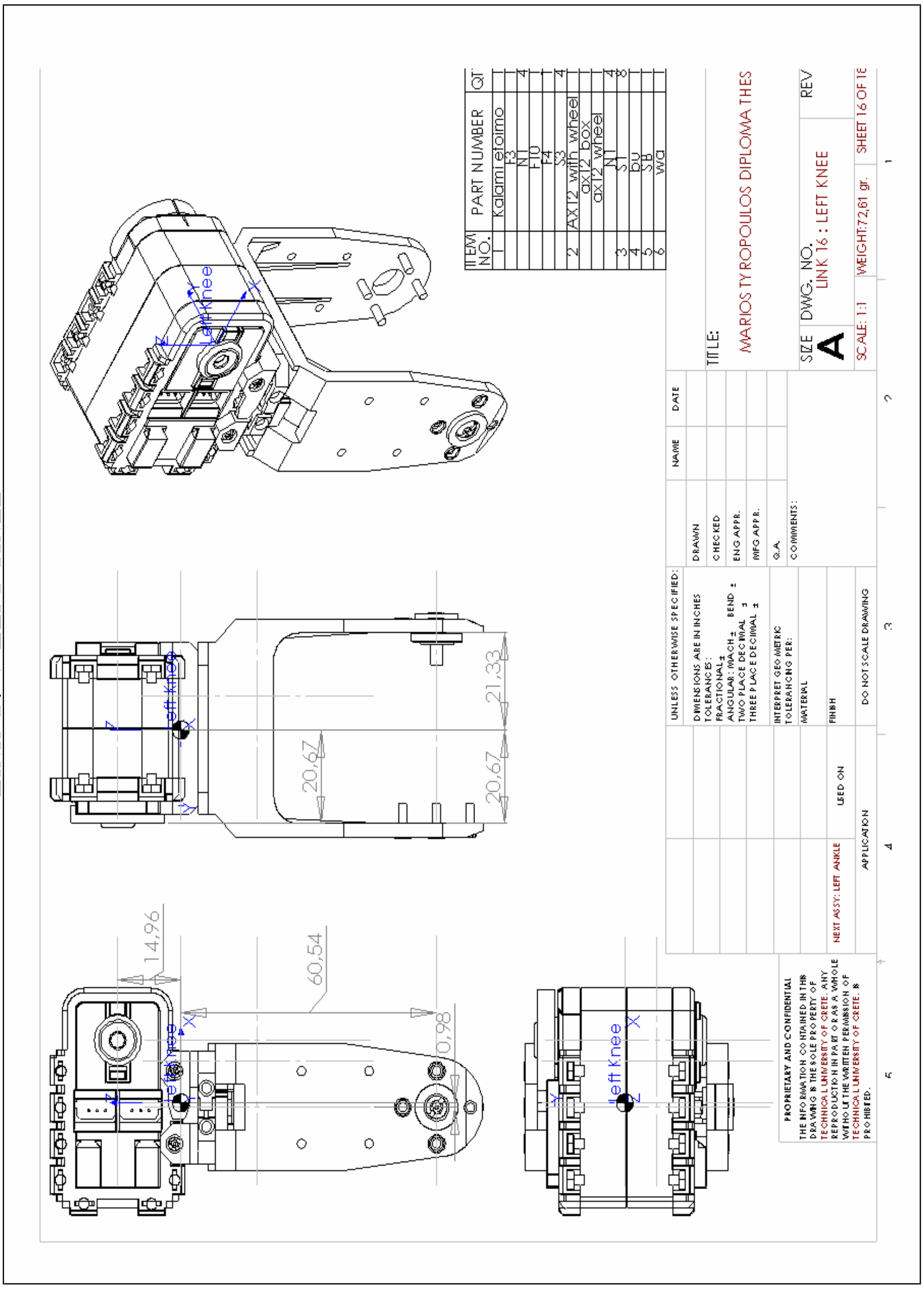
Link 15 : Left Hip



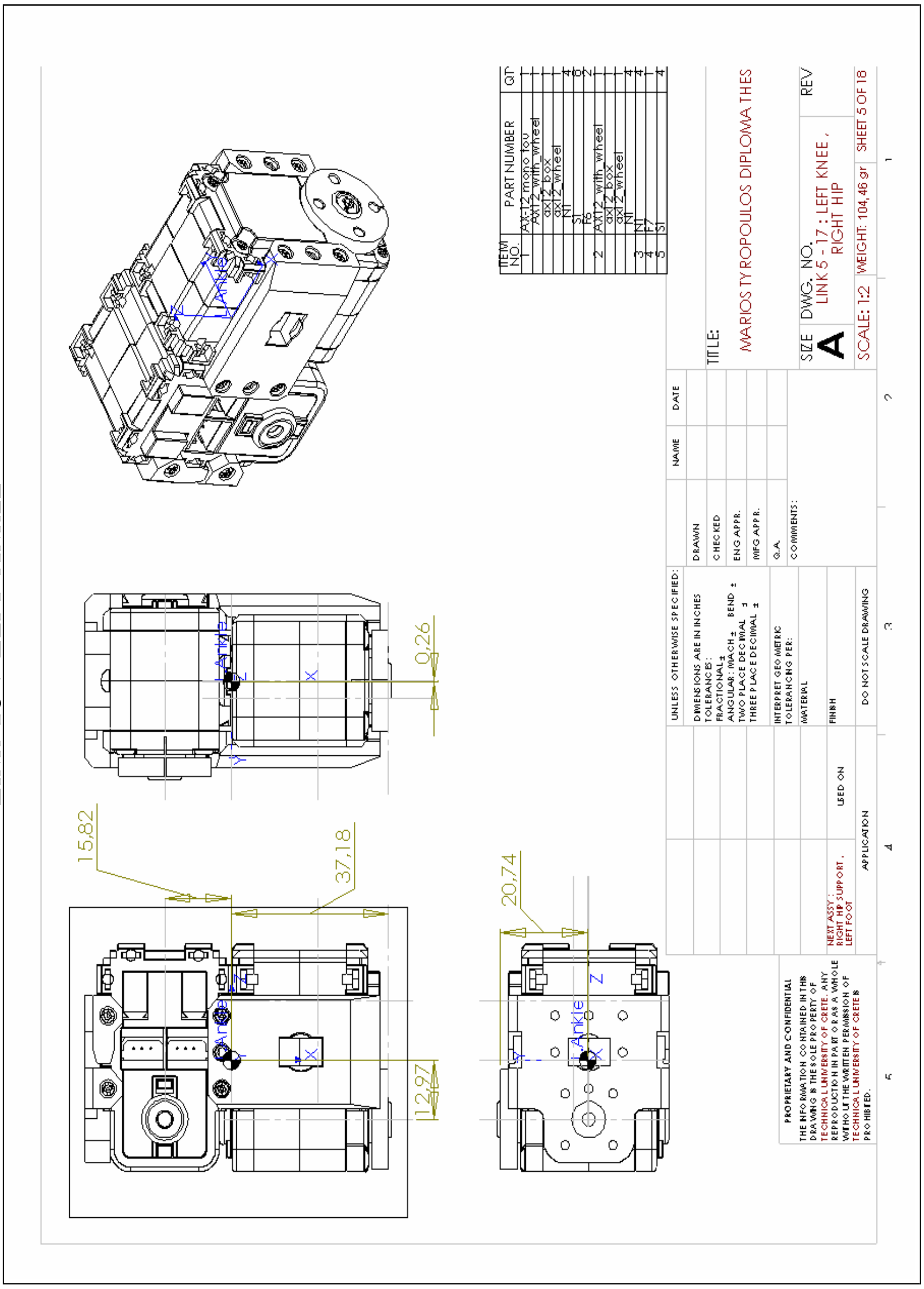
LINK 16 : LEFT THIGH



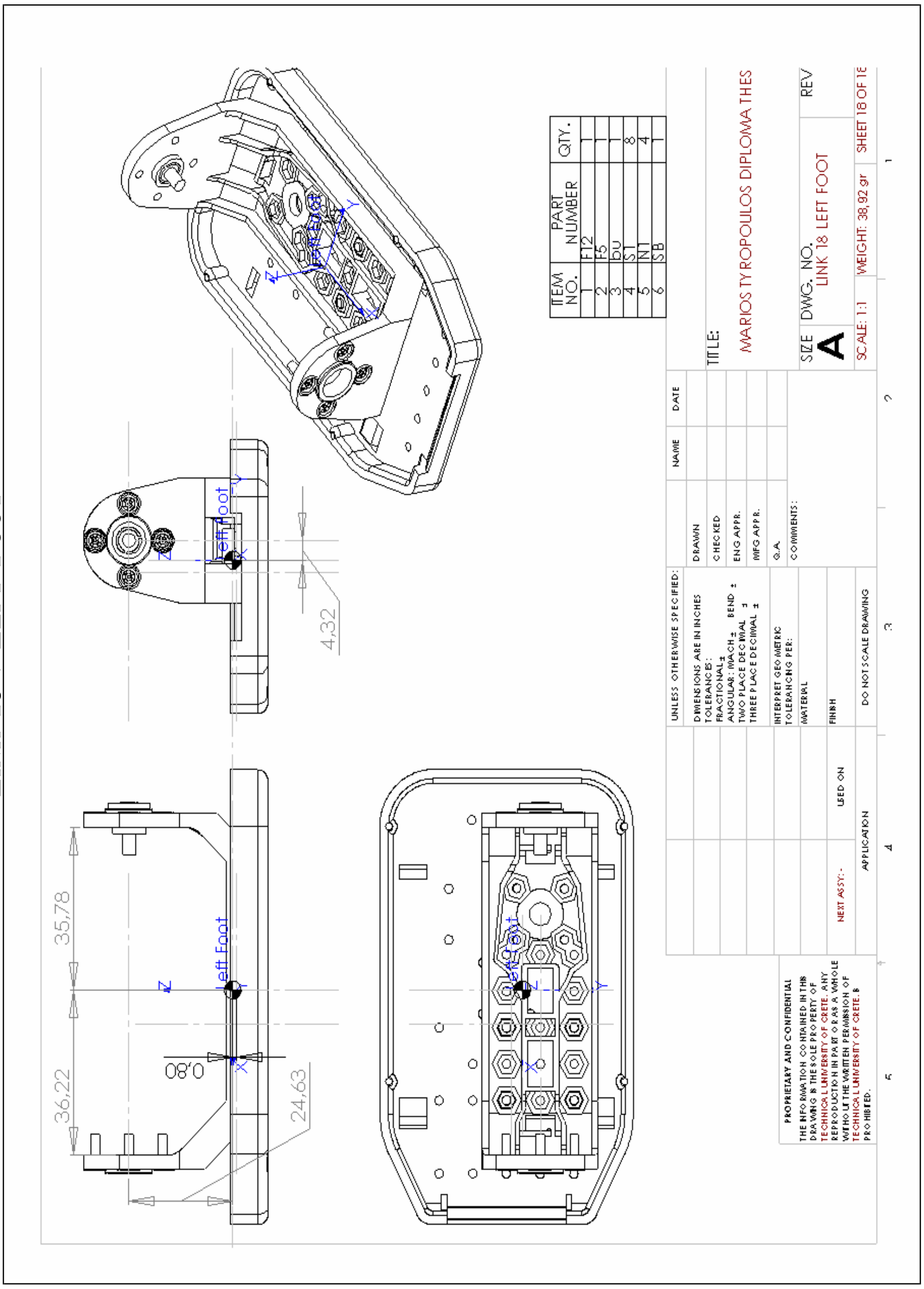
LINK 17 : LEFT KNEE



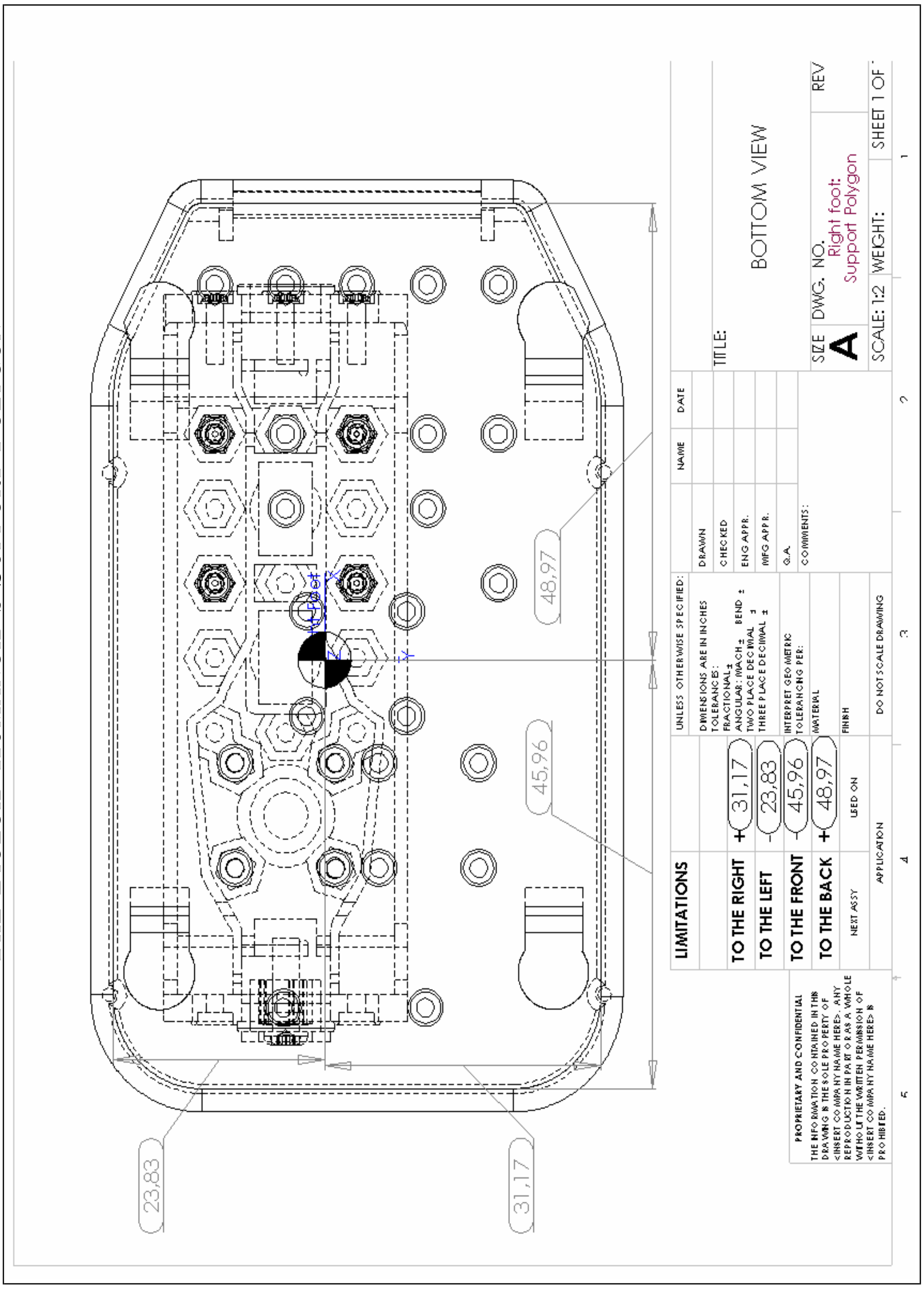
Link 18 : Left Ankle



Link 19 : Left Foot



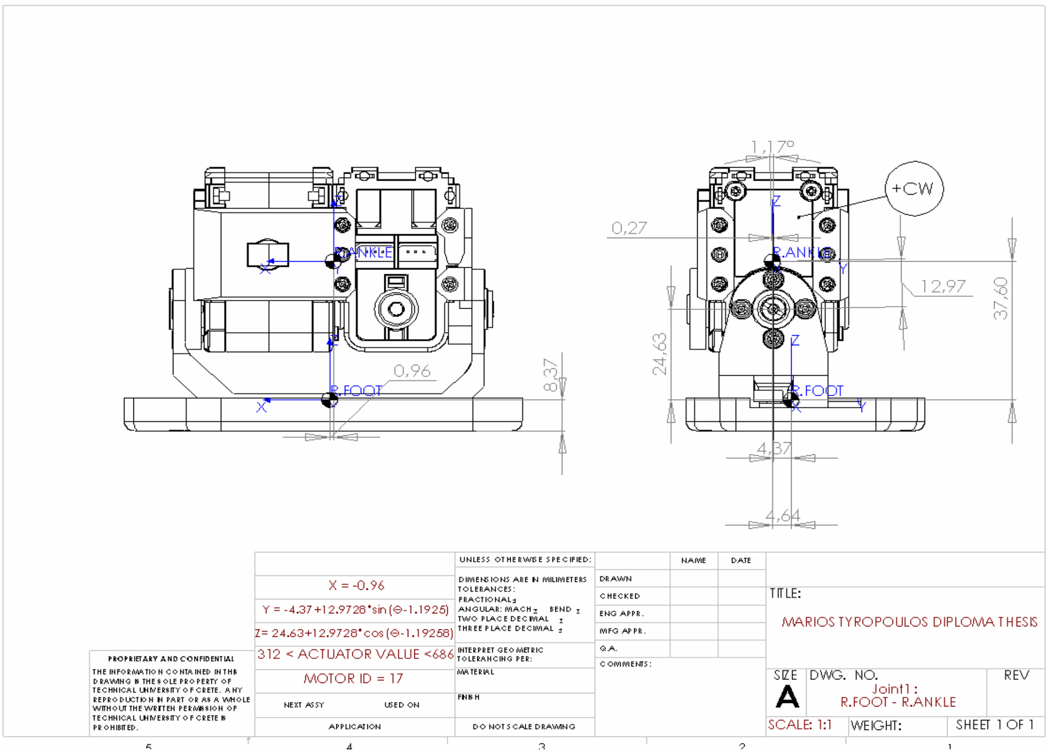
The Bioloid Humanoid's Support Polygon



Chapter 9

JOINTS & EQUATIONS

Joint 1 : - Connectiong Right Foot & Right Ankle



The set of equations describing joint 1 are the following:

$$X = -0.96$$

$$Y = -4.37 + 12.9728 * sin(\theta \ - 1.1925°)$$

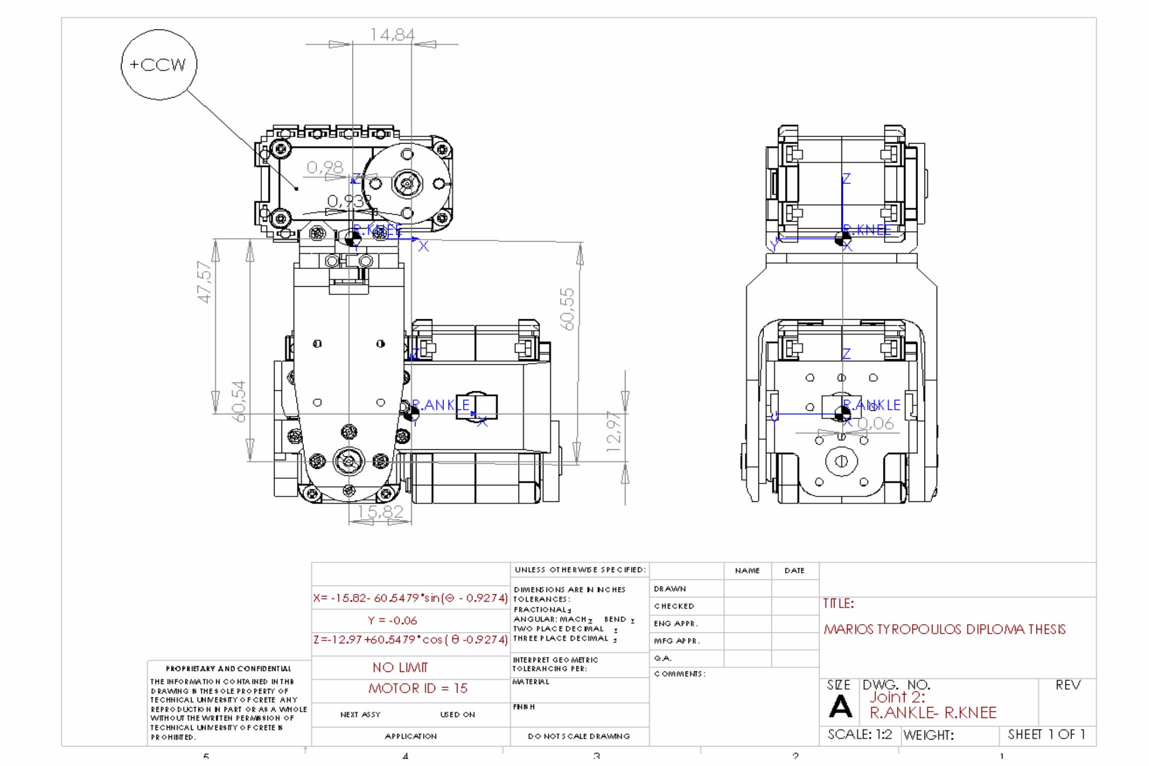
$$Z = 24.63 + 12.9728 * cos(\theta \ - 1.19258°) \qquad (9.1)$$

$$motorID : 17$$

THE LIMITATION OF THE ACTUATOR VALUE IS :
$312 <$ actuator value < 686 .

Transcribed in degress: $-58,64° <$ actuator angle $\theta < 51,0168°$

Joint 2 : - Connecting Right ankle & Right Knee -



The set of equations describing joint 2 are the following:

$$X = -15.82 - 60.5479 * sin(\theta \ - 0.9274°)$$

$$Y = -0.06$$

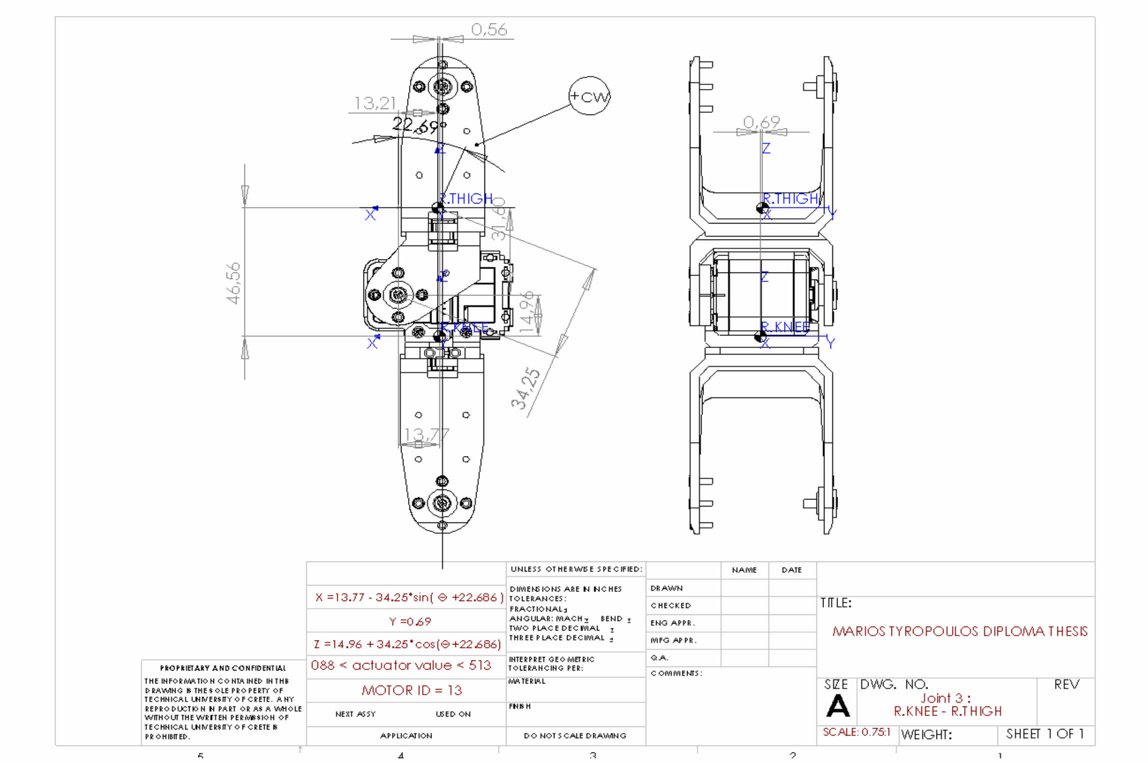
$$Z = -12.97 + 60.5479 * cos(\theta \ - 0,9274°) \tag{9.2}$$

$$motorID : 15$$

There is no limitation for the actuator's value:
0 < actuator value < 1023

Transcribed in degress: $-150° <$ actuator angle $\theta < 150°$

Joint 3 : - Connecting Right Knee & Right Thigh -



The set of equations describing joint 3 are the following:

$$X = 13.77 - 34.25 * sin(\theta \ + 22,686°)$$

$$Y = 0.69$$

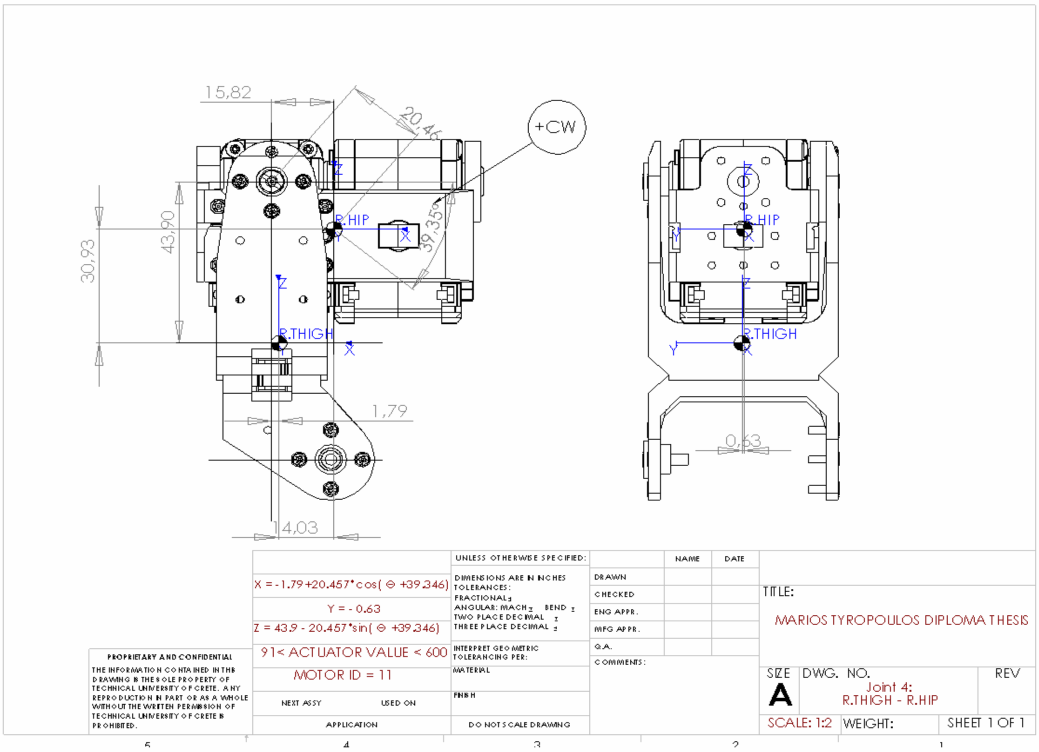
$$Z = 14.96 + 34.25 * cos(\theta \ + 22,686°) \tag{9.3}$$

$$motorID : 13$$

The limitation of the actuator value is :
088 < actuator value < 513

Transcribed in degress: $-124,3168° <$ actuator angle $\theta < 0,2932°$

Joint 4 : - Connecting Right Thigh & Right Hip -



The set of equations describing joint 4 are the following:

$$X = -1.79 + 20.457 * cos(\theta \ + 39.346)^{\circ})$$

$$Y = -0.63$$

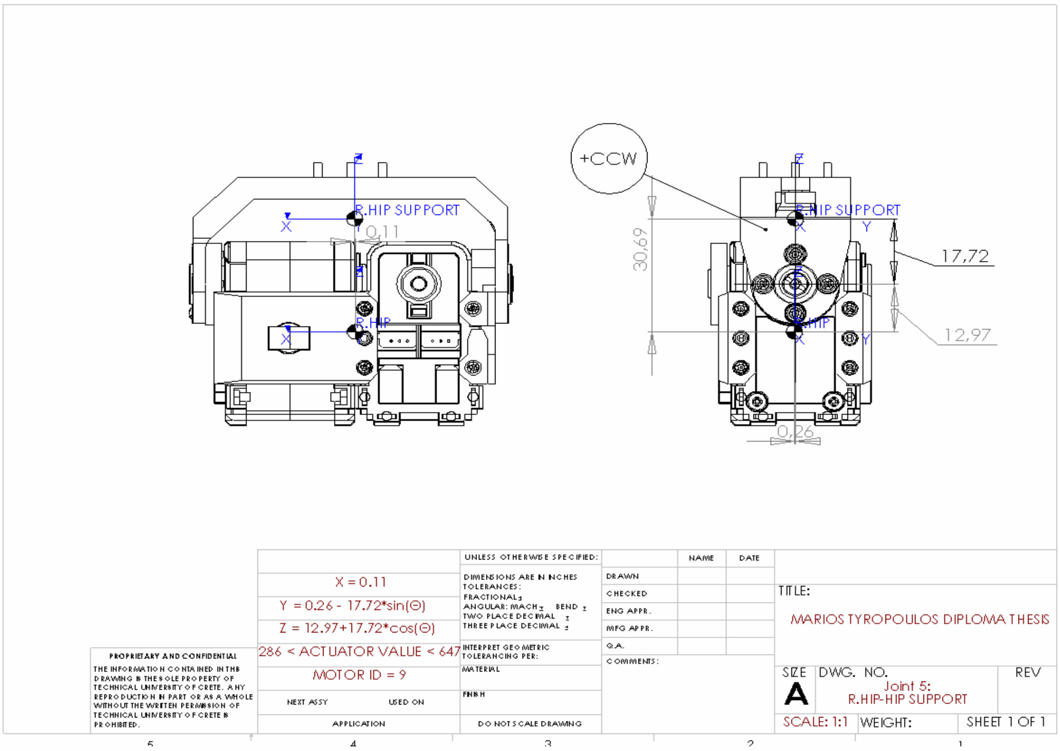
$$Z = 43.9 - 20.457 * sin(\theta \ + 39.346^{\circ}) \tag{9.4}$$

$$motorID : 11$$

The limitation of the actuator value is :
91 < actuator value < 600

Transcribed in degress: $-123,4372^{\circ}$ < actuator angle θ < $25,8016^{\circ}$

JOINT 5 : - CONNECTING RIGHT HIP & RIGHT HIP SUPPORT -



THE SET OF EQUATIONS DESCRIBING JOINT 5 ARE THE FOLLOWING:

$$X = 0.11$$

$$Y = 0.26 - 17.72 * sin(\theta\ ^\circ)$$

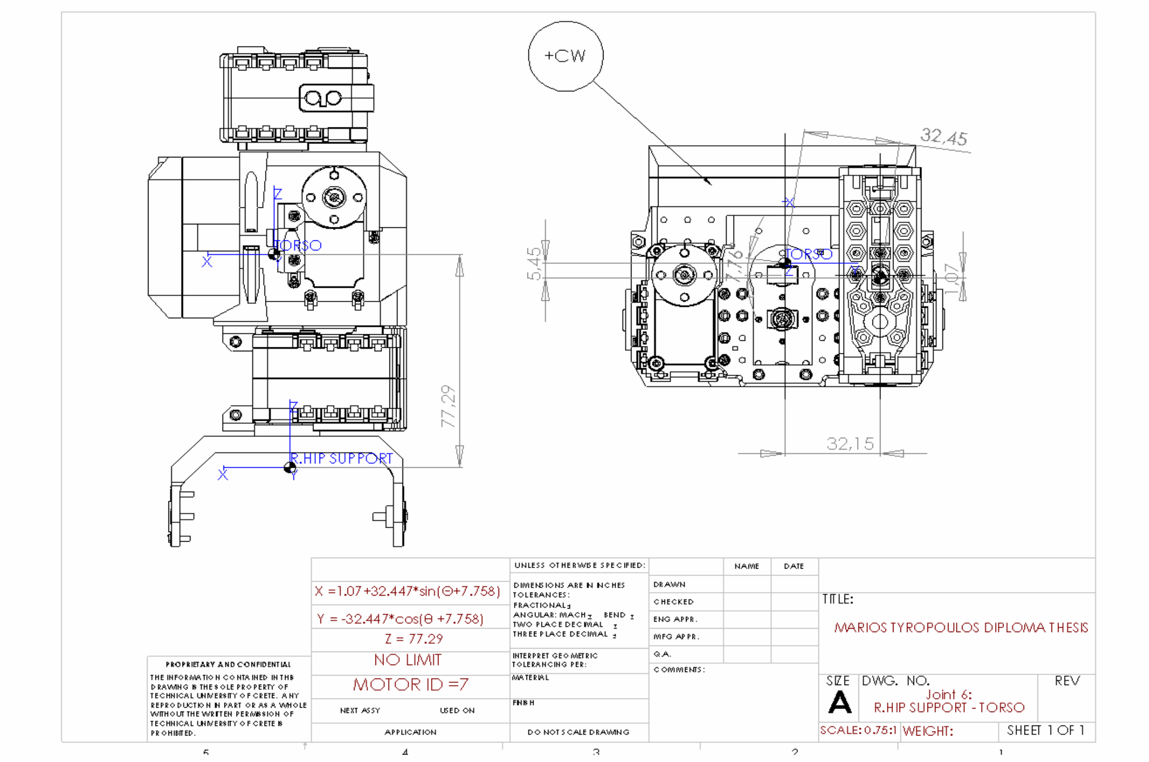
$$Z = 12.97 + 17.72 * cos(\theta\ ^\circ) \tag{9.5}$$

$$motorID : 9$$

THE LIMITATION OF THE ACTUATOR VALUE IS :
286 < actuator value < 647

Transcribed in degress: $-66,2632^\circ <$ actuator angle $\theta < 39,582^\circ$

Joint 6 : - Connecting Right Hip Support & Torso -



The set of equations describing joint 6 are the following:

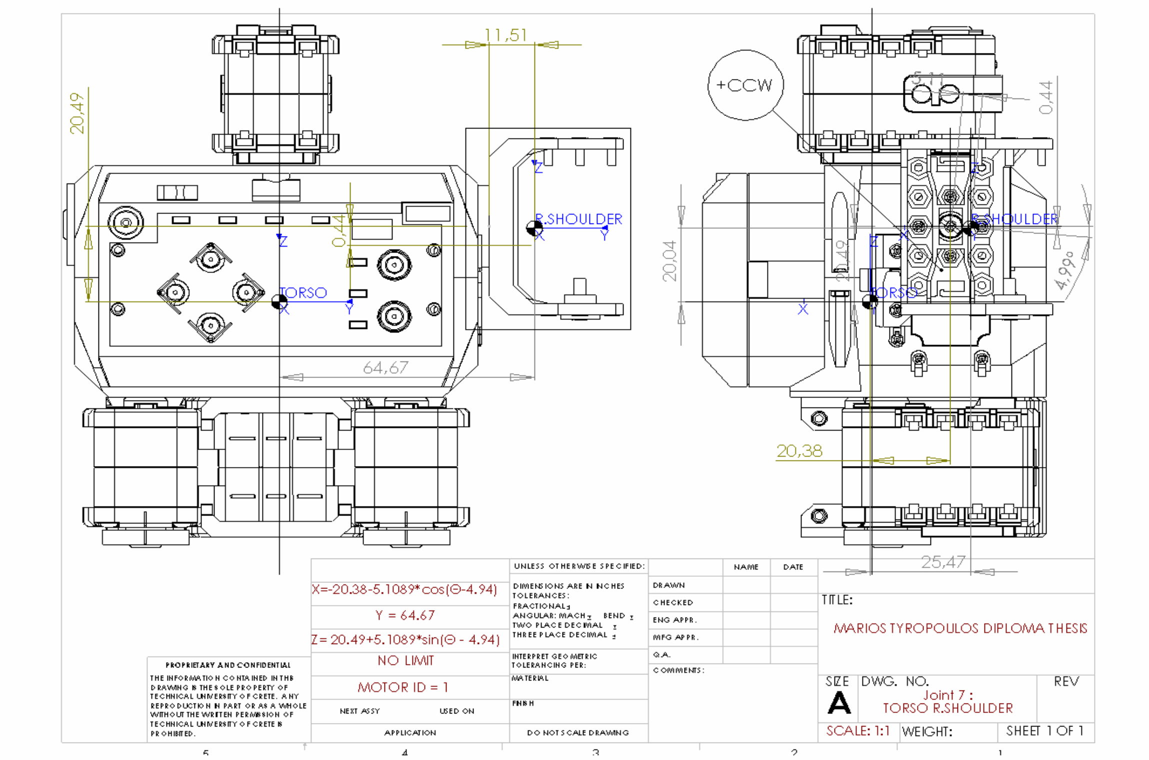
$$X = 1.07 + 32,447 * sin(\theta \ + 7.758°)$$
$$Y = -32.447 * cos(\theta \ + 7.758°)$$
$$Z = 77.29 \tag{9.6}$$

$$motorID : 7$$

The is no limitation for the actuator's value :
0 < actuator value < 1023

Transcribed in degress: $-150° <$ actuator angle $\theta < 150°$

Joint 7 : - Connecting Torso & Right Shoulder -



The set of equations describing joint 7 are the following:

$$X = -20.38 - 5,1089 * cos(\theta \ - 4.94°)$$

$$Y = 64.67$$

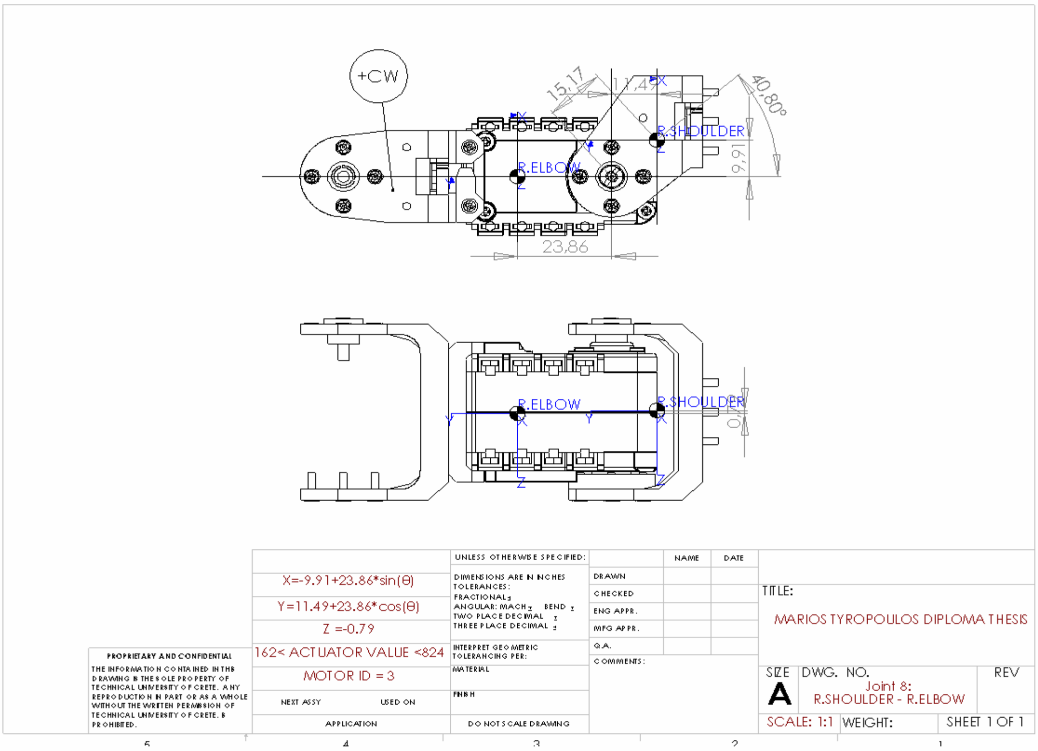
$$Z = 20.49 + 5.1089 * sin(\theta \ - 4.94°) \tag{9.7}$$

$$motorID : 1$$

The is no limitation for the actuator's value :
0 < actuator value < 1023

Transcribed in degress: $-150° <$ actuator angle $\theta < 150°$

Joint 8 : - Connecting Right Shoulder & Right Elbow -



The set of equations describing joint 8 are the following:

$$X = -9.91 + 23.86 * sin\theta\ ^\circ$$

$$Y = 11.49 + 23.86 * cos(\theta\ ^\circ)$$

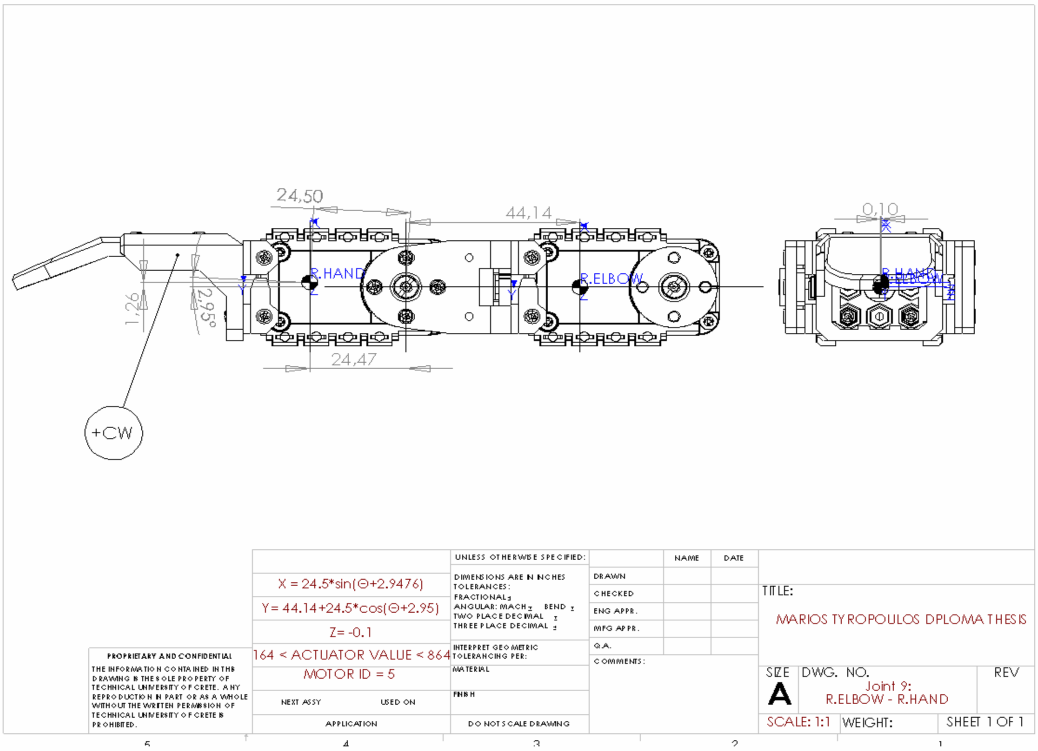
$$Z = -0.79 \tag{9.8}$$

$$motorID : 3$$

The limitation of the actuator value is :
162 < actuator value < 824

Transcribed in degress: $-102, 62^\circ <$ actuator angle $\theta < 91, 4784^\circ$

Joint 9 : - Connecting Right Elbow & Right Hand -



The set of equations describing joint 9 are the following:

$$X = 24.5 * sin(\theta \ + 2.9476^\circ)$$

$$Y = 44.14 + 24.5 * cos(\theta \ + 2.95^\circ)$$

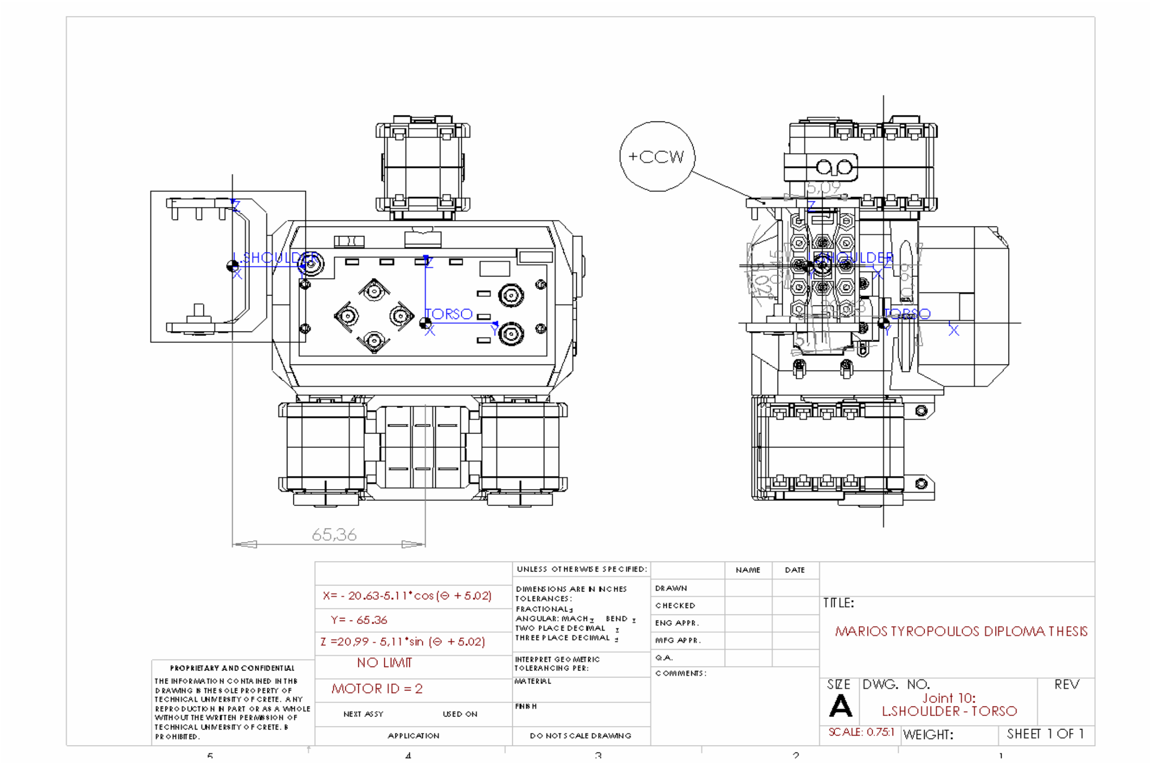
$$Z = -0.1 \tag{9.9}$$

$$motorID : 5$$

The limitation of the actuator value is :
162 < actuator value < 824

Transcribed in degress: $-102,62^\circ <$ actuator angle $\theta < 91,4784^\circ$

JOINT 10 : - CONNECTING TORSO & LEFT SHOULDER -



THE SET OF EQUATIONS DESCRIBING JOINT 10 ARE THE FOLLOWING:

$$X = -20.63 - 5.11 * cos(\theta \ + 5.02°)$$

$$Y = -65.36$$

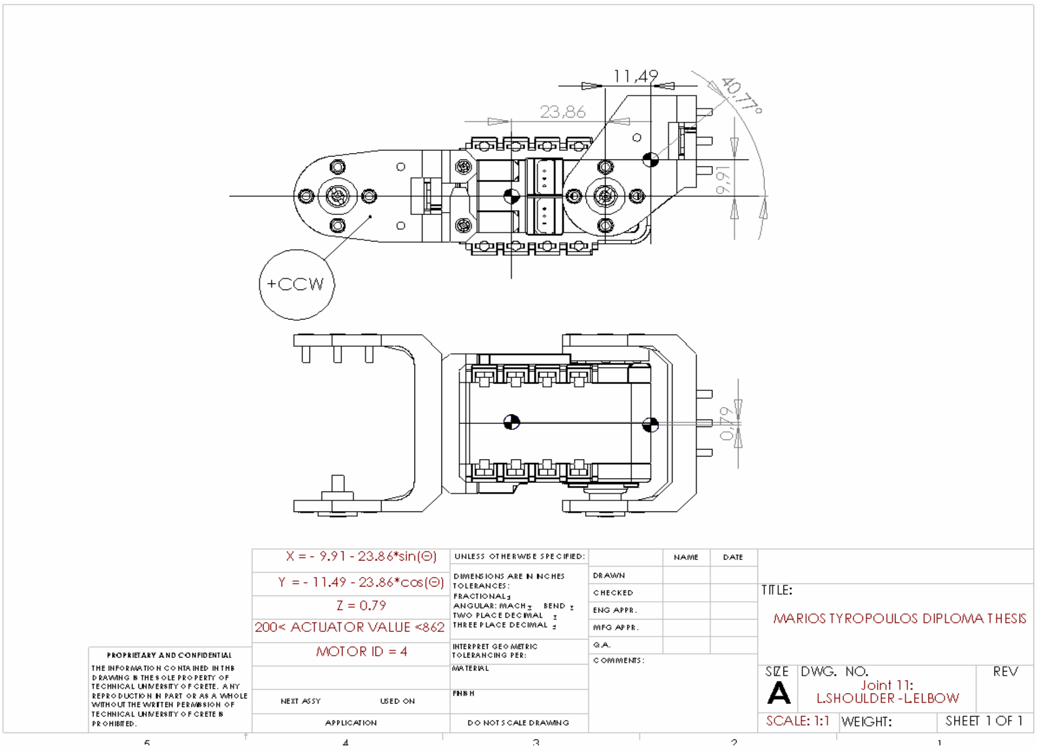
$$Z = 20,99 - 5,11 * sin(\theta \ + 5.02°) \tag{9.10}$$

$$motorID : 2$$

THERE IS NO LIMITATION FOR THE ACTUATOR'S VALUE :
$0 <$ actuator value < 1023

Transcribed in degress: $-150° <$ actuator angle $\theta < 150°$

Joint 11 : - Connecting Left Shoulder & Left Elbow -



The set of equations describing joint 11 are the following:

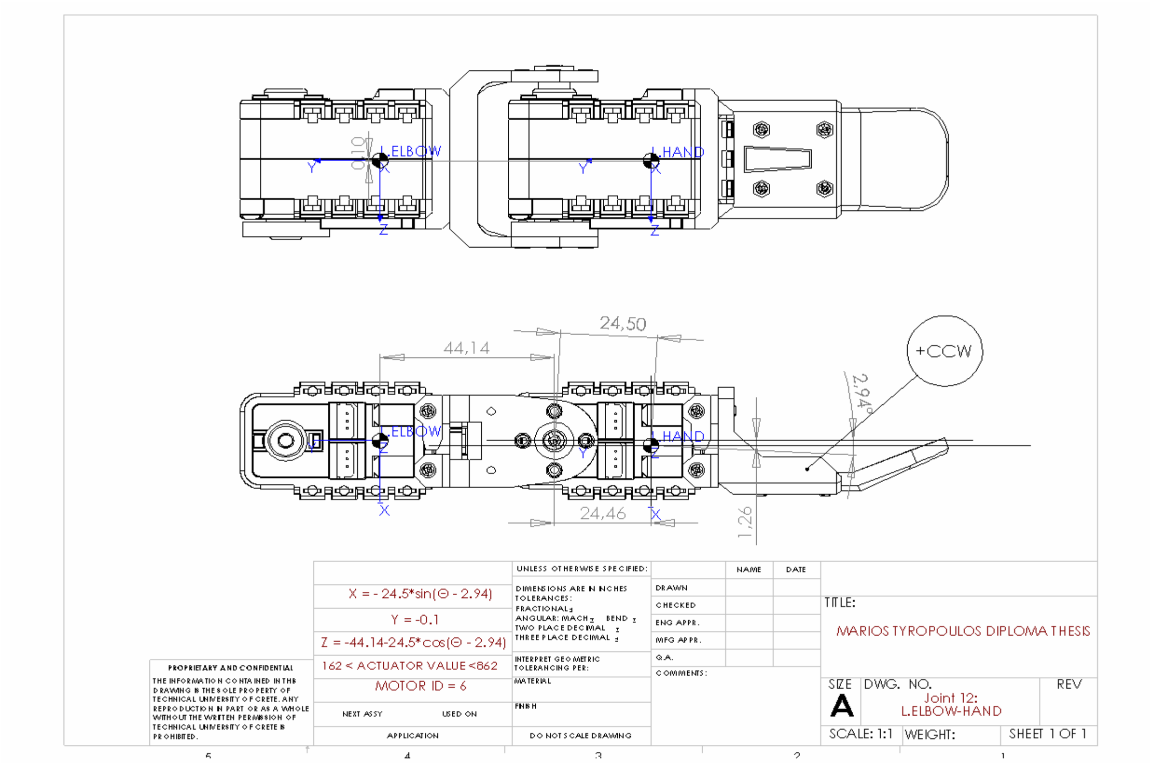
$$X = -9.91 - 23.86 * sin(\theta\ ^{\circ})$$
$$Y = -11.49 - 23.86 * cos(\theta\ ^{\circ})$$
$$Z = 0.79 \tag{9.11}$$

$$motorID : 4$$

The limitation of the actuator value is :
200 < actuator value < 862

Transcribed in degress: $-91,4784° <$ actuator angle $\theta < 102,62°$

Joint 12 : - Connecting Left Elbow& Left Hand -



The set of equations describing joint 12 are the following:

$$X = -24.5 * sin(\theta \ - 2.94°)$$

$$Y = -0.1$$

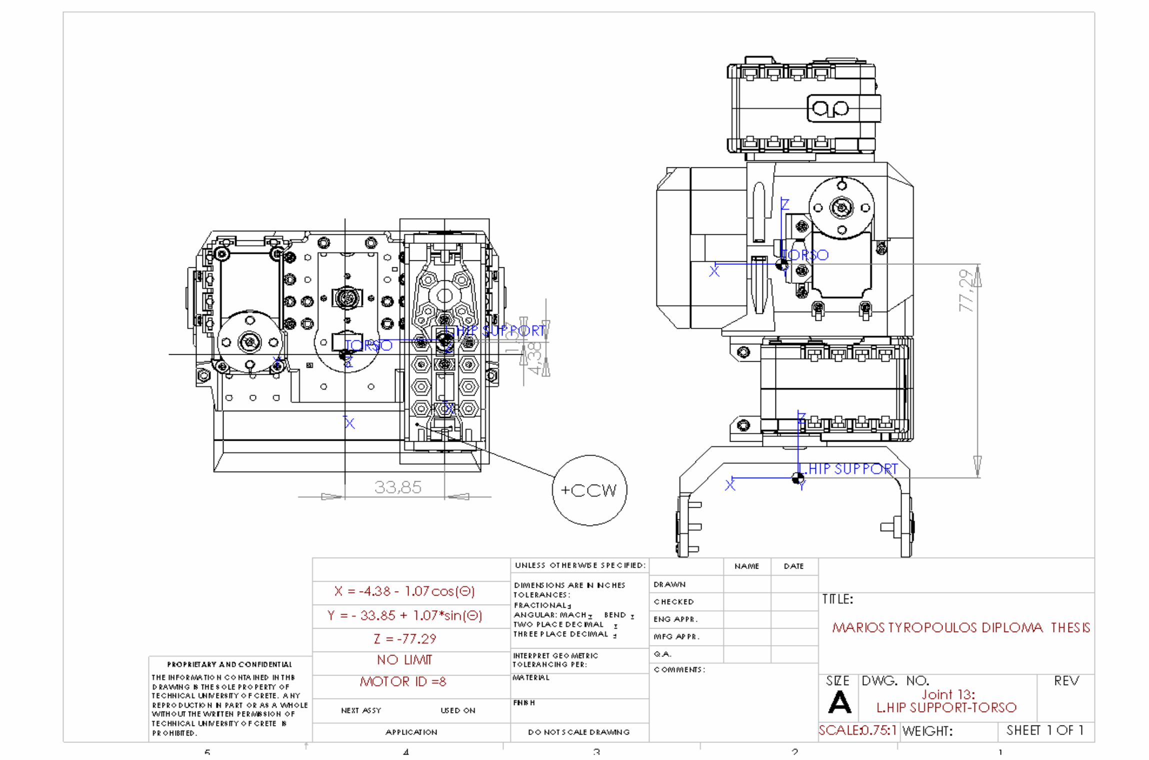
$$Z = -44.14 - 24.5 * cos(\theta \ - 2.94°) \tag{9.12}$$

$$motorID : 6$$

The limitation of the actuator value is :
162 < actuator value < 862

Transcribed in degress: $-102,62° <$ actuator angle $\theta < 102,62°$

Joint 13 : - Connecting Torso& Left Hip Support -



The set of equations describing joint 13 are the following:

$$X = -4.38 - 1.07cos(\theta)$$

$$Y = -33.85 + 1.07 * sin(\theta)$$

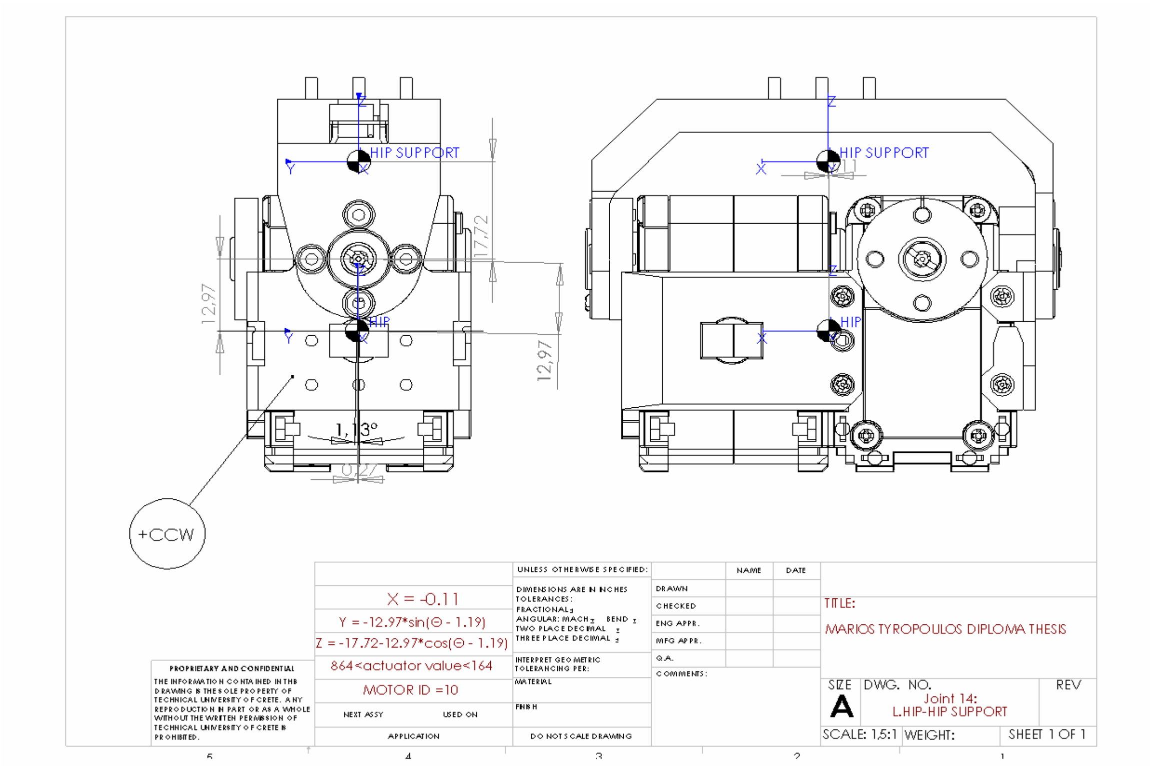
$$Z = -77.29 \tag{9.13}$$

$$motorID : 8$$

There is no limitation for the actuator's value :
$0 <$ actuator value < 1023

Transcribed in degress: $-150° <$ actuator angle $\theta < 150°$

Joint 14 : - Connecting Left Hip Support & Left Hip



The set of equations describing joint 14 are the following:

$$X = -0.11$$

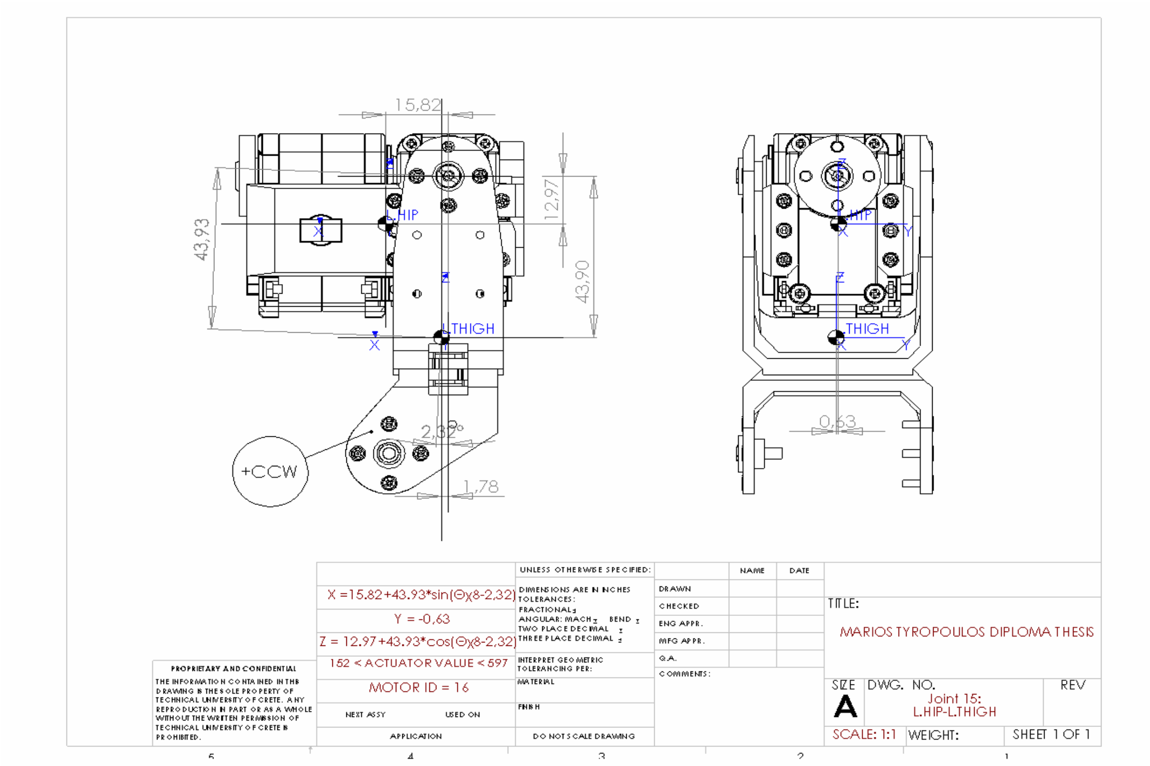
$$Y = -12.97 * sin(\theta \; - 1.19°)$$
$$Z = -17.72 - 12.97 * cos(\theta \; - 1.19°) \quad (9.14)$$

$$motorID = 10$$

The limitation of the actuator value is :
377 < actuator value < 738

Transcribed in degress: $-39,582°$ < actuator angle θ < $66,2632°$

Joint 15 : - Connecting Left Hip & Left Thigh



The set of equations describing joint 15 are the following:

$$= 15.82 + 43.93 * sin(\theta \;\; - 2,32°)$$

$$Y = -0,63$$

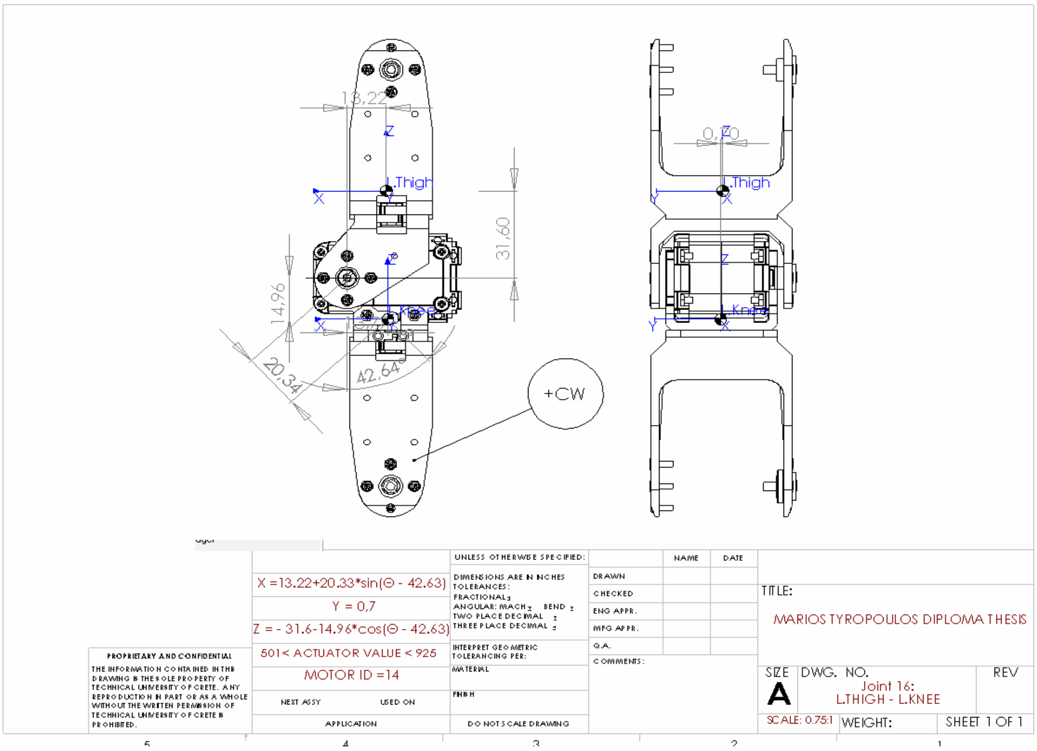
$$= 12.97 + 43.93 * cos(\theta \;\; - 2,32°) \qquad (9.15)$$

$$motorID : 16$$

The limitation of the actuator value is :
152 < actuator value < 597

Transcribed in degress: $-105,552° <$ actuator angle $\theta < 24,922°$

Joint 16 : - Connecting Left Thigh & Left Knee



The set of equations describing joint 16 are the following:

$$= 13.22 + 20.33 * sin(\theta \ - 42,63°)$$

$$= 0,7$$

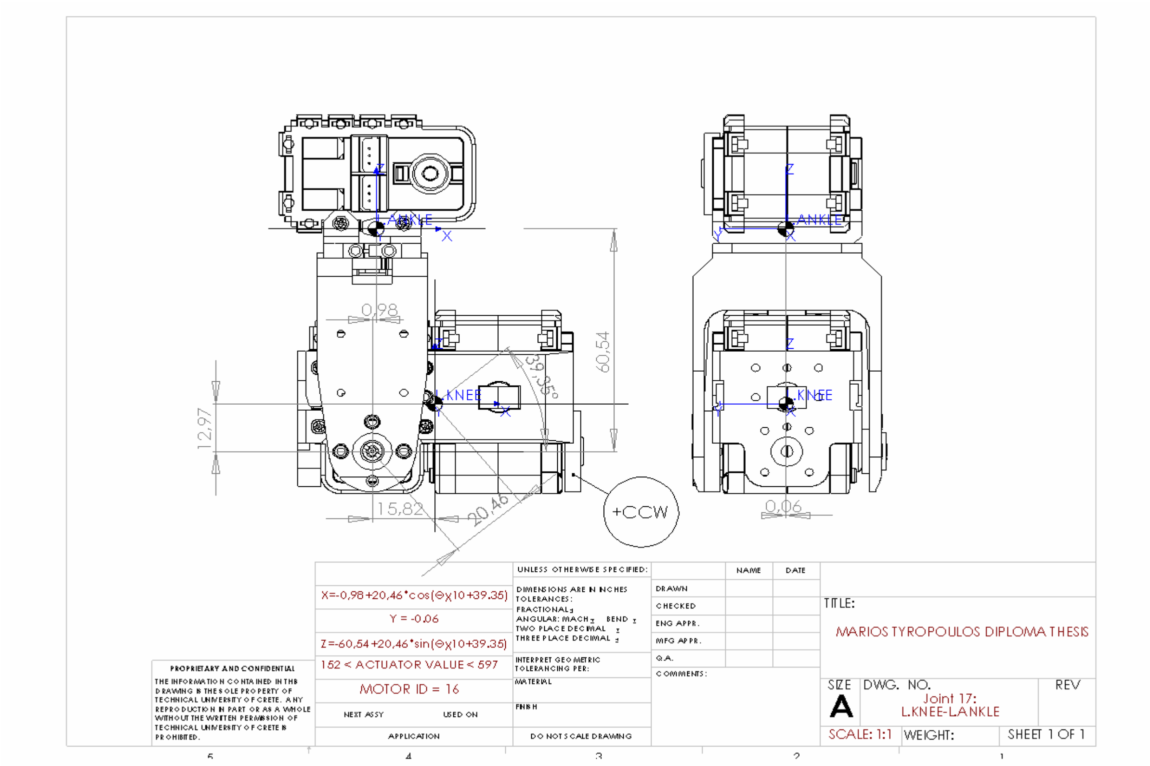
$$Z = -31.6 - 14.96 * cos(\theta \ - 42,63°) \tag{9.16}$$

$$motorID : 14$$

The limitation of the actuator value is :
501 < actuator value < 925

Transcribed in degress: $-3,2252° <$ actuator angle $\theta < 121,0916°$

Joint 17 : - Connecting Left Knee & Left Ankle



The set of equations describing joint 17 are the following:

$$X = -0,98 + 20,46 * cos(\theta \ + 39,35°)$$

$$Y = -0.06$$

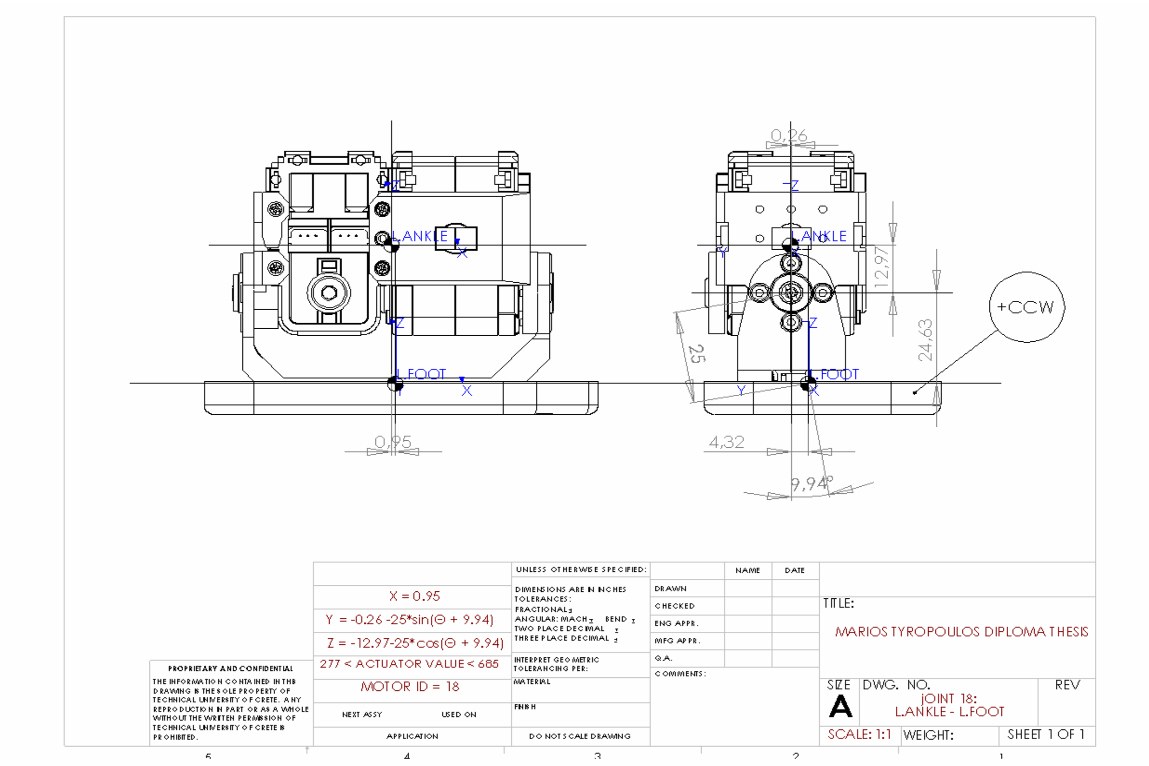
$$= -60,54 + 20,46 * sin(\theta \ + 39,35°) \tag{9.17}$$

$$motorID : 17$$

The limitation of the actuator value is :
312 < actuator value < 686

Transcribed in degress: $-58,64° <$ actuator angle $\theta < 51,0168°$

Joint 18 : - Connecting Left Ankle & Left Foot



The set of equations describing joint 18 are the following:

$$X = 0.95$$

$$Y = -0.26 - 25 * sin(\theta \ + 9.94^{\circ})$$
$$= -12,97 - 25 * cos(\theta \ + 9.94^{\circ}) \quad (9.18)$$

$$motorID : 18$$

The limitation of the actuator value is :
$277 <$ actuator value < 685

Transcribed in degress: $-68,902^{\circ} <$ actuator angle $\theta < 50,7236^{\circ}$

Bibliography

[1] Antonio Pickel. *Control for a Biped Robot with minimal number of Actuators* University of Applied Sciences Koblenz Department of Electrical Engineering, Diplomarbeit (MAY 2003)

[2] Timothy Bretl. *Motion Planning of Multi-Limbed Robots Subject to Equilibrium Constraints* International Journal of Robotics Research Volume 25 , Issue 4 (April 2006)

[3] Philippe Sardain and Guy Bessonnet. *Forces Acting on a Biped Robot. Center of PressureZero Moment Point.* IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS Part A: SYSTEMS AND HUMANS Volume 34, Issue 5 (SEPTEMBER 2004)

[4] Byung-Hun Hwang, Jung-Shik Kong, Bo-Hee Lee ,Jin-Geol Kim and Uk-Youl Huh. *ZMP Compensation Algorithm for Stable Posture of a Humanoid Robot.* Institute of Control, Robotics and systems .Journal of control automation and systems engineering Volume 11, Issue 1 (JANUARY 2005)

[5] McGhee, and Frank. *Generation of Periodic Gaits.* Quadrupedal Locomotion Springer London (February 2007)

[6] Freyr Hardarson.*Stability analysis and synthesis of statically balanced walking for quadruped robots.* Doctoral Thesis Mechatronics Lab Department of Machine Design Royal Institute of Technology, KTH (2002)

[7] Rich Hooper. *www.learnaboutrobots.com*

[8] Pedro Teodoro *Humanoid Robot Development of a simulation environment of an entertainment humanoid robot* (April 2007)

[9] Bio Robotics Laboratory University Of Washington. *http://brl.ee.washington.edu/*

[10] TU Berlin. *Powered Hand Exosceleton* http://pdv.cs.tu-berlin.de/HandExoskeleton/HandExoIntro.html

[11] James J. Kuffner, Satoshi Kagami, Koichi Nishiwaki, Masayuki Inaba and Hirochika Inoue *Dynamically-stableMotion Planning for Humanoid Robots* Autonomous Robots Volume 12, Issue 1 pp 105-118 (JANUARY 2002)