DESIGN, DEVELOPMENT, MODELING AND CONTROL OF A STEWART PLATFORM

A Thesis Submitted to
the Graduate School of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Electronics and Communication Engineering

by
Ali Bars GÜNDÜZ

July 2020
İZMİR

ACKNOWLEDGMENTS

I wish to express my deepest gratitude to my advisor Prof. Dr. Enver TATLICIOĞLU for his endless help, guidance and motivation. His support and encouragement led me to complete this thesis. It is an honor for me to be his student.

I would like to express my greatest thanks to my employer Dr. Türker KARAYUMAK for his support, efforts and for giving me the chance and providing conditions to work on this project.

I would also like to thank to Deniz KURTOĞLU for her suggestions and help in writing the thesis.

Finally, I am extremely thankful to my family. I wouldn't have succeeded without them.

ABSTRACT

DESIGN, DEVELOPMENT, MODELING AND CONTROL OF A STEWART PLATFORM

Need for a six degrees of freedom test platform to be used in the inertial stabilization development tests of turrets, remote controlled weapon systems or sight systems is present in modern military technology development. To address this research issue, within this thesis study, a Stewart platform is built and is used for stabilization development tests of a sight system.

To give a clear picture of the whole system, mechanical design criterion, electrical architecture, communication between system units and chosen hardware of the designed Stewart platform are presented in detail. After the design and development phases are completed, kinematic model is obtained via utilizing the computer–aided design model of the Stewart platform. The inverse kinematic expressions of the platform are made use of to calculate the required actuator behaviors for the platform to achieve the desired motion. Then, a cascade control structure is designed for control of both speed and position of the actuators where proportional integral controller is preferred. The designed control strategy is implemented to the Stewart platform where satisfactory performance is observed for it to be used in military vehicles.

ÖZET

STEWART PLATFORMUNUN TASARIMI, GELİŞTİRİLMESİ, MODELLENMESİ VE DENETLENMESİ

Günümüz askeri teknoloji geliştirmelerinde silah kulelerinin, uzaktan komutalı silah sistemlerinin ve görüş sistemlerinin ataletsel stabilizasyon geliştirme testlerinde kullanılmak üzere altı serbestlik derecesine sahip bir test platformu ihtiyacı bulunmaktadır. Tezde bu araştırma konusu ele alınarak bir görüş sisteminin ataletsel stabilizasyon geliştirme testlerinde kullanılmak üzere bir Stewart platformu üretildi ve kullanıldı.

Tüm sistemin net bir şekilde anlaşılabilmesi için Stewart platformunun mekanik tasarım kriterleri, sistem mimarisi, birimler arası haberleşme ve seçilen donanımlar detaylıca sunuldu. Tasarım ve geliştirme safhaları tamamlandıktan sonra Stewart platformunun bilgisayar destekli tasarımı kullanılarak kinematik modeli elde edildi. Platformun ters kinematik ifadeleri platformun istenen hareketleri yapabilmesi için gerekli eyleyici davranışlarını hesaplamada kullanıldı. Eyleyicilerin hız ve pozisyon denetimi için tercih edilen PI denetleyiciler kademeli denetim yapısında kullanıldı. Tasarlanan denetleyici stratejisi Stewart platformuna uygulandı ve askeri sistemlerin testlerinde kullanmak için tatmin edici başarım gözlemlendi.

TABLE OF CONTENTS

LIST OF FIGURES

Figure **Page**

Figure **Page**

LIST OF TABLES

LIST OF SYMBOLS

D ..... Diameter of the wire

V_{drop} ..... Voltage drop across the wire

I ..... Current flowing from the wire

$R(D)$ ..... Resistance of the wire

ℓ ..... Length of the wire

P_{loss} ..... Power loss

m ..... The dimension of the end–effector space

f_i ..... Degrees of freedom of each joint

R ..... Rotation matrix

$\vec{d}$ ..... Displacement vector

l_i ..... Required lengths of each leg

$\vec{p}$ ..... Connection points vector of the top plate

$\vec{b}$ ..... Connection points vector of the base plate

α ..... Angular position around x-axis

β ..... Angular position around y-axis

γ ..... Angular position around z-axis

$R_x(\alpha)$ ..... Rotation matrix around x-axis

$R_y(\beta)$ ..... Rotation matrix around y-axis

$R_z(\gamma)$ ..... Rotation matrix around z-axis

J^{-1} ..... Inverse Jacobian matrix

$\vec{v}$ ..... Variable vector

$\vec{l}$ ..... Leg lengths vector

$u(t)$ ..... Control input

$e(t)$ ..... Error value

K_p ..... Proportional gain

K_i ..... Integral gain

T_i ..... Integral time constant

K_d ..... Derivative gain

T_d ..... Derivative time constant

LIST OF ABBREVIATIONS

APG . . . . . Aberdeen Proving Ground
AWG . . . . . American Wire Gauge
BOM . . . . . Bill of Materials
CAD . . . . . Computer Aided Design
COM . . . . . Communication
CRC . . . . . Cyclic Redundancy Check
DC . . . . . Direct Current
DOF . . . . . Degrees of Freedom
GND . . . . . Ground
IC . . . . . Integrated Circuit
ID . . . . . Identifier
IMU . . . . . Inertial Measurement Unit
MFD . . . . . Multi–Functional Display
P . . . . . Proportional
PC . . . . . Personal Computer
PD . . . . . Proportional Derivative
PI . . . . . Proportional Integral
PID . . . . . Proportional Integral Derivative
PWM . . . . . Pulse–Width Modulation
RC . . . . . Remote Control
RCWS . . . . . Remote Controlled Weapon System
RX . . . . . Receive
SLRT . . . . . Simulink Real–Time
SPI . . . . . Serial Peripheral Interface
SPT . . . . . Stabilized Pan–Tilt
TTL . . . . . Transistor–Transistor Logic
TX . . . . . Transmit
UART . . . . . Universal Asynchronous Receiver/Transmitter
USB . . . . . Universal Serial Bus

CHAPTER 1

INTRODUCTION

Land forces preserve their significant role still in 21st century warfare. Asymmetrical combats constitute most of the wars where land forces have the most important role in these combats. Due to their high mobility and firepower, battle tanks and armored vehicles are used in combats for decades. And to adapt to new warfare conditions like aerial, naval and/or cyber physical, technologies in these vehicles are quickly evolving and to keep up with these evolving technologies, these vehicles and their mission hardware such as turrets or weapon systems and sensor systems must be developed progressively.

Most of the modern weapon systems are remote controlled weapon systems (RCWS) or turrets, and an officer/operator can operate the weapon system either remotely or from inside the vehicle via a multi-functional display (MFD) unit and a joystick/control console. The officer who operates the weapon system is usually referred to as gunner or shooter. MFD unit displays the view of the camera mounted on the weapon or sight system and the joystick console allows the gunner to aim, control and adjust weapon parameters such as ammunition type, weapon type and gyro drift offset. In some weapon systems, mostly to increase the responsiveness of the view of the gunner, cameras and other sensors are placed on independent vision systems (where camera is not placed on the gun barrel thus can move independently) instead of fixing them on the weapon system. Independently placed vision systems are usually utilized in turret systems to avoid significantly larger inertia of the turrets, and camera and other sensors can move with larger accelerations.

Land vehicles are dynamic systems mostly because they are operated usually on harsh terrain which induces disturbances on the turret, RCWS and vision system of the vehicle. Thus, the mission objective is obviously affected by the disturbances especially by the ones acting on the vision system or on the cameras mounted on the RCWS. This problem dates back to World War II where in the African battlefields, operators were forced to stop the tanks to target correctly to minimize the negative effects of the disturbances caused by the rough terrain (MIL-HDBK-799(AR), 1996).

Designers of weapon systems started working on this problem during World War II, and it was concluded that to overcome the disturbances acting on the vision and/or weapon systems caused by the motion of the vehicle on rough terrain, these systems must have inertial stabilization capabilities. Via using inertial measurement sensors and appropriate electro-mechanical hardware, vision system and/or RCWS can stabilize itself in the inertial frame. The weapon systems with inertial stabilization capability greatly increased the gunner's targeting performance (Karayumak, 2011). A comparison of turret motion of a battle tank in elevation and azimuth axis with and without inertial stabilization are demonstrated in Figures 1.1 and 1.2.

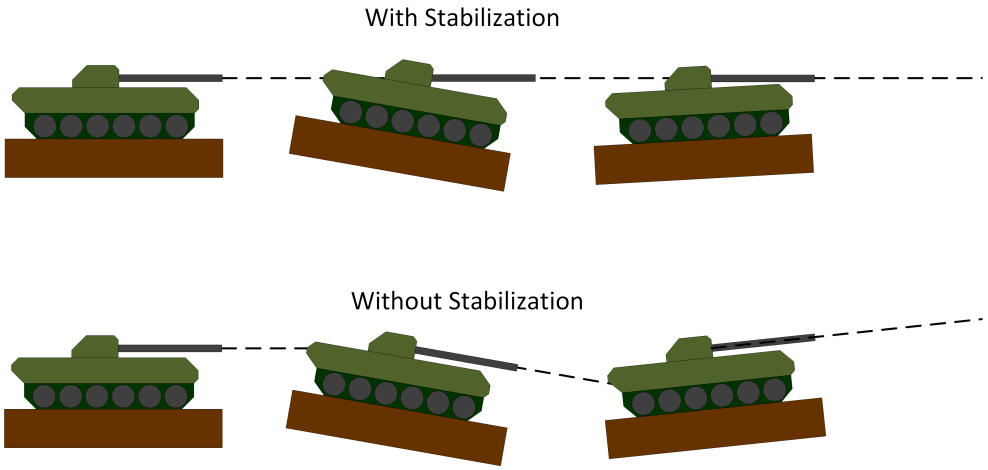

Figure 1.1. Turret motion in elevation axis with and without inertial stabilization

The performance of the inertial stabilization of the systems are commonly tested on a track called Aberdeen Proving Ground (APG) track (Cech et al., 2014). The APG track has different barriers at various distances with varying heights which are all determined according to military standards (01-1-011A, 2012; Gürbüz, 2006). As the vehicle passes through the barriers, the vehicle is disturbed as angular velocities and translational accelerations are directly affected. The translational accelerations are along the three linear/cartesian axes while the three angular velocities are about these linear axes as shown in Figure 1.3. In APG track, the angular velocity and the linear acceleration measured by gyroscopes and accelerometers are collected to be used in performance evaluations. An example of a typical pitch and yaw data obtained with a gyroscope is shown in Figure 1.4. These data are collected from a moving armored ground vehicle (Nurol's Ejder Yal-

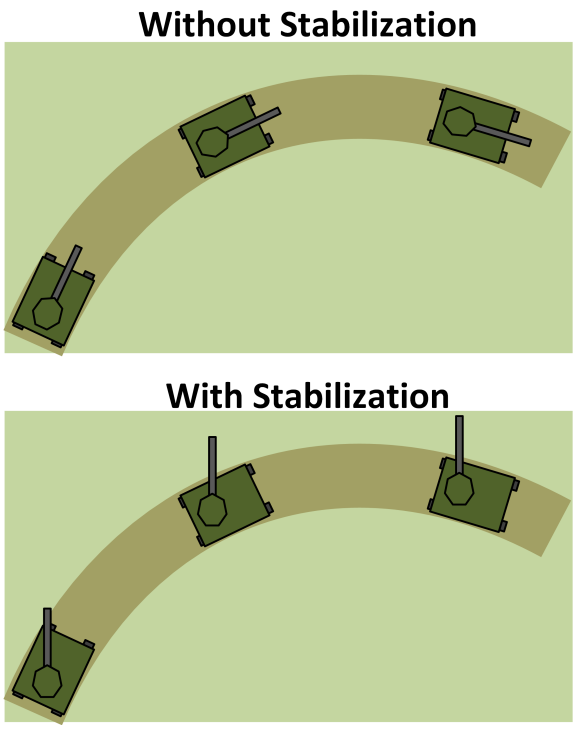

Figure 1.2. Turret motion in azimuth axis with and without inertial stabilization

cin[1] armored vehicle) during a vision system stabilization study in ASELSAN[2] facilities. These data demonstrate the disturbance acting on vision systems.

Additionally, military equipment including weapon systems can be subjected to different testing procedures depending on different needs for varying working scenarios. Specifically, engineering tests which are called development testing are essential during the development phases to investigate whether a proposed solution is feasible or not. Especially, the disturbances acting on the weapon systems are required to be simulated for the development tests in the laboratory. Since the vehicle has to have motion about all 6 axes, the platform used for simulating the disturbances should be capable of generating 6 degrees of freedom motion. After the development phases, operational tests are to be carried out to make a rational decision whether proceeding with a full scale production is feasible or not. In all of these tests, Stewart platforms can be utilized as disturbance simulators due to their several advantageous properties that are discussed in the next section. Motivated by the need of disturbance simulators for almost all weapon systems, in this thesis, design, development, modeling and control of a Stewart platform that is used to carry out the development and operational testing of a stabilized vision system developed

[1]Nurol Makina, Ejder Yalcin: https://www.nurolmakina.com.tr/tr/urunler/ejder-yalcin
[2]https://www.aselsan.com.tr/en/about-us/who-we-are

by Teknodef Defence Systems[3] is investigated.

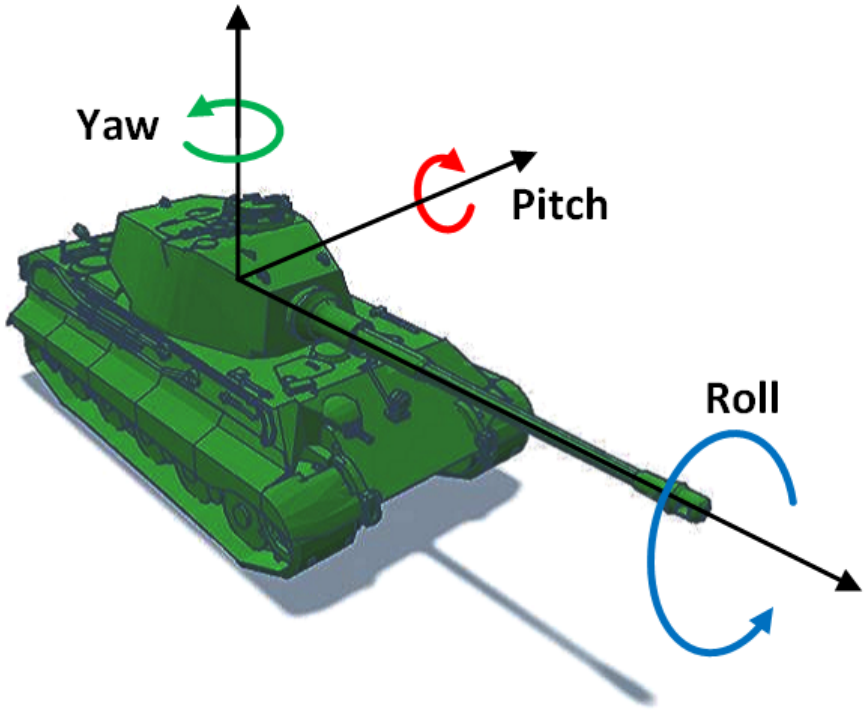

Figure 1.3. Angle axis representation of tank motion

1.1. Stewart Platform

Stewart platform was originally designed in 1954 by Eric Gough and publicized by D. Stewart in 1965 (Stewart, 1965). The original design of the Stewart platform by Eric Gough was slightly different from the Stewart platform used nowadays. Due to its motion and its number of legs, the Stewart platform is alternatively called hexapod, 6 degrees of freedom platform or 6-axis platform (Cirillo et al., 2017; Li et al., 2019; Thoendel, 2011).

Stewart platform is a parallel mechanism consisting of 2 plates and has 6 degrees of freedom. One of the plates, called the top plate, is connected to the other plate, commonly named as base or fixed plate, via 12 universal joints (6 on each plate) and 6 independently moving extensible legs. The 3D computer–aided design (CAD) model of the Stewart platform designed at Teknodef Defence Systems is presented in Figure 1.5.

Stewart platforms or parallel mechanisms in general have several advantageous properties that allows them to be preferred for several applications. One of these advan-

[3] www.teknodef.com

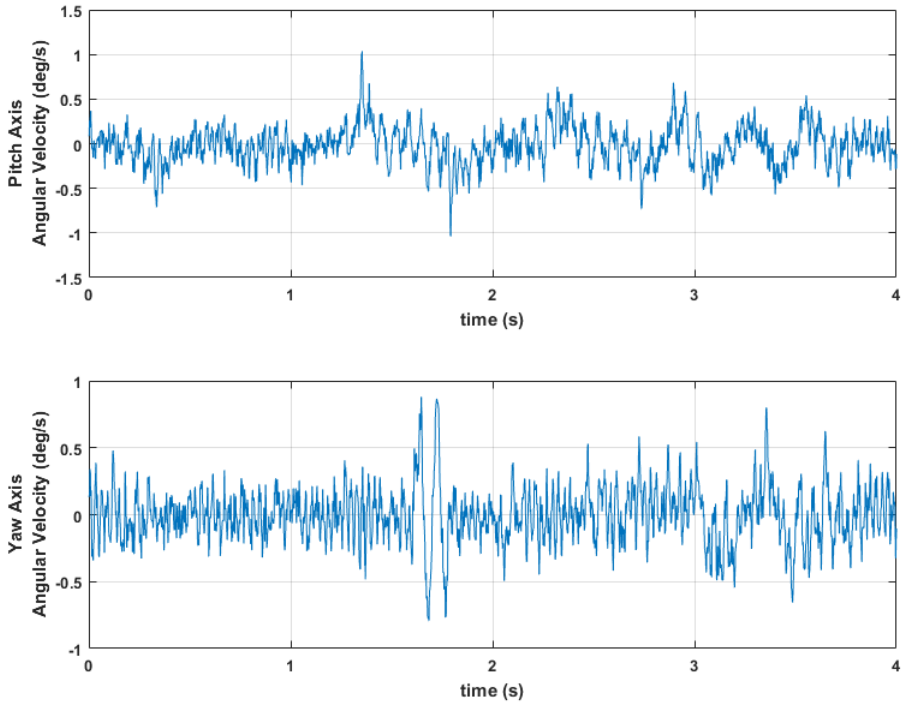

Figure 1.4. Example pitch (top) and yaw (bottom) axis gyroscope data collected

tages of the Stewart platform is that it has high load carrying capacity due to its parallel mechanism. Stewart platforms and parallel mechanisms are capable of positioning their end effectors more precisely. High rigidity of the Stewart platform is another desired property of it due to the increased controllability of the mechanism. Despite their advantages, disadvantages of them are not discardable. Mechanical design of Stewart platforms are complex and the kinematic equations are hard to derive when compared to the serial mechanisms. They have relatively small workspace and control algorithm design process is usually harder. When their advantages and disadvantages are compared, Stewart platforms are usually preferred for applications that require precise positioning of the manipulator in 6 degrees of freedom especially when there is need for higher load carrying capacity.

Stewart platforms are preferred for several applications in military research, robotics, aerospace, medicine, underwater research, flight simulators, entertainment technology and crane technology. In Figure 1.6, a Stewart platform is used for positioning of a space observation system where a large mass is required to be clear-cut positioned (Ho et al., 2009). Because of their high number of degrees of freedom, Stewart platforms are used

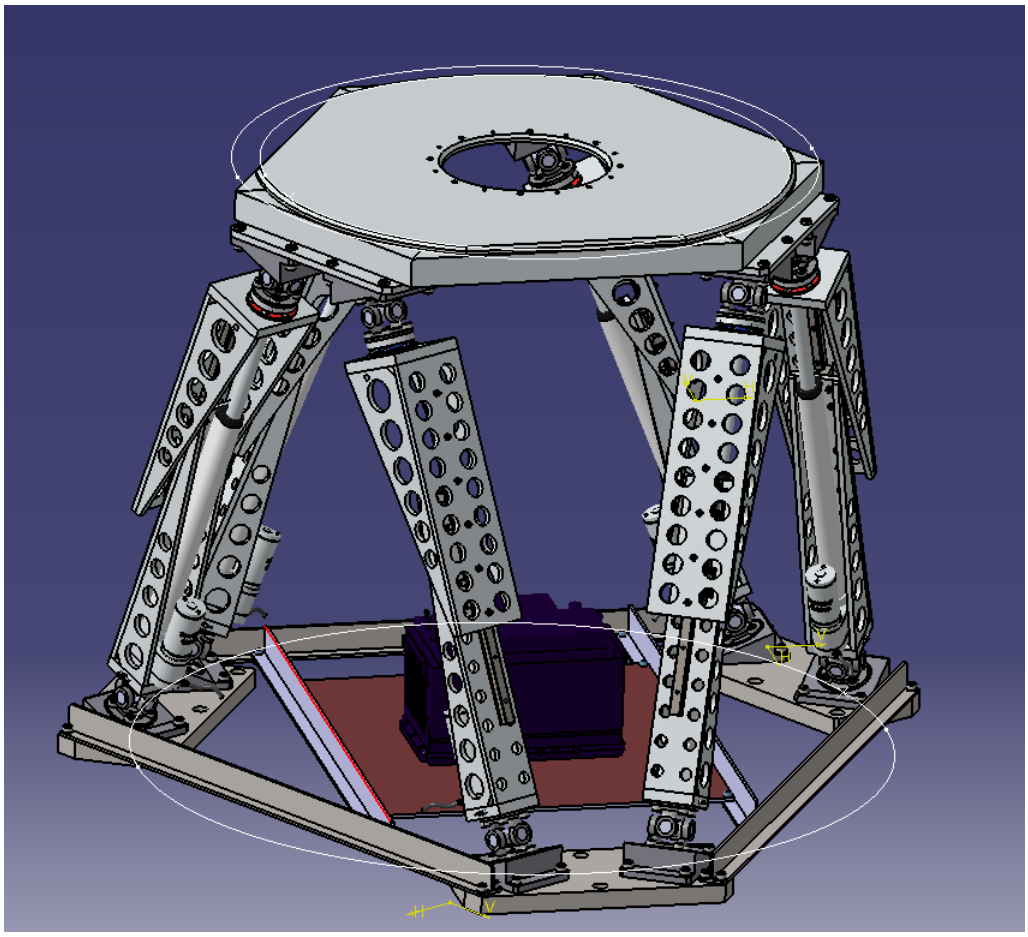

Figure 1.5. CAD model of the Stewart platform developed by Teknodef Defence

as an endoscopic tool manipulator in surgeries as demonstrated in Wendlandt and Sastry (1994). Stewart platforms are often used as flight simulators which are extremely important for training, research and development phases. In the training phase, it decreases the cost and increase the safety, while in research and development, it is used for evaluating the controllability of the aircraft (Dongsu and Hongbin, 2007). As an interesting fact, the first publication on Stewart platform proposed it to be used as a flight simulator (Stewart, 1965).

1.2. Organization of the Rest of This Thesis

This thesis is focused on utilizing the Stewart platform for military research where design, development, modeling, and control phases of a Stewart platform are presented.

Figure 1.6. Stewart platform used on the AMiBA-7 Observator

The rest of this thesis is organized as follows. In Chapter 2, mechanical design principles and electrical system architecture are described. The software environment is introduced along with the mechanical and electrical components and their configurations. In Chapter 3, kinematic model of the Stewart platform is formed, and degree of freedom calculations of the mechanism and inverse kinematic solutions are verified. In Chapter 4, control of the Stewart platform is discussed. Finally, concluding remarks and possible future research problems are presented in Chapter 5.

CHAPTER 2

STEWART PLATFORM

In this chapter, design and development of mechanical and electrical systems along with the software components are presented in detail.

2.1. Mechanical System Design

The Stewart platform mechanism has two plates, base plate and top plate. Base plate and top plate are connected with 6 legs via 12 universal joints with 6 on each plate. These 6 legs are the actuators of the mechanism and can extend or shrink. While there are some applications that utilize the base plate as the end–effector, in most of the applications including this study, end–effector is the top plate. The position and orientation of the end–effector are changed by adjusting the lengths of the legs via shrinking or extending them.

There are several key parameters in the design of mechanical system of the Stewart platform. These parameters play important roles in the behavior and performance of the Stewart platform. Three of the important properties of the Stewart platform are load carrying capacity, the workspace and the rigidity of the system. By varying the parameters of the mechanical design, we can change the load carrying capacity, workspace and rigidity, thus controllability of the Stewart platform.

Load carrying capacity is decided by the weight of the payload that is an extremely important property for this study. The payloads that will be used with this Stewart platform will be a commander and shooter vision system which weights approximately about 35 kg.

Workspace of the platform should contain the predetermined motion that simulates the armored vehicle moving on the terrain. The needed workspace of the end–effector will affect the required strokes of the linear actuators.

Rigidity of the mechanism directly affects the controllability of the system. By increasing the rigidity, we can increase the frequency range that the output of the system can follow the input in terms of magnitude and phase. To increase the rigidity of the

system, support structures like carriages and/or rails may be considered in the design of the platform.

2.1.1. Design Criteria

Load carrying capacity of Stewart platforms are adjusted by the ratio of the radius of the top plate to the radius of the base plate as can be seen in Figure 2.1 (Rastegarpanah et al., 2013). If an angle between a force vector and an arbitrary axis increase, the force acting on that axis will decrease as demonstrated in Figure 2.2. Therefore, the highest load carrying capacity is obtained when the radii of base and top plates are equal (the left design in Figure 2.2), and as the ratio of the radius of the top plate to the radius of the base plate decreases the load carrying capacity decreases where the load carrying capacity of the middle design is less than the left one while the right design has the minimum load carrying capacity.

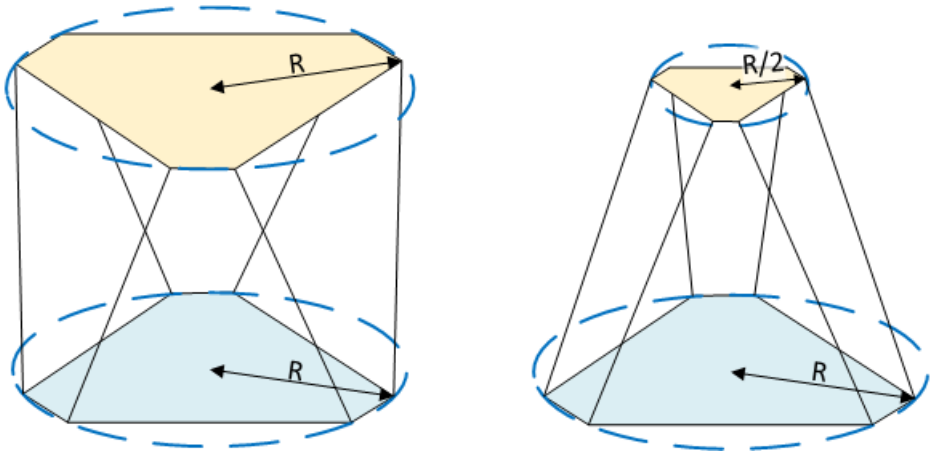

Figure 2.1. Two platform with different top plate to base plate diameter ratios

On the other hand, there is a trade off of using equal radii for base and top plates because the actuators' ability to work on the horizontal axes will be greatly reduced which can also be seen from Figure 2.2. This will yield a decreased workspace for the Stewart platform. The effect of the actuator lengths on the workspace can be seen in Figure 2.3.

The above mentioned design parameters are adjusted to yield different properties specific to this study. And to reach an optimal performance, there will be sacrifices from

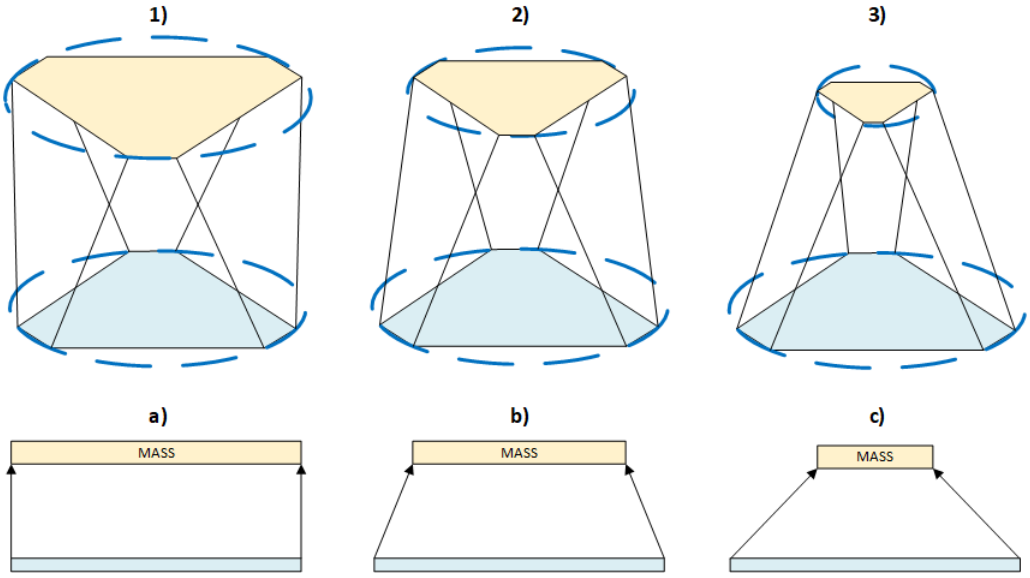

Figure 2.2. Load carrying capacities of platforms with different top plate diameters

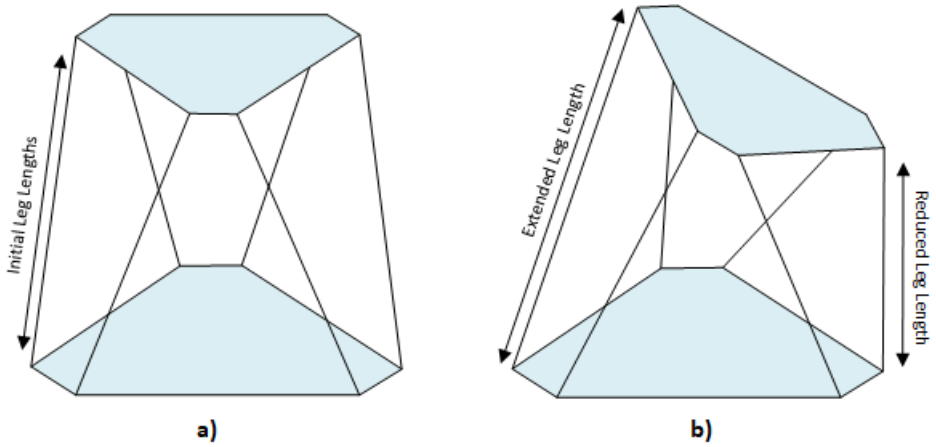

Figure 2.3. a) Top plate's initial orientation b) Top plate's arbitrary orientation

either load carrying capacity or workspace.

The Stewart platform in this thesis should be able to emulate the motion of an armored vehicle moving on a terrain which is characterized by the exemplary data shown in Figure 1.4. So the workspace of the Stewart platform should be able to perform these motions, thus the strokes of the linear actuators should be enough for the motion, in a kinematic manner.

After deciding the critical mechanical design parameters, solid design process is completed on Catia V5 software[1]. Mechanical design parameters which will also be re-

[1]https://www.3ds.com/products-services/catia/

quired for inverse kinematic calculations of the Stewart platform are shown in Figures 2.4, 2.5, 2.6 and Tables 2.1 and 2.2.

Table 2.1. Mechanical design parameters of the plates

Design parameter	Value
Radius of the base plate	505 mm
Thickness of the base plate	25 mm
Radius of the top plate	353 mm
Thickness of the top plate	50 mm

Table 2.2. Mechanical design parameters of the legs

Design parameter	Value
Length of upper leg	50 mm
Length of lower leg	463 mm
Leg radius	40 mm
Collar thickness	10 mm
Flange length	30 mm
Flange width	22.5 mm
Flange thickness	8 mm

2.2. System Architecture

In this subsection, the system architecture is presented.

2.2.1. Host Computer, Target Computer and Simulink Real–Time

As demonstrated in the simplified system architecture in Figure 2.7, there are 2 computers in the system. One of them, called the host computer, is used for receiving user input and performing kinematic calculations, while the other one, named as the target computer, is for real time execution and for communication with the motor drivers that control and drive the linear DC motors.

Host computer is typically a desktop PC or a laptop computer which runs Matlab, Simulink or Stateflow. In this study, a laptop computer is used as the host computer. There are two major requirements for the host computer. First requirement is to communicate

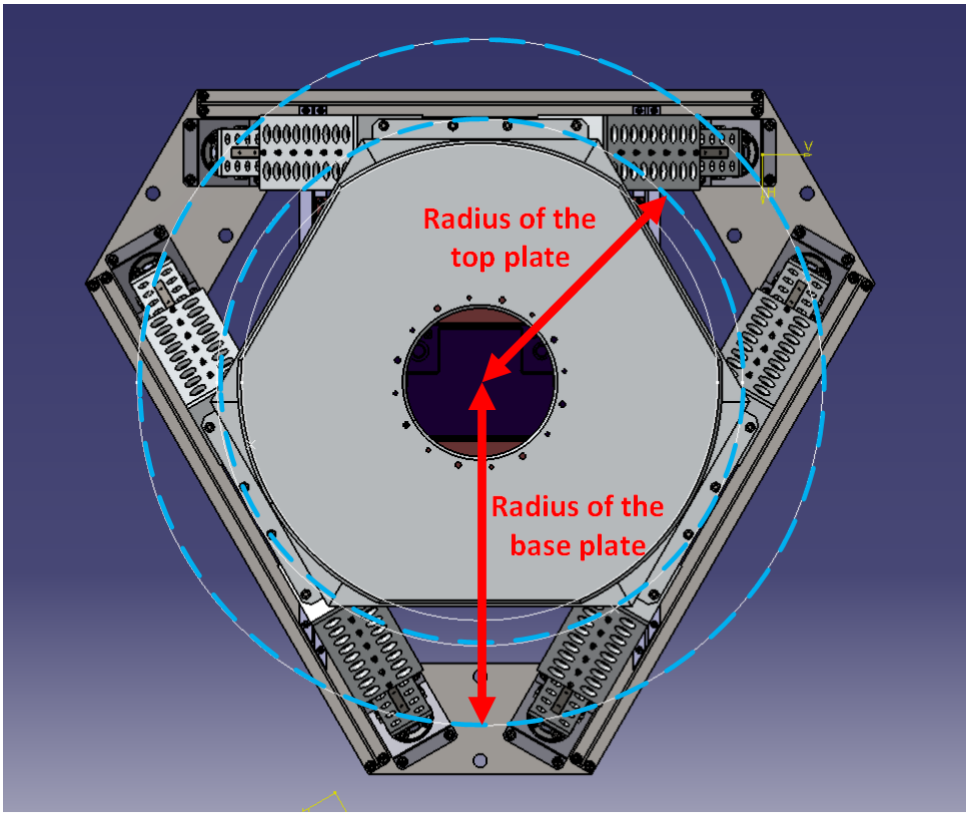

Figure 2.4. Top plate and base plate radii of the Stewart platform

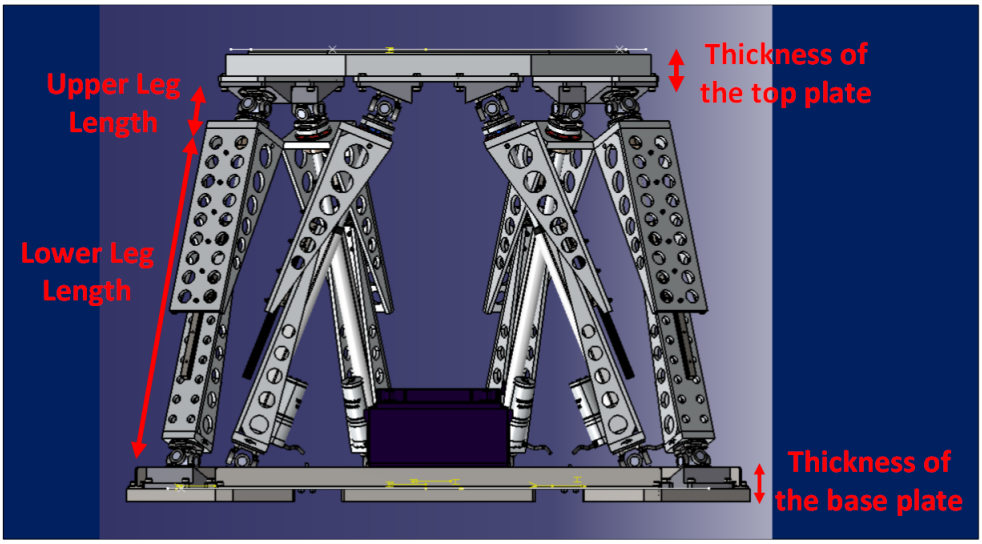

Figure 2.5. Mechanical properties of the Stewart platform's legs

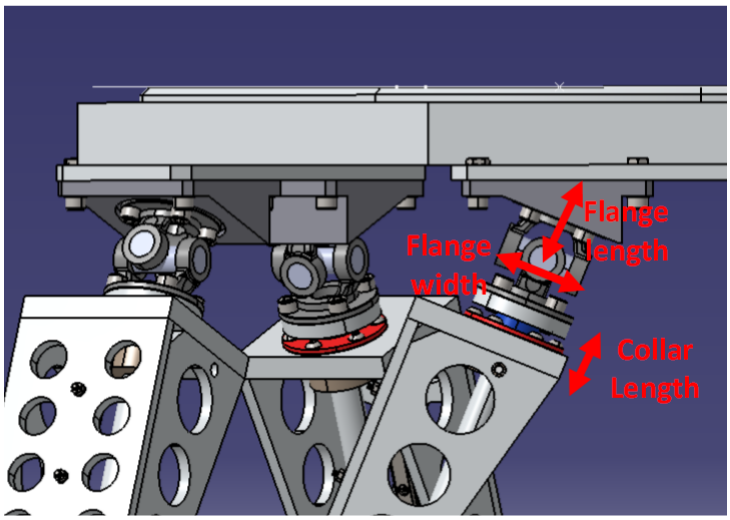

Figure 2.6. Mechanical properties of the Stewart platform's flanges

with the target computer and to transfer data to it. For this requirement to be realized, the host computer should have an ethernet or RS232 port. Second requirement is for running Matlab and Simulink Real–Time (SLRT). Mathematical models are created on the host computer. Specifically, desired motion of the end–effector of the Stewart platform is entered to the host computer as the user input and the host computer evaluates the rotation matrix and displacement vector, solves the inverse kinematic calculations, obtains the required leg lengths for the desired motion, and sends the required leg lengths to the target computer via gigabit ethernet. An executable code is created from the model with a C compiler. This executable code is transferred to the target computer that runs SLRT kernel. Thus, the host computer is used to communicate with the target computer, and this communication involves only sending the desired leg lengths and leg velocities.

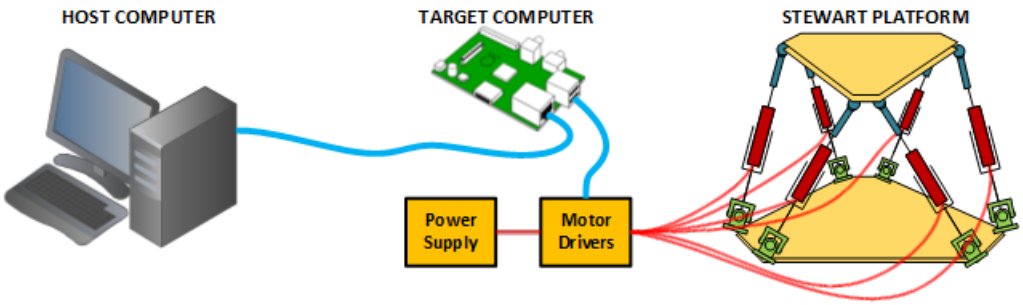

Figure 2.7. Simplified System Architecture

[2]PFM-CVS from AAEON: https://www.aaeon.com/en/c/pc-104-modules

Target computer[2], shown in Figure 2.8, is a PC/104+ with an onboard Intel Atom processor N2600. This embedded single board computer has three RS-232 ports, one RS-232/422/485 port, four USB2.0 ports, and eight bit digital I/O. It supports 2GB of DDR3 memory and has msata connectors for SSD storage. Network connectivity is achieved by the onboard 10/100/1000 Base-Tx Ethernet. Using the PC104 bus or the Mini PCI Express expansions, functionality of the board can be expanded. Operating temperature range of the computer is between -40°C and +80°C.

After receiving the required leg lengths, target computer uses this information to send command messages to the motor drivers over the RS–232 serial bus. After downloading the executable code, application can be run and tested on the target computer in real–time. So, the real–time applications running on the target computer and the host computer are separate from each other.

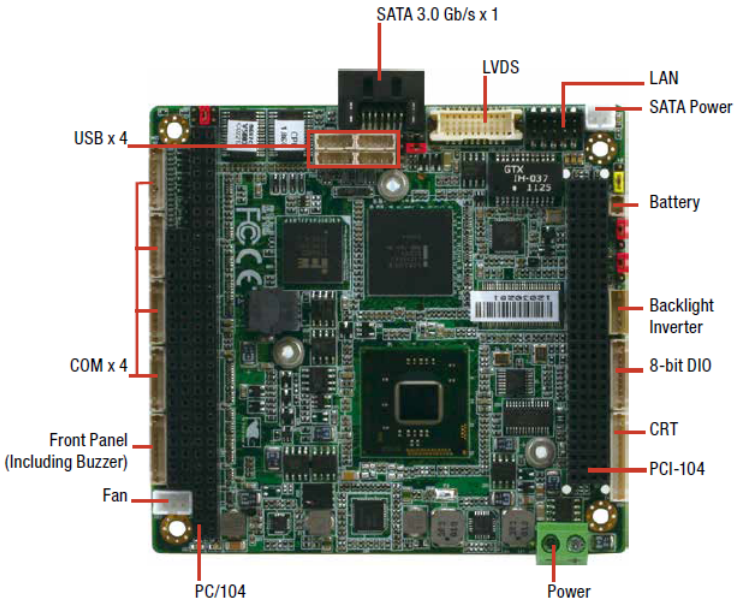

Figure 2.8. Target computer: PFM-CVS from AAEON

SLRT is a MATLAB product for prototyping, testing and deploying real–time systems using hardware. Configuration of the SLRT environment can be found in Matlab Documentations[3].

Motor drivers have built-in cascaded control loop including both position and speed controllers which will be detailed in Section 4.2. The required leg lengths are taken from the target computer over RS-232 and required leg velocities are calculated by

[3]Simulink Real-Time Configuration: https://www.mathworks.com/help/xpc/gs/set-up-and-configure-xpc-target.html

the motor drivers. Output of the position control loop is the demand for the speed control loop, and the output of the speed control loop is the PWM duty cycle of the actuator driver voltage.

2.2.1.1. Power Card

Instead of directly supplying power to the target computer, a power card[4], shown in Figure 2.9, that is connected through the PC104 bus is used specifically to prevent possible damages to the target computer that may be caused as a result of spikes and surges on the power line. Operating temperature of the power card is between -40°C and +80°C while its power output capability is 50W.

Figure 2.9. Power Card: PFM-P13DW2 from AAEON

2.2.2. Motor Drivers

Legs of the Stewart platform are actuated with linear DC motors. To drive and control these linear DC motors, Roboclaw 2×15A motor drivers[5] shown in Figure 2.10 are used. Each motor driver has the capability of driving two motors, so for six legs, three motor drivers are used as shown in the flow diagram in Figure 2.11.

[4]PFM-P13DW2: https://www.aaeon.com/en/p/pc104-modules-pfm-p13dw2
[5]Roboclaw 2×15A: https://www.pololu.com/product/3285

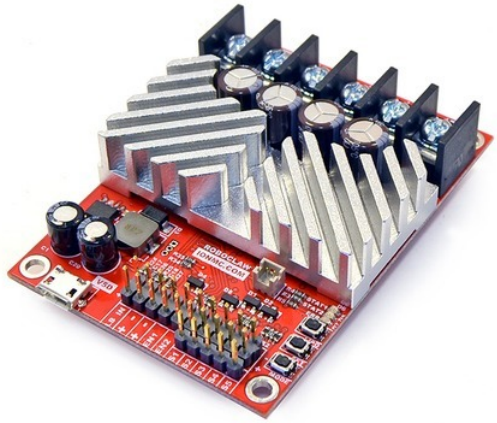

Figure 2.10. Roboclaw 2×15A, 34VDC Dual Channel Brushed DC Motor Driver

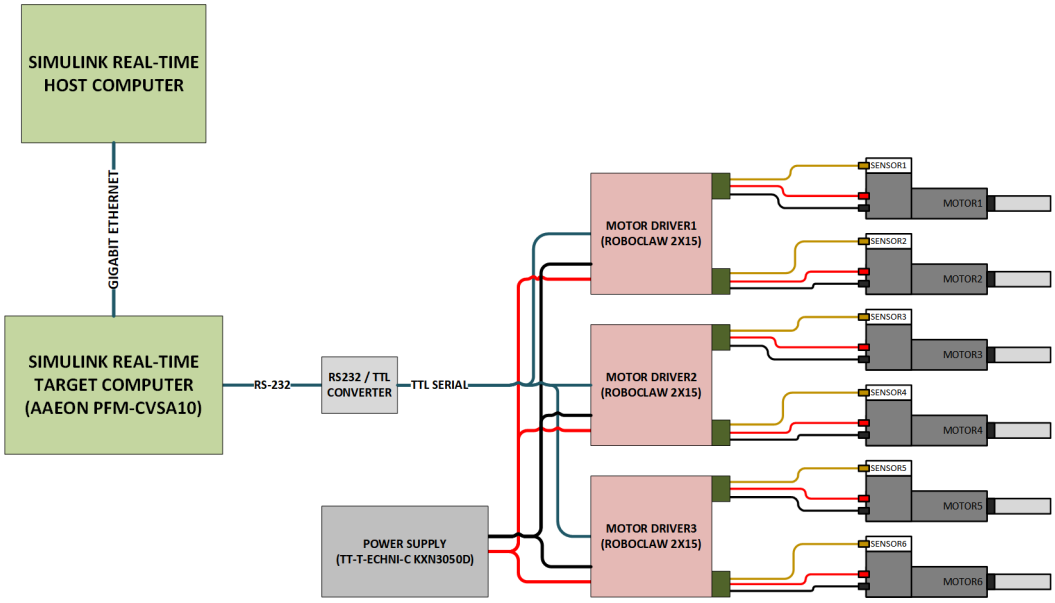

Figure 2.11. System Architecture

Roboclaw 2×15A motor drivers are high performance motor controllers that can supply 30A peak per channel. They can be controlled from USB, RC radio, PWM, TTL serial or analog signal. In this study, TTL serial will be used for controlling motor drivers. These motor drivers support a wide variety of feedback sensors ranging from quadrature encoders, absolute encoders to potentiometers which can be easily configured via the driver configuration software ION Studio[6]. Roboclaw 2×15A motor drivers also has several protection features like temperature, current, over voltage and under voltage limitations.

2.2.2.1. Driver Configuration Software: ION Studio

Roboclaw motor drivers come with a software called ION Studio which allows users to configure, monitor and maintain them. Control parameter/gain tuning and monitoring can be accomplished using this software as can be seen from the screenshot of the software in Figure 2.12 and feedback type, maximum allowed current for motors, supply voltage, baudrate and control mode can be set as shown in the screenshot in Figure 2.13.

2.2.2.2. Baudrate Calculations for Drivers

TTL serial is used to send commands to the motor drivers. The command structures consist of an address byte which is the ID of a driver, a command byte that describes the type of the command, data bytes and a cyclic-redundancy-check byte. The three drivers have different addresses or IDs which are presented in Table 2.3. Roboclaw drivers use a CRC-16 cyclic-redundancy check calculation to validate each command it receives. This reduces probability of sending unintended commands to the driver. Structure of the command message can be seen in Table 2.4.

There are different command types sent to the driver which are reading the PID gains, setting the PID gains, driving 2 channels simultaneously or driving 2 channels independently. In this study, “Drive Motor 1 with Position Command” and “Drive Motor 2 with Position Command” command types are used to send desired position values of

[6]BasicMicro Motion Studio (formerly known as ION Studio), https://www.basicmicro.com/downloads

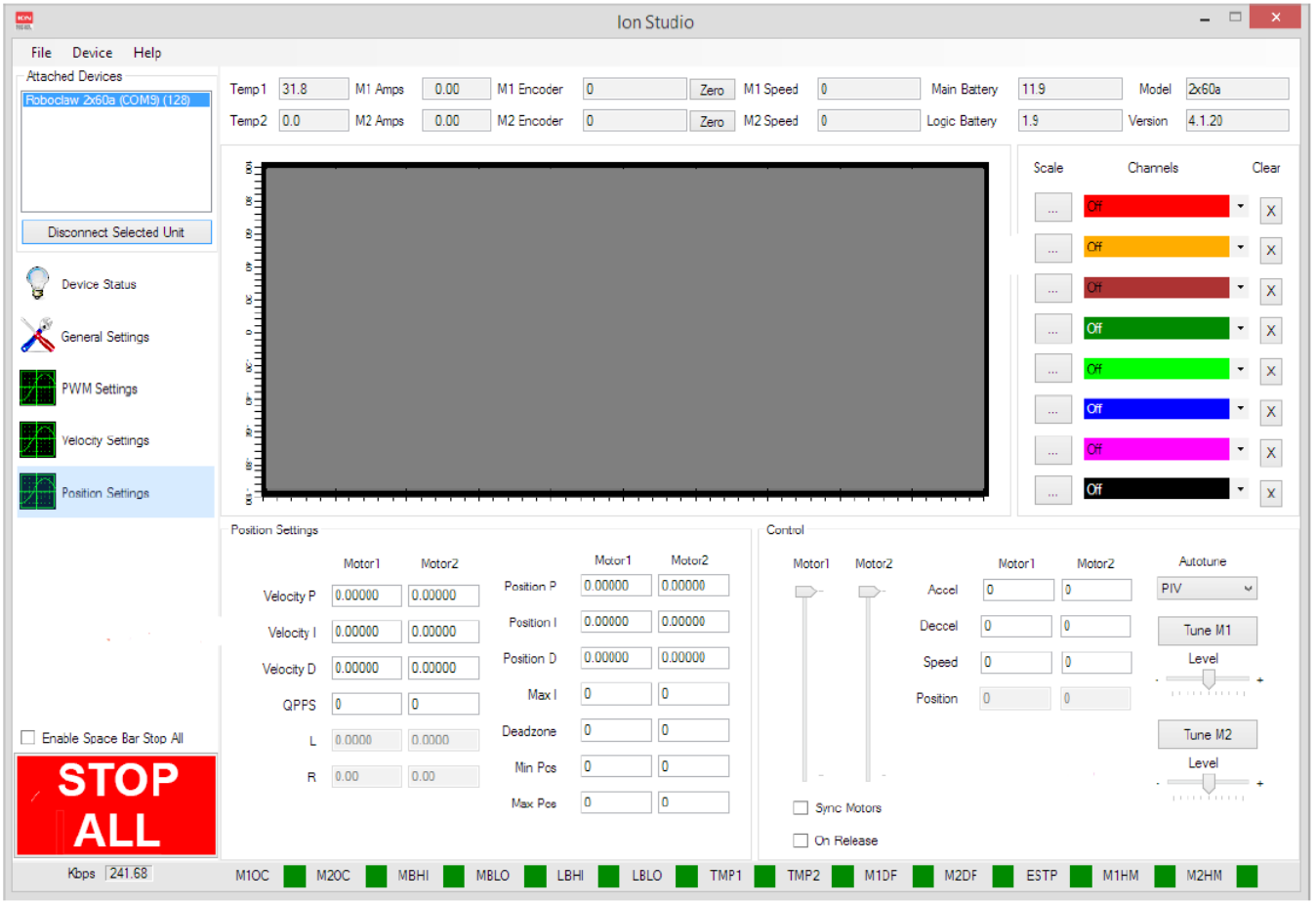

Figure 2.12. ION Studio control loop parameters

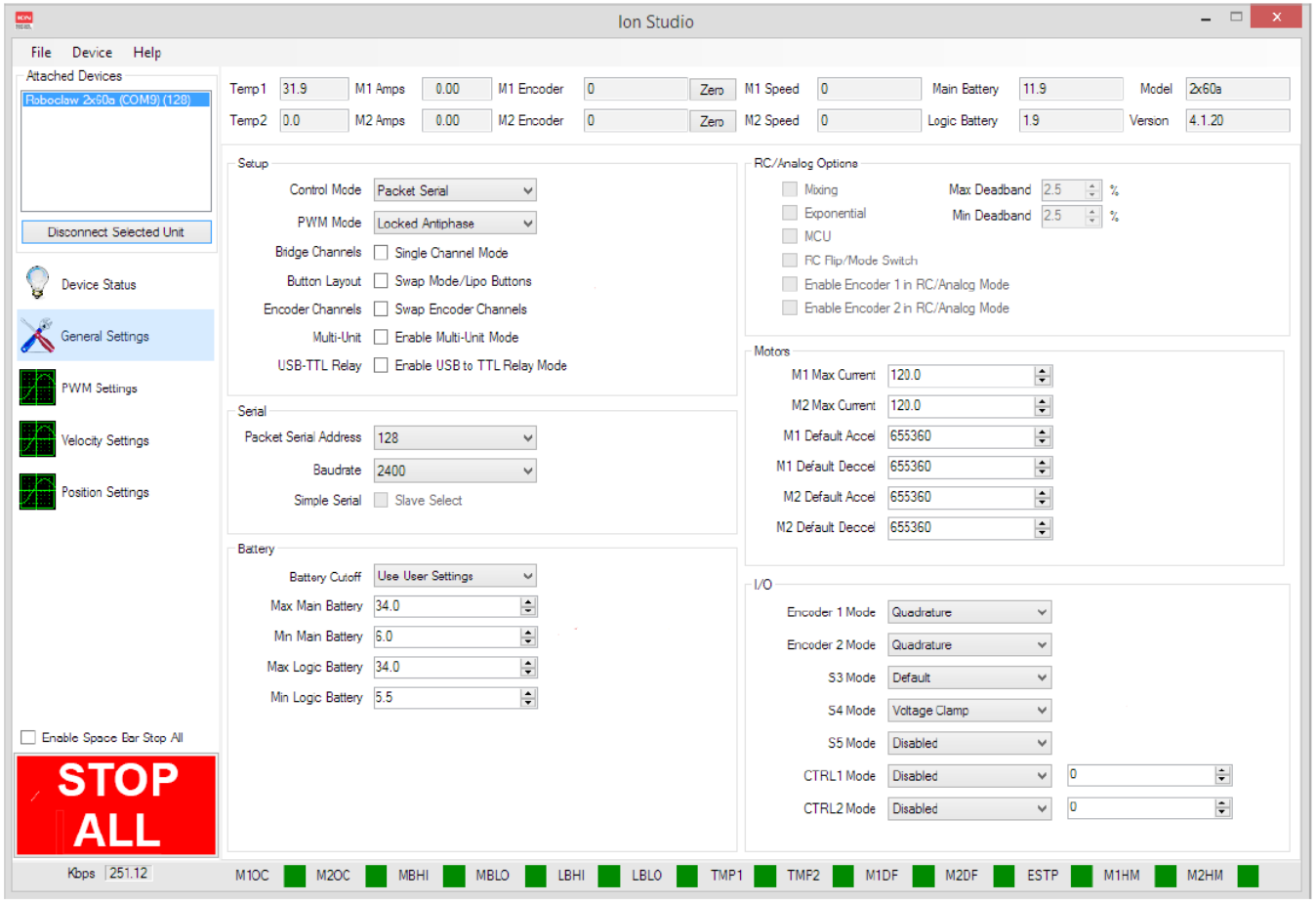

Figure 2.13. ION Studio driver configuration

Table 2.3. Driver IDs

Driver number	ID
Driver 1	128
Driver 2	130
Driver 3	132

the linear actuators to the drivers. Command bytes are 65 for motor 1 and 66 for motor 2 for each driver.

Data bytes in this command type are 4 byte signed integer acceleration limit values, 4 byte signed integer deceleration limit values, 4 byte signed integer speed limit values, 4 byte signed integer position value and the buffer size. Buffer parameter can be 1 or 0. If buffer is set to 0, new commands arriving to the driver should wait until the current command is executed. If buffer is set to 1, new commands arriving to the driver is executed immediately without waiting for the execution of the previous command. Setting acceleration, deceleration and speed limits to 0 means there is no limit for those parameters. The only variable in these data bytes are the 4 bytes of 32 bit signed integer position demands. Drivers send an acknowledgement message back to the computer upon receiving a validated command message. The acknowledgement message is a 1 byte 0xFF value.

Total number of bytes sent to the drivers are 20, and each byte is framed with a 1 start bit and 1 stop bit. In other words, each byte sent is actually 10 bit long in hardware level. Therefore, to send a single command message to drive a single channel, 200 bit long messages should be sent. Since the Roboclaw motor driversâ maximum allowed communication rate is 115200 Baudrate, 115200 bit per second / 200 bit = 576 command messages can be sent per second. Since there are 3 drivers, and each driver has 2 channels, 576 messages per second are distributed to 6 channels thus for each channel, 96 command messages can be sent in one second.

2.2.3. Network Topology

The process of sending data sequentially which is bit by bit data transmission over a computer bus is called serial communication. Parallel communication on the other

Table 2.4. Command Message Structure

Name	Address	Command Type	Acceleration Limit	Speed Limit	Deceleration Limit	Position Demand	Buffer	CRC16
Type	uint8	uint8	int32	int32	int32	int32	uint8	uint8
Decimal Value	Variable	65 or 66	0 (No Limit)	0 (No Limit)	0 (No Limit)	Variable	1	Variable
Length	1 Byte	1 Byte	4 Bytes	4 Bytes	4 Bytes	4 Bytes	1 Byte	1 Byte

hand is transmitting bits on several data lines simultaneously. RS232 standard utilizes the asynchronous serial communication. In asynchronous data transfer, bits of data are not synchronized by a clock pulse, while in synchronous data transfer, bits are synchronized with a clock pulse. Examples of synchronous data transfer are SPI or I^2C protocols.

There are similar standards to RS232 such as RS422 and RS485. The RS422 standard uses differential transmission and differential receive lines to increase noise immunity. The RS485 standard is a slightly evolved version of RS422 standard. Both RS232 and RS422 standards have the ability to communicate peer to peer, on the other hand the RS485 standard utilizes bus communication. Flow diagrams of peer–to–peer communication and bus communication can be seen in Figure 2.14.

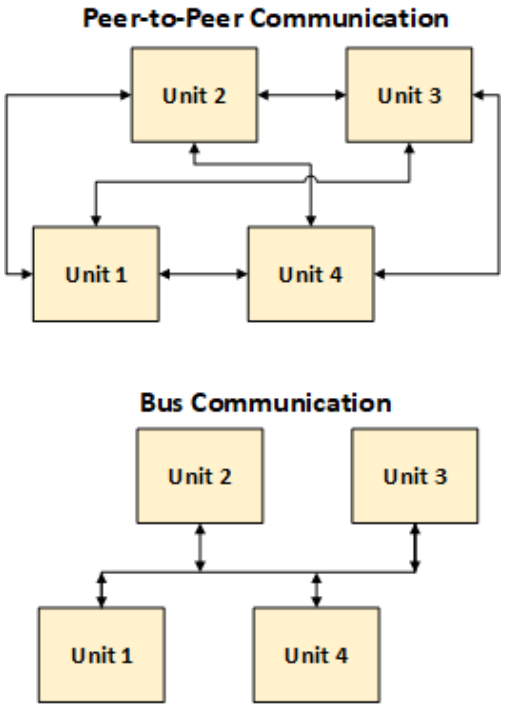

Figure 2.14. Flow diagrams of peer–to–peer and bus communications

Instead of utilizing 3 COM ports on the target computer for connecting to 3 drivers, an alternative approach is used in this thesis. The motor driver supports multi unit packet serial wiring where an example wiring can be seen in Figure 2.15. There are 3 cables to connect 2 devices in RS232 which are Transmit (TX), Receive (RX) and Ground (GND). The TX pin of the target computer is connected to RX pins of all three drivers while the RX pin of the target computer is connected to TX pins of all three drivers, and all GND pins are grounded to a common ground. With this method, a message sent from

the target computer will be received by all drivers. Via using a software filter acting on the received messages, a driver will be able to understand if a message was directed to itself or other drivers. While this method greatly reduces the RS232 communication length, since the target computer and motor drivers are closely placed inside our mechanism, communication in this study will not be affected by length limitations.

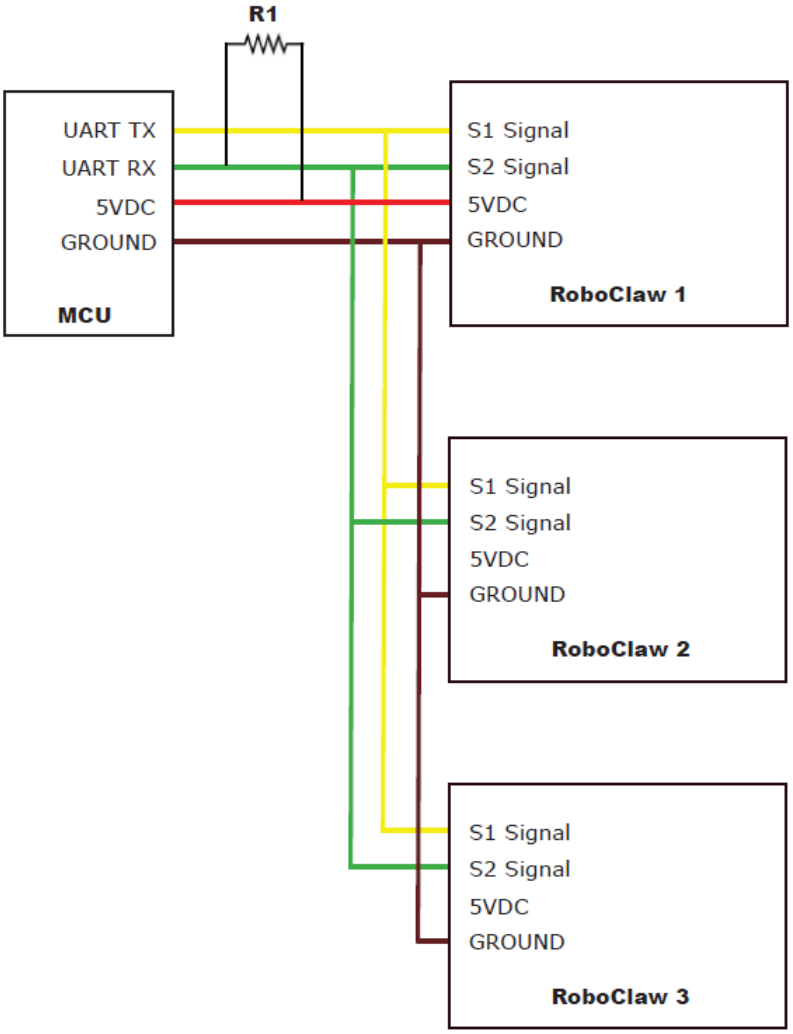

Figure 2.15. Multi unit packet serial wiring

2.2.3.1. RS232–TTL Converter

The Roboclaw motor drivers have TTL serial communication port. Serial communication at TTL level remains between 0V and 5V (or Vcc) where logic high is represented by 5V, and logic low is represented by 0V. On the other hand, target computer has RS232 ports. RS232 signals are similar to the TTL serial signals. They both transmit

one bit at a time, at a specific baud rate with or without parity and/or stop bits. Only difference is in the hardware level. In RS232, logic high is represented by a negative voltage between -25V and -3V, and the logic low is represented between 3V and 25V. On the target computer logic low is 13V, and logic high is -13V. Increasing the logic low voltage level and decreasing the logic high voltage level makes RS232 less susceptible to noise and interference. However differences in voltage levels between RS232 and TTL serial may harm the motor drivers. To interface these two signals, a converter as shown in Figure 2.16 is used. This converter includes a MAX–232 and a few passive components around it. MAX-232 is an integrated circuit (IC) widely used in applications that require RS232 communication. It inverts the signal and regulates the voltage levels. Almost every RS232–TTL Converter in the market uses MAX–232 IC.

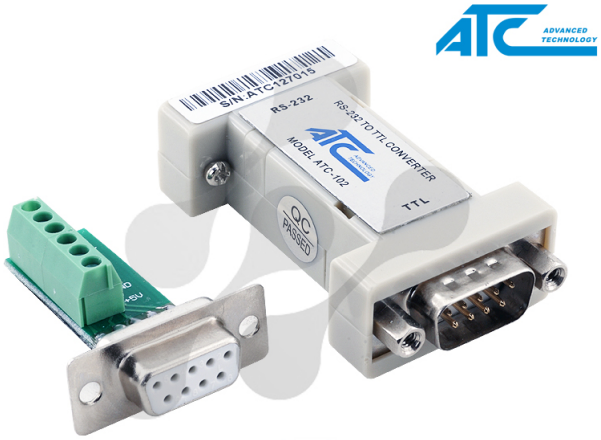

Figure 2.16. RS232–TTL Converter

2.2.4. Linear DC Motors

All six legs of the Stewart platform are actuated by linear DC motors. Linear DC motors convert rotational motion obtained from electric motors into linear motion. When compared to hydraulic actuators, linear DC motors are less expensive and smaller. Despite being capable of applying larger forces, hydraulic actuators require additional components such as pressure valves, pressure pumps, pipes and liquid tanks. Linear actuator by Pololu[7] having 1:20 gear ratio is considered to be sufficient for the Stewart platform

[7]https://www.pololu.com/product/2313

in this study. It has 200 mm stroke and a built-in potentiometer that is used for position feedback. Current–load graph of the motor can be seen in Figure 2.17.

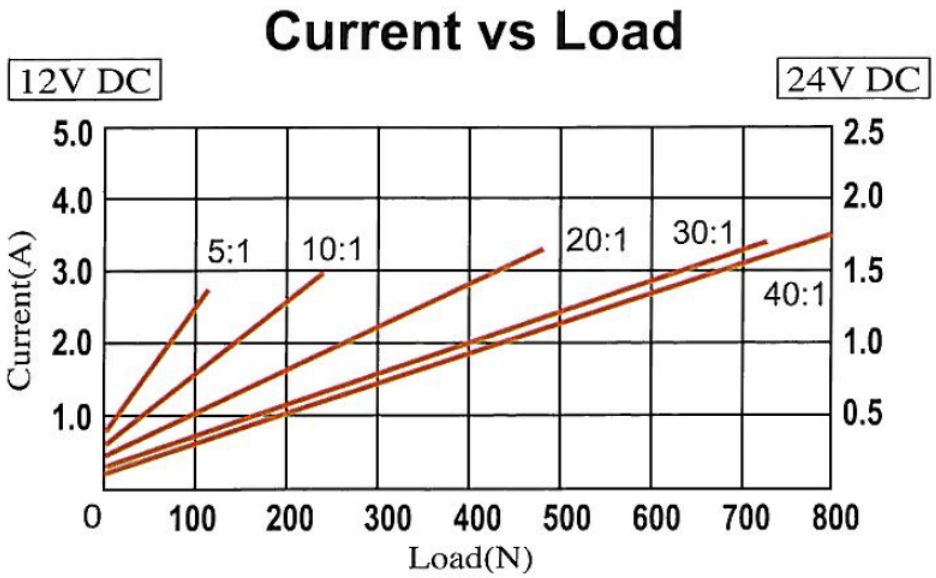

Figure 2.17. Current–load graph of the linear DC motors for different gear ratios

2.2.5. Wiring and Power Loss Calculations

cables on the system are chosen from the American Wire Gauge (AWG) system, which defines the cable diameter sizes. The AWG formula (ASTM International, 2018) has the following form

$$D(AWG) = 0.005 \times 92^{((36-AWG)/39)} \tag{2.1}$$

which gives the diameter of the cable in inches. For example, a AWG-20 cable's diameter is obtained as

$$D(20) = 0.005 \times 92^{((36-20)/39)} = 0.031 \text{ inches.} \tag{2.2}$$

As a result of this formula, as the AWG number increase, diameter of the cable decrease. For another example, a AWG-4 cable's diameter is 0.204 inches. AWG-4 cable is almost 6 times larger than the AWG-20 cable. Most common AWG values and their resistances are shown in Table 2.5.

Conductance or cable resistance changes with the diameter of the cable. As the cable diameter increases, resistance of the cable decreases. Increasing the cable length

Table 2.5. Resistance values for different AWG values

AWG	Ohms per kilometer
4	0.81
10	3.27
12	5.2
16	13
20	33
24	84
26	133
28	212

increases the cable resistance. Resistance of the cables act just like a resistor placed to the pins of devices, harness or wiring design should be done carefully.

If a cable has length ℓ and diameter D, and the current flowing from the cable is I, voltage drop across the cable can be calculated by

$$V_{drop} = I \times R(D) \times \ell \tag{2.3}$$

where $R(D)$ is the resistance of the cable with diameter D per unit length.

Acceptable voltage drop levels are usually less than 3% of the supplied voltage of the cable. As the voltage drop across the cable increases, power loss increases. The power loss on the cable mostly results from the heat generated on the cable resistance and is proportional to square of the current passing through the cable. Using the proper cable insulation material, heat emissions can be tolerated. Power loss can be calculated from

$$P_{loss} = V_{drop} \times I. \tag{2.4}$$

The largest current flow in the Stewart platform is on the cables between the motor drivers and power supply. As seen in Figure 2.17, each linear DC motor can drain approximately 3A of current at maximum load so six of them will drain approximately 18A of current at most. Power supply cables of the drivers are chosen to be 5 meters long which are supplied with 12V. Voltage drop should be less than 3% of 12V. Therefore 18A of current flowing through the resistance of 5 meters cable may be 0.36V at most. Calculations of the formula in (2.3) yields 4Ω/Km thus from Table 2.5, using AWG-10 cables will be enough for the power cables of the drivers. Another method for decreasing voltage drop and power loss is using multiple cables instead of a single cable. This way,

cables will act like parallel resistors and current will be distributed to the cables, resulting in less power loss. Similar approach is used in cable size selection of the cables between motors and drivers, and AWG-16 cables are used. Other cables of the system such as the ones connecting RS232 or Ethernet are carrying low currents so they are chosen to be AWG-24.

2.3. Production and Manufacturing

Bill of materials (BOM) is created and the materials that will be manufactured and the materials that will be purchased are determined. Electrical parts of the system, described in the previous sections, are bought from suppliers. Most of the mechanical parts of the design are manufactured by subcontractors. However, some mechanical parts like rails and joints are widely available in the market, and they are bought instead of manufacturing. Completed Stewart platform can be seen in Figure 2.18.

2.4. Payload

The payload of the Stewart platform is a stabilized pan-tilt (SPT) system as shown in Figure 2.19. SPT system is designed and developed for optoelectronic payloads located on the armored military vehicles. Both axes of the SPT system is direct–driven. This reduces the lower accuracy disadvantage caused by the backlash and flexibility of the geared systems. It also has multi–turn slip rings on both axes which improve the accuracy greatly. The Stewart platform developed in this study is used for the inertial stabilization development tests of the SPT system.

SPT system has an inertial measurement unit (IMU) for inertial stabilization feedback. Stabilization controller is decided to be a sliding mode controller with the hyperbolic tangent function instead of the signum function. There are friction compensation algorithms to cope with the static and sliding frictions. Inertial stabilization accuracy of the SPT system is lower than 0.3 mrad on both axes. Stabilization performance can be seen in Figures 2.20 and 2.21. Disturbances acting on the SPT system in the tests are created by the Stewart platform developed in this thesis.

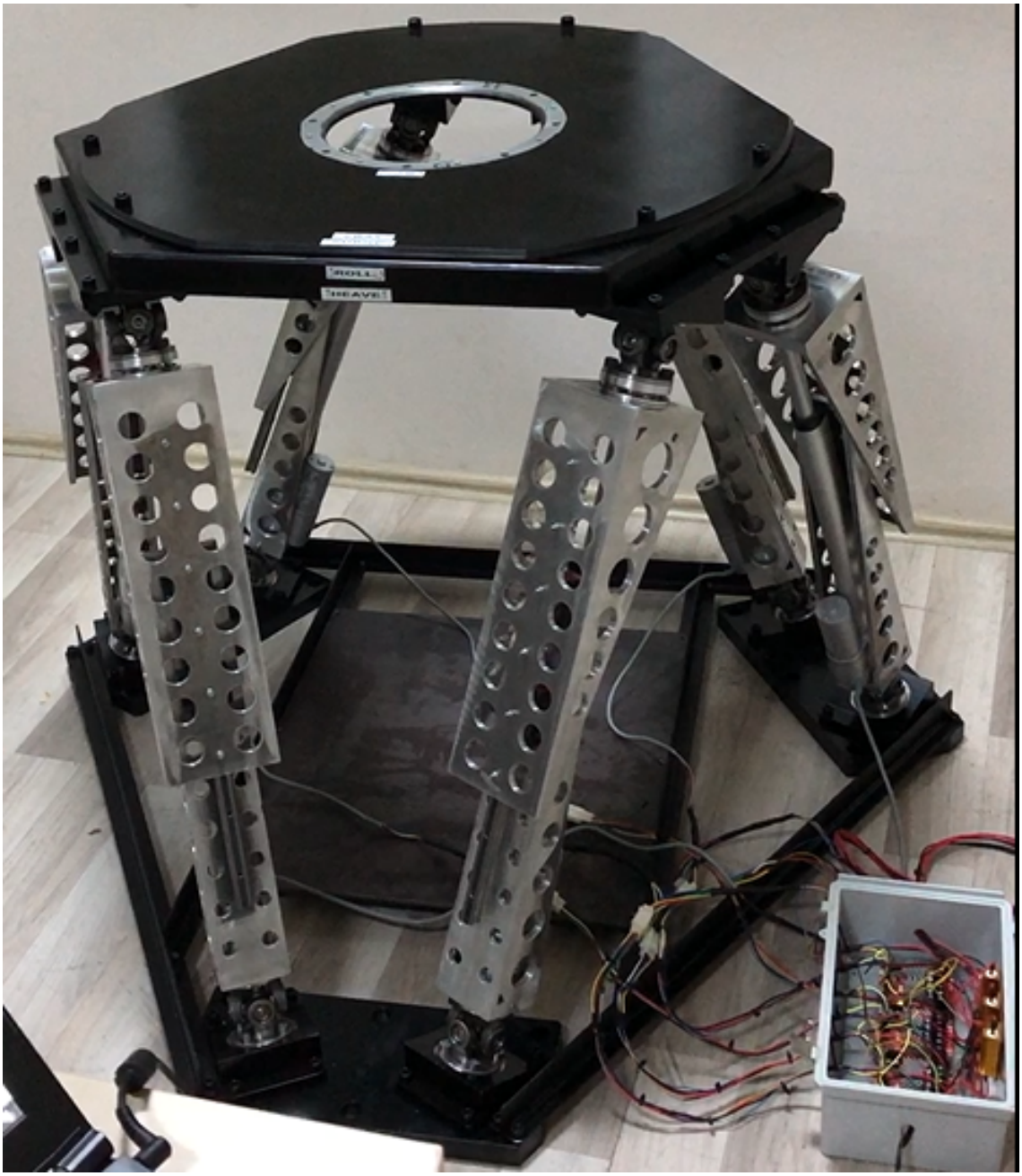

Figure 2.18. Completed Stewart platform

Figure 2.19. Stabilized Pan–Tilt system developed by Teknodef

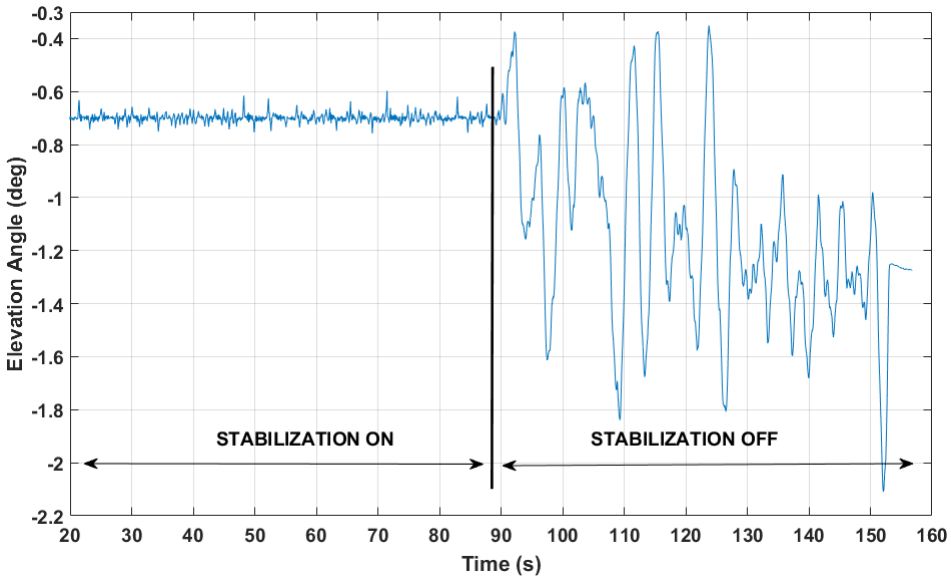

Figure 2.20. Inertial stabilization accuracy of the SPT system on elevation axis

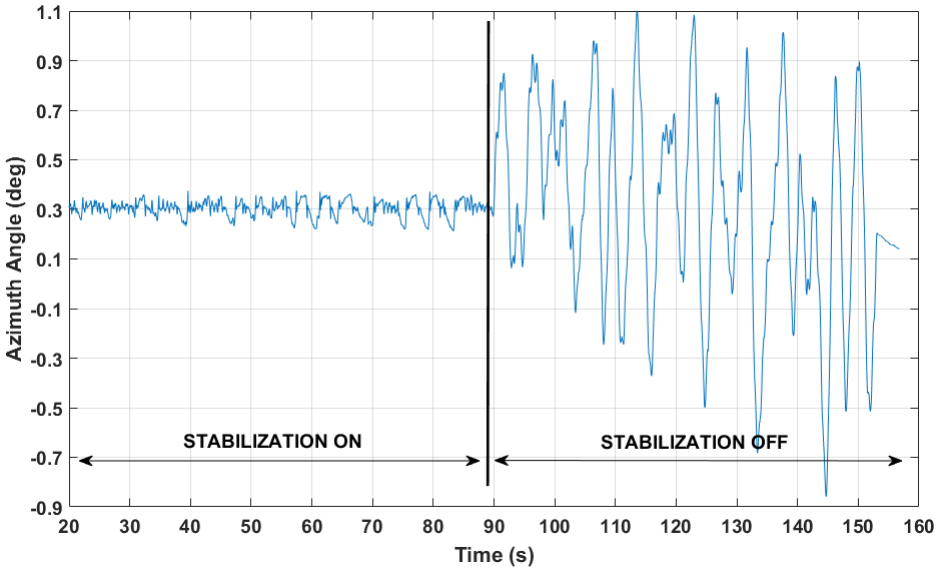

Figure 2.21. Inertial stabilization accuracy of the SPT system on azimuth axis

CHAPTER 3

MATHEMATICAL MODELING

In this chapter, inverse kinematic model of the Stewart platform, degrees of freedom calculations and kinematic model verification are presented.

3.1. Number of Degrees of Freedom Calculations

Number of degrees of freedom of the Stewart platform can be calculated with Grüblerâs degrees of freedom formula (Albayrak, 2005)

$$\text{dof} = m \times (N - 1 - J) + \sum_{i=1}^{J} f_i \tag{3.1}$$

where m is the dimension of the end–effector space, N is the total number of links, J is the total number of joints, and f_i represents the degrees of freedom of each joint. The Stewart platform has 6 legs connecting the base plate to the top plate, and each leg consists of two links and a universal joint, a prismatic joint, and a spherical joint as seen in Figure 3.1. Since each leg has 2 links, there is a total of 12 links in the legs, and adding base and top plate makes 14 links in total so $N = 14$. Each leg has 3 joints with 1 degrees of freedom coming from the prismatic joint, 2 degrees of freedom coming from the universal joint, and 3 degrees of freedom coming from the spherical joint yields the total number of joints being obtained as $J = 18$ and the summation on the right hand side of (3.1) is calculated as

$$\sum_{i=1}^{J} f_i = 6 \times (1 + 2 + 3) = 36. \tag{3.2}$$

Noting that the mechanism moves in 3 dimensional space, so $m = 6$. Substituting the above expression into Grübler's formula in (3.1), number of degrees of freedom of a Stewart platform can be calculated as

$$\text{dof} = 6 \times (14 - 1 - 18) + 36 = 6. \tag{3.3}$$

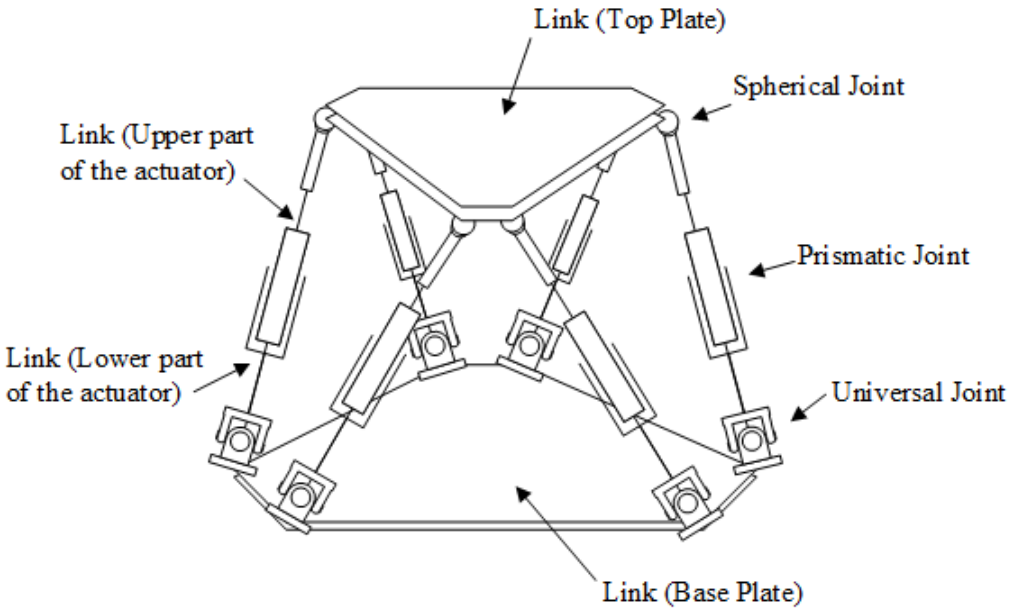

Figure 3.1. Links and joints of the Stewart platform

3.2. Inverse Kinematic Model of the Stewart Platform

To change the position and orientation of the Stewart platformâs top plate, length of the legs must be changed. For an arbitrary position and orientation of the top plate, required leg lengths must be known which can be determined by the inverse kinematics calculations of the Stewart platform.

All connection points of legs, base plate and top plate can be seen in Figure 3.2. In this figure, connection points on the top plate are vectors defined in top plate coordinate system, and points on the base plate are the vectors that are defined in base plate coordinate system.

To obtain the leg lengths, connection points on the top plate should be represented in the coordinate system of the base plate. The orientation of the top plate is represented in the coordinate system of the base plate via a rotation matrix, and the position of the top plate is represented via a displacement vector.

After pre–multiplying the connection point vectors of the top plate with rotation matrix R, and adding the displacement vector $\vec{d}$, these point vectors are represented in the coordinate system of the base plate. Subtracting the connection point vectors of the base plate from the connection point vectors of the top plate (represented in coordinates of base

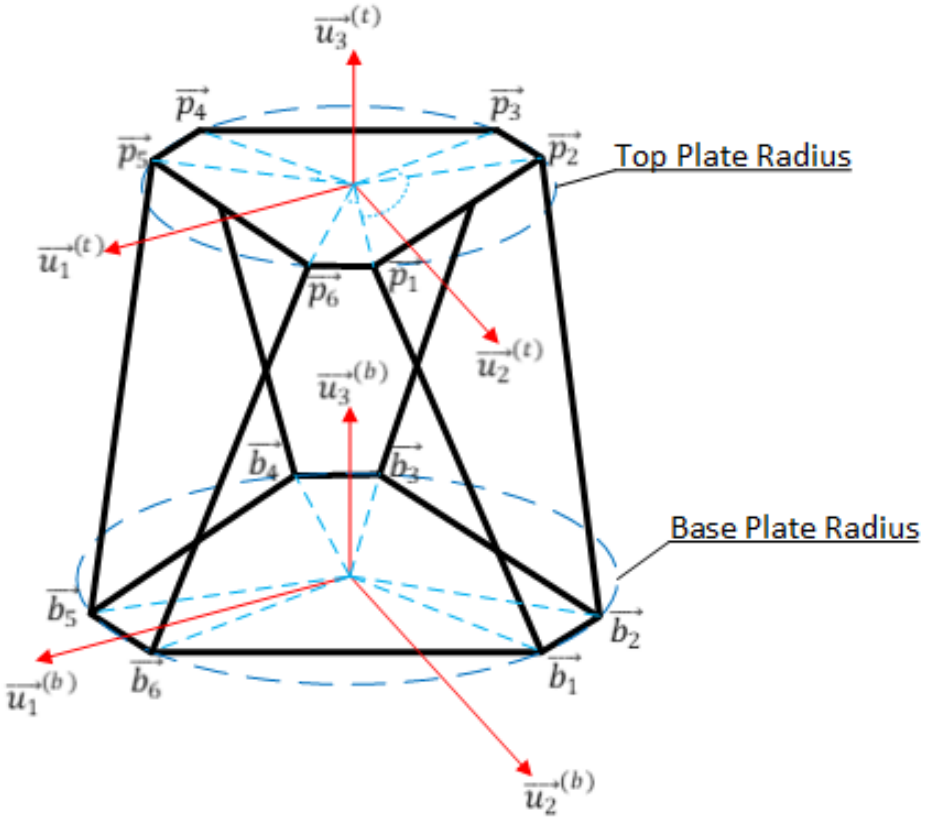

Figure 3.2. Kinematic terminology of the Stewart platform

plate) results in the leg vectors where the lengths of the leg vectors give the required leg lengths for the desired orientation and position of the top plate (Toz and Küçük, 2016). Thus the required leg lengths, shown with l_i, $i = 1, \cdots, 6$, can be calculated from

$$l_i = \|(R\vec{p_i} + \vec{d}) - \vec{b_i}\| \tag{3.4}$$

where

$$R = R_z(\gamma)R_y(\beta)R_x(\alpha) \tag{3.5}$$

with $R_x(\alpha)$, $R_y(\beta)$ and $R_z(\gamma)$ being the rotation matrices around x, y and z axes defined as

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix}, \tag{3.6}$$

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}, \tag{3.7}$$

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.8}$$

3.2.1. Inverse Jacobian Matrix

After obtaining the required leg lengths corresponding to position and orientation of the upper platform, inverse Jacobian matrix, J^{-1}, can be calculated. Partial derivative of vector of all 6 leg lengths, shown with $\vec{l} = [l_1, l_2, l_3, l_4, l_5, l_6]^T$, with respect to the variable vector $\vec{v}$ containing the orientation and position demands of the top plate gives the inverse Jacobian Matrix (Toz and Küçük, 2016). The variable vector $\vec{v}$ contains six variables where three of them are for orientation demand which are rotation around x axis α, rotation around y axis β and rotation around z axis γ while the other three variables are the components of the displacement demand which are $x^{'}$, $y^{'}$ and $z^{'}$. Inverse Jacobian matrix can be obtained as follows

$$\frac{\partial \vec{l}}{\partial t} = J^{-1} \frac{\partial \vec{v}}{\partial t}, \tag{3.9}$$

$$\Rightarrow J^{-1} = \frac{\partial \vec{l}}{\partial t} \frac{\partial t}{\partial \vec{v}}, \tag{3.10}$$

$$J^{-1} = \frac{\partial \vec{l}}{\partial \vec{v}} = \begin{bmatrix} \frac{\partial l_1}{\partial \alpha} & \frac{\partial l_1}{\partial \beta} & \frac{\partial l_1}{\partial \gamma} & \frac{\partial l_1}{\partial x^{'}} & \frac{\partial l_1}{\partial y^{'}} & \frac{\partial l_1}{\partial z^{'}} \\ \frac{\partial l_2}{\partial \alpha} & \frac{\partial l_2}{\partial \beta} & \frac{\partial l_2}{\partial \gamma} & \frac{\partial l_2}{\partial x^{'}} & \frac{\partial l_2}{\partial y^{'}} & \frac{\partial l_2}{\partial z^{'}} \\ \frac{\partial l_3}{\partial \alpha} & \frac{\partial l_3}{\partial \beta} & \frac{\partial l_3}{\partial \gamma} & \frac{\partial l_3}{\partial x^{'}} & \frac{\partial l_3}{\partial y^{'}} & \frac{\partial l_3}{\partial z^{'}} \\ \frac{\partial l_4}{\partial \alpha} & \frac{\partial l_4}{\partial \beta} & \frac{\partial l_4}{\partial \gamma} & \frac{\partial l_4}{\partial x^{'}} & \frac{\partial l_4}{\partial y^{'}} & \frac{\partial l_4}{\partial z^{'}} \\ \frac{\partial l_5}{\partial \alpha} & \frac{\partial l_5}{\partial \beta} & \frac{\partial l_5}{\partial \gamma} & \frac{\partial l_5}{\partial x^{'}} & \frac{\partial l_5}{\partial y^{'}} & \frac{\partial l_5}{\partial z^{'}} \\ \frac{\partial l_6}{\partial \alpha} & \frac{\partial l_6}{\partial \beta} & \frac{\partial l_6}{\partial \gamma} & \frac{\partial l_6}{\partial x^{'}} & \frac{\partial l_6}{\partial y^{'}} & \frac{\partial l_6}{\partial z^{'}} \end{bmatrix}. \tag{3.11}$$

3.3. Simulink Implementation of the Mathematical Model

Given the desired orientation and position of the top plate, required leg lengths are calculated using MATLAB Simulink's matrix operation blocks. A simplified block diagram representation of the flow can be seen in Figure 3.3, and the detailed structure of the Simulink model is given in Appendix A. Desired orientation and position of the top plate are the inputs of the system. Orientation and position has six components which are displacements on three axes and rotations around three axes. Each entry is considered to be a sinusoid that has amplitude, frequency, phase and offset as parameters. As an example, flow diagram for construction of the desired displacement on the z-axis is shown in Figure 3.4.

After the construction of the desired orientation and position demands, inverse kinematic calculations are done as shown in the flow diagram in Figure 3.5. Results of these calculations are the required lengths of each leg to meet the desired orientation and position demands.

3.3.1. Verification of the Simulink Model

Stewart platform can be simulated using MATLAB Simulink Simscape Multibody software. Mechanical parts, joints and links of the Stewart platform are formed in Simscape Multibody with the exact parameters given in Tables 2.1 and 2.2. The outcome of the Simscape Multibody can be seen in Figure 3.6.

All six individual movements of the Stewart platform are simulated in Simscape using sinusoidal inputs for the displacements and orientations. In Figures 3.7, 3.8 and 3.9, rotation of the top plate around z-axis, y-axis and x-axis obtained for sinusoidal reference trajectories with 5 degrees of amplitude and 0.5 Hz of frequency are presented.

In Figures 3.8 and 3.9, it is observed that the orientation of the top plate can track the reference trajectory with some error. This may be caused by several reasons. The velocities of the linear actuators are limited and if an arbitrary trajectory requires rapid leg length changes, orientation of the top plate may differ slightly from the reference trajectory. Another reason that can cause tracking error may be the reference trajectory being in the vicinity of singularities of the Stewart platform. To be able to pinpoint the

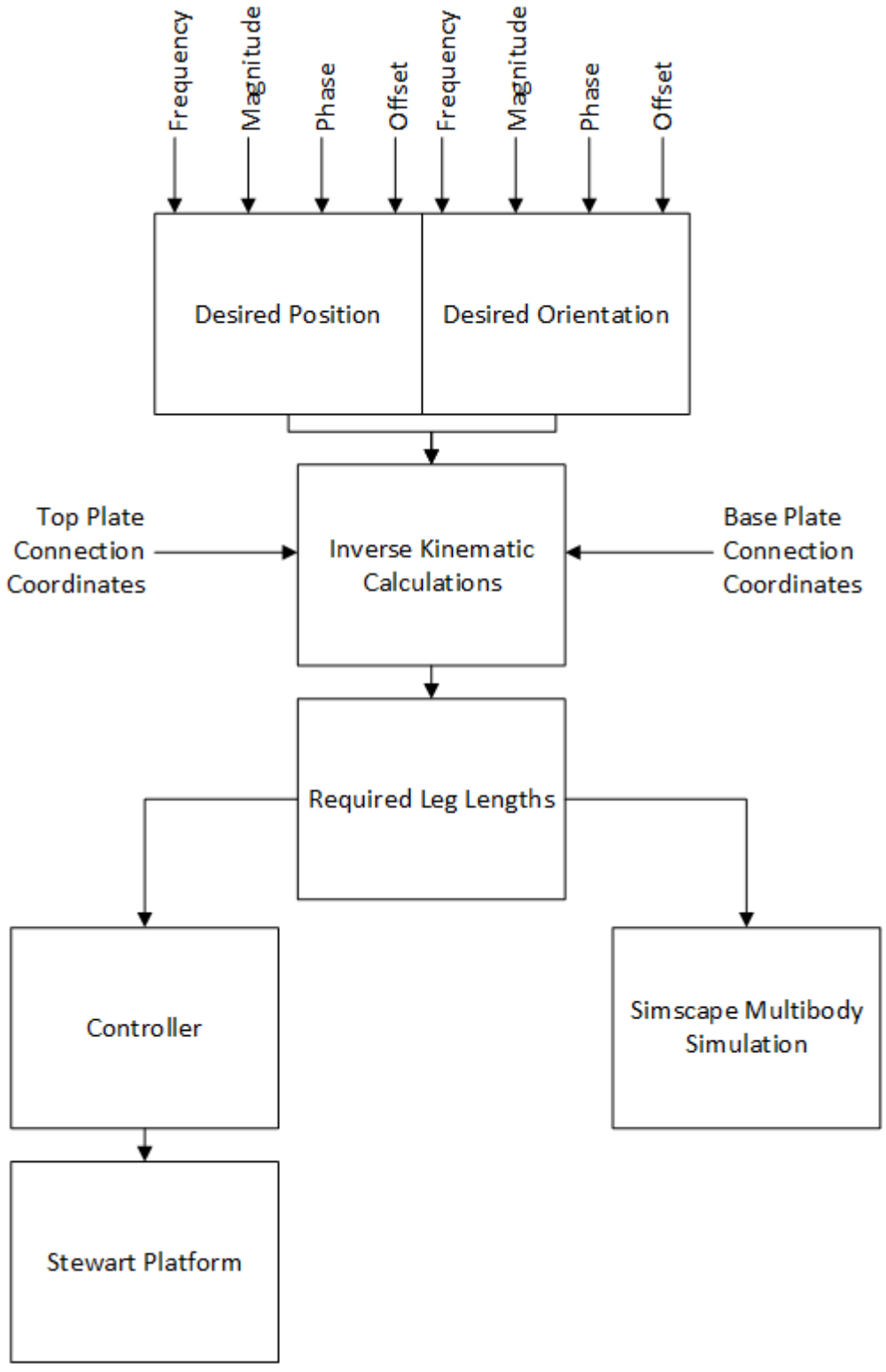

Figure 3.3. Simplified required leg length calculation algorithm

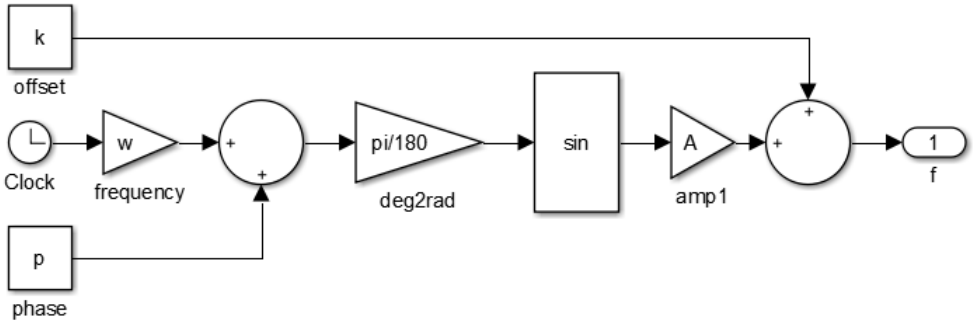

Figure 3.4. The flow diagram for a sinusoidal desired displacement for z-axis

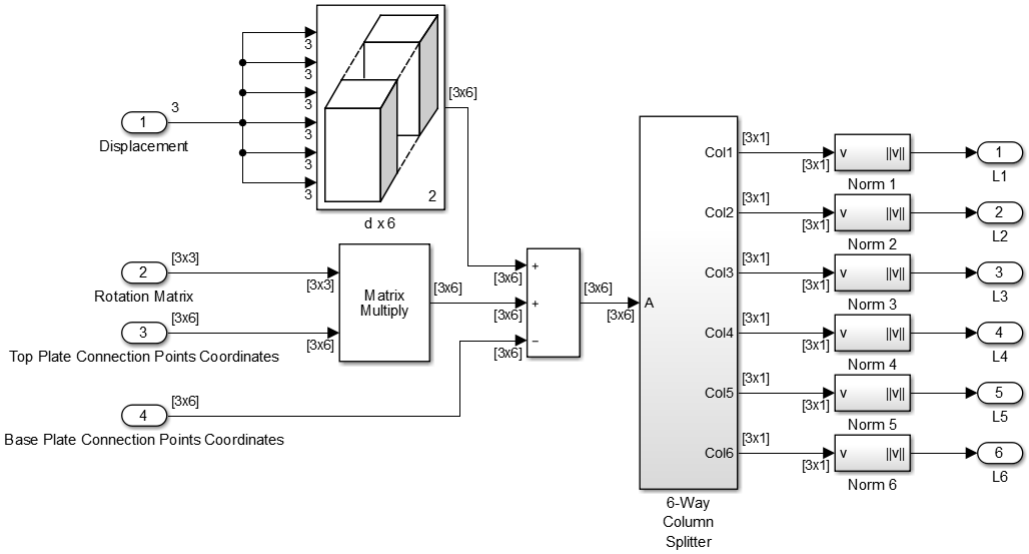

Figure 3.5. Inverse kinematic calculations in Simulink

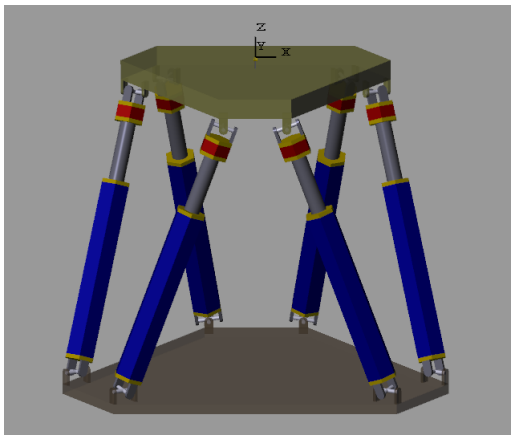

Figure 3.6. Stewart platform model in Simscape Multibody

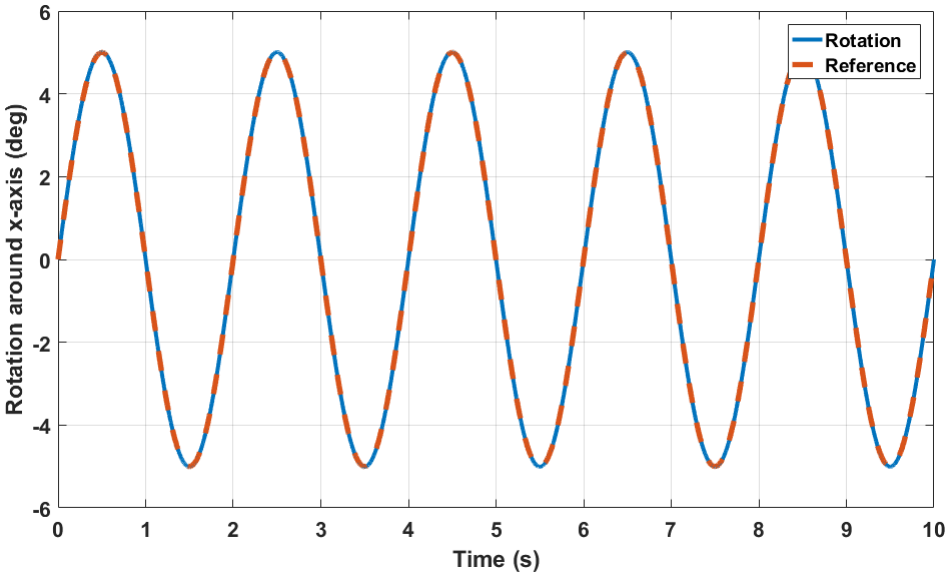

Figure 3.7. Rotation of the top plate around z-axis

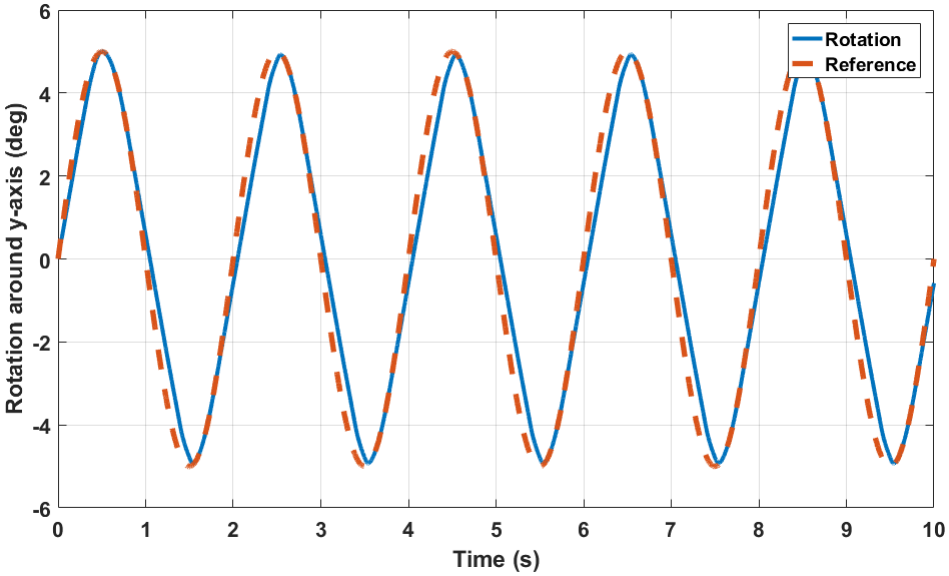

Figure 3.8. Rotation of the top plate around y-axis

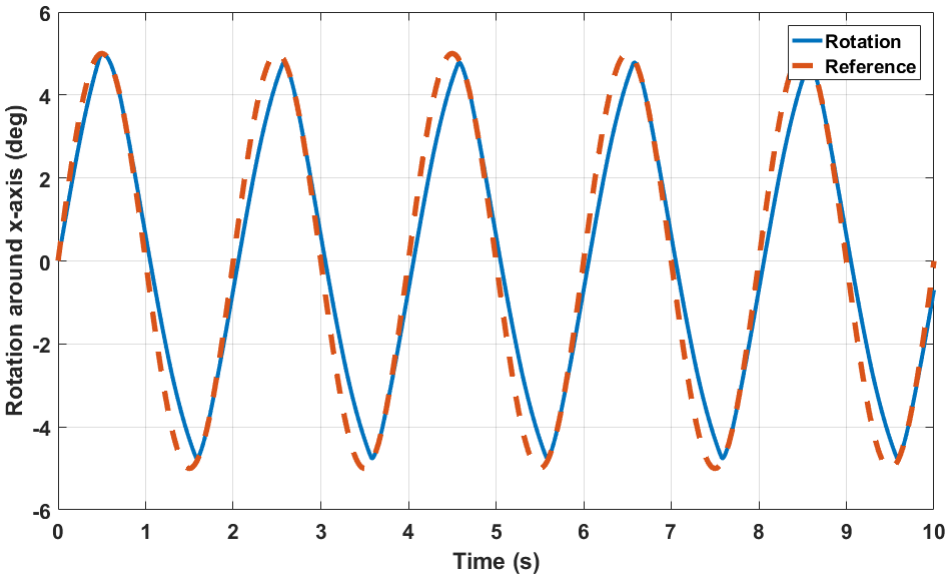

Figure 3.9. Rotation of the top plate around x-axis

reason of this tracking error, a simulation where velocity limits of the legs are not taken into account is conducted with the same amplitude and frequency of the desired trajectory in Figure 3.9 where the results are shown in Figure 3.10. A closer look at Figures 3.9 and 3.10 reveals that the tracking error is decreased significantly.

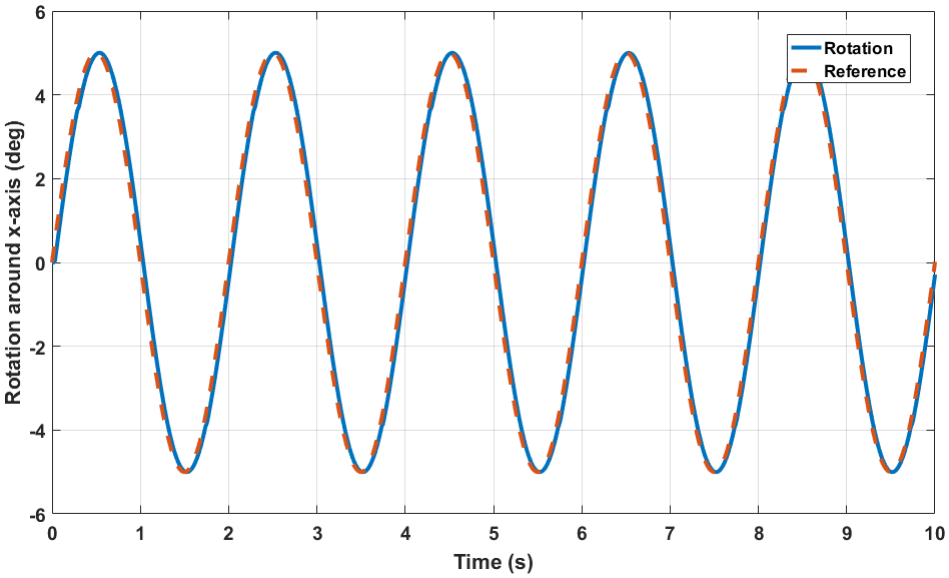

Figure 3.10. Rotation of the top plate around x-axis without leg velocity limits

In Figures 3.11, 3.12 and 3.13, displacement of the top plate for each axis are presented for sinusoidal reference trajectories having 5 cm of amplitude and 0.2 Hz of frequency.

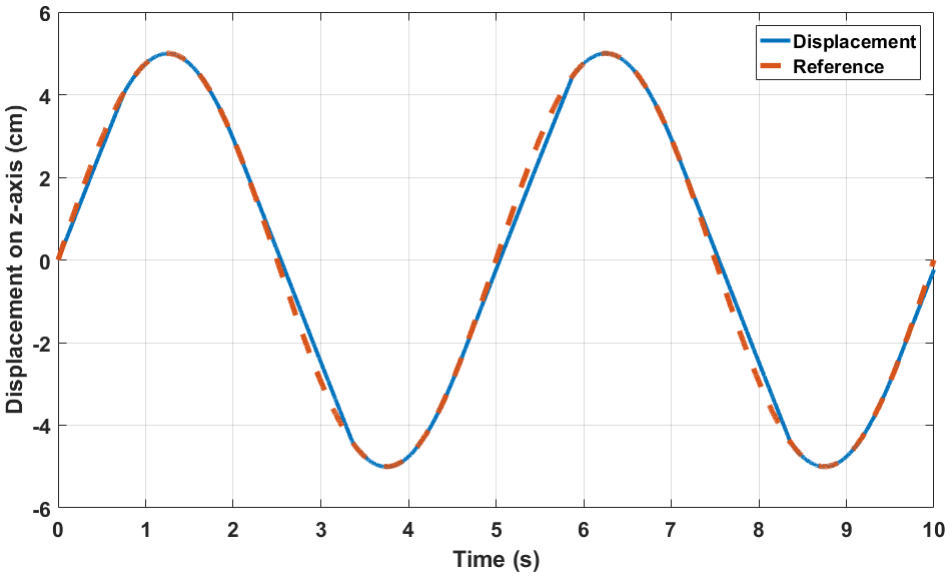

Figure 3.11. Displacement of the top plate on z-axis

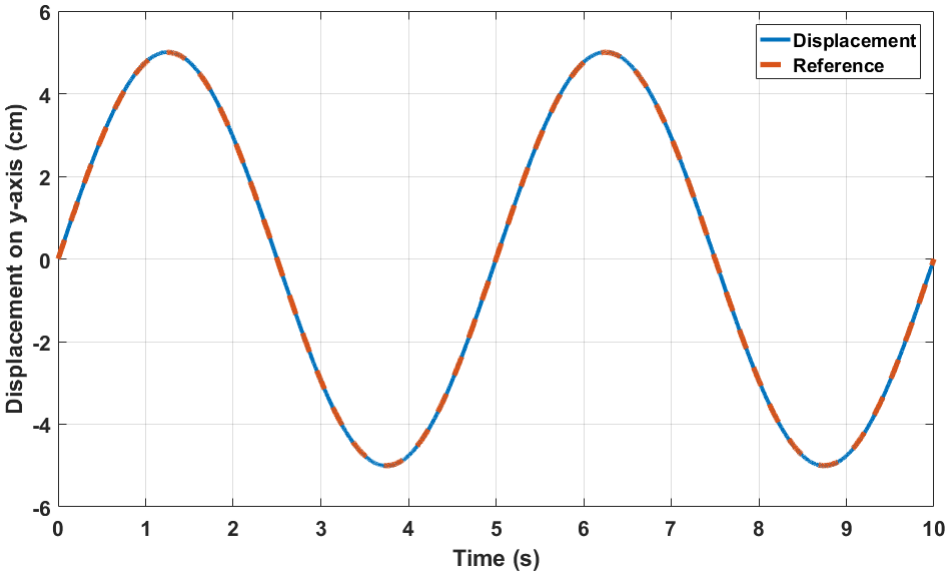

Figure 3.12. Displacement of the top plate on y-axis

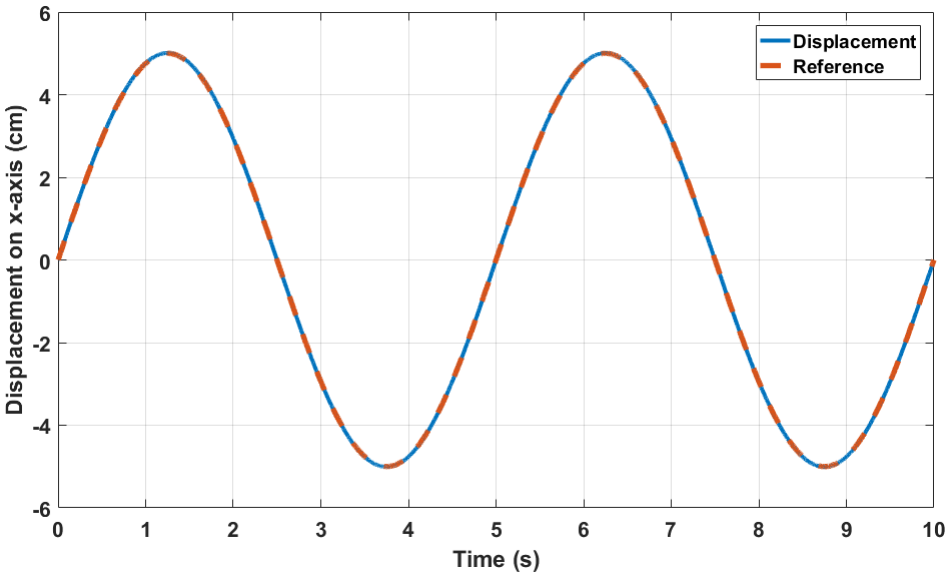

Figure 3.13. Displacement of the top plate on x-axis

Finally, leg lengths are shown in Figure 3.14 for a trajectory which is the combination of sinusoidal inputs on displacement on x-axis, y-axis, z-axis and rotation around x-axis.

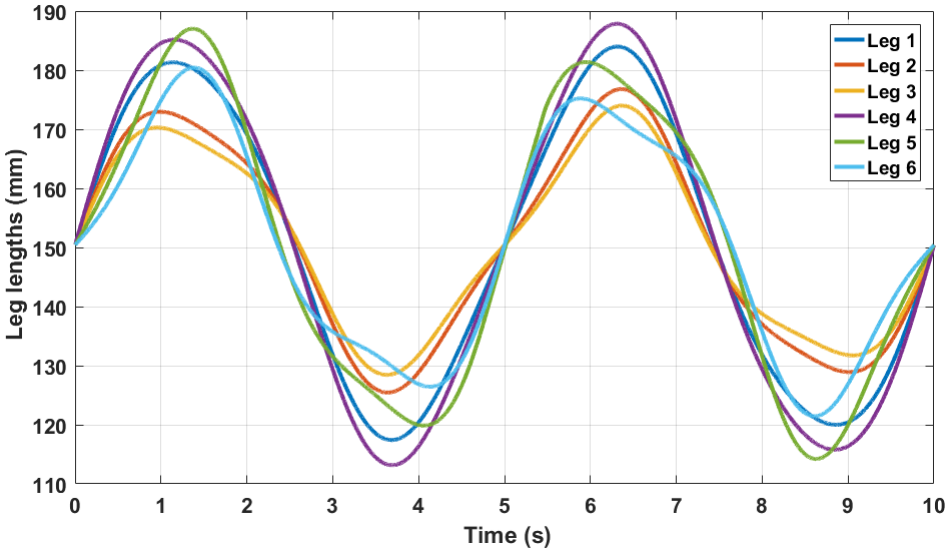

Figure 3.14. Leg lengths while performing a rotation and displacement trajectory

CHAPTER 4

CONTROL OF STEWART PLATFORM

This chapter represents the control algorithm applied on the Stewart platform, tuning of the controller parameters and the experimental results.

4.1. PID Controllers

PID, short for proportional integral derivative, is the most common control type in the industry mostly because of its mathematical simplicity and it being applied to wide range of systems (Ogata, 2001). Almost 90% of the controllers in the industry utilizes PID or modified PID schemes. The input to the controller is the error which is the difference between reference trajectory and the system output.

4.1.1. P Controller

In P controllers, control input is proportional to the error value – hence the name proportional controllers. Error value is multiplied with a constant K_p, that is usually named proportional gain

$$u(t) = K_p e(t). \tag{4.1}$$

Proportional control increases the response speed, but as a drawback it causes greater transient overshoot. Decreasing K_p decreases the overshoot, but it also decreases the disturbance rejection of the system. Therefore, K_p should be tuned well to find an optimal performance between response time and transient overshoot.

4.1.2. PI Controller

PI controller is achieved by adding an integrator term to the P controller. Control input is the summation of a term that is proportional to the error and the integral of the error. Integral term is multiplied with an integral constant K_i. Integral of the error is calculated from the initial time to the current time, and the control input is evaluated from

$$u(t) = K_p e(t) + K_i \int_0^t e(t')dt' \tag{4.2}$$

which can alternatively be represented as

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t')dt' \right) \tag{4.3}$$

where T_i is the integral time constant.

Integral term accumulates the error of the past. This way, PI control reduces the steady–state errors that P control wasn't able to. On the other hand, accumulated error can increase the overshoot. Even when the error is zero, integral term can be non–zero. This can also result in oscillations in the steady state. Moreover, by introducing the integral term, system can be subject to the integral windup problem. Integral windup happens when the actuator limits are exceeded by the controller. For example, the linear DC motors of the Stewart platform have a velocity limit. If the setpoint of a speed controller is larger than the linear DC motors' velocity limit, integrator will continue to wind up and result in much larger output values. There are many anti-windup algorithms to prevent this.

4.1.3. PID Controller

PID controller introduces a derivative term in addition to the proportional and integral terms. Derivative term is a time derivative of the error signal multiplied by a derivative constant K_d. Derivative term is proportional to the rate of change of the error. It can increase the speed of the system response. While integral term stores the past error

as a memory of the controller, derivative term predicts the future system behaviour and can increase the system stability. It can damp out the overshoot and decrease the settling time. However, noises on the feedback and error signal will be amplified by the derivative operation. Also derivative term can not act on the steady state error itself, because the derivative of a constant error will be zero. The general form of a PID controller has the following structure

$$u(t) = K_p e(t) + K_i \int_0^t e(t')dt' + K_d \frac{de(t)}{dt} \tag{4.4}$$

which can alternatively be represented as

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t')dt' + T_d \frac{de(t)}{dt} \right) \tag{4.5}$$

where T_d is the derivative time constant.

PID controller can act on the system based on it's past, present and future values with the help of the integral, proportional and derivative terms.

4.2. Cascade Control

Cascade control is a method that use sequential two or more controllers together (Ellis, 2004). The block diagram of an example cascade control algorithm where position and speed are the variables to be controlled is presented in Figure 4.1. The output of the first controller, which is the position controller, is treated as the setpoint of the second controller, which is the speed controller. This method can provide a better reaction to disturbances (Lee et al., 2006).

Roboclaw 2×15A motor drivers have cascade control system that uses the output of the position controller as the setpoint of the speed controller and the output of the speed controller as the PWM duty cycle.

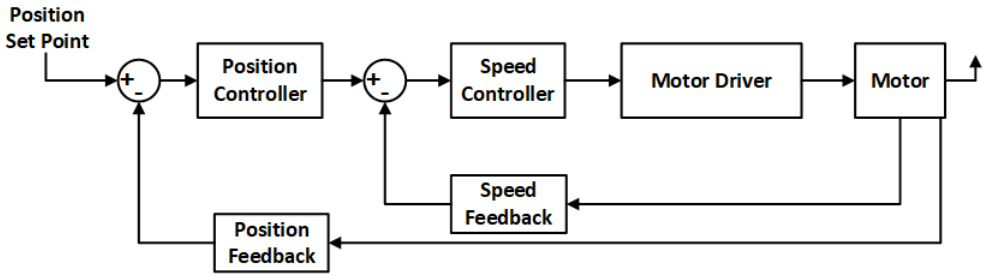

Figure 4.1. Cascade control

4.3. Motor Driver PID Parameters Tuning

4.3.1. Speed Controller Tuning

Controllers of the linear DC motors for each leg are tuned by using the ION Studio software. Both speed and position controllers are chosen as PI controllers. In cascade control systems, innermost controller is tuned first (Ellis, 2004), so the speed controller, which is the innermost controller, is tuned first. Each leg is removed from the platform and tuning is done one by one. Since the dynamic model is not available, controller tuning is done with trial and error. Step response of the speed controller can be seen in Figure 4.2. Best speed controller performance was achieved with the control parameters given below

$$K_p = 9, \quad K_i = 0.8. \tag{4.6}$$

4.3.2. Position Controller Tuning

After tuning the speed controller, tuning of the position controller is carried on again via trial and error method. PI parameters that resulted in the best position control performance are given below

$$K_p = 10, \quad K_i = 0.01. \tag{4.7}$$

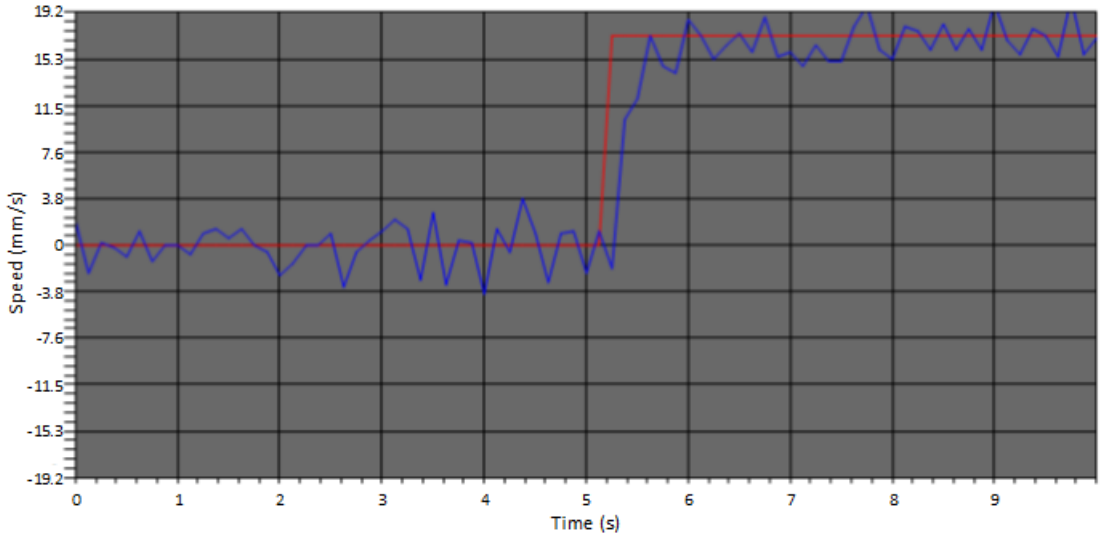

Figure 4.2. Step response of the speed controller

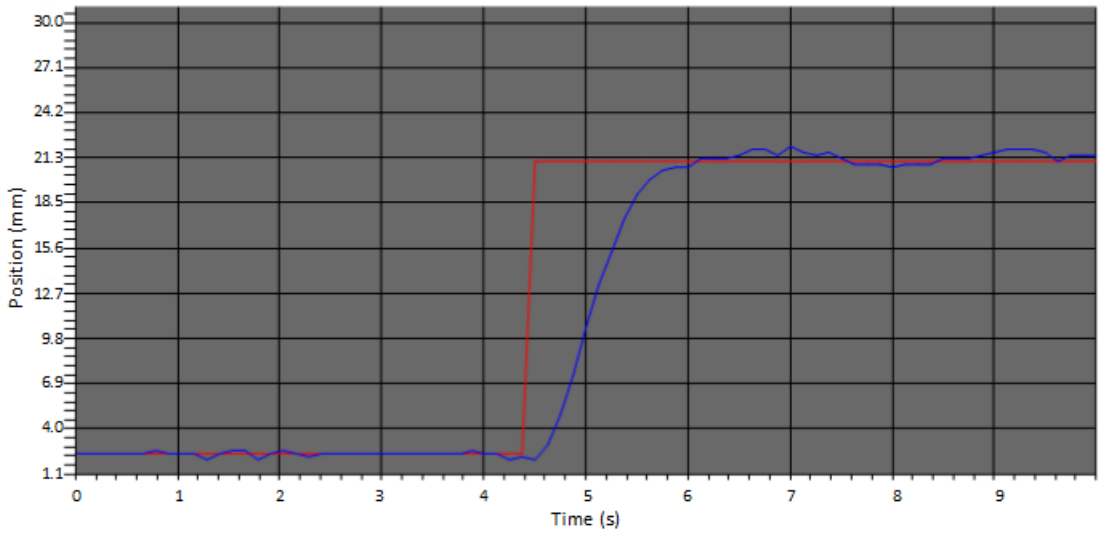

Figure 4.3. Step response of the position controller

Step response of the position controller can be seen in Figure 4.3.

4.4. Verification of the Motion of the Top Plate

The correct motion of the top plate is the ultimate aim of this study. Top plate is expected to follow the given trajectory without significant error. The Simulink model embedded into the target computer is given in Appendix B. To measure the rotation and position of the top plate, an inertial measurement unit (IMU) is placed on the top plate. The IMU can measure the angular velocity with a 3-axis gyroscope and acceleration with

a 3-axis accelerometer. Data rate of the IMU is 30 Hz. Rotation information is obtained by integrating the angular velocity of the corresponding axis. Displacement information is obtained by integrating the acceleration of the corresponding axis twice.

4.4.1. Rotation

Rotation of the top plate can be obtained by integrating the gyroscope data, and before the integration, a basic drift compensation algorithm is applied to the collected gyroscope data. Desired rotation and actual top plate rotation is shown below for three different rotation demands.

First rotation demand is a sinusoidal rotation around x-axis with 4 deg amplitude and 0.1 Hz frequency. Figure 4.4 shows the actual rotation and desired rotation in x-axis.

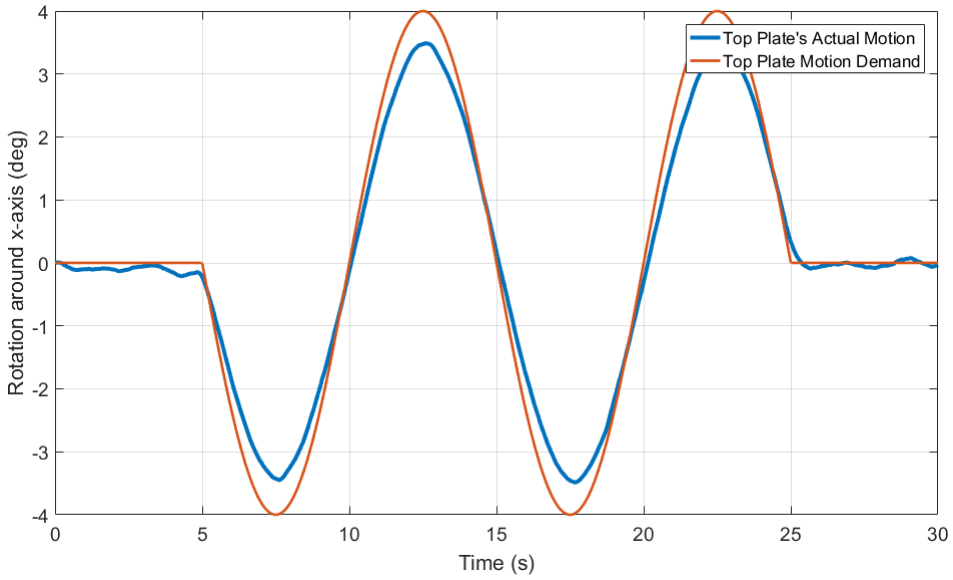

Figure 4.4. Top plate rotation around x-axis with 4 deg 0.1 Hz sinusoidal input

Second rotation demand is a sinusoidal rotation around z-axis with 5 deg amplitude and 0.2 Hz frequency. Figure 4.5 shows the actual rotation and desired rotation in z-axis.

Third rotation demand is a sinusoidal rotation around z-axis with 0.5 deg amplitude and 1 Hz frequency. Figure 4.6 shows the actual rotation and desired rotation in z-axis.

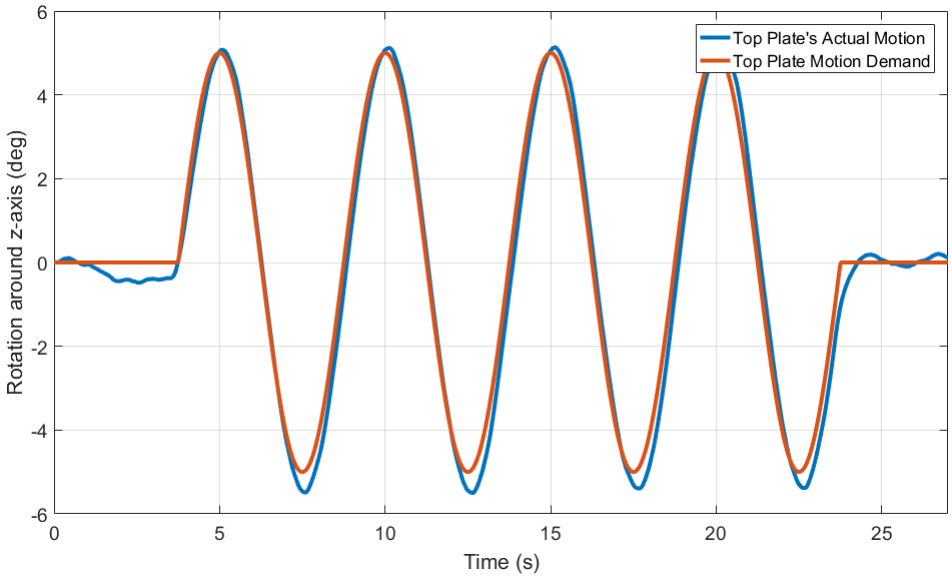

Figure 4.5. Top plate rotation around z-axis with 5 deg 0.2 Hz sinusoidal input

4.4.2. Displacement

Displacement is measured by taking the integral of the collected accelerometer data twice. Desired displacements and actual displacements of the top plate are shown in Figure 4.7, Figure 4.8 and Figure 4.9 for 3 different desired displacement trajectories. First desired displacement trajectory is a sinusoidal on z-axis with 0.2 cm amplitude and 1 Hz frequency. Second input is a sinusoidal again on z-axis with 10 cm amplitude and 0.025 Hz frequency. Third input is another sinusoidal on x-axis with 0.5 cm amplitude and 1 Hz frequency.

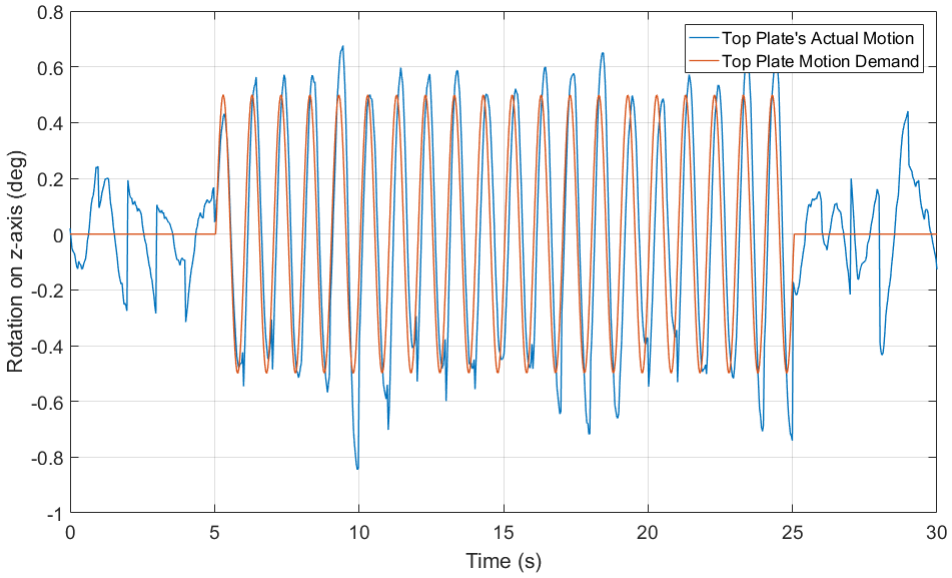

Figure 4.6. Top plate rotation around z-axis with 0.5 deg 1 Hz sinusoidal input

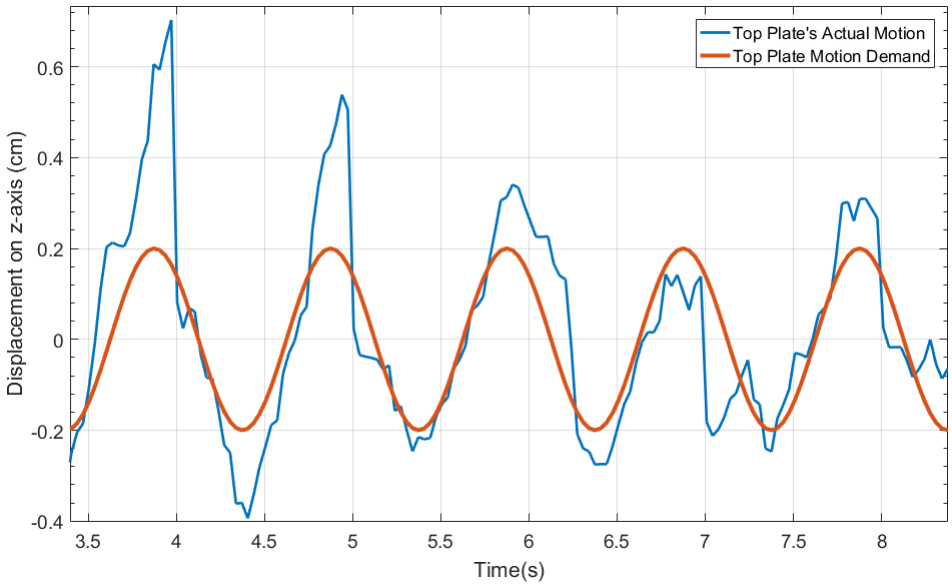

Figure 4.7. Top plate displacement on z-axis with 0.2 cm 1 Hz sinusoidal input

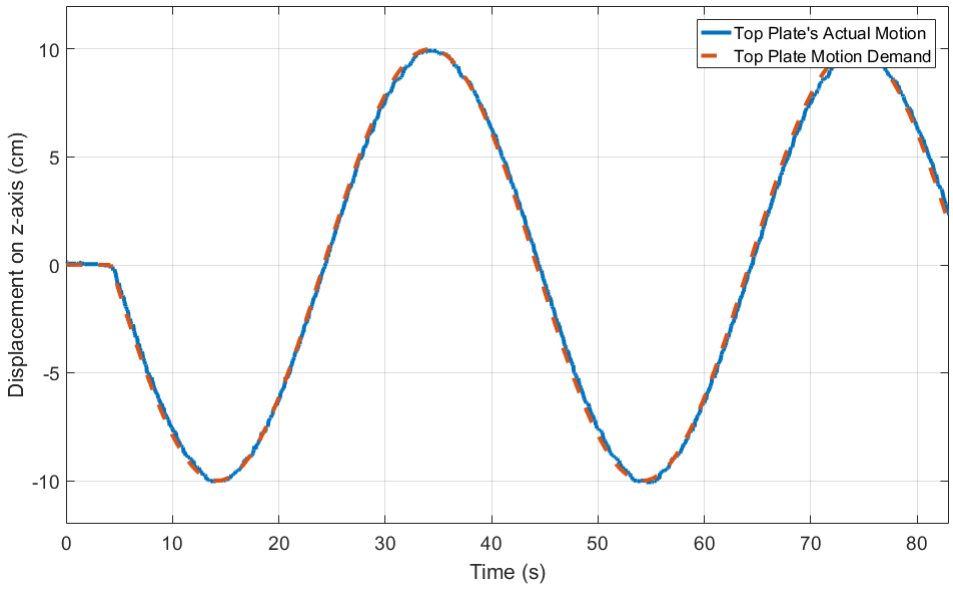

Figure 4.8. Top plate displacement on z-axis with 10 cm 0.025 Hz sinusoidal input

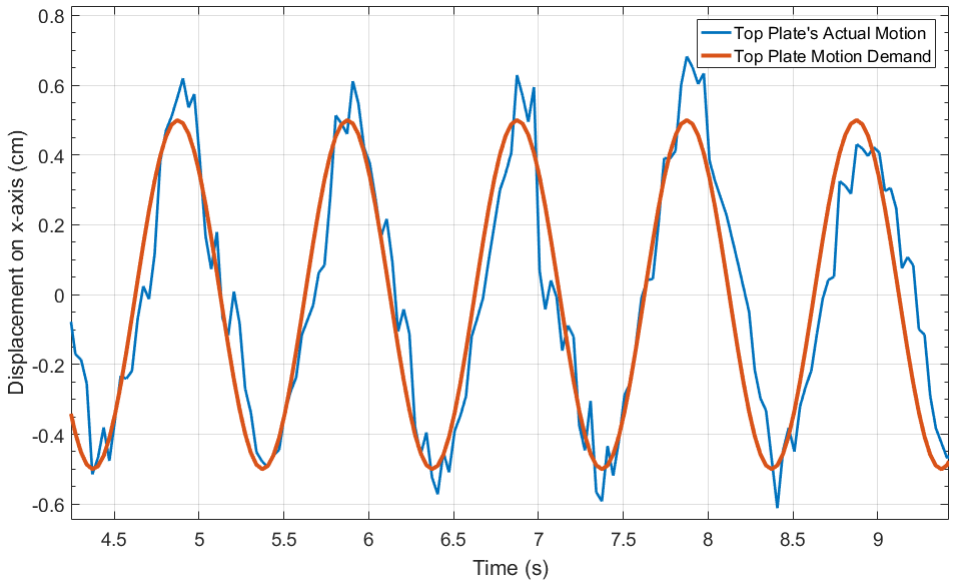

Figure 4.9. Top plate displacement on x-axis with 0.5 cm 1 Hz sinusoidal input

CHAPTER 5

CONCLUSIONS

Inertial stabilization of remote controlled weapon systems mounted on military vehicles is extremely important in modern warfare. A mechanism that has a crucial role in the development phases of remote controlled weapon systems is a simulator that mimics the behavior of a vehicle moving in harsh terrain. In this thesis, design and development of a Stewart platform that is used in the stabilization tests of a remote controlled weapon system was presented. Specifically, mechanical design criterion were described and discussed as well as its mechanical properties. System configuration, architecture and network topology were presented along with detailed descriptions. Important properties of the hardware and harness were also given. Communication between the hardware components were presented. Link and joint structure of the Stewart platform was given. Platform's degree of freedom was verified using Grübler's formula. Inverse kinematic model of the Stewart platform was constructed using the mechanical parameters of the CAD model. Kinematic model was implemented to the Simulink model and verified via numerical simulations. The control structure, including a brief description of the motor driver's controller structure and the cascade control method which was preferred to control both position and speed were presented. Both rotation and displacement tracking performance of the proposed control strategy was experimentally verified on the developed Stewart platform where the motion of the top plate was captured with an inertial measurement unit. Satisfactory performance was obtained in the experimental results.

While satisfactory performance was obtained in the experimental results, there are several possible extensions that could be considered as future work. Specifically, to increase the performance of the developed Stewart platform, a more sensitive kinematic model can be constructed to minimize the mechanical differences between CAD model and the actual manufactured Stewart platform. Differences between the CAD model based on which the kinematic model was obtained and the actual platform could result in an inaccurate kinematic model, thus incorrect calculations of the required leg lengths for an arbitrary trajectory of the top plate.

In this thesis, PID controllers were examined experimentally. However, the performance of the Stewart platform could also be improved via utilizing advanced control algorithms. To do this, the control loops of the motor drivers must be integrated to the target computer instead of using the built-in PID controllers and then nonlinear control algorithms could be utilized.

Also, instead of the one utilized in the current design, a motor driver with current or torque control capabilities could be used. Using a driver with a current or torque controller, nonlinear control algorithms could be designed to actuate the legs dynamically.

Another improvement to increase the performance of the Stewart platform could be using linear DC motors having higher velocity limits. Specifically, in the current design, the linear DC motors have 19 mm/sec velocity limit, and the motion of the top plate are limited with these limitations. Increasing the velocity of the legs will allow tracking of desired rotation and displacement motions for the top plate with higher frequency and amplitude values.

There are several research avenues that can be considered as possible future works. The kinematic singularities of the developed Stewart platform can be found analytically to be used in the design of the desired trajectories. Obtaining a dynamic model of the developed Stewart platform could be another future work. The dynamic model can be used in the design of control algorithms to improve the tracking performance of the platform. Another way to improve the tracking performance could be designing advanced control algorithms.

REFERENCES

Albayrak, Onur (2005). Modeling and real-time control system implementation for a Stewart platform. Master's thesis, Middle East Technical University.

ASTM International, West Conshohocken, PA (2018). Astm b258-18, standard specification for standard nominal diameters and cross-sectional areas of awg sizes of solid round wires used as electrical conductors. www.astm.org.

Cirillo, Andrea and Cirillo, Pasquale and De Maria, Giuseppe and Marino, Alessandro and Natale, Ciro and Pirozzi, Salvatore (2017). Optimal custom design of both symmetric and unsymmetrical hexapod robots for aeronautics applications. *Robotics and Computer-Integrated Manufacturing 44*, 1–16.

D. Stewart (1965). A platform with six degrees of freedom. *Proceedings of the Institution of Mechanical Engineers 180*(1), 371–386.

Dongsu, Wu and Hongbin, Gu (2007). Adaptive sliding control of six-dof flight simulator motion platform. *Chinese Journal of Aeronautics 20*(5), 425–433.

Ellis, George (2004). *Control system design guide: using your computer to understand and diagnose feedback controllers*. Butterworth-Heinemann.

Gürbüz, Sarper (2006). Design and construction of a six degree of freedom platform. Master's thesis, Middle East Technical University.

Ho, Paul TP and Altamirano, Pablo and Chang, Chia-Hao and Chang, Shu-Hao and Chang, Su-Wei and Chen, Chung-Cheng and Chen, Ke-Jung and Chen, Ming-Tang and Han, Chih-Chiang and Ho, West M and others (2009). The yuan-tseh lee array for microwave background anisotropy. *The Astrophysical Journal 694*(2), 1610.

Karayumak, Türker (2011). *Modeling and stabilization control of a main battle tank*. Ph. D. thesis, Middle East Technical University.

Lee, Yongho and Skliar, Mikhail and Lee, Moonyong (2006). Analytical method of PID controller design for parallel cascade control. *Journal of Process Control 16*(8), 809–818.

Li, Bin and Guo, Song and Mao, Xuyao and Cai, Biaohua and Wang, Bo (2019). Design and simulation analysis of 6-dof electric platform. In *IOP Conference Series: Materials Science and Engineering*, pp. 52–59.

MIL-HDBK-799(AR) (1996). *Fire Control Systems*. Department of Defence Handbook.

Ogata, Katsuhiko (2001). *Modern Control Engineering* (4th ed.). USA: Prentice Hall PTR.

Rastegarpanah, Alireza and Saadat, Mozafar and Rakhodaei, Hamid (2013). Analysis and simulation of various stewart platform configurations for lower limb rehabilitation. In *Birmingham Environment for Academic Research Conference, University of Birmingham*, pp. 49–57.

TestOperationsProcedure 01-1-011A (2012). *Vehicle test facilities at Aberdeen test center and Yuma test center*. U.S. Army Test and Evaluation Command Test Operations Procedure.

Thoendel, Evzen (2011). Design and optimal control of a linear electromechanical actuator for motion platforms with six degrees of freedom. In S.-I. Ao, M. Amouzegar, and B. B. Rieger (Eds.), *Intelligent Automation and Systems Engineering*, Chapter 7, pp. 65–77. Springer New York.

Toz, Metin and Küçük, Serdar (2016). Development of derivation of inverse Jacobian matrices for 195 6-dof gsp mechanisms. *Turkish Journal of Electrical Engineering & Computer Sciences 24*(5), 4142–4153.

Vladimir Cech and Jiri Jevicky and Milan Jus (2014). Simulation of the vehicle passing the gun stabilization bump course of the aberdeen proving ground.

Proceedings of the 16th International Conference on Mechatronics - Mechatronika, 331–337.

Wendlandt, Jeff M and Sastry, S Shankar (1994). Design and control of a simplified stewart platform for endoscopy. In *Proceedings of IEEE Conference on Decision and Control*, pp. 357–362.

APPENDICES

APPENDIX A

HOST COMPUTER SOFTWARE DOCUMENTATION

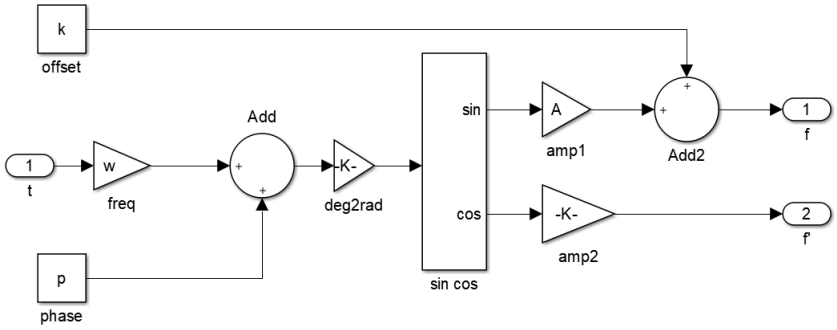

Figure A.1. Matlab Simulink blocks for sinusoidal trajectory generation for an axis

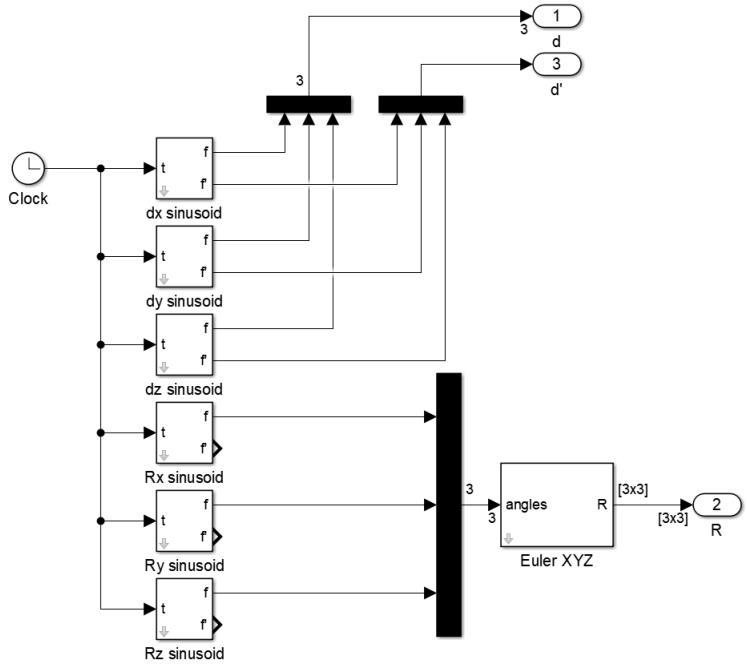

Figure A.2. Matlab Simulink blocks for sinusoidal trajectory generation for the top plate

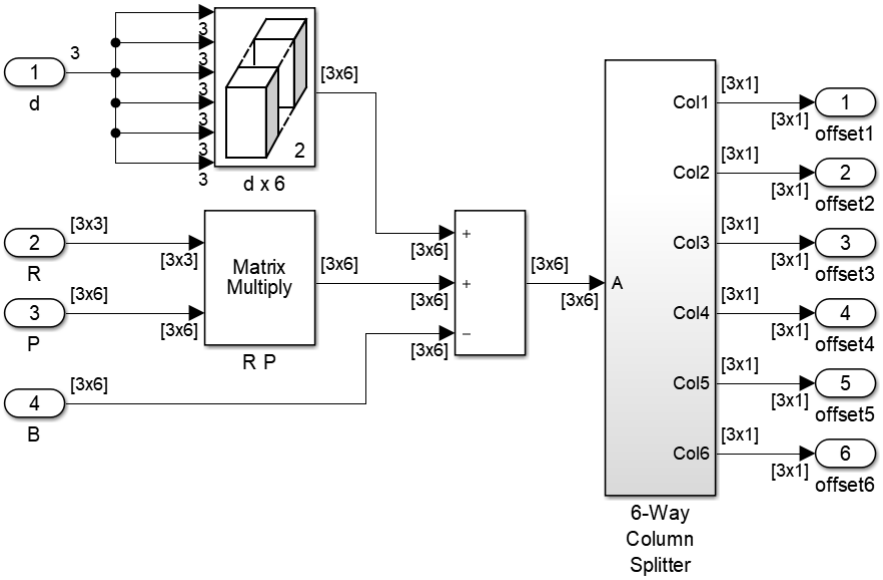

Figure A.3. Matlab Simulink blocks for inverse kinematics calculations

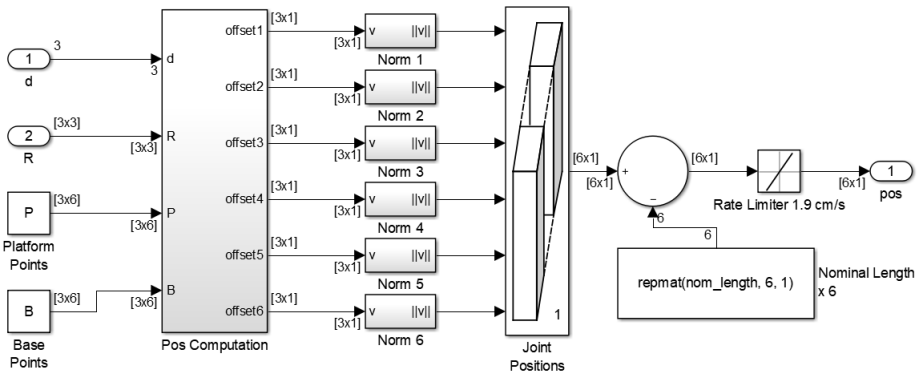

Figure A.4. Matlab Simulink blocks for calculation of the leg lengths

APPENDIX B

TARGET COMPUTER SOFTWARE DOCUMENTATION

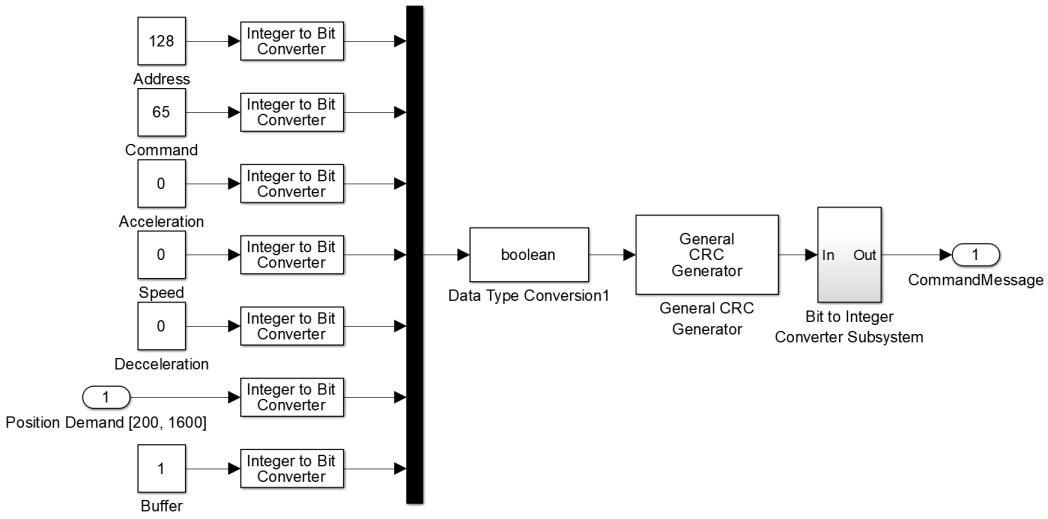

Figure B.1. Matlab Simulink blocks for command message generation

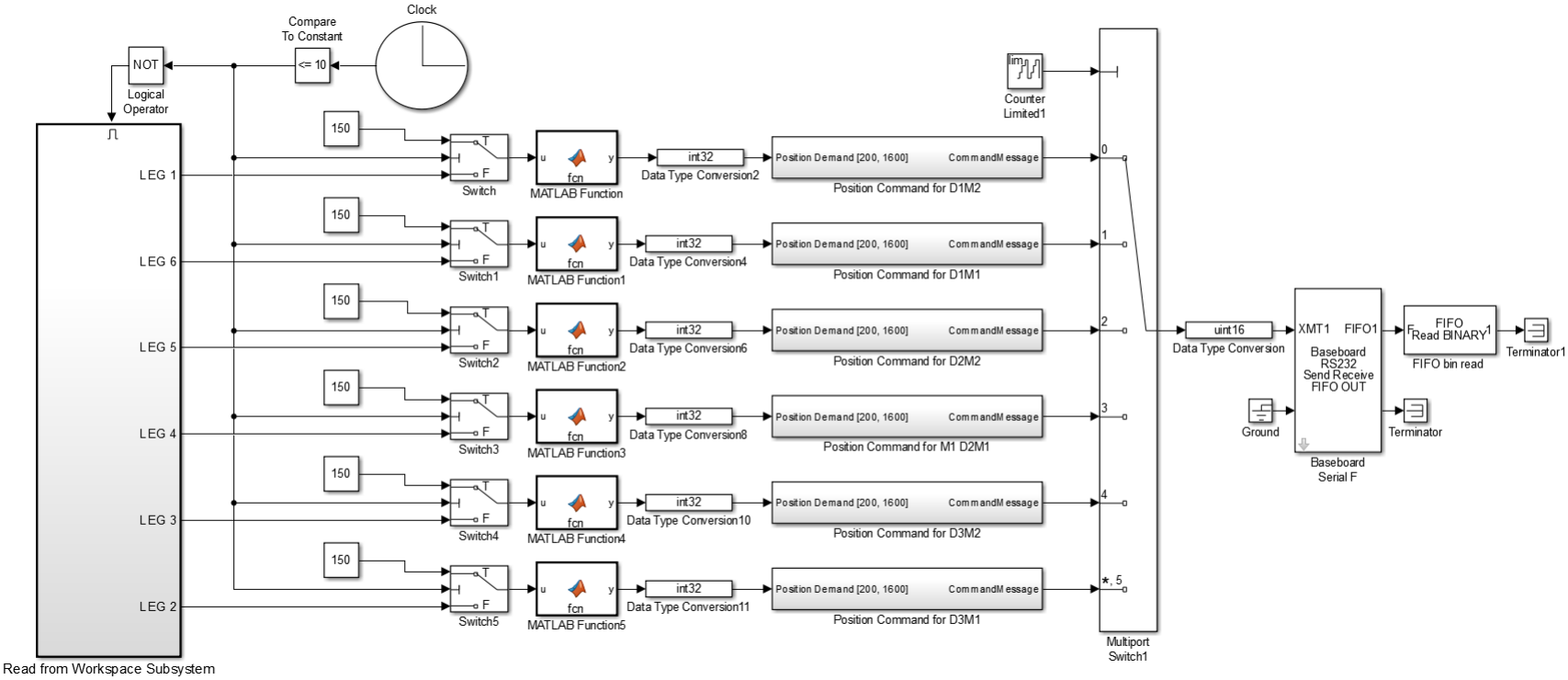

Figure B.2. Matlab Simulink blocks for communicating with the motor drivers