Functional Specification

**MIDDLESEX PLATFORM MEASURING SYSTEM**

LIST OF CONTENTS

[1. Abstract 2](#_Toc507509184)

[2. Background 2](#_Toc507509185)

[3. Introduction 4](#_Toc507509186)

[4. Functionalities 5](#_Toc507509187)

[4.1. Sensor Server 5](#_Toc507509188)

[4.2. Monitor 6](#_Toc507509189)

[5. Understanding and Viewing the Output Data 7](#_Toc507509190)

[6. Increasing Accuracy 8](#_Toc507509191)

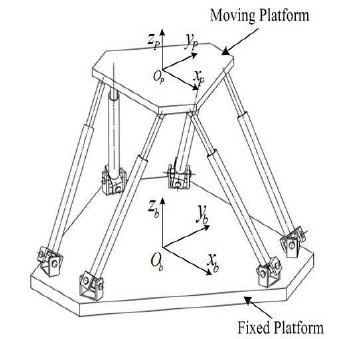
[7. Troubleshooting 8](#_Toc507509192)

# 1. Abstract

This document describes the functional specifications for the hardware and software system that monitors the real-time performance of the Middlesex 6DOF platform (The Chair) and also provides the future capability to enhance the accuracy by error correction of existing software that controls the platform. The specifications define what the system does, the user-system relationships and how the end user will interact and use the system. This document acts as guide for system architecture and development. It also provides troubleshooting details including possible cause of errors and their solutions for various functionalities of the system.

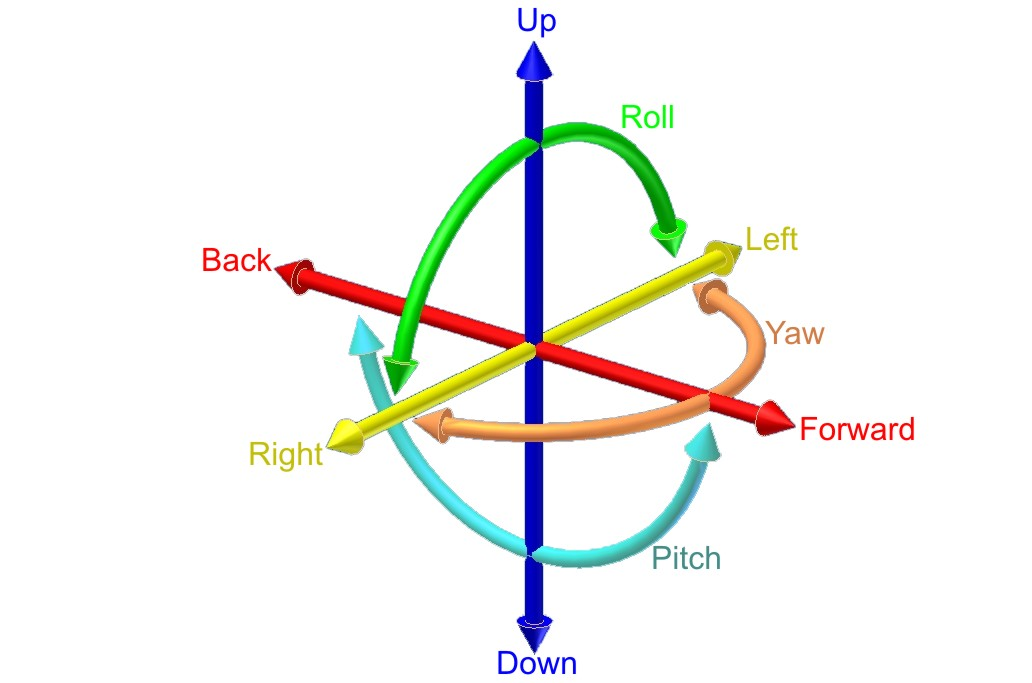
# 2. Background

The Chair is basically a 6DOF motion platform built using a robotic configuration called Stewart platform. It has two platforms, Fixed Platform and a Moving Platform which are attached together with six pneumatic actuators. The moving platform can move in six degrees of freedom using six prismatic actuators as shown in the picture below. Note that the MDX platform is inverted (the moving platform is below the fixed platform.



**Figure 1. Stewart Platform [1]**

The six degrees include three linear movements x, y, z and three rotations yaw, pitch, roll.



**Figure 2. Six Degrees of Freedom [2]**

The actual picture of the chair is shown below.

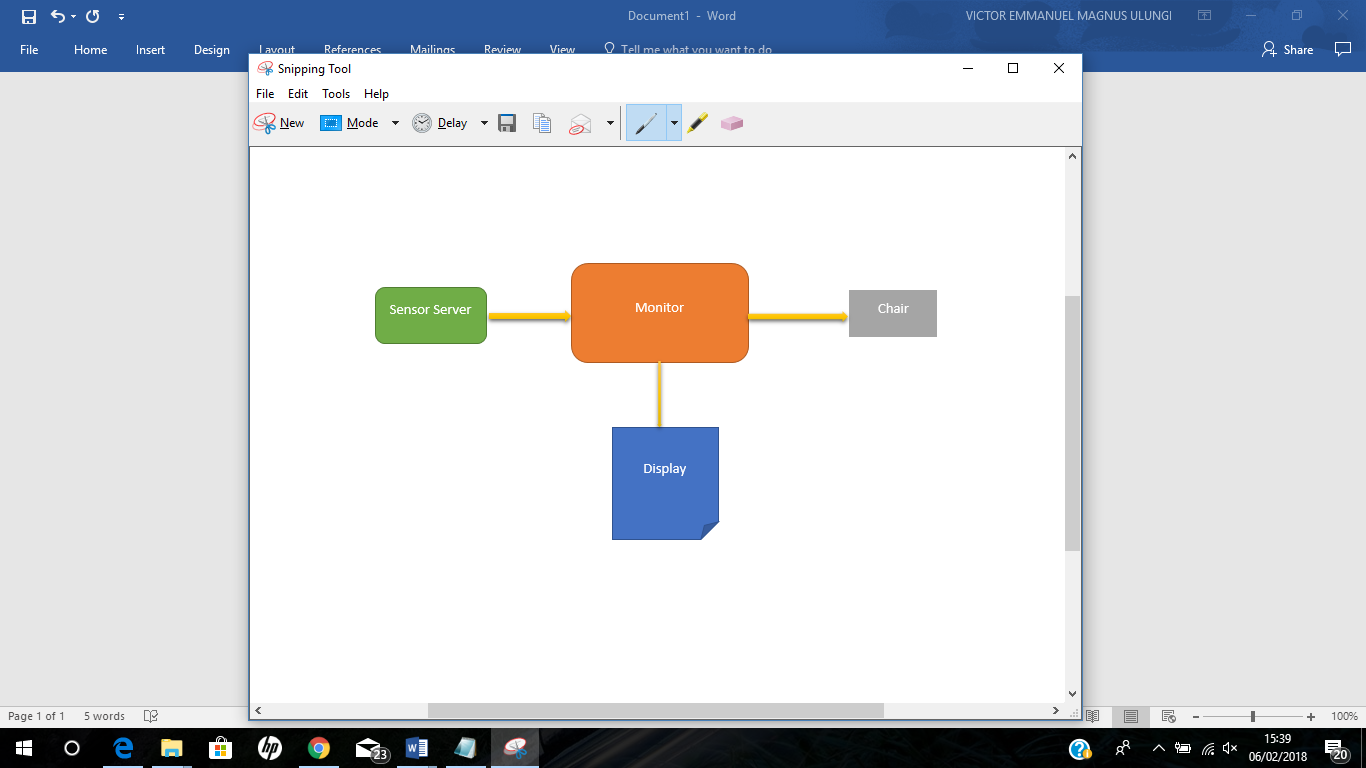


**Figure 3. The Chair**

The actuation of the chair is done by using 6 hydraulic ‘muscles’ as seen in the picture.

# 3. Introduction

The purpose of this project is to design and develop a complete system including hardware and software that provides error information on the real-time performance of the Middlesex 6DOF Platform. The system will measure the actual movements of the platform by using distance and angle sensors and will compare the measured lengths and angles with the desired movements that are set up by a monitor module. The Monitor will consists of the following functions which are, sending a message to the chair that will move the chair, receiving a message from the chair which consists of the actuator lengths, compares the received data from a sensor server and the chair and storing and displaying the results.



**Figure 4. Block Diagram**

The system can be subdivided into 2 main functional blocks:

1. **Sensor Server**

The Sensor Server consists of mainly two parts, that is the sensors that are used to read the actuator lengths and the row, pitch and yaw angles and the second part is sending this data to the monitor for accuracy testing.

1. **Monitor**

The Monitor will consist of several different functions which will be setting the desired values for the movements of the chair, receiving actuator lengths values from the chair, receiving actual data from the sensor server, compare the actual data with the commanded data and finally displaying and storing the error results.

The desired accuracy for static translational measurements is to the nearest millimetre and for angular measurements is to the nearest 1/2 degree. The orientation of the chair is updated with a delay of at least 2 seconds.

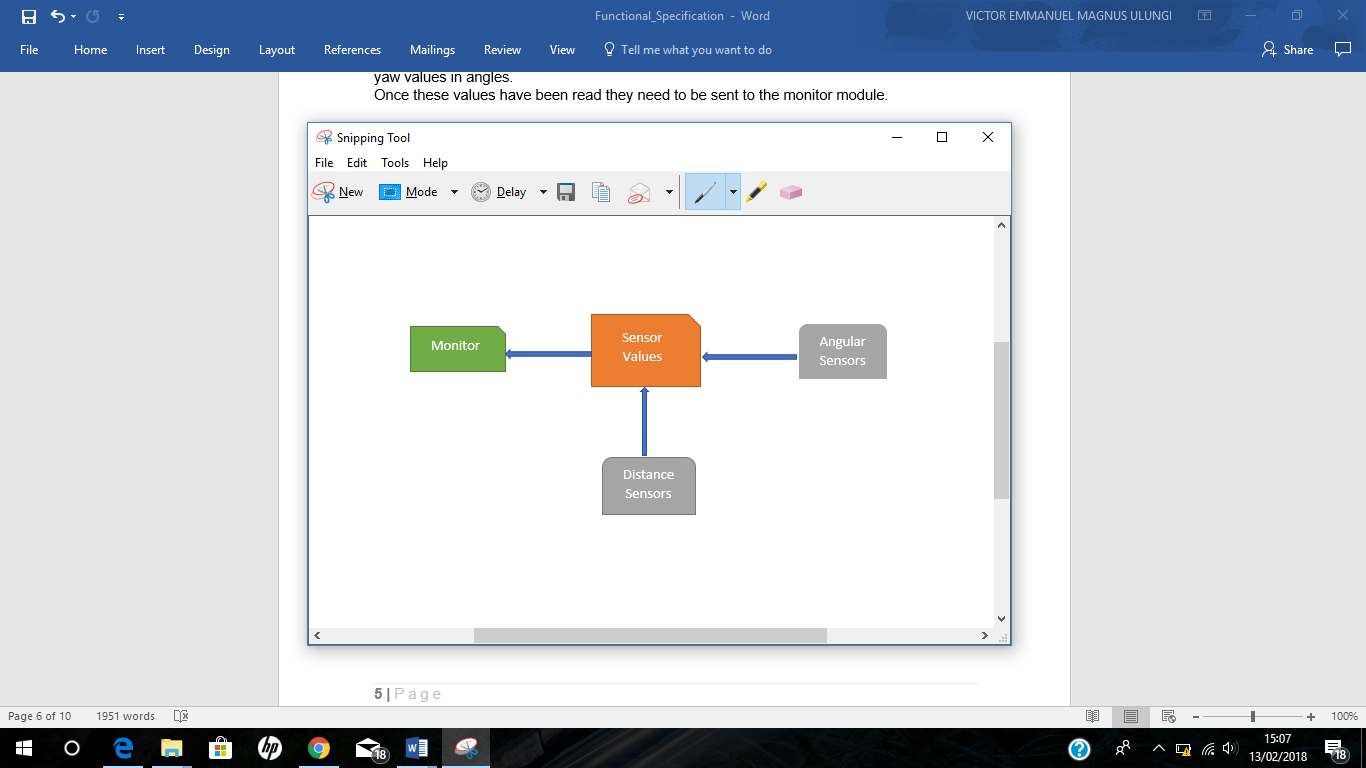
# 4. Functionalities

The two main modules are designed so that each module has a specific function that it is designed to perform. The functionalities of each modules are specified as follows:

## 4.1. Sensor Server

The capturing of the actual motion of the platform is handled by a module called Sensor Server. The Sensor Server will read the actual position and movement of the platform in real-time, at least every 90 msec using distance and angular sensors. The function of the sensors is to continuously measure the lengths of actuators and measure the roll, pitch and yaw values in angles.

Once these values have been read they need to be sent to the monitor module.



**Figure 5. Block Diagram of Sensor Server**

There is a total of 9 parameters that are be measured by the Sensor Server i.e.

6x Prismatic Arm Length (L1-L6) (See Table 5.1)

As there are six arms used in the platform, so the Sensor Server would measure and store the actual length of each arm individually in mm.

3x Angular Motion Roll, Pitch and Yaw (R, P, Y)

The angular motion of three axes would be stored in floating point as degrees.

The format for sending the actual sensor data is as follows:

H, xxxx, xxxx, xxxx, xxxx, xxxx, xxxx, yyyy, yyyy, yyyy

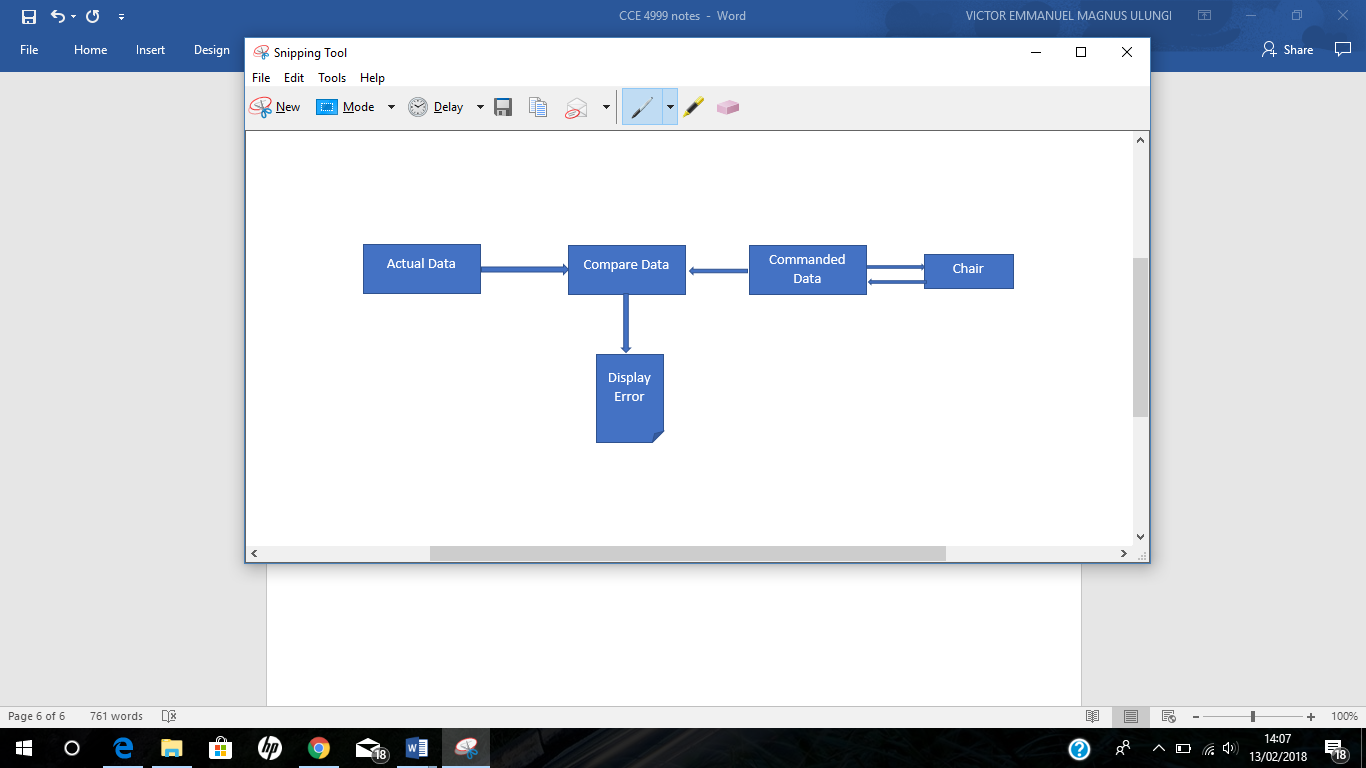
Where the xxxx represents ASCI digits for the actuator lengths and the yyyy represents floating point numbers for the angles.

This format will be sent to the monitor via a serial communication.

## 4.2. Monitor

The monitor is made up of the following functions which are; Sending a message with specified angles and height of the chair to the chair, receiving actuator lengths from the chair, receive actual sensor data from the sensor server, comparing the commanded data with the actual sensor data, displaying and storing the data.

What happens is that the monitor sends the desired values that the chair needs to move with then receives the actuator length values and saves both the angular values and the actuator length values with an extension of a CSV file. The monitor receives actual sensor values from the sensor server and compares these values with the ones that it saved in the CSV file to calculate the error, and finally display these two sets of values with the errors calculated.



**Figure 6. Block Diagram of the Monitor**

The commanded data function acts as a controller where it makes the following decision on how many steps the chair needs to move.

The movement for the angles should be in a way that the chair starts moving from a neutral point and it moves five steps from the neutral point to the maximum angle which is +20 degrees, once those five steps are completed the chair will move one step straight back to the neutral position, then it will move again from the neutral position five steps to the minimum angle which is -20 degrees and then it will move back one step straight from the minimum angle to the neutral point. This movement will be done for each type of the roll, pitch and yaw angles.

Although the commanded data can set what angle the chair needs to move with, this is not possible for the actuator lengths instead commanded data sets the height in which the chair needs to move and a message will be received from the chair with the actuator lengths which then can be used to compare with the sensor readings.

The movement for the height of the chair will take the same number of steps that were used for the angles in which the maximum length is +10cm and the minimum length is -10cm.

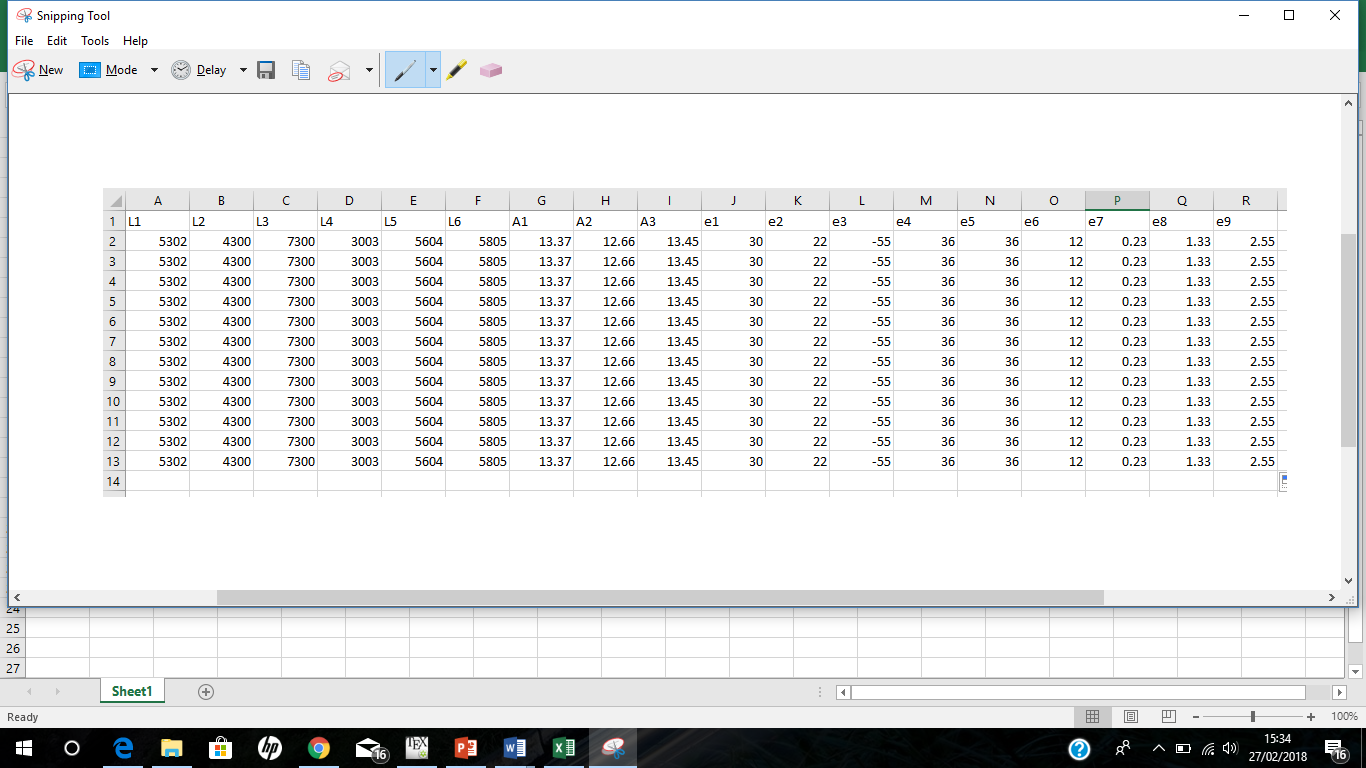
For each of these steps that the chair moves there is a delay of 2sec for each step.

The messages that are sent and received from the chair are found in the following repository (github.com/michaelmargolis/MDXeMotionV2).

# 5. Understanding and Viewing the Output Data

The result for the output data will consist of the commanded data together with the errors.

These commanded data will be 6 actuator lengths and 3 angles separated by commas, the errors will be the difference between the commanded data and the sensor data.



**Table 5.1**

The above table shows how the data will be displayed once the errors have been calculated.

The data stored by Monitor is in a simple comma separated values format and saved as an extension of a CSV file.

# 6. Increasing Accuracy

Once the accuracy of the system is measured, it can be increased by minimizing the errors. The errors can be categorized in the following two categories.

1. **Actuator Length Errors**

These errors indicate the difference between the commanded length and the actual length of the actuators. These kind of errors, can occur due to several reasons which are discussed below.

One possible reason for these errors is that the sensors, that are used to measure the actual length of the actuators, are not properly fixed in place. This could cause inconsistent values and hence large errors values. To minimize these errors, make sure that all the six sensors for measuring the actuator lengths are securely fixed in place.

Another reason for these kind of errors, could be overloading the system with more weight than it can handle. To avoid such a situation, verify the loading limit of the actuators via their specification document, and make sure to keep the load under this limit.

1. **Linear and Angular Motion Errors**

These errors occur when there is a difference between the commanded motion of the chair and the actual angular motion.

To minimize these kind of errors, make sure that the motion sensor is perfectly aligned and secured with the chair, otherwise it would generate persistent errors.

# 7. Troubleshooting

As with any other system errors may occur while running the system. These errors can be either communication problems or hardware issues. The following are different types of errors that may occur:

**Hardware Issues** – Some of the hardware may not respond and function as required, so to solve this issue try restating the device, if it does not help reprogram the device to make sure the device functions properly.

**Sensors not responding** – sometimes the sensors might not respond and thus not getting any reading due to the reason that they are not wired properly or faulty sensor, to solve this issue ensure all the wires are intact properly.

Sometimes only one of the sensor might not be getting any readings, how is one supposed to know which sensor is faulty, on the sensor server whenever there is no reading for a certain sensor a (0) will be displayed for that sensor and so once the monitor receives this value it will stop the program and display an error message indicating which sensor is faulty.

**Intermodule connectivity/Network issues** – There are several software modules that communicate with each other using sockets. If any of the modules is unable to communicate with other modules verify the port number for each module. If the problem still exists try closing all the programs and restart the PC.

**Wiring Issues** - As the platform moves with high acceleration and makes jerks so it is likely that the wiring of the sensor or the controller goes loose. To avoid such a situation, keeps the wires and the modules tightly secured with their location.

**Power Supply Issues** - Using long wires to provide power can result in voltage drop that is much significant with low voltage systems. If the power lights of the modules start to get dim during operation, it's time to use thicker wires for supplying power.