DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DESIGN AND ANALYSIS OF A HEXAPOD WITH UPUR STRUCTURE

by
Okan TONGA

September, 2010
İZMİR

DESIGN AND ANALYSIS OF A HEXAPOD WITH UPUR STRUCTURE

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
in Mechanical Engineering, Machine Theory and Dynamics Program

by
Okan TONGA

September, 2010
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"DESIGN AND ANALYSIS OF A HEXAPOD WITH UPUR STRUCTURE"** completed by **OKAN TONGA** under supervision of **PROF. DR. HİRA KARAGÜLLE** 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.

Prof. Dr. Hira KARAGÜLLE

Supervisor

Assoc. Prof. Zeki KIRAL

(Jury Member)

Asst. Prof. Dr. Ahmet ÖZKURT

(Jury Member)

Prof.Dr. Mustafa SABUNCU

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENT

I am greatly indebted to my thesis supervisor Prof. Dr. Hira KARAGÜLLE for kindly providing guidance and interest throughout the development of this study.

I would like to thank to Research Assistant Dr. Murat AKDAĞ, for his help, with valuable suggestions and discussions that he has provided me during the research.

I am also thankful to all my friends for their valuable help throughout this study.

This thesis is dedicated to my family. I would like to thank them for their support, patience and sacrifice.

Okan TONGA

DESIGN AND ANALYSIS OF A HEXAPOD WITH UPUR STRUCTURE

ABSTRACT

In this study, a hexapod with universal- prismatic- universal- revolute (UPUR) structure is designed and analyzed. Parallel manipulators have high stiffness and accuracy, and they are used in medical, aviation and astronomy. Universal joints, parallel motion actuators (linear actuators) and revolute joints (rolling element bearings) are available in the market with competitive prices and thus, hexapods with UPUR structure can be produced with low costs.

SolidWorks is used for modeling and assembly, CosmosMotion is used for rigid body dynamics and ANSYS is used for the finite element rigidity analysis. A program developed in MATLAB is also used for the rigid body dynamics analysis. A control system is developed for the system having linear actuators with step motors. PC-based motion control is used.

It is observed that MATLAB results are in agreement with CosmosMotion results. The natural frequencies and the static deflections of the system are given for various positions. Hexapod motions are analyzed considering kinematic, kinetic and rigidity limits.

Keywords: Hexapod, simulation, parallel manipulators, CosmosMotion, ANSYS, SolidWorks.

UPUR BAĞLANTILI HEGZAPOD TASARIMI VE ANALİZİ

ÖZ

Bu çalışmada, evrensel- prizmatik- üniversal- döner (UPUR) bağlantılı bir hegzapod tasarlamış ve analiz edilmiştir. Paralel manipülatörler yüksek rijitlik ve hassasiyete sahiptirler ve tıp, havacılık ve astronomide kullanılmaktadır. Üniversal mafsallar, prizmatik hareket aktüatörleri (lineer aktüatörler) ve döner eklemler (dönme elemanı rulmanlar) piyasada uygun fiyatlarla mevcuttur ve bu nedenle UPUR bağlantılı hegzapodlar düşük maliyetle üretilebilir.

Modelleme ve montaj için SolidWorks, rijit cisim dinamiği için CosmosMotion ve sonlu elemanlarla mukavemet analizi için ANSYS kullanıldı. Rijit cisim dinamiği analizi için MATLAB'da geliştirilen bir program kullanıldı. Adım motorlu lineer aktüatörlere sahip sistem için bir kontrol sistemi geliştirildi. PC tabanlı hareket kontrolü kullanıldı.

MATLAB sonuçlarının CosmosMotion sonuçlarıyla tutarlılık gösterdiği gözlendi. Sistemin doğal frekansları ve statik sapmaları değişik pozisyonlar için verilmiştir. Hegzapod hareketleri kinematik, kinetik ve mukavemet sınırları göz önüne alınarak analiz edildi.

Anahtar sözcükler: Hegzapod, simülasyon, paralel manipülatörler, CosmosMotion, ANSYS, SolidWorks.

CONTENTS

Page

CHAPTER ONE
INTRODUCTION

The developments of six-dof parallel manipulators were firstly used by Gough & Whitehall (1962) as a tire-testing machine. The testing machine was consisted of a six-linear jack system. Stewart (1965) designed a 6-DOF parallel manipulator for the usage of flight simulator. A systematic study of kinematic structures of parallel manipulators was made by Hunt (1983). Since then, parallel manipulators have been taking interests of researchers.

Wendlandt & Sastry (1994) examined a Stewart platform for endoscopy in order to design and control. The researchers aimed the platform to follow a circled path. McInroy (1999) investigated controlling of hexapods by dynamic modelling. Base accelerations were included and the model was experimentally verified. A comprehensive literature review study was made by Dasgupta & Mruthyunjaya (2000). The study contained all topics about hexapods and researches related to these topics made until the publication year. An important study about parallel manipulators was the investigating new kinematic structures for parallel manipulators (Gao, Li, Zhao, Jin & Zhao, 2002). In that study, researchers developed new types of composite links. Joint types related to degree of freedom were presented. Alizade & Bayram (2004) classified parallel manipulators according to their platform types and connections between them. In another study, active vibration control of a hexapod was achieved with sensitivity weightened linear quadratic Gaussian (SWLQG) controller (Hauge & Campbell, 2004). Inverse kinematics, forward kinematics, error analysis and workspace evaluation were examined in the paper of Jelenkovic, Jakobovic & Budin (2004). Drive singularities of parallel manipulators were investigated by Ider (2005). Kim, Cho & Lee successfully controlled a 6-DOF parallel manipulator, namely hexapod, with respect to robust nonlinear control. The researchers considered friction of each actuator because friction may degrade control performance. In this manner, in order to determine friction values, a friction estimator was used.

1.1 Parallel Manipulators and Hexapod

A parallel manipulator typically consists of a moving platform that is connected to a fixed base by several limbs or legs. Typically, the number of limbs is equal to the number of degrees of freedom such that every limb is controlled by one actuator and all the actuators can be mounted at or near the fixed base (Tsai, 1999).

A six-degree of freedom parallel mechanism is named as a hexapod. A hexapod robot has been first proposed by Stewart (D. Stewart, 1965) as a flight simulator. A schematic view of a hexapod is given in Figure 1.1.

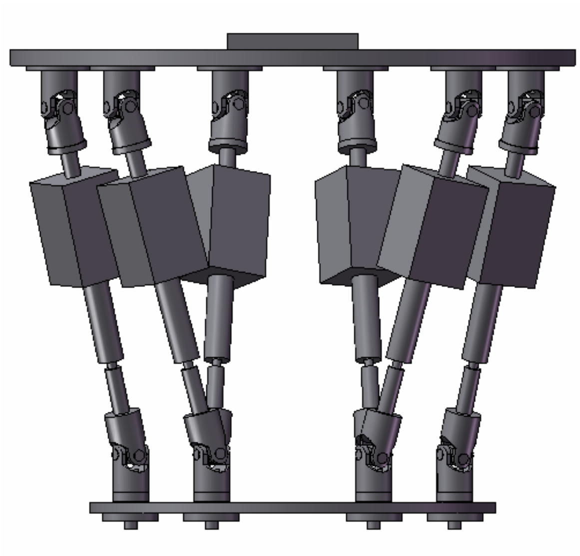


Figure 1.1 A schematic view of a hexapod.

Hexapods due to the advantages mentioned above can be utilized for different purposes such as robotic surgery (Rogers, 2007), flight simulation (Dongsu, 2007), automation in civil engineering (Kim, 2008), active vibration control (Preumont, 2002). Hexapods are widely used in micro-positioning due to their high stiffness characteristics (Arumugam, 2004). Main safety requirements and features for robotic surgery by a hexapod robot are explained to prevent any uncontrolled or unwanted movements in (Laible, 2004).

1.2 Advantages and Disadvantages of Hexapods

The advantages of hexapods in general are;

• Excellent load/weight ratios.

• High stiffness, as the kinematics chains (limbs) are sharing loads and in many cases the links can be designed such that they are exposed to tensile and compressive loads only. This high stiffness insures that the deformations of the links will be minimal and this feature greatly contributes to the high positioning accuracy of the manipulator.

• Low inertia, because most of the actuators are attached to the base, and thus no heavy mass need to be moved.

• The position of the end-effector is not sensitive to the error on the articular sensors. Higher accuracy due to non-cumulative joint error.

• Many different designs of parallel manipulators are possible.

• The mechanisms are of low cost since most of the components are standard.

• Usually, all actuators can be located on the fixed platform.

• Work-space is easily accessible.

• The possibility of using these mechanisms as a 6-component force-sensor. Indeed it can be shown that the measurement of the traction-compression stress in the links enables to calculate the forces and torques acting on the mobile platform.

On the other hand, the disadvantages of hexapods in general are;

• Limited useful work-space compared to the mechanism size.

• Limited dexterity.

• Potential mechanical-design difficulty.

• Mechanism assembly has to be done with care.

• Time-consuming calibration might be necessary.

• Complex kinematical computations and control algorithms

• Length of actuator

CHAPTER TWO
INTEGRATED DESIGN ANALYSIS AND CONTROL OF A HEXAPOD

2.1 Flow Chart

The flow chart of the integrated design process developed in this study and applied to a hexapod robot is shown in Fig. 2.1. The flow chart is explained below without the details of the software commands. The emphasis in this paper is the flow chart of the design process. The details of the software to achieve the tasks in the flow chart can be found in the manuals.

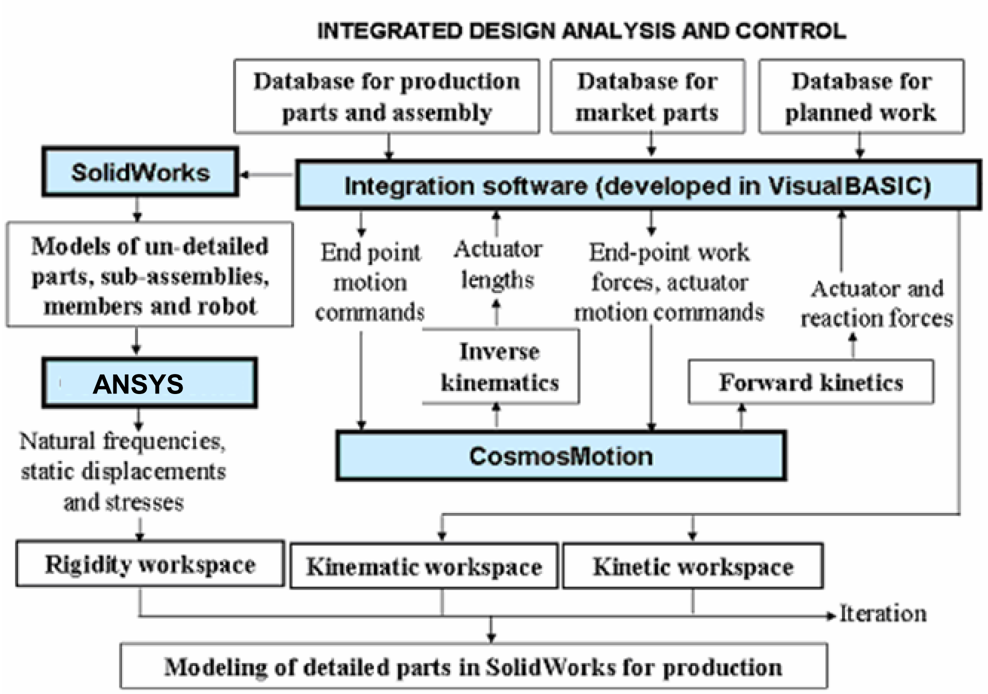


Figure 2.1 Flow chart of mechatronic design process

2.2 Parametric Modelling

There are production parts and market parts. The production parts are designed, whereas the market parts are selected. The database for production parts and assembly consists of the information on dimensions and angles which are necessary for modeling. The database for market parts consist of the information on technical specifications such as actuation and load limits. The database for planned work consists of the information on the end-point motion inputs.

The modeling software developed in VisualBASIC reads the data from the databases and models the parts, sub-assemblies, members and the robot automatically, using the Application Programming Interface (API) capabilities of SolidWorks. SolidWorks + CosmosMotion package and ANSYS are used in this study. The robot model developed in ANSYS and SolidWorks. SolidWorks is used automatically in CosmosMotion. Solid modeling of parts and assembly can also be done using the Graphical User Interface (GUI) capabilities of SolidWorks without the modeling software.

2.3 Kinematic and Kinetic Workspace Evaluation

The integration software (IS) developed in VisualBASIC has “Inverse Kinematics”, “Kinematic Workspace Evaluation”, “Forward Kinetics”, “Kinetic Workspace Evaluation” and “Rigidity Workspace Evaluation” options. IS reads the database for the planned work and sends the end-point motion commands to CosmosMotion using API, when “Inverse Kinematics” option is clicked. The actuator motions are assigned as free at this step and the actuator displacements are found. The time histories of the actuator displacements are transferred back to IS. The results are evaluated whether the motion is possible considering the kinematic limits (displacement and velocity limits) of the actuators, when “Kinematic Workspace Evaluation” option is clicked.

The end-point motion is set to free and the actuator motion commands together with the inputs for the end-point work forces are sent to CosmosMotion, when “Forward Kinetics” option is clicked. The time histories of the actuator motion generator forces and reaction forces are found. The results are evaluated whether the kinetic limits (the actuator motion generation and joint force limits) are exceeded.

After the rigid body dynamics analyses are done, the finite element (FE) natural frequency and static displacement and strength analyses are performed in ANSYS. ANSYS does not have API capabilities in the software package used in this study and these analyses are performed using GUI. "Rigidity Workspace Evaluation" option in IS is inserted as a reminder. The lowest natural frequency (f_{min}), maximum displacement (u_{max}) and maximum VonMises stress (s_{max}) are evaluated whether the rigidity limits of the robot structure are exceeded. The higher values of f_{min}, and the lower values of u_{max} and s_{max} indicate higher static and dynamic rigidity. The precision of the work performed by the robot is related to these rigidity values.

2.4 Iteration and Final Design

The database for planned works are arranged for various works and simulations are performed for the kinematic, kinetic and rigidity workspace evaluations. The databases for modeling are changed and the simulations and evaluations are repeated until a final design is achieved.

The details such as holes, threads, tolerances are inserted to the un-detailed parts in SolidWorks after the final design, using GUI. Market parts usually have the detailed models, which can be downloaded from the manufacturer websites. Un-detailed models of market parts are produced by the designer. Un-detailed models must have equivalent rigid body dynamic properties and approximately the same rigidity in FE analyses.

CHAPTER THREE

DESIGN OF HEXAPOD

3.1 Modeling of Members and Hexapod Robot

The model of the hexapod with un-detailed parts designed with the process given in chapter 2 is shown in Fig. 3.1. The model of the hexapod robot assembly consists of motion subassemblies which are called as members. Each member consists of models of parts or subassemblies which have a common rigid body motion. Members are connected by joints. The models of the members with un-detailed parts are shown in Fig. 3.2.

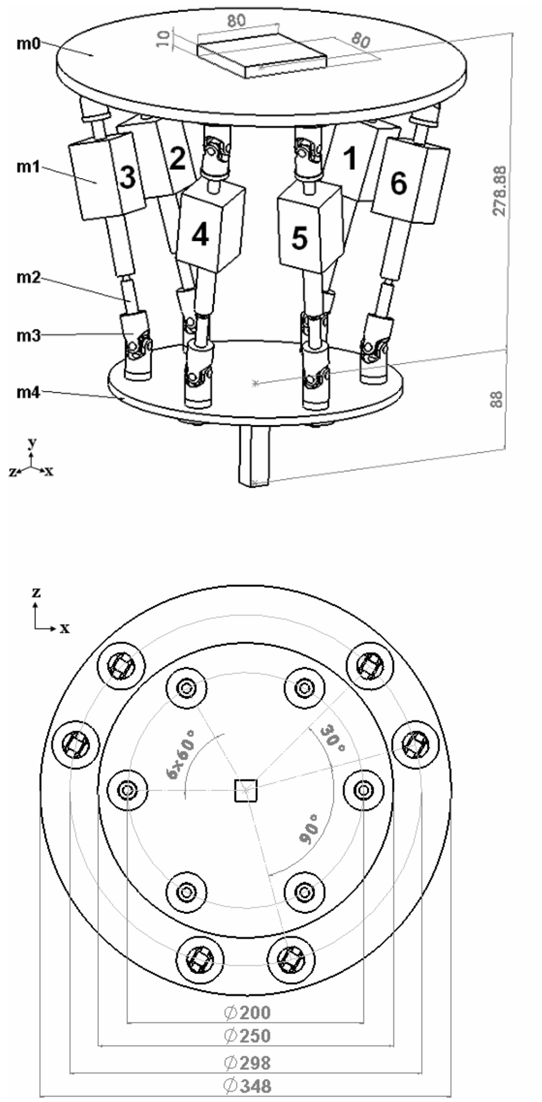


Figure 3.1 Un-detailed model of hexapod

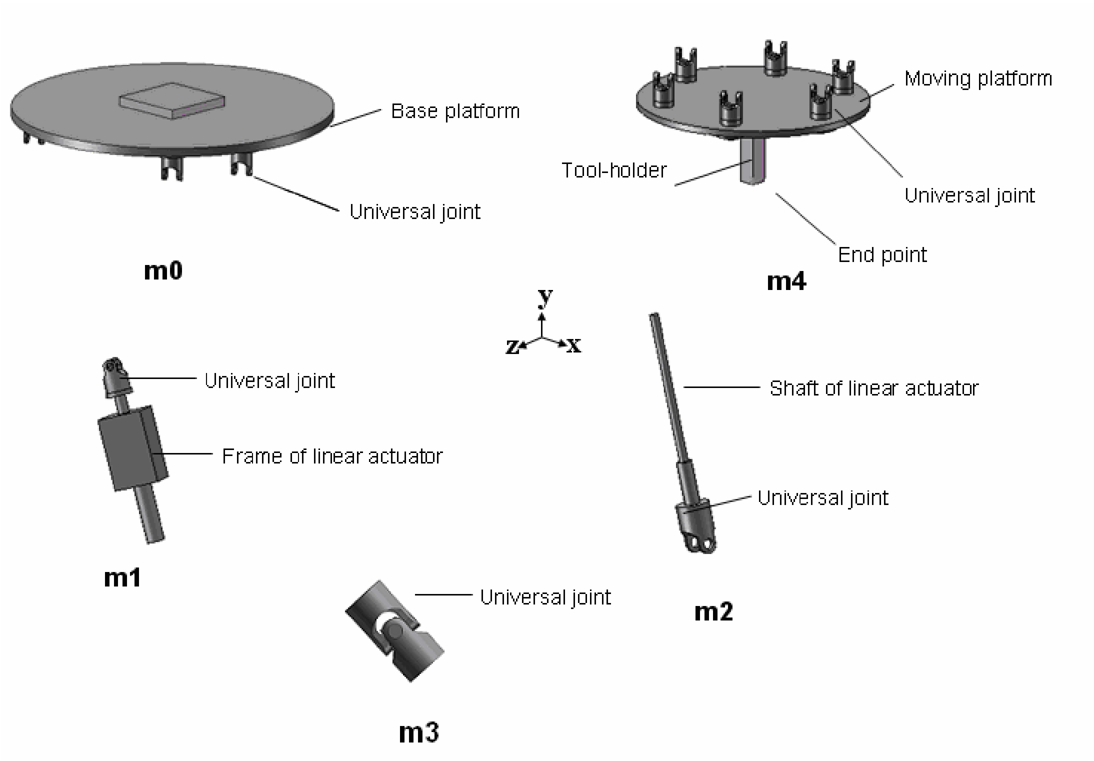


Figure 3.2 Models of members with un-detailed parts

The members of the hexapod are titled as m0, m1, m2, m3 and m4. There are six pieces of m1 and m2, also twelve pieces of m3. The m1-members are connected to the m0-member by universal joints. The m2-members are connected to the m4-member by universal joints. The m1-members and m2-members are connected by translational joints through their axes. The diameter of the base and moving platforms are 348 and 250mm, respectively. Their thicknesses are 10 and 7mm, respectively. The centers of universal joints are at a distance of 33.3mm to the closest platform surfaces. The distance between the centers of platforms is 278.88mm when the actuators are fully retracted. The materials selected for all the parts are steel (AISI 1020). The weight of the hexapod model is 18.375kg.

The linear actuators (HSI Company Model 43K4U-05-032ENG), universal joints (U5-13 161 type) and bearings (FAG 608-2Z) are market parts. The actuators have a stroke of 50.8mm and a nominal actuation capacity of 225 N. They have 1.5 micron/pulse precision. The universal joint has a maximum torque capacity of 22 Nm. The bearing has load capacity of 275 N.

First, the un-detailed parts are modeled using the input database. Then the parts are inserted to the assembly file for the member to be modeled. The parts are located to their positions. No mate commands are used for the modeling of the members. After modeling all the members, the assembly files of the members are inserted to the assembly file of the hexapod robot. The members are oriented and located to their approximate positions. The mates are defined between the members for joints. The points on the platforms are mated to the center points on the moving parts of the universal joints. The axes on the frames of the linear actuators are mated to the axes on the universal shafts of the linear actuators. CosmosMotion defines universal and translational joints automatically after the ground and moving members are defined. The translational joints on the linear actuator parts have 2 degrees of freedom titled as TranslateZ and RotateZ. The RotateZ components of the translational joints are set to zero.

3.2 Integration Software (IS)

3.2.1 Work Database

The position of the hexapod is defined by the position of the moving platform (the m4 member). x, y and z are the global coordinates and x_b, y_b and z_b are the local coordinates attached to the platform. The orientation of the platform is a set of Euler angles φ, θ, ψ, which are the successive rotations around x_b, y_b, z_b axes, respectively. The position of the platform is defined by the generalized coordinate vector given below.

$$q = [x_1, y_1, z_1, \varphi, \theta, \psi]^T \tag{1}$$

The generalized coordinate is $x_1=0$, $y_1=278.88$, $z_1=0$, $\varphi=0$, $\theta=0$, $\psi=0$ when the linear actuators are fully retracted. This position is called the assembly position. The distances are given in mm, and the angles are given in degrees unless otherwise is stated.

The work force is defined as f_{xb}, f_{yb}, f_{zb} with respect to the 'tool holder' part of 'm4'. The work force is acting on the end-point shown in Fig. 3.2. Different work database formats could created for different purposes. Three types of motion information can be given.

First type (w-type) given as an example;

6, 0.2, 0.1, w, 2, -10, 3, 4, -1, 1.8, f, 10, 100, -20

Here, $t_d = 6$s, $r_1 = 0.2$, $r_2 = 0.1$, $x_i = 2$mm, $y_i = -10$mm, $z_i = 3$mm, $\varphi_i = 4^\circ$, $\theta_i = -1^\circ$, $\psi_i = 1.8^\circ$, $f_{xb} = 10$ N, $f_{yb} = 100$ N, $f_{zb} = -20$ N. t_d is the duration of the motion. r_1 and r_2 are the percentages of t_d for the acceleration and deceleration times, respectively. x_i, y_i, z_i are the incremental translational motions in the x, y, z directions respectively. φ_i, θ_i, ψ_i are the incremental Euler angles. Characters following 'f' are indicated forces acting on face of 'tool holder'.

Second type (r-type) given as an example;

4, 0.14, 0.12, r, 3, -5, -2.4, f, -20, 60, 10

Here, $t_d = 4$ s, $r_1= 0.14$, $r_2 =0.12$. $x_{bi} = 3$mm, $y_{bi} = -5$mm, $z_{bi}= -2.4$mm, $f_{xb} = -20$ N, $f_{yb} = 60$ N, $f_{zb} = 10$ N. No rotational displacements are given in the r-type. x_{bi}, y_{bi} and z_{bi} are the relative incremental motion of the platform from its previous position. x_{bi}, y_{bi}, z_{bi} are the incremental translational motions in the x_b, y_b, z_b directions respectively. Characters following 'f' are indicated same values as w type.

Third type (wait-type) given as an example;

3, wait, f, 8, 90, -22

Here, the hexapod is required to stand without motion for $t_d = 3$ s, and $f_{xb} = 8$ N, $f_{yb} = 90$ N, $f_{zb} = -22$ N. Characters following 'f' are indicated same values as w type.

All the incremental motions follow the trapezoidal velocity profile given in Fig.3.3.

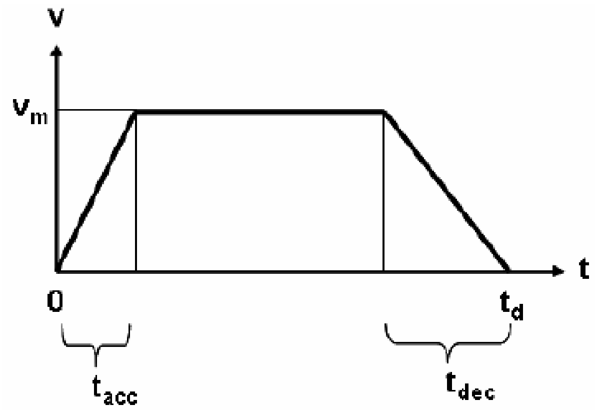


Figure 3.3 Trapezoid Velocity profile

In figure t_{acc} is acceleration time, t_{dec} is deceleration time ($t_{acc} = r_{acc} * t_d$, $t_{dec} = r_{dec} * t_d$). v_m is maximum velocity. Area under the velocity - time curve indicates translation or rotation in td time (d_0). Maximum velocity is founded with formula: $v_m = d_0 / (t_d - 0.5 * t_{acc} - 0.5 * t_{dec})$. Software and hardware for controlling actuators according profile are commercial and could supply from market.

3.2.2 Inverse Kinematics

The following vector-loop equation is used for the inverse kinematics of the hexapod (H.B.Guo & H.R. Li 2006).

$$b_n + a_n = u_n + p_n \ (n = 1\text{-}6) \quad (2)$$

The vector-loop for n=3 is shown in Fig. 3.4. The end points of b_n and p_n are located at the centers of the universal joints on the base and moving platforms, respectively, and they are known from the design. Knowing u_n as the input, the limb vector, a_n can be found from Eq. 2.

A motion object is assigned to the m4-member (moving platform) in CosmosMotion. The motion object has six degrees of freedom titled as TranslateX, TranslateY, TranslateZ, RotateX, RotateY, RotateZ, which correspond to x_i, y_i, z_i,

φ_i, θ_i, ψ_i, respectively. Six components of the motion object are set to the displacement mode.

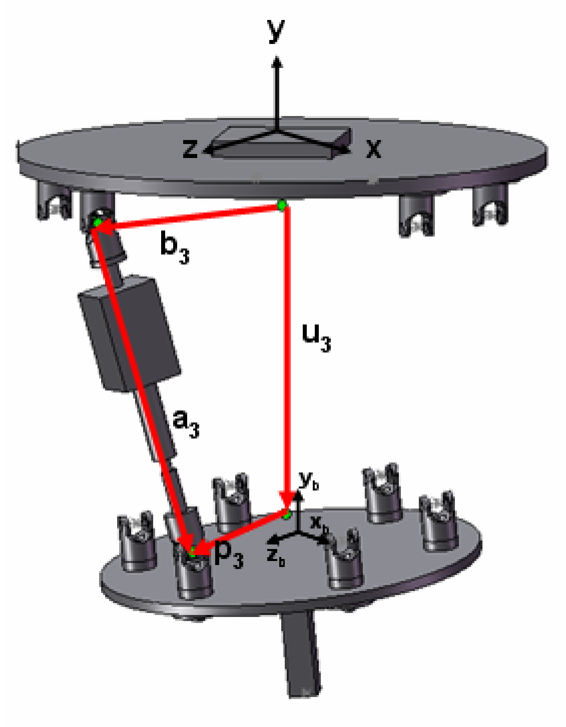


Figure 3.4 Vector loop for the hexapod

Angular displacement objects for the universal joint angles, plot objects for the translational joint displacements, and plot objects for the angular displacement objects are defined after running the simulation in CosmosMotion. The plot objects give the time histories, which are exported to text files. The results in these text files are read by IS and evaluated whether the limits are exceeded. The limits for the translational displacement, velocity and universal joint angles are 50.8mm, 2.28mm/s, and 45°, respectively. The translational velocity limit is given for the average actuator load, and it can be increased depending on the load-velocity diagram of the actuator. This control can be done after the forward kinetic analysis.

The algorithm for the kinematic analysis given in (H.B.Guo & H.R. Li 2006) has been implemented by using MATLAB to obtain analytical results and the analytical results are compared with the CosmosMotion results in Table 3.1 and 3.2. It is observed from Table 3.1 and 3.2 that the results are in agreement.

Case 1

The instantaneous input values are;

x_i = 5.9, y_i = -30.5, z_i = -4.9, φ_i = -9.9, θ_i =7.9, ψ_i = -6.9

x'_i = 0.9, y'_i = -4.7, z'_i = 0.7, φ'_i = -1.5, θ'_i =1.2, ψ'_i = -1

x''_i = - 4.4, y''_i = 23.5, z''_i = -3.7, φ''_i = 7.4, θ''_i = -5.9, ψ''_i = 5.1

Table 3.1 CosmosMotion and analytical solution results for inverse kinematic analysis for case 1

Actuator	1	2	3	4	5	6
CosmosMotion	39.3720	17.5714	15.4179	19.9342	45.0914	50.5632
Analytical Solution	39.3719	17.5714	15.4179	19.9342	45.0914	50.5631

Results are given as the incremental actuator displacements. Translational displacements, velocities and accelerations are in mm, mm/s and mm/s^2, respectively. Rotational displacements, velocities and accelerations are in degree, deg/s and deg/s^2, respectively. The incremental inputs and results are from the assembly position.

Case 2

The instantaneous input values are;

x_i =3.9, y_i = -21.5, z_i = 2.9, φ_i = -8.9, θ_i = -3.9, ψ_i = 7.9

x'_i =0.7, y'_i = -1, z'_i = 0.6, φ'_i = -1.3, θ'_i =1.1, ψ'_i = -1.2

x''_i = - 1.4, y''_i = 2.5, z''_i = -1.9, φ''_i = 3.2, θ''_i = -2.5, ψ''_i = 1.3

Table 3.2 CosmosMotion and analytical solution results for inverse kinematic analysis for case 2

Actuator	1	2	3	4	5	6
CosmosMotion	40.7380	37.0003	14.9090	3.5292	7.4103	27.9183
Analytical Solution	40.7380	37.0003	14.9090	3.5292	7.4103	27.9183

Results are given as the incremental actuator displacements. Translational displacements, velocities and accelerations are in mm, mm/s and mm/s^2, respectively. Rotational displacements, velocities and accelerations are in degree, deg/s and deg/s^2, respectively. The incremental inputs and results are from the assembly position.

3.2.3 Forward Kinetics

The dynamic equations of the hexapod are written in the general form as follows (H.B.Guo & H.R. Li 2006).

$$M(q)q''+C(q,q')q'+G(q) = H(q)F \tag{3}$$

Where q is given in Eq. 1. Note that the generalized velocity of the platform is not defined as the time derivative of q, but as (H.B.Guo & H.R. Li 2006).

$$q' =[x_1', y_1', z_1',\omega_1, \omega_2, \omega_3]^T \tag{4}$$

$\omega = [\omega_1, \omega_2, \omega_3]^T$ denotes the angular velocity of the platform with respect to (w.r.t) the global coordinates.

Note that the generalized acceleration of the platform is defined as

$$q'' = [x_1'', y_1'', z_1'', \omega_1', \omega_2', \omega_3']^T \tag{5}$$

f_n is the driving force for the n-th actuator and $F = [f_1, f_2, f_3, f_4, f_5, f_6]^T$ in Eq. 3. Refer to (H.B.Guo & H.R. Li 2006) for the equations for M(q), C(q), G(q) and H(q). The actuator forces are found by solving Eq. 3 after the inverse kinematic analysis. The inertia properties of the members of the designed hexapod are given as follows.

m_p = 3482.1 gr, m_b = 963.98 gr, m_t = 104.91 gr.

$$I_{pg} = \begin{bmatrix} 140.96 & 0 & 0 \\ 0 & 267.73 & 0 \\ 0 & 0 & 140.99 \end{bmatrix} x10^5 \,(gr*mm^2)$$

$$I_{tg} = \begin{bmatrix} 1.56 & 0 & 0 \\ 0 & 0.0473 & 0 \\ 0 & 0 & 1.57 \end{bmatrix} x10^5 \,(gr*mm^2)$$

$$I_{bg} = \begin{bmatrix} 9.73 & 0 & 0 \\ 0 & 2.26 & 0 \\ 0 & 0 & 9.74 \end{bmatrix} x10^5 \, (gr*mm^2)$$

m_p is the mass of the m4-member (the moving platform), m_b is the mass of the m1-member (the base-part of the actuator), m_t is the mass of the m2-member (the moving-part of the actuator). The origins of the m1-member and m2-member are coincident with the centers of the universal joints. The position vectors of the centers of gravities (c.o.g) of the members w.r.t. the member coordinates are given by $[0, -1.129, 0]^T$ for the m4-member, $[0, -75.74, 0]^T$ for the m1-member, $[0, 34.42, 0]^T$ for the m2-member. I_{pg}, I_{bg} and I_{tg} are the moments of inertias of the m4-member, m1- member and m2-member, respectively. The inertias are taken at the c.o.g.'s and aligned with the member coordinate systems.

The position vector of the end-point where the work force is acting is given as $[0, -88, 0]^T$ w.r.t. the member coordinates.

In this work, CosmosMotion is used for the kinetic analysis as explained below.

Action force objects in the x_b, y_b and z_b directions are defined at the acting face of the tool holder part of m4 member. All the six components of the motion object for the m4-member are set to the free mode and the TranslateZ motion components for all the 6 translational joints are set to the displacement mode at this step. The displacements of the linear actuators for each step are read from the exported text files in the inverse kinematic analysis by IS. The time histories of the translational displacements of the actuators are created so that they follow the velocity profile given in Fig. 3.3, and they are assigned to the TranslateZ motion components of the translational joints. The values of the work forces for the successive motions are read from the planned work database file and they are assigned to the corresponding force objects.

Plot objects for the magnitudes of the translational motion generator forces for the translational joints are defined. The time histories of the actuating forces are exported to text files. The results in these text files are read by IS and evaluated whether the limits are exceeded. The limit for the actuation forces is determined by the force-velocity curve. This curve is approximately linear for the actuators used in this study and the coordinates of two points on this line are (0 N, 2. 9 mm/s) and (225 N, 1.5 mm/s). The force determined by this line at a particular speed sets the limit force. The limit reaction force for the universal angles used in this study is higher (540 N), and thus the limit control for the actuators is adequate.

The algorithm for the kinetic analysis given in (H.B.Guo & H.R. Li 2006) to solve Eq. 3 has been implemented by using MATLAB to obtain analytical results. The analytical results are compared with the CosmosMotion results in Table 3.3 and 3.4. It is observed from Table 3.3 and 3.4 that the results are in agreement.

Table 3.3 CosmosMotion and analytical solution results for kinetic analysis for case 1

Actuator	1	2	3	4	5	6
CosmosMotion	126.6324	109.5473	69.1185	166.1269	76.2285	3.3636
Analytical Solution	126.6104	109.5106	69.2067	166.1674	76.2532	3.3188

Results are given as the actuator forces. The instantaneous payload is f_{xb}*=60,* f_{yb}*= 100,* f_{zb}*= -20. The force values are in N. The kinetic analysis has been done after obtaining the kinematic analysis given in Table 3.1. The gravity in the –y direction has been included.*

Table 3.4 CosmosMotion and analytical solution results for kinetic analysis for case 2

Actuator	1	2	3	4	5	6
CosmosMotion	85.8200	80.7062	36.8115	109.7221	58.1596	21.4297
Analytical Solution	85.8035	80.6771	36.8458	109.7486	58.1694	21.4255

Results are given as the actuator forces. The instantaneous payload is f_{xb}*=50,* f_{yb}*= 80,* f_{zb}*= -30. The force values are in N. The kinetic analysis has been done after obtaining the kinematic analysis given in Table 3.2. The gravity in the –y direction has been included.*

3.3 Rigidity Analysis

The finite element (FE) analyses for the natural frequency and static displacement calculations are done in ANSYS. The planned work database file is organized and the hexapod is moved to an analysis position. After evaluating the limits for the kinetic and kinematic workspace, the FE studies are performed. The study type (frequency or static) is selected, the materials are assigned, the restraints are defined, and the FE model is created by meshing. The work forces are defined for the static analysis. Finally, the FE study is performed and the results are observed.

CHAPTER FOUR
HEXAPOD MOTION CONTROL SYSTEM

4.1 Hexapod Motion Control System

For controlling actuators it is used motion control cards from ADLINK company. 2 pieces of motion control card, PCI 8154 and PCI 8164 is plugged into PCI slots in PC and drivers installed. Each motion control card can drive 4 axes. Technical properties of motion control cards are given in Appendix-B. Schema of control system is shown in figure 4.1. Motion control cards and actuator drivers power inputs are 24 V DC, rotary encoder inputs are 5 V DC. Encoders are attached to actuators directly and output pulses are transmitted by terminal board. Linear actuator motors are step motors. Step motor drivers are type of 40105 from HSI company, with maximum current value of 2 A, full or half step drive. Connection list of electrical panel is in Appendix-C.

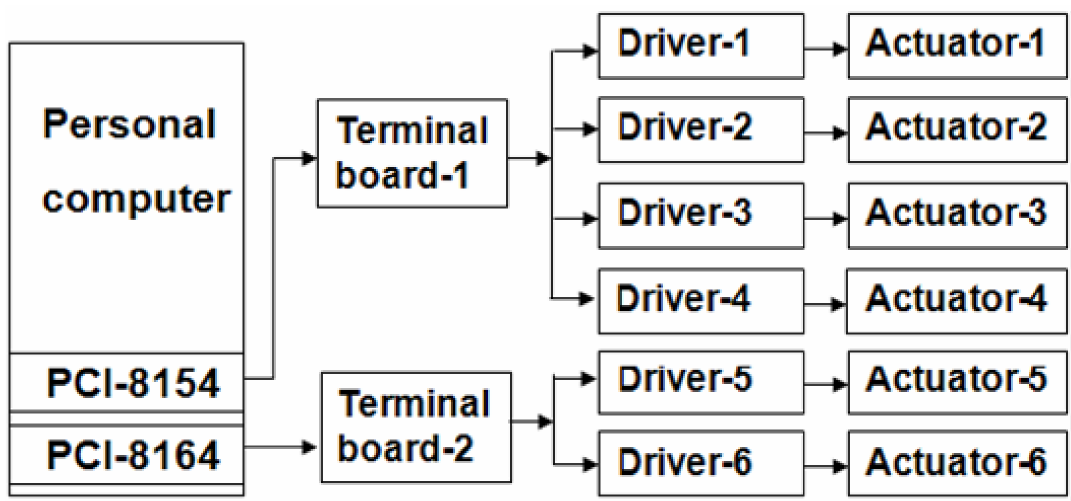


Figure 4.1 Control system schema of hexapod

In Figure 4.2, it is seen a detailed picture of electrical panel of drivers.

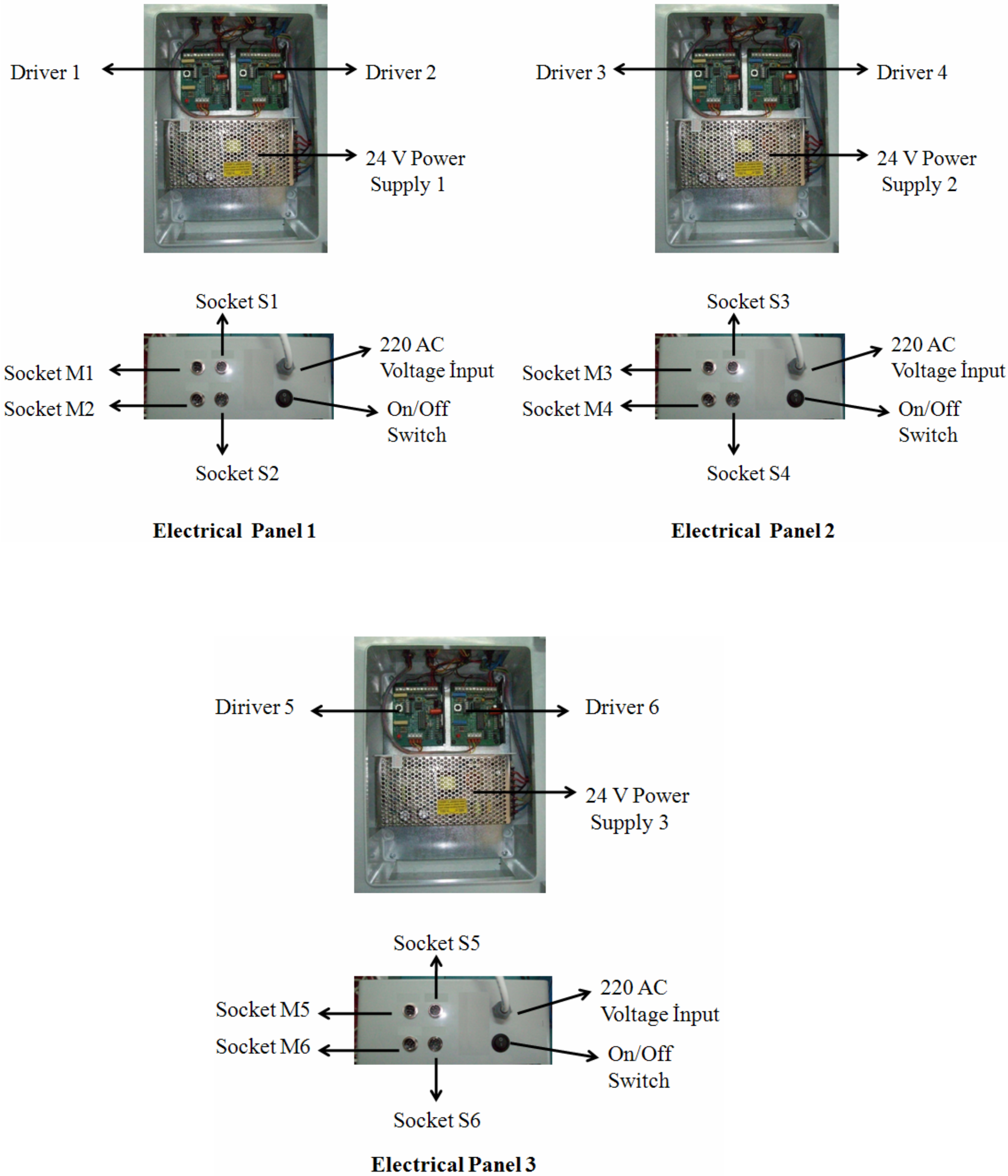


Figure 4.2 Electrical panel of drivers

In Figure 4.3, it is seen a detailed picture of electrical panel of terminals.

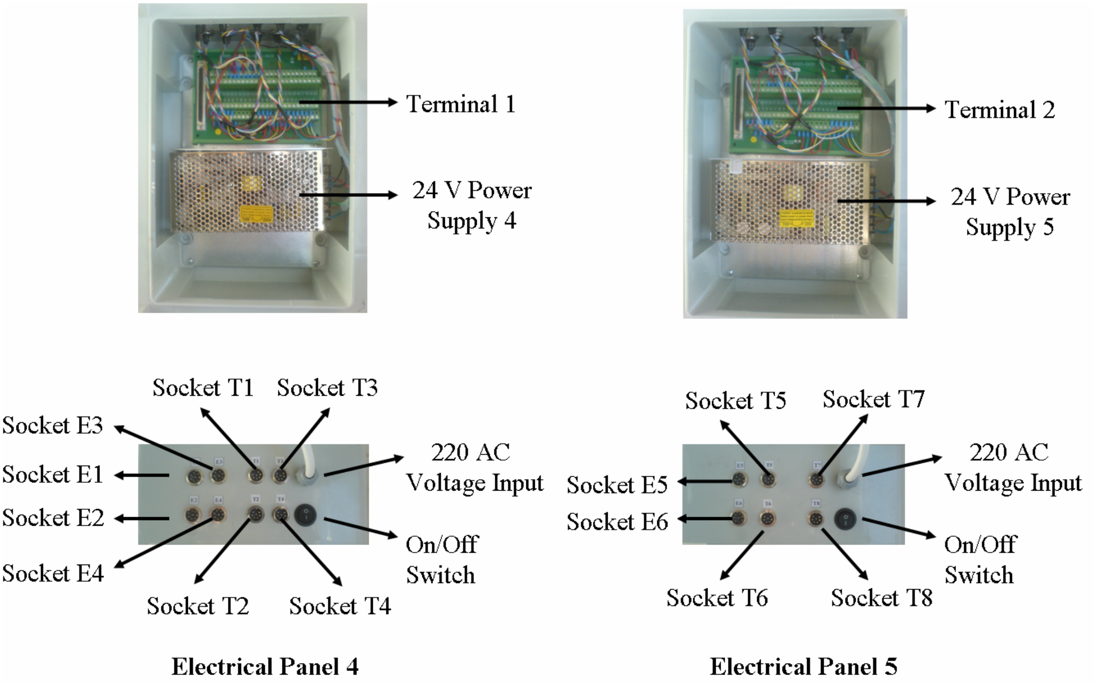


Figure 4.3 Electrical panel of terminals

In Figure 4.4, it is seen a detailed picture of step motor drivers

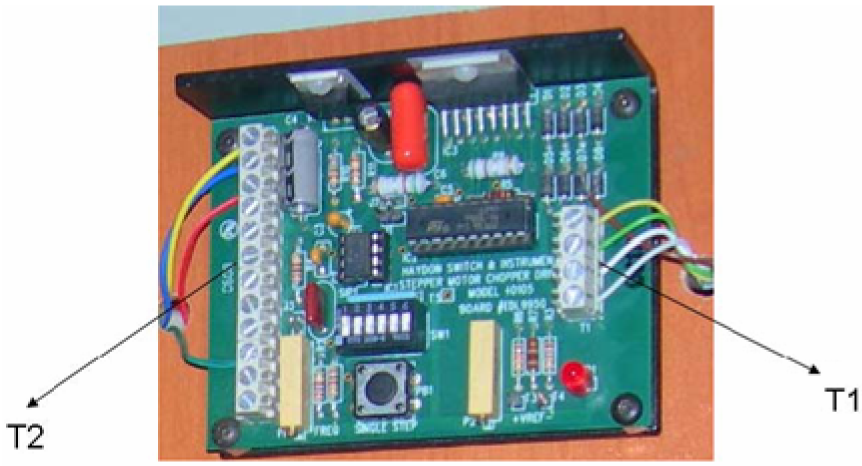


Figure 4.4 Step motor drivers

CHAPTER FIVE
SIMULATION RESULT OF HEXAPOD

5.1 Rigidity Analysis Result

The hexapod is moved to its final position by giving the incremental displacements x_i, y_i and z_i. Rigidity workspace analyses stress, displacement and modal analyses are done manually using ANSYS.

The modal analysis results are given for the assembly position (x_1=0, y_1=278.88, z_1=0, φ=0, θ =0, ψ =0) in Fig. 5.1. 1st mode is seen as shape of bending around x axis, 2nd mode is seen as shape of bending around z axis, 3rd mode is seen as shape of bending around y axis. Elongation along y axis mode shape is seen at very high natural frequency level, 1st and 2nd modes natural frequencies are close.

The number of elements is 30888, the number of nodes is 69820 for the hexapod at the assembly position.

The minimum natural frequencies (f_{min}) are given for the hexapod for various positions in Fig. 5.2. The static FE results are given for example work forces in Fig. 5.3. The values of f_{min} (minimum natural frequency), u_{max} (maximum static displacements) and s_{max} (maximum VonMises stress) for various positions and loadings are given in Table 5.1. Hexapods can be mounted in different positions (horizontal, vertical or combination). Therefore in static and frequency analyses gravity is not taken into calculations.

When gravity along –y direction are taken into calculation, the values of u_{max} (maximum static displacements) and s_{max} (maximum VonMises stress) for various positions and loadings are given in Table 5.2

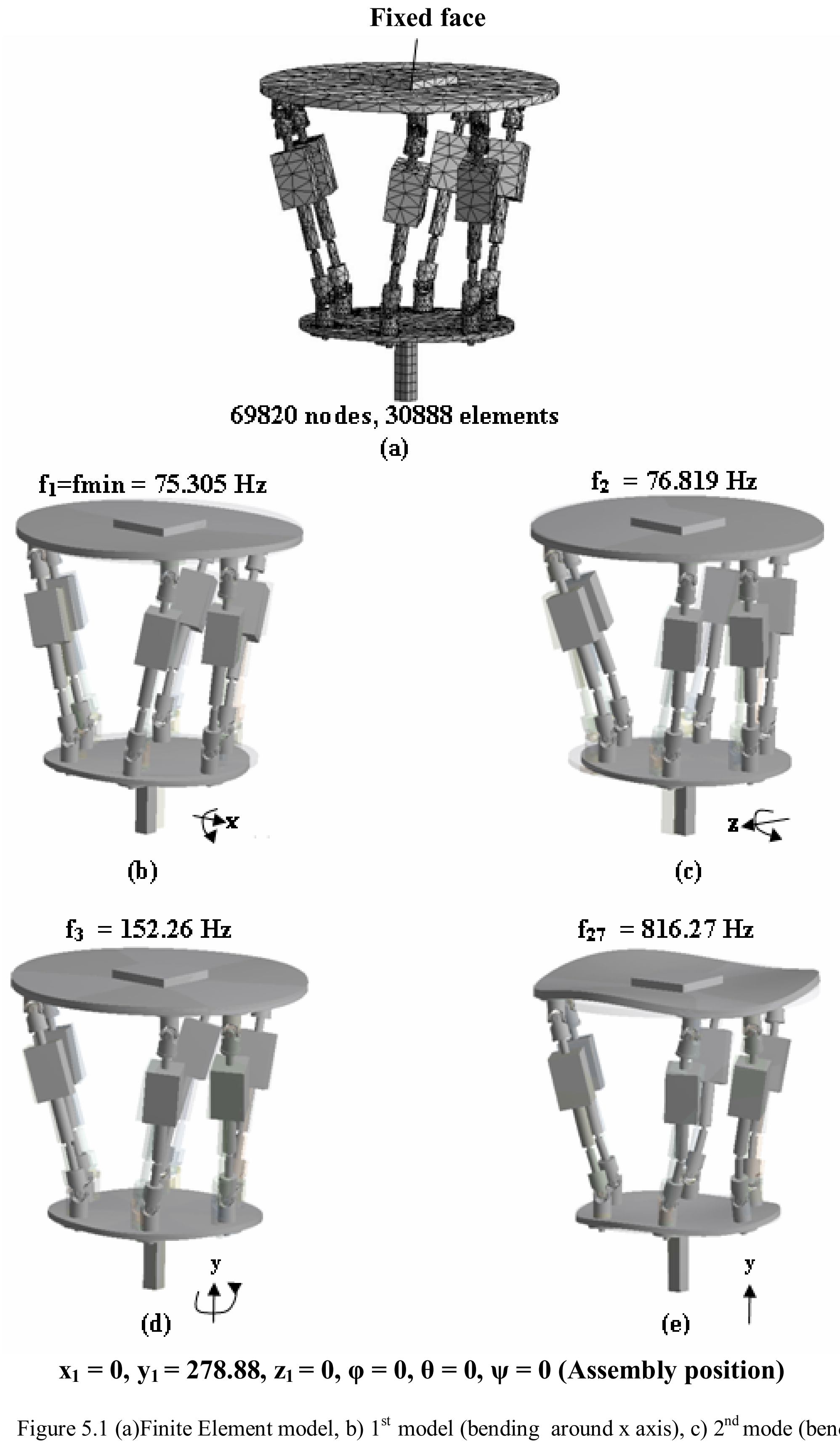
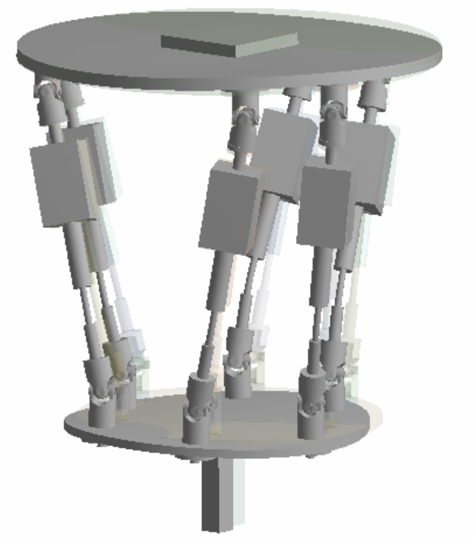


Figure 5.1 (a)Finite Element model, b) 1st model (bending around x axis), c) 2nd mode (bending around z axis), d) 3rd mode (bending around y axis), e) 27th mode (elongation along y axis).

In figures minimum natural frequency (f_{min}) values are shown for different positions of hexapod.

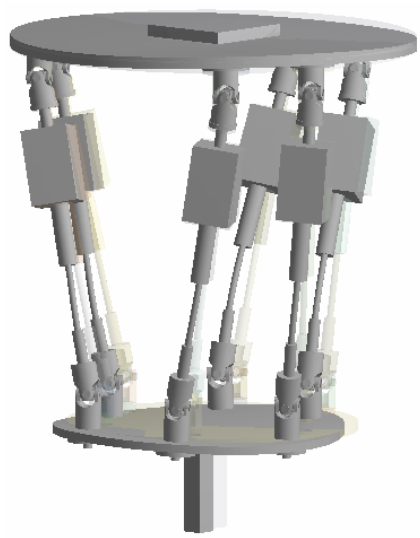
fmin = 65.622 Hz



x_i= 0, y_i = -26, z_i = 0
φ_i = 0°, θ_i = 0°, ψ_i = 0°

(a)

fmin = 59.041 Hz



x_i= 0, y_i = -52, z_i = 0
φ_i = 0°, θ_i = 0°, ψ_i = 0°

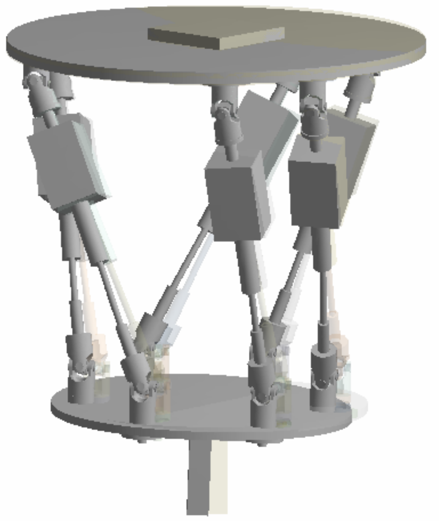
(b)

fmin = 63.927 Hz

x_i= 0, y_i= -26, z_i= 0
φ_i = 17°, θ_i = 0°, ψ_i = 0°

(c)

fmin = 60.692 Hz



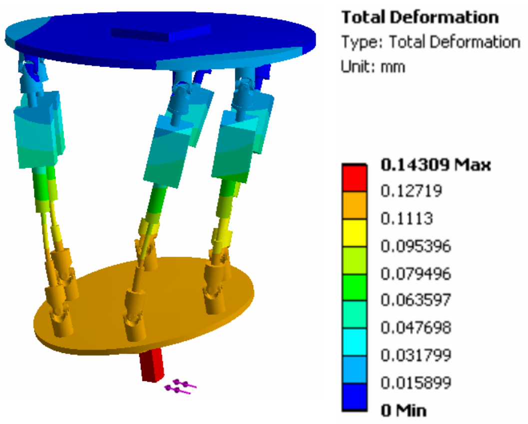
x_i= 0, y_i= -26, z_i= 0
φ_i = 0°, θ_i = 43°, ψ_i = 0°

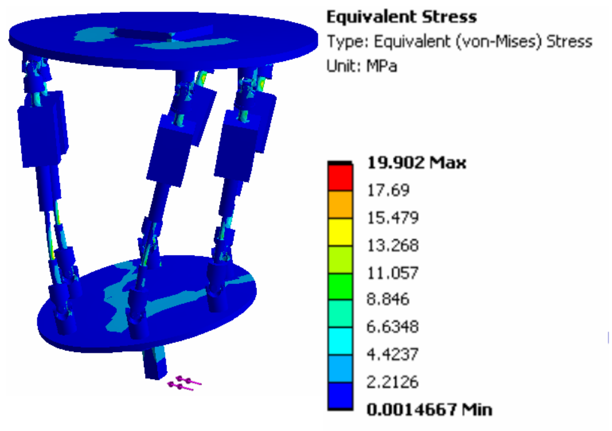
(d)

Figure 5.2 First natural frequencies of the hexapod for various positions

In figures static finite element analyses are shown as example.

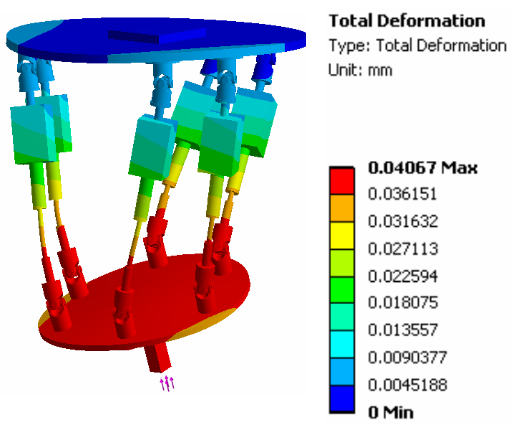
u_{max}= 143.09 micron **s_{max}= 19.902 MPa** **von Misses**

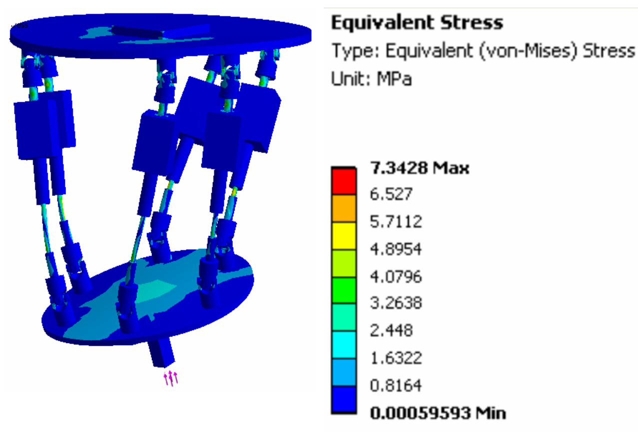




Position: $x_i = 0$, $y_i = -26$, $z_i = 0$, $\varphi_i = 17^o$, $\theta_i = 0^o$, $\psi_i = 0^o$. Load: $f_{xb} = -100$N

u_{max}=40.67 micron **s_{max}= 7.3428 MPa** **von Misses**





Position: $x_i = 0$, $y_i = -26$, $z_i = 0$, $\varphi_i = 17^o$, $\theta_i = 0^o$, $\psi_i = 0^o$. Load: $f_{yb} = 100$N

Figure 5.3 Static finite element results

Table 5.1 Rigidity values for various positions and loadings.

Position	**f_{min} (Hz)**	**Load: f_{xb} = - 100N**		**Load: f_{yb} = 100N**	
		u_{max} (micron)	**s_{max} (MPa)**	**u_{max} (micron)**	**s_{max} (MPa)**
Figure 5.1	75.305	103.48	21.592	8.445	4.630
Figure 5.2 (a)	65.622	129.84	16.940	8.812	4.203
Figure 5.2 (b)	59.041	160.27	20.728	8.999	4.235
Figure 5.2 (c)	63.927	143.09	19.902	40.670	7.342
Figure 5.2 (d)	60.692	197.47	34.201	29.669	13.679

Table 5.2 When gravity along –y direction are taken into calculation, rigidity values for various positions and loadings.

Position	**Load: f_{xb} = - 100N**		**Load: f_{yb} = 100N**	
	u_{max} (micron)	**s_{max} (MPa)**	**u_{max} (micron)**	**s_{max} (MPa)**
Figure 5.1	103.90	21.833	4.101	4.315
Figure 5.2 (a)	130.23	16.873	5.411	3.955
Figure 5.2 (b)	160.63	20.666	5.861	3.980
Figure 5.2 (c)	143.57	19.618	37.027	6.608
Figure 5.2 (d)	198.54	47.616	13.449	4.483

Maximum limit of s_{max} are determined as a specific percent (typically %50) of yield strength of material. In this study s_{max} values are far below then selected structural steel, yield strength of 350 MPa. The limits for f_{min} and u_{max} can be determined considering the precision of the application. Typical limit values may be given as in Table 5.3.

Table 5.3 Typical rigidity limits of hexapod in position $x_i = 0$, $y_i = -26$, $z_i = 0$, $\varphi_i = 0^o$, $\theta_i = 0^o$, $\psi_i = 0^o$

f_{min}	**u_{max} (Load: f_{xb} = -100N)**	**u_{max} (Load: f_{yb} = 100N)**	**u_{max}**
above 70 Hz	below 200 micron	below 10 micron	below 175 MPa

5.2 Kinematic and Kinetic Analysis Result

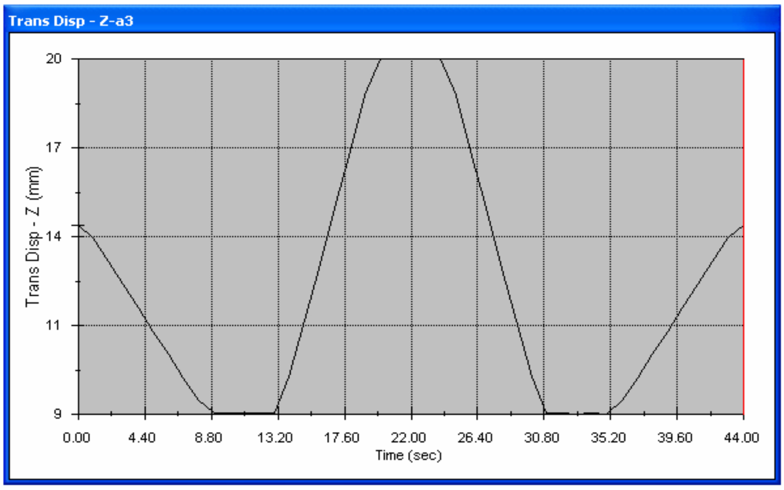
ANALYSIS - 1

As first analysis example, below shown sample work database is examined.

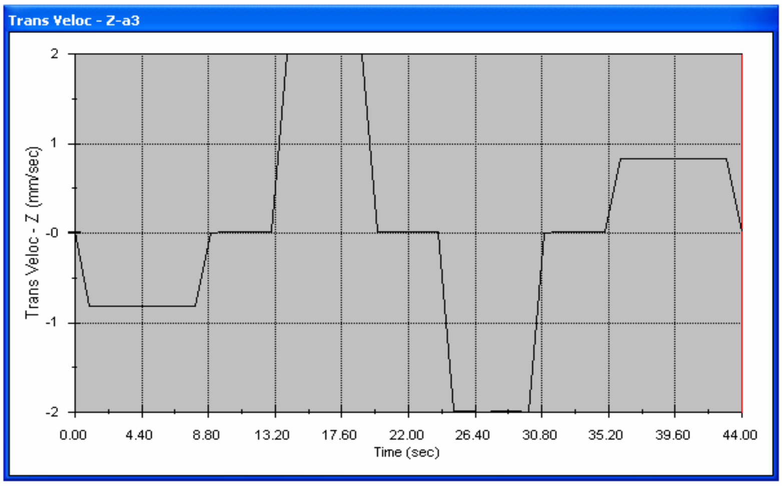
```
Analysis-1 Database
9,0.1,0.2,w,0,-5,0,-8,0,0,f,0,0,0
4,wait,f,0,0,0
7,0.2,0.1,r,0,-12,0,f,0,100,0
4,wait,f,0,0,0
7,0.2,0.1,r,0,12,0,f,0,0,0
4,wait,f,0,0,0
9,0.1,0.2,w,0,5,0,8,0,0,f,0,0,0
-1
```

The starting position for Analysis-1 is given by the generalized coordinate vector $\mathbf{q}_s = [0, -15, 0, 0, 0, 0]^T$. According to Analysis-1 database, the hexapod moves in directions of y_i = -5 mm and φ_i = -8° in 9 s with acceleration and deceleration rate, 0.1, 0.2, respectively. Then the hexapod stands for 4 s. Then in it translate y_{bi} =12 mm in its orientation in 7 s with acceleration and deceleration rate, 0.2, 0.1, respectively. While these move an external force of 100 N is exerted in y_{bi} direction. Then waits 4 s without exerting force, moves back 12 mm in y_{bi} direction, waits 4 s and returns starting position. The total duration is 44 s.

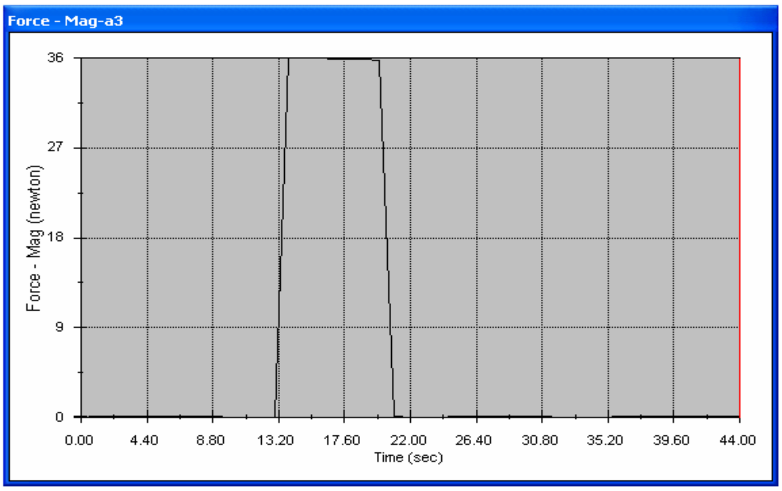
The inverse kinematic analysis gives the result that the actuator displacements are all positive and the maximum actuator displacement (r_{max}) is equal to 42.51 mm; the maximum universal joint angle (α_{max}) is 14.11°, and the maximum actuator velocity (v_{max}) is equal to 1.98 mm/s. These values are below the limits and the motion is possible according to the kinematic workspace evaluation. The time histories of displacement, velocity and actuating force for the actuator-3 are given in Fig. 5.4. The maximum actuator force (f_{max}) is 36.22 N for all the actuators for all the motion time. So, this value is below the limit and the motion is possible according to the kinetic workspace evaluation.



(a)

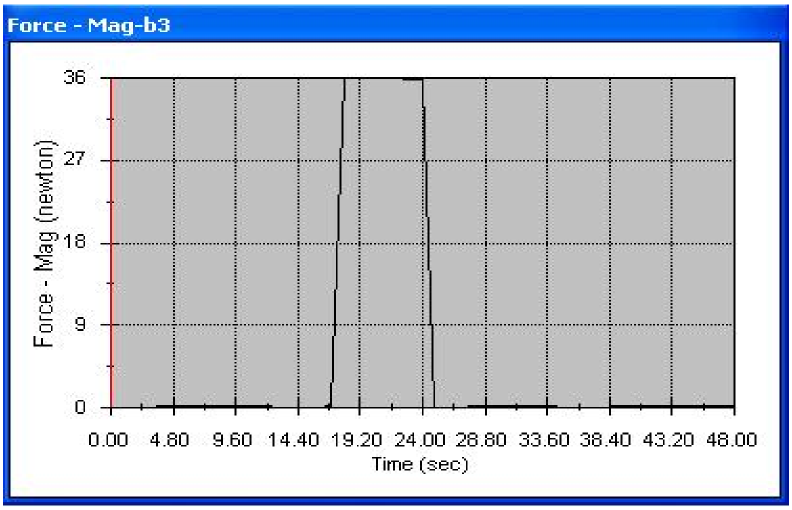


(b)

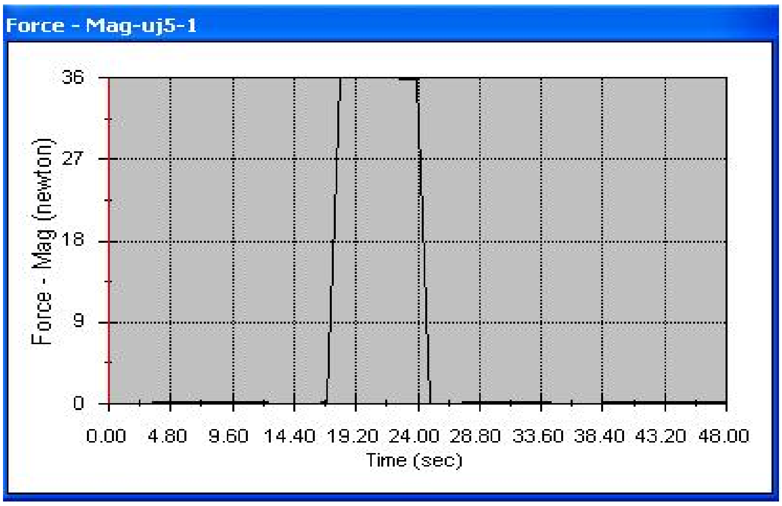


(c)

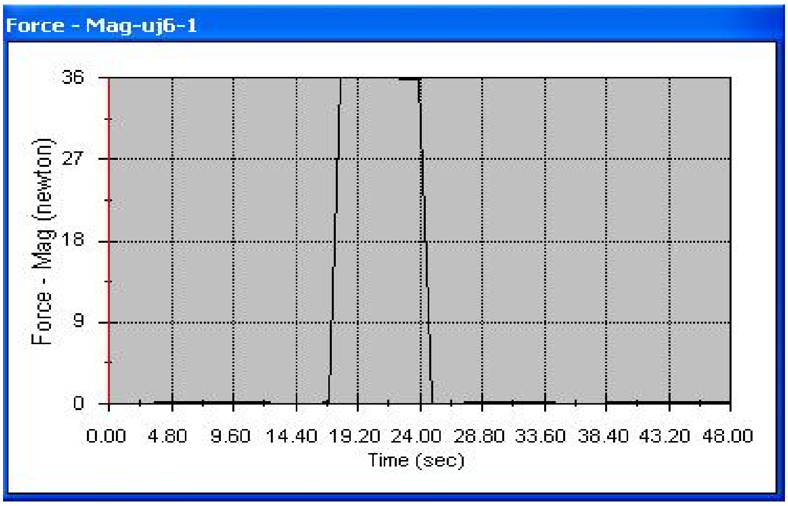
Figure 5.4 Time histories of (a) displacement-time, (b)velocity-time and (c) actuation force-time for the actuator-3 for Analysis-1



(a)



(b)



(c)

Figure 5.5 Time histories of (a) bearing-3 actuation force-time, (b) universal joints-5 actuation force-time, (c) universal joints-6 actuation force-time for Analysis-1

The time histories of bearing-3 actuation force, universal joints-5 actuation force and universal joints-6 actuation force are given in Fig. 5.5. The maximum bearing force (f_{max}) is 36.22 N for all the bearings for all the motion time. The maximum universal joint force (f_{max}) is 36.22 N for all the universal joints for all the motion time. So, this values are below the limit and the motion is possible according to the kinetic workspace evaluation.

The work force is available for the time interval 13-20 s. The rigidity analysis results in this time interval are given in Table 5.4.

Table 5.4 Rigidity analysis results for Analysis-1

t (s)	**f_{min} (Hz)**	**u_{max} (micron)**	**s_{max} (MPa)**
13	65.639	24.847	4.8039
15	64.980	25.192	4.7422
17	63.720	25.860	4.6628
20	62.521	26.527	4.6244

As seen in table rigidity of hexapod is change along motion. Less rigidity is at 20^{th} second. Rigidity workspace evaluation can be used to evaluate fatigue effect of Analysis-1.

ANALYSIS - 2

In second example, the hexapod move the end-point of the tool 120 mm in 20 s against a work force of 100 N is considered. The rigidity of the hexapod is higher and the actuator forces are lower if it is operated in the y direction. But, the stroke is lower in this direction and the work for Analysis-2 cannot be performed. Two alternatives are possible of application, are shown in fig. 5.6. In case-1 ‘m4’ sub-assembly is moved x direction without initial rotation. In case-2 ‘m4’ sub-assembly is moved with an initial rotation of $\psi_i = 8^o$ around z.

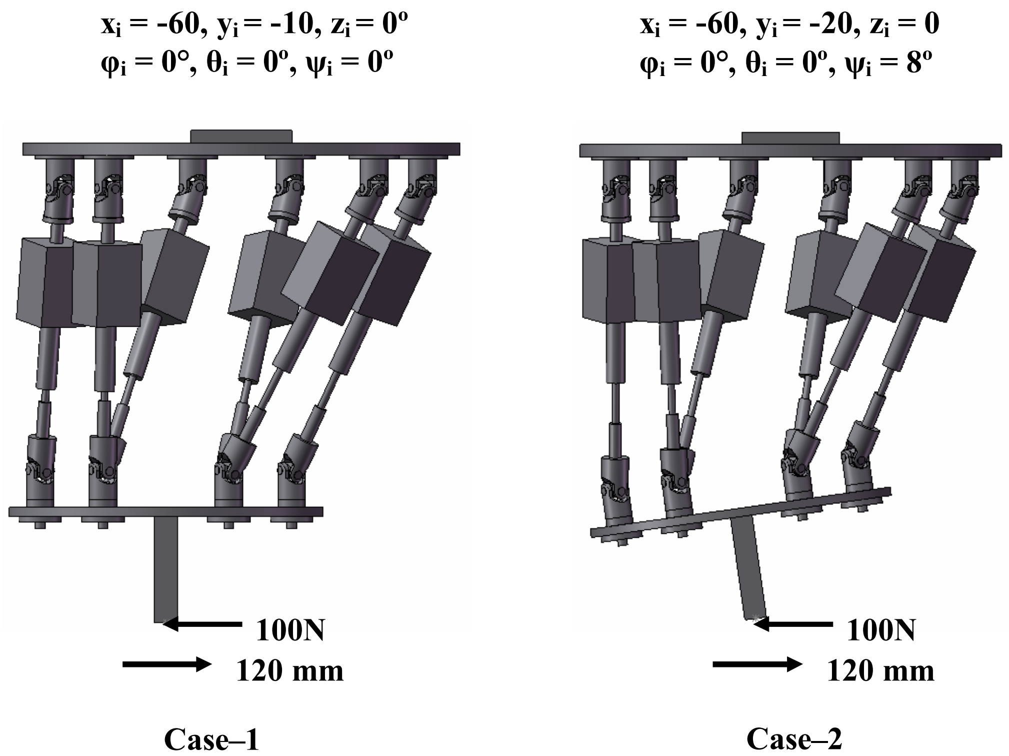


Figure 5.6 Initial positions in Analsys-2.

Kinetic and rigidity workspace evaluation results are presented in Table 5.5. As seen as in table 5.5 during work, that the actuator driving forces are less and the hexapod structure are less deformed for the staring position case-1 as compared with case-2 during the work defined for Analsys-2. With this example it is pointed out the importance of integration software not only using initial design and simple control process also generating best path for accuracy and longer life of mechatronic systems.

Table 5.5 Kinetic and rigidity analyses results for Analysis-2

Results	Case-1	Case-2
f_{max} (Maximum actuator force) (N)	167.34	175.79
f_{min} (1^{st} Natural frequency) (Hz)	69.99	65.07
u_{max} (The end point displacement) (micron)	117.72	133.72
s_{max} (Maximum vonMises stress) (MPa)	25.09	26.04

Extreme values are found at t= 20s. Gravity is not considered.

5.3 Detailed Model of Hexapod

As a result of the above analysis, after un-detailed models production parts of hexapod are determined, the production parts are detailed by considering the connection of details of production and market parts.

Production parts are detailed in SolidWorks. Technical drawings are prepared. Machining processes of production parts are done according tolerances and dimensions according technical drawings. Technical drawings of production parts are given in Appendix-A.

CHAPTER SIX
CONCLUSIONS

In this study, a hexapod with universal- prismatic- universal- revolute (UPUR) structure is designed and analyzed. Universal joints, parallel motion actuators (linear actuators) and revolute joints (rolling element bearings) are available in the market with competitive prices and thus, hexapods with UPUR structure can be produced with low costs.

There are parts readily available in the market (market parts) and parts to be produced (production parts) in the hexapod design. For the initial design, un-detailed solid models of market and production parts prepared preserving mass and strength properties. Relations between the parts were identified and a detailed hexapod model built.

An integrated design and analysis method developed in Computer Aided Design and Analysis Laboratory at Dokuz Eylül University (BATÜL) is used for the hexapod considered in this work. In this method, SolidWorks is used for modeling and assembly, CosmosMotion is used for rigid body dynamics and ANSYS is used for the finite element rigidity analysis. The natural frequencies and the static deflections of the system are given for various positions. Hexapod motions are analyzed considering kinematic, kinetic and rigidity limits.

A program developed in BATÜL is used for analytical rigid body dynamics results. The program uses the theory given in H.B.Guo & H.R. Li 2006. The theory uses the explicit compact closed-form dynamic equations applying the combination of the Newton-Euler method and the Lagrange formulation. The program is modified fort the hexapod considered in this study and the analytical results are obtained. It is observed that MATLAB results are in agreement with CosmosMotion results.

A control system is developed for the system having linear actuators with step motors. PC-based motion control is used. Motor controls are done with ADLINK software. Softwares are controlled by developed integration program developed in VisualBASIC.

The results obtained in the programs are transferred between them as explained with the flow chart given in this work and the system is evaluated in its workspace with kinematic, kinetic and rigidity considerations. These evaluations are important at the programming stage when the system is in use as well as at the design stage.

The hexapod determined to be capable of performing targeted functions and operations by kinematics workspace evaluation as a result of studies conducted. Compatibility of market parts chosen were determined by kinetic workspace evaluation and compatibility of dimensions and materials of un-detailed production parts determined by rigidity workspace evaluation. Detailed models of production parts required for manufacturing a prototype of the hexapod robot with UPUR connections prepared and their technical drawings produced.

REFERENCES

Alizade, R., & Bayram, C. (2004). Structural Synthesis of Parallel Manipulators. *Mechanism and Machine Theory, 39*, 857-870.

Dasgupta, B., & Mruthyunjaya, T.S. (2000). The Stewart Platform Manipulator: A Review. *Mechanism and Machine Theory, 35*, 15-40.

E.F. Fichter, A Stewart platform based manipulator: general theory and practical considerations, *International Journal of Robotics Research*. 5 (1986) 157–182.

Gao, F., Li, W., Zhao, X., Jin. Z., & Zhao H. (2002). New Kinematic Structures for 2-, 3-, 4-, and 5- DOF Parallel Manipulator Designs. *Mechanism and Machine Theory, 37*, 1395-1411.

Gough, V.E., & Whitehall, S.G. (1962). Universal Tyre Test Machine. *Proceedings of the 9th International Technical Congress (FISITA), 1*, 177.

Han, S.H., Kim, Y.H., Ha, I.J., Lee, S.T., & Park, J.J. (1995). A Learning Approach to High Precision Speed Control of Servo Motors. *IEEE, 26* (3), 221-226.

Hauge, G.S., & Campbell, M.E. (2004). Sensors and Control of a Space-Based Six-Axis Vibration Isolation System. *Journal of Sound and Vibration, 269*, 913-931.

Haydon Switch & Instrument (HSI) Corp. (2006). *Haydon Switch & Instrument.* Retrieved 2006, from http://www.haydonkerk.com/LinearActuatorProducts/StepperMotorLinearActuators/LinearActuatorsHybrid/Size17LinearActuator/tabid/79/Default.aspx

H.B. Guo, H.R. Li, Dynamic analysis and simulation of a six degree of freedom Stewart platform manipulator, Proc Inst Mech Eng Part C J. *Mechanical Engineering Science.* 220 (2006) 61-72.

Hunt, K.H. (1983). Structural Kinematics of In-Parallel - Actuated Robot Arms. *ASME J. Mech. Transm. Autom. Des., 105*, 705-712.

Ider, S.K. (2005). Inverse Dynamics of Parallel Manipulators in the Presence of Drive Singularities. *Mechanism and Machine Theory, 40*, 33-44.

Jelenkovic, L., Jakobovic, D., & Budin, L. (2004). Hexapod Structure Evaluation as WEB Service. *Proceedings of the 1st International Conference on Informatics and Robotics (ICINCO), 1.*

Karagülle, H., Sarıgül, S., Kıral, Z., Varol, K., & Malgaca, L. (01 November 2008). Mikro-konumlandırıcı Robot Tasarımı ve Prototip İmalatı. *TÜBİTAK Araştırma Projesi Sonuç Raporu, Project No: 104M373.*

Kim, H.S., Cho, Y.M., & Lee, K. (2005). Robust Nonlinear Task Space Control for 6 DOF Parallel Manipulator. *Automatica*, 41, 1591-1600.

Lenze Company. (2006). Lenze. Retrieved 2006, from http://www.lenze.co.uk_en_active/200_Partner_Products-/210_AT_Partner_Products/0110_Verbindungstechnik/0150_Wellengelenke_Gelenkswellen/0110_Wellengelenke/PP_Produkt_Wellengelenke.uk.jsp?cid=0b0164e0800ec60e

Lin, C.L., Jan, H.Y., Hwang, T.S., & Tsai, R.C. (2003). Control Design for a Mixed Rotary and Linear Motors Based Manipulator. Proceedings of the 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics *(AIM 2003), 1*, 1298-1303.

McInroy, J. (1999). Dynamic Modelling of Flexure Jointed Hexapods for Control Purposes. *Proceedings of the 1999 IEEE International Conference on Control Applications, 1*, 508-513.

M.J. Rogers, I. McFayden, J.A. Livingstone, F. Monsell, M. Jackson, R. Atkins, Computer hexapod assisted orthopaedic surgery (chaos) in the correction of long bone fracture and deformity, *Journal of Orthopedic Trauma*. 21(2007) 337-342.

Physik Instrumente (PI) Company. (2006). *PI.* Retrieved 2006, from http://www.physikinstrumente.com/en/index.php.

RS Company (2006). *RS Company Turkey.* Retrieved 2006, from http://tr.rsdelivers.com/catalogue/category/güç-kaynakları/25.aspx

Simplex Cnc System Company. (2008), from http://www.simplex-cnc.com.au/download/ParallelKinemtaicsMachines.pdf

SolidWorks Corporation . (2007). *SolidWorks.* Retrieved January 14, 2007, from http://www.solidworks.com/sw/resources.htm

Stewart, D. (1965). A Platform with Six Degrees of Freedom. *Proceedings of the Institution of Mechanical Engineers, 180*, 371-386.

Tsai, L. W. (1999). *Robot Analysis (The Mechanics of Serial and Parallel Manipulators).* New York: John Wiley & Sons Inc.

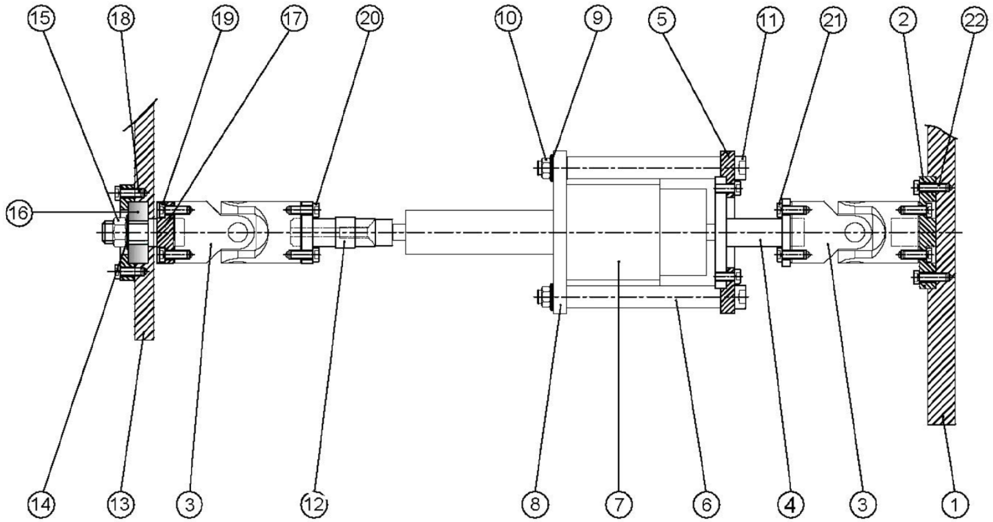
U. Laible, T. Bürger, G. Pritschow, A fail-safe dual channel robot control for surgery applications, *Safety Science*. 42 (2004) 423–436.

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

Wendlandt, J.M., & Sastry, S.S. (1994). Design and Control of a Simplified Stewart Platform for Endoscopy. *Proceeding s of the 33rd IEEE Conference on Decision and Control, 1*, 357-362.

APPENDIX A
TECHNICAL DRAWINGS OF HEXAPOD

Technical drawings of production parts and Hexapod assembly are given in following pages.



NO	ADET	PARÇA ADI
22	24	M3x10mm imbus civata
21	24	M3x10mm imbus civata
20	24	M3x10mm imbus civata
19	24	M3x10mm imbus civata
18	24	M3x10mm imbus civata
17	6	Mafsal-üst plaka bağlantı elemanı
16	6	Rulman
15	6	M8 somun
14	6	M8 düz rondela
13	1	Üst plaka
12	6	Motor mili-mafsal bağlantı elemanı
11	24	M5x75mm imbus civata
10	24	M5 somun
9	24	M5 düz rondela
8	6	Motor üst bağlantı plakası
7	6	Motor
6	24	Saplama borusu
5	6	Motor alt bağlantı plakası
4	6	Motor-mafsal bağlantı elemanı
3	12	Mafsal
2	6	Alt plaka mafsal bağlantı elemanı
1	1	Alt plaka

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

NEXT ASSY | USED ON
APPLICATION

UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

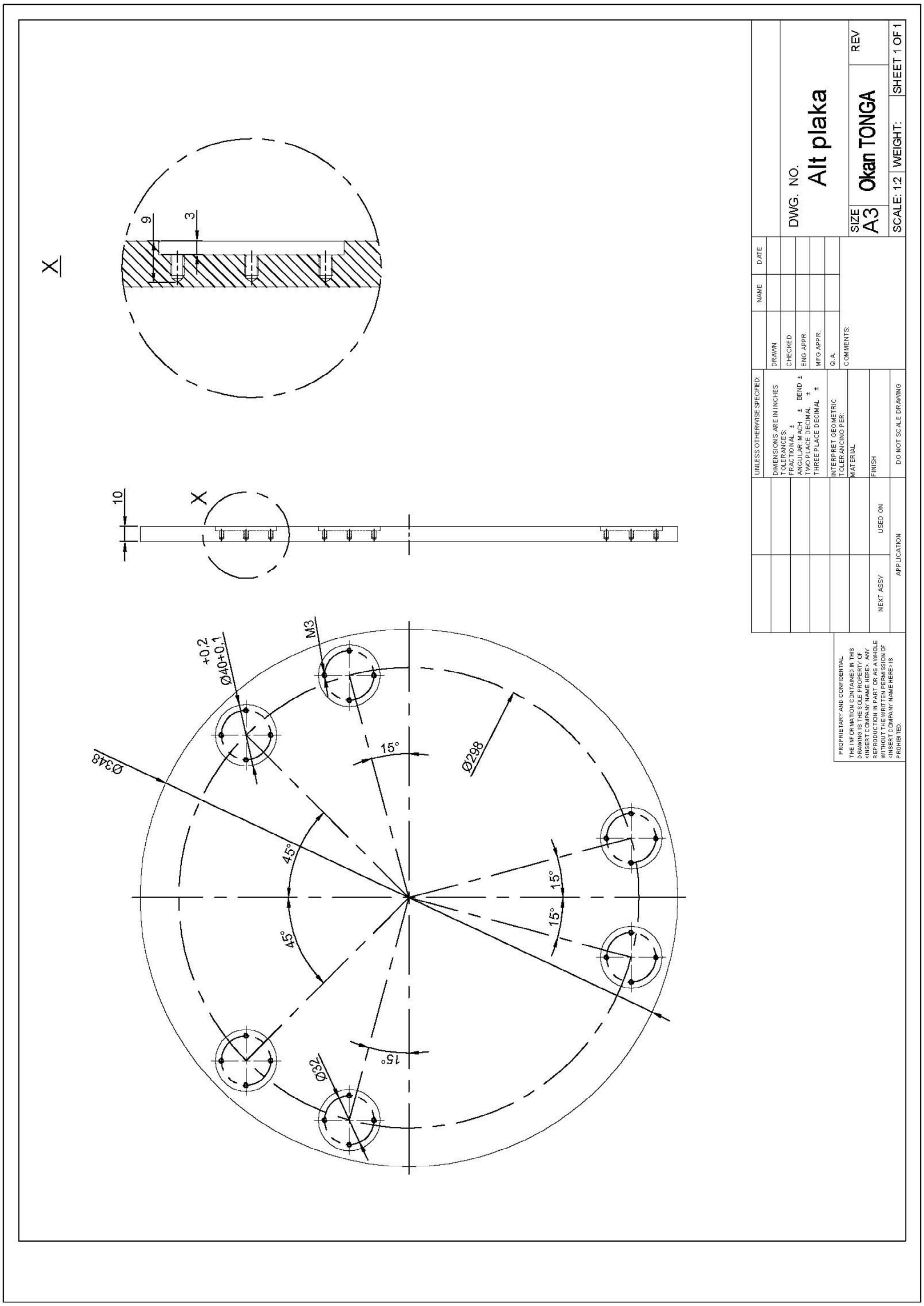
COMMENTS:

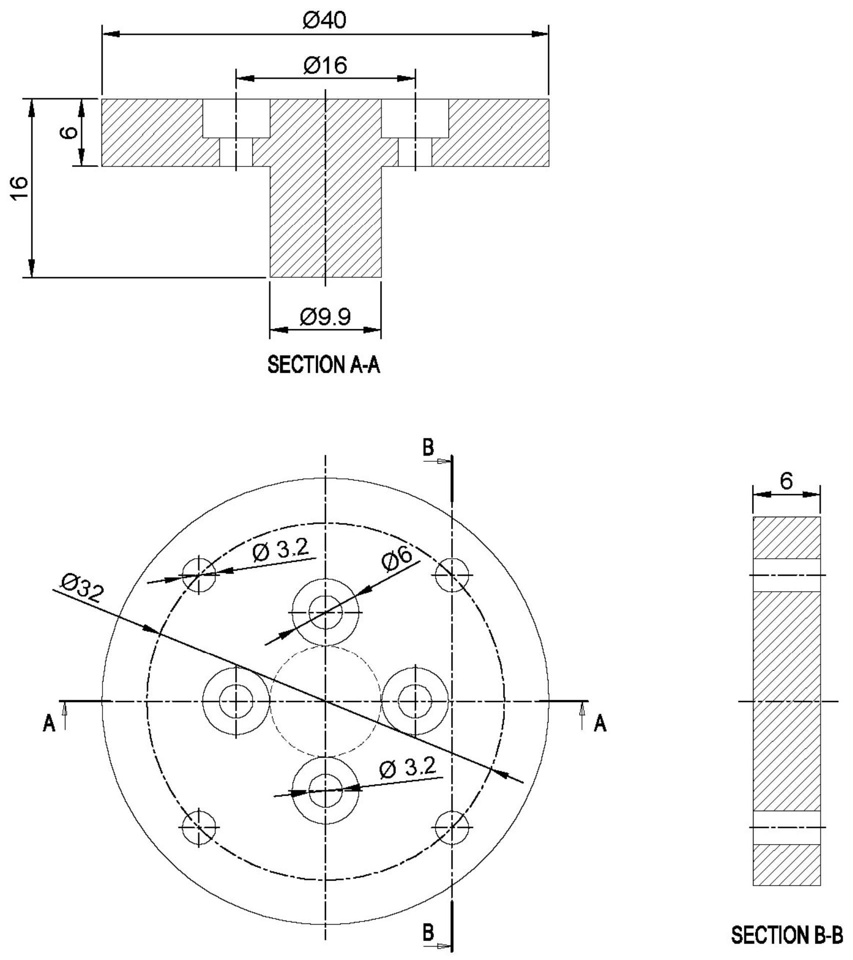
DWG. NO.

Hexapod elemanları

SIZE A2 | Okan TONGA | REV

SCALE: 1:1 | WEIGHT: | SHEET 1 OF 1


X
9
3
10
X
M3
+0,2
Ø40+0,1
Ø348
15°
Ø298
45°
45°
15°
15°
Ø32
15°
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
NEXT ASSY
USED ON
APPLICATION
PROPRIETARY AND CONFIDENTIAL
DWG. NO.
Alt plaka
SIZE
A3
Okan TONGA
REV
SCALE: 1:2
WEIGHT:
SHEET 1 OF 1



PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

NEXT ASSY | USED ON

APPLICATION

DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±

MATERIAL

FINISH

DO NOT SCALE DRAWING

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

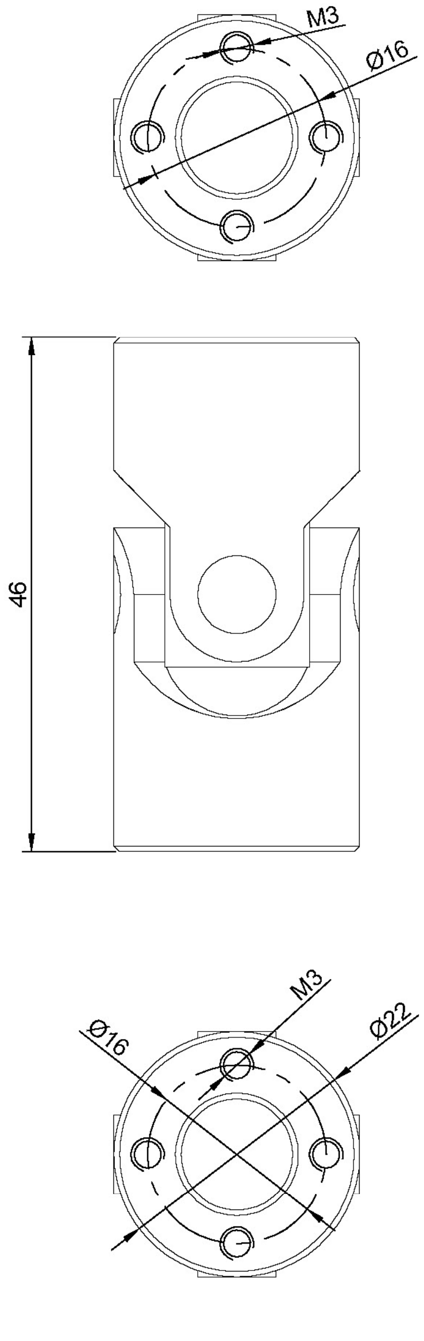
COMMENTS:

DWG. NO.

Alt plaka-mafsal bağlantı elemanı

SIZE A4 | Okan TONGA | REV.

SCALE 2:1 | WEIGHT: | SHEET 1 OF 1



PROPRIETARY AND CONFIDENTIAL

THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

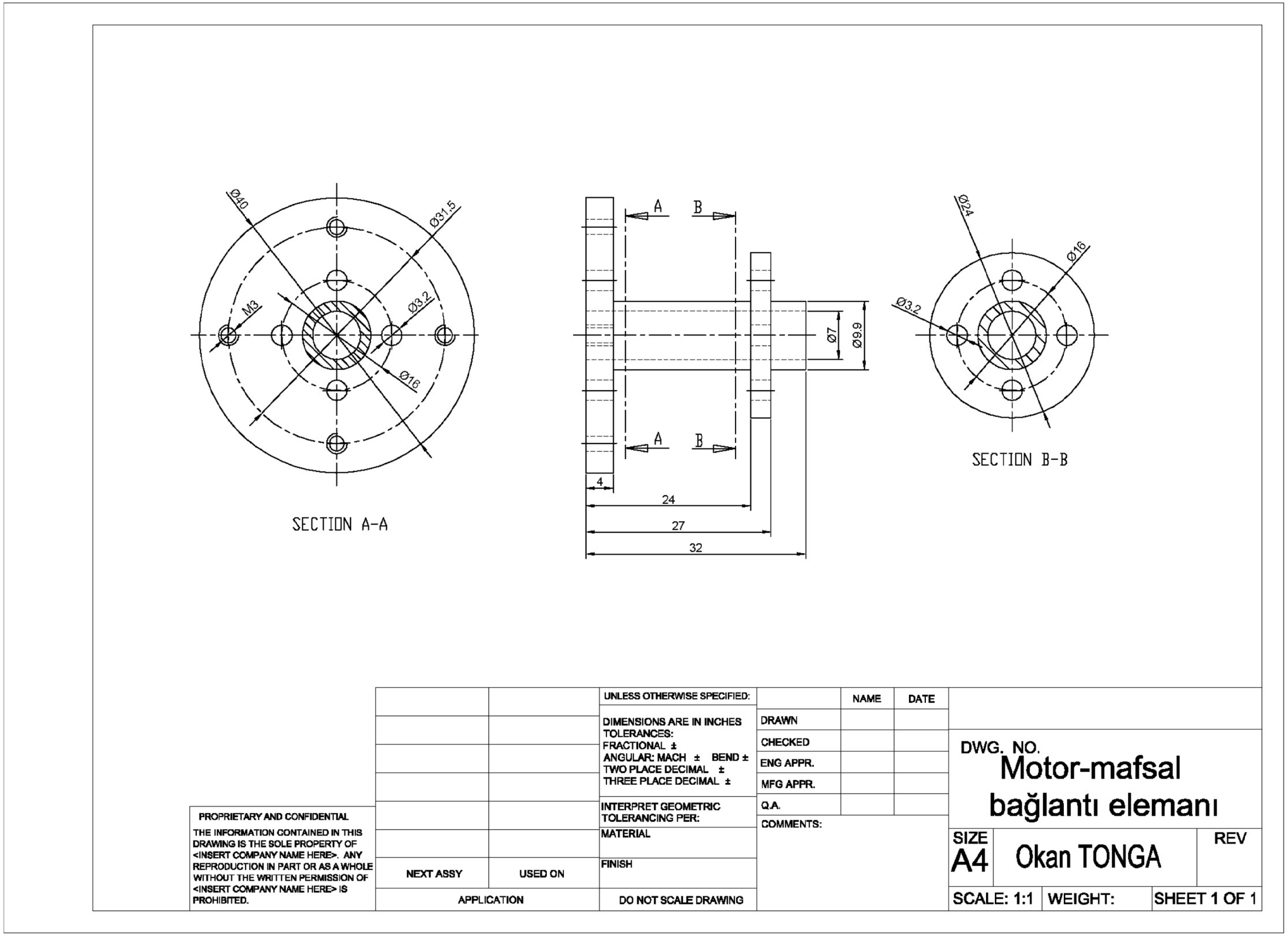
NEXT ASSY	USED ON	DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL ± ANGULAR: MACH ± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±		NAME / DATE
APPLICATION		MATERIAL	DRAWN	
		FINISH	CHECKED	
		DO NOT SCALE DRAWING	ENG APPR.	
			MFG APPR.	
			Q.A.	
			COMMENTS:	

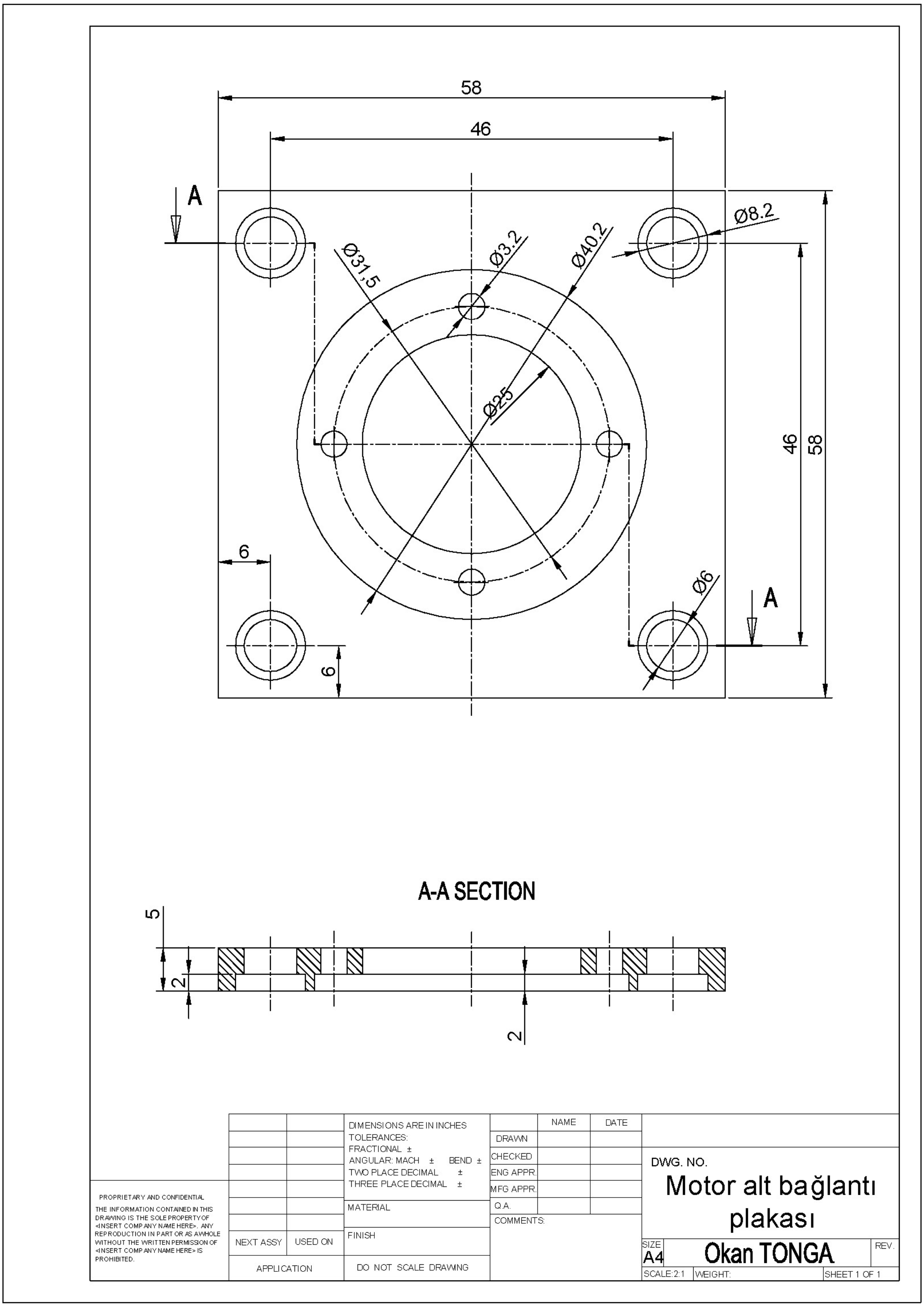
DWG. NO.

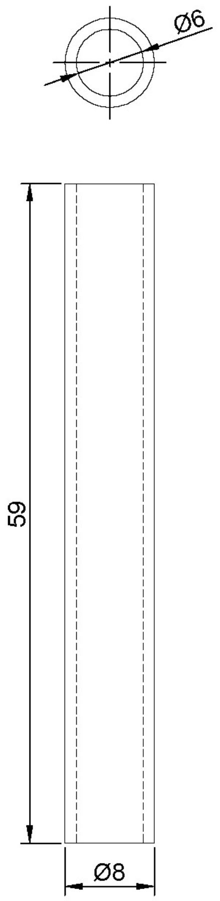
Mafsal

SIZE A4 — Okan TONGA — REV.

SCALE 2:1 — WEIGHT: — SHEET 1 OF 1


Ø40
Ø31.5
M3
Ø3.2
Ø16
SECTION A-A
A
B
4
24
27
32
Ø7
Ø9.9
Ø24
Ø16
Ø3.2
SECTION B-B
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
DWG. NO.
Motor-mafsal bağlantı elemanı
SIZE
A4
Okan TONGA
REV
SCALE: 1:1
WEIGHT:
SHEET 1 OF 1
NEXT ASSY
USED ON
APPLICATION
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.


58
46
A
Ø8.2
Ø3.2
Ø40.2
Ø31,5
Ø25
46
58
6
6
Ø6
A
A-A SECTION
5
2
2
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
MATERIAL
FINISH
DO NOT SCALE DRAWING
NEXT ASSY
USED ON
APPLICATION
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.
DWG. NO.
Motor alt bağlantı plakası
SIZE
A4
Okan TONGA
REV.
SCALE:2:1
WEIGHT:
SHEET 1 OF 1



		DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL ± ANGULAR: MACH ± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±		NAME	DATE
			DRAWN		
			CHECKED		
			ENG APPR.		
			MFG APPR.		
		MATERIAL	Q.A.		
		FINISH	COMMENTS:		
NEXT ASSY	USED ON				
APPLICATION		DO NOT SCALE DRAWING			

PROPRIETARY AND CONFIDENTIAL

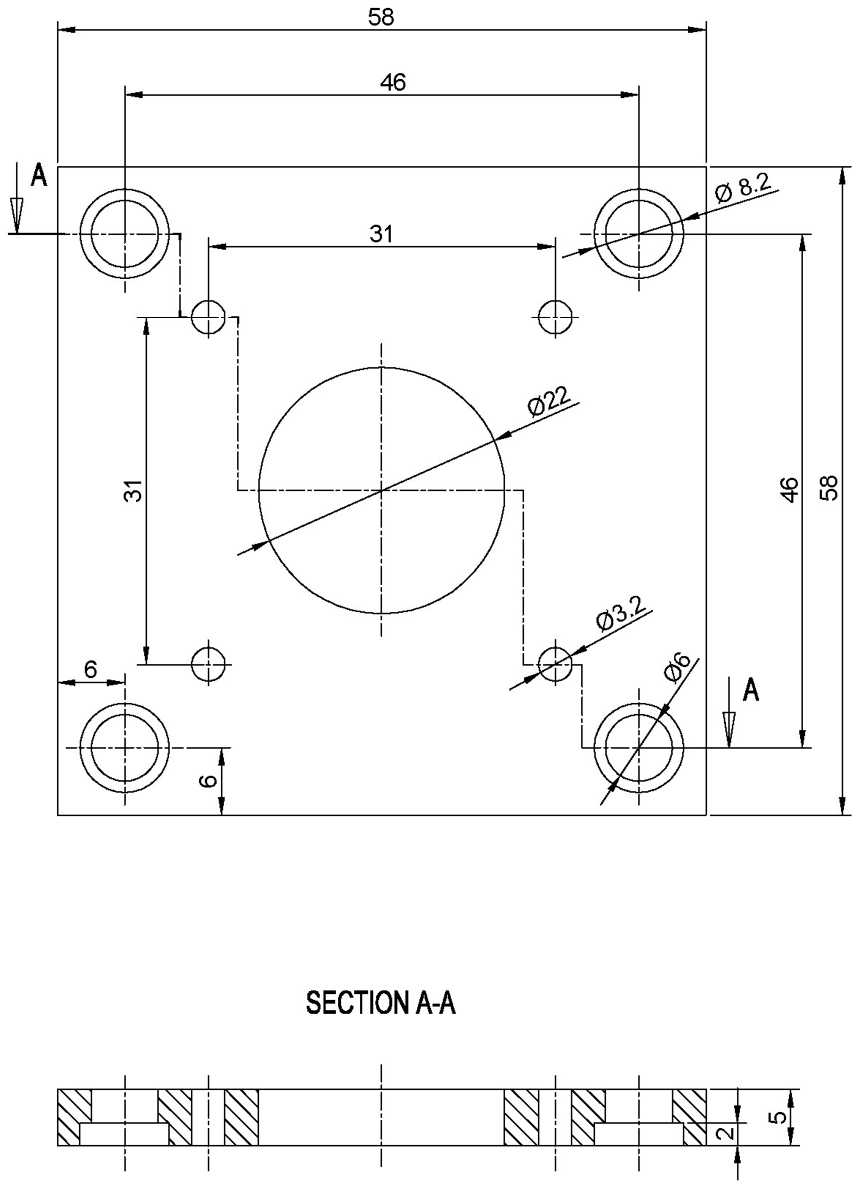
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

DWG. NO.

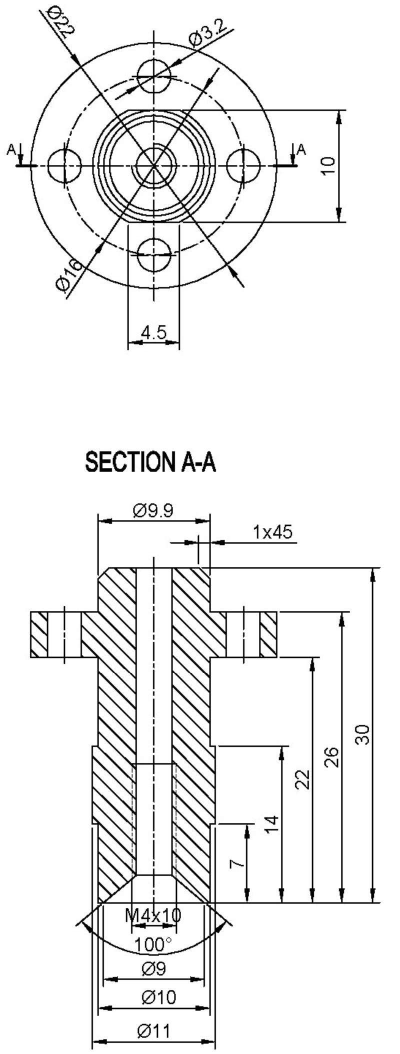
Saplama borusu

SIZE A4 | Okan TONGA | REV.

SCALE:2:1 | WEIGHT: | SHEET 1 OF 1



		DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL ± ANGULAR: MACH ± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±		NAME	DATE	
			DRAWN			
			CHECKED			DWG. NO.
			ENG APPR.			Motor üst bağlantı plakası
			MFG APPR.			
		MATERIAL	Q.A.			
		FINISH	COMMENTS:			
NEXT ASSY	USED ON					SIZE A4 · Okan TONGA · REV.
APPLICATION		DO NOT SCALE DRAWING				SCALE 2:1 · WEIGHT: · SHEET 1 OF 1



DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

COMMENTS:

MATERIAL

FINISH

NEXT ASSY | USED ON

APPLICATION

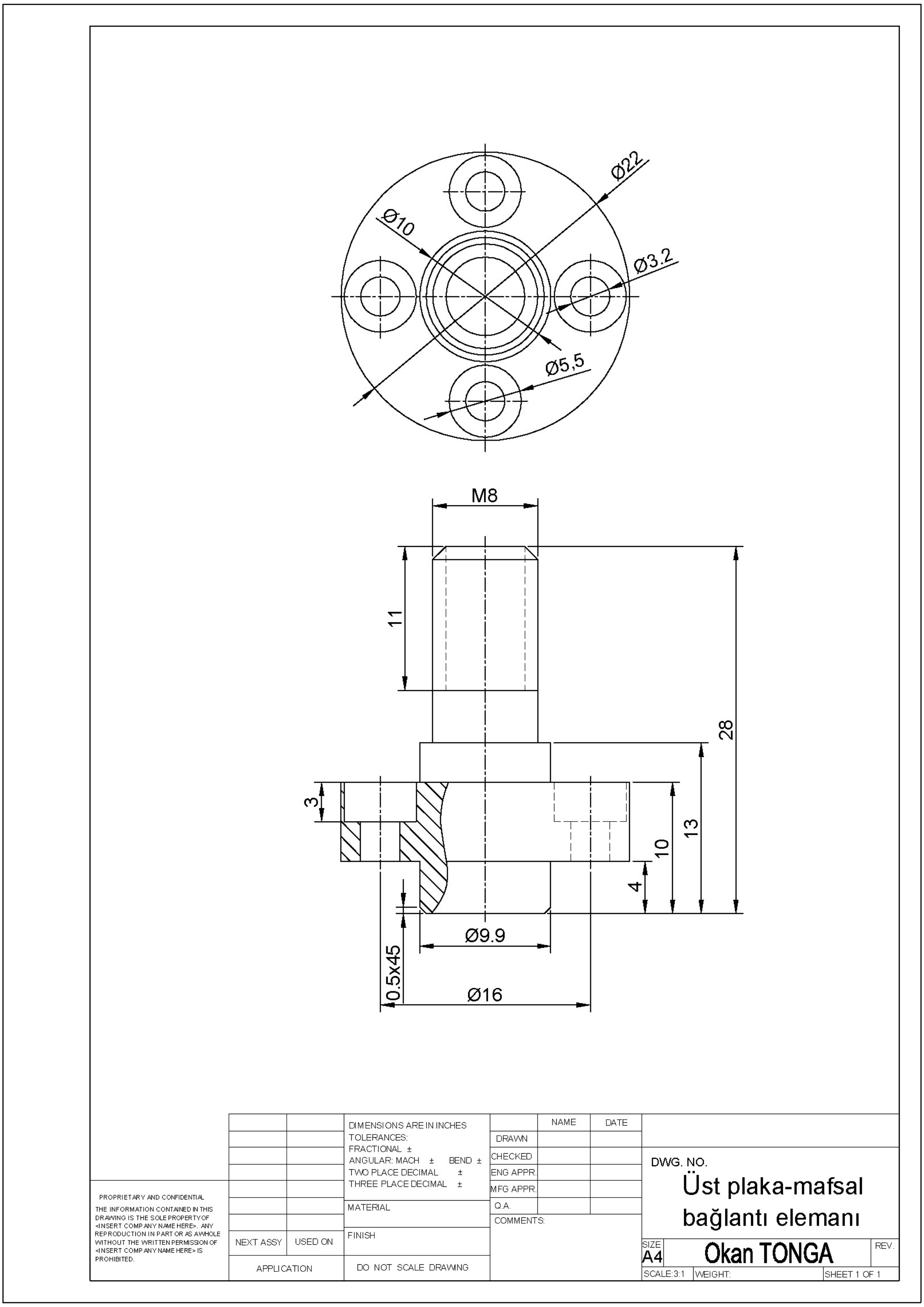
DO NOT SCALE DRAWING

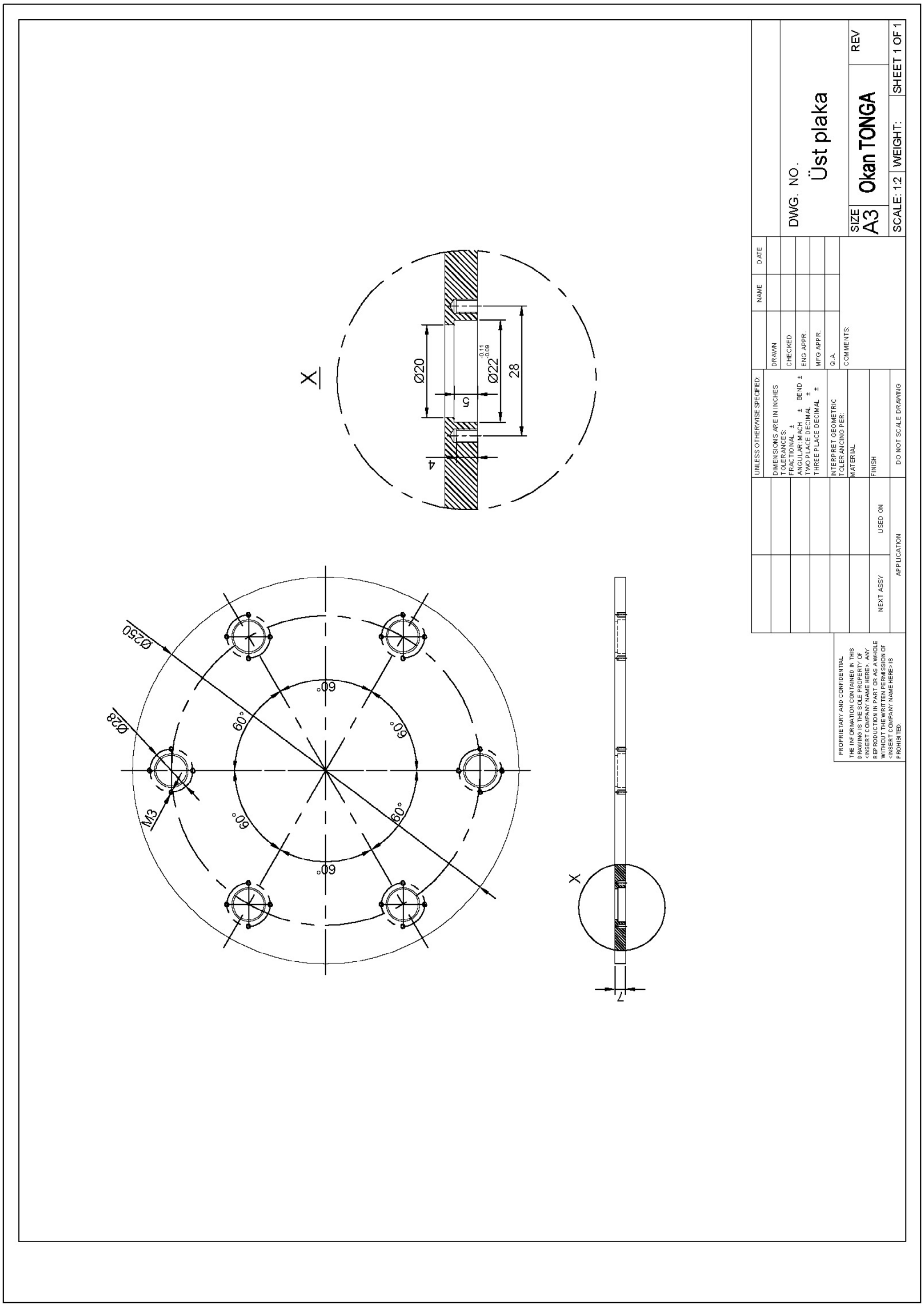
DWG. NO.

Motor mili-mafsal bağlantı elemanı

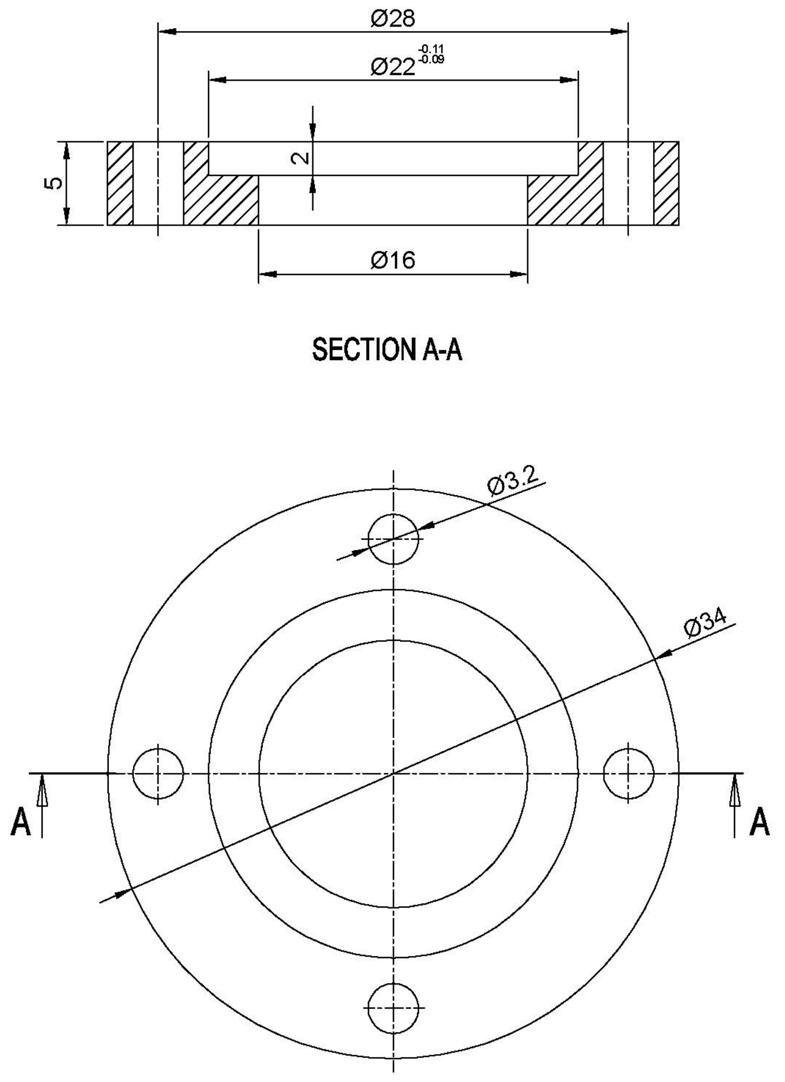
SIZE A4 — **Okan TONGA** — REV.

SCALE:2:1 WEIGHT: SHEET 1 OF 1


Ø22
Ø10
Ø3.2
Ø5,5
M8
11
28
3
13
10
4
Ø9.9
0.5x45
Ø16
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
MATERIAL
FINISH
NEXT ASSY
USED ON
APPLICATION
DO NOT SCALE DRAWING
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.
DWG. NO.
Üst plaka-mafsal bağlantı elemanı
SIZE
A4
Okan TONGA
REV.
SCALE 3:1
WEIGHT:
SHEET 1 OF 1



Ø250
Ø28
M3
60°
X
Ø20
Ø22
28
5
4
7
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING
NEXT ASSY
USED ON
APPLICATION
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
DWG. NO.
Üst plaka
SIZE
A3
Okan TONGA
REV
SCALE: 1:2
WEIGHT:
SHEET 1 OF 1
PROPRIETARY AND CONFIDENTIAL



PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

NEXT ASSY | USED ON
APPLICATION

DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
MATERIAL
FINISH
DO NOT SCALE DRAWING

	NAME	DATE
DRAWN		
CHECKED		
ENG APPR.		
MFG APPR.		
Q.A.		

COMMENTS:

DWG. NO.

Rulman kapağı

SIZE A4 | Okan TONGA | REV.
SCALE:3:1 | WEIGHT: | SHEET 1 OF 1

APPENDIX B
ADLINK PCI 8154 AND PCI 8164 MOTION CONTROL CARDS

Features of PCI 8154 and PCI 8164

The PCI 8154 and PCI 8164 cards are 2 axes and 4 axes motion control cards with PCI interface, respectively. It can generate high frequency pulse trains to drive stepping motors and servo motors. Multiple PCI 8154 and PCI 8164 cards can be used in one system. Incremental encoder interface provide the ability to correct for positioning errors generated by inaccurate mechanical transmissions (Adlink Inc., 2006).

The following lists summarize the main features of the PCI 8154 and PCI 8164 motion control card. The information listed below can be found at Adlink Co. (Adlink Inc., 2009).

• 32-bit PCI bus, plug and play.
• 4 axes of step and direction pulse output for controlling stepping or servomotor.
• 6.55MPPS maximum pulse output frequency, linear, trapezoidal, or S-Curve velocity profile drive.
• Any 2 of 4 axes circular interpolation.
• Any 2-4 of 4 axes linear interpolation.
• Continuous interpolation for contour following motion.
• Change position and speed on the fly.
• Change speed by condition comparing.
• 13 home return modes with auto searching.
• Hardware backlash compensator and vibration suppression.
• 2 software end-limits for each axis.
• 0~268.435.455 or –134.217.728 to +134.217.727, 28-bit up/down counter for incremental encoder feedback.
• 2-axis high speed position latch input.

• 2-axis position compare trigger output with 4k FIFO auto loading.

• Simultaneous start/stop motion on multiple axes.

• Manual pulser input interface.

• Software supports a maximum of up to 12 PCI-8164 cards (48 axes) operation in one system.

• Libraries and utilities support DOS, Windows® 9X/NT/2000/XP, and Linux.

• 19.66 MHz internal reference clock.

• Pulse rate setting ranges (pulse ratio = 1: 65535).

• Position pulse setting range (28-bit): -134,217,728 to +134,217,728.

APPENDIX C
CONNECTION LIST OF ELECTRICAL PANEL

POWER SUPPLY

24 V SUPPLY - 1 (For 1-2 Actuators)	On/Off Switch	220 V AC Connection
24 V SUPPLY - 2 (For 3-4 Actuators)	On/Off Switch	220 V AC Connection
24 V SUPPLY - 3 (For 5-6 Actuators)	On/Off Switch	220 V AC Connection
24 V SUPPLY - 4 (For Terminal 1)	On/Off Switch	220 V AC Connection
24 V SUPPLY - 5 (For Terminal 2)	On/Off Switch	220 V AC Connection

24 V SUPPLY - 4 (For Terminal 1)

Terminal 1	100	Power Supply - 4	24 VDC
Terminal 1	98	Power Supply - 4	GND

24 V SUPPLY - 5 (For Terminal 2)

Terminal 2	100	Power Supply - 5	24 VDC
Terminal 2	98	Power Supply - 5	GND

ACTUATOR 1

SOCKET-M1 (MOTOR 1)	1	Driver 1	T1-3
SOCKET-M1 (MOTOR 1)	2	Driver 1	T1-4
SOCKET-M1 (MOTOR 1)	3	Driver 1	T1-1
SOCKET-M1 (MOTOR 1)	4	Driver 1	T1-2

SOCKET-E1 (ENCODER 1)	1	Terminal 1	12
SOCKET-E1 (ENCODER 1)	2	Terminal 1	18
SOCKET-E1 (ENCODER 1)	3	Terminal 1	14
SOCKET-E1 (ENCODER 1)	4	Terminal 1	13-15-17-19
SOCKET-E1 (ENCODER 1)	5	Terminal 1	16

Driver 1	T2-1	Power Supply - 1	GND
Driver 1	T2-2	Power Supply - 1	24 VDC
Driver 1	T2-3	Terminal 1	4
Driver 1	T2-10	Terminal 1	6

SOCKET S1	1	Driver 1	T2-3
SOCKET S1	2	Driver 1	T2-10
SOCKET S1	7	Power Supply - 1	GND

SOCKET T1	1	Terminal 1	4
SOCKET T1	2	Terminal 1	6
SOCKET T1	7	Power Supply - 4	GND

ACTUATOR 2

SOCKET-M2 (MOTOR 2)	1	Driver 2	T1-3
SOCKET-M2 (MOTOR 2)	2	Driver 2	T1-4
SOCKET-M2 (MOTOR 2)	3	Driver 2	T1-1
SOCKET-M2 (MOTOR 2)	4	Driver 2	T1-2

SOCKET-E2 (ENCODER 2)	1	Terminal 1	30
SOCKET-E2 (ENCODER 2)	2	Terminal 1	36
SOCKET-E2 (ENCODER 2)	3	Terminal 1	32
SOCKET-E2 (ENCODER 2)	4	Terminal 1	31-1-33-35
SOCKET-E2 (ENCODER 2)	5	Terminal 1	34

Driver 2	T2-1	Power Supply - 1	GND
Driver 2	T2-2	Power Supply - 1	24 VDC
Driver 2	T2-3	Terminal 1	22
Driver 2	T2-10	Terminal 1	24

SOCKET S2	1	Driver 2	T2-3
SOCKET S2	2	Driver 2	T2-10
SOCKET S2	7	Power Supply - 1	GND

SOCKET T2	1	Terminal 1	22
SOCKET T2	2	Terminal 1	24
SOCKET T2	7	Power Supply - 4	GND

ACTUATOR 3

SOCKET-M3 (MOTOR 3)	1	Driver 3	T1-3
SOCKET-M3 (MOTOR 3)	2	Driver 3	T1-4
SOCKET-M3 (MOTOR 3)	3	Driver 3	T1-1
SOCKET-M3 (MOTOR 3)	4	Driver 3	T1-2

SOCKET-E3 (ENCODER 3)	1	Terminal 1	62
SOCKET-E3 (ENCODER 3)	2	Terminal 1	68
SOCKET-E3 (ENCODER 3)	3	Terminal 1	64
SOCKET-E3 (ENCODER 3)	4	Terminal 1	63-65-67-69
SOCKET-E3 (ENCODER 3)	5	Terminal 1	66

Driver 3	T2-1	Power Supply - 2	GND
Driver 3	T2-2	Power Supply - 2	24 VDC
Driver 3	T2-3	Terminal 1	54
Driver 3	T2-10	Terminal 1	56

SOCKET S3	1	Driver 3	T2-3
SOCKET S3	2	Driver 3	T2-10
SOCKET S3	7	Power Supply - 2	GND

SOCKET T3	1	Terminal 1	54
SOCKET T3	2	Terminal 1	56
SOCKET T3	7	Power Supply - 4	GND

ACTUATOR 4

SOCKET-M4 (MOTOR 4)	1	Driver 4	T1-3
SOCKET-M4 (MOTOR 4)	2	Driver 4	T1-4
SOCKET-M4 (MOTOR 4)	3	Driver 4	T1-1
SOCKET-M4 (MOTOR 4)	4	Driver 4	T1-2

SOCKET-E4 (ENCODER 4)	1	Terminal 1	80
SOCKET-E4 (ENCODER 4)	2	Terminal 1	86
SOCKET-E4 (ENCODER 4)	3	Terminal 1	82
SOCKET-E4 (ENCODER 4)	4	Terminal 1	81-51-83-85
SOCKET-E4 (ENCODER 4)	5	Terminal 1	84

Driver 4	T2-1	Power Supply - 2	GND
Driver 4	T2-2	Power Supply - 2	24 VDC
Driver 4	T2-3	Terminal 1	72
Driver 4	T2-10	Terminal 1	74

SOCKET S4	1	Driver 4	T2-3
SOCKET S4	2	Driver 4	T2-10
SOCKET S4	7	Power Supply - 2	GND

SOCKET T4	1	Terminal 1	72
SOCKET T4	2	Terminal 1	74
SOCKET T4	7	Power Supply - 4	GND

ACTUATOR 5

SOCKET-M5 (MOTOR 5)	1	Driver 5	T1-3
SOCKET-M5 (MOTOR 5)	2	Driver 5	T1-4
SOCKET-M5 (MOTOR 5)	3	Driver 5	T1-1
SOCKET-M5 (MOTOR 5)	4	Driver 5	T1-2

SOCKET-E5 (ENCODER 5)	1	Terminal 2	30
SOCKET-E5 (ENCODER 5)	2	Terminal 2	36
SOCKET-E5 (ENCODER 5)	3	Terminal 2	32
SOCKET-E5 (ENCODER 5)	4	Terminal 2	31-1-33-35
SOCKET-E5 (ENCODER 5)	5	Terminal 2	34

Driver 5	T2-1	Power Supply - 3	GND
Driver 5	T2-2	Power Supply - 3	24 VDC
Driver 5	T2-3	Terminal 2	22
Driver 5	T2-10	Terminal 2	24

SOCKET S5	1	Driver 5	T2-3
SOCKET S5	2	Driver 5	T2-10
SOCKET S5	7	Power Supply - 3	GND

SOCKET T5	1	Terminal 2	22
SOCKET T5	2	Terminal 2	24
SOCKET T5	7	Power Supply - 5	GND

ACTUATOR 6

SOCKET-M6 (MOTOR 6)	1	Driver 6	T1-3
SOCKET-M6 (MOTOR 6)	2	Driver 6	T1-4
SOCKET-M6 (MOTOR 6)	3	Driver 6	T1-1
SOCKET-M6 (MOTOR 6)	4	Driver 6	T1-2

SOCKET-E6 (ENCODER 6)	1	Terminal 2	62
SOCKET-E6 (ENCODER 6)	2	Terminal 2	68
SOCKET-E6 (ENCODER 6)	3	Terminal 2	64
SOCKET-E6 (ENCODER 6)	4	Terminal 2	63-65-67-69
SOCKET-E6 (ENCODER 6)	5	Terminal 2	66

Driver 6	T2-1	Power Supply - 3	GND
Driver 6	T2-2	Power Supply - 3	24 VDC
Driver 6	T2-3	Terminal 2	54
Driver 6	T2-10	Terminal 2	56

SOCKET S6	1	Driver 6	T2-3
SOCKET S6	2	Driver 6	T2-10
SOCKET S6	7	Power Supply - 3	GND

SOCKET T6	1	Terminal 2	54
SOCKET T6	2	Terminal 2	56
SOCKET T6	7	Power Supply - 5	GND

SOCKET

MOTOR 1	SOCKET M1
ENCODER 1	SOCKET E1
SOCKET S1	SOCKET T1

MOTOR 4	SOCKET M4
ENCODER 4	SOCKET E4
SOCKET S4	SOCKET T4

MOTOR 2	SOCKET M2
ENCODER 2	SOCKET E2
SOCKET S2	SOCKET T2

MOTOR 5	SOCKET M5
ENCODER 5	SOCKET E5
SOCKET S5	SOCKET T5

MOTOR 3	SOCKET M3
ENCODER 3	SOCKET E3
SOCKET S3	SOCKET T3

MOTOR 6	SOCKET M6
ENCODER 6	SOCKET E6
SOCKET S6	SOCKET T6

APPENDIX D

VISUALBASIC PROGRAM FOR SIMULATION OF THE HEXAPOD

motion_h.frm --

```
Private Sub Command1_Click()
konc = "kinematic1": Call kinematic1: Form1.Print c1 + konc
End Sub

Private Sub Command2_Click()
If konc = "kinematic1" Then konc = "result1": Call result1: Print c1 + konc + "  Error : "; kerror
End Sub

Private Sub Command3_Click()
If konc = "result1" Then konc = "kinetic1": Call kinetic1: Print c1 + konc
End Sub

Private Sub Command4_Click()
If konc = "kinetic1" Then konc = "result2": Call result2: Print c1 + konc; "  Error : "; kerror
End Sub

Private Sub Command5_Click()
Print c1, "Make this analysis by GUI using CosmosWorks"
End Sub

Private Sub Form_Click()
Cls
End Sub

Private Sub Form_Load()
fl0 = "d:\hexapod\motion\"
naxis(1) = 0: naxis(2) = 1: naxis(3) = 2
naxis(4) = 3: naxis(5) = 4: naxis(6) = 5
WindowState = 2: pi = 4 * Atn(1): AutoRedraw = True
c1 = "                              ": c1 = c1 + c1 + c1
Form1.Caption = "Hexapod- Integrated Analysis"
Command1.Caption = "Inverse Kinematics"
Command2.Caption = "Kinematic Workspace Evaluation"
Command3.Caption = "Forward Kinetics"
Command4.Caption = "Kinetic Workspace Evaluation"
Command5.Caption = "Rigidity Workspace Evaluation"
Timer1.Enabled = False
End Sub
Private Sub Timer1_Timer()
ktimer = ktimer + 1: If ktimer = 1 Then ktimer = 0: Timer1.Enabled = False
End Sub
```

motion_h.bas --

```
Public Const nmax1 = 1024, ramax = 50.8, ujmax = 45, v0max = 2.28, famax = 225
Public fl0 As String, c1 As String, pi As Double, dt As Double, mc(nmax1, 13) As String
Public nsim As Long, tsim As Double
Public xb As Double, yb As Double, zb As Double, rb(6, 6) As Double
Public xw As Double, yw As Double, zw As Double, nmove As Integer
Public elem As Object, td As Double, n1 As Long
```

```
Public tmove(nmax1), ts(nmax1) As Double, ras(6, nmax1) As Double, uja(6, nmax1) As Double,
ujb(6, nmax1) As Double
Public ra(6, nmax1) As Double, da(6, nmax1) As Double, kerror As Integer
Public xenc As Double, yenc As Double, zenc As Double, rxenc As Double, ryenc As Double, rzenc
As Double
Public f(nmax1) As Double, fa(6, nmax1) As Double
Public m0(6, nmax1) As Long, vm0(6, nmax1) As Long, tac(nmax1) As Double, tde(nmax1) As
Double
Public tp(nmax1) As Double, v(nmax1) As Double, mt(nmax1) As String, kadlink As Integer
Public nd As Long, konc As String, na As Integer, naxis(6) As Integer, ktimer As Integer, konmove
As Integer
Public r1enc As Double, r2enc As Double

Sub inp1()
  Open fl0 + "data_encoder.txt" For Input As 1
  Line Input #1, xc: Input #1, r1enc, r2enc: Line Input #1, xc
  Input #1, xenc, yenc, zenc, rxenc, ryenc, rzenc: Close #1
  xa = xenc = ya = yenc: za = zenc: rxa = rxenc: rya = ryenc: rza = rzenc
  xb = 0: yb = 0: zb = 0
'---
  Open fl0 + "data_motion.txt" For Input As 1: nmove = 0: tsim = 0
  Input #1, xc, kmint, xc, kadlink, xc, dt
5 Line Input #1, xc: If xc <> "hexapod" Then GoTo 5
10 Input #1, t: If t = "-1" Then Close #1: nsim = CLng(tsim / dt): GoTo 20
  nmove = nmove + 1: tmove(nmove) = t: tsim = tsim + t
  For n = 1 To 5: Input #1, mc(nmove, n): Next n
  If mc(nmove, 1) = "wait" Then mt(nmove) = mc(nmove, 1): GoTo 10
  If mc(nmove, 3) = "w" Then For n = 6 To 13: Input #1, mc(nmove, n): Next n
  If mc(nmove, 3) = "r" Then For n = 6 To 10: Input #1, mc(nmove, n): Next n
  mt(nmove) = mc(nmove, 3): racc = Val(mc(nmove, 1)): tac(nmove) = racc * t
  rdec = Val(mc(nmove, 2)): tde(nmove) = rdec * t: GoTo 10
20
End Sub
Sub result2()
Set cmaddin = GetObject(, "cmotionswapi.cmotionswaddin"): Set ms =
cmaddin.ActiveAssembly.Mechanism
nf = ms.Simulation.frames: np = ms.Plots.Count
For n = 1 To np: fl = "z" + Mid(Str(n), 2): Call ms.Plots.Item(n, elem): Call elem.ExportCSVFile(fl):
Next n
For n = 1 To np
 fl = fl0 + "z" + Mid(Str(n), 2) + ".csv": Open fl For Input As 1
 Line Input #1, xc1: If Mid(xc1, 2, 5) <> "Force" Then Close #1: GoTo 12
 Line Input #1, xc2: xc1 = Mid(xc1, 2, Len(xc1) - 2)
 For m = 1 To nf: Input #1, ts(m), x: GoSub 40: Next m: Close #1
12 Next n
  GoSub 50: Exit Sub

40 If xc1 = "Force - Mag-a1" Then fa(1, m) = x: Return
  If xc1 = "Force - Mag-a2" Then fa(2, m) = x: Return
  If xc1 = "Force - Mag-a3" Then fa(3, m) = x: Return
  If xc1 = "Force - Mag-a4" Then fa(4, m) = x: Return
  If xc1 = "Force - Mag-a5" Then fa(5, m) = x: Return
  If xc1 = "Force - Mag-a6" Then fa(6, m) = x: Return
   Return
'---
50 td = 0: fmax = 0
    For kmove = 1 To nmove
   t3 = tmove(kmove): t = td + 0.5 * t3: m = CInt(t / dt) + 1
```

```
   If mc(kmove, 1) = "wait" Then GoTo 52
   racc = Val(mc(kmove, 1)): rdec = Val(mc(kmove, 2))
    For na = 1 To 6
    If fmax < Abs(fa(na, m)) Then fmax = Abs(fa(na, m))
    vm = Abs(da(na, kmove)) / (1 - 0.5 * racc - 0.5 * rdec) / t3: fm = -160.7143 * vm + 466.0714
    If Abs(fa(na, m)) > fm Then kerror1 = 10000: Form1.Print c1, kerror1, kmove, fa(na, m), vm
     Next na

   td = td + t3
52 Next kmove
   If fmax > famax Then kerror1 = 10000
   Form1.Print c1 + "fmax= "; fmax
   kerror = kerror + kerror1
   Return
End Sub
Sub kinetic1()
'---
Set cmaddin = GetObject(, "cmotionswapi.cmotionswaddin"): Set ms =
cmaddin.ActiveAssembly.Mechanism
Call ms.DeleteSimulation
'Call inp2
'---
td = 0: fxp = 0: typ = 0: fzp = 0: expfx = "": expfy = "": expfz = ""
For kmove = 1 To nmove
 t = tmove(kmove)
 If mc(kmove, 1) = "wait" Then fx = Val(mc(kmove, 3)): fy = Val(mc(kmove, 4)):  fz =
Val(mc(kmove, 5)): GoTo 7
 If mc(kmove, 3) = "w" Then fx = Val(mc(kmove, 11)): fy = Val(mc(kmove, 12)):  fz =
Val(mc(kmove, 13)): GoTo 7
 fx = Val(mc(kmove, 8)): fy = Val(mc(kmove, 9)):  fz = Val(mc(kmove, 10))
7 If fx - fxp <> 0 Then expfx = expfx + "+STEP(TIME," + Str(td) + ",0," + Str(td + dt) + "," + Str(fx -
fxp) + ")"
  If fy - fyp <> 0 Then expfy = expfy + "+STEP(TIME," + Str(td) + ",0," + Str(td + dt) + "," + Str(fy -
fyp) + ")"
  If fz - fzp <> 0 Then expfz = expfz + "+STEP(TIME," + Str(td) + ",0," + Str(td + dt) + "," + Str(fz -
fzp) + ")"
  fxp = fx: fyp = fy: fzp = fz: td = td + t

 Next kmove
 Call ms.GetElementByName("ForceAO", elem): Call elem.Function.SetExpression(expfy)
 Call ms.GetElementByName("ForceAO2", elem): Call elem.Function.SetExpression(expfx)
 Call ms.GetElementByName("ForceAO3", elem): Call elem.Function.SetExpression(expfz)


Call ms.GetElementByName("Motion", elem)
elem.Motions.TranslateX.MotionType = 2: elem.Motions.TranslateY.MotionType = 2
elem.Motions.TranslateZ.MotionType = 2: elem.Motions.RotateX.MotionType = 2
elem.Motions.RotateY.MotionType = 2: elem.Motions.RotateZ.MotionType = 2

na = 1: jc = "a1": GoSub 40
na = 2: jc = "a2": GoSub 40
na = 3: jc = "a3": GoSub 40
na = 4: jc = "a4": GoSub 40
na = 5: jc = "a5": GoSub 40
na = 6: jc = "a6": GoSub 40

Call ms.Simulate(tsim, nsim): Exit Sub
'---
```

```
40 n = 1: np = 0: tp(0) = 0: v(0) = 0
 For kmove = 1 To nmove
 t3 = tmove(kmove): d0 = da(na, kmove): If mc(kmove, 1) = "wait" Then d0 = 0
 racc = Val(mc(kmove, 1)): rdec = Val(mc(kmove, 2))
 If d0 = 0 Then
  ns1 = 2: dt1 = t3 / ns1: t = dt1
  For m = 1 To ns1
  v(n) = v(np): tp(n) = tp(np) + t: n = n + 1: t = t + dt1
  Next m: np = n - 1: GoTo 12
  End If


 t1 = racc * t3: t2 = (1 - rdec) * t3
 vm = d0 / (0.5 * t1 + t2 - t1 + 0.5 * (t3 - t2)): v(0) = 0

 ns1 = 2: dt1 = t1 / ns1: t = dt1
 For m = 1 To ns1
  v(n) = v(np) + 0.5 * vm * t ^ 2 / t1: tp(n) = tp(np) + t: n = n + 1: t = t + dt1
 Next m: nd = m - 1

 ns1 = 2: dt1 = (t2 - t1) / ns1: t = t1 + dt1
 For m = nd + 1 To nd + ns1
  v(n) = v(np) + 0.5 * vm * t1 + vm * (t - t1): tp(n) = tp(np) + t: n = n + 1: t = t + dt1
 Next m: nd = m - 1: vp = v(n - 1)

 ns1 = 2: dt1 = (t3 - t2) / ns1: t = t2 + dt1
 For m = nd + 1 To nd + ns1
  v(n) = v(np) + 0.5 * vm * t1 + vm * (t2 - t1) + vm * (t - t2) - 0.5 * vm * (t - t2) ^ 2 / (t3 - t2)
  tp(n) = tp(np) + t: n = n + 1: t = t + dt1
 Next m: np = n - 1

12 nd = n - 1

 Next kmove

 For n = 0 To nd
  If Abs(v(n)) < 0.000001 Then v(n) = 0
  v(n) = v(n) + ra(na, 0)
 Next n
  Call ms.GetElementByName(jc, elem): elem.Motions.TranslateZ.MotionType = 3
  Call elem.Motions.TranslateZ.Function.SetDataPoints(2, nd + 1, tp, v): Return
End Sub
Sub result1()
Set cmaddin = GetObject(, "cmotionswapi.cmotionswaddin"): Set ms =
cmaddin.ActiveAssembly.Mechanism
nf = ms.Simulation.frames: np = ms.Plots.Count: konmove = 1

For n = 1 To np: fl = "z" + Mid(Str(n), 2): Call ms.Plots.Item(n, elem): Call elem.ExportCSVFile(fl):
Next n

For n = 1 To np
 fl = fl0 + "z" + Mid(Str(n), 2) + ".csv": Open fl For Input As 1
 Line Input #1, xc1: If Mid(xc1, 2, 5) = "Force" Then Close #1: GoTo 12
 Line Input #1, xc2: xc1 = Mid(xc1, 2, Len(xc1) - 2)
 For m = 1 To nf: Input #1, ts(m), x: GoSub 40: Next m: Close #1
12 Next n
  GoSub 50: Call save1: Exit Sub
'---
```

```
40 If xc1 = "Trans Disp - Z-a1" Then ras(1, m) = x: Return
   If xc1 = "Trans Disp - Z-a2" Then ras(2, m) = x: Return
   If xc1 = "Trans Disp - Z-a3" Then ras(3, m) = x: Return
   If xc1 = "Trans Disp - Z-a4" Then ras(4, m) = x: Return
   If xc1 = "Trans Disp - Z-a5" Then ras(5, m) = x: Return
   If xc1 = "Trans Disp - Z-a6" Then ras(6, m) = x: Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then uja(1, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement2" Then uja(2, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement3" Then uja(3, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement4" Then uja(4, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement5" Then uja(5, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement6" Then uja(6, m) = Abs(90 - x): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(1, m) = Abs(x - 90): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(2, m) = Abs(x - 90): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(3, m) = Abs(x - 90): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(4, m) = Abs(x - 90): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(5, m) = Abs(x - 90): Return
   If xc1 = "Angular Disp - Mag-ADisplacement" Then ujb(6, m) = Abs(x - 90): Return
  Return
'---
50 kerror1 = 0: kerror2 = 0: kerror3 = 0
   ymax1 = ras(1, 1): ymin1 = ymax1: ymax2 = uja(1, 1)
  For n = 1 To nf
    For na = 1 To 6
     If ymax1 < ras(na, n) Then ymax1 = ras(na, n)
     If ymin1 > ras(na, n) Then ymin1 = ras(na, n)
     If ymax2 < uja(na, n) Then ymax2 = uja(na, n)
     If ymax2 < ujb(na, n) Then ymax2 = ujb(na, n)
   Next na
    If Abs(ymin1) < 0.000001 Then ymin1 = 0
   Next n
 If ymin1 < 0 Or ymax1 > ramax Then kerror1 = 1: Form1.Print c1, kmove, kerror1, ymin1, ymax1
muj = c1 + "ra:" + Str(ymin1) + "," + Str(ymax1)

 If ymax2 > ujmax Then kerror2 = 10: Form1.Print c1, kmove, kerror2, ymax2
  muj = muj + "  uj:" + Str(ymax2)
Form1.Print muj
'---
td = 0
For na = 1 To 6: ra(na, 0) = ras(na, 1): Next na
For n = 1 To nmove
 td = td + tmove(n): m = CInt(td / dt) + 1
 For na = 1 To 6
  ra(na, n) = ras(na, m): da(na, n) = ra(na, n) - ra(na, n - 1)
  If Abs(da(na, n)) < 0.000001 Then da(na, n) = 0
 Next na
 Next n
'---
vmax = 0
For kmove = 1 To nmove
  t = tmove(kmove): If mc(kmove, 1) = "wait" Then GoTo 45
  racc = Val(mc(kmove, 1)): rdec = Val(mc(kmove, 2))
  For na = 1 To 6
   vm = da(na, kmove) / (1 - 0.5 * racc - 0.5 * rdec) / t
   If vmax < Abs(vm) Then vmax = Abs(vm)
   If Abs(vm) > v0max Then kerror3 = 100: Form1.Print c1, kmove, kerror3, na, vm
  Next na
45 Next kmove
```

```
Form1.Print c1, "vmax="; vmax
kerror = kerror1 + kerror2 + kerror3: Return
End Sub
Sub kinematic1()
'Set swapp = GetObject(, "sldworks.application"):Set asmbl = swapp.ActivateDoc2(fla +
".SLDASM", False, n1)
Set cmaddin = GetObject(, "cmotionswapi.cmotionswaddin"): Set ms =
cmaddin.ActiveAssembly.Mechanism
  Call inp1: td = 0: expx = "0": expy = "0": expz = "0": exprx = "0": expry = "0": exprz = "0"
  expfx = "": expfy = "": expfz = "": fxp = 0: fyp = 0: fzp = 0: konmove = 0
  If xenc <> 0 Then expx = Str(xenc)
  If yenc <> 0 Then expy = Str(yenc)
  If zenc <> 0 Then expz = Str(zenc)
  If rxenc <> 0 Then exprx = Str(rxenc) + "D"
  If ryenc <> 0 Then expry = Str(ryenc) + "D"
  If rzenc <> 0 Then exprz = Str(rzenc) + "D"
For kmove = 1 To nmove
 t = tmove(kmove): If mc(kmove, 1) = "wait" Then td = td + t: GoTo 32
 If mc(kmove, 3) = "w" Then
   x = Val(mc(kmove, 4)): y = Val(mc(kmove, 5)): z = Val(mc(kmove, 6))
   rx = Val(mc(kmove, 7)): ry = Val(mc(kmove, 8)): rz = Val(mc(kmove, 9)): GoTo 22
 End If
   xb = Val(mc(kmove, 4)): yb = Val(mc(kmove, 5)): zb = Val(mc(kmove, 6))
   Call rb_m: x = xw: y = yw: z = zw: rx = 0: ry = 0: rz = 0
22 If x <> 0 Then expx = expx + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(x) + ")"
  If y <> 0 Then expy = expy + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(y) + ")"
  If z <> 0 Then expz = expz + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(z) + ")"
  If rx <> 0 Then exprx = exprx + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(rx) + "D)"
  If ry <> 0 Then expry = expry + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(ry) + "D)"
  If rz <> 0 Then exprz = exprz + "+STEP(TIME," + Str(td) + ",0," + Str(td + t) + "," + Str(rz) + "D)"
  xenc = xenc + x: yenc = yenc + y: zenc = zenc + z
  rxenc = rxenc + rx: ryenc = ryenc + ry: rzenc = rzenc + rz: td = td + t
32 Next kmove
  ms.DeleteSimulation
Call ms.GetElementByName("a1", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("a2", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("a3", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("a4", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("a5", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("a6", elem): elem.Motions.TranslateZ.MotionType = 2
Call ms.GetElementByName("Motion", elem)
elem.Motions.TranslateX.MotionType = 3: elem.Motions.TranslateY.MotionType = 3
elem.Motions.TranslateZ.MotionType = 3: elem.Motions.RotateX.MotionType = 3
elem.Motions.RotateY.MotionType = 3: elem.Motions.RotateZ.MotionType = 3
  Call elem.Motions.TranslateX.Function.SetExpression(expx)
  Call elem.Motions.TranslateY.Function.SetExpression(expy)
  Call elem.Motions.TranslateZ.Function.SetExpression(expz)
  Call elem.Motions.RotateX.Function.SetExpression(exprx)
  Call elem.Motions.RotateY.Function.SetExpression(expry)
  Call elem.Motions.RotateZ.Function.SetExpression(exprz)

GoTo 55
 Call ms.GetElementByName("Revolute", elem): elem.Motions.RotateZ.MotionType = 3
 elem.Motions.RotateZ.Function.SetConstant (0)
 Call ms.GetElementByName("Revolute2", elem): elem.Motions.RotateZ.MotionType = 3
 elem.Motions.RotateZ.Function.SetConstant (0)
55
 Call ms.Simulate(tsim, nsim)
```

```
End Sub
Sub rb_m()
Cx = Cos(rxenc * pi / 180): sx = Sin(rxenc * pi / 180): Cy = Cos(ryenc * pi / 180)
sy = Sin(ryenc * pi / 180): cz = Cos(rzenc * pi / 180): sz = Sin(rzenc * pi / 180)
rb(1, 1) = Cy * cz: rb(1, 2) = -Cy * sz: rb(1, 3) = sy
rb(2, 1) = sx * sy * cz + Cx * sz: rb(2, 2) = -sx * sy * sz + Cx * cz: rb(2, 3) = -sx * Cy
rb(3, 1) = -Cx * sy * cz + sx * sz: rb(3, 2) = Cx * sy * sz + sx * cz: rb(3, 3) = Cx * Cy
xw = rb(1, 1) * xb + rb(1, 2) * yb + rb(1, 3) * zb
yw = rb(2, 1) * xb + rb(2, 2) * yb + rb(2, 3) * zb
zw = rb(3, 1) * xb + rb(3, 2) * yb + rb(3, 3) * zb
End Sub
Sub wait1()
If td = 0 Then Exit Sub
Form1.Timer1.Interval = td * 1000: Form1.Timer1.Enabled = True
10 If Form1.Timer1.Enabled = True Then DoEvents: GoTo 10
End Sub
Sub save1()
For n = 1 To nmove
 For na = 1 To 6
 m0(na, n) = CLng(da(na, n) / 0.00075): vm0(na, n) = CLng(Abs(m0(na, n)) / (tmove(n) - 0.5 * tac(n)
- 0.5 * tde(n)))
 Next na
Next n
Open fl0 + "data_forward_h.txt" For Output As 1
Print #1, "dt," + Str(dt)
For na = 1 To 6: Print #1, Str(ra(na, 0)): Next na
Print #1, "---": Print #1, Str(nmove)
For n = 1 To nmove
 If mt(n) = "wait" Then Print #1, mt(n) + "," + Str(tmove(n)): GoTo 60
 Print #1, "m," + Str(tmove(n))
 For na = 1 To 6
 xc = Str(naxis(na)) + "," + Str(da(na, n)) + "," + Str(m0(na, n))
 xc = xc + "," + Str(vm0(na, n)) + "," + Str(tac(n)) + "," + Str(tde(n)): Print #1, xc
 Next na
60 Next n
   Print #1, "---"
   Print #1, Str(xenc) + "," + Str(yenc) + "," + Str(zenc) + "," + Str(rxenc) + "," + Str(ryenc) + "," +
Str(rzenc)
   Close #1
Form1.Print c1, "Enc :"; xenc, yenc, zenc, rxenc, ryenc, rzenc
End Sub
```

APPENDIX E
MATLAB PROGRAM FOR ANALYTICAL SOLUTION OF THE HEXAPOD

a1.m ---

```
clc,clear
a1_constants
load('d1.txt');
x1=d1(1,1)/1000;;y1=d1(1,2)/1000;;z1=d1(1,3)/1000;
fi=d1(1,4)*pi/180;;th=d1(1,5)*pi/180;;psi=d1(1,6)*pi/180;
x1d=d1(2,1)/1000;;y1d=d1(2,2)/1000;;z1d=d1(2,3)/1000;
fid=d1(2,4)*pi/180;;thd=d1(2,5)*pi/180;;psid=d1(2,6)*pi/180;
x1dd=d1(3,1)/1000;;y1dd=d1(3,2)/1000;;z1dd=d1(3,3)/1000;
fidd=d1(3,4)*pi/180;;thdd=d1(3,5)*pi/180;;psidd=d1(3,6)*pi/180;
fxb=d1(4,1);fyb=d1(4,2);fzb=d1(4,3);
%---------------------
a1_solve
ra*1000
F
```

a1_constants.m ---

```
g=[0;-9810;0]*1e-3; %m/s^2
l0=222.3983e-3; %m
lb=75.7412e-3;lt=34.4187e-3; % m
mb=963.98e-3;mt=104.91e-3;mp=3482.1e-3; %kg
cp=[0;-1.1289;0]*1e-3;
b1=[-38.564;-31;-143.923]*1e-3; b2=[-143.923;-31;38.564]*1e-3; b3=[-105.359;-31;105.359]*1e-3;
%m
b4=[105.3589;-31;105.3589]*1e-3;  b5=[143.923;-31;38.564]*1e-3;  b6=[38.564;-31;-143.923]*1e-3;
%m
p1=[-50;33.3;-86.6025]*1e-3;p2=[-100;33.3;0]*1e-3;p3=[-50;33.3;86.6025]*1e-
3;p4=[50;33.3;86.6025]*1e 3; p5=[100;33.3;0]*1e-3;p6=[50;33.3;-86.6025]*1e-3; %m
Ipg=[140.96,0,0;0,267.73,0;0,0,140.99]*1e-4;Ip=Ipg;%kg*m^2,w.r.t. c.o.g.
It=[1.56,0,0;0,0.0473,0;0,0,1.57]*1e-4;%kg*m^2,w.r.t.c.o.g.
Ibg=[9.73,0,0;0,2.26,0;0,0,9.74]*1e-4;Ib=Ibg;%kg*m^2, w.r.t. c.o.g.
Ib(1,1)=lb^2*mb+Ibg(1,1);Ib(3,3)=Ib(1,1);  % w.r.t center of rotation
re=[0;-88;0]*1e-3;
```

a1_solve.m ---

```
a=cp;skew;cps=as;
cfi=cos(fi);sfi=sin(fi);cth=cos(th);sth=sin(th);cpsi=cos(psi);spsi=sin(psi);
cx=cfi;cy=cth; cz=cpsi; sx=sfi; sy=sth; sz=spsi;
r=[cy*cz,-cy*sz,sy;sx*sy*cz+cx*sz,-sx*sy*sz+cx*cz,-sx*cy;-
cx*sy*cz+sx*sz,cx*sy*sz+sx*cz,cx*cy];   %Cosmos
ta=[x1;-0.27888+y1;z1];qt=[fi;th;psi];q=[ta;qt];
ea=[cpsi*cth,spsi,0;-spsi*cth,cpsi,0;sth,0,1]; %Cosmos
qtd=[fid;thd;psid];wp=ea*qtd;w=r*wp;a=w;skew;ws=as;
a=p1;skew;p1s=as;a=p2;skew;p2s=as;a=p3;skew;p3s=as;
a=p4;skew;p4s=as;a=p5;skew;p5s=as;a=p6;skew;p6s=as;

tad=[x1d;y1d;z1d];qd=[tad;w];
ead=[-spsi*cth*psid-cpsi*sth*thd, cpsi*psid, 0;-cpsi*cth*psid+spsi*sth*thd,-spsi*psid,0;cth*thd,0,0];
%Cosmos
qtdd=[fidd;thdd;psidd]; %qtdd=[x1dd;y1dd;z1dd];
```

```
wpd=ead*qtd+ea*qtdd;
 a=wpd;skew;wpds=as;wds=r*wpds*transpose(r);wd=[wds(3,2);wds(1,3);wds(2,1)];

qdd=[x1dd;y1dd;z1dd;wd];
l1=ta+r*p1-b1;l2=ta+r*p2-b2;l3=ta+r*p3-b3;l4=ta+r*p4-b4;l5=ta+r*p5-b5;l6=ta+r*p6-b6;
lm1=sqrt(l1(1,1)^2+l1(2,1)^2+l1(3,1)^2);lm2=sqrt(l2(1,1)^2+l2(2,1)^2+l2(3,1)^2);lm3=sqrt(l3(1,1)^2
+l3(2,1)^2+l3(3,1)^2);
lm4=sqrt(l4(1,1)^2+l4(2,1)^2+l4(3,1)^2);lm5=sqrt(l5(1,1)^2+l5(2,1)^2+l5(3,1)^2);lm6=sqrt(l6(1,1)^2
+l6(2,1)^2+l6(3,1)^2);
ra1=lm1-l0;ra2=lm2-l0;ra3=lm3-l0;ra4=lm4-l0;ra5=lm5-l0;ra6=lm6-l0;ra=[ra1;ra2;ra3;ra4;ra5;ra6];

qp1=ta+r*p1;qp2=ta+r*p2;qp3=ta+r*p3;qp4=ta+r*p4;qp5=ta+r*p5;qp6=ta+r*p6;
pb1=r*p1;pb2=r*p2;pb3=r*p3;pb4=r*p4;pb5=r*p5;pb6=r*p6;
qp1d=tad+w.*pb1;qp2d=tad+w.*pb2;qp3d=tad+w.*pb3;qp4d=tad+w.*pb4;qp5d=tad+w.*pb5;qp6d=t
ad+w.*pb6;

n1=l1/lm1;n2=l2/lm2;n3=l3/lm3;n4=l4/lm4;n5=l5/lm5;n6=l6/lm6;
a=n1;skew;n1s=as;a=n2;skew;n2s=as;a=n3;skew;n3s=as;a=n4;skew;
n4s=as;a=n5;skew;n5s=as;a=n6;skew;n6s=as;
l1d=transpose(n1)*qp1d;l2d=transpose(n2)*qp2d;l3d=transpose(n3)*qp3d;l4d=transpose(n4)*qp4d;l5
d=transpose(n5)*qp5d;l6d=transpose(n6)*qp6d;
wl1=n1.*qp1d/lm1;wl2=n2.*qp2d/lm2;wl3=n3.*qp3d/lm3;wl4=n4.*qp4d/lm4;wl5=n5.*qp5d/lm5;wl
6=n6.*qp6d/lm6;

vt1=qp1d+wl1.*(lt*n1);vt2=qp2d+wl2.*(lt*n2);vt3=qp3d+wl3.*(lt*n3);vt4=qp4d+wl4.*(lt*n4);vt5=q
p5d+wl5.*(lt*n5);vt6=qp6d+wl6.*(lt*n6);
vb1=wl1.*(lb*n1);vb2=wl2.*(lb*n2);vb3=wl3.*(lb*n3);vb4=wl4.*(lb*n4);vb5=wl5.*(lb*n5);vb6=wl
6.*(lb*n6);
I=eye(3,3);
qp1dd=[I, r*transpose(p1s)*transpose(r)]*qdd+ws^2*r*p1;qp2dd=[I,
r*transpose(p2s)*transpose(r)]*qdd+ws^2*r*p2;
qp3dd=[I, r*transpose(p3s)*transpose(r)]*qdd+ws^2*r*p3;qp4dd=[I,
r*transpose(p4s)*transpose(r)]*qdd+ws^2*r*p4;
qp5dd=[I, r*transpose(p5s)*transpose(r)]*qdd+ws^2*r*p5;qp6dd=[I,
r*transpose(p6s)*transpose(r)]*qdd+ws^2*r*p6;

M11=transpose(I+lt*n1s^2/lm1)*mt*(I+lt*n1s^2/lm1);M12=transpose(I+lt*n2s^2/lm2)*mt*(I+lt*n2s
^2/lm2);
M13=transpose(I+lt*n3s^2/lm3)*mt*(I+lt*n3s^2/lm3);M14=transpose(I+lt*n4s^2/lm4)*mt*(I+lt*n4s
^2/lm4);
M15=transpose(I+lt*n5s^2/lm5)*mt*(I+lt*n5s^2/lm5);M16=transpose(I+lt*n6s^2/lm6)*mt*(I+lt*n6s
^2/lm6);
M21=((It+Ib)*transpose(n1s)*n1s)/lm1^2;M22=((It+Ib)*transpose(n2s)*n2s)/lm2^2;
M23=((It+Ib)*transpose(n3s)*n3s)/lm3^2;M24=((It+Ib)*transpose(n4s)*n4s)/lm4^2;
M25=((It+Ib)*transpose(n5s)*n5s)/lm5^2;M26=((It+Ib)*transpose(n6s)*n6s)/lm6^2;

Qmt1g=transpose(I+lt*n1s^2/lm1)*mt*g;Qmt2g=transpose(I+lt*n2s^2/lm2)*mt*g;Qmt3g=transpose(
I+lt*n3s^2/lm3)*mt*g;
Qmt4g=transpose(I+lt*n4s^2/lm4)*mt*g;Qmt5g=transpose(I+lt*n5s^2/lm5)*mt*g;Qmt6g=transpose(
I+lt*n6s^2/lm6)*mt*g;
Qmb1g=transpose(lb*transpose(n1s)*n1s/lm1)*mb*g;Qmb2g=transpose(lb*transpose(n2s)*n2s/lm2)
*mb*g;Qmb3g=transpose(lb*transpose(n3s)*n3s/lm3)*mb*g;
Qmb4g=transpose(lb*transpose(n4s)*n4s/lm4)*mb*g;Qmb5g=transpose(lb*transpose(n5s)*n5s/lm5)
*mb*g;Qmb6g=transpose(lb*transpose(n6s)*n6s/lm6)*mb*g;

Ca1=mt*lt/lm1^2*(n1*transpose(qp1d)*transpose(n1s)*n1s+transpose(n1)*qp1d*transpose(n1s)*n1s
+transpose(n1s)*n1s*qp1d*transpose(n1))
```

```
−mt*lt^2/lm1^3*(transpose(n1)*qp1d*transpose(n1s)*n1s+transpose(n1s)*n1s*qp1d*transpose(n1))-
2*((It+Ib)/lm1^3)*(transpose(n1s)*n1s*qp1d*transpose(n1));

Ca2=mt*lt/lm2^2*(n2*transpose(qp2d)*transpose(n2s)*n2s+transpose(n2)*qp2d*transpose(n2s)*n2s
+transpose(n2s)*n2s*qp2d*transpose(n2))

−mt*lt^2/lm2^3*(transpose(n2)*qp2d*transpose(n2s)*n2s+transpose(n2s)*n2s*qp2d*transpose(n2))-
2*((It+Ib)/lm2^3)*(transpose(n2s)*n2s*qp2d*transpose(n2));

Ca3=mt*lt/lm3^2*(n3*transpose(qp3d)*transpose(n3s)*n3s+transpose(n3)*qp3d*transpose(n3s)*n3s
+transpose(n3s)*n3s*qp3d*transpose(n3))

−mt*lt^2/lm3^3*(transpose(n3)*qp3d*transpose(n3s)*n3s+transpose(n3s)*n3s*qp3d*transpose(n3))-
2*((It+Ib)/lm3^3)*(transpose(n3s)*n3s*qp3d*transpose(n3));

Ca4=mt*lt/lm4^2*(n4*transpose(qp4d)*transpose(n4s)*n4s+transpose(n4)*qp4d*transpose(n4s)*n4s
+transpose(n4s)*n4s*qp4d*transpose(n4))

−mt*lt^2/lm4^3*(transpose(n4)*qp4d*transpose(n4s)*n4s+transpose(n4s)*n4s*qp4d*transpose(n4))-
2*((It+Ib)/lm4^3)*(transpose(n4s)*n4s*qp4d*transpose(n4));

Ca5=mt*lt/lm5^2*(n5*transpose(qp5d)*transpose(n5s)*n5s+transpose(n5)*qp5d*transpose(n5s)*n5s
+transpose(n5s)*n5s*qp5d*transpose(n5))

−mt*lt^2/lm5^3*(transpose(n5)*qp5d*transpose(n5s)*n5s+transpose(n5s)*n5s*qp5d*transpose(n5))-
2*((It+Ib)/lm5^3)*(transpose(n5s)*n5s*qp5d*transpose(n5));

Ca6=mt*lt/lm6^2*(n6*transpose(qp6d)*transpose(n6s)*n6s+transpose(n6)*qp6d*transpose(n6s)*n6s
+transpose(n6s)*n6s*qp6d*transpose(n6))

−mt*lt^2/lm6^3*(transpose(n6)*qp6d*transpose(n6s)*n6s+transpose(n6s)*n6s*qp6d*transpose(n6))-
2*((It+Ib)/lm6^3)*(transpose(n6s)*n6s*qp6d*transpose(n6));

qc=ta+r*cp;

Mp=[mp*I,mp*r*transpose(cps)*transpose(r)
   mp*r*cps*transpose(r),mp*r*cps*transpose(cps)*transpose(r)+r*Ip*transpose(r)];
zero=zeros(3,3);
Cp=[zero, zero
   zero,ws*r*Ip*transpose(r)];
Hp=[I,I,I,I,I,I

r*p1s*transpose(r),r*p2s*transpose(r),r*p3s*transpose(r),r*p4s*transpose(r),r*p5s*transpose(r),r*p6s
*transpose(r)];

Mq=Mp+[I;r*p1s*transpose(r)]*(M11+M21)*[I,r*transpose(p1s)*transpose(r)]
   +[I;r*p2s*transpose(r)]*(M12+M22)*[I,r*transpose(p2s)*transpose(r)]
   +[I;r*p3s*transpose(r)]*(M13+M23)*[I,r*transpose(p3s)*transpose(r)]
   +[I;r*p4s*transpose(r)]*(M14+M24)*[I,r*transpose(p4s)*transpose(r)]
   +[I;r*p5s*transpose(r)]*(M15+M25)*[I,r*transpose(p5s)*transpose(r)]
  +[I;r*p6s*transpose(r)]*(M16+M26)*[I,r*transpose(p6s)*transpose(r)];

Cq=Cp*qd+[mp*I;mp*r*transpose(cps)*transpose(r)]*ws^2*r*cp

+[I;r*p1s*transpose(r)]*Ca1*[I,r*transpose(p1s)*transpose(r)]*qd+[I;r*p1s*transpose(r)]*(M11+M2
1)*ws^2*r*p1
```

```
+[I;r*p2s*transpose(r)]*Ca2*[I,r*transpose(p2s)*transpose(r)]*qd+[I;r*p2s*transpose(r)]*(M12+M2
2)*ws^2*r*p2

+[I;r*p3s*transpose(r)]*Ca3*[I,r*transpose(p3s)*transpose(r)]*qd+[I;r*p3s*transpose(r)]*(M13+M2
3)*ws^2*r*p3

+[I;r*p4s*transpose(r)]*Ca4*[I,r*transpose(p4s)*transpose(r)]*qd+[I;r*p4s*transpose(r)]*(M14+M2
4)*ws^2*r*p4

+[I;r*p5s*transpose(r)]*Ca5*[I,r*transpose(p5s)*transpose(r)]*qd+[I;r*p5s*transpose(r)]*(M15+M2
5)*ws^2*r*p5

+[I;r*p6s*transpose(r)]*Ca6*[I,r*transpose(p6s)*transpose(r)]*qd+[I;r*p6s*transpose(r)]*(M16+M2
6)*ws^2*r*p6;

Gq=-[mp*g;mp*r*cps*transpose(r)*g]
   -[I;r*p1s*transpose(r)]*(Qmt1g+Qmb1g)-[I;r*p2s*transpose(r)]*(Qmt2g+Qmb2g)-
[I;r*p3s*transpose(r)]*(Qmt3g+Qmb3g)
   -[I;r*p4s*transpose(r)]*(Qmt4g+Qmb4g)-[I;r*p5s*transpose(r)]*(Qmt5g+Qmb5g)-
[I;r*p6s*transpose(r)]*(Qmt6g+Qmb6g);

fe=[fxb;fyb;fzb];a=re;skew;Gq=Gq-[r*fe;r*(as*fe)];

Hq=[n1,n2,n3,n4,n5,n6

(r*p1s*transpose(r))*n1,(r*p2s*transpose(r))*n2,(r*p3s*transpose(r))*n3,(r*p4s*transpose(r))*n4,(r*
p5s*transpose(r))*n5,(r*p6s*transpose(r))*n6];
F=Hq\(Mq*qdd+Cq+Gq);
```