DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

EMBEDDED DESIGN OF BALANCE BOARD FOR BALANCE AND LOCOMOTOR SYSTEM FUNCTION TESTS

by

Cihan DAYANGAÇ

June, 2019

İZMİR

EMBEDDED DESIGN OF BALANCE BOARD FOR BALANCE AND LOCOMOTOR SYSTEM FUNCTION TESTS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science of Department of Biomedical Technologies Program

by

Cihan DAYANGAÇ

June, 2019

İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"EMBEDDED DESIGN OF BALANCE BOARD FOR BALANCE AND LOCOMOTOR SYSTEM FUNCTION TESTS"** completed by **CİHAN DAYANGAÇ** under supervision of **ASSIST. PROF. DR. AHMET ÖZKURT** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Ahmet ÖZKURT

Supervisor

Prof. Dr. Seydi DOĞAN

(Jury Member)

Prof. Dr Mehmet Kuntalp

(Jury Member)

Prof.Dr. Kadriye ERTEKİN

Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to thank everyone who supported and encouraged me in every respect, who helped me to achieve my master's degree and ensure academic performance.

I would like to thank Assist. Prof. Dr. Ahmet ÖZKURT, my esteemed adviser, who will share his valuable knowledge in the realization of this work and will never forget the value of every word that he uses.

I would like to thank my family, Hacer DAYANGAÇ and Fahri DAYANGAÇ, who support me in all cases and do not spare any technical information

I would like to thank my friends, who helped me and supported me at every stage of my work, Gizem ALKAN, Enes YAMAN, Can YILDIRIM, Aykut ALTIOK and Başak LÖKLÜOĞLU ONGAR.

I would like to express my gratitude to Prof.Dr. Zeki KIRAL, Assist. Prof. Dr. Taner AKKAN, Dr. Tolga OLCAY ve Dr. Ata ELVAN my esteemed teachers for their support and knowledge.

This study is supported by TUBITAK (The Scientific and Technological Research Council of Turkey) a number 215M933 under the project. Thanks to TÜBİTAK for their support.

Cihan DAYANGAÇ

EMBEDDED DESIGN OF BALANCE BOARD FOR BALANCE AND LOCOMOTOR SYSTEM FUNCTION TESTS

ABSTRACT

In this study, a balance platforms prototype with three motor two degrees of freedom was designed for advanced physiotherapy scenarios by taking of the consideration with general features of parallel mechanisms, kinematics and study spaces is constructed. This prototype structure, considering the problems that limited usage areas such as existing balance assessment and training platform systems need fixed and large areas lack of portable features, is designed to be portable, small in size and movable in two degrees of freedom. By Portable active balance platform prototype controlled by an embedded computer, it will also be possible to conduct tests and exercises routinely in clinics with a remote and traceable manner. As a result of these studies, two pre-prototype and a real scale prototype which testable movement of humanoid robot which is average human weight has produced. During the ongoing process within this project; development of software and improved interface which one processing of data that comes from prototype and user friendly interface software has completed. A patent application has been made for the entire platform with the combination of mechanic, electronic and cloud data software systems by the Scientific Research Project Group from Dokuz Eylül University.

Keywords: Motion platform, human balance, two degree of freedom active balance platform

DENGE VE LOKOMOTOR SİSTEM FONKSİYON TESTLERİ İÇİN DENGE PLATFORMU GÖMÜLÜ TASARIMI

ÖZ

Bu çalışmada, paralel mekanizmaların genel özelikleri, kinematik ve çalışma uzayları hakkında yapılan araştırmalar göz önünde bulundurularak ileri derece fizyoterapi senaryolarına yönelik olarak 3 motorlu 2 serbestlik derecesine sahip bir denge platformu prototipi tasarlanmış ve gerçeklenmiştir. Bu prototip yapı, mevcut denge değerlendirme ve eğitim platform sistemlerinin sabit ve geniş alanlara ihtiyaç duymaları, taşınabilir özeliklerinin bulunmaması gibi kullanım alanlarını kısıtlayan sorunlar göz önünde bulundurularak taşınabilir, küçük boyutlu ve iki serbestlik derecesinde hareketli olarak tasarlanmıştır. Gömülü bilgisayar tabanlı bu taşınabilir aktif denge platformu prototipi ile rutin olarak kliniklerde yapılan test ve egzersizlerin uzaktan ve izlenebilir şekilde yapılması da mümkün olabilecektir. Bu çalışmalar sonucunda iki adet ön prototip ile bir adet gerçek ölçülerde; ortalama bir insan ağırlığında olan insansı robotun hareketlerini test edebilecek; prototip tasarım ve imalatı yapılmıştır. Proje kapsamında başlanan ve devam eden süreçte yazılımın geliştirilmesi, üretilen prototipten elde edilen verilerin işlenmesi ve kullanım kolaylığı sağlayan ara yüzün geliştirilmesini içeren çalışmalar tamamlanmıştır. Mekanik, elektronik ve bulut veri yazılım sistemlerinin kombinasyonu ile meydana getirilen platform sisteminin tamamı için uygun iş paketi doğrultusunda patent başvurusunda bulunulmuştur.

Anahtar Kelimeler: Hareket platformu, insan dengesi, iki serbest dereceli manipülatör

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

Nowadays, studies on rehabilitation of human balance measurement and computer-based force platforms increase rapidly. (Haksever et al., 2017; Soyuer et al., 2001; Alveroğlu et al., 2015).

Studies in the field of health, human and balance are not yet sufficient to show the natural balance reactions of human beings (Horst R. et al.,1999). In recent years, not only balance factor but also daily routine simulations are draw attention (Prosperin, 2013).

It is common that force platforms which are useful as a rehabilitation equipmentare useful to show especially CoG (Center of Gravity) and CoP (Center of Pressure) parameters. In most of the previous studies, these parameters were used. (Duarte , 2010; Piirtola, 2006). These data obtained with the sensors which are located on the corners of the force platforms. Devices that used in clinics, with general features, collect data with this way and identifying balance loses. (Piirtola et al., 2006). Fundamentally, computer-based platforms using physically similar methods, but they may contain software-based differences, which cause that users of the system may require to be specially trained. (Duarte et al., 2010).

Improvements related to force platforms continue on reducing costs and taking up less space. In a study by Clark and et al., it is stated that balance evaluations with the clinic “Nintendo Wii Balance Board” can be used as a valid assessment method, and it can also be a bridge between clinical tests and laboratory studies (Clark et al., 2014; Monteiro-Junior et al., 2015). Indexes are prepared for new improved devices but it is notified that must be improved with clinic adaptations (Ambrozy, 2011). In a study which on tracing the real time balance parameters during the evaluation with

new softwares should be retrofitted signalization (Gopalai et al., 2011). Although significant step for the improvement of this platforms which contained perturbations; some publications indicated that force platforms cannot be used for alone for the balance evaluations and rehabilitations (Kingma et al., 2011). It is declared that in an article published in 2012, the improvements are going on about the platform that make both balance evaluation and perturbation training. However, that developed system require large area and unportable (Walsh et al., 2011). In another publication that criticizes the platforms about cost and unportable, it is stated that a system is based on pressure analyses has been made and it has success on the validity tests. The study was performed with a very small number of participants and it was stated that there should be more participants (Walsh et al., 2011). A low cost and portable system, which is ideal for measuring the dynamic balance of the system, has been developed by Zhu in 2017. Therefore, it is far from evaluating the responses of the biological system to perturbation induced stimuli inherent in the dynamic balance (Zhu, 2017).

The purpose of the balance platforms used in the medical field is to observe how and how long the patients with balance problems completed the motion scripts given by the doctors; or how long they can maintain their balance position. For this purpose, it was decided to use parallel manipulators in order to produce a moving and portable force platform which can work with high precision and high speed under the heavy loads. Parallel manipulators are robots in which the actuator rods are connected in parallel to the fixed lower platform and the mobile upper platform.

Parallel manipulators, by means of this connection structure allows to work with higher precision and higher speed under heavy loads compared to serial manipulators. That is why, it is widely used in the industrial field, used in simulation systems, medical fields and automation mechanisms. The basis of parallel manipulators is based on the theoretical paper published by Maxwell in 1890 (Clerc et al., 2002). The first of the examples of spatial parallel mechanism used today is the

mechanism in Figure 1.1 that James E. Gwinnet made for the entertainment industry in 1928. (Zalbaza et al., 2002)

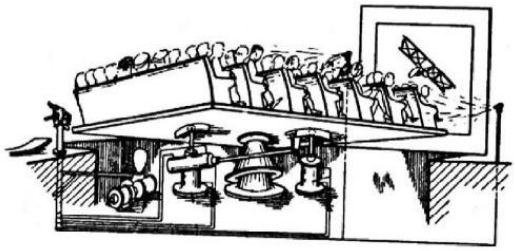


Figure 1.1 Mechanism for entertainment industry (Pollard, 1942)

In the following years Eric G. and Whiteall invented a parallel manipulator in 1962 for tire testing (Figure 1.2) (Gough, 1962). With this parallel manipulator has accelerated in academic and industrial studies about, the use of parallel actuators in industry.

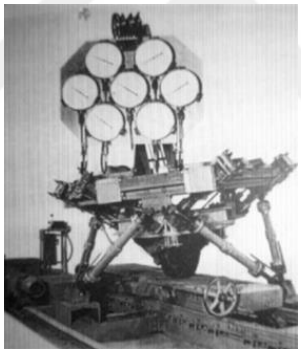


Figure 1.2 Manipulator for elasticity test that have invented Eric et al, 1962)

The one of the most common of these type mechanisms, known as the Stewart platform in the literature developed by Stewart for flight simulations in 1965, is a mechanism with 6 control degrees, each with 6 motors connected to the ground (Anlı et al., 2005).

Except for the parallel manipulator that mentioned below, Hunt has been introduced the idea of parallel manipulator with rotary engine in 1983 (Hunt , 1983). Studies that on the resolution and design of the kinematic equation of parallel

manipulators have continued since the 1900s up to now. There are many studies on this subject. Chen Zhang and Liyan Zhang have studied kinematics analysis for vehicle simulation mechanisms (Zhang et al., 2013). Liu, et al. have also investigated the optimal design of the 2-DOF PRRRP parallel mechanisms (Li et al., 2006). Ng, et al. studied the design and kinematic design of 3-DOF modular micro-parallel manipulators. Consequence of this study classical Stewart platforms caused more power consumption and more cost compared to 3-motor designs (Ng et al., 2006). A study by Arockia Selvakumar A. and Arul Kumar on the position analysis of M 3-DOF parallel manipulators have been carried out (Selvakumar et al., 2014).

Marjan Sikandar proposed a design for industrial applications, during examined the kinematics and algorithm of motion of 3-DOF parallel mechanisms (Sikandar et al., 2015). Fang at al. examined the structural conditions for the finite movement of parallel manipulators (Fang et al., 2004). Aşkın Dilan et al. have carried out study the MATLAB SimMechanics and Simulink to find the power consumption required for various configurations of 3-DOF manipulators (Aşkın Dilan et al., 2011). Aminzadeh et al. made 3-DOF dynamic analysis of parallel platforms (Aminzadeh et al., 2009). Hua et al. have examined the security positions for the Junkao Stewart Platform (Hua et al., 2007). Dongsu et al. worked on adaptive control for a 6-axis parallel simulated flight simulator (Dongsu, 2007). Anlı et al. published an article about summary of the studies on parallel mechanisms until 2005, sharing the general characteristics of the parallel mechanisms, kinetic, dynamic and working spaces and made recommendations for future studies. (Anlı et al., 2005)

In this study, the kinematic solution and prototype production of parallel manipulator was made with adding of fixed support to the center of 2-DOF with 3 motors with commensally 2-DOF, 3-DOF and 6-DOF parallel mechanism mentioned above. The actuator performed a desired movement that low power consumption and heavy loads with high velocity, accompanied by fixed support added to the fixed support on the center restricted the motion.

This thesis consists of seven chapters. In the first chapter, the aim of the study and the previous studies have been mentioned. In the second chapter, the designed system has been introduced and information about the kinematic solution has been given. In the third chapter, detailed information about the system components have been given. In the fourth chapter, mechanical design of prototype have been explained and the movement in the simulation environment has been compared with the real movement. In the fifth chapter, information which about kinetic and structural analysis of the prototype has been given. In the sixth chapter, information about mobile software being developed for prototype control has been given. The information and results obtained from the are briefly explained and it has been mentioned about studies that can be conducted in the future in the last chapter.

CHAPTER TWO
SYSTEM DEFINING AND KINEMATICAL SOLUTION

2.1 System Definition

The moving platform, which is composed of 2 surfaces can be seen in the Figure 2.1. One of these surfaces is mobile and the other are fixed. The movement threshold on the axis of mobile surface determined freedom degree of system. The mobile platform of the platform that we produced, can stagger movement which is doing rotational movement around the X axis, and trundle movement which is doing rotational movement around the Y axis.

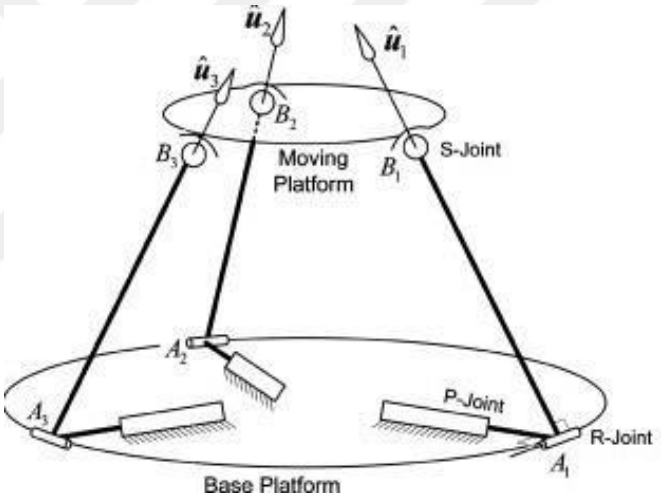


Figure 2.1 Moving platform surfaces (Axis)

Kinematic is to examine platform movements with the cause force, without torque (Unsal, 2007). When the kinematic solutions of parallel mechanism examine, is used a base that two main problems. The first one of these is a kinematic problem. When the length of legs is given, it is calculated the translation and orientation of mobile upper platform and fixed lower platform (Sunguray et al., 2014). The second one is inverse kinematic problem. It is calculated the position of orientation of the orientation. In the serial mechanisms, forward kinematic problem is simple, inverse

kinematic problem is complicated. The point in question is total opposite. Forward kinematic problem is complicated, inverse one is easy (Güney, 2010).

The kinematical problem of the platform is to determine the distance between mobile upper platform's joints and fixed lower platform's joints. In the studies about the kinematical problems, the solution of inverse kinematic problems and kinematics are properly with mechanic features of produced prototypes. For the solution of inverse kinematic problem, it must be determined the coordinates of joint points which is located between mobile upper platform and fixed lower platform. (Figure 2.2)

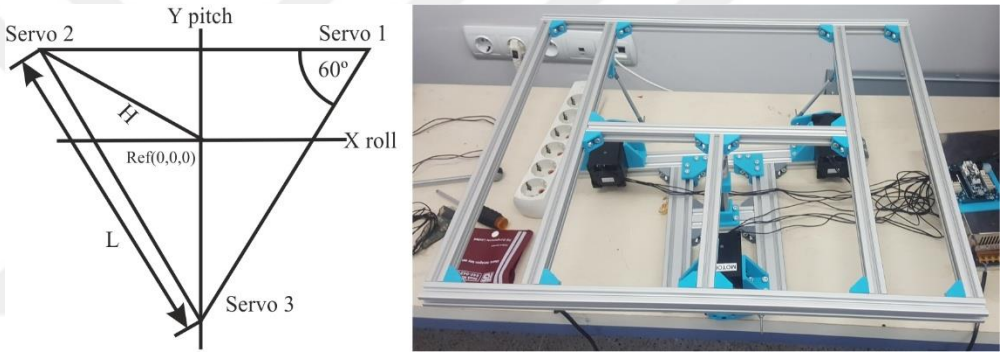


Figure 2.2 Distance of ports (Personal archive, 2018)

The First Coordinate Matrix of the Joints can be written using formula (2.1) and (2.2).

$$P_1 = (X_1, Y_1, Z_1)\ \ P_2 = (X_2, Y_2, Z_2)\ \ P_3 = (X_3, Y_3, Z_3) \tag{2.1}$$

$$P_{123} = \begin{bmatrix} X_1 & X_2 & X_3 \\ Y_1 & Y_2 & Y_3 \\ Z_1 & Z_2 & Z_3 \end{bmatrix} \tag{2.2}$$

2.2 Transforming of Matrix

A 3x3 transformation matrix can be written for the rotational movement of the moving platform on any axis. The transformation matrices created for the rotational motion in two different axes are multiplied and a global transformation matrix is obtained for all possible movements in the movement space (Ulaş, 2009). If the transformation matrix multiple (Rx*Ry*Rz) respectively, the global transformation matrix is obtained. The resultant global matrix makes relativistic transformation and according to this, any axis reaches the new tending after one axis around coordinate system. Next movement is happened in around the axis which is related coordinate system that has new tending (McKerrow, 1991).

2.2.1 Roll Movement

The Stagger motion is the movement around the X-Axis and can be illustrated as in Figure 2.3.

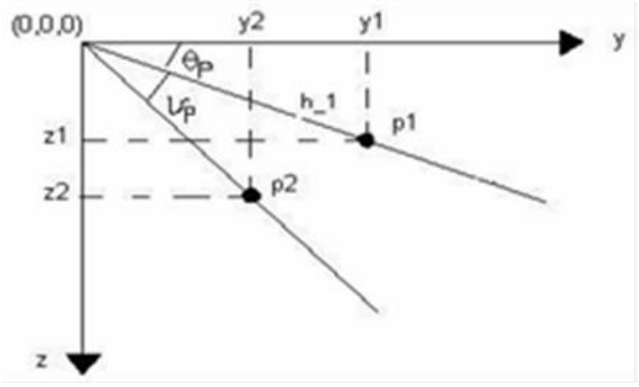


Figure 2.3 Rotation Around the X-Axis

Transformed matrix for trundle movement:

$$Rx = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\varphi & -sin\varphi \\ 0 & sin\varphi & cos\varphi \end{bmatrix} \quad (2.3)$$

2.2.2 Pitching Movement

Pitching movement is the name given to the movement around the y-axis. This movement is similar to the stagger motion.

Matrix transformation for the pitching movement:

$$Ry = \begin{bmatrix} cos\theta & 0 & sin\theta \\ 0 & 1 & 0 \\ -sin\theta & 0 & cos\theta \end{bmatrix} \quad (2.4)$$

2.2.3 Yaw (Aberration) Movement

Stagger movement is the name given to the movement around the y- axis. This movement is similar to the pitching movement.

$$Rz = \begin{bmatrix} cos\psi & -sin\psi & 0 \\ sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

2.2.4 Rotational Matrix

If the rotation equations found to find the new position coordinates of the platform are multiplied in order (matrix 2.3, 2.4, 2.5) and the rotation matrix (2.6) is found.

$$Rb = \begin{bmatrix} cos\psi cos\theta & -sin\psi cos\varphi + cos\psi sin\theta sin\varphi & sin\psi sin\varphi + cos\psi sin\theta cos\varphi \\ sin\psi cos\theta & cos\psi cos\varphi + sin\psi sin\theta sin\varphi & -cos\psi sin\varphi + sin\psi sin\theta cos\varphi \\ -sin\theta & cos\theta sin\varphi & cos\theta cos\varphi \end{bmatrix} \quad (2.6)$$

Actuator rod joints, which are located between fixed lower platform and movable upper platform, ought to state with coordinate system to find actuator vectors. The location of actuator rod vectors is shown in Figure 2.4.

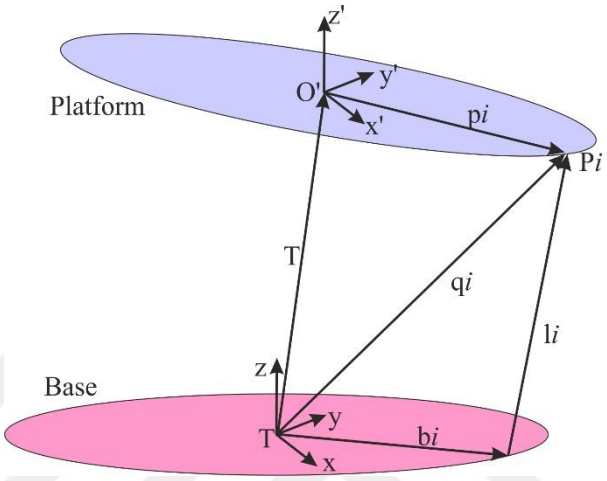


Figure 2.4 Actuator rod vectors

If the actuator rod vector is calculated in Figure 2.4;

$$\overrightarrow{q_\iota} = \vec{T} + R_B * \overrightarrow{p_\iota} \quad (2.7)$$

$$\overrightarrow{l_\iota} = \vec{T} + R_B * \overrightarrow{p_\iota} + \overrightarrow{b_\iota} \quad (2.8)$$

2.2 Calculation of Engine Angles

In this project, the I*i* vector is connected with actuator rods angle because of the using of servo motors. The calculation of the angle between fixed lower platform and smart servo motors, is given detailed below.

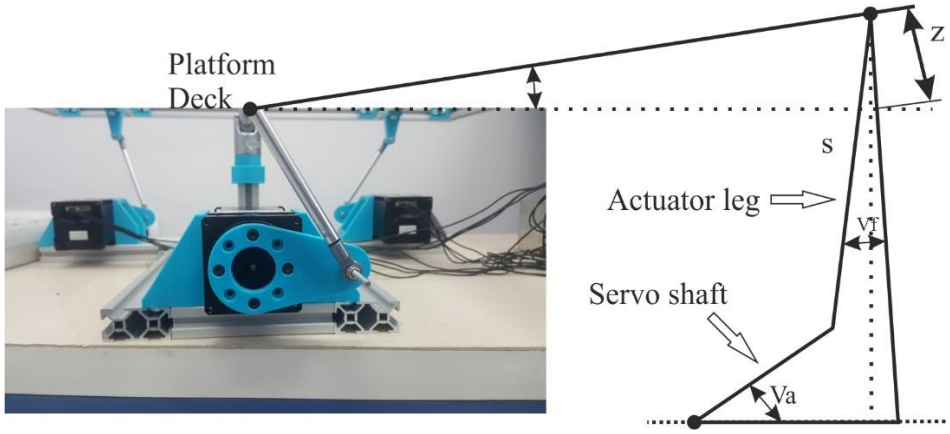


Figure 2.5 Shaft angle (Personal archive, 2018)

- r smart servo actuator arm length
- s length of arm that connected to moving platform
- A*i* connection point that between actuator arm and the fixed platform
- α the angle between x-y plane and servo rod
- β the angle between x-axis and servo rod

According to the Figure 2.5;

$$x_a = r * cos\alpha * \cos\beta + x_b \quad (2.9)$$

$$y_a = r * sin\alpha * \sin\beta + x_b \quad (2.10)$$

$$z_a = r * sin\alpha + z_b \quad (2.11)$$

$$\vec{r} = \vec{A} - \vec{B} \quad (2.12)$$

$$\vec{l} = \overrightarrow{q_i} - \vec{B} \quad (2.13)$$

$$\vec{s} = \overrightarrow{q_i} - \vec{A} \quad (2.14)$$

According to the Pythagoras' theorem;

$$l^2 = s^2 + r^2 \quad (2.15)$$

$$r^2 = (x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2 \quad (2.16)$$

$$= (x_a^2 + y_a^2 + z_a^2) + (x_b^2 + y_b^2 + z_b^2) - 2(x_a x_b + y_a y_b + z_a z_b) \quad (2.17)$$

$$l^2 = \left(x_q - x_b\right)^2 + \left(y_q - y_b\right)^2 + \left(z_q - z_b\right)^2 \quad (2.18)$$

$$=\left(x_q^2+y_q^2+z_q^2\right)+\left(x_b^2+y_b^2+z_b^2\right)-2\left(x_qx_b+y_qy_b+z_qz_b\right) \quad (2.19)$$

$$s^2=\left(x_q-x_b\right)^2+\left(y_q-y_b\right)^2+\left(z_q-z_b\right)^2 \quad (2.20)$$

$$=\left(x_q^2+y_q^2+z_q^2\right)+\left(x_b^2+y_b^2+z_b^2\right)-2\left(x_qx_b+y_qy_b+z_qz_b\right) \quad (2.21)$$

$s^2=r^2+l^2$ substituting into the (2.17), (2.19) and (2.21) equations;

$$s^2=l^2+\left(x_b^2+y_b^2+z_b^2\right)+2\left(x_qx_b+y_qy_b+z_qz_b\right)+r^2-\left(x_b^2+y_b^2+z_b^2\right)+$$
$$2(x_ax_b+y_ay_b+z_az_b)-2\left(x_qx_a+y_qy_a+z_qz_a\right) \quad (2.22)$$

Which reorganize to (2.22),

$$l^2-(s^2-r^2)=2\left(x_b^2+y_b^2+z_b^2\right)+2x_a\left(x_p-x_b\right)+2y_a\left(y_p-y_b\right)\ \ 2z_a\left(z_p-z_b\right)-$$
$$2\left(x_qx_b+y_qy_b+z_qz_b\right) \quad (2.23)$$

x_a, y_a, z_a values are written in equation 2.23 in the form of equations 9,10 and 11;

$$l^2-s^2+r^2=2\left(x_b^2+y_b^2+z_b^2\right)+2r\left(x_q-x_b\right)cos\alpha cos\beta+2x_qx_b-2x_b^2+$$
$$2r\left(y_q-y_b\right)cos\alpha sin\beta+2y_qy_b-2y_b^2+2rsin\alpha\left(z_q-z_b\right)+2z_qz_b-2z_b^2-$$
$$\left(x_qx_b+y_qy_b+z_qz_b\right) \quad (2.24)$$

Which reorganize to (2.22),

$$l^2-s^2+r^2=2r\left(z_q-z_b\right)\sin\alpha+2\mathrm{r}\left(x_q-x_b\right)\cos\alpha\cos\beta+$$
$$2\mathrm{r}\left(y_q\ y_b\right)cos\alpha sin\beta \quad (2.25)$$

$$\underbrace{l^2-s^2+r^2}_{L}=\underbrace{2r\left(z_q-z_b\right)}_{M}sin\alpha+\underbrace{2r\left[\left(x_q-x_b\right)cos\beta+\left(y_q-y_b\right)sin\beta\right]}_{N}\cos\alpha \quad (2.26)$$

New form of the equation is;

$$L=Msin\alpha+Ncos\alpha \quad (2.27)$$

Equation (2.27) can be written trigonometrically like (2.28), for to find how many degrees (α) engines should turn.

$$\alpha = \sin^{-1}\frac{L}{\sqrt{M^2+N^2}} - \tan^{-1}\frac{N}{M} \tag{2.28}$$

$$L = l^2 - (s^2 - a^2)$$

$$M = 2a(z_p - z_b)$$

$$N = 2a[cos\beta(x_p - x_b) + sin\beta(y_p - y_b)$$

It is possible to calculate the rotation angles of each smart servo motor with the inverse kinematic equation obtained above. In the physical and simulation tests performed with the inverse kinematic equation found, it was observed that the moving platform completed the movement with an error of less than 1%.

CHAPTER THREE
SYSTEM IMPLEMENTATION

3.1 System Introduction

After completion of the literature survey and kinematic analysis, two pre-prototypes were produced according to the project and the prototype planned to be produced.(Figure 3.1 a, Figure 3.1 b)

Figure 3.1 **a** Primary prototype 1 **b**: Primary prototype 2 (Personal archive, 2018)

Figure 3.2 Ultimate prototype (Personal archive, 2018)

In order to test kinematic equations, first prototype which has system design and given in Figure 3.2, is designed and produced. In mechanical design, it was used 3mm sheet metal, for the movement of platform it is used high torque power, position, flow, Ex-106+ smart servos that can be read voltage information, and USB2Dynamixel motor driver is used. The control of the motors is provided by the

embedded micro PC with Linux operating system. The MPU6050 3-axis tilt sensor was used to assess the movement accuracy of the platform (Figure 3.3).

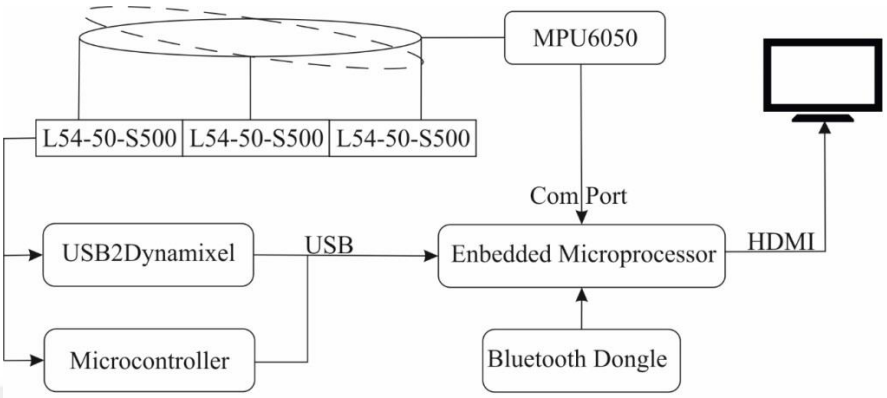


Figure 3.3 System design

As a result of the studies; prototypes are in order to be portable, the data that is reachable from elsewhere and reduce prime costs, system design is updated as Figure 3.4)

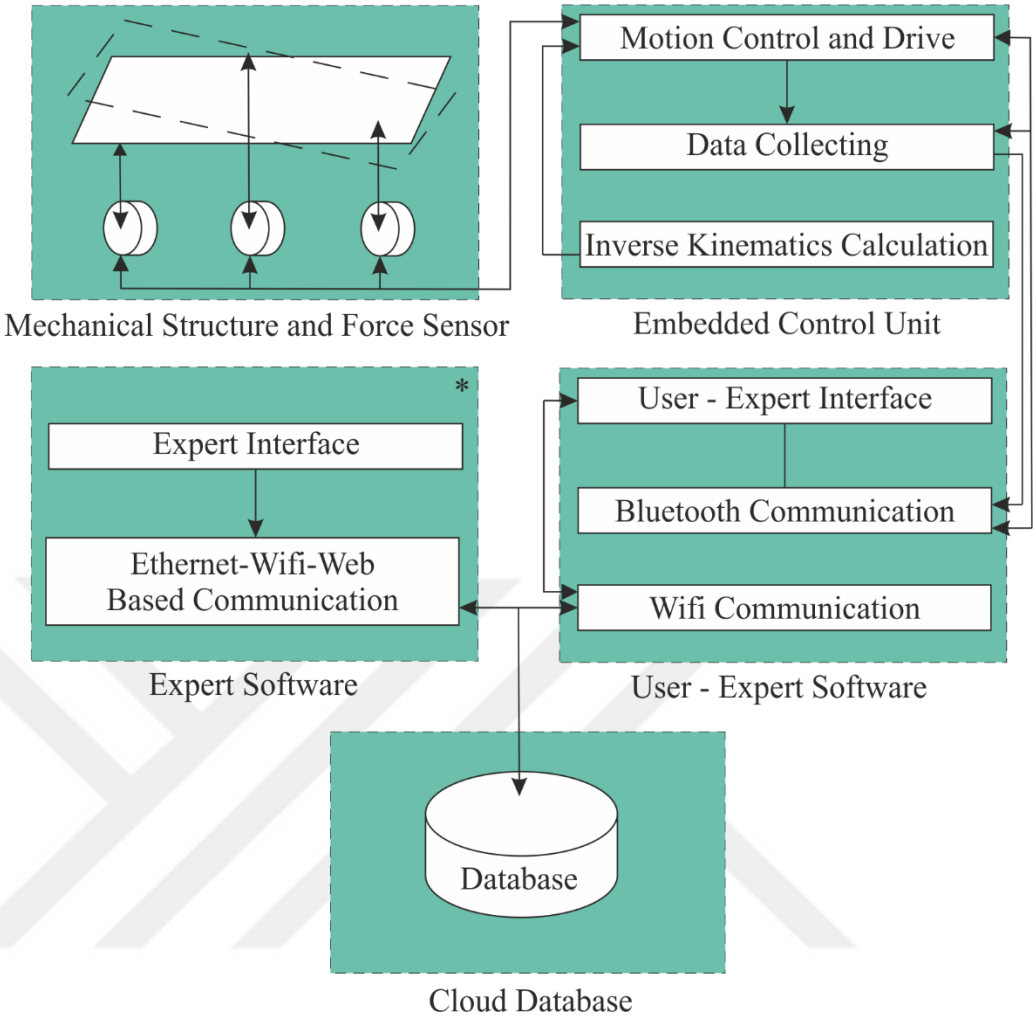


Figure 3.4 Updated system design (*This software prepared with professional help)

After the updated system design, L54-50-S500 smart servos, which are planned to be used in the final prototype instead of Ex-106+ motors, are used in the second prototype produced in mechanical structure with 20x20 and 20x40 sigma profiles.

Detailed information which used on the equipment in system design and prototypes is listed below.

3.2 Control Card

The control cards are cards which are produced to assignment that do predefined duties. They have reduced the product size and cost pretty much in the system with a certain purpose.

3.2.1 OpenCM9.04

The OpenCM9.04 (Figure 3.5) used in this study is a control card used to connect and control ARM Cortex-M3 32bit CPU based Dynamixel engines. The OpenCM9.04 platforms are open source. ROBOTIS OpenCM, with its Processing and Wiring languages, allows for easy programming in C/C++ languages such as the Arduino IDE. ROBOTIS OpenCM; Supported by Windows, Linux and MacOS operating systems.

Figure 3.5 Open CM9.04 example (Robotis, 2018)

3.3 UDOO

Another control card that used during the prototype developing is UDOO Quard (Figure 3.6). And it is compatible with open source software Linux and Arduino and has 1GHz ARM Cortex-A9 and ARM Cortex-M3 processor, used for engine control with the support of 1 GB DDR3 RAM, OpenGL, getting the feedback of platform angle and angle sensor, is used for development of interface.

Figure 3.6 UDOO Quard(Udoo, 2018)

3.4 USB2Dynamixel

USB2Dynamixel is used for basic operations such as calibration of smart servos and communication with embedded operating system, baud rate control and settings. USB2Dynamixel is used in the first prototype for tracing smart servo motors via embedded operating system with C code without having to another software.

3.5 Drivers and Engines

The reason for preference of Ex-106+ and L54-50-S500 smart servo which is used in project, can be read instantly velocity, position, temperature, voltage and current value, by means of OpenCm 485 expansion card which is specially produced for this engines.

3.6 Open CM 485 Expansion Board

The OpenCm 485 expansion card was used with OpenCm 9.04 (Figure 3.7). The OpenCm 485 expansion card has 5piece 3-pin TTL and 5piece 4-pin RS485 connectors compatible with Dynamixel motors. By means of this card, which supports 5 volts up to 30 volts, has been enough to connect the 24volt power supply to this card.

Figure 3.7 Open CM 485 expansion board (Robotis, 2018)

3.7 EX 106+ and L54-50-S500

In order to supply the system movement during the test of kinematic solving of first prototype, is used Dynamixel Ex-160+ engine. (Figure 3.8a) This engine has 8Nm torque power at the same time with constant working and its adequate during the kinematic tests. New prototype which has Dynamixel Pro L54-50-S500 smart servo (Figure 3.8b) is designed for the tests of average human weight humanoid robot that aim of this project. 3 piece L54-50-S500 smart servo which is used in mechanical design, can carry the average human weight without using full force.

Figure 3.8 a:Dynamixel Ex-160+ Motor b: Dynamixel Pro L54-50-S500 (Robotis, 2018)

Dynamixel engines that we use in mechanical designs is enabled to restriction of position, torque, velocity, flow and tracing the load changes, with the microprocessor on it. Thanks to these restrictions, software or mechanic problems is avoided.

3.8 Sensors

Sensors are often used in robot and machine technologies. Positioning and movement sensors in the machine control the answers to code and the motion of machine. (MEB, 2012)

3.8.1 Angle Sensor

The MPU6050 is used as an angle sensor. MPU6050 are used sensors in aircraft, balance robots or many robot projects in common (Figure 3.9). It is an IMU sensor board with 3-axis gyro and 3-axis angular accelerometer. The voltage regulator on it, is supplied with 5v voltage via USB.

In the project, angle values come from MPU6050, with current software, when position of motors simultaneity move, or ensure to trace of surface angle at that

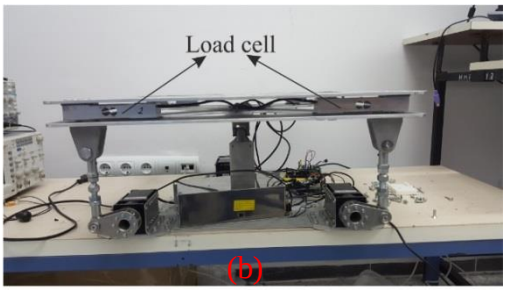
moment. MPU6050 is provided great convenience during the prototyping with the system which is cheap and embedded operating system via USB or because of the contacting via series channel.

Figure 3.9 Angle sensor (Robotistan, 2018)

3.8.2 Pressure Analysis and Center of Gravity Sensor

In order to track the balance function analysis and user's center of gravity, 4 load cells are placed in 4 corners of prototype (Figure 3.10). The task of the load cell is to convert the force into electrical signals. Data form load cells which have the capacity to carry 120 kg, process in MATLAB numerical calculation, then analyzed user's movements of balance functions and center of gravity.

(a)



(b)

Figure 3.10 Left: (a) Load cell, Right: (b) The using of load cell (Personal archive, 2018)

CHAPTER FOUR
MECHANICAL DESIGN AND PRELIMINARY PROTOTYPING

The manipulator could have carry with heavy loads in mechanic design and constitution of the lightest structure which can be held possible loads during the swinging movement. In order to achieve this, it is decided that the structure body is made of aluminum and the is made of steel. The forces to be exposed the manipulator tested in simulation and it is verified that this design can be hold on the forces.

4.1 Mechanic Design and Primary Prototype Implementation

The details about the aims and purposes of design have been given in Table 1

Table 4.1 The aim of the mechanism design

The Aim of the Design	Purpose
(1) Using no more than 3 motors	Providing security and lowering costs
(2) Load sharing of motors	Prevent motors from overloading
(3) Increase load/structural weight ratio	To obtain maximum performance from current power
(4) Using embedded system	Embedded system to provide remote control without a display or control panel
(5) Portable design	Use in multiple spaces
(6)Observing of data usage	Storage of data at the end of use and comparison with the previous ones.

In line with this purpose, 3D modeling was performed using SolidWorks for the primary prototype (Figure 4.1a) and the ultimate prototype (Figure 4.1b).

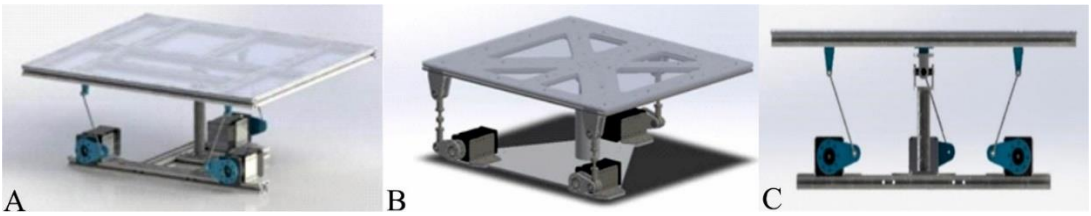


Figure 4.1 (a) pre-prototype, (b) ultimate prototype, (c) test prototype front view

The stagger and trundle of platform, is provided by 3 smart servos where have located under the platform like isosceles triangle. These servo-motors are limited to move between the limit values to prevent the user from falling over the platform except for trundle and stagger. These motors are connected to the movable platform by metal legs. Therefore, the angular motion is transferred linearly. In the design, it is supplied to effect seesaw movement linearly due to the engine motion center and rods are located linearly. The cardan shaft and the support post were placed between the midpoint of the mobile platform and the fixed lower platform to reduce the load on the engines, thus preventing power overconsumption and overloading of the motors (Figure 4.2).

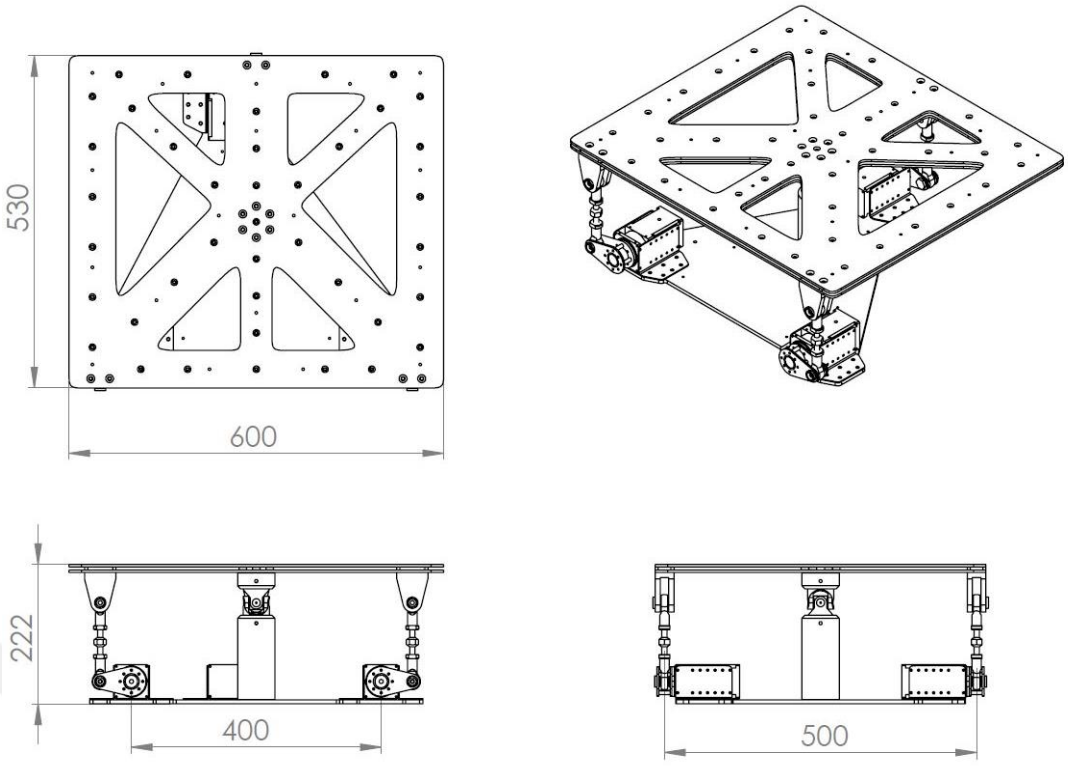


Figure 4.2 3D image of moving platform having dimensions

4.2 Movable upper platform

Movable upper platform is made of aluminum material and the dimensions of platform 530mm and 600mm size and 4mm thick. The connection points on the platform, are composed for the surfacing and actuator is given in Figure 4.3.

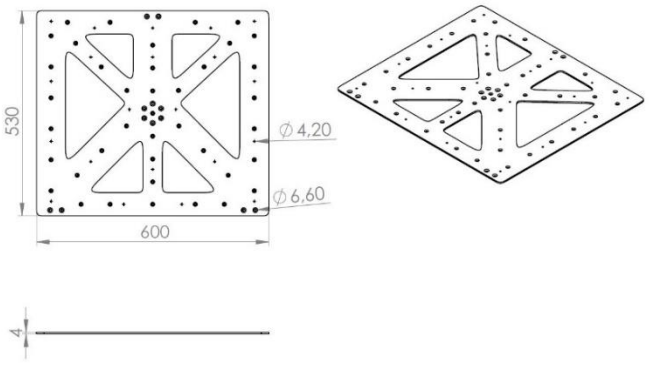


Figure 4.3 3D dimensions of the upper platform

4.3 The Connector of moving upper platform

This piece is made of aluminum material, designed to connect with drive arm and the moving upper platform. The dimensions are given in Figure 4.4. The connecting part of Figure 4.3 is used in 3 drive arms.

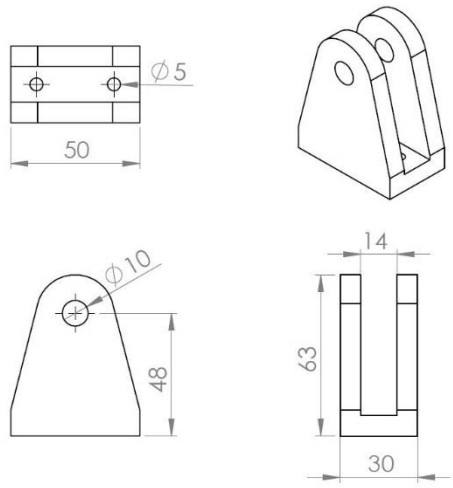


Figure 4.4 3D dimensions of moveable upper platform connection piece

4.4 Engine connection rod

This part is designed to convey the motor motions to the upper platform and is made of 3mm steel. The dimensions of this part are given in Figure 4.5.

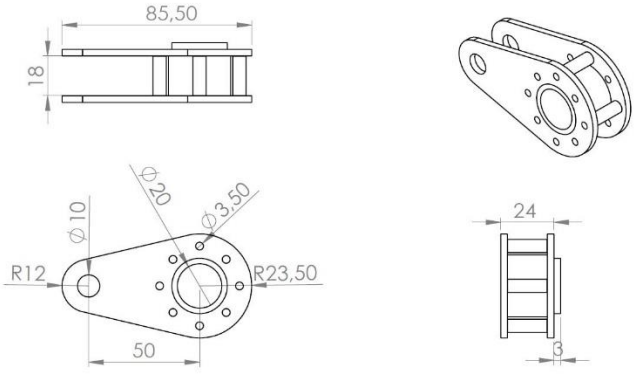


Figure 4.5 3D dimension of engine connection road

4.5 Fixed Leg and Movement Joint

The connector where is located between fixed lower platform and movable upper platform, consist 2 tower shaped aluminum block and a cardan shaft. The dimensions of this part are given in Figure 4.6.

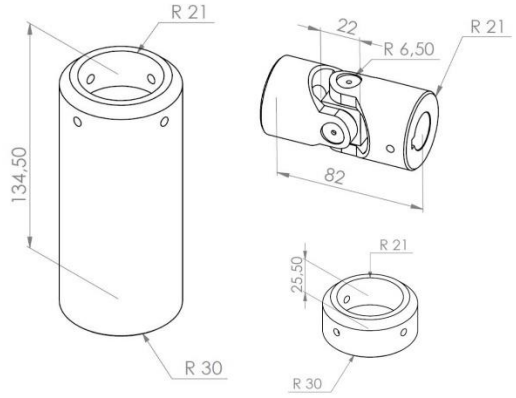


Figure 4.6 3D measurement of the fixed rod and movement joint

4.6 Fixed Lower Platform

This piece where is located in lower part of manipulator, is isosceles triangle shaped and 464mm and 534mm sized. All other parts of the manipulator are located on this platform. The dimensions of this piece with protrusions designed for engine mounting are given in Figure 4.7.

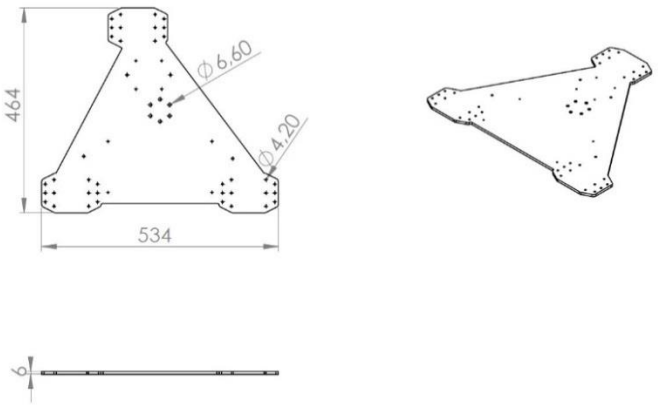


Figure 4.7 3D dimensions of fixed bottom platform

4.7 Prototype Production

In this study 3 prototypes were produced. In the pre-prototype, EX106 + motors and 3 mm sheet metal were used instead of L54-50-S500 engines. (Figure 4.8.a-b-c) Table 1 also contains the mechanical information about the first prototype.

Figure 4.8 a: Pre-prototype with small scale b: Pre-prototype with normal scale c: Ultimate prototype with normal scale (Personal archive, 2018)

It was tried different scripts on the scope of the balance platform working before pre-prototype. With the help of the SolidWorks program, the construction under 80 kg load is simulated in order to anticipate and shape the problems that may occur with the tried and tested scripts. At the end of the simulation, axial bending and displacement values were found to be below the safety values (Figure 4.9).

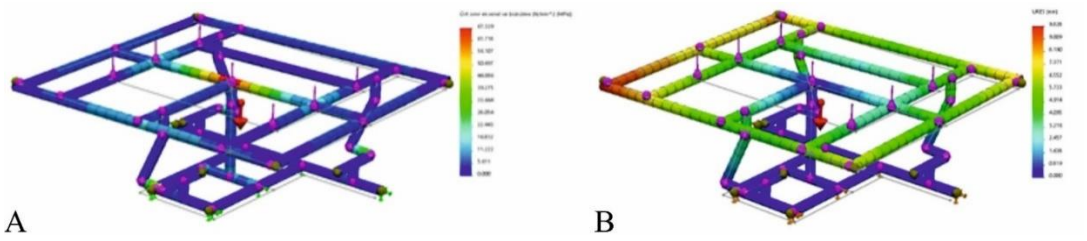


Figure 4.9 a: Axial twist b: Translocation values

In consequence of simulations and tests, it is decided to initiate ultimate prototype, in order to use robotic tests. After the kinematic and mechanical tests of the produced prototype were made, it was observed that there was a flexion of 19 mm at the corner points. After the necessary changes have been made in the selection of the material to minimize this extension in the main product, the final prototype has been produced (Figure 4.10).

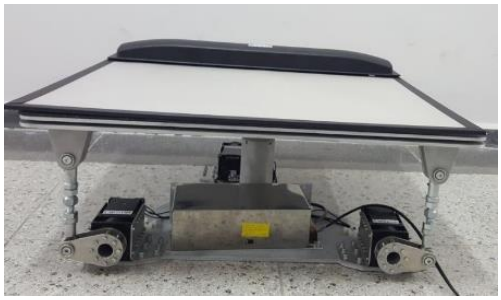


Figure 4.10 Ultimate prototype (Personal archive, 2018)

4.8 Comparison of Simulation and Real Motion

The using of modern simulation was used during development of nuclear weapon, at the end of the Second World War, by John con Neumann and Stanislaw Ulam. Thanks to the simulation technique, the solution of the experimental procedures which are very difficult and expensive to solve is provided successfully. With the developing computer technology, some experiments which is similar with laboratory experiments can be made in computer based. (Aydın, 2014)

Scope of work; simulation is made based on prototype design which is completed tests of weights. The comparison between simulation and real movement of 5degree stagger and trundle movements, it is observed that the highest level of deviation value is %5. (Figure 4.11a, Figure 4.11b, Figure 4.11c, Figure 4.11d).

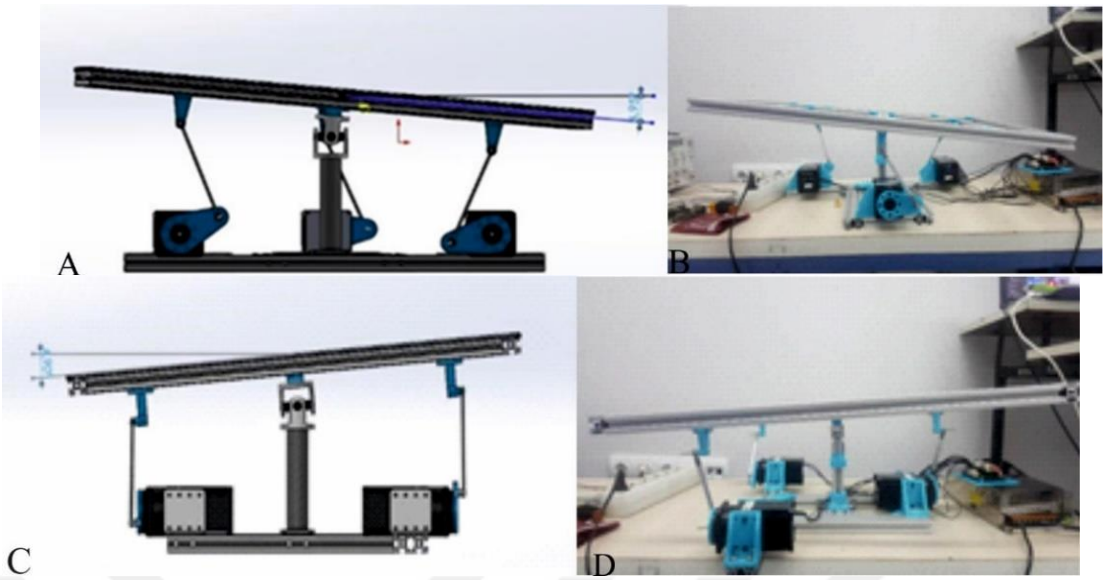


Figure 4.11 a-b-c-d Simulation of deviation value (Personal archive, 2018)

CHAPTER FIVE
KINETIC ANALYSES

5.1 Kinetical Analysis

Kinematic and kinetic analysis were performed to determine the motor power requirements of the portable balance platform.

For kinematic analysis, the balance platform is patterned taking into account the model front view and the one-way rotation movement of the cardan joint in the middle zone. Figure 5.1 shows the front view and basic dimensions of the balance platform, with a tetramand mechanism.

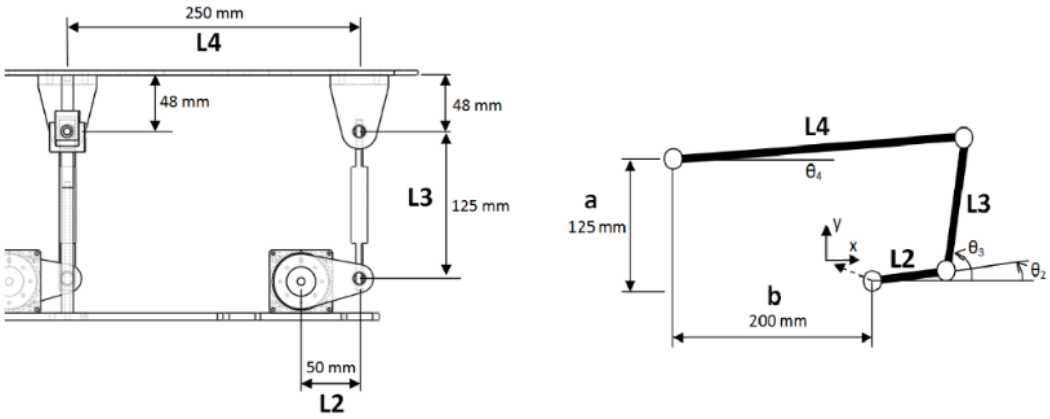


Figure 5.1 Kinematical analysis model of movable balance platform

The kinematic relations of the balance platform can be written as follows.

Location equations:

$$\begin{aligned} L_2 \cos\theta_2 + L_3 \cos\theta_3 - L_4 \cos\theta_4 &= -b \\ L_2 \sin\theta_2 + L_3 \sin\theta_3 - L_4 \sin\theta_4 &= a \end{aligned} \tag{5.1}$$

Rate equations:

$$\begin{aligned} -\omega_2 L_2 \sin\theta_2 - \omega_3 L_3 \sin\theta_3 + \omega_4 L_4 \sin\theta_4 &= 0 \\ \omega_2 L_2 \cos\theta_2 + \omega_3 L_3 \cos\theta_3 - \omega_4 L_4 \cos\theta_4 &= 0 \end{aligned} \tag{5.2}$$

Acceleration equations:

$$\begin{aligned} -\alpha_2 L_2 \sin\theta_2 - \omega_2^2 L_2 \cos\theta_2 - \alpha_3 L_3 \sin\theta_3 - \omega_3^2 L_3 \sin\theta_3 + \alpha_4 L_4 \sin\theta_4 + \omega_4^2 L_4 \cos\theta_4 &= 0 \\ \alpha_2 L_2 \cos\theta_2 - \omega_2^2 L_2 \sin\theta_2 + \alpha_3 L_3 \cos\theta_3 - \omega_3^2 L_3 \sin\theta_3 - \alpha_4 L_4 \cos\theta_4 + \omega_4^2 L_4 \sin\theta_4 &= 0 \end{aligned} \tag{5.3}$$

Newton-Raphson method was used for solution because of non-linearity of position equations. 2 number rod which is drive member, for known θ_2 angular displacement, θ_3 and θ_4 angular displacement can be determined through with iteration technique in below.

$$\begin{Bmatrix} \theta_3 \\ \theta_4 \end{Bmatrix}_{i+1} = \begin{Bmatrix} \theta_3 \\ \theta_4 \end{Bmatrix}_i - \begin{bmatrix} -L_3 \sin\theta_3 & L_4 \sin\theta_4 \\ L_3 \cos\theta_3 & -L_4 \cos\theta_4 \end{bmatrix}^{-1} \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} \tag{5.4}$$

Where f_1 and f_2 are location equations. In Figure 5.2, angular displacement ısal2 which is oscillating with angular frequency of 6.28 rad / s (1 Hz) and related angular velocity and acceleration graphs are given. For the balance table, ± 5 angular displacements are taken into account in accordance with the prototype.

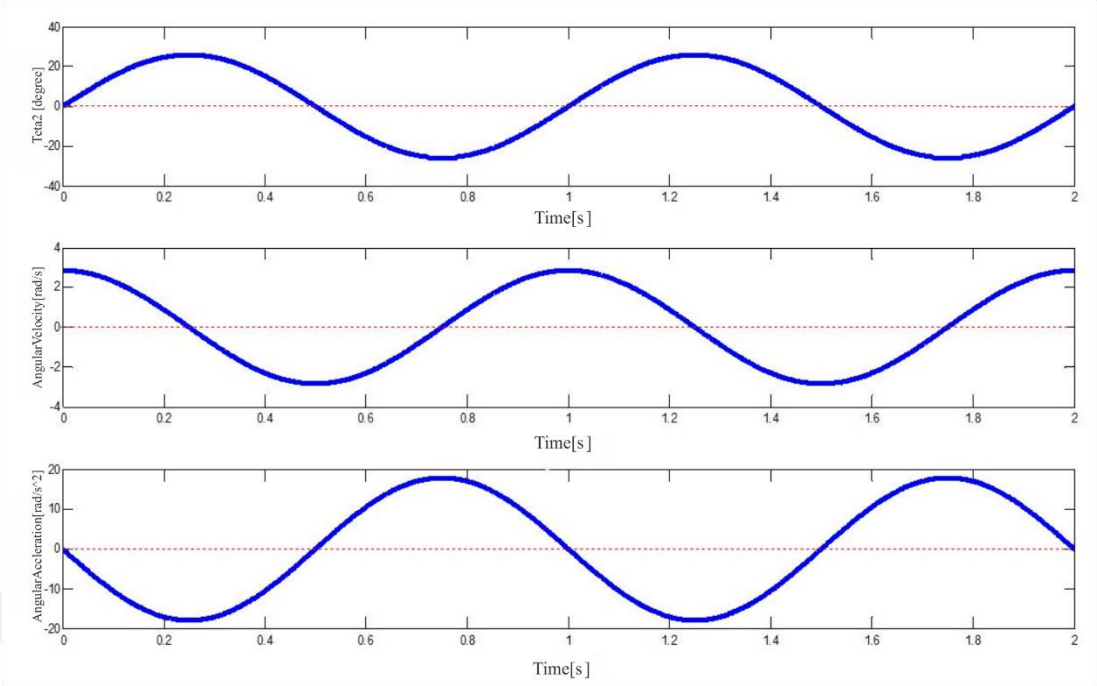


Figure 5.2 Angular position, speed and acceleration changes for L_2 length with input

As a result of kinematic analysis which for θ_2, ω_2 and α_2 values where have shown in Figure 5.2, the changes of θ_3 and θ_4 obtained for the balance table which is 4 number member is given Figure 5.3. Velocity and acceleration equations are linear and can be obtained ω_3, ω_4, α_3 and α_4 values with linear equations team solution.

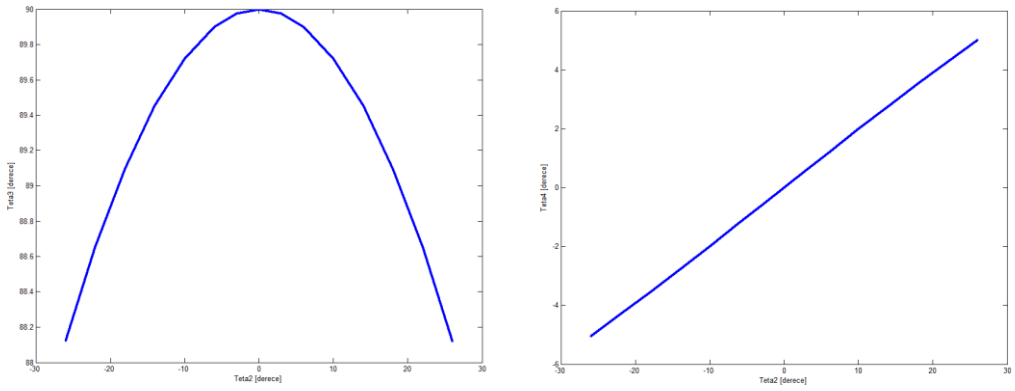


Figure 5.3 Conversion of θ_3 and θ_4 by θ_2

Angular velocity and acceleration values that used movement data belonged limb 2, which have obtained for limbs 3 and 4 are given in Figure 5.4.

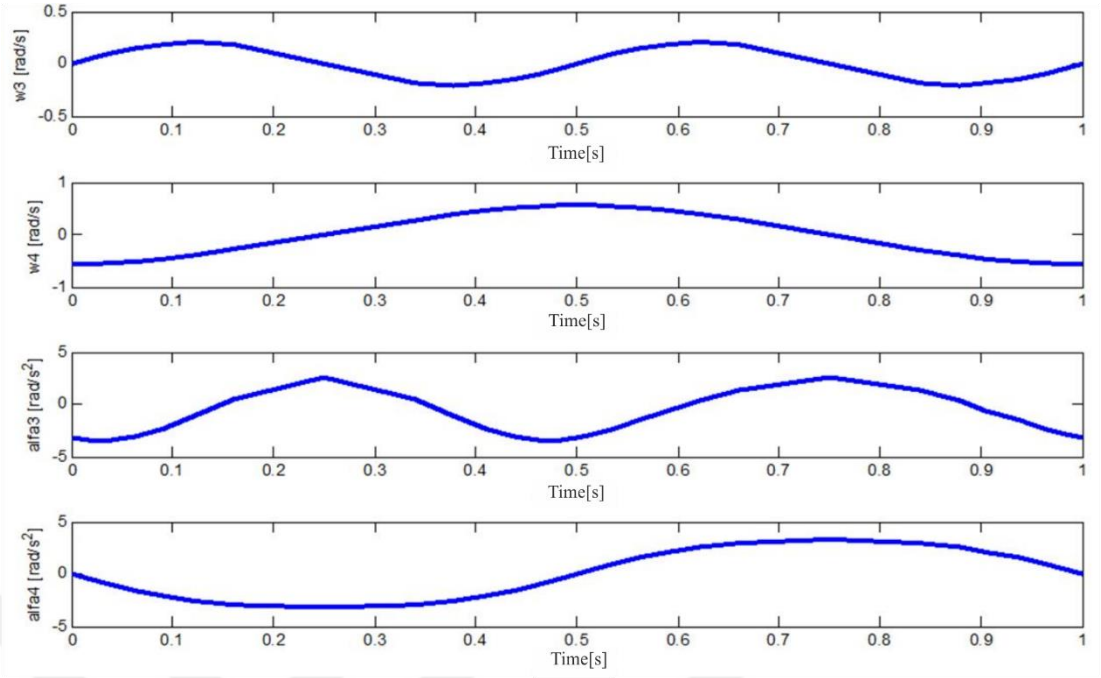


Figure 5.4 Angular velocity and angular acceleration values of limbs 3 and 4

The information are given that mass and mass moment of inertia of the limbs in the Figure 5.5 that consist movable balance platform. Classical machine dynamics methods can be used to calculate the motor power requirements needed to achieve the movement that movable table which is limb 4 in Figure 5.2.

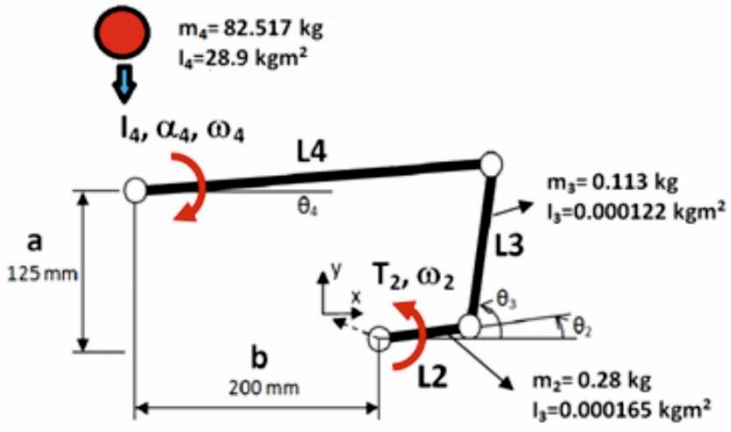


Figure 5.5 Simplified model of balance platform

However, considering mass, moments of inertia of mass, velocity and acceleration of the limbs which consist balance platform, in the balanced load limb 4 engaged to ground with its body and universal joint, using the information that the center of mass of the tables are in the joint area and the basic resistance will be formed due to the mass moment of inertia, can be used simplified power balance formula which is given below for calculation of motor power.

$$T_2\omega_2 + (I_4\alpha_4)\omega_4 = 0 \tag{5.5}$$

Motor power requirements that obtained with used this formula is given in Figure 5.6.

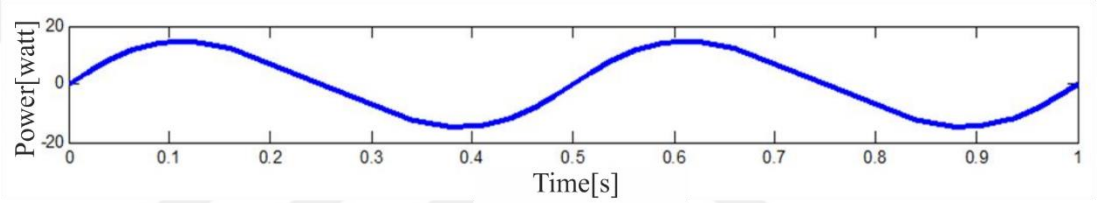


Figure 5.6 Motor power requirement for motion number 2 with speed and acceleration analysis results

According to the values that given in Figures 5.6, it is given in Figure 5.2 for θ_2, the power requirement for limit movement which observed in the movable balance platform design, obtained about at 20 Watt. The above mentioned analysis were also performed with SolidWorks Motion Simulation software and it is given that motor power requirements that belongs to limb 2 for the different situation, in between Figure 5.7- Figure 5.9. According to the results, it was determined that the motor power requirements for normal and abnormal load conditions were compatible with electric motor powers (50 Watt) used in this project.

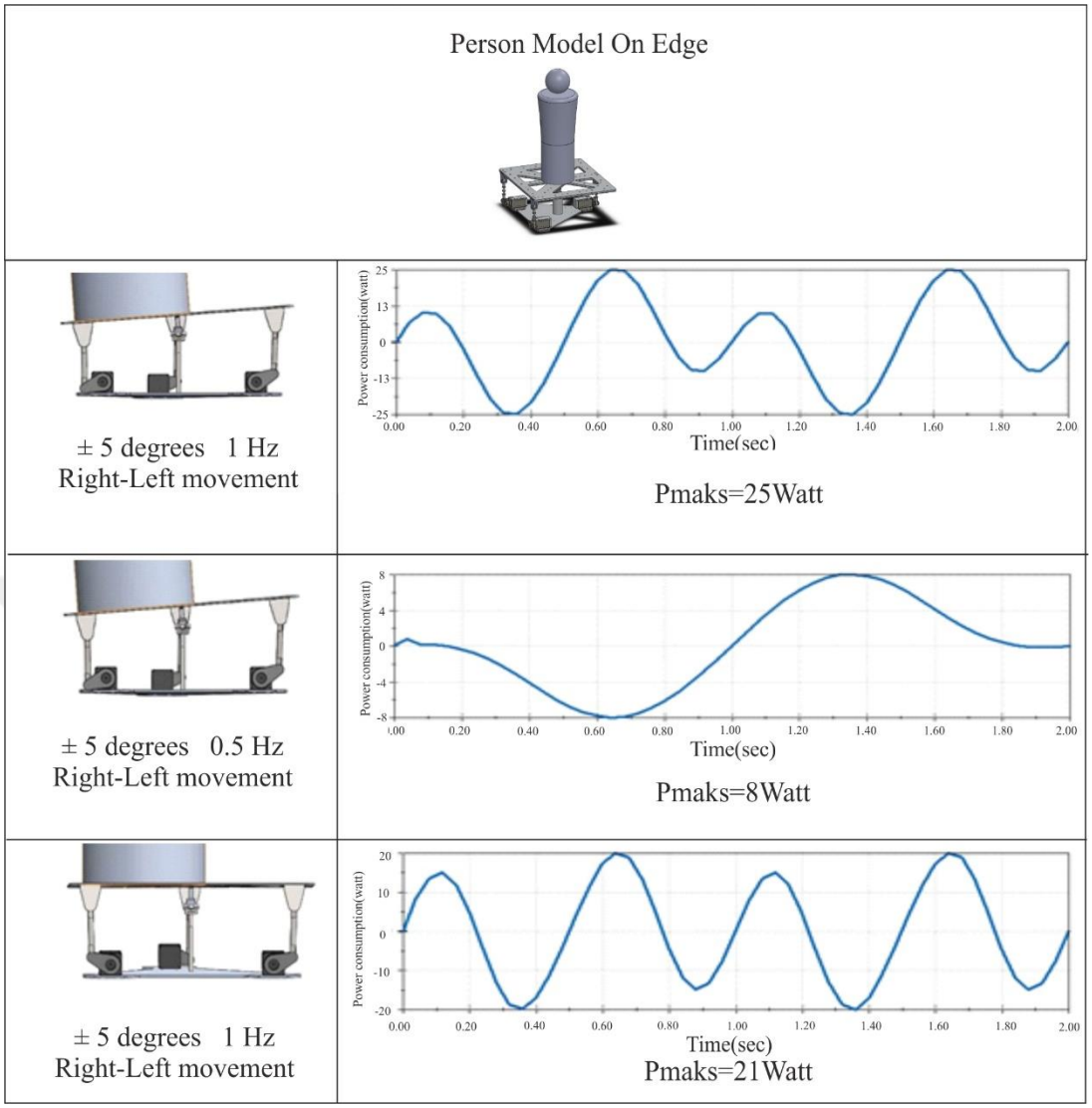


Figure 5.7 Power requirement for normal operating conditions

In the Figure 5.7, during the normal terms of use, the person in the middle of mechanism, the maximum power requirement for the right-left movement in 5 degree 1 Hz and 0.5 Hz is shown. Normal terms of use, it is calculated that maximum power requirement is 21 Watt and this power requirement considerably below the maximum power of seminal motors.

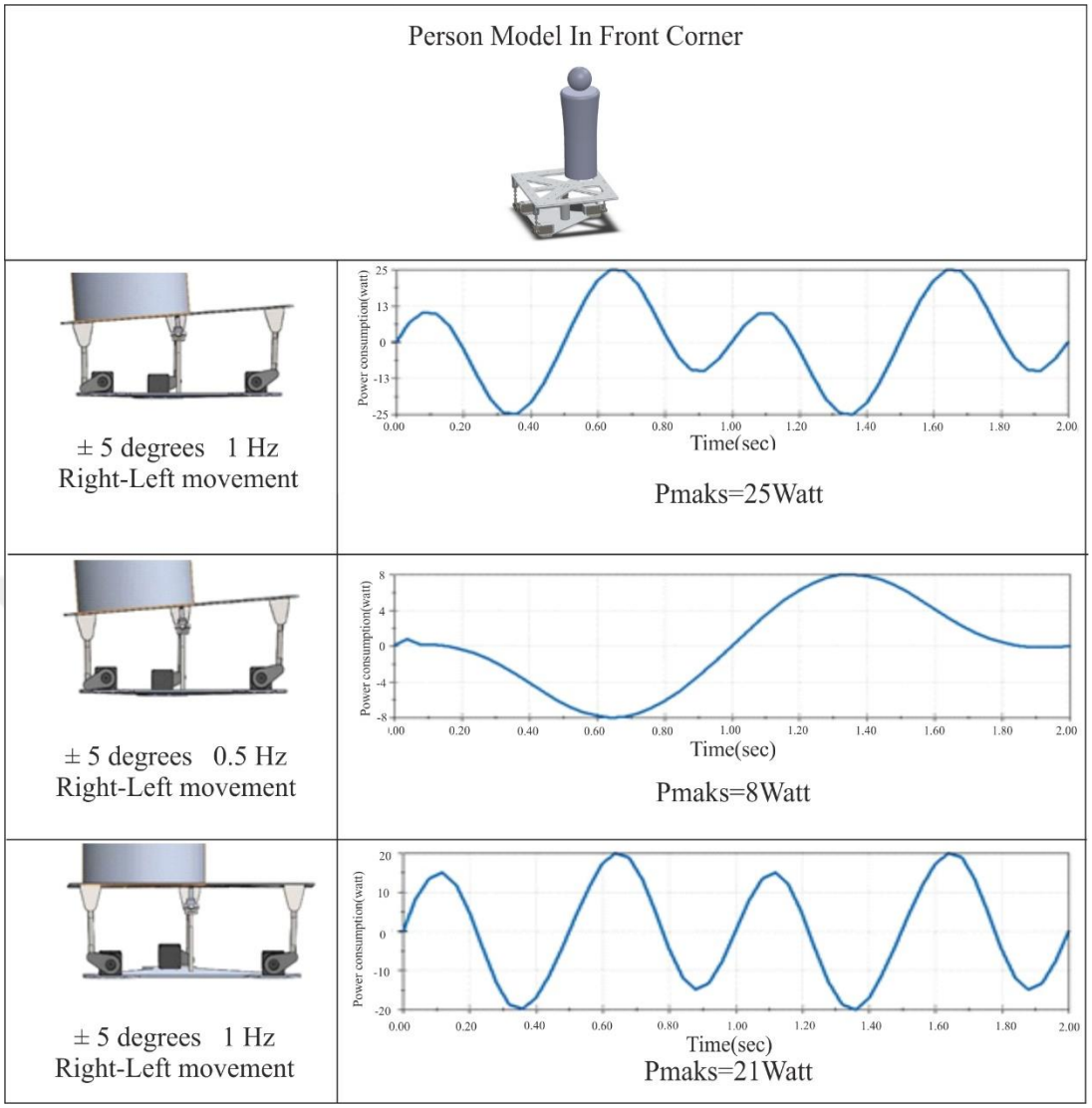


Figure 5.8 Power requirement for abnormal installation conditions when the person model is on the front

In the Figure 5.8, apart from the normal terms of use, the person is located in the approximate left-back of mechanism, the maximum power requirement for the right-left movement in 5 degree 1 Hz and 0.5 Hz is shown.

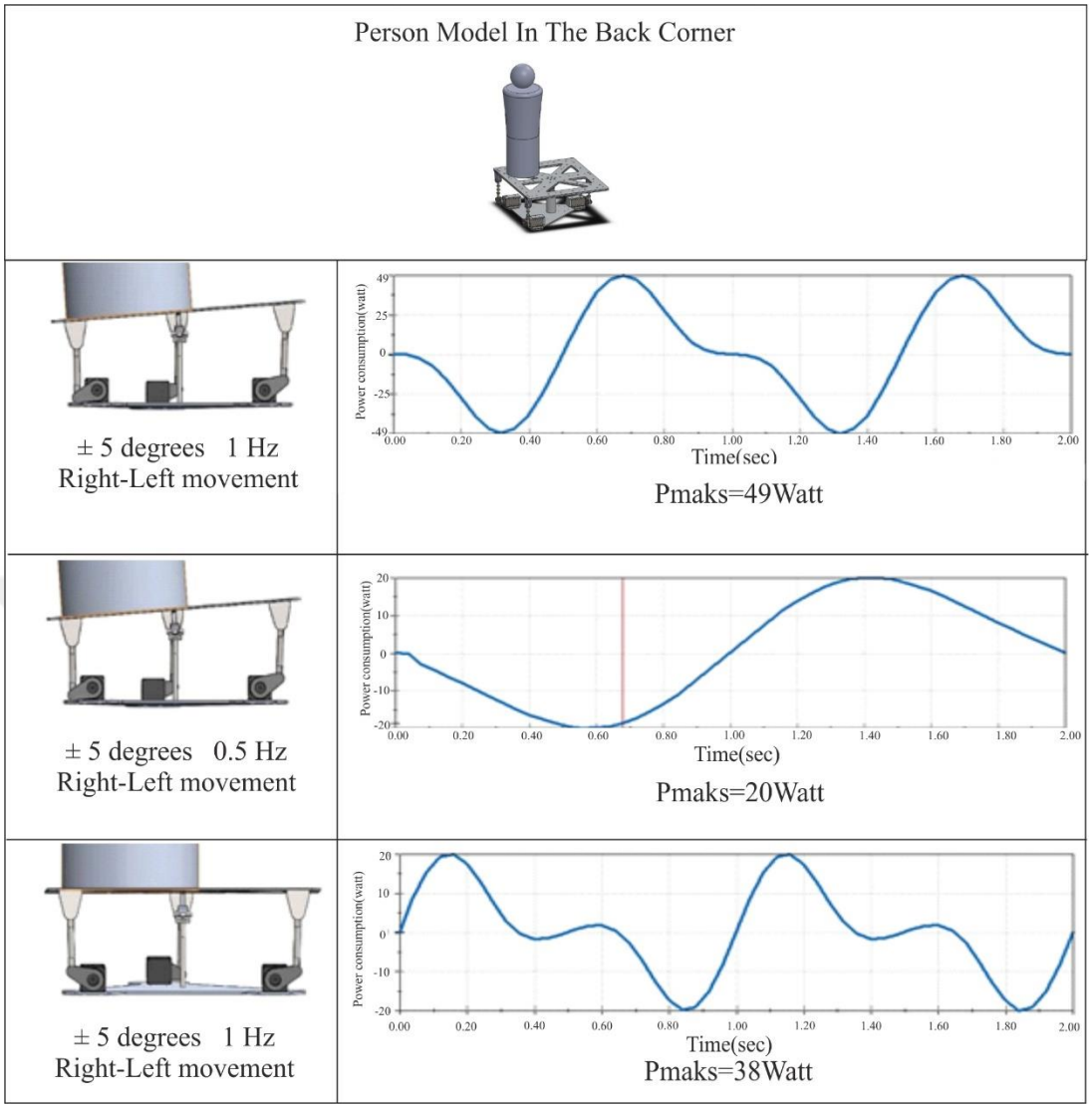


Figure 5.9 Power requirement for abnormal installation conditions when the person model is on the back side

In the Figure 5.9, apart from the normal terms of use, the person is located in the back of mechanism, the maximum power requirement for the right-left movement in 5 degree 1 Hz and 0.5 Hz is shown.

CHAPTER SIX
STRUCTURAL ANALYSIS OF FINITE ELEMENT METHOD OF BALANCE PLATFORM

The Finite Element Method (FEM) is a numerical solution which try to find out a solution with acceptable for various engineering problems. It is used in civil engineering in 1950's for the first time. Efficient usage is provided with development in computer technology. Even though this method is improved for original construction systems, accordingly of cornerstones, it is used as a means of problem solving in many department as fluid mechanics, electromagnetic fields, aircraft engineer, thermal analysis, soil mechanics, nuclear engineering and rock mechanics.

It is given the SolidWorks model which belongs movable portable balance platform which is finished prototype in this project, in Figure 6.1. This model is finished with considering a person who is 80 kg, on it, for the effect of the load on different parts of the balance platform with SolidWorks Simulation software and static finite element analysis. Consequences of these analysis, it is analyzed safety coefficient for the platform.

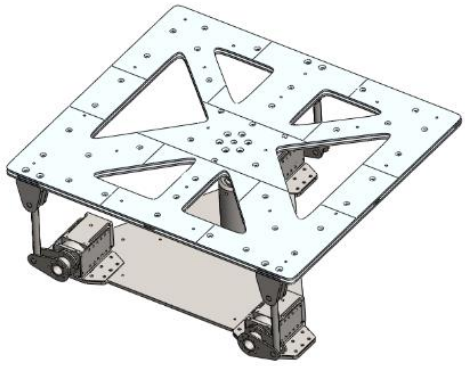


Figure 6.1 Montage model for mobile moving balance platform

For the finite element analysis, firstly the finite element model which is belong to montage model, which is shown in Figure 6.2, is composed. Moving balance platform model is a mounting model, which is consisting of pieces of different sizes,

6 mm for double-deck platform table and 15 mm for other parts, is used for meshing. There are total 333892 elements and 585657 nodes in final finite elements model. It is used triangle tetrahedron elements that have 10 nodes, curvilinear based mesh implementation, in detail parts of it, it is provided to reduction element size.

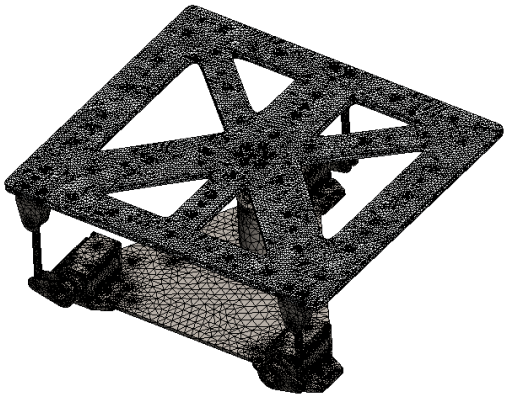


Figure 6.2 Finite element model for portable moving balance platform

The force application has been made from nine different zones on the platform in order to represent the situation that the person presses on different parts on the balance platform. Results are shown in Figure 6.3 and Figure 6.4.

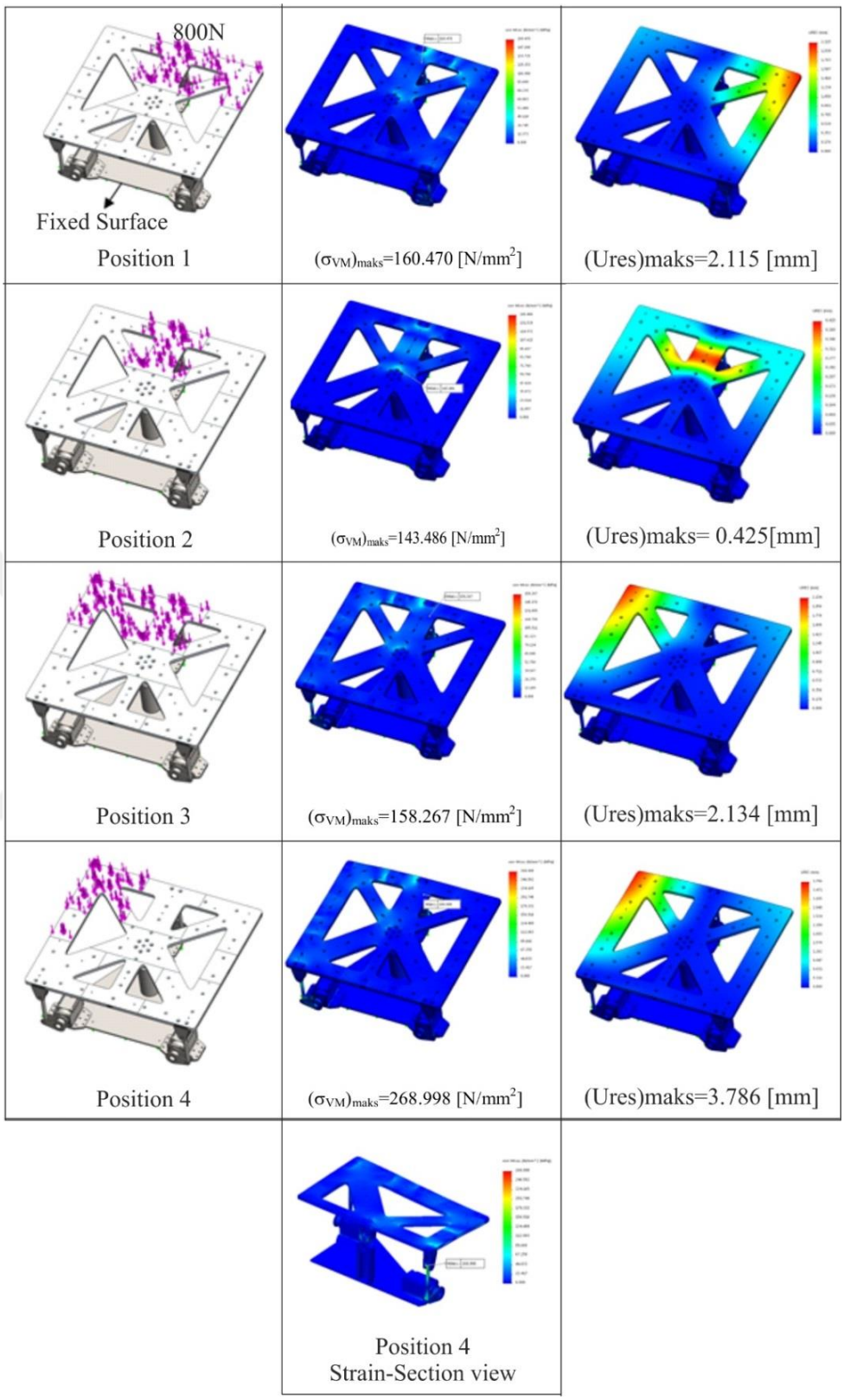


Figure 6.3 Stress and combination displacement results for different load locations

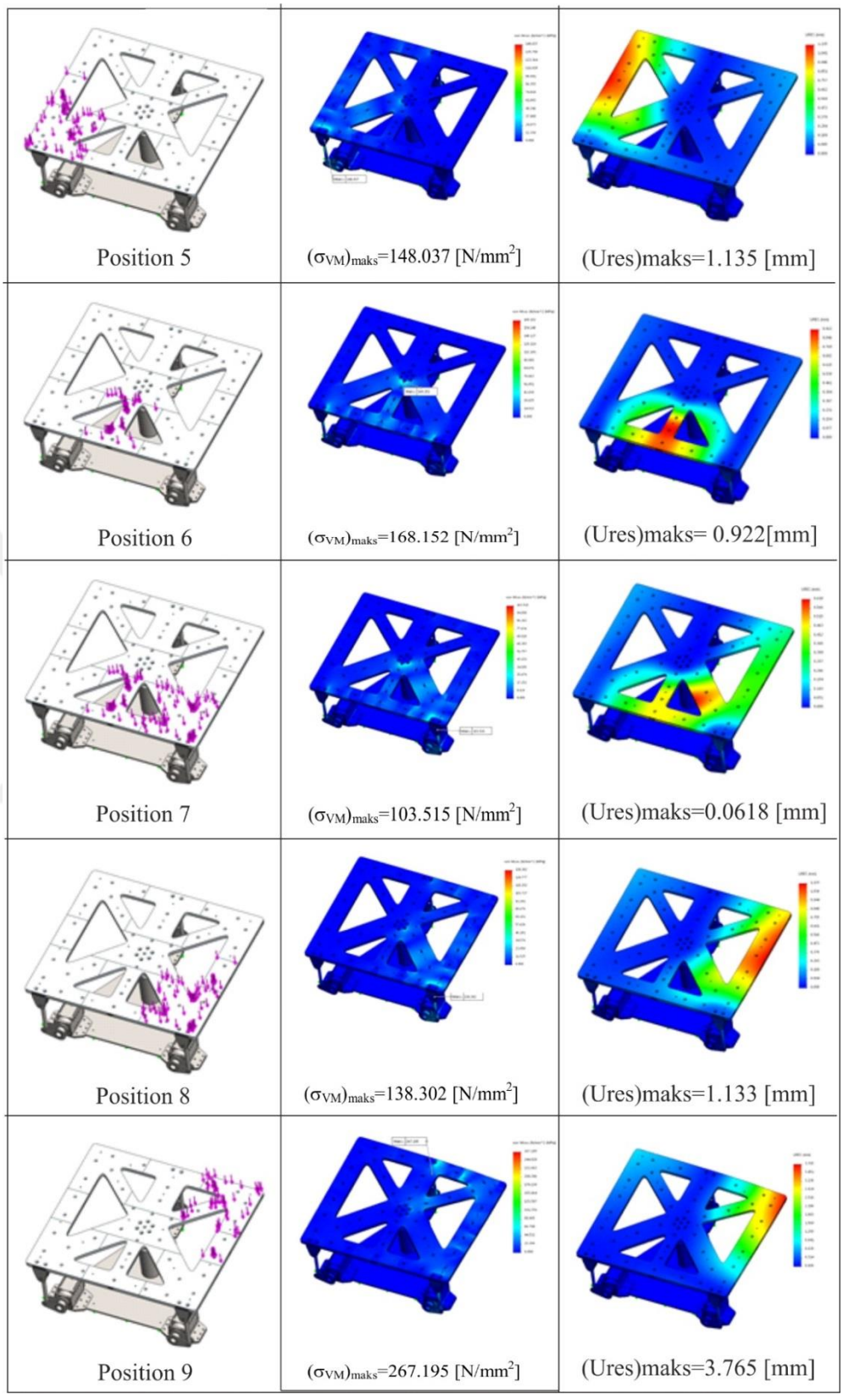


Figure 6.4 Stress and combination displacement results for different load locations

The highest von Mises tension value is obtained on actuator arm.

The highest von Mises tension value that obtained from nine different zones and the biggest intersection shifting are given in Figure 6.5.

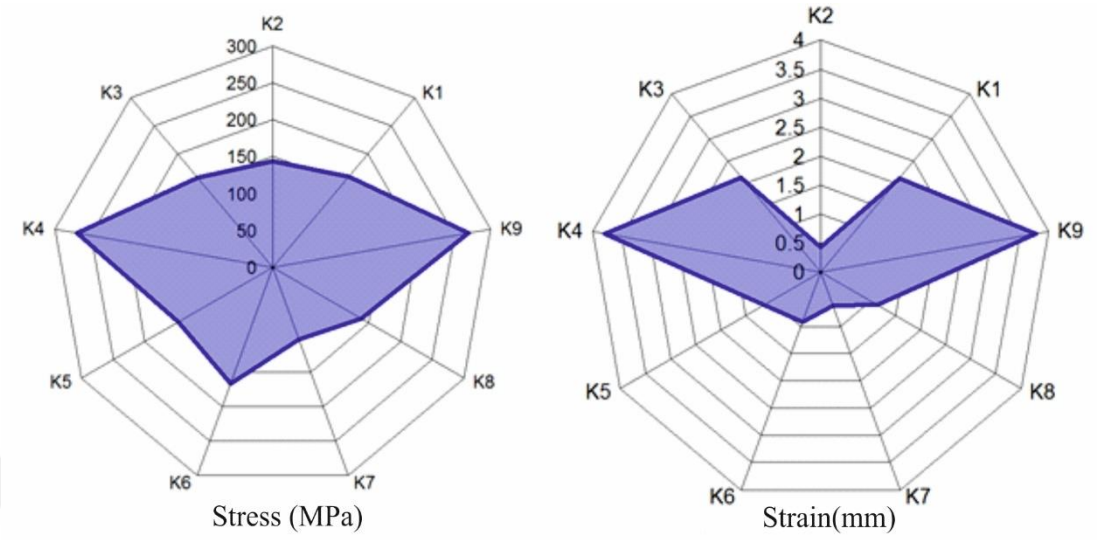


Figure 6.5 Tension and displacement value changes by load position

As a result of the static analysis, it is observed that the biggest stress where actuator arm hinged joint area, the highest values are for location 4 and location 9, 268.99 N/mm^2 and 267.19 N/mm^2, respectively. It is used AISI-1020 material for the actuator arm and yield stress is σ_{ak}=351.57 N/mm^2 for this material. Accordingly, the lowest safety coefficient number is s=351.57/268.99=1.3 for movable balanced platform. During tests, the upper platform which is the person on it, is made of 6063 aluminums and its yield stress is 145.00 N/mm^2. Accordingly, stress analysis, it is observed that the highest stress value on upper platform is under the 70.00 N/mm^2, thus the lowest safety coefficient is about s=2 for the upper platform. Static analysis has taken into account the worst cases for the loading position, and the obtained safety coefficients are considered to be sufficient for the current output. The largest displacement values obtained were evaluated not to affect the balance analysis on the platform.

CHAPTER SEVEN
ELECTROMECHANICAL SYSTEM DATA COLLECTION AND CONTROL

Although the motor motions of the produced prototype are not expected to be as perfect as in the simulation, they are very close. (Figure 7.1) We continued to study on necessary software changes that aim to the remove rapid acceleration at the end of the movement.

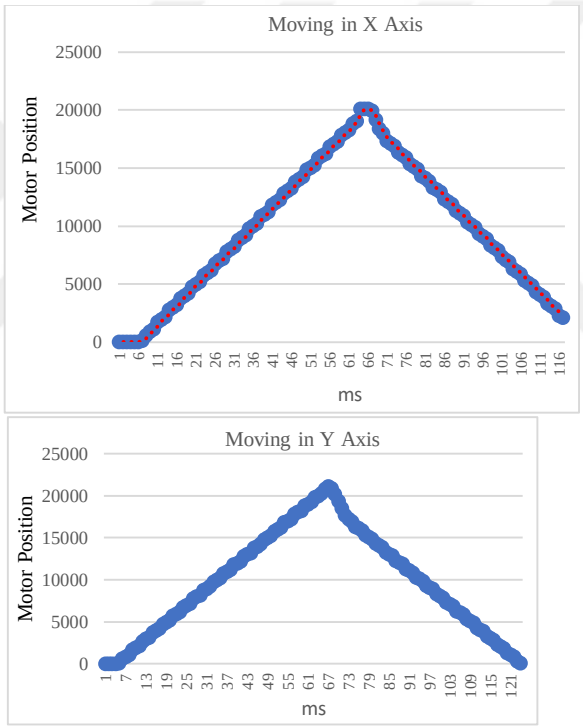


Figure 7.1 Motor motions

It is shown in Figure 7.2 that motion curve which is obtained in the equals solution for first prototype and in motors are used in prototype. It is continued to improve prototype motions.

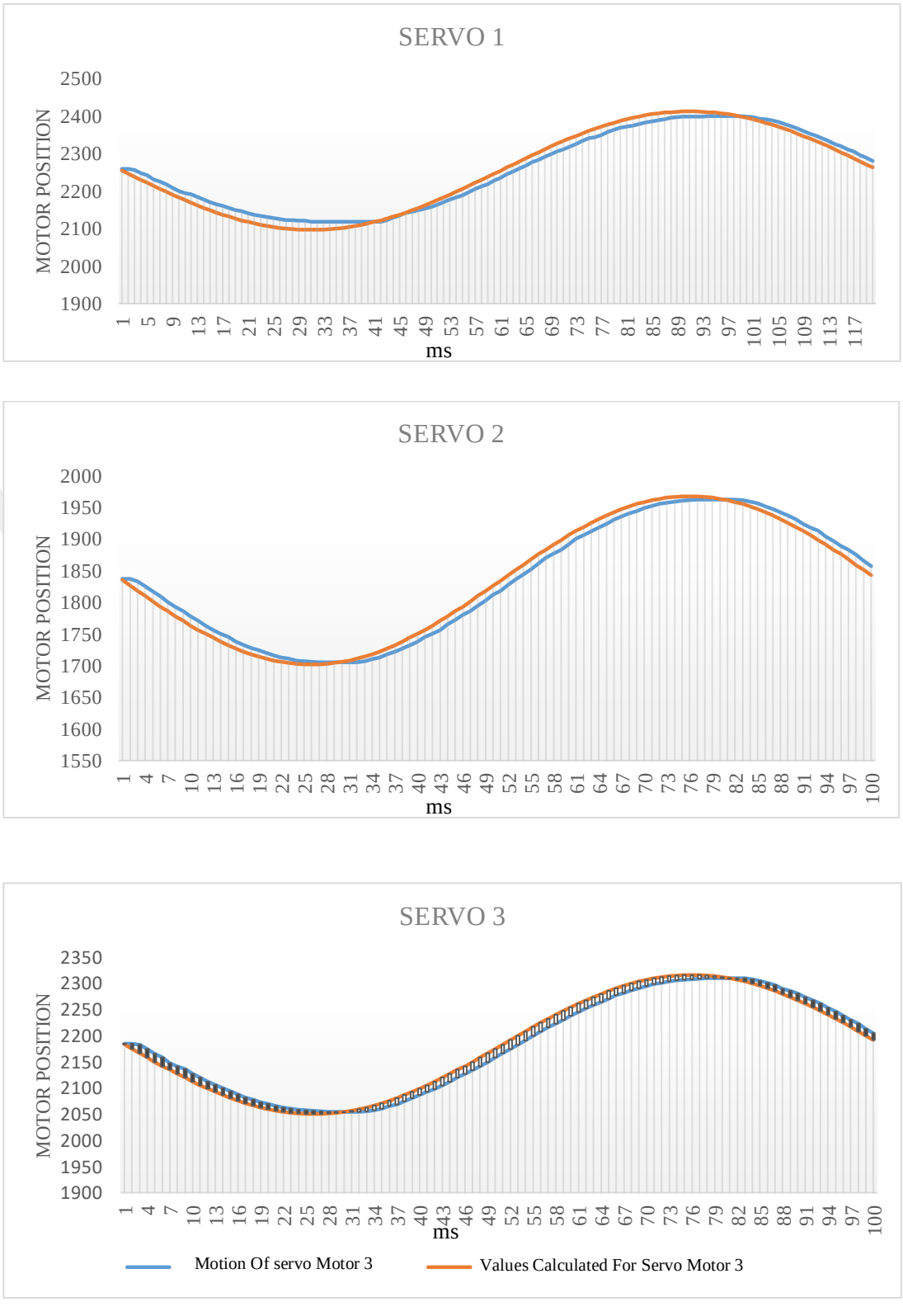


Figure 7.2 Servo motor motions

It is calculated correlation coefficient for evaluation between motion curves.

Motor 1 correlation coefficient= 0.990675442

Motor 2 correlation coefficient= 0.992565784

Motor 3 correlation coefficient= 0.991391033

The fact that the values are quite close to 1, it is shown that the motors are in desired position.

CHAPTER EIGHT
DEVELOPMENT OF BALANCE MONITORING / EXCITATION SOFTWARE

Balance monitoring and excitation software has been developed for the prototypes produced, with the completion of prototypes. The developed software consists of two section as embedded system and remote access user interface.

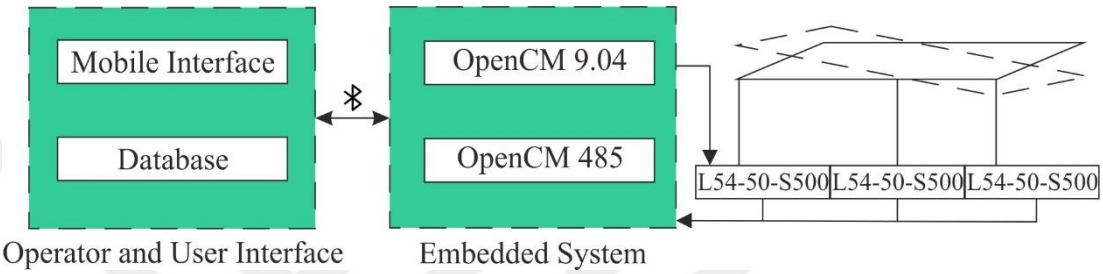


Figure 8.1 Developed software

The embedded system part is consisting of kinematic calculation, motion control and data acquisition. Kinematic calculation is given in previous chapters detailed. Motion control software and data acquisition steps are as in the Figure 8.1.

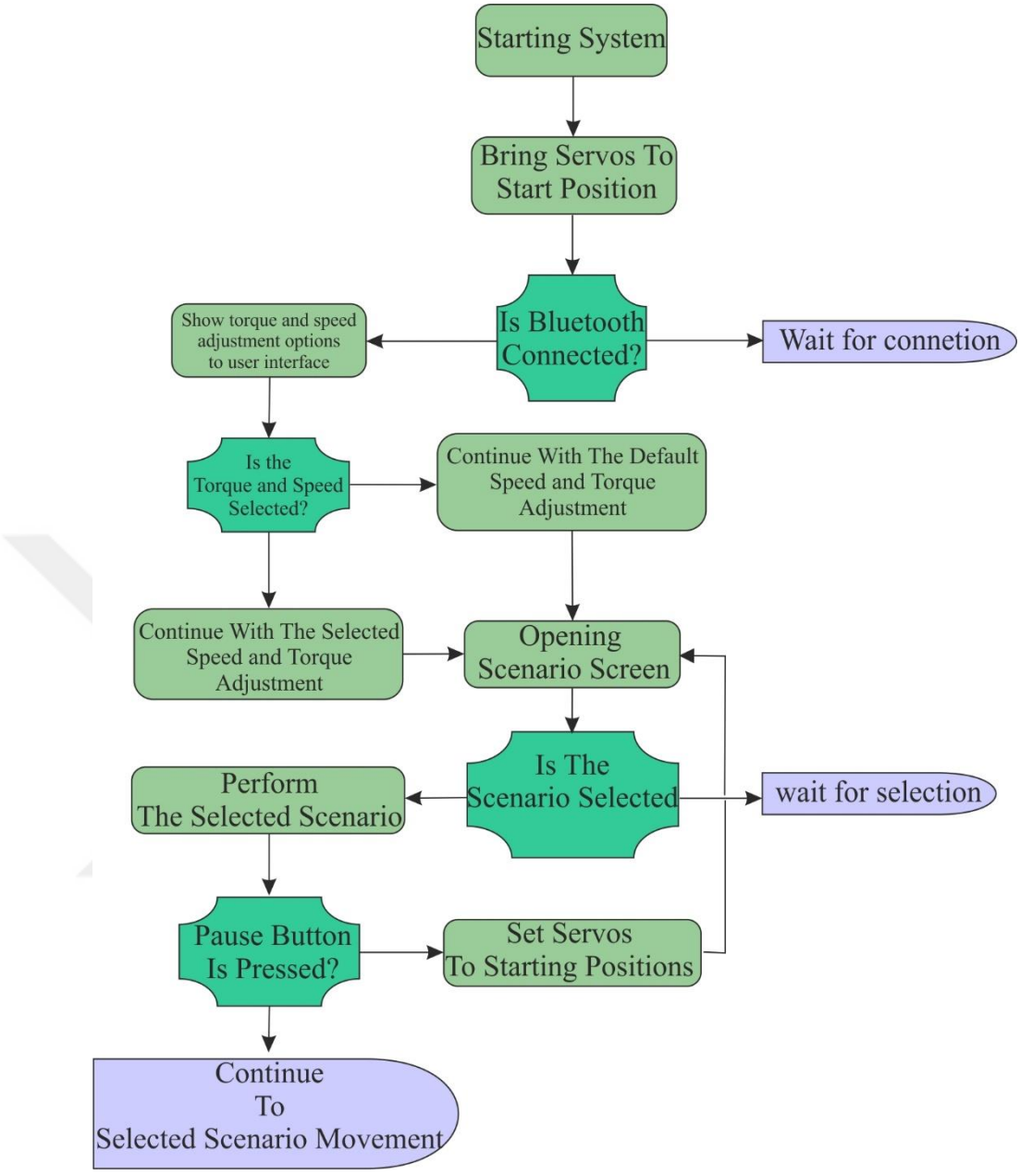


Figure 8.2 Motion control software and data acquisition block diagram

The embedded system has been connected to remote access user interface software via Bluetooth. Predetermined instruction by specialist or with the angle coming from master controller is sent to embedded OpenCm9.04 motor driving and control card by using serial communication based protocol.

During or after the motor state information has been sent to user interface for reporting to user. The screens and algorithm for this software are given below.

8.1. Development of user interface

Mobile interface is designed in order to use in mobile platform for balance excitation software and movable balance platform. (Figure 8.3)

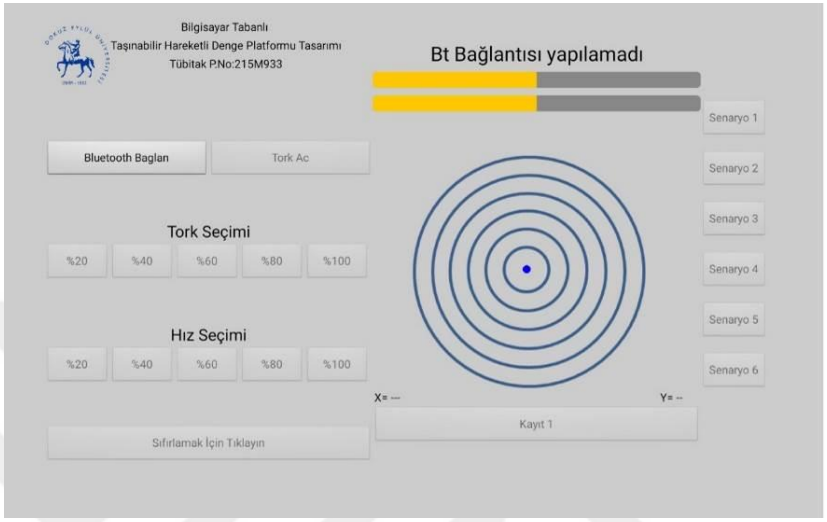


Figure 8.3 Development of user interface (Personal archive, 2018)

It is used to develop this interface from App Inventor2 that supported by MIT. (Figure 8.3)

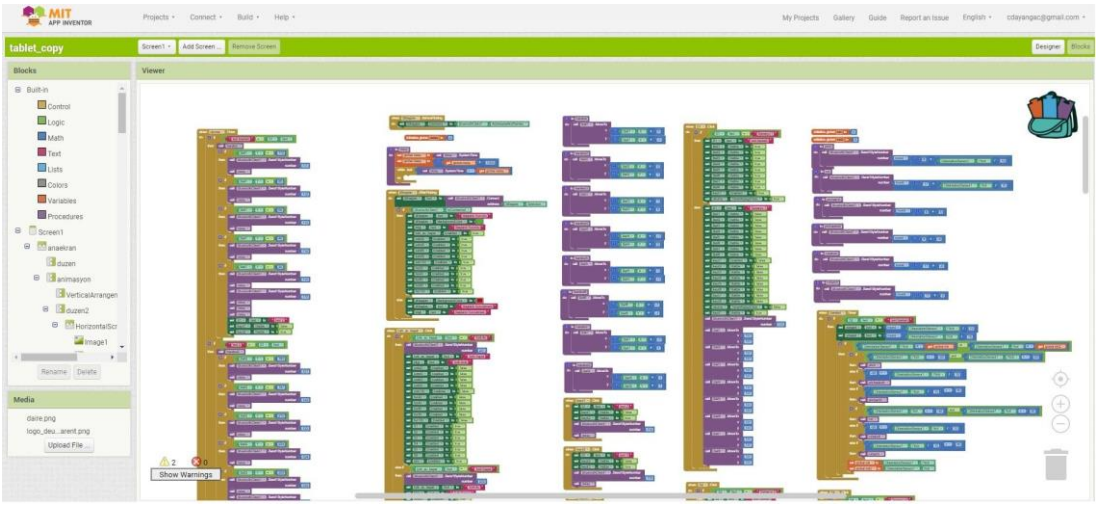


Figure 8.4 App Inventor2 application (Personal archive, 2018)

App Inventor2 has been provided to create a lots of software for Android platform. The application which is created thereby using App Inventor2 is used for experimental studies. But it has a professional prepared a mobile application with service procurement in order to present debugged and more proper application to the ultimate customer. It is given that a block diagram about mobile application operation at below. (Figure 8.4)

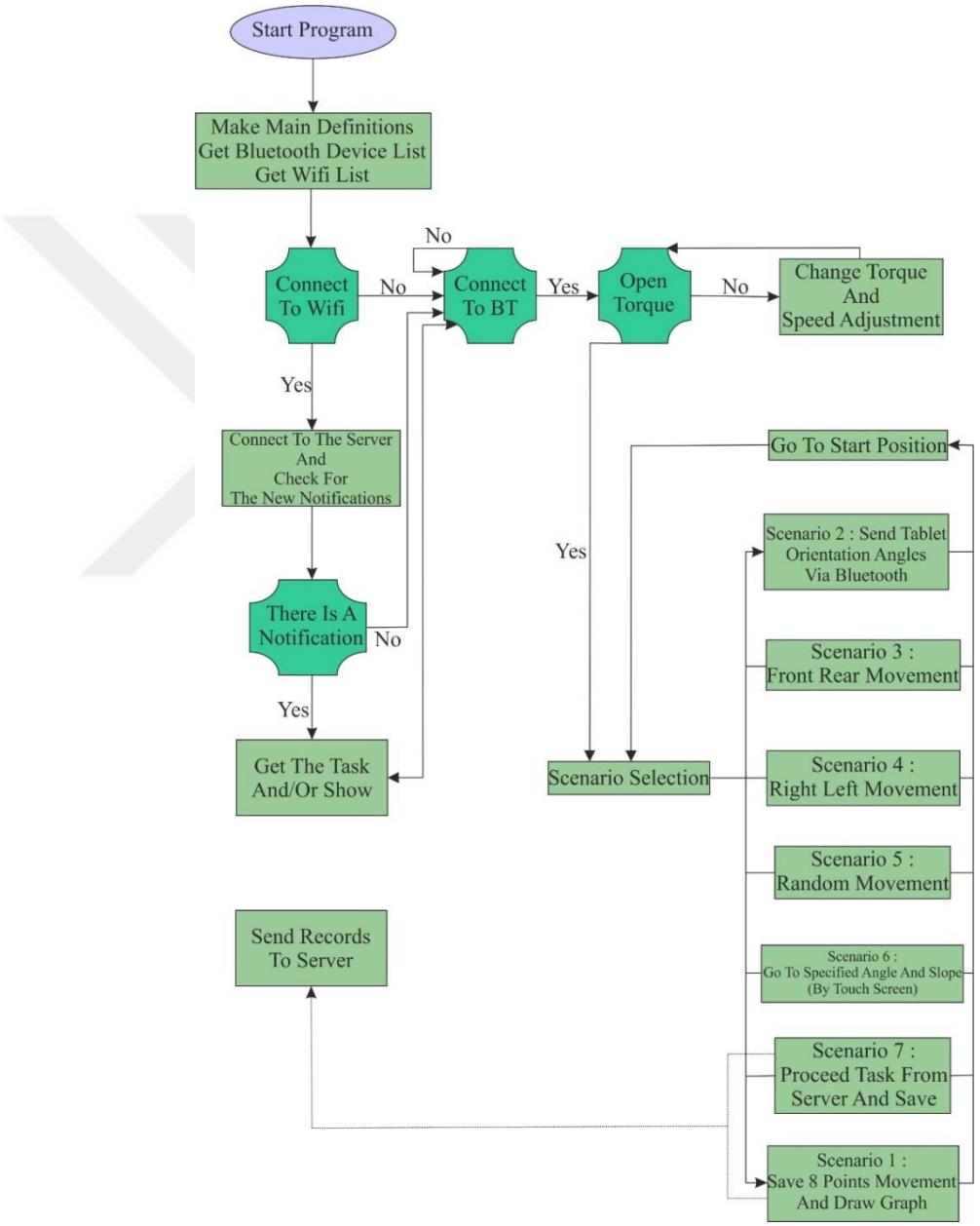


Figure 8.5 Mobile application operation

In Figure 8.2 and 8.5, application planning, initialization and sequent sorting. It is preferred to use firebase system of Google so as to composing database, a server where is located in Turkey in Google's firebase is chosen in order to fast access to data every time if internet connection is available. As a development tool, Android Studio which is suggested by Android, has used.

Tasks assigned to the patient are held by the specialist. Each patient has a unique key in which his or her duties are kept, and tasks are separated from each other. On the posts side, each user has his / her own title and the results of the test with a unique key within these titles and the date on which these results are stored.

CHAPTER NINE
CONCLUSION

Balance platform which is developed in this study, is cheap, portable and mobile. The mobility of this balance platform will be superior to the dynamic balance evaluation and static systems. It is important to evaluate the balance in human with different balance analysis and development scenarios during clinical studies and to be measurable and traceable. With the support of clinical personal could improve and it will be helpful for future studies.

In this study, by the light of the motivation given above, a mechanical device prototype, which has two degrees of freedom (2DOF), embedded motor controller with unique software, three smart servo motors with RS484 Daisy-chain connection and computer software for evaluation the results, was designed and implemented. As more detailed information, the designed prototype has several properties given as:

1. three small scale movable motion platforms, the biggest of which is 530mm X 600mm, have been developed.
2. The platform, that has 3 actuators and with 2 degree of freedom, makes pitching and stagger motion with maximum +/- 5 degree changing and it will not disturb the user.
3. All prototypes of the mechanical structure were subjected to mechanical analysis and simulations; and with the finite elements method for static case is analyzed, and required motor torques is calculated.
4. Motion platform includes 3 smart servo motor which can safely move person who is about 80 kg. These motors at the same time using as a sensor then can record angle, torque and velocity information.
5. Platform is scaled harmonic with Tekscan force mat which is used in Tübitak project numbered 215M933. With this equipment and software, high resolution force map can get. At the same time system can be monitored its center of gravity with 4 load cell that set up in the 4 edges.
6. The developed motion platform has been made wireless controllable from mobile Android devices. This function and platform motion can be changed

instantly with using mobile devise's gyro sensor, 8 scenarios can be automatically applied from simple front back motion to complex random shaking movements. This software can work remotely via the cloud.

7. With this study 2 different hardware and motor driver control applications was produced. If desired, system can remote both Windows and Linux.
8. The developed system has comparison tested with some test equipment which is most used in physical therapy. Accordingly, when the balance platform is fixed position, the robot on the top, when four main direction movement, CoG swing is compatible with swing data on Balance Master which is on the robot that used during test, so as to clinical balance evaluation and education. Likewise, movable platform can measure rightly Cog swinging, as accurate as Balance Master do.

The prototype which has produced with consequential of these studies above, use with test purpose in the Dokuz Eylül University, School of Physical Therapy and Rehabilitation Vocational High School. With the help the information which has made with this prototype, it is possible to new devices and analysis procedure.

By the help of the this study, it may be expected that physical therapy and rehabilitation devices which are small-sized, low-cost and compatible to personal usable, can improve the human health. Both the patients and healthy persons may use the system by changing the motion limits locally and remotely. Remote operations may develop response time for medical professionals.

REFERENCES

Alveroğlu, A. (2015). *Emg Kontrollü Alt Ekstremite Rehabilitasyon Denge Sistemi*. M.Sc Thesis, Marmara Üniversitesi, İstanbul.

Ambrozy, C., Rattay, F. (2011). Assessment criteria for evaluating the stability and position of the centre of gravity on a balance training platform: a simulation with Matlab. *Journal Of Medical Engineering & Technology*. *35*(5), 239-45.

Aminzadeh, M., Mahmoodi, A., Sabzehparvara, M. (2009). Dynamic Analysis of a 3DoF Motion Platform. *International Journal Of Robotics, 1*(1).

Anlı, E., Alp, H., Yurt, S. N., Özkol, I. (2005). Paralel mekanizmalarin kinematiği, dinamiği ve çalişma uzayi. *Havacilik Ve Uzay Teknolojileri Dergisi, 2*, 19-36.

Arockia, S., A, Arul, K. M. (2014). Experimental investigation on position analysis of 3 DOF parallel manipulators. *Procedia Engineering, 97*, 1126 – 1134.

Aydın, Ö.,(n.d).*Simülasyon*. Retrieved October 18, 2018, from https://docplayer.biz.tr/38756562-Simulasyon-hazirlayan-ozlem-aydin.html

Clark, D. J., Christou, E. A., Ring, S. A., Williamson J. B., Doty, L. (2014). Enhanced somatosensory feedback reduces prefrontal cortical activity during walking in older adults. *The journals of gerontology. Series A, Biological sciences and medical sciences, 69*(11), 1422-8.

Clerc, J. P., Tol, U.A., Wiens, G. J. (2002). *Deburring Using A Micro/Macro Parallel Kinematic Machine, Proc. Conference on Recent Advances in Robotics. Design and Modeling of a 3 DOF Machine*. Retrieved June 15, 2018. http://www.mae.ufl.edu/SAMM/Publications/FCRAR2002sentbyDRG.pdf

Dilan, R. A., Yaman, U., Çalişkan, H., Demirer, S., Balkan, T. (2011). 3 serbestlik derecesine sahip bir hareket taklitçisinin mafsal uzayi eniyilemesi ve çözüm analizi. *VI. ulusal hidrolik pnömatik kongresi, 12-15 ekim 2011,* Izmir.

Dongsu, W., Hongbin, G. (2007). Adaptive Sliding Control of Six-DOF Flight Simulator Motion Platform. *Chinese Journal of Aeronautics, 20*, 425-433.

Duarte, M., Freitas, SM. (2010). Revision of posturography based on force plate for balance evaluation. *Brazilian Journal of physical therapy*, *14*(3), 183-92.

Fang,Y.,Tsai, L. (2004). Structure Synthesis of a Class of 3-DOF Rotational Parallel Manipulators *Transactions on Robotics And Automation*, *20*(1), 117-121.

Gopalai, A. A., Senanayake, S. M., Kiong, L. C., Gouwanda, (2011). D. Real-time stability measurement system for postural control. *Journal of bodywork and movement therapies, 15*(4), 453-64.

Gough, V. E., Whitehall, S. G., (1962). Universal tyre test machine, *Proc. 9th International Technical Congress,* 177.

Güney, A., (2010*). İnsan Makine Arayüzü olarak 3x3 Stewart Platformunun Empedans Kuvvet Kontrolü,* M Sc. Thesis, Yıldız Teknik Üniversitesi, İstanbul.

Haksever, B., Düzgün, I., Yüce, D., Baltacı, G. (2017). Sağlıklı Bireylere Standart Denge Eğitiminin Dinamik, Statik Denge Ve Fonksiyonellik Üzerine Etkileri. *Gazi Üniversitesi Sağlık Bilimleri Dergisi*, *2*(3), 40-49.

Hua, C., Weishan, C., Junkao, L. (2007). Optimal Design of Stewart Platform Safety Mechanism *Chinese Journal of Aeronautics, 20*, 370-377.

Hunt, K. H., (1983). Structural kinematics of in-parallel-actuated robot-arms, *Journal of Mechanisms, Transmissions, and Automation in Design 105*, 705–712.

Kingma, H., Gauchard, G.C., Waele, C., Nechel, C., Bisdorff, A., Yelnik, A, et al. (2011). Stocktaking on the development of posturography for clinical use. *Journal of vestibular research : equilibrium & orientation*, *21*(3), 117-25.

Konrad, HR., Girardi, M., Helfert, R. (1999). Balance and aging. *The Laryngoscope*. *109*(9), 1454-60.

Liu, X., Wang, J., Pritschow, G. (2006). On the optimal kinematic design of the PRRRP 2-DoF parallel mechanism. *Mechanism and Machine Theory 41*, 1111–1130

McKerrow, P. J., (1991). *Introduction to Robotics*. Boston: Addison-Wesley Publishing Company.

Monteiro-Juniorf, RS., Ferreira, AS., Puell, VN., Lattari, E., Machado, S., Otero Vaghetti, CA., et al., (2015). Wii Balance Board: Reliability and Clinical Use in Assessment of Balance in Healthy Elderly Women. *CNS & neurological disorders drug targets. 14*(9), 1165-70.

NgS, C.C., OngA, K., Nee, Y. C. (2006). Design and development of 3–DOF modular micro parallel kinematic manipulator. *The International Journal of Advanced Manufacturing Technology, 31,* 188–200.

Piirtolaf, M., Era, P. (2006). Force platform measurements as predictors of falls among older people - a review. *Gerontology*, *52*(1), 1-16.

Pollard, W. L. (1942). *Position Controlling Apparatus*, June 18 US Patent No: 2.286.571,.

Prosperini, L., Pozzilli, C. (2013). The Clinical Relevance of Force Platform Measures in Multiple Sclerosis: A Review. *Multiple Sclerosis International,* 9.

Robotis. (2018). *Dynamixel Ex-106+ smart servo* . Retrieved March 15, 2018, http://support.robotis.com/en/product/actuator/dynamixel/ex_series/ex-106.htm

Robotis. (2018). *Dynamixel L54-50-S500 smart servo* . Retrieved March 15, 2018, http://emanual.robotis.com/docs/en/dxl/pro/l54-50-s500-r/

Robotis. (2018). *OpenCM 9.04 Arm controller*. Retrieved March 15, 2018, http://emanual.robotis.com/docs/en/parts/controller/opencm904/

Robotistan. (2018). *MPU6050 6 Eksen İvme ve Gyro Sensörü*. Retrieved March 15, 2018, https://www.robotistan.com/mpu6050-6-eksen-ivme-ve-gyro-sensoru-6-dof-3-axis-accelerometer-and-gyros

Soyuer, F., Köseoğlu, E. (2001). Dengenin Klinik Değerlendirilmesi, *Erciyes Üniversitesi, Sağlık Bilimleri Dergisi*, 2001, 75-82.

Sunguray, C., Ürgün, S., Demirtaş, H., Güngör, S. (2014). Konum Kontrollü 6x6 Serbestlik Dereceli Stewart Platformu'nun Tasarımı Modellenmesi ve Simülasyonu. *SDU International Technologic Science*, *6*(3), 49-61.

T.C. Milli Eğitim Bakanlığı, (2012), *Tekstil Teknolojisi - Robotik* , Ankara 101.

Udoo. (2018). *Udoo quad mini PC*. Retrieved March 15, 2018, https://www.udoo.org/udoo-dual-and-quad/

Ulaş, B. (2009). *Stewart Platformu Tasarimi*. M.Sc. Thesis, Istanbul Teknik Üniversitesi, İstanbul.

Ünsal A. (2007). *Farklı Yapıdaki Stewart Platform Mekanizmalarının Düz ve Ters Kinematik Analizi*, Post Graduate Thesis, Yıldız Teknik Üniversitesi, İstanbul.

Walsh, L., Greene, BR., McGrath, D., Burns, A., Caulfield, B. (2011). Development and validation of a clinic based balance assessment technology. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society Engineering in Medicine and Biology Society Annual Conference, 2011*, 1327-30.

Zabalza, I., Ros, J., Gil, J. J., Pintor, J. M., Jimenez, J. M., (2002). *TRI-SCOTT. A MICABO like 6-DOF Quasi-Decoupled Parallel Manipulator*, Retrieved March 15, 2018, http://www.imem.unavarra.es/web_imac/documentation/Tri-cott_A_Micabo_like_6-DOF_Quasi_Decoupled_Parallel_Manipulator_Workshop_on_Fundamental_Issues_and_Future_Research_Directions_for_Parallel_Mechanisms_and_Manipulator.pdf

Zhang, C., Zhang, L. (2013). Kinematics analysis and workspace investigation of anovel 2-DOF parallel manipülatör applied in vehicle driving simulator. *Robotics and Computer-Integrated Manufacturing, 29*, 113–120.

Zhu Y. (2017). Design and Validation of a Low-Cost Portable Device to Quantify Postural Stability. *Sensors(Basel), 17*(3), 619.

APPENDICES

APPENDIX 1: Dynamixel L54-50-S500-R Control Codes

```
/* DYNAMIXEL L54-50-S500-R control program.

Rotational resolution is 361,384 steps / 360 degree rotation

*/

// PARAMETERS

#define BAUDRATE_9600      0

#define BAUDRATE_57600     1

#define BAUDRATE_115200    2

#define BAUDRATE_1M        3

// RS485EXP REGISTERS

#define RS485EXP_BUTTON1   16

#define RS485EXP_BUTTON2   17

#define RS485EXP_REDLED    18

#define RS485EXP_GREENLED  19

#define RS485EXP_BLUELED   20

// MOTOR REGISTERS

#define OPERATING_MODE     11

#define TORQUE_LIMIT       30

#define TORQUE_ENABLE      562

#define VELOCITY_P_GAIN    588

#define POSITION_P_GAIN    594

#define GOAL_POSITION      596

#define GOAL_VELOCITY      600
```

```
#define GOAL_TORQUE       604
#define MOVING            610
#define PRESENT_POSITION  611
#define PRESENT_VELOCITY  615
#define INTTEMP           625
#define REDLED            563
#define GREENLED          564
#define BLUELED           565
#define POSITION_HIGH_LIMIT  180684
#define POSITION_LOW_LIMIT  -180684
// MOTORS
#define MOTORCOUNT        3     // motor count
#define MOTOR1            1     // motor 1 id
#define MOTOR2            2     // motor 2 id
#define MOTOR3            3     // motor 3 id

// CONTROL PARAMETERS
#define MAX_PROGRAM_MODE=4;  // 1: normal run 2: test run
// VARIABLES
int program_mode;    // 0: no program run
                     // 1: 5 degree run test
int motorno;
float     goal_angle[MOTORCOUNT];
long int  goal_position[MOTORCOUNT];
long int  goal_velocity[MOTORCOUNT];
```

```
long int  present_position[MOTORCOUNT];
int      Torque_Enable[MOTORCOUNT]={0,0,0};
int      model;
int      i;
char btt;
int devam;
char torkacik;
char menusel;
char secim;
int temp; // pitch acısı için gelen veri
int junk;
float A;
float AA,AAA,psi,phi,theta,L;
double U,UU;
float L1,l1,l2,l3,M1,N1,J1,U1;
int B,b,h;
float a,c;
float n,nn;
Dynamixel Dxl(3);  // 3: OpenCM 485EXP
void setup() {
program_mode=0;   // NO PROGRAM MODE SET NOW
  Serial2.begin(9600);
  SerialUSB.begin();                    // USB Serial init
  Dxl.begin(BAUDRATE_57600);                // 1
  pinMode(RS485EXP_BUTTON1,  INPUT_PULLDOWN);
```

```
pinMode(RS485EXP_BUTTON2,  INPUT_PULLDOWN);

pinMode(BOARD_LED_PIN,     OUTPUT);

pinMode(RS485EXP_REDLED,   OUTPUT);

pinMode(RS485EXP_GREENLED, OUTPUT);

pinMode(RS485EXP_BLUELED,  OUTPUT);

digitalWrite(RS485EXP_REDLED, 1);

digitalWrite(RS485EXP_GREENLED, 1);

digitalWrite(RS485EXP_BLUELED, 1);

attachInterrupt(RS485EXP_BUTTON1,torque_onoff, RISING);

attachInterrupt(RS485EXP_BUTTON2,program_run, RISING);

delay(4000);                          // GETTING READY

for (i=0; i <= MOTORCOUNT; i++)

{

model = Dxl.getModelNumber(i);  if(model == 38152) { motor_init(i);  SerialUSB.print("L54 PRO Motor
"); SerialUSB.print(i); SerialUSB.println(" Initialized"); }

}

 delay(1000);

  torque_on();

  goal_angle={0.0,0.0,0.0};

  for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,7000);    motor_goal(i,goal_angle[i]); }

  delay(5000);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }

  SerialUSB.println();

  delay(2000);

 // torque_off();

  //SerialUSB.print("PROGRAM MODE NUMBER= ");     SerialUSB.println(program_mode);
```

```
    SerialUSB.attachInterrupt(usbInterrupt);
  // Serial2.attachInterrupt(serialInterrupt);
a=0.0;
n=0;
nn=0;
temp=0;
h=0;
}
void loop() {
if(Serial2.available()){
if(Serial2.available()>0){
   temp+=Serial2.read();
 }
SerialUSB.println(temp);
if(temp==201){
 torque_on();
 SerialUSB.println("torque on");
 temp=0;
}
if(temp==202){
 torque_off();
 SerialUSB.println("torque off");
 temp=0;
}
if(temp==110){
```

```
 if(h==0){
SerialUSB.println("merkezlendi");
 temp=0;
 psi=0;
 theta=0;
 phi=0;
 hesap();
  goal_angle={A,AA,AAA};
  for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
  delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
 }
 else{
 temp=0;
 psi=0;
 theta=0;
 phi=0;
 hesap();
  goal_angle={A,AA,AAA};
  motor_speed(0,2200); motor_goal(0,goal_angle[0]);
  motor_speed(1,1000); motor_goal(1,goal_angle[1]);
  motor_speed(2,2200); motor_goal(2,goal_angle[2]);
 }
}
if(temp==111){
 SerialUSB.println("top 1 1 derece");
```

```
 temp=0;
  psi=0;
  theta=0;
  phi=1*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }

   h=0;
}
if(temp==112){
 SerialUSB.println("top 1 2 derece");
 temp=0;
   psi=0;
  theta=0;
  phi=2*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==113){
 SerialUSB.println("top 1 3 derece");
```

```
 temp=0;
   psi=0;
  theta=0;
  phi=3*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==114){
 SerialUSB.println("top 1 4 derece");
 temp=0;
   psi=0;
  theta=0;
  phi=4*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==115){
 SerialUSB.println("top 1 5 derece");
 temp=0;
```

```
  psi=0;
  theta=0;
  phi=5*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==121){
 SerialUSB.println("top 2 1 derece");
 temp=0;
  psi=0;
  theta=0;
  phi=-1*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==122){
 SerialUSB.println("top 2 2 derece");
 temp=0;
  psi=0;
```

```
  theta=0;
  phi=-2*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==123){
 SerialUSB.println("top 2 3 derece");
 temp=0;
   psi=0;
  theta=0;
  phi=-3*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==124){
 SerialUSB.println("top 2 4 derece");
 temp=0;
   psi=0;
  theta=0;
```

```
  phi=-4*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==125){
 SerialUSB.println("top 2 5 derece");
 temp=0;
  psi=0;
  theta=0;
  phi=-5*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}

if(temp==131){
 SerialUSB.println("top 3 1 derece");
 temp=0;
  psi=0;
  theta=-1*0.0175;
```

```
  phi=0;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==132){
 SerialUSB.println("top 3 2 derece");
 temp=0;
  psi=0;
  theta=-2*0.0175;
  phi=0;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==133){
 SerialUSB.println("top 3 3 derece");
 temp=0;
  psi=0;
  theta=-3*0.0175;
  phi=0;
```

```
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==134){
 SerialUSB.println("top 3 4 derece");
 temp=0;
  psi=0;
  theta=-4*0.0175;
  phi=0;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=0;
}
if(temp==135){
 SerialUSB.println("top 3 5 derece");
 temp=0;
  psi=0;
  theta=-5*0.0175;
  phi=0;
  hesap();
```

```
        goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=0;
}
if(temp==141){
 SerialUSB.println("top 4 1 derece");
 temp=0;
   psi=0;
  theta=1*0.0175;
  phi=0;
  hesap();
        goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=0;
}
if(temp==142){
 SerialUSB.println("top 4 2 derece");
 temp=0;
   psi=0;
  theta=2*0.0175;
  phi=0;
  hesap();
        goal_angle={A,AA,AAA};
```

```
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=0;
}
if(temp==143){
 SerialUSB.println("top 4 3 derece");
 temp=0;
   psi=0;
  theta=3*0.0175;
  phi=0;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==144){
 SerialUSB.println("top 4 4 derece");
 temp=0;
   psi=0;
  theta=4*0.0175;
  phi=0;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
```

```
}
if(temp==145){
 SerialUSB.println("top 4 5 derece");
 temp=0;
   psi=0;
  theta=5*0.0175;
  phi=0;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==151){
 SerialUSB.println("top 5 1 derece");
 temp=0;
   psi=0;
  theta=-1*0.0175;
  phi=1*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=1;
}
if(temp==152){
```

```
 SerialUSB.println("top 5 2 derece");
 temp=0;
   psi=0;
  theta=-2*0.0175;
  phi=2*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==153){
 SerialUSB.println("top 5 3 derece");
 temp=0;
  psi=0;
  theta=-3*0.0175;
  phi=3*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==154){
 SerialUSB.println("top 5 4 derece");
 temp=0;
   psi=0;
```

```
  theta=-4*0.0175;
  phi=4*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==155){
 SerialUSB.println("top 5 5 derece");
 temp=0;   psi=0;
  theta=-5*0.0175;
  phi=5*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==161){
 SerialUSB.println("top 6 1 derece");
   temp=0;
  psi=0;
  theta=1*0.0175;
  phi=1*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
```

```
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=1;
}
if(temp==162){
 SerialUSB.println("top 6 2 derece");
 temp=0;
   psi=0;
  theta=2*0.0175;
  phi=2*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==163){
 SerialUSB.println("top 6 3 derece");
 temp=0;
   psi=0;
  theta=3*0.0175;
  phi=3*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
```

```
}
if(temp==164){
 SerialUSB.println("top 6 4 derece");
 temp=0;
   psi=0;
  theta=4*0.0175;
  phi=4*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==165){
 SerialUSB.println("top 6 5 derece");
 temp=0;
   psi=0;
  theta=5*0.0175;
  phi=5*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==171){
 SerialUSB.println("top 7 1 derece");
```

```
 temp=0;
  psi=0;
  theta=1*0.0175;
  phi=-1*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
   h=1;
}
if(temp==172){
 SerialUSB.println("top 7 2 derece");
 temp=0;
  psi=0;
  theta=2*0.0175;
  phi=-2*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==173){
 SerialUSB.println("top 7 3 derece");
 temp=0;  psi=0;
  theta=3*0.0175;
```

```
  phi=-3*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==174){
 SerialUSB.println("top 7 4 derece");
 temp=0;
  psi=0;
  theta=4*0.0175;
  phi=-4*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);  for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==175){
 SerialUSB.println("top 7 5 derece");
 temp=0;
  psi=0;
  theta=5*0.0175;
  phi=-5*0.0175;
  hesap();
      goal_angle={A,AA,AAA};
```

```
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==181){
 SerialUSB.println("top 8 1 derece");
 temp=0;
   psi=0;
  theta=-1*0.0175;
  phi=-1*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
    h=1;
}
if(temp==182){
 SerialUSB.println("top 8 2 derece");
 temp=0;   psi=0;
  theta=-2*0.0175;
  phi=-2*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
```

```
if(temp==183){
 SerialUSB.println("top 8 3 derece");
 temp=0;
   psi=0;
  theta=-3*0.0175;
  phi=-3*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==184){
 SerialUSB.println("top 8 4 derece");
 temp=0;
   psi=0;
  theta=-4*0.0175;
  phi=-4*0.0175;
  hesap();
       goal_angle={A,AA,AAA};
   for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
   delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp==185){
 SerialUSB.println("top 8 5 derece");
 temp=0;
```

```
   psi=0;
   theta=-5*0.0175;
   phi=-5*0.0175;
   hesap();
        goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(500);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
}
if(temp>0&&temp<11){
   n=0;
   n=temp-6;
    psi=0;
   theta=nn*0.0175;
   phi=n*0.0175;
   hesap();
        goal_angle={A,AA,AAA};
    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
    delay(5);
    temp=0;
}
if(temp>10&&temp<22){
   nn=0;
   nn=temp-16;
   psi=0;
   theta=nn*0.0175;
```

```
 phi=n*0.0175;
 hesap();
      goal_angle={A,AA,AAA};
  for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,3000); motor_goal(i,goal_angle[i]); }
  delay(5);  // for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }
 temp=0;
}
if(temp==203){
 SerialUSB.println("Durdurma");
 temp=0;
}
}//Serial available
}//loop
// ---------------------------------------------------------------------- SUBROUTINES
int angle2pos(float angle) {  long int value = ((angle * POSITION_HIGH_LIMIT )/ 180);   return value;
// if(goal_position== POSITION_HIGH_LIMIT)  goal_position = POSITION_LOW_LIMIT;  else goal_position= POSITION_HIGH_LIMIT;
}
//---------------------------------------------------------------------- TORQUE ON
void torque_on(void) {  // RS485_EXP karttaki Button1 Torque acar kapatir
 int motor_no;
 delay(1000);
 for (motor_no=1; motor_no <= MOTORCOUNT;motor_no++)
 {
 Torque_Enable[motor_no]=1;
```

```
  digitalWrite(BOARD_LED_PIN, !Torque_Enable[motor_no]);
  Dxl.writeByte(motor_no, TORQUE_ENABLE, Torque_Enable[motor_no]);
  Dxl.writeByte(motor_no, REDLED, 255);
    }
}
//------------------------------------------------------------------------- TORQUE OFF
void torque_off(void) {   // RS485_EXP karttaki Button1 Torque acar kapatir
 int motor_no;
 delay(1000);
 for (motor_no=1; motor_no <= MOTORCOUNT;motor_no++)
 {
 Torque_Enable[motor_no]=0;
  digitalWrite(BOARD_LED_PIN, !Torque_Enable[motor_no]);
  Dxl.writeByte(motor_no, TORQUE_ENABLE, Torque_Enable[motor_no]);
  Dxl.writeByte(motor_no, REDLED, 0);
 Dxl.writeByte(motor_no, BLUELED, 255);
Dxl.writeByte(motor_no, GREENLED, 255);
 }
}
//------------------------------------------------------------------------- TORQUE ON/OFF INTERRUPT
void torque_onoff(void) {   // RS485_EXP karttaki Button1 Torque acar kapatir
 int motor_no;
 delay(2000);
 for (motor_no=1; motor_no <= MOTORCOUNT;motor_no++)
 {
```

```
 if (Torque_Enable[motor_no]==0) Torque_Enable[motor_no]=1;
 else Torque_Enable[motor_no]=0;
  digitalWrite(BOARD_LED_PIN, !Torque_Enable[motor_no]);
  Dxl.writeByte(motor_no, TORQUE_ENABLE, Torque_Enable[motor_no]);
  if (Torque_Enable[motor_no]==0) Dxl.writeByte(motor_no, BLUELED, 0);
  if (Torque_Enable[motor_no]==1) Dxl.writeByte(motor_no, BLUELED, 255);
  SerialUSB.print("Torque= "); SerialUSB.println(Torque_Enable[motor_no]);
 }
}
//------------------------------------------------------------------- TEST TORQUE  INTERRUPT
void program_run(void) {   // butonunun her basısında farklı bir program moduna gecer
   delay(2000);
    program_mode++;
   if (program_mode>2)  program_mode=0;

    if (program_mode==1) {  // TEST COLOR BLUE
    for (i=0; i < MOTORCOUNT; i++)  {
  Dxl.writeByte(i, REDLED, 0);
   Dxl.writeByte(i, BLUELED, 255);
  Dxl.writeByte(i, GREENLED, 0);
 }
 }
 if (program_mode==2) {  // TEST COLOR GREEN
    for (i=0; i < MOTORCOUNT; i++)  {
  Dxl.writeByte(i, REDLED, 0);
```

```
    Dxl.writeByte(i, BLUELED, 0);
   Dxl.writeByte(i, GREENLED, 255);
  }
  }
    SerialUSB.print("PROGRAM MODE NUMBER= "); SerialUSB.println(program_mode);
}
//------------------------------------------------------------------------ MOTOR GOAL WRITE
void motor_goal(int motor_no, float motor_aci)
{   motor_no=motor_no+1;
 goal_position[motor_no]=angle2pos(motor_aci);
}
    Dxl.writeDword(motor_no,GOAL_POSITION, goal_position[motor_no] );
}
//------------------------------------------------------------------------ MOTOR GOAL WRITE
void motor_speed(int motor_no, int motor_hiz)
{
  motor_no=motor_no+1;
   goal_velocity[motor_no]=motor_hiz;
}
  Dxl.writeDword(motor_no,GOAL_VELOCITY, goal_velocity[motor_no] );
//------------------------------------------------------------------------ MOTOR POS READ
void motor_pos(int motor_no)
{
   motor_no=motor_no+1;
    present_position[motor_no] = Dxl.readDword(motor_no, PRESENT_POSITION);
```

```
      SerialUSB.print("Motor "); SerialUSB.print(motor_no); SerialUSB.print(" Present Position = ");SerialUSB.println( present_position[motor_no]);

    }

    //------------------------------------------------------------------------- MOTOR INIT

    void motor_init(int motor_no)

    {

      motor_no=motor_no+1;

      Torque_Enable[motor_no]= Dxl.readByte(motor_no,TORQUE_ENABLE);

      Torque_Enable[motor_no]=0;

      Dxl.writeByte(motor_no, BLUELED, 0);

      Dxl.writeByte(motor_no, TORQUE_ENABLE, Torque_Enable[motor_no]);

      Dxl.writeByte(motor_no, OPERATING_MODE, 3);

      Dxl.writeWord(motor_no, TORQUE_LIMIT, 250);

      Dxl.writeWord(motor_no, VELOCITY_P_GAIN , 225);

      Dxl.writeWord(motor_no, POSITION_P_GAIN, 200);

      Dxl.writeWord(motor_no, GOAL_TORQUE, 200);

    }

    void readfromkb()

    {

       if (SerialUSB.available() == 0)  // Wait here until input buffer has a character

      { menusel = SerialUSB.read();

      if (menusel>0) {  SerialUSB.print("Menu selection is = ");  SerialUSB.println(menusel, DEC); }

       while (SerialUSB.available()>0) junk = SerialUSB.read(); // clear the keyboard buffer

      }

    }
```

```
//----------------------------tork acma kapama

void torkkb()

{

  if (SerialUSB.available()==0){

SerialUSB.println("torque o/f");delay(4000);//Torku ac kapa

torkacik=SerialUSB.read();

if(torkacik=='o'){

 torque_on();

 SerialUSB.println("torque on");}

if (torkacik=='f'){

 torque_off();

 SerialUSB.println("torque off");}

 while (SerialUSB.available()>0) junk = SerialUSB.read();}

}

//---------------------------------programı durdurmak icin interrupt

void usbInterrupt(byte* buffer, byte nCount)

{

SerialUSB.print("buffer=");

SerialUSB.println(buffer[0]);

if (buffer[0]==107){

    goal_angle={0.0,0.0,0.0};

    for (i=0; i < MOTORCOUNT; i++)  {  motor_speed(i,7000); motor_goal(i,goal_angle[i]); }

    delay(5000);   for (i=0; i < MOTORCOUNT; i++)  {  motor_pos(i); }

    SerialUSB.println();

    while (SerialUSB.available()>0) junk = SerialUSB.read();
```

```
    loop();}

}

void hesap(){

    L1=sq(250*sin(theta) - 250*cos(theta)*sin(phi) + 125)+ sq(250*cos(psi)*cos(theta) - 250*cos(phi)*sin(psi) + 250*cos(psi)*sin(phi)*sin(theta) - 200) + sq(250*cos(phi)*cos(psi) + 250*cos(theta)*sin(psi) + 250*sin(phi)*sin(psi)*sin(theta) - 250) - 13125;

    l1=250*cos(phi)*sin(psi) - 250*cos(psi)*cos(theta) - 250*cos(psi)*sin(phi)*sin(theta) + 200;

    l2=250 - 250*cos(theta)*sin(psi) - 250*sin(phi)*sin(psi)*sin(theta) - 250*cos(phi)*cos(psi);

    l3=250*sin(theta) - 250*cos(theta)*sin(phi) + 125;

    M1=25000*sin(theta) - 25000*cos(theta)*sin(phi) + 12500;

    N1=2*50*((cos(3.14159)*l1)+(sin(3.14159)*l2));  // 50=a servo bağlantı uzunlugu

    U1=sq(M1)+sq(N1);

    J1=N1/M1;

    A=(asin(L1/sqrt(U1))-atan(J1))*57.32;

    AA=(1433*asin(((250*cos(psi)*sin(phi)*sin(theta) - 250*cos(phi)*sin(psi) + 50)*(250*cos(psi)*sin(phi)*sin(theta) - 250*cos(phi)*sin(psi) + 50) + (250*cos(phi)*cos(psi) + 250*sin(phi)*sin(psi)*sin(theta) - 250)*(250*cos(phi)*cos(psi) + 250*sin(phi)*sin(psi)*sin(theta) - 250) + (250*cos(theta)*sin(phi) + 125)*(250*cos(theta)*sin(phi) + 125) - 13125)/(2500*sqrt((25000*cos(psi)*sin(phi)*sin(theta) - 25000*cos(phi)*sin(psi) + 5000)*(25000*cos(psi)*sin(phi)*sin(theta) - 25000*cos(phi)*sin(psi) + 5000)/6250000 + (25000*cos(theta)*sin(phi) + 12500)*(25000*cos(theta)*sin(phi) + 12500)/6250000))))/25 - (1433*atan((25000*cos(psi)*sin(phi)*sin(theta) - 25000*cos(phi)*sin(psi) + 5000)/(25000*cos(theta)*sin(phi) + 12500)))/25;

    AAA=-1*(1433*atan((25000*cos(psi)*cos(theta) + 25000*cos(phi)*sin(psi) - 25000*cos(psi)*sin(phi)*sin(theta) - 20000)/(25000*sin(theta) + 25000*cos(theta)*sin(phi) - 12500)))/25 - (1433*asin(((250*sin(theta) + 250*cos(theta)*sin(phi) - 125)*(250*sin(theta) + 250*cos(theta)*sin(phi) - 125) + (250*cos(psi)*cos(theta) + 250*cos(phi)*sin(psi) - 250*cos(psi)*sin(phi)*sin(theta) - 200)*(250*cos(psi)*cos(theta) + 250*cos(phi)*sin(psi) - 250*cos(psi)*sin(phi)*sin(theta) - 200) + (250*cos(phi)*cos(psi) - 250*cos(theta)*sin(psi) + 250*sin(phi)*sin(psi)*sin(theta) - 250)*(250*cos(phi)*cos(psi) - 250*cos(theta)*sin(psi) + 250*sin(phi)*sin(psi)*sin(theta) - 250) - 13125)/(2500*sqrt((25000*sin(theta) + 25000*cos(theta)*sin(phi) - 12500)*(25000*sin(theta) + 25000*cos(theta)*sin(phi) - 12500)/6250000 + (25000*cos(psi)*cos(theta) + 25000*cos(phi)*sin(psi) -
```

```
25000*cos(psi)*sin(phi)*sin(theta) - 20000)*(25000*cos(psi)*cos(theta) + 25000*cos(phi)*sin(psi) - 25000*cos(psi)*sin(phi)*sin(theta) - 20000)/6250000))))/25;

    SerialUSB.print("A1=");

    SerialUSB.print(A);

    SerialUSB.print("A2=");

    SerialUSB.print(AA);

    SerialUSB.print("A3=");

    SerialUSB.print(AAA);

    delay(35);

    }
```