Design and Development of a "Two- Axis Leveling Platform"

by

Anwarudin bin Saidu Mohamed

Dissertation submitted in partial fulfillment of
the requirements for the Degree of
Bachelor of Engineering (Hons)
(Mechanical Engineering)

JUNE 2004

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750, Tronoh
Perak Darul Ridzuan.

CERTIFICATION OF APPROVAL

Design and Development of a "Two- Axis Leveling Platform"

by

Anwarudin bin Saidu Mohamed

A project dissertation submitted to the
Mechanical Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF ENGINEERING (Hons)
(MECHANICAL ENGINEERING)

Approved by,

(Mr. Mark Ovinis)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
JUNE 2004

CERTIFICATE OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the reference and acknowledgements, and that the original work contained herein have not been taken or done by unspecified sources or persons.

ANWARUDIN BIN SAIDU MOHAMED

ABSTRACT

Electronic drives such as stepper motor; either rotary or linear, have found many useful application and are regarded as the major components in automation. Many applications evolved rapidly with integrating components such as sensors, controllers, actuators, drives and switches. Variety of systems that performs similar applications could be developed using these combinations. Simplicity, better control and cost are the main criteria in selecting the best configuration. Therefore a good grasp of knowledge and technology on these elements will be able to enhance the development of the automation industry. One of such applications in automation filed is a stabilizing or leveling system.

A two-axis –leveling platform is a system that is capable of leveling a platform although the base is subjected to uneven motion. The word two-axis in this context represents two drives (linear stepper motor) that are used to control the platform's motion compare to some other stabilizing systems that use many drives that eventually lead to costly design. The project is aimed at designing and developing a physical model that is capable of demonstrating the idea of leveling a platform when the base is subjected to uneven motion. The model consists of the basic structure of the platform, control device such as controller board to control the motor, computer as a processor, serial port as an I/O card, linear actuators and control program developed using LabVIEW. Thus, the project consists of a system that has both hardware (model) and software (controller program) components. There are some initiatives taken to keep the overall cost within the budget since the main constraint in this project is the cost. Under those constraints, author has found ways to utilize both parallel and serial COM port as data acquisition medium.

The motor selection was guided by the physical constrains of the system as well as equations to calculate the required torque, allowable load and speed. The expected result of this project is to deliver and communicate the idea of leveling a platform in many applications. Some of the main advantages of using the linear actuators are its high durability and flexibility of programming. In between the accuracy in term of tilt-angle is an impressive result of the system. Stabilizing a solid surface has found many useful applications in automotive, military, micro-inspection and measurement field. This project could be a springboard to explore the new ways in leveling a platform which carry vast practical applications.

ACKNOWLEDGEMENT

The author would like to take this opportunity to extend his utmost gratitude to the people who have made this project to the path of among the successful projects. Ultimate gratitude extended to God for his blessings to complete the project. Sincere gratitude is due to the author's supervisor, **Mr. Mark Ovinis**, Lecturer in Mechanical Engineering Department for his guidance and support. Being under his supervision has allowed the author to explore elements of project management, research activities and independent decision-making in this project.

Secondly, the great appreciation directed to author's Co-Supervisor **Dr. S.I Gilani,** Lecturer in Mechanical Engineering Department for providing sufficient information on this project. He had been the major contributor in technical information regarding the drives and the overall mechanism of the system. The author also would like to thank **Dr. Fakhruldin Mohd Hashim**- Program head of Mechanical Engineering Department, **Ted Scheumann** -Sales Engineer from motionshop.com, USA, for providing useful information on the elements of the system (i.e stepper motor, tilt –angle detector, I/O card, controller card) and **Ir. Idris bin Ismail** for useful comments on the project.

Special thanks extended to **Mr. Rosdiazli** and **Mr. Bashir**, Lecturer from Electrical & Electronic department for their willingness to help author to test the drives as well as advice on proper selection of these drives. Their contribution is highly appreciated. Gratitude is also extended to **Mr. Haris** Lecturer from Electrical & Electronic department who provide sufficient reference on the LabVIEW software.

Finally, but definitely not the least, the message of thanks goes to **Mr.Teh Eng Soon** for his willingness to share some ideas and methodology of developing LabVIEW controller, **Mr. Ahmad Rafeek Ibrahim,** Final Year student, UIA, Malaysia for kindly share some ideas related to this project and **Mr. Jonathan** form PSI Pneumatic Control Sdn Bhd to aid in finding the linear stepper motors. Regards are also expressed to the author's parents for their constant blessings and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABBREVIATIONS AND NOMENCLATURES

Bounding box	The diagonally opposite corners of a 3D box that encloses the solid.
Controller	Circuits that control the stepping motor's angular distance of rotation and rotation speed by pulse signals according to settings.
Centroid	A 3D point that is the center of mass for solids.
CCW	Rotation in Counter Clockwise direction.
CW	Rotation in Clockwise direction.
FYP	Final Year project.
Holding Torque	Sometime called static torque, it specifies the maximum external force/torque that can be applied to a stopped, energized motor without causing the rotor to rotate continuously.
LabVIEW	Graphical programming language that uses icons instead of lines of text to create applications.
Ramping	The motion of the motor at the beginning and ending of the rotation sent.
Open Loop	Refer to a motion control system whereby no external senses are used to provide position or velocity correction signals.
Pulse rate	The frequency of the step pulses applied to a motor driver. The pulse rate multiplied by the resolution of the motor (in steps per revolution) yields the rotational speed in revolutions.
Rated torque	The torque producing capacity of a motor at a given speed. This is the maximum torque the motor can deliver to a load and is usually specified with a torque/speed curve.
Resolution	The smallest positioning increment that can be achieved. Frequently defined as the number of steps required for a motor's shaft to rotate one complete revolution.
RS-232c	A data communication standard that encodes a string of information on a single line in a time sequence format. It specifies standard requirements so that different manufacturer devices are compatible.
Static torque	The maximum torque available at zero speed.
Stepper motor	Specialized motors that create incremental steps of motion rather than a smooth, unbroken rotation. Its performance is measured by step angle, max. rotational speed, resolution. They are classified by either VR (Variable reluctance) stepper motor or PM (Permanent magnet) stepper motor.
Step angle	The angle the shaft rotates upon receipt of a single step command.
Torque	Force tending to produce rotation
VISA	A virtual instrumentation function that can write data to the device synchronously or asynchronously.

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND OF THE STUDY

"Two-axis leveling platform" also known as stabilizing platform is a system that aimed to stabilize a platform which is attached to a base that is subjected to uneven motion. The term two axis is used because the system uses two axis's to describe the motion. It is controlled by two actuators to coordinate the orientation of the platform. Besides leveling a surface, the system can also be used for other application which requires precise angle coordination of the object. Basically the main components of the system are the platform structure, linear actuators (linear stepper motor), controller board, I/O card and the software (LabVIEW, motion controller). The software will process the retrieved data and relate the angle (pitch and roll) to required actuation. *(Refer to Section 2.7 for analysis on calculation of the pitch and roll of the platform*). Based on the information provided in the program, the required number of pulses to extend/retract the actuators for a specified angle will be identified. This information will be sent to controller board through serial port. The controller will generate the amount of pulses as requested by the program in pulse rate (frequency). *Refer to Figure 20 for algorithm and process flow.* Therefore the actuators will move independently as they are assigned in pitch and roll direction.

The detail analysis on the relationship between the tilted angles, radius of rotation, pulse rate, half/full step mode, required retraction or extension of the actuators and time required for the leveling action is well demonstrated in *section 2.7.* In addition, tilt-angle sensor is could be placed on the platform (top) to detect the deflection angle when the base is tilted. This element is capable of detecting the orientation of the platform at any instant in dual axis; pitch, roll and required signal can be sent to processor (PC) to be acquired by LabVIEW and execute the algorithm accordingly. Thus a closed loop algorithm can be incorporated. But, this report will not cover the integration of tilt-angle sensor as its addition raised the cost above the allowable limit.

Therefore the addition of this sensor to fully automate the leveling process will be the continuation for this project. The methodology of integrating the tilt-angle sensor will be briefly explained in *section 3.7.2.2* since the software is well prepared with sufficient flexibility for modification.

Actuator 1: pitch angle compensation

Actuator 2: roll angle compensation

**Refer to *Figure 9, pg.18* for 3D drawing of the platform structure.

The scope of this project is limited to the demonstration of the open-loop control on the actuators to control the angle of the platform in both pitch and roll axis.

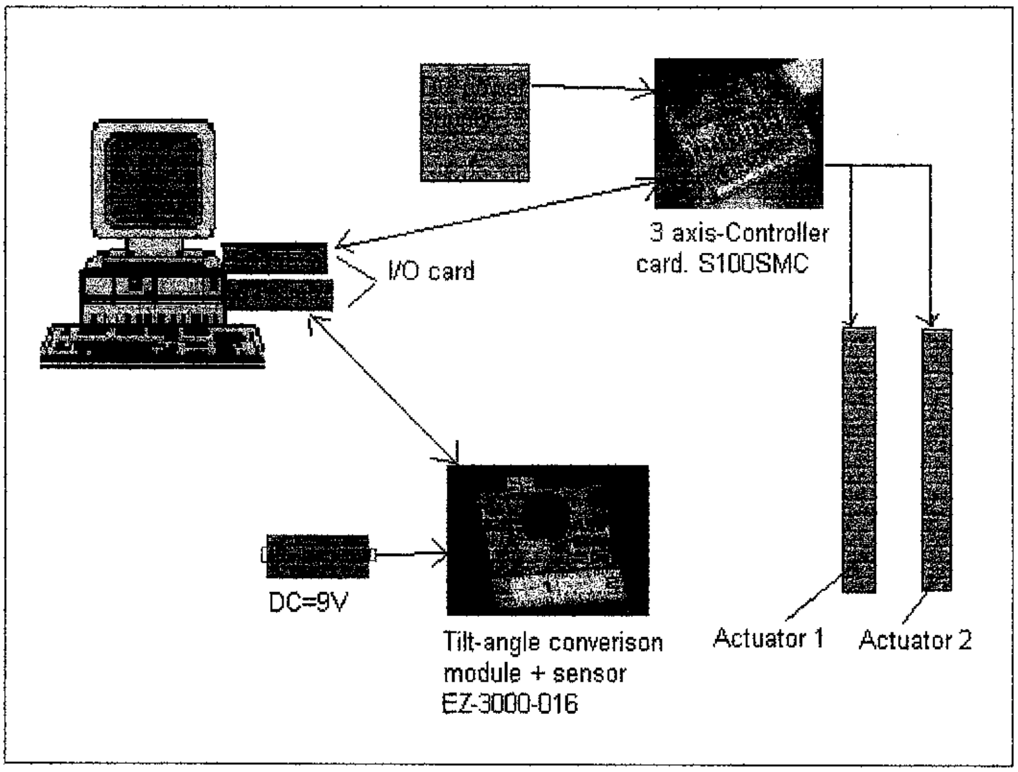


Figure 1: The solution for system configuration.

A lot of research has been done on angle coordination system design and development and there are many solutions have been developed. Most of the designs are using the electric actuators or pneumatic drives to control the angle of the platform. The electric actuator is feasible to meet high mechanical accuracy for leveling requirements **[1]**. But those designs (system) are very expensive due to the need for many actuators and independent controllers that are very costly.

1.2 PROBLEM STATEMENT

1.2.1 Problem identification

This project as explained before is actually a problem solving-driven design project. The problem is on how to maintain the platform horizontal even-though the base experiences uneven motion. Refer to *Appendix I* for physical structure of the model. The main scope of the project will be finding the best way of constructing a stable structure (model) that is suitable for leveling concept demonstration and interface all the components of the system in the most efficient manner using proper algorithm. Both hardware (model) and software (controller) which are the main constituents of the system are required to be constructed to demonstrate the leveling concept. The available stabilizing platform design is very costly due to a lot of actuators and independent controllers. Therefore, this project is capable of producing a low-cost design that could be used for educational and demonstration purposes. Commercializing this design will need a lot improvement in software, controller and other components to increase the overall performance of the system. Besides, further development in this project will be able to prepare a springboard for a solid technology ownership in this field considering vast real-life application.

1.2.2 Significance of the project

This project have strong relevancy in mechanical engineering field especially in engineering design and development. This design project has vast applications in real-life to solve many problems. Besides, the project would be a good hands-on practice that would enhance the innovativeness, problem solving skills, engineering design skills project management skills, communication skills and medium on improving the technical knowledge. Most importantly this project will enable the application of the knowledge gained throughout the academic years in UTP. This project also prepares a challenging environment to utilize available resources in limited time frame to construct the effective and efficient solution independently and present the findings through minimum guidance and supervision.

1.3 OBJECTIVE AND SCOPE OF THE STUDY

1.3.1 Objective of the project

The objective of the project is to design and develop a system that is capable of demonstrating the idea of leveling a platform. The project requires building physical model (hardware) and software to interface the linear actuators. The actuation and retraction of the motor will be controlled by using the controller software (LabVIEW) and suitable algorithm to level the platform that is tilted due to base uneven motion. Besides, the project is also expected to prepare the preliminary skeleton for future expansion (i.e integrating the tilt-angle sensor). Following are the list of sub objectives that are specified to guide the author throughout the project.

1. To understand the principles in product design.
2. To develop an understanding in the software development (LabVIEW).
3. To understand the underlying steps in designing.
4. To learn the methodology of learning and applying the learned knowledge.

1.3.2 Scope of the project

Scope of the project has been narrowed down for better focus on the project. The project started with identification of the elements of the system, purchase of elements considering the limited budget; thus simplifying design and finding the alternative to use common material and tools, construct the model, find a way to communicate with actuators using a PC and finally develop controller software to demonstrate the leveling concept. Thus the scope drawn is until the ability to prove the concept of leveling without tilt-angle sensor in place, which means that the actuators will be in open-loop control.

1.3.3 Relevancy of the project

This system must be capable to demonstrate the method of maintaining the platform (top part of the table) horizontal although the base is subjected to uneven motion by interfacing with PC. There is lot of other design that utilizes similar concepts. For example the VAM (Vertical Aimed Motion) control, vertical pendulum and so on. Projects with different concept utilize similar approach as a means of controlling the devices such as motors, solenoids, actuators and etc. The ability to interface the components and control the drives or actuators would be a great advantage in exploring other concepts or designs. This project would be an opportunity to develop the skills in control, engineering design and fabrication. The concept of stabilizing a platform is actually a field that can be considered for enormous improvements. Although the concept seems to be expensive, but the application which the design intent to is the advance high-technological field which the precise control is a requirement.

1.3.4 Feasibility of the project within the time frame

Submission of this report indicates the completion of the project that could be compared with the objective and scope set before. The author has utilized the project flow chart in *Appendix A* as the prime reference to progress in the project. Milestones as indicated in flow chart were strictly followed. Although a lot of difficulties and delays incurred due to difficulty in getting the suitable actuator and facility , tools for fabrication, the author had utilize the time to perform some other analysis such as preparation of references, survey and study the software (LabVIEW) to be used for developing the control program. Beside the progress of the project, checkpoints of review and complete methodology was development as complete plan as to comply with the time frame of the project.

CHAPTER 2

LITERATURE REVIEW AND THEORY

2.1 APPLICATIONS USING SIMILAR CONCEPT

There is a lot of interesting designs and creative inventions are developed towards achieving the ultimate aim of leveling a predefined surface. Such inventions have the similar purpose as this project which utilizes the concept of stabilizing/leveling a platform.

1. VAM (Vertical Aimed Motion)
2. Kite-Ariel Photography - to capture the images from flying object.
3. Electric powered motion platform – to maintaining horizontal position of various objects and equipment as well as for positioning objects accurately aimed for sensitive inspection machines or usage under the angular fluctuation of the platform.
4. Missile base leveling- for precision targeting when it is to be launched from a ship or tanker
5. Self/controlled-stabilizing platform- for image capturing via camera mounted on the platform.
6. Stabilizing ship borne antenna – broadcasting applications.

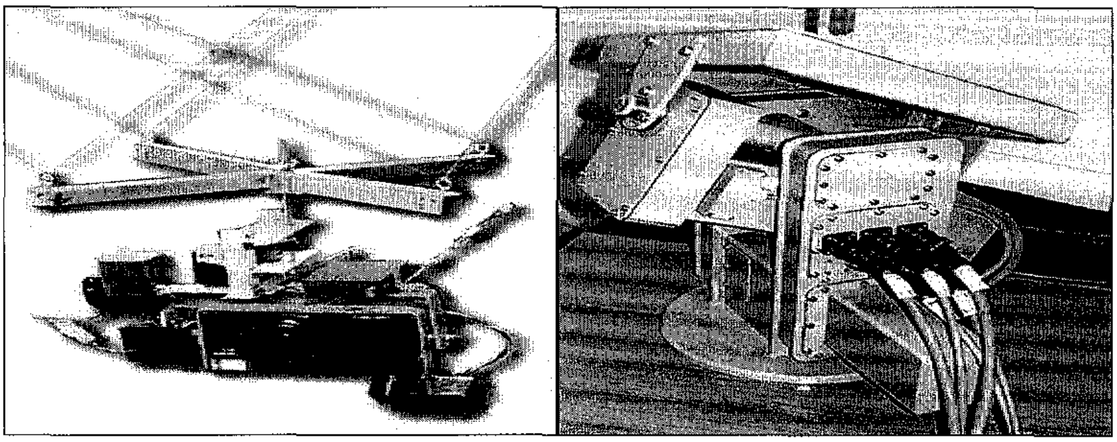


Figure 2: Similar design concepts of the leveling platform

Stabilizing a ship borne antenna is well demonstrated by Sheng Xiaowen from *Beijing Institute of Electronic System Engineering, Beijing, China* in his thesis **[2]**. According to Shang, inertial measurement elements were used to sense the ship rock. Thus the ship rock angle was compensated by reverse rotation to isolate the ship rock and stabilize the antenna direction. The platform has two types: an exclusive platform, which sets up a two-axis servo mechanism parallel with the pitching axis and the rolling axis of the ship. The mechanism senses the ship rock by high precision and high sensitivity inertial measurement elements and compensates the ship rock angle by the reverse rotation. According to Sheng, the principle of this type of platform is very simple and it solves the problems of the quick response and stabilizing accuracy very well, but its volume and mass have no obvious improvement and its cost is very high, so it is seldom used practically.

Another example was demonstrated by Matt Rizzo, Brian Clark, Brian Yurconis, Jelena Nicolic from Oakland University. They have used the stabilizing principle to construct their "self-stabilizing platform" to maintain vertical positioning of the camera. **[3]**

2.2 DESIGN PROCESS

Some of the design procedure has been adopted to ease the design methodology and problem solving **[4]**.

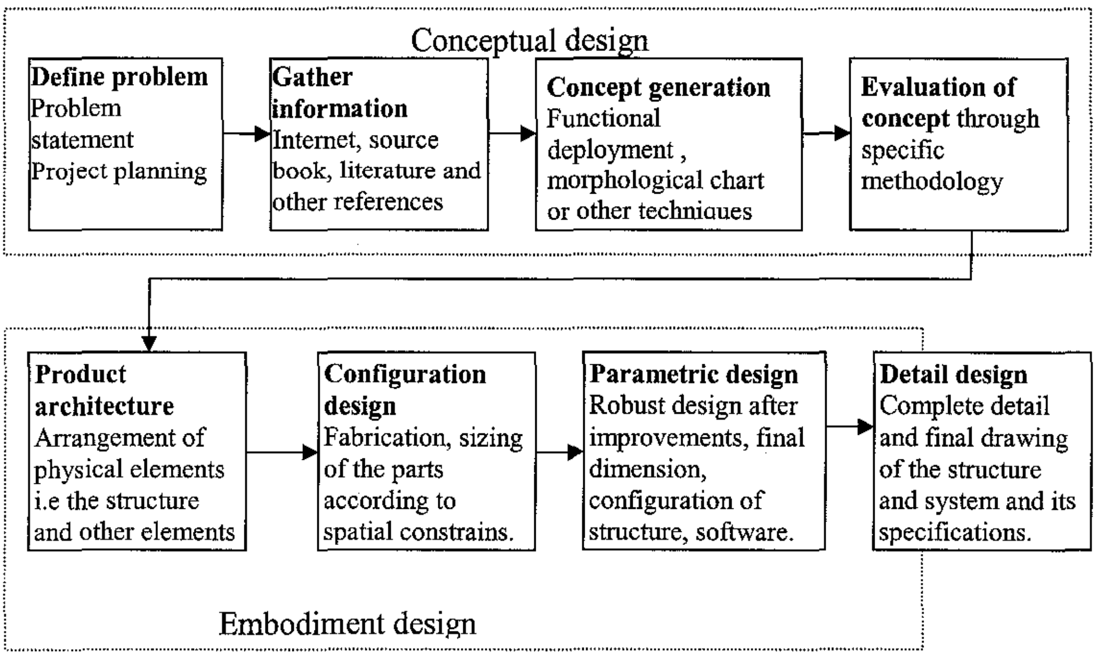


Figure 3: Engineering design procedure adopted in undertaking the project.

According to Geoerge E. Dieter [4] Pugh concept selection and weighted matrix evaluation are among the effective methods to select the most promising design concept. These concepts are later will be used to select the best design among the 5 alternatives (A, B, C, D and E). Please refer to sketches in *Appendix B*.

2.3 DESIGN OF THE STRUCTURE

The structure of the beam can be constructed with using combination of different materials (i.e wood, solid steels beams, aluminum beams). But the usage of aluminum beams can enhance the aesthetics property of the structure while providing sufficient strength with less weight. From the research, aluminum alpha iron beams suits the best due to factors discussed in the following paragraph.

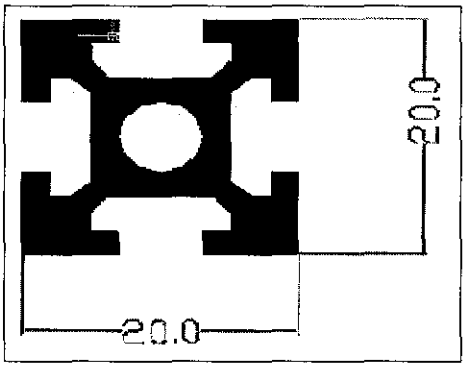


Figure 4: Structural beam cross section view.

This alpha-iron beam is made of aluminum alloy and well known for its ultimate strength with minimum material. These types of beams have remarkable ability to sustain impact and compression. Besides, it is corrosion-resistant and therefore will have good aesthetics appeal for a longer period. Therefore the structure of two-axis leveling platform can be considered to be constructed using alpha-iron beam with cross section 20mm X 20mm as shown in *Figure 4*.

2.4 DRIVE SYSTEM

From the beginning of the project, the author is engaged in finding a suitable drives that can control the rotation of the lower and upper platform (*refer to Figure 9, pg.18 and Appendix H for 3D view of the structure*). The main criteria for the drives selection were:

1. Does not induce "jerky" motion.
2. Open-loop control

3. Cost of the drive
4. Load handling capacity (Holding torque, load torque, acceleration torque)
5. Fast action.
6. The precision of actuation for tilting minimum angle.(resolution)
7. Availability of the component.

RS catalogue **[5]** was used for cost comparison between the alternatives, their respective load handling capability and the speed. Initially the following alternatives were found for the actuators.

2.4.1 Alternatives for the driver.

Table 1: Alternatives for the driver

Drive system	Description
DC stepper motor	The control will be in term of number pulses sent through I/O card, serial or parallel port. Gear reduction is essential to increase the torque. Risk of missing steps and jerky motion.
DC servo motor	Needs some feedback controller (optical encoder) to recognize the position of the shaft. **[6]** The most precise. Gear reduction is essential to increase the torque.
Rotary actuator	The control will be in term of number pulses send through I/O card, serial or parallel port. Gear reduction is essential to increase the torque, and sometime the geared rotary actuators are available in the market as well. Sometime the steppers are needed to create the discrete steps. Jerky motion due to backlash and long radius of gyration.
Linear type actuator (drive: stepper motor)	The easiest way, this is directly control the motion with controller communication. The control will be in term of number pulses sent through I/O card, serial or parallel port.
Linear type actuator (drive: DC motor)	The mean of control for this type, will be the voltmeter to calibrate some discrete steps instead of full throttle per actuation.
Pneumatic actuator.	The actuation is controlled by the air-pressure. Therefore sometime pressure will lead to inaccurate actuation. Mostly very few discrete steps of actuation, which is governed by the timer, counter or stopper. **[6]**

According to Parker Automation **[7]**, stepper motors are among the best choice for open-loop control with easy angle and speed control, high torque and good response when coupled with a gear box, high resolution and positioning precision. They suggest the linear actuator embedded with stepper motor; ball screw driven configuration for this application. Besides, Ted motionshop.com **[8]** also suggested this solution for precision positioning application.

2.4.1.2 Advantages of Linear Actuator with stepper motor drive (quoted from parker automation).

Linear Actuator with stepper motors is an excellent solution for positioning applications that require rapid acceleration and high-speed motion with low mass payloads. Mechanical simplicity and precise open-loop operation are additional features of the stepping linear motor systems. The Linear Actuator is not subject to the same linear velocity and acceleration limitations inherent in systems converting rotary to linear motion. For example, in a lead screw system, the inertia of the lead screw frequently exceeds the inertia of the load. Attempting a high-speed move with a low mass payload results in the majority of the motor torque applied to overcoming the lead screw inertia. As the length of the screw increases, the inertia of the screw increases and the maximum critical speed is reduced. With linear actuator, all the force generated by the motor is efficiently applied directly to the load, and length has no effect on system inertia.

2.4.1.3 Additional benefits of Linear Actuator with stepper motor drive.

Table 2: Some of the additional benefits of using linear stepping motor.

High Throughput	The motors are capable of speeds to 2 m/s and the low mass force allows high acceleration.
Mechanical Simplicity	The need for lead screws or belts and pulleys is eliminated. The mechanical design is pre-engineered.
High Reliability	Fewer moving parts and a friction-less air bearing design results in a longer, maintenance-free life.
Long Travel	Length of travel is limited only by the length of the platen; increasing length causes no degradation in performance.
Precise Open-Loop Operation	Unidirectional repeatability to 2.5 microns without the added expense of feedback devices.
Small Work Envelope	A linear motor is usually smaller in all three dimensions than comparable systems where rotary motion is converted to linear
Multiple Motion	More than one force can operate on the same plane with overlapping trajectories.

Linear Actuator with stepper motor drive is mostly voltage controlled and are capable of controlled in term of discrete steps (pulses). The common parameters that govern the performance of such devices are maximum working torque, resolution (mm/rev), number of pulses/rev, holding torque, step angle tolerance, max. radial force, max. Axial force and load torque. After all these considerations, the author decided to use the linear actuator with rod bearing configuration. But the motor with required physical constrains (i.e mounting, size, stroke) were hardly available.

Beside the linear actuator mentioned earlier, the author found that LVDT (Linear velocity displacement transformer) might suit the requirement of this project which due to long stroke and fast linear motion. But, this device is basically used to measure the distance when the rod is displaced. Where else the application needs the reverse concept. But according to I.Boldea, Syed A. Nasar [1] LVDT could be used in reverse concept. *"Just as rotary electromagnetic machines do, LEAs (Linear electric actuators) and LEGs(Linear electric generators) share the blessing of reversibility: that is, the same machine can act as an actuator or as a generator."*

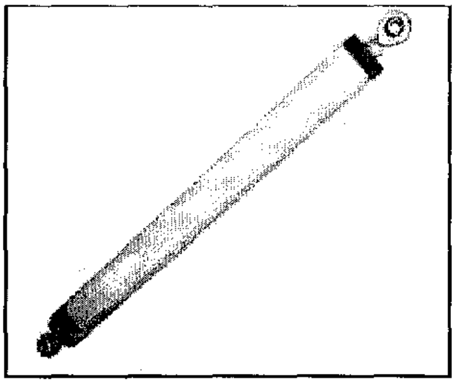


Figure 5: A type of LVDT which suits the physical requirements.

This statement suggest that, LVDT (linear velocity displacement transformer) can be used as actuator as reverse to displacement measuring device. The principle of LVDT is that, the voltage induced respect to the amount of displacement for measuring purposes. Where else, in the reverse manner, the displacement can be controlled by the voltage for positioning purposes.

2.4.1.4 The specification of the selected linear actuator

The linear actuator obtained was taken from a CNC machine without catalogs or any information booklet stating the characteristic, performance and properties of the motor. Therefore, the author sleeked for the aid from manufacturer to get relevant information. The linear actuator obtain was attached with OS 22A Compumotor stepper that which is a high torque type motor. But due to limited current load that could be supplied by the controller board, using a pair of those motor was impossible. Therefore one different stepper motor; MINEABBA have to be used to cater within the current limit. The actuator is extremely flexible whereby any stepper motor with sufficient shaft length and suitable size (FEMA 23) can be used since the actuator was driven by the timing belt and the coupler can be taken out of the shaft. *Please refer to Appendix I for the picture of the actuator.* The linear actuator works with stepper motor (PM-permanent magnet type) of 1.8 ^{o} stepping angle embedded with 2 magnetic sensors to adjust the stroke. Please refer to *Appendix L* for dimensions of the motor and *Table 8* for performance specification of the actuators.

The summary of the information obtained on the motor (OS 22A – PARKER Compumotor)

Brand : Parker Compumotor (Automation Actuator), USA
Model : OEM-57-83 (Bipolar)
$I_{dc/\ rms}$: 3.4 A
$V_{dc/rms}$: 50 V
Step angle: 1.8° deg
Power : 91 W
ω : 1500 RPM

The summary of the information obtained on the motor (MINEABBA MINIANGLE STEPPER)

Brand : ASTROSYN – MINIANGLE STEPPER
Model : 23LM-C312-P2
$I_{dc/\ rms}$: 1.2 A
$V_{dc/rms}$: 5.1 V
Step angle : 1.8° deg
ω : 800 RPM

2.4.1.5 Limit switches

In *Appendix O* there are three positions shown on the left side of the board labeled L0, L1 and L2. These positions are used on the terminal block are used to connect limit switches inputs to the S100MC. The type of limit switches in the motor is magnetic type that emits lights and send appropriate output signal when the magnetic stripe on the shaft crosses the limit switch. Therefore, the input received can be used for to stop the motor or probably change the frequency of the pulses and so on. Preferably in our case the output from the limit switch can be used to stop the motor.

2.4.1.6 Current requirement

Considering the fact that the maximum allowable current through controller board is 5 A, it was impossible to use 2 stepper motor from PARKER, Compumotor which requires 3.4 A per motor. Therefore the total required current will be 6.8 A which is above the allowable limit. Therefore the author decide to use stepper motor from PARKER, Compumotor for the bottom stage which eventually exposed to higher load compare to the actuator attached to the top platform. Therefore, a stepper motor from MINEBEA (ASTROSYN) was used with only required current of 1.2 A. Therefore the total current requirement at anytime is 4.4A which is below the limit.

2.5 CONTROLLER BOARD

The controller board to control the actuation of the motor is readily available with reasonable prize in http://*www.steppercontrol.com* **[9]**. Basically there were two options (model S100SMC and A100SMC) available. Both suit the application in the sense that A100SMC provide two-axis control, whereby S100SMC with 3-axis control. The expandability of the S100SMC was better and the performance specification was far more impressive then A100SMC. Therefore S100SMC was considered for the application. This board can be interfaced to computer through serial port. Refer to *Appendix O* for connection diagram to interface motors to the controller board.

S100SMC Features:

- 3-Axis Control
- Independent speed, direction, and number of steps for each axis.
- Speeds in excess of 800 steps per second (depends on motor and power supply voltage)
- Execute discrete numbers of steps or continuous rotation.

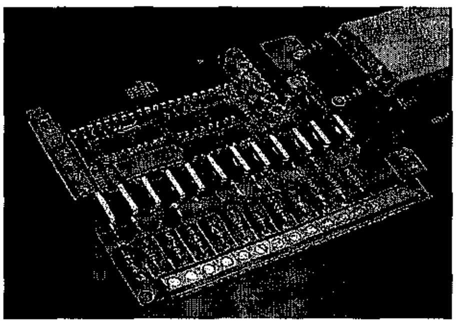


Figure 6: S100SMC controller board.

From the research, author found that there were a lot of limitations imposed by the controller board in determining the performance of the whole system. For example, the maximum deliverable current was only 5 A and voltage of 50 V. Besides, there is also limitation on the supply frequency (pulses/sec) that downgraded the actual speed of the motors. More detail discussion regarding this matter is given in *Table 8.*

2.5.1 Controller board protocol

In order to interface the motors with customized LabVIEW program appropriate protocol has to be followed. The basic process for executing a command involves sending the various settings for to the board, such as motor selection, half step mode, CCW direction, etc., and then appropriate signal in term of ASCII code will be sent to instruct the board to execute the parameters sent. For custom communications, all that is necessary is a device capable of sending bytes using the RS232 protocol. The port settings for communicating with the S100SMC are:

Table 3: Serial Port setting for communicating with S100SMC.

Baud Rate	Parity	Word Length	Stop Bits	Xon/Xoff Handshaking
9600 bps	None	8 data bits	1	Off

For example, the code "M0CIE" will instruct motor 0 ("M0") to rotate clockwise ("C"), infinitely ("I", and "E" starts the execution), and then later send "S" to stop the rotation, all that need to be done to repeat this exact movement is send "E" once more. But in our case the letter 'I' will never appear in instruction and replaced by letter 'D' denoting the definite number of steps. After sending either "D" the S100SMC then expects two bytes that will correspond to a 16-bit value on the board. The first byte that is sent represents the high byte on the controller, and the second represents the low byte. Please refer to *Table 6* for conversion of numerical value to a high and low byte ACSII code.

2.6 CONTROLLER SOFTWARE

The controller board was provided with standard software which is mean for testing the motor and other outputs. Please refer to *Appendix P* for software provided by *http://www.steppercontrol.com* **[9].** Beside the standards software provided, proper controller interface was developed using LabVIEW to suit the application.

2.6.1 LabVIEW

LabVIEW Motion Controller was selected to develop controller software in Graphical User Interface. LabVIEW is a graphical programming language that uses icons instead of lines of text to create applications. In contrast to text-based programming languages, where instructions determine program execution, LabVIEW uses dataflow programming, where the flow of data determines execution. In LabVIEW, user interface can be developed by using a set of tools and objects. The user interface is known as the front panel. LabVIEW is integrated fully for communication with hardware such as GPIB, VXI, PXI, RS-232, RS-485 and pug-in DAQ devices. The controller board S100-SMC uses RS-232 communication protocol. Using LabVIEW, creating, testing and measurement, data acquisition, instrument control, datalogging, measurement analysis and report generation application can be done in fraction of time taken by other programming language. Besides, stand-alone executables and shared libraries, like DLLs also can be created since LabVIEW is a true 32-bit compiler. Any program developed via LabVIEW is called VI (Vitual Instrumentation). Virtual instrumentation is defined as combination hardware and software with industry-standard computer technologies to create user-defined instrumentation solutions.

2.7 THEORETICAL REVIEW ON THE CONTROL ALGORITHM

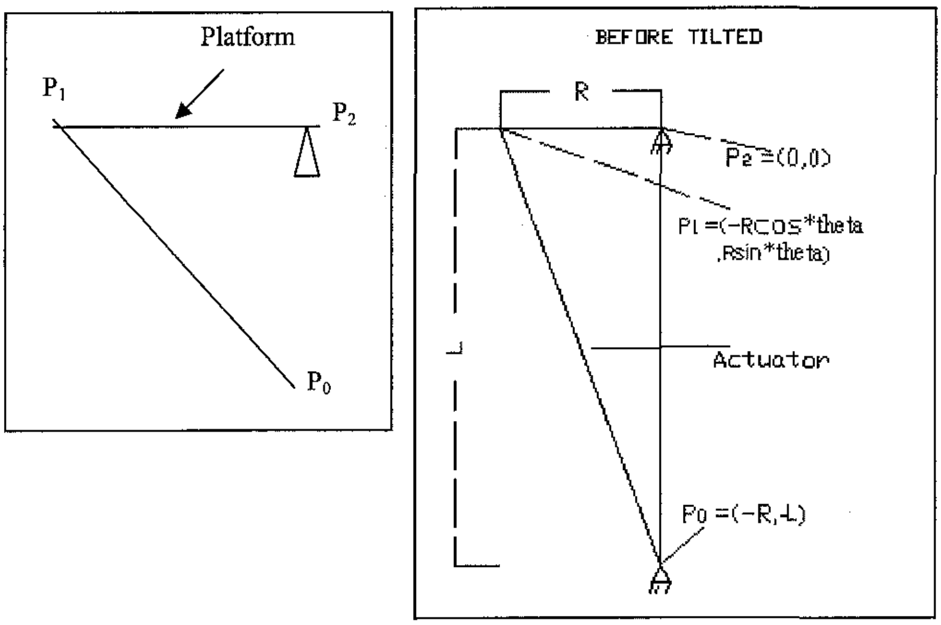


Figure 7: Coordinate representation of the structure (left); Kinematics diagram of the actuator and the radius of rotation (right).

P_0, P_1 and P_2 are coordinated describe the actual physical model of the platform.

Stepping angle of the actuator	$= (\alpha)$
Resolution (full step mode)	$= n_s = 360/\alpha$
Resolution (Half step mode)	$= n_s = 360/0.5\alpha$
Precision of the actuator at full step	$= P_{full}$
Precision at the actuator half step	$= P_{half}$
Solid length of the actuator	$= SL$
Height diff. btwn platform and the actuator mounting	$= L$
Radius of rotation	$= R$
Pulse rate applied	$= f_p$
Detected tilt in pitch direction	$= \theta_{pitch}$
Detected tilt in roll direction	$= \theta_{roll}$

Coordinates after tilted:

$P_1' = (-R\cos\theta_{pitch/roll}, R\sin\theta_{pitch/roll})$

$P_2' = (0, 0)$ -----------remain the same because this is the center of rotation.

$P_0' = (0, -L)$ -----------remain the same because this is the center of rotation.

Actuation required for compensate the tilt angles accordingly, $\delta = -[(P_1' - P_0')] - SL$
Number of pulses required for actuation, $n_{p1/2} = \delta / \{P_{full}, P_{half}\}$
Time required for actuation for actuator 1(Motor 0), $t_1 = n_{p1} / f_p$
Time required for actuation for actuator 2,(Motor 1) $t_2 = n_{p2} / f_p$
The required time for leveling process, $t_{reaction} = \text{Max}\{t_1, t_2\}$

Equation 1: Actuation-tilt angle relationship

The detail calculation on finding the allowable tilt angle range in both pitch and roll axis is presented in *Appendix J*. The calculation use the basis described by *Figure 7*.

2.7.1 Equation for the conversion from angle to required steps

2.7.1.1 Pitch axis: (Motor 0) –M0

$P_0 = (0,-L) = (0,-350)$
$P_1 = (-165\cos□, 165\sin□)$
$P_2 = (0,0)$
□ = angle of tilt (user input)

From the *Equation 1* the required extension / contraction of the actuator in Pitch axis:-

$\sqrt{[(165\cos□)^2 + (165\sin□ + 350)^2]} - 386.94$
Number steps required = (required extension / contraction) mm / 0.016 mm
Linear speed (mm/s) = pulse rate (steps/sec) × 0.016 mm/step
Time taken to achieve the coordinate = Number steps required / pulse rate (steps/sec)

Equation 2: Final equation for pitch axis (Motor 0) used in LabVIEW.

2.7.1.2 Roll axis: (motor 1) – M1

$P_0 = (0,-360)$
$P_1 = (-135\cos\square\ , 135 \sin\square)$
$P_2 = (0,0)$
$\square$ = angle of tilt (user input)

From the *Equation 1* the required extension / contraction of the actuator in roll axis:-

$\sqrt{[(135\cos\square)^2 + (135 \sin\square + 360)^2]} - 384.48$
Number steps required = (required extension / contraction) mm / 0.016 mm
Linear speed (mm/s) = pulse rate (steps/sec) × 0.016 mm/step
Time taken to achieve the coordinate = Number steps required / pulse rate (steps/sec)

Equation 3: Final equation for roll axis (Motor 1) used in LabVIEW.

2.8 TILT-ANGLE SENSOR AND CONVERSION MODULE

Tilt-angle detector is an additional part in the first phase of the project. This element is very costly. Therefore the integration of this element will be the continuation of the project to fully automate the leveling process by converting the current configuration to closed-loop control. Tilt-angle sensor have been proven in many applications to provide a reliable way to measure tilt angles in static and dynamic environments **[10].** The sensor, when connected to electronic conversion module, provides an output proportional to a corresponding inclination angle. The EZ-Tilt-3000 series was strongly recommended by "sensor" magazine **[11]** as a cheap and best alternative for angle measurement.

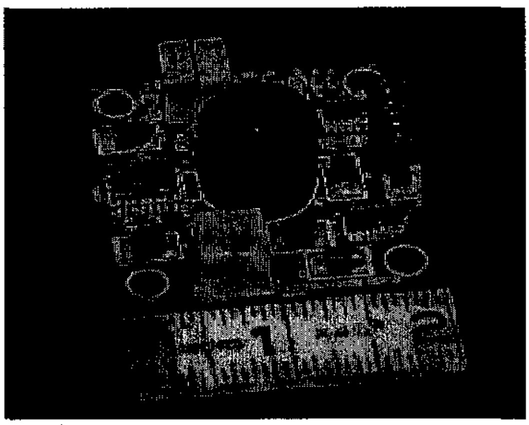


Figure 8: EZ-TILT-300-016 Dual Axis High Resolution Angle Conversion Module with tilt angle sensor.

The High Resolution two degree of freedom tilt module is built around a customized state of the art CMOS SMT and Feed Through components. A 2" x 2" assembled PCB has a total height including the sensing element of 17mm max. Analog proportional-to-tilt DC voltage output provides continuous inclination information for both axes of tilt. The module includes independent adjustments for both Pitch and Roll. Temperature value is available as a separate output for external reference. The module is capable of operating with any type of Single or Dual axis Electrolytic tilt sensor. Sensor could be remotely located if so required.

CHAPTER 3
METHODOLOGY

3.1 THE OVERALL DESIGN CONCEPT OF THE LEVELING PLATFORM

The author constructed the structure as closest possible to solid model developed using Auto CAD 2002. CAD model has been the main reference for step by step structural construction. The structure was built using a mixture of alpha-iron beams and aluminum beams. But, the position of beams was carefully planned and therefore the whole structure is considerably strong and rugged. The construction of model took 8 weeks including the testing of the motor, wiring and communication with the controller board. The resolution of the rotation angle was impressive although there were some unexpected results obtained such as bottle neck imposed on the speed of actuation by the controller board due to limitation on the supply frequency in term of pulses/sec, The methodology of finding proper wire connection was followed as recommended by *http://www.compumotor.com* **[12]** and attached in *Appendix N.* The model was having good aesthetics values due to the alpha-iron beam and simple construction. The stroke range of the selected actuators allowed appreciable rotation. Please refer to *Table 16* for detail specification of the model.

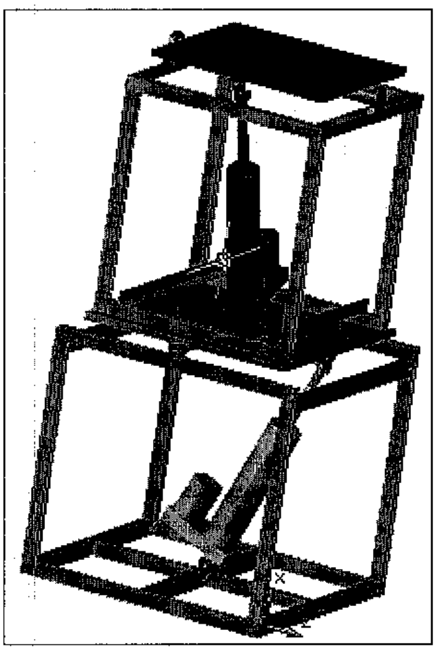


Figure 9: The CAD design of the structure.

3.2 PROJECT METHODOLOGY

Proper methodology, which is a part of the project management, was already developed at very beginning of the last semester. The Engineering design process *(Figure 3, pg 7)* was very useful in break-down the whole project to sub-modules and as mean of systematic project progress flow. The methodology was modified from time to time to adapt to the changes in design requirement and the scope of the project. The flow chart showing the overall progress of this project is demonstrated in *Appendix A.* The bottle-neck of the progress is the process involved in developing the model due the availability of the facility and tools. Basically, the overall flow of the project follows the flow chart.

3.3 DESIGN SELECTION

"Pugh's concept selection method" and weighted matrix evaluation was used to select the best among the alternatives generated. Please refer to *Appendix C* for the Pugh's concept selection table and *Appendix B* for sketches of alternatives.

3.3.1 The evaluation criteria's

1. *Fewest components*: the fewer the components the less the complexity of the structure. Therefore the higher value given for the design concept.

2. *Availability of the components*: this criteria underline the possibility of doing the structure within the time frame and limited resources (budget). Therefore the easier the components in the reach, the more the value for the design.

3. *Fast action*: the faster the driving equipment to react with the input, the better it is to be selected.

4. *Cost of components*: the cheaper the cumulative total cost for assembling the structure the higher the value given for the design concept.

5. *The kinematics compliance of the system*: this is one of the main element of the design, whereby the kinematics of the motion should have at least 4 DOF for better visualization on the stabilizing concept.

6. *Load handling capacity*: the higher the load that the table (platform) could handle, the better it is in approaching the practicability of the design. The linear actuators are more robust and capable of handling the load without much backlash.

7. *The precision of actuation for tilting minimum angle per pulse or step angle*: It is found that linear actuator have better precision in adjusting the angle of platform compare to the stepper motor due smaller discrete steps of rack motion induced by stepper motor. Therefore the use of linear actuator will be given the priority.

8. *Ease of assembling or making*: the keyword that associated with this criteria is the processes and tools needed for doing the model. The less sophisticated and cheaper process will be an added advantage in selecting the concept.

9. *Ease of PC-interface with components*: the easier the system to be interfaced with PC (CPU) for control purpose, the higher the value given for the design concept.

The weight factor was selected based on the pair wise comparison or inter-benchmarking as suggested by George E.Dieter (2000) pg. 188. The design concept with highest score will be selected.

Decision

From the ranking, the design concept C has been selected. The alternative B and C have the same score, but the selection was mode on the basis of less complexity and aesthetics property.

3.3.2 Weighted decision matrix [4]

Table 4: The weighted decision matrix for concept evaluation.

Criteria	Weight factor	Design A		Design B		Design C		Design D		Design E	
		score	rating	score	rating	score	rating	score	rating	score	rating
Fewest components	0.3	10	3	8	2.4	9	2.7	8	2.4	9	2.7
Availability of the components.	0.6	8	4.8	8	4.8	8	4.8	8	4.8	8	4.8
Ease of assembling or making.	0.4	9	3.6	7	2.8	8	3.2	6	2.4	6	2.4
Cost of components.	0.5	4	2	8	4	8	4	9	4.5	9	4.5
The kinematics of the system Comply to DOF	0.8	0	0	10	8	10	8	10	8	10	8
Load handling capacity	0.1	9	0.9	9	0.9	9	0.9	8	0.8	8	0.8
Fast action.	0.2	8	1.6	9	1.8	9	1.8	9	1.8	9	1.8
The precision of actuation for tilting minimum angle per pulse or step angle.	0.25	9	2.25	9	2.25	9	2.25	7	1.75	7	1.75
Ease of PC-interface with components.	0.55	7	3.85	8	4.4	8	4.4	8	4.4	8	4.4
Total Score			20		31.35		32.05		30.85		31.15
Ranking			5		2		1		4		3

Score range:
0 – Poorly comply to the criteria
10- Excellent compliance to the criteria

Selection: Alternative C appears to be the best with both Pugh's selection method weighted matrix evaluation. The alternative B has very little difference compare to alternative D, considering the fact that the structure and the size of the components are the same except for the platform support.

Rationale on selecting the design C.

The conceptual design C complied with 6 DOF using only two actuators which are connected between the platform support and the base. It has long support at the middle of the platform to avoid deflection and the actuator. The linear actuator would have smaller discrete steps for tilting angle. There are two main disadvantage of using the rotary actuator or stepper motor compare to linear actuator.

1) Overshoot effect due to rotational driving and the moment of inertia offset from the centre of rotation. According to RS datasheet **[13],** the discrete motion of the stepper motor whether stands-alone or embedded in the rotary actuator is subject to overshoot in high-torque application. This is due to moment of inertia of the driven component/part that eventually will resist the sudden stop by the motor. Therefore the jerky-motion will be incurred. Besides, the stepper motor or rotary actuators are subjected to "backlash" compare to the linear type-actuator.

2) *Poor angle stepping precision.* The precision by directly connecting the rotary actuator or stepper is the stepping angle of the motor itself, which is approx. 1.8°. In contrast, by using a linear-type actuator, the precision can as good as 0.05° per step. Therefore the backlash do not really play significant role for linear-type actuator. **[14]**

Most linear-type actuator have 10 times better payload, linear force and holding torque compare to direct drive with stepper or servo motor. For rotary actuator and stepper motor, the gear box can be used to enhance the performance in term of precision and load handling capability. However, the accumulative cost will be higher compare to buying a single linear-type actuator.

3.4 STEPPER MOTOR SELECTION

Considering the cost constrains and project requirements, the most suitable stepper motor was chosen. The flow chart below shows the procedure of motor selection that was used to check for compliance to requirements before purchasing the linear stepper motor. Some of the basic calculation methodology is given in *Appendix G.*

1. Determining the drive mechanism component
Determine the mechanism and required specifications. First, determine certain features of the design, such as mechanism, rough dimensions, distances moved, and positioning period.

↓

2. Calculate the required resolution
Find the resolution the motor requires. From the required resolution, determine whether a motor only or a geared motor is to be used.

↓

3. Determine the operating pattern
Determine the operating pattern that fulfills the required specifications. Find the acceleration (deceleration)period and operating pulse speed in order to calculate the acceleration torque.

↓

4. Calculate the required torque
Calculate load torque. Calculate acceleration torque. Calculate required torque. Calculate the load torque andacceleration torque and find the required torque demanded by the motor.

↓

5. Select the motor
Make a provisional selection of a motor based on required torque. Determine the motor to be used from the speed-torque characteristics. Select a motor whose speed-torque characteristics satisfy the requirement.

↓

6. Check the selected motor
Confirm the acceleration/deceleration rate and inertia ratio. Check the acceleration deceleration rate and inertia ratio in order to determine the suitability of the selection.

↓

Selection complete

Figure 10: Stepper motor selection methodology. (Adopted from Compumotor Catalogue [15])

3.5 DESIGN TOOLS

Table 5: The list of tools required to construct the model.

Name / Item	Size	Quant.	Ref:
Alpha beams – large – medium (Aluminum metal)	40×3×1.5 cm 25×3×1.5 cm	12 8	4,5
Ball bearing - large - medium	45mm 25 mm	2 2	6,8
Roller bearing	Thread size 10 x 1.5 mm (hollow ball)	1	10
Universal Joint	Thread size 10 x 1.5 mm	1	-
Prospect board (transparent)	45 x 45 x 10 mm	1	9
Prospect board (black)	30 x 30 x 10 mm	1	7
Metal joints for alpha beams	40 mm long (2 holes)	12	-
Plastic joints for alpha beams	40 mm long (2 holes)	12	-
Solid steel rod (steel) A36	D– 2 cm, L -36.5 cm, T-0.2 cm D– 1.5 cm, L -9.0 cm, T-0.2 cm	1 1	5
Standard nut and bolt (for joining the alpha beams)	Various sizes	48 sets	-
Standard nut and bolt for locking the actuator, roller bearing, universal joint, ball bearings,	Various sizes	20 sets	-
U – shape nut and bolt (for locking the board to solid steel rods)	Various sizes	4	-
Linear Actuator -range -type -rpm -payload	 150mm DC – 12 – 48 V Max. 1500 rpm approx. 710kgf	2	1,2
Controller board (min. 2-axis) S100SMC.	-	1	-
LabVIEW motion controller	For graphical interface controller development.	1	-
AutoCAD 2000	For 3D solid model construction	1	-

3.5.1 The major processes or machines involved in developing the model.

- Folding machine – to fold the connectors of alpha-iron beams, bracket for platform-linear actuator connection
- Wheel cutter machine - Cut the alpha-iron beams to custom size, prepare the alloy base for actuators (to clamp the ball bearing).

- Drilling machine – drilling holes on aluminum beams, alloy base for actuators, connectors of alpha-iron beams, couple and etc.
- Threading tools - to thread the alloy base for fixing the ball bearing or universal-joint.
- CNC trainer - to prepare bores in the alloy to form bearing brackets. Refer to *Appendix I.*

3.6 CONTROL ALGORITHM DEVELOPMENT

Step 1: Identification of the kinematics spatial constrain of the structure referring to the actual size of the components.

↓

Step 2: Finding the relationship between the tilt angle and the required actuation/retraction.

↓

Step 3: Relate to the required pulses and time for compensation.

↓

Step 4: Formulate the relationships.

↓

Step 5: Validate the relationship by using Matlab or Excel sheet.

↓

Step 6: Analyze the variable for allowable maximum tilt.

↓

Step 7: Input the algorithm in the form of graphical language-block diagram (LabVIEW).

↓

Step 8: Develop user graphical interface (Front panel via LabVIEW) and validate the results.

Figure 11: Methodology of controller software development.

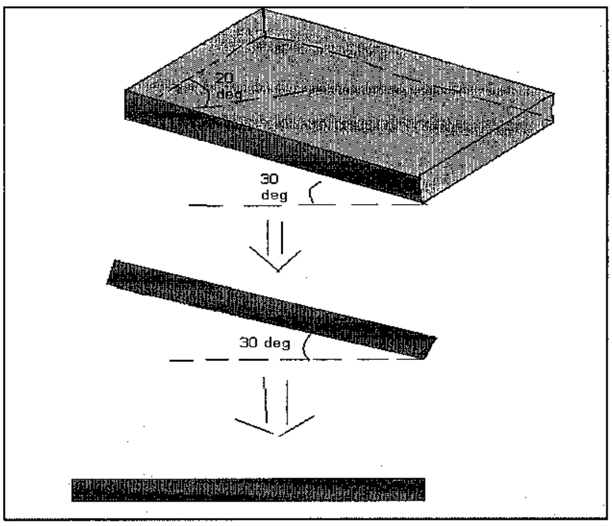


Figure 12: Illustration on segregating the tilted angle into pitch and roll accordingly

Figure 12 shows the segregation of the platform's tilted angle into pitch and roll. The methodology of angle segregation by the tilt angle sensor covered in *section 2.8* is also similar to *Figure 12*. For the demonstration purposes the angles in both pitch and roll direction are presumed to be uncoupled. The user will input both angles in the controller interface (front panel). The platform will be tilted according to the input angles. If the tilt angle sensor is to be used, the leveling process can be automated, please refer to *Figure 20 and 21* for comparison.

3.7 DESIGN DEVELOPEMNT OF THE CONTROLLER PROGRAM

A custom front panel (Graphical User Interface) was designed to interface with the controller board. The tutorials and design procedure were obtained from *http://www.ni.com* **[16]**. This kind of front panel is often used in process control programs. The message below the digital control is the message (*Figure 13*) for the users to keep within the motor limits and board specification as determined through experimentation. A message will pop-up on the interface, prompting the user to enter the required filed accordingly in the event the user doesn't follow the instruction. The message prompt is accompanied by an instruction to halt the motor. Basically the number steps and the speed be converted to a minimum delay value to be used for communication with the controller board. The absolute result refers to absolute coordinate of the platform that determines the limitation and incremental refers to the angle input compare to previous state. The limitation displayed refers to the absolute Home button is placed for the user to bring the platform to original position which is by default 0^0 degree (absolute) in both pitch and roll direction.

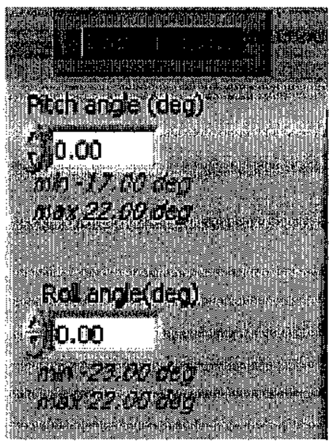


Figure 13: Message for users on the limitation on the performance.

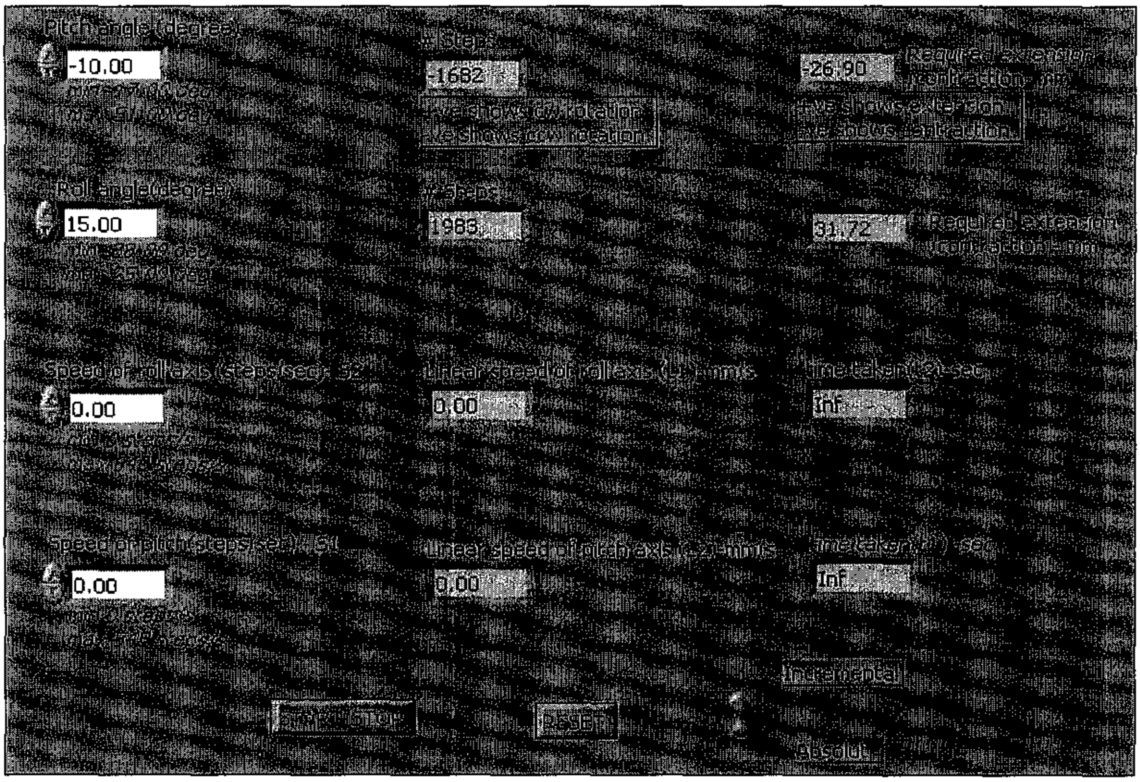


Figure 14: Preliminary design of the front panel using LabVIEW.

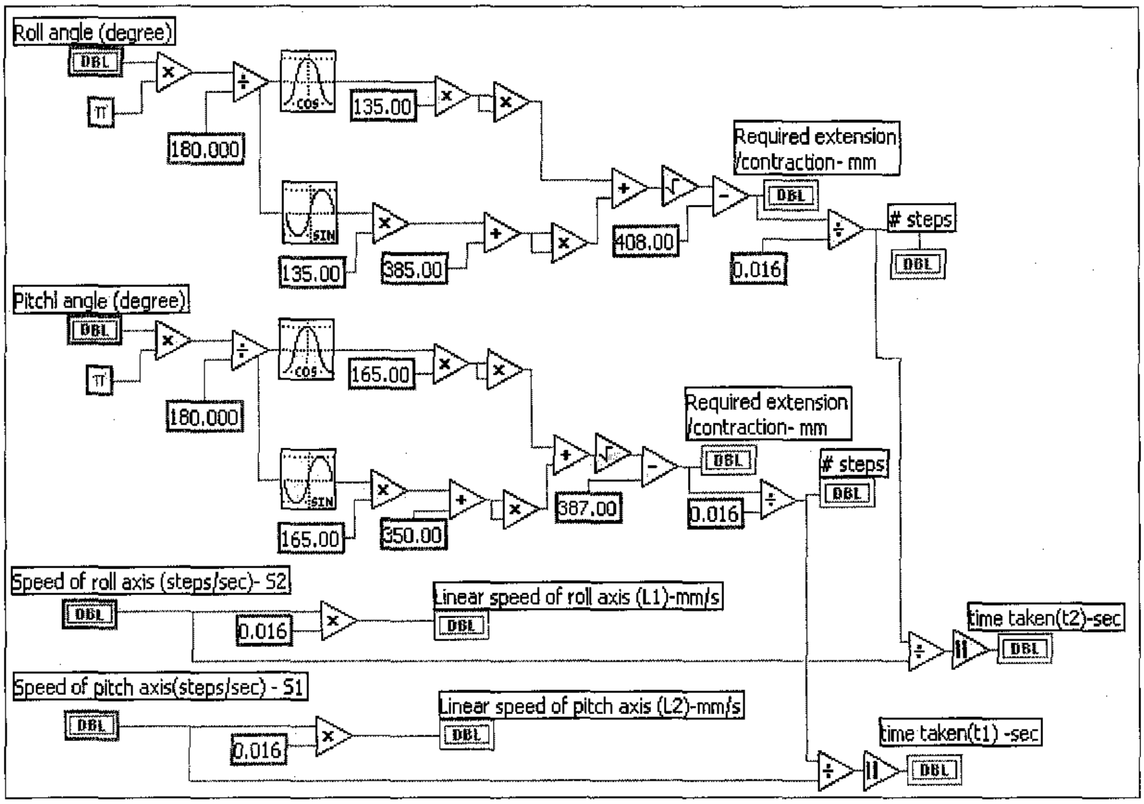


Figure 15: Preliminary design of the control diagram (block diagram).

3.7.1 Using LabVIEW functions to match the board protocol.

In order to communicate with the controller board the ACSII code was used as a medium of instruction to the board to control the motor accordingly. ASCII was used as the medium of language and the protocol of communication. Below is the example of converting a numerical value into high and low byte ACSII code which is the frequent case in developing the controller in this project.

Table 6: Example on converting the number of steps to ASCII code.

Value to be sent:	19307
Find High Byte:	19307 / 256 = 75.418 High Byte = **75**
Find Low Byte:	256 * 75 = 19200 19307 - 19200 = 107 Low Byte = **107**

So, the character "M0CD" followed by the letter designated (ASCII code) for the value 75 and 107 which is "K" and "k".

3.7.1.1 Sign convention for the Clockwise and Counter Clock Wise direction

Numeric value 67 represents 'c' in the ASCII code which communicate with board to rotate the motor in counter clockwise direction. Numeric value 99 represents 'C' in the ASCII code which communicate with board to rotate the motor in clockwise direction. Therefore, algorithm below used to compare the values (output) of the function in *Equation 3, pg 16.*

if
{Steps < 0
mode = 99}
if
{Steps ≥ 0
mode = 67}

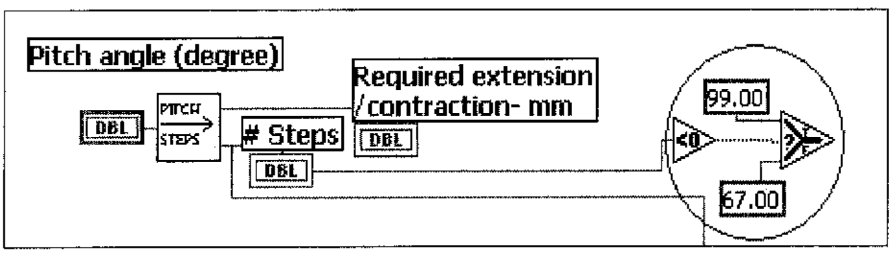


Figure 16: ASCII code for clockwise and counter clockwise direction.

3.7.1.2 Other parameters

Motor designation - Pitch axis will be referred as Motor 0 by sending '**M0**' in ASCII code and Roll axis will be referred as Motor 1 by sending '**M1**' in ASCII code.

Ramping – ramping is basically the motion of the motor at the beginning and ending of the rotation sent. Ramping function will be disabled in our application by setting the constant '**r**' as ASII code. But depending on the situation the program can be modified to enable the ramping function by just changing the ASCII character.

Full step/half step- For the purpose of the concept demonstration the mode will be set to full step by setting the character '**F**'.

Holding Torque – Holding torque refers to the windings that were energized on the last step of the rotation will continue to be energized until another movement is executed when holding torque is enabled. In our application ASCII code '**H**' will be set to denote that holding torque is enabled.

Type of rotation (Infinite/definite) – the rotation in our application is in definite mode of rotation according to angle input and the tilt-angle sensor. The angle will be automatically be processed according to time delay specification. Therefore the character '**D**' denoting the definite type is enabled.

Min/ Max delay value – When ramping is enabled the motor will slow down from min delay value to max delay value before brought to halt and vice versa. But since the ramping is disabled, only min delay value used to control the speed of the motor. Therefore, a character'**t**' followed by two digit numeric value send to denote the speed of the motor. The value range from 1 to 65535 and the lower the value the higher the RPM of the motor.

3840/x=y
x= user input (steps/sec)
y= min. delay value (nearest smallest number)

Equation 4: Relationship between min. delay and steps/sec.

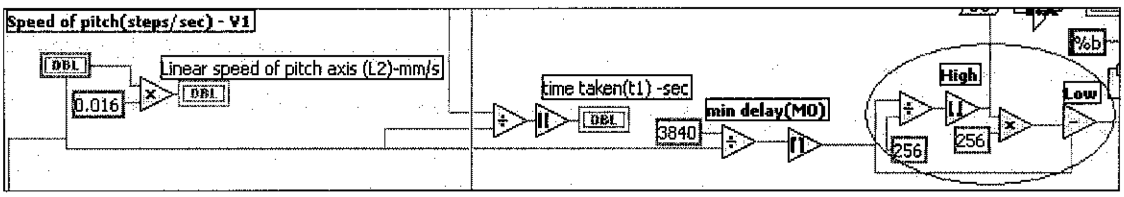


Figure 17: ASCII code for min delay value in the block diagram.

The sequences of instruction in ASCII codes are very important in order for the controller board to execute the instruction accordingly. The hierarchy of information was already set by the manufacturer of the board.

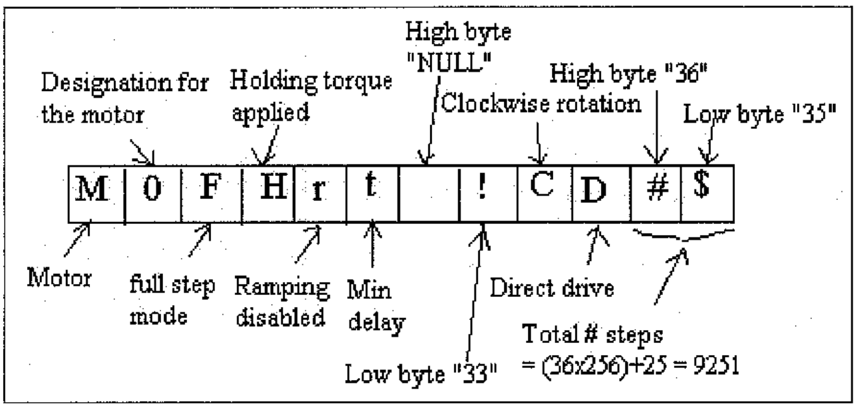


Figure 18: ASCII code sequence for communication protocol.

3.7.2 Data flow in the program and control process.

Figure 19 shows how the data flow from the output of the function in *Figure 14* to finally instruct the motor in term of rotation and the speed of the motor by complying with the communication protocol of controller board which requires ASCII code to understand the instruction.

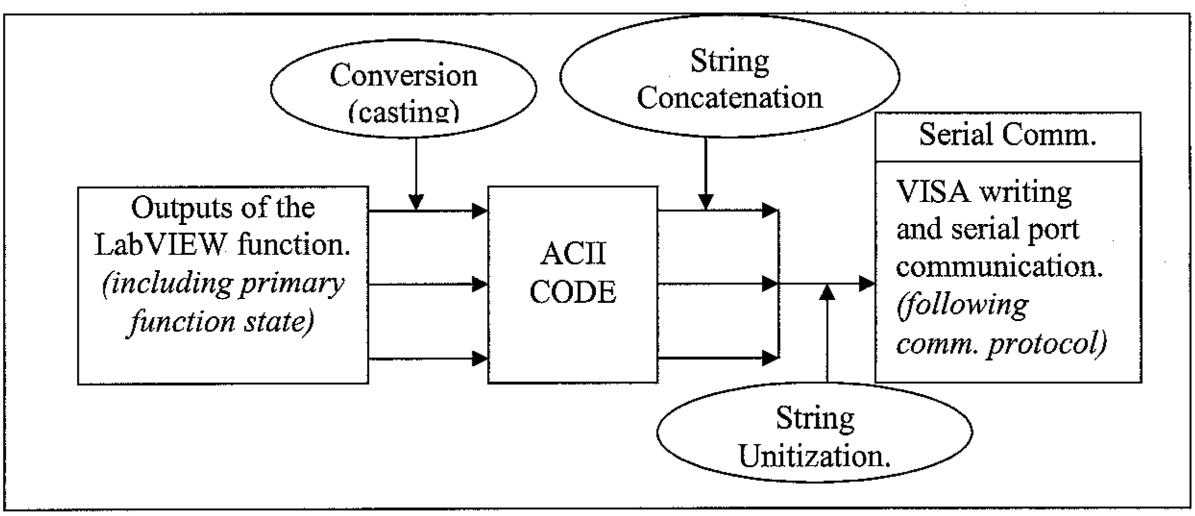


Figure 19: The flow of the data from the user input to the instruction received by the board.

3.7.2.1 Direct control flow (manual tilt angle control)

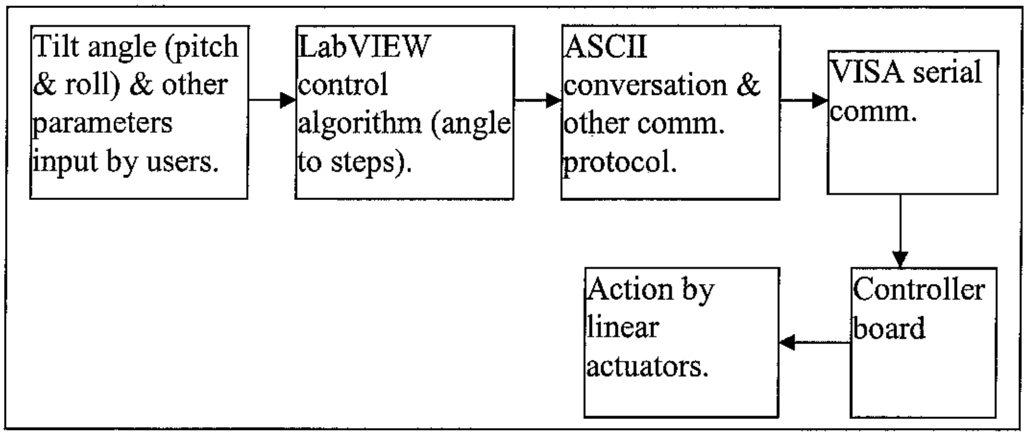


Figure 20: Control flow chart of direct control.

Figure 20 shows the control flow chart that starts with the inputs by the user being converted into ASCII codes as medium of instruction to the board and ends with communication between board and the motor. The tilt-angle sensor was not used in this process which leads to manual control of the platform's tilt-angle.

3.7.2.2 Automated leveling with using the tilt-angle sensor.

Tilt-angle sensor module could be utilized for enhance the functionality of this system by fully automate the leveling task. Meaning to say there will be no user input in term of angle (pitch and roll), whereby this angle will be automatically detected at prescribed interval (depending on the sensor's specification) to be updated in the controller program. Therefore, the controller software developed was properly documented with sufficient description and references in order for easier modification to integrate the sensor. So, the control flow will be converted from open-loop concept to a closed loop whereby the tilt angle of the platform could be checked in some interval. Therefore it is expected the continuation of this project, will be able to utilize the advantage of this element. Also, more applications could be justified thereafter.

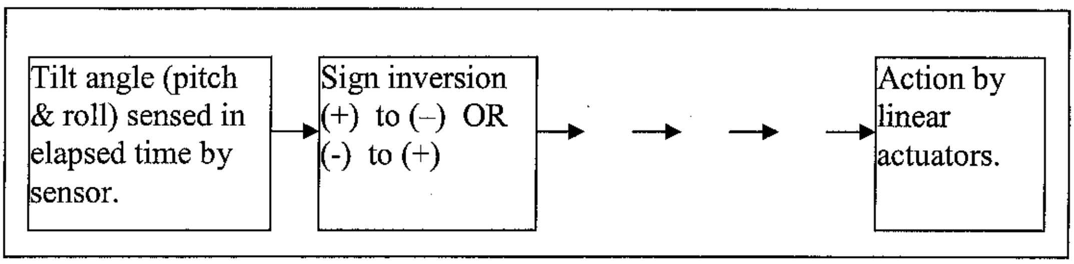


Figure 21: Control flow chart of automatic leveling process.

CHAPTER 4
RESULTS AND DISCUSSION

4.1 ANALYSIS ON THE WORKING PRINCIPLE OF THE "TWO -AXIS LEVELING PLATFORM"

Components relation diagram, *Figure 1* portrays the prime tools to enable the leveling platform to perform its intended task. First the tilt-angle sensor will detect the tilted plane coordination at any instant in terms of by pitch and roll. A dual axis sensor should be used for this purpose. This angle will be converted into proportional electrical message (i.e current) by a conversion module. The converted data will be sent to PC (software) through a parallel port.

This data will be processed trough control program algorithm to convert it into required retraction/extension of the actuators (1 and 2) for tilted angle (pitch and roll). Knowing the resolution of the actuator (mm/rev) and the pulses/rev, the amount of pulses to be sent to actuators will be determined. Next, this information will be sent to controller board through the serial port. Finally the controller board will energize the motor by the amount of pulses required. The time period for the leveling is depend on the pulse rate, which is adjustable as long as the resultant rev/min does not exceed the allowable speed of the motor. The control flow above applies to current design except that the users have to enter the value of pitch and roll angle.

4.2 PHYSICAL MODEL

Again the scope of this project is to design and develop both physical model (hardware) and the controller (software). The model was successfully built according to the project progress flow chart in *Appendix A.* The model was very strong, rugged and having sufficient aesthetics values. Despite of that, the construction of the model was planned well for modularity and ability to re-assemble each component to make it flexible for future modification.

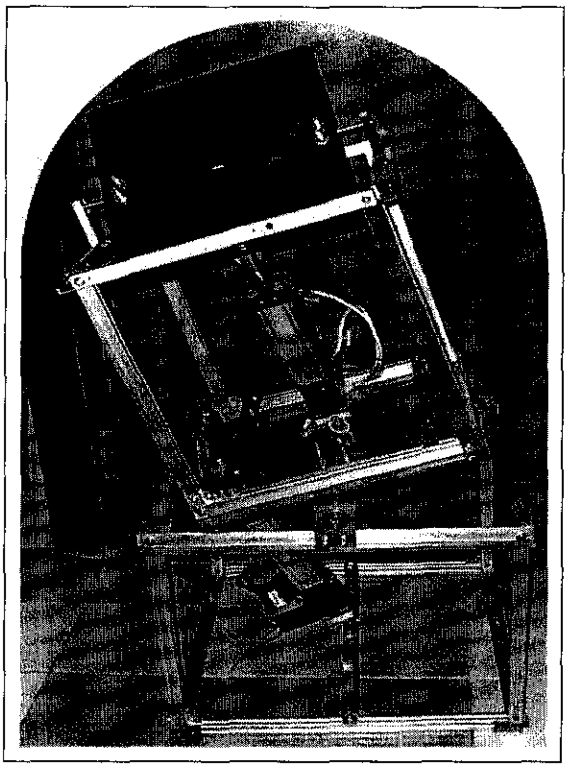


Figure 22: Physical model appearance.

4.2.1 Analysis on the architecture of the platform.

4.2.1.1 Solid properties

Since the physical model constructed is a very close to the CAD solid model, the properties obtained from the CAD model were representing the properties of the physical model as shown in the *Table 6*. But, the values given are only approximation of the actual value.

Table 7: Solid properties of solid model drawn in AutoCAD.

Description	Property values (approx.)
Total volume	6000 cm^3
Approx. ave. density	7.5 g/cm^3
Total mass	6000 $cm^2 \times 7.5\ g/cm^3 \approx 45$ kg
Centroid	X: 403mm Y: -204mm Z: -205 mm
Moments of inertia	X: 6.46E+11mm Y: 1.64E+12mm Z: 1.64E+12mm
Bounding box	X: -16 to 784mm Y: -435 to 20mm Z: -418 to -3 mm

(Please refer to technical drawing in attached CD for coordinates of the structure)

4.2.1.2 COG (Center Of Gravity) of the structure

The model created using solid model AutoCAD had a very high COG. Therefore, some modification done on the structure; to lower the COG and the resulting structure was very stable even with the linear actuators moving. Refer to *Appendix H* for technical drawings of the model. Some of the techniques used to lower the overall height of the model were:-

- Place the actuator in slanted position and center of each stage.
- Distribute the weight of the actuators facing opposite each other as to act as counter weight.
- Design shorter couplings to attach the actuator to the platform (top and bottom).
- Usage of solid steel shaft to support the platform in the middle.

4.2.2 Performance characteristic of the motors.

As mentioned earlier in *section 2.5*, due to compliance to the board specification the author has no other option than use two different stepper motor for same type of linear actuator. *Table 8* summarizes the specification obtained for those motors.

Table 8: Performance comparison between theoretical and actual characteristics of the motor.

	Performance parameters	Theoretical / Target performance	Actual / tested performance
PARKER, Compumotor	Max. RPM	1500 (5000steps/s)	164.4 (548 steps/s)
	Max. Linear speed (mm/s)	80	8.768
	Resolution (mm/step)	0.016	0.016
MINEBEA, Astrosyn	Max. RPM	800 (2667 steps/s)	192 (640 steps/s)
	Max. Linear speed (mm/s)	42.67	10.24
	Resolution (mm/step)	0.016	0.016

4.2.3 Connection of the linear stepper motor to controller board.

As mentioned earlier, the actuators obtained without manuals or specification sheets. The basic theory of wiring was followed as per recommended by Compumotor catalogue [7]. Many trials were made by interchanging connection of wires to figure out the correct configuration. Many observations were made during the test. Those observations are:-

1. There is more than one possible configuration in communicating the actuator (PARKER) to the controller board.
2. There is only one configuration is possible for MINEABBA motor and the color codes matched the recommended types by *http://www.steppercontrol.com [9]*
3. The type of the connection made with the controller board is a parallel type connection.
4. The motor used are 2-phase type.
5. Usage of two PARKER Compumotor as mentioned earlier will result in more than allowable current requirement which is 6.8 A. Therefore testing the motors simultaneously results in current overload.

4.2.3.1 Connection according to wire color codes (PARKER Compumotor)

Number of phases – 2 (A & B) - parallel
Number of leads - 8
Size - FEMA 23

Table 9: The connection of the motor according to color codes.

Lead Number							
1	2	3	4	5	6	7	8
Red	Blue	Green	Orange	Yellow	Black	Brown	White

Phase power leads (Center taps)
Centre taps A – 2, 5 – blue and yellow } V^+
Centre taps B – 4,7 – orange and brown }

4.2.3.2 Possible combinations of connection for PARKER Compumotor in series. *(Please refer to Appendix M for connection diagram)*

Table 10: Phase identification on the controller board input pins

Pin number in the controller board	Input
1	Phase 1 (A^+ / A^-)
2	Phase 2 (B^+ / B^-)
3	Center taps (A & B)
4	Phase 3 (A^+ / A^-)
5	Phase 4 (B^+ / B^-)

Table 11: The possible connections configuration for stepper motor from PARKER Compumotor.

Identification number on the controller board (PARKER Compumotor)				
1	2	3	4	5
Red	Green	Blue, Yellow, Orange, Brown.	Black	White
Red	White	Blue, Yellow, Orange, Brown.	Black	Green
Black	White	Blue, Yellow, Orange, Brown.	Red	Green
Black	Green	Blue, Yellow, Orange, Brown.	Red	White
White	Black	Blue, Yellow, Orange, Brown.	Green	Red
White	Red	Blue, Yellow, Orange, Brown.	Green	Black
Green	Black	Blue, Yellow, Orange, Brown.	White	Red
Green	Red	Blue, Yellow, Orange, Brown.	White	Black

4.2.3.3 Connection according to wire color codes (ASTROSYN MINEABBA)

Number of phases – 2 (A & B) - parallel
Number of leads - 6
Size - FEMA 23

Table 12: Connection configuration for stepper motor from ASTROSYN-MINEBEA

Position of the pins in the controller card (ASTROSYN – MINEBEA)				
1	2	3	4	5
Red	green	Black & white	Red stripe on white	Green stripe on white

4.3 CONTROLLER PROGRAM

The controller program for this project is created using LabVIEW from National Instrument. Graphical programming language provides easier access to many functions that were embedded in the software and the creation takes a fraction of time other programming language. The debugging was also very simple. The created program (VI) was converted to a stand-alone application (exe) and the methodology of creating the said program was available in LabVIEW help menu. Therefore the controller could be used in any PC even if the LabVIEW software is not installed. *Figure 23* shows the interface of the controller (front panel) with all the necessary basic features. The program was properly documented with detail descriptions for future modification especially for integrating the tilt-angle sensor. Please refer to *Appendix Q* for complete documentation of the controller and the language used for creating this interface known as the "control panel".

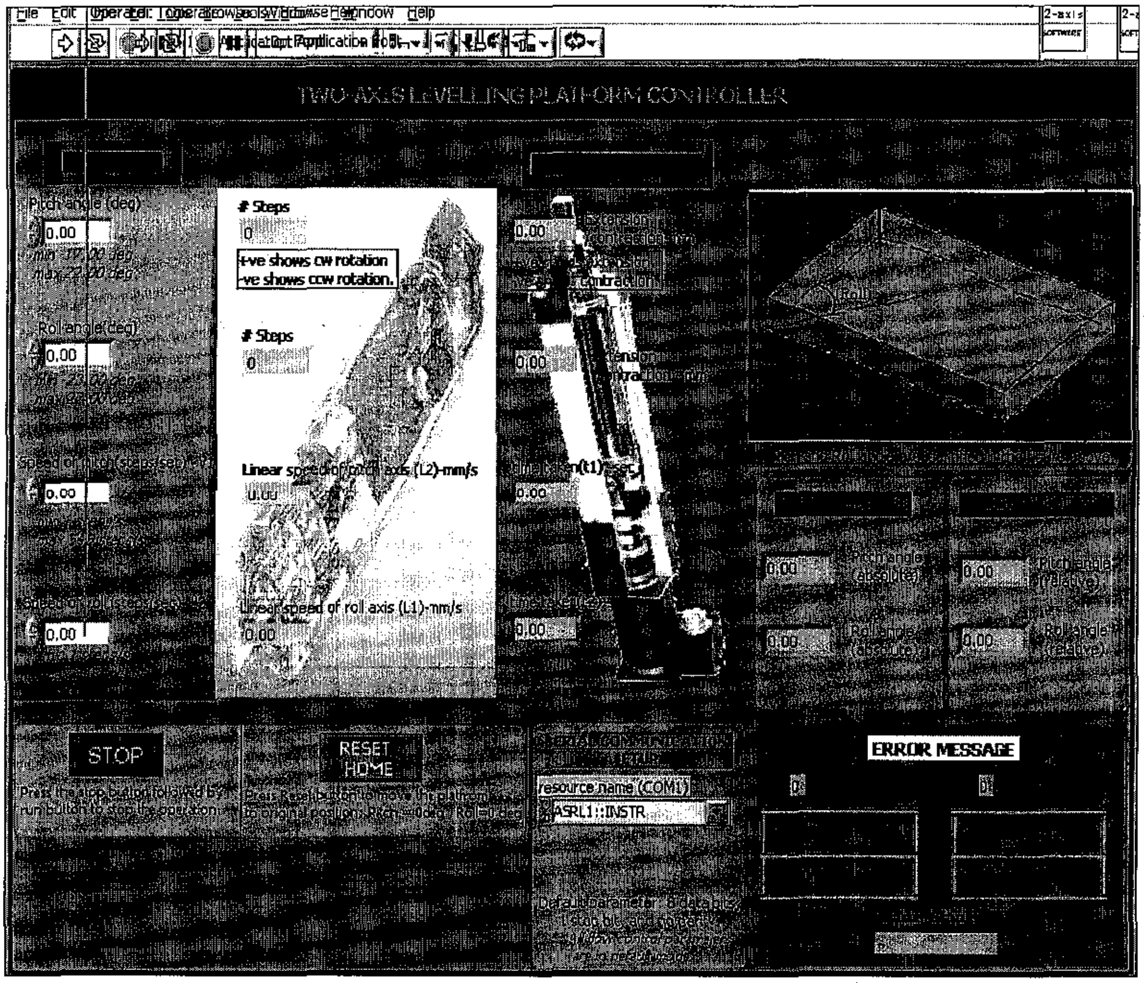


Figure 23: Controller Interface for Two-axis leveling platform.

4.3.1 Usage of Sub-VI to simply the function and control diagram

A VI within another VI is called a subVI. A subVI corresponds to a subroutine in text-based programming languages. Using subVIs helps to manage changes and debug the block diagram quickly. Typically, when creating a LabVIEW application, start at the top-level VI and define the inputs and outputs for the application. Then, construct subVIs to perform the smaller tasks within the top-level VI. This modular approach is one of the strengths of LabVIEW. Using subVIs make applications easy to understand, debug, and maintain. The circled icon in diagram below represents a sub-VI that contains a mathematical function of converting the tilt- angle to the required actuation/retraction. Therefore, the block diagram (control panel) is very much simplified.

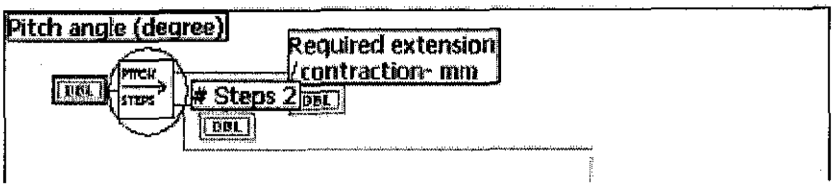


Figure 24: Control diagram (block diagram) simplification via Sub VI.

4.3.2 Analysis on the interface of the system and result of interfacing between the elements.

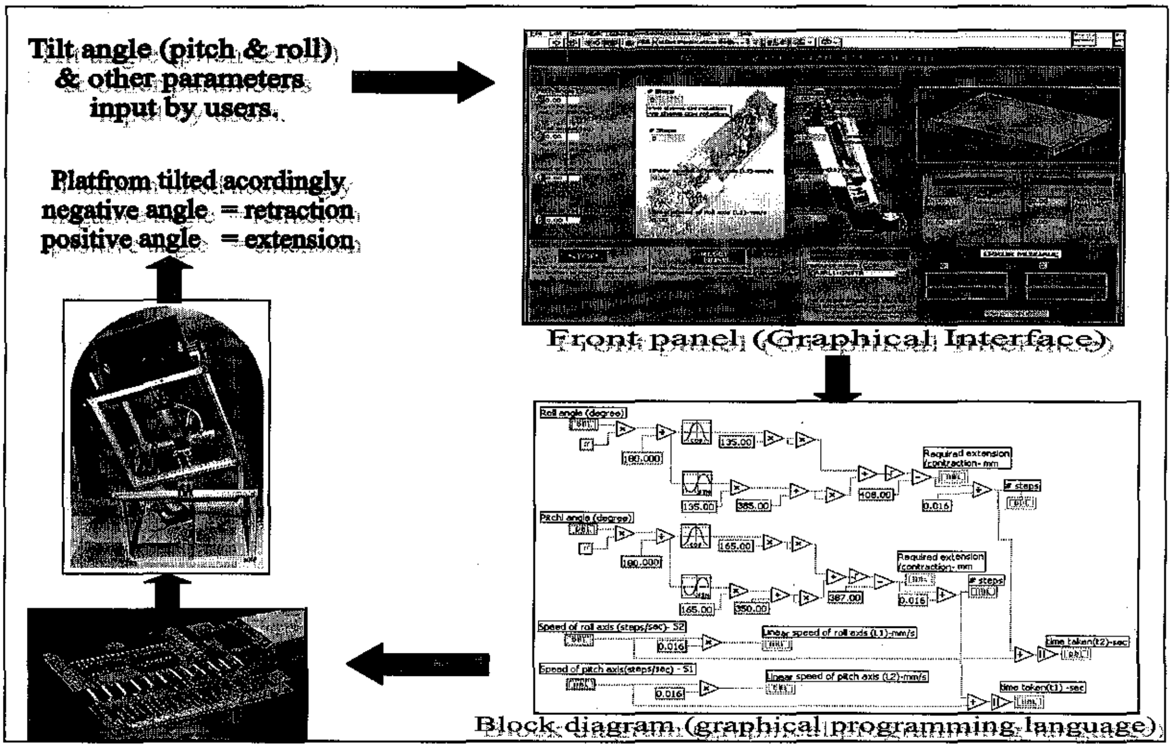


Figure 25: The result of the system interfacing.

Figure 25, shows the final result of the interfacing all the elements in the system including the user input, front panel (GUI), block diagram, controller board and actuator. The user input is interfaced to the Graphical user Interface (GUI) or front panel via controller. These inputs are to be analyzed by the functions in the block diagram and the output will displayed in the front panel. The final output in the form of ASCII character will be processed and arranged by VISA function, and the instruction will be sent to controller board. The pulse generator in the controller board will generate required number of pulses with requested frequency as to comply for the speed input ands the direction of the motor rotation. Finally the actuator will actuate or retract depending on the instruction on the direction of rotation.

4.3.3 Comparison between manual control (open-loop) and automated self-stabilizing function (closed-loop)

For the purpose of demonstrating the concept, the platform is tilted directly with the input of the user. This open-loop control could be used for some application such as remote controlling the camera by using computer or any means micro-controller. But for the application requires closed-loop control whereby the stabilizing process must be automated using tilt-angle sensor and the direction of the actuator's actuation must inverted in term of the number of steps calculated form the function. Please refer to *Figure 21* for control flow for self-stabilizing application. For application like camera positioning the tilt angles can be manually entered depending on the need. Interfacing the tilt-angle sensor using LabVIEW is simple as icons are readily available for different type of instruments and sensors. The protocol must be matched to enable communication.

In contrast to manual angle input by the user in manual control, the tilt angle sensor normally comes with interval time specification in updating the data (i.e the tilt-angle into software) in fraction of second that can be set according user preference and the application. For example, if 0.1 seconds of delay is set, than the tilt-angle sensor will update the angles in the *Figure 21* every 0.1 seconds. Therefore the final product design specification (PDS) of this system depends partially on the application and the specification of the tilt-angle sensor. If the angles changes; ***(d□/dt)*** are very large, then very high speed steppers are required to compensate the tilted angle. Notice that since the data is updated in a fraction of time the motor does not necessarily have to rotate to the full extension of the linear rod. Therefore, it is very rare that, angles such as 20 deg would be detected in any axis. The tilted angle might in the range of 1 to 5 deg maximum.

4.3.4 Using the features of the controller interface.

There are few basic interactive features designed in the controller interface such as pitch and roll angle input by user, speed of the axis input by user, error denotation, error message prompt, reset/home button and stop button.

Table 13: Comparison between absolute and incremental results.

Cyc.	Pitch angle input/ any other input	Resultant angle (Absolute)	Resultant angle (Incremental)
1	+ 10°, 500 steps/s	+10°	+10°
2	+ 10°	+10° *(remain in the same position because no absolute difference)*	0° *(no increment in the angle)*
3	+ 30°	+10° *(message prompt alerting user the cumulative angle exceeded the limit and the error denotation box for angle shows numerical value 1)*	+10° *(no change, remain in previous state due to input error)*
4	Home button pressed.	0° *(back to original/home position)*	-10° *(to move from +10 to 0°)*
5	+ 21°	+ 21°	+ 21°
6	- 10 °	- 10 °	- 31 °
7	+ 30°,500steps/s	-10 ° *(message prompt alerting user the speed exceeded the limit and the error denotation box for speed shows numerical value 1)*	- 31 ° *(no change, remain in previous state due to input error)*

*** Sign convention:*
-Ve of angle, actuation refers to retraction of the linear actuator and vice versa.
Positive tilt angle is any angle above the horizontal plane.

4.3.5 Sample result of the leveling process.

Some denotation and symbols were used throughout the analysis and programming to ease the development and as mean of consistency. Those symbols are presented in Literature Review and Theory *chapter 2.* A sample of the simulation results via using the controller interface is presented in *Table 14.* Consistent symbols were used throughout the analysis.

Table 14: Result of the sample calculation for analysis.

Description	Symbol(unit)	Formula	Value
Stepping angle of the actuator	α (°)	-	1.8 °
Full/Half step	-	Full step	Full step
Resolution	n_s (numb. of steps/rev)	360 °/ α	200
Precession at full step	P_{full} (mm)	-	0.016
Precision at half step	P_{half} (mm)	-	0.008
Stroke (actuator 1& 2)	S (mm)	-	100
Solid length (actuator 1)	SL_1 (mm)		387
Solid length (actuator 2)	SL_2 (mm)		384.5
Radius of rotation (actuator 1)	R_1 (mm)	-	165
Radius of rotation (actuator 2)	R_2 (mm)	-	135
Pulse rate applied	f_p (pulses/sec or Hz)	-	548 (actuator 1) 548 (actuator 2)
Detected tilt in pitch direction	θ_{pitch} (°)	-	+ 10° (above the horizontal plane)
Detected tilt in roll direction	θ_{roll} (°)	-	+ 10° (above the horizontal plane)
Outputs			
The actuation required by actuator 1	δ_1 (mm)	-[(P'_1-P_2)] -386.94 mm]	-25.11 (required retraction)
The actuation required by actuator 2	δ_2 (mm)	-[(P'_1-P_2)] -384.45 mm]	-21.36 (required retraction)
Number of pulses required, actuator 1	n_{p1} (pulses)	δ_1 / P_{full}	1569
Number of pulses required, actuator 2	n_{p2} (pulses)	δ_2 / P_{full}	1335
Time required for actuation, actuator 1	t_1 (seconds)	n_{p1}/f_p	2.86
Time required for actuation, actuator 2	t_2 (seconds)	n_{p2}/f_p	2.44
The required time for leveling process.	$t_{reaction}$ (seconds)	Max {t_1, t_2}	2.86

**Please refer to *Figure7* for identification of the symbols like P_1-P_2.

Notice that the maximum time taken is equals to longest time taken by any actuator because both the actuators will be acting simultaneously. The bottle neck in term of reaction time in stabilizing the platform is the parameter pulse rate f_p or simply the frequency of pulses send to the motor by controller board ;800 steps/s. Thus the reaction time can be shortened by increasing the frequency.

4.4 SPECIFICATION OF THE SYSTEM

Table 15: Product design specification.

Physical model		
Description	**Specification**	
Total weight	45 kg	
Total size	800x455x415mm (HxWxL)	
Allowable load	10 kg	
Material for construction	Alpha-iron beams and aluminum beams	
Material for platform	Prospect boards	
COG	410 cm from the base	
Actuators		
	Actuator 1(pitch axis)	**Actuator 2 (roll axis)**
Type	Bipolar – 2-phase	
Limit switches	Included (5Vdc)	
Drive configuration	Ball screw, belt-driven.	
Gear box	N/A	
Operating temperature	0-60°C	
Max speed	152 mm/s	
Duty cycle	75%	
Max. Acc	1.0 G	
Max Thrust	220 Ib.f : 970 N	
Max Travel	100mm	
Repeatability	16μm	
Other special features	- Precision anti-rotation roller bearing rod support carriage - Angular contact thrust bearings - Ground and polished stainless steel thrust tube - Modular construction.	
Closed length	340 mm	330 mm
Solid length	387 mm	384.5 mm
Open length	440 mm	430 mm
Allowable actuation	- 47 mm to + 53 mm	-54.5 mm to 45.5 mm
Voltage requirement	48 Vdc	12 Vdc
Current requirement	3.4 A	1.2 A
Power	91 W	-
Tilt angle range	-17.2 deg to 22.3 deg	-23.6 deg to 22.4 deg
Precision	$\approx\pm$ 0.005 deg	$\approx\pm$ 0.005 deg
Resolution	200 steps/rev	200 steps/rev
Max RPM	1500	800
Linear speed(actual)	8mm/s	10mm/s
Linear speed(Theoretical)	80mm/s	42mm/s
Controller board		
Type	S100SMC from steppercontrol.com	
Number of control	3 axis (independently)	
Communication input	DB9 serial cable (only using Rx, Tx, and Gnd)	
Communication protocol	ASCII code	
Frequency limit	600 Hz (600 steps/s)	
Type of motor	Unipolar or bi-polar	
Power supply	120V 60Hz	
Load handling capability	50 V max and 5A max.	
Limit switches input	3 slots which requires 5V.	
Heat dissipation	Additional port to add TIP 120 chips	

Custom communication	Through RS232 protocol
Outputs	Capable of Sinking up to 200mA Each for Relays, etc.
Protection	True opti-isolation to protect the computer
Controller software	
Platform	LabVIEW, motion controller
Software communication protocol	ASC11 codes
Software basic features	- Angle input in pitch and roll - Speed input in steps/s - Output parameter indication - Concept illustration - Absolute and relative angle indication - Error message box - Message prompts upon input limit reach. - Serial communication set-up function - Context helps for each element. - Emergency stop button - Home/reset button.

The dimensions of the structure and position of the actuators roughly follows the CAD model. But some of the actual dimensions are simplified in the solid model due to its complexity. But, primarily the radius of rotation, solid length, open length and closed length are followed accordingly. All the calculation and analysis are done based on the AutoCAD drawing of the system and the actual physical model itself.

4.5 COSTING

The author put-in a lot of effort throughout the project to keep the overall cost within the allocated budget, RM500. The used material and actuator were purchased to comply with budget constrain. Table below shows the break-down for the cost of the project.

Table 16: Cost estimation for the model

Item	**Cost**
Actuators (x 2)	RM 150
Controller board	RM 200
Construction metals - Alpha beams - Aluminum bars	RM 50
Joints and bearings	RM 50
Prospect boards (x2)	RM 20
Miscellaneous	RM 30
Total	**RM 480**

4.6 DISCUSSION

Most of the analyses were done based on the CAD drawing. The dimensions were based on the actual beams used to construct the model and the linear stepper motor. The specifications obtained were based on the CAD model and the actual physical model. The tilt angle-actuation relationship demonstrated in *chapter 2* was used as the base for all the analysis. A simple coordinate system and kinematics analysis was used to analyze the relationship. From the sample result in *section 4.3.4*, at nominal pulse rate of 500 Hz, the reaction time is considerably slow. This is due to bottle neck imposed by the controller board output which is limited to 800pulses/s (Hz). This scenario is clearly indicated in *Table 9* whereby the linear motor has better specification compared to controller board. The reaction time can be further improved by replace the controller board. The working principle of the system is well portrayed in *section 4.1*. The overall process flow is clearly indicated in *Figure 1, pg.2*. The algorithm of controlling the actuators and the equations underlying working principle of the platform has been developed. This equations and algorithms were used to determine the actual specification of the motor and the controller. The constructed physical model continuously modified to balance the parameters pitch and roll angles range, and COG. Some of basic variables considered in the modification were mounting of the actuators, height of the platform from the pivot denoted by letter "***L***" and radius of rotation denoted by "***R***" in *Figure 7.* As a rule of thumb as ***L*** increases the range of the tilt angle skew towards negative direction and the same goes for variable ***R***.

The linear actuator was not provided with the data sheet thus the complete specification is unknown. A trial and error method was used to identify the proper connection to the controller board. However, the stepper motor used in the actuator was provided with useful information such as resolution, feed rate (mm/rev), and step angle, number of pulses/rev and so on. The best configuration and arrangement of the elements of the system has been identified. In fact, the detail description of the elements used in the system was obtained.

Basically all the components except for the tilt-angle conversion module and sensor were available. The integration of this element will enable closed-loop control of the system which eventually leads to a self-stabilizing feature for the system. Due to budget constrains, the integration of this element will be a continuation of this project. Although the limit switch does not play any role or specific function in manual control of the platform, but it might be crucial in closed-loop control for future modification. Some of the impressive result of the system is the precise angle coordination or the resolution which is presented in *Table 15*.

As a nutshell, the objective and the scope of the project, which is to demonstrate the concept of leveling a platform with means of mechanical motion and to design and develop both hardware (physical model) and software (controller program) were well accomplished.

CHAPTER 5
CONCLUSION AND RECOMENDATIONS

5.1 PROJECT CONCLUSION

Submission of this report indicates the completion of the project as defined by the given scope as to construct both hardware (physical model of the two-axis leveling platform) and the software (controller program). Basically, the constructed model is capable of demonstrating the idea of leveling a flat surface with two axis of rotation; pitch and roll. The main aim of the project is to demonstrate how a platform can be stabilized when its base is subjected to an uneven motion. The software developed was embedded with some features to demonstrate the concept and as a platform for further improvements in features and functions specifically embedding a tilt-angle sensor to automate the leveling process. Therefore, the main aim of the project is achieved via completion of the software development as a controller of the system.

The overall project progress was almost as planned earlier and the milestones were within schedule except for the inability to integrate the tilt-angle sensor in the system. That would require more timeframe and budget in contrast to limited time frame and resources available. The results of the project are impressive and comply with the objectives cited within the scope of the project. The author has evaluated many possible solutions in terms of trade-off between the cost, performance and ease of design and development. The configuration shown in *Figure 1* is a cost effective solution for this project by maximizing the utilization of available components (i.e serial and parallel port). The overall performance specifications obtained through numerous calculations were very promising. Although not many references were available for this specific application, the author was able to simplify the problem with some basic knowledge on kinematics and able to formulate the control algorithm. This is very significant in a real life engineering problem that requires fast and simplified analysis. The project served a significant role by invoking a challenging environment for author to achieve the objective within a tight constrains such as cost, performance, time, efficiency, model ruggedness and so on. The detail planning has been done to ensure the smooth project flow. The project was served as a tool for sharpening sound technical knowledge gained throughout the academic years. The author had the opportunity to enhance all the relevant skills needed to undertake this problem solving-driven design project. The idea of leveling a flat surface could be employed into many useful and significant applications. This project could be a springboard to explore new ways in utilizing this concept in various situations.

5.2 RECOMMENDATIONS.

5.2.1 Heat dissipation from the controller board.

One of the main problems is the heat dissipation via controller board. The continuous operation of the motor will lead to a great amount of heat dissipation. Therefore, it is advisable to monitor the board's temperature from time to time to avoid any malfunction of the components including the motor itself. Therefore, as a safety measure extra heat sinks can be mounted on the TIP 120 chips for added heat dissipation.

5.2.2 Integration of tilt-angle sensor

A tilt-angle sensor could be integrated in the system to completely automate the leveling process via a closed-loop control. It is expected that the continuation of the project will be able to integrate this components to fully control the actuator in closed-loop type control system. The controller software was properly documented for future modification considering the integration of tilt-angle sensor. A tilt-angle sensor with very low delay time is recommended for fast feedback and reaction time while reducing the need for the actuator to react for high detection range. A tilt-angle sensor with digital display would be a best option although the display can also created using LabVIEW functions.

5.2.3 Improvements of the performance of the system.

There are few things can be done to enhance the performance of the system as a whole. Mainly it involves increasing the specification of the system by using better components for the system. As mentioned earlier, the controller card imposes a bottle-neck on the speed of the actuator. Therefore, the author would like to recommend purchasing a controller card with better specifications with a higher current rating to widen the motor selection and higher frequency limit. Besides, although the developed system have remarkable precision/resolution, replacing stepper motor with a DC servo motor with encoders will further enhance the performance of the system by elimination of "jump" or missing pulses in stepper motors.

5.2.4 Methodology of concept demonstration

The author would like suggest adding more features in the developed software to better demonstrate the significance of the leveling concept. Some of the suggested improvements are motion profile selector (actuator distance vs. time) that very much relate to application such as the capturing of pictures or using any other visual application that requires continuous angle change. An equation describes planar coordinate could be used to instruct the platform to move the required angle. This equation could be segregated accordingly into pitch and roll angle. To enhance the scope of the application, the graphical angle input in addition to numerical input could be integrated. The red line (*left corner in Figure 26*) will be enabled tilting with respect to a fix point and the angle will change accordingly in the digital indicator. This method is an equivalent analogue input for the angles in pitch and roll direction.

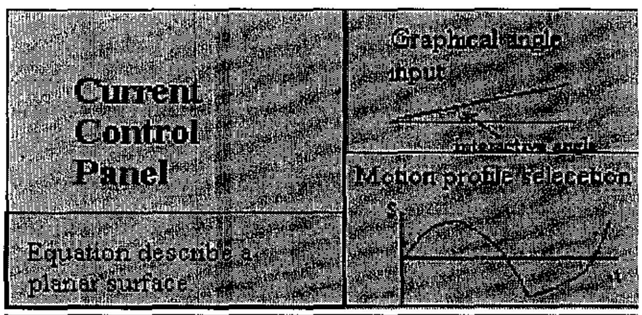


Figure 26: Suggested additional features on the front panel.

5.2.5 Programmer software

Figure 27 shows the programmer software the can perform the customized cycles of operation. A series of commands can be created and performed using a routine available in the software. Therefore, motion profile can be created according to preference. The software in the *figure 27* is created by steppermotor.com **[24]** for a different type of controller board. The author would like to suggest create similar software for current board or any other board considered for improvement.

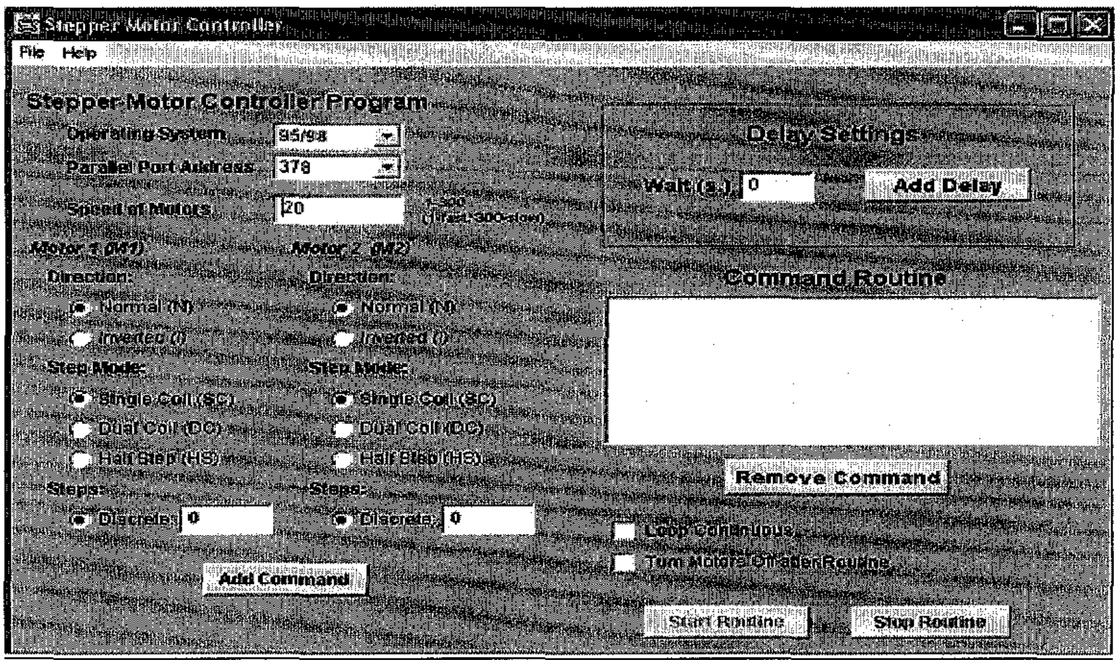


Figure 27: Programmer software front panel.

5.2.6 Improvement on physical model.

The model constructed is very stable, strong and rigid. However it can be further improved by covers and necessary means of supports so as to build a robust machine and improve the reliability. Besides, the addition of multiple holding mechanisms (fixture) on the top of the platform to hold objects with different geometry can improve the functionality of the system. Some of the suggested holding mechanisms are screws to mount camera and jaws to lock a solid material.

5.2.7 Future mathematical work.

Although the specification of the system is quite complete, the author would like to suggest the addition of more specifications such as allowable height, width and breadth of the object to be stabilized. The limitation on the positioning of the object should also be considered. The derivation of these specifications requires more extensive mathematical analysis.

5.2.8 Addition of train and elevator.

The modularity of physical model will ease the future modification. The physical is designed considering ease of dissembling. Therefore the author would like to suggest the addition of trails by means of motors and tracks to enable movement in X, and Z direction as indicated in *Figure 27*. This addition will enhance the functionally of the system in application requiring leveling or stabilizing while moving in linear profile.

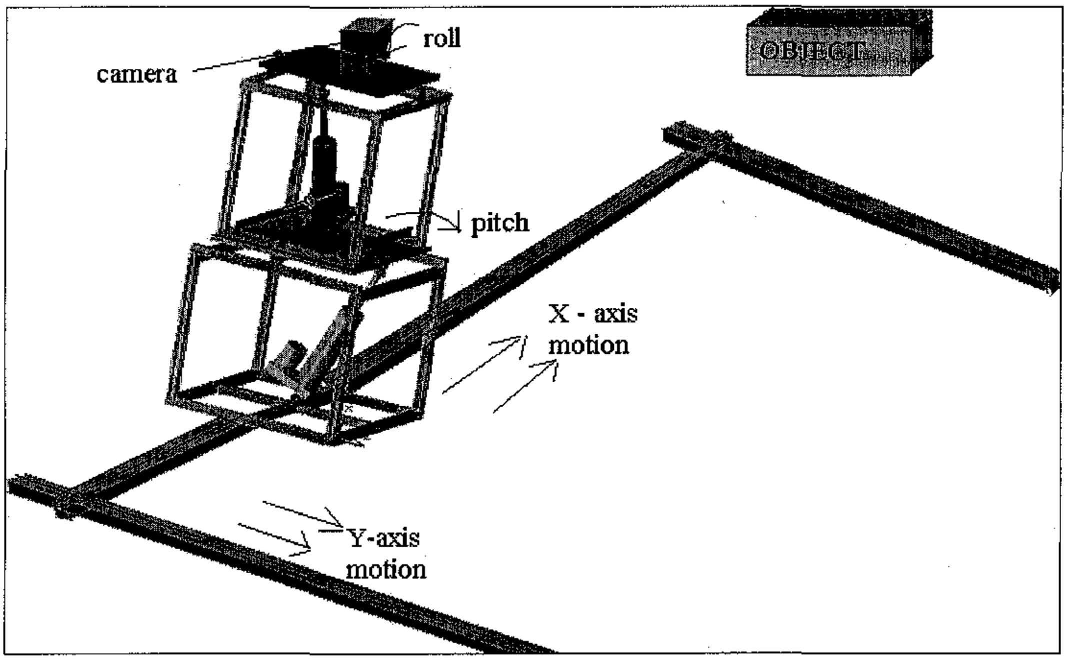


Figure 28: Example of Industrial application.

5.2.9 Filed test.

As many applications utilizes this concept, the author would recommend that the testing to be done on these different applications. This is to test the suitability and applicability of the concept.

REFERENCES

[1] I.Boldea, Syed A. Nasar, (2000) *Linear electric actuators and generators*; Cambridge university press.

[2] Sheng Xiaowen, Ni Xiaofeng, Zhang Canrong and Ma Yifeng; 2001. Ship borne Satellite Antenna Stabilizing and Tracking Platform, *Beijing Institute of Electronic System Engineering, Beijing, China.*

[3] Matt Rizzo, Brian Clark, Brian Yurconis, Jelena Nicolic, June21, 2002. *Intelligent Ground vehicle Competition Design Project 2002*, Oakland University.

[4] George E. Dieter, (2000) *Engineering Design* (3rd edition); McGraw-Hill.

[5] RS Catalogue april 2001, 2002.

[6] Dr. Devdas Shetty, Richard A. Kolk, (1997) *Mechatronics System Design*; PWS Publishing Company.

[7] Compumotor catalogue, 1894-1/US ,Parker Automation – Linear actuator.

[8] http://www.motionshop.com - linear stepper motor

[9] http://www.steppercontrol.com – controller

[10] http://www.aositilt.com/Ez316.htm - angle detector

[11] "Sensor" Volume 168 (monthly magazine). July, 2000.

[12] www.compumotor.com – wiring connection and linear actuator data sheet.

[13] RS Data Sheet, March 1997, 425-6229.

[14] RS Data Sheet , March 1997, 232-3939

[15] Compumotor catalogue, Engineering Reference (8000-4 USA), Parker Automation – Sizing and Selection Process.

[16] http://www.ni.com – National instrument (LabVIEW)

[17] R.C Hibler , (1999) *Engineering Mechanics: Dynamics*, SI edition ;Prentice Hall

[18] William D. Callister,(1999) JR, *Material Science and Engineering: An Introduction.* (5th edition); John Wiley & Sons Inc.

[19] Mikell P. Groover, (2001) *Automation, Production systems, and Computer-Integrated Manufacturing*, 2nd edition; Prentice Hall.

[20] Eugene A. Avallone. & Theodore Baumeister, (1999) *Marks Standard Handbook for Mechanical Engineering* (10^{th} edition); McGraw-Hill.

[21] Control Lab manual, 2003, UTP.

[22] Timothy J. Maloney, (2001) *Modern Industrial Electronics*, 4^{th} edition; Prentice Hall.

[23] *Journal of Vision*, 1(3), 135a, March 2002.

[24] Labview Basics 1: Introduction Course manuals. Course Software Version 6.1, May 2002 edition.

[25] Daniel Germann, Oliver Lenord ,"*Modular architecture of a motion controller for quadrupeds applied to the walking robots ALDURO and BISAM*". Mechatronics Laboratory, University of Duisburg-Essen, Lotharstr.

[26] ANNA UNIVERSITY, Chennai-25, Syllabus for M.E (Full Time) Computer Aided Design.

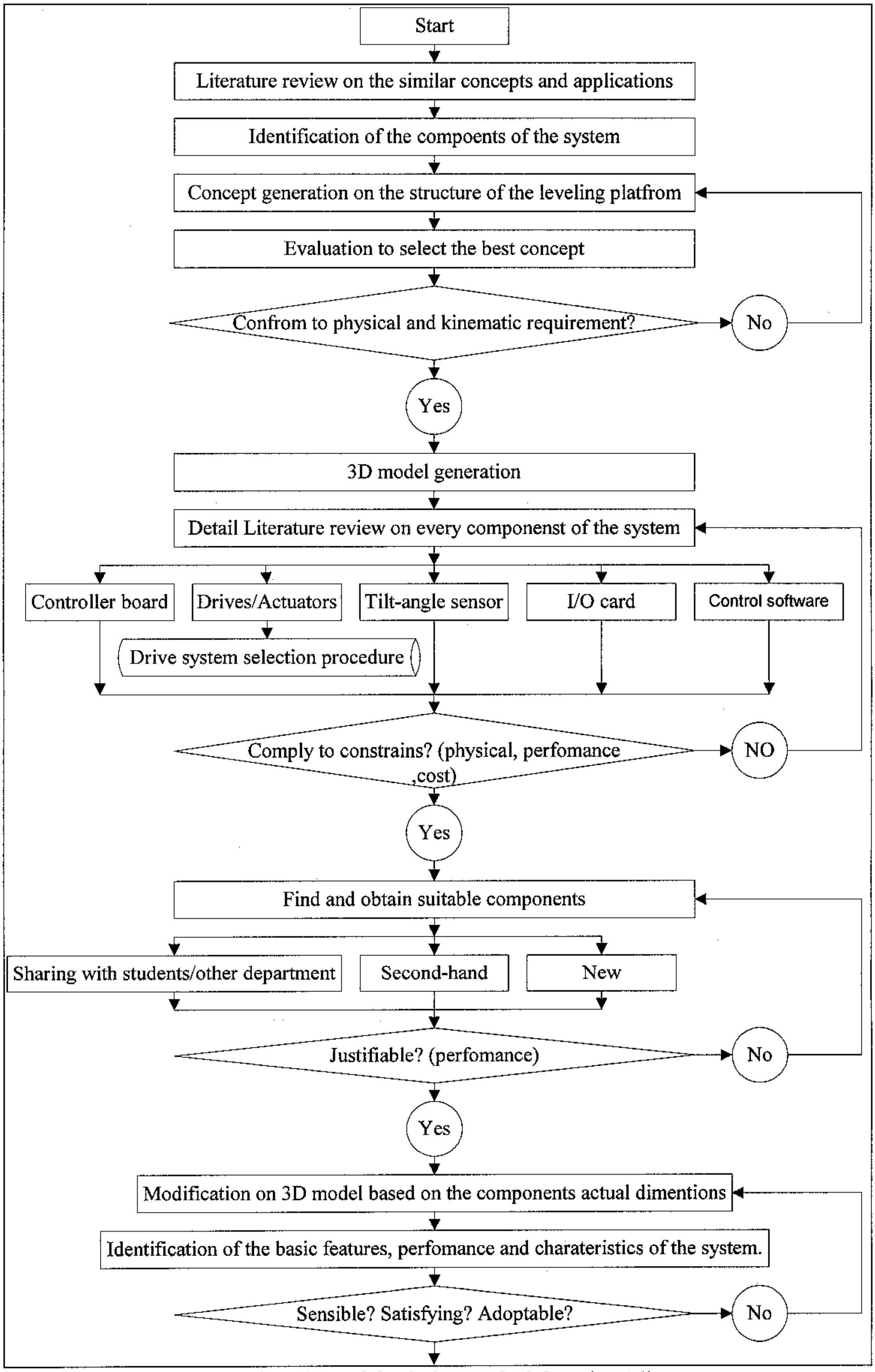
[27] www.robotics.com – applications

[28] http://www.drxproducts.com/drtilt.htm - angle detector

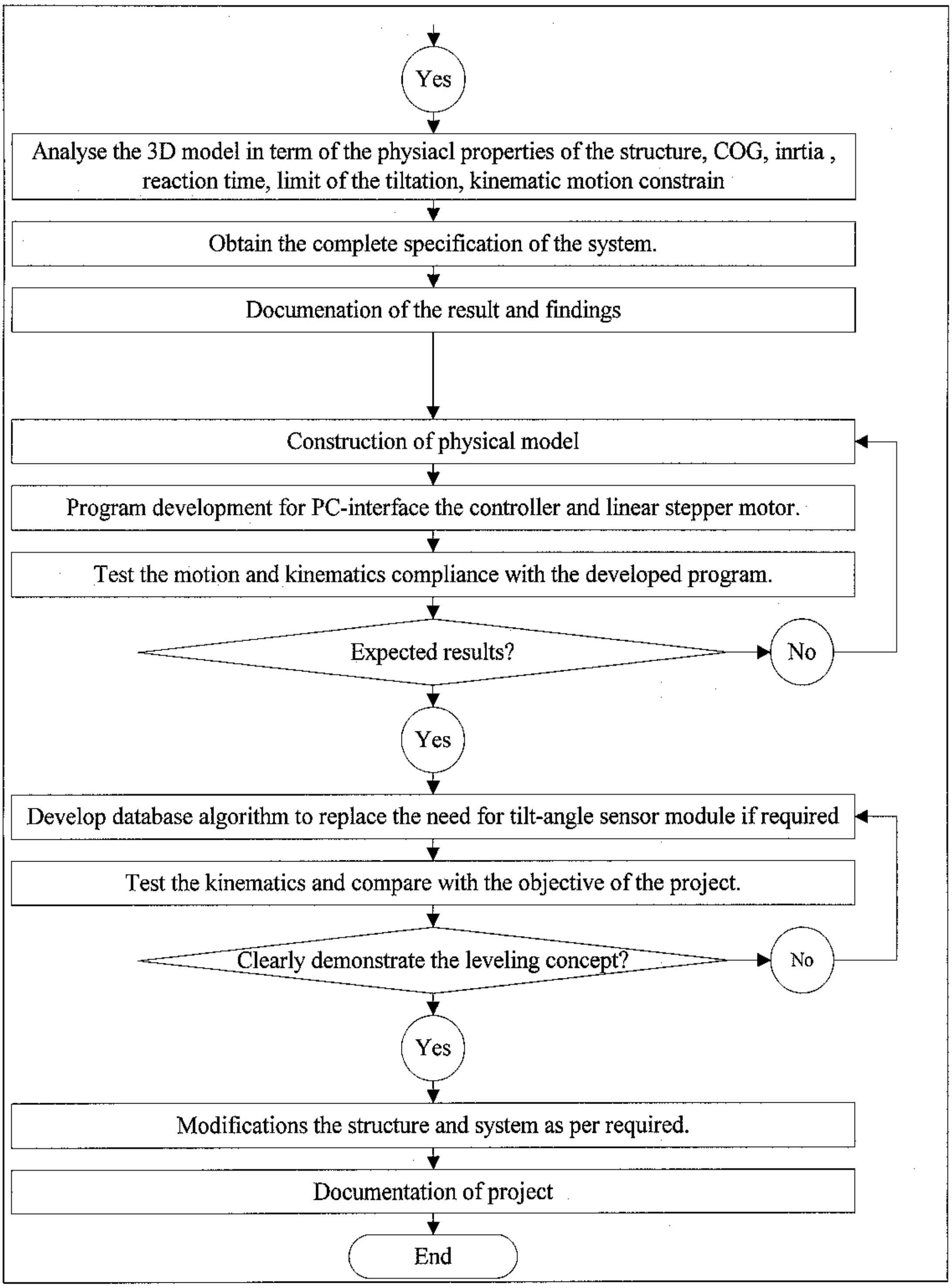
[29] http://www.intellidrives.com – linear stepper motor

APPENDICES

Appendix A: Project progress flow chart

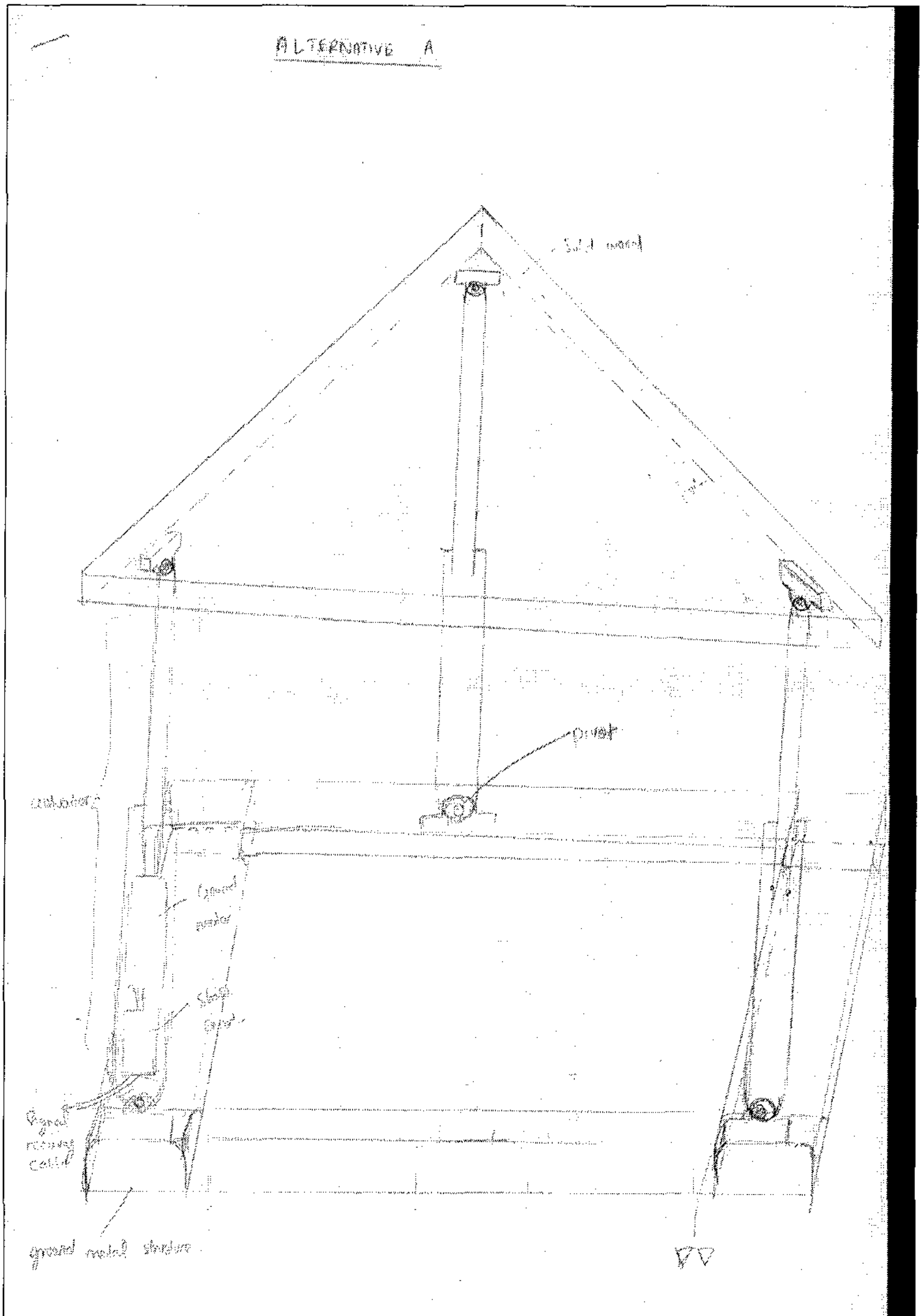


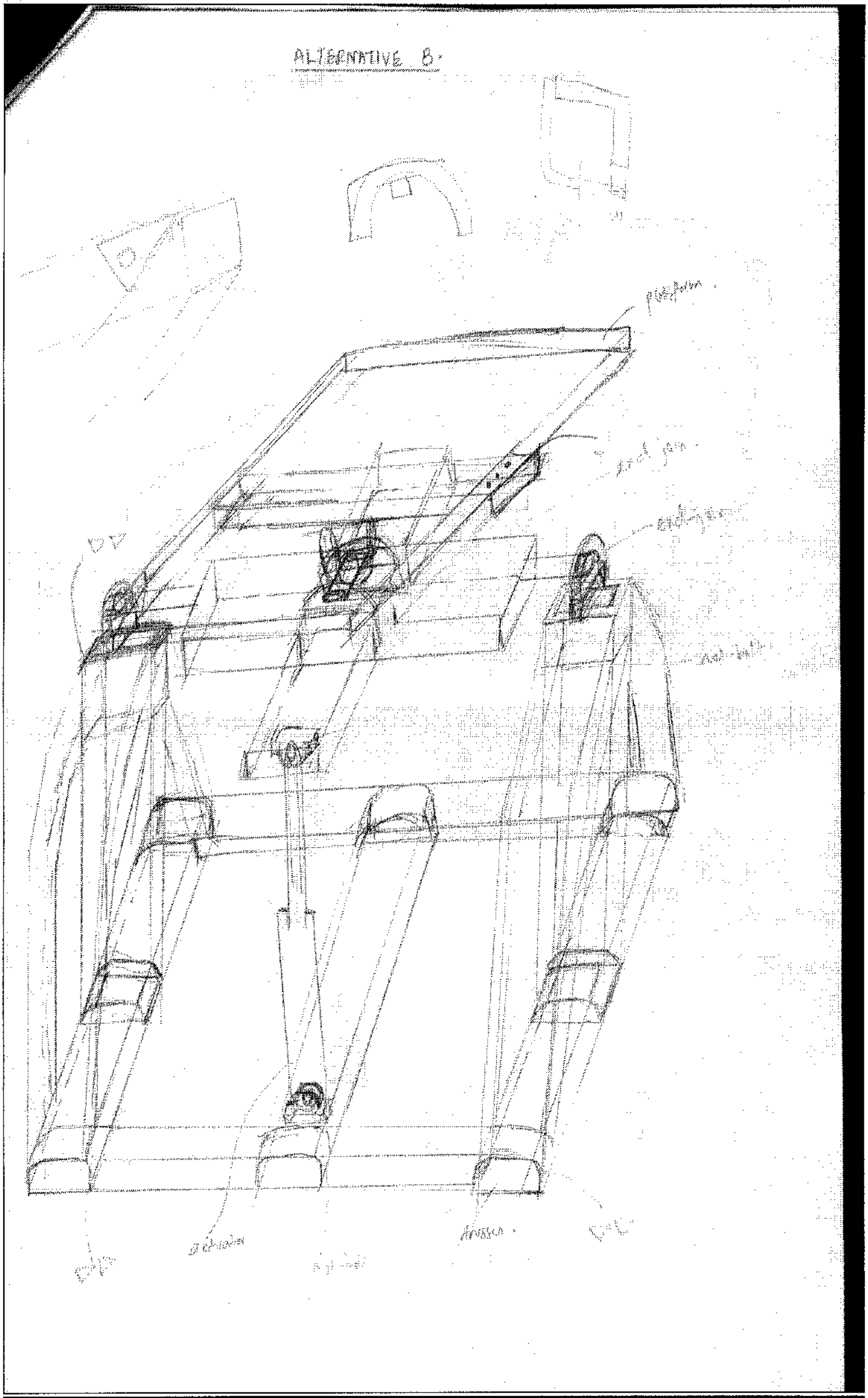
The flowchart of the design methodology (part 1).



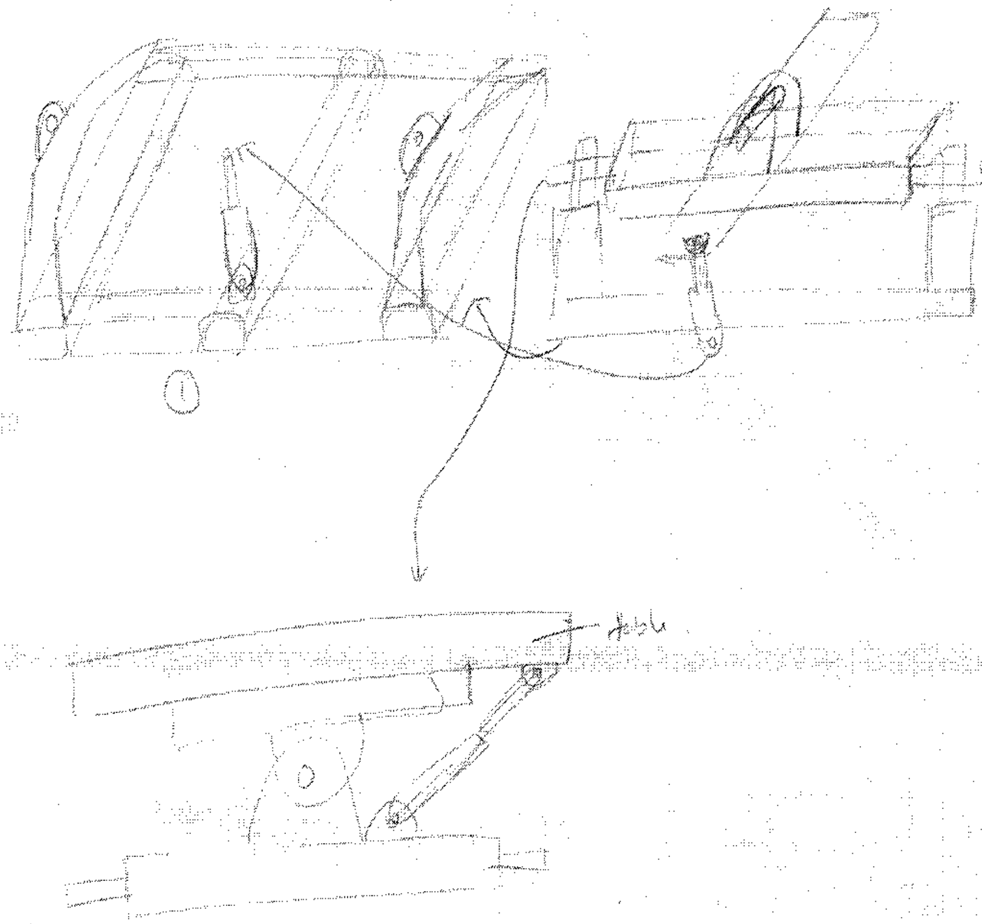
The flowchart of the design methodology (part 2)

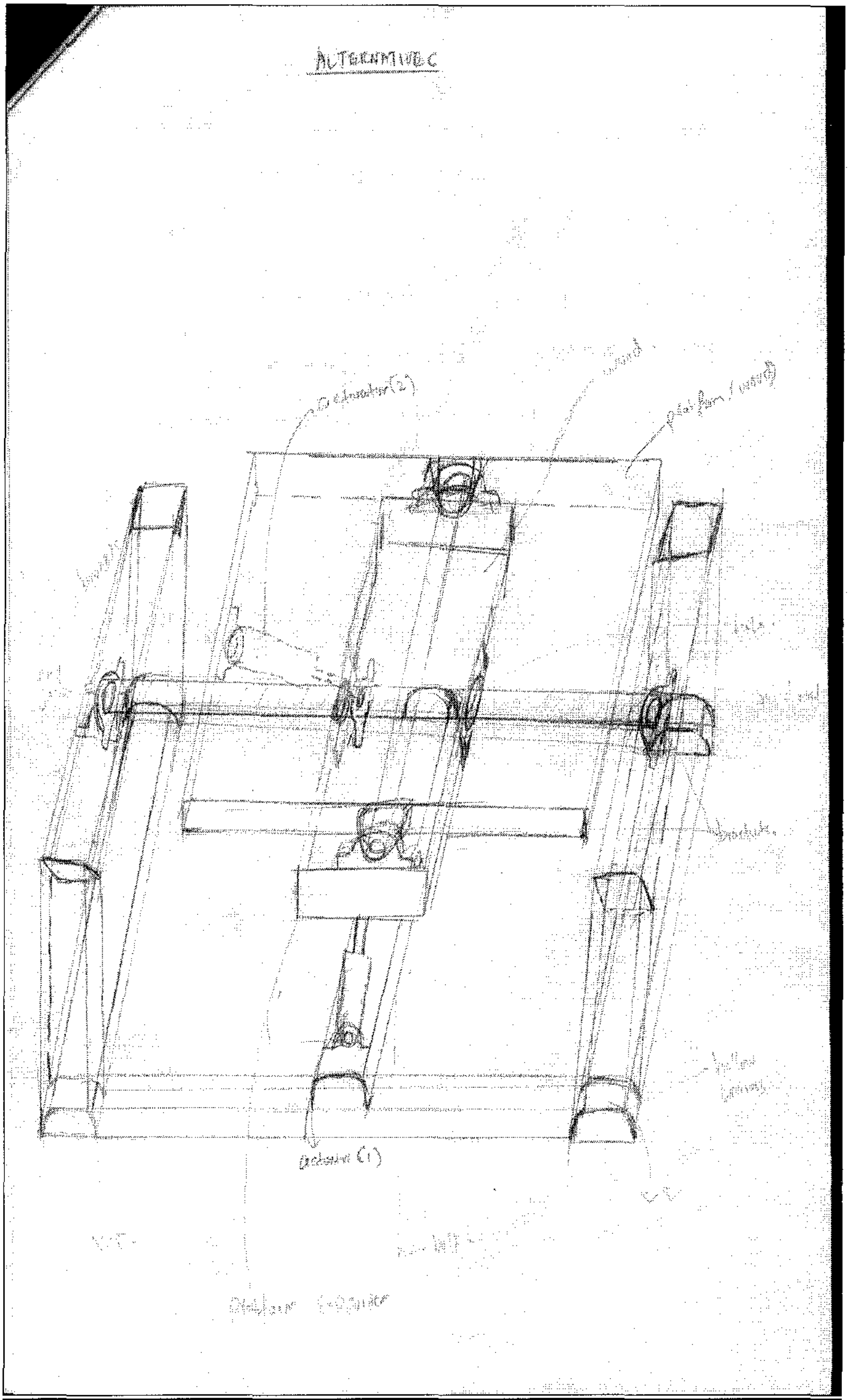
Appendix B: Sketches of the conceptual designs




ALTERNATIVE B.

ASSEMBLY DRAWINS FOR ALTERNATIVE B




ALTERNATIVE C
actuator (2)
platform (wood)
actuator (1)

ALTERNATIVE 'D'

- The structure is the same as appears for ALTERNATIVE B, but using rotary actuator instead of linear-type actuator.

ALTERNATIVE 'E'

- The structure is the same as appears for ALTERNATIVE C, but using rotary actuator instead of linear-type actuator.

Appendix C: Screening evaluation of the alternatives (Pugh concept selection method)

Criteria	Concepts				
	A (3 linear actuator) DOF-3	B (2 linear actuator) DOF-6	C (2 linear actuator) DOF-6	D (2 rotary actuator) DOF-6	E (2 rotary actuator) DOF-6
Fewest components	D A T U M	-	-	-	-
Availability of the components.		S	S	S	S
Ease of assembling or making.		+	+	+	+
Cost of components.		+	+	+	+
The kinematics of the system Comply to DOF=6		+	+	+	+
Load handling capacity		+	+	S	S
Fast action.		S	S	S	S
The precision of actuation for tilting minimum angle per pulse or step angle.		S	S	-	-
Ease of PC-interface with components.		S	S	S	S
∑+		4	4	3	3
∑-		1	1	2	2
∑S		4	4	4	4
Net score		3	3	1	1
Ranking		2	1	3	4

Legend:

Symbol/key	Description
+	The concept has better value in the particular criteria compare to the reference or datum design concept.
-	The concept has lesser value in the particular criteria compare to the reference or datum design concept.
S	The concept have approximately about the same value in the particular criteria compare to the reference or datum design concept.
∑+	summation of positive signs
∑-	summation of negative signs
Net score	(∑+) - (∑-)

Appendix D: Load capacity calculation for actuators

(Please refer to technical drawing for identification)

<u>Total weight of the structure</u>

Total volume = 5967500.75 ≈ 6000000 mm^3
Approx. Average density = 7.5 g/cm^3
Total mass = 6000 cm^3 × 7.5 g/cm^3 = 45 kg

<u>Actuator 1 (bottom)</u>

Total effective load volume = 5106760.68 ≈ 5000000 mm^3
Average density of the material = 6.0 g/cm^3
Total load mass = (5000 cm^3) × 6.0 g/cm^3 = 30 kg
Factor (due to actuator position offset from centroid) = 0.5
Effective load: 0.5 × 30 kg = 15 kg ≈ 150 N
Additional load range on the table = 10 kg = 100N

Actual Payload or linear force = 25 Kg = 250 N

<u>Actuator 2 (top)</u>

Total effective load volume = 115720.5 ≈ 120000 mm^3
Average density of the material = 7.0 g/cm^3
Total load mass = (120 cm^3) × 7.0 g/cm^3 = 0.84 kg
Factor (due to actuator position offset from centroid) = 0.5
Effective load: 0.5 × 0. 84kg = 0.42 kg ≈ 4.2 N
Additional load range on the table = 10 kg = 100N

Actual Payload or linear force = 10.42 Kg = 104.2 N

Appendix E: Mass properties and the formulas

Mass properties for solids	
Mass property	**Description**
Mass	The measure of inertia of a body.
Volume	The amount of 3D space that a solid encloses.
Bounding box	The diagonally opposite corners of a 3D box that encloses the solid.
Centroid	A 3D point that is the center of mass for solids.
Moments of inertia	The mass moments of inertia, which is used when computing the force required to rotate an object about a given axis, such as a wheel rotating about an axle. The formula for mass moments of inertia is $mass\_moments\_of\_inertia = object\_mass * radius_{axis}^2$ Mass moments of inertia unit is mass (grams or slugs) times the distance squared.
Products of inertia	Property used to determine the forces causing the motion of an object. It is always calculated with respect to two orthogonal planes. The formula for product of inertia for the *YZ* plane and *XZ* plane is $product\_of\_inertia_{YZ,XZ} = mass * dist_{centroid\_to\_YZ} * dist_{centroid\_to\_XZ}$ This *XY* value is expressed in mass units times the length squared.
Radii of gyration	Another way of indicating the moments of inertia of a solid. The formula for the radii of gyration is $gyration\_radii = (moments\_of\_inertia/body\_mass)^{1/2}$ Radii of gyration are expressed in distance units.
Principal moments and *X,Y,Z* directions about centroid	Calculations derived from the products of inertia and that have the same unit values. The moment of inertia is highest through a certain axis at the centroid of an object. The moment of inertia is lowest through the second axis that is normal to the first axis and also passes through the centroid. A third value included in the results is somewhere between the high and low values.

Appendix F: Basic equations to analyze the rotational to linear motion using stepper motor.

$\boldsymbol{\alpha = 360 / n_s}$
α = stepping angle(degree)
n_s = number of steps/revolution of motor. Also referred as resolution

$\boldsymbol{A_m = n_p \alpha}$
A_m = angle of motor shaft rotation (degrees)
n_p = number of pulses received by the motor

$\boldsymbol{A = n_p \alpha / r_g}$
A = angle of lead screw/ball screw rotation (degrees)
r_g = gear ratio

$\boldsymbol{x = pA/360}$
x= linear displacement of the actuator OR linear drives
p = pitch of the lead screw/ball screw

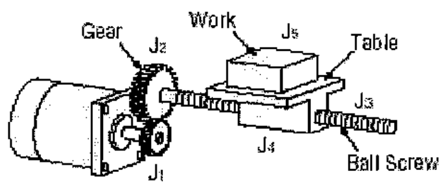
$\boldsymbol{n_p = 360 x r_g / p\alpha}$

$\boldsymbol{N = 60 f_p / n_s r_g}$
N= lead screw or ball screw rotational speed (rev/min)
f_p= pulse train frequency (Hz, pulses/sec)

$\boldsymbol{v_t = Np}$
v_t = linear displacement speed (mm/min)

Resolution of the drive system

Ball screw



$$\Delta l = \frac{P_B}{360^\circ} \cdot \frac{\theta s}{i} \text{ [inch/step]} \cdots\cdots (4)$$

$$P_B = \frac{360^\circ \cdot \Delta l \cdot i}{\theta s} \text{ [inch/rev]} \cdots\cdots (5)$$

Δl = Resolution (minimum feed) [inch/step]
Δl_o = Unit of movement at final step [inch/°]
θs = Step angle [°/step]
i = Gear ratio
P_B = Lead pitch [inch/rev]
v = Movement speed [inch/sec]
f = Pulse speed [Hz]
D = Final pulley diameter [inch]
N = Number of pulses [pulse]
l = Movement [inch]
t = Positioning period [sec]

Equation of motion (relative motion)
***Applies to kinematics analysis of pitch and roll*

$r_B = r_A + r_{B/A}$
$V_B = V_A + V_{B/A}$
$a_B = a_A + a_{B/A}$

Appendix G: Selection of motor

Operating pattern

$$\text{Operating pulses } A \text{ [Pulses]} = \frac{\text{(Distance per movement)}}{\text{(Distance per motor rotation)}} \times \begin{matrix}\text{(No. of pulses} \\ \text{required for} \\ \text{1 motor rotation)}\end{matrix}$$

$$= \frac{l}{lrev} \times \frac{360°}{\theta_s} \qquad \theta_s : \text{Step angle}$$

$$\text{Operating pulse speed } f_2 \text{ [Hz]} = \frac{\text{Number of operating pulses [pulses]}}{\text{Positioning period [sec]}}$$

$$= \frac{A}{t_0}$$

Calculating required torque

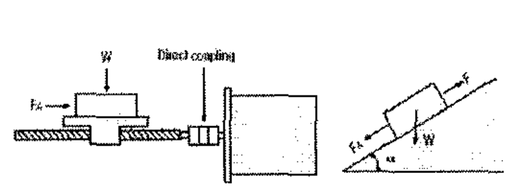
Acceleration torque

$$\text{Acceleration torque } T_a \text{ [oz-in]} = \frac{\text{Inertia of rotor [oz-in}^2\text{]} + \text{Total inertia [oz-in}^2\text{]}}{\text{Weight acceleration [in./sec}^2\text{]}} \times \frac{\pi \times \text{Step angle [°]} \times \text{(Operating pulse speed)}^2 \text{[Hz]}}{180° \times \text{coefficient}}$$

$$= \frac{J_a + J_L}{g} \times \frac{\pi \cdot \theta S \cdot f_2^2}{180 \cdot \mathbf{n}}$$

Load torque

Ball screw



$$T_L = \left(\frac{FP_B}{2\pi\eta} + \frac{\mu_0 F_0 P_B}{2\pi}\right) \times \frac{1}{i} \text{ [oz-in]} \quad \text{⑳}$$

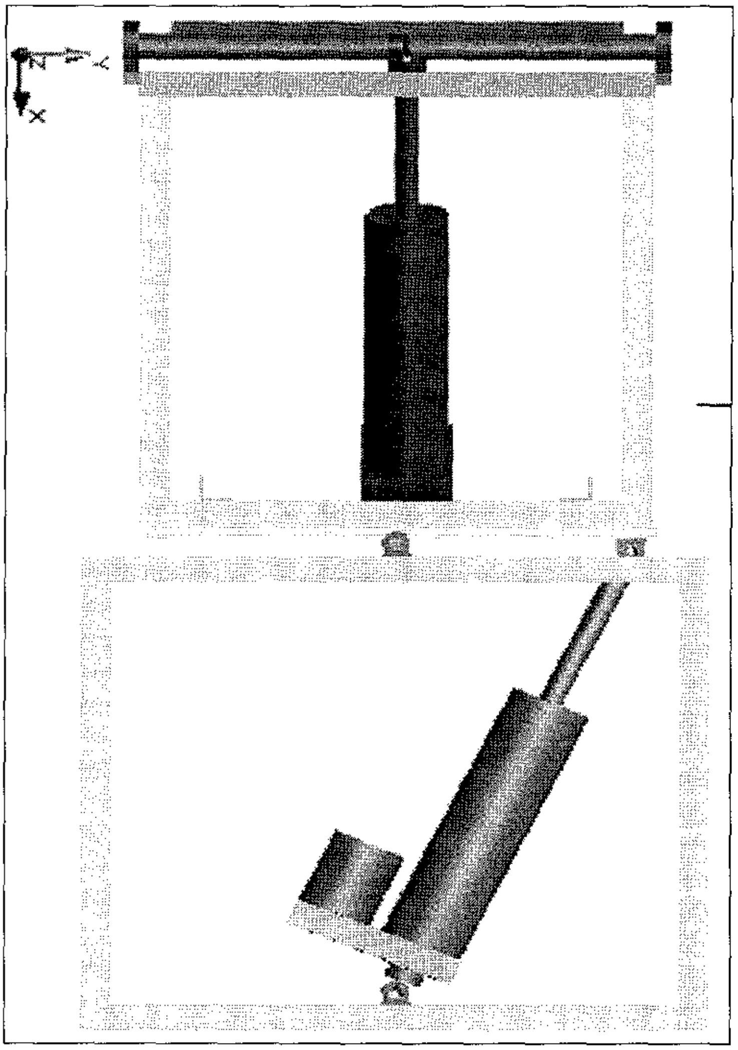
$$F = F_A + W(\sin\alpha + \mu\cos\alpha) \text{ [lb.]} \quad \text{㉑}$$

Required torque

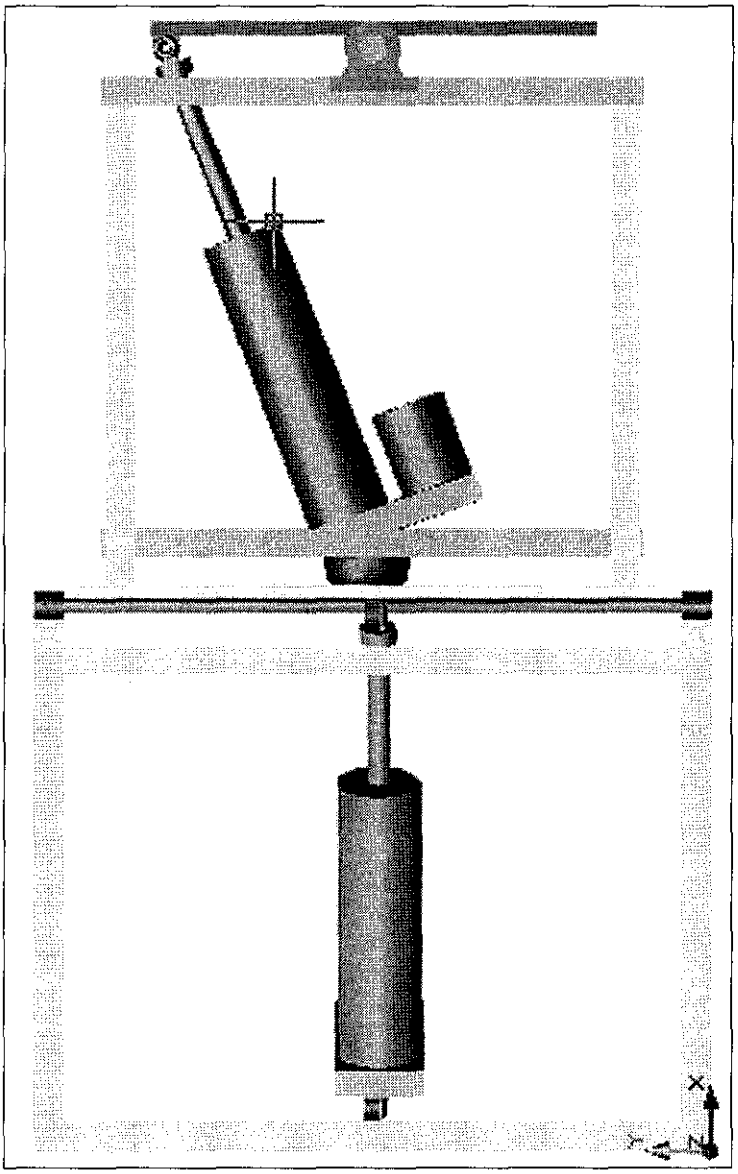
$$\text{Required torque } T_M \text{ [oz-in]} = \left(\text{Load torque [oz-in]} + \text{Acceleration torque [oz-in]}\right) \times \text{Safety factor}$$

$$= (T_L + T_a) \times 2$$

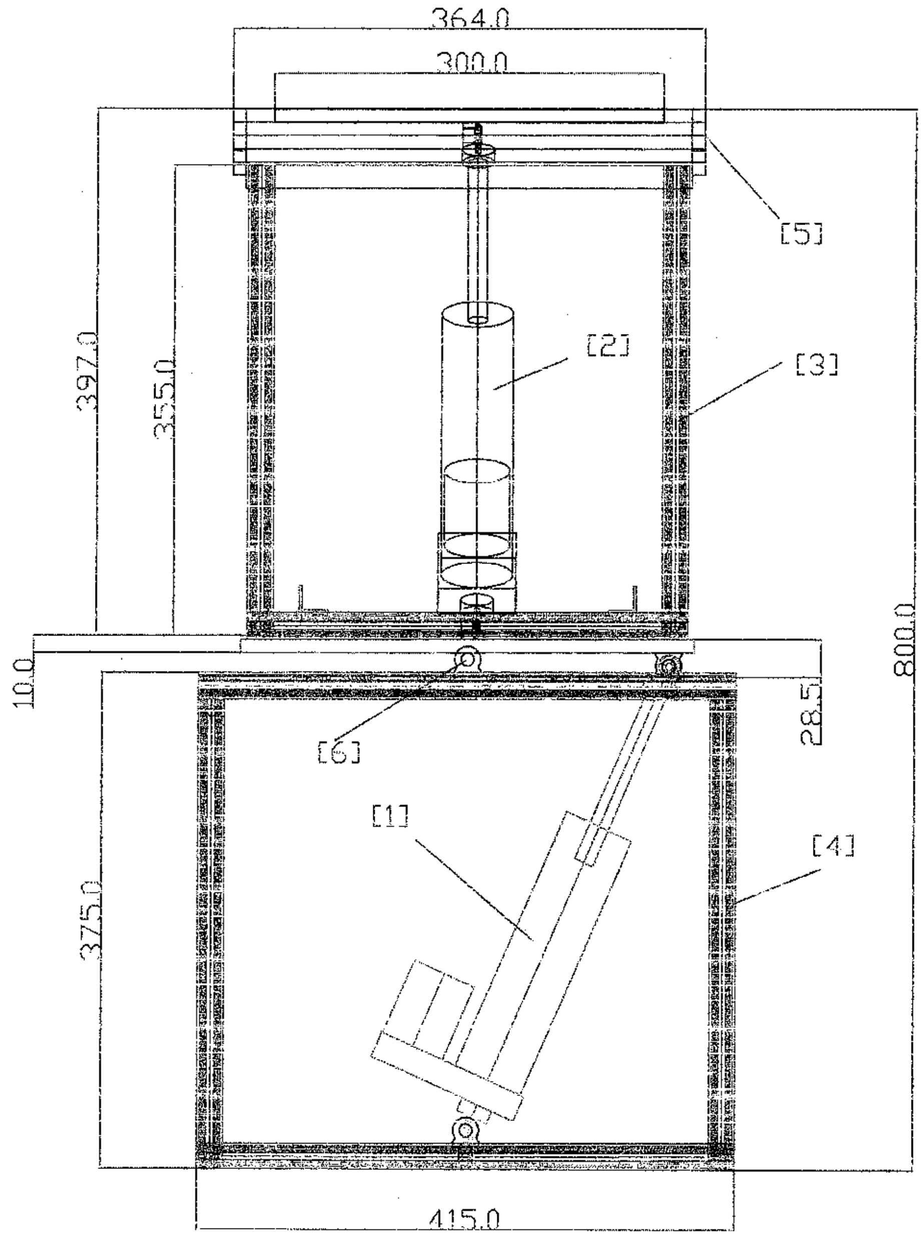
Appendix H: Technical drawings of the structure



Left view of the structure showing the two actuators on top of each other. The addition of the actuators height contributed to high COG.

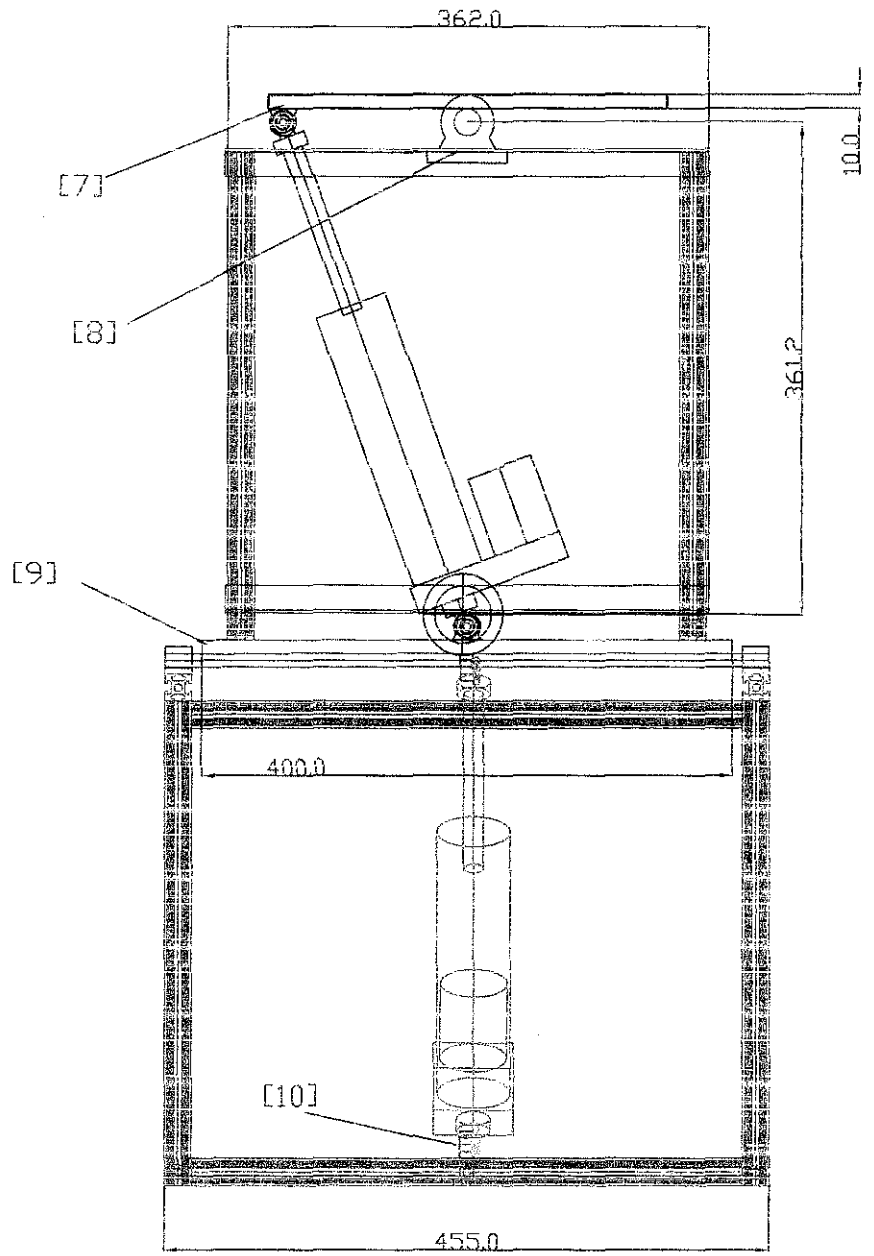


Front View of the structure showing the formation of radius of rotation about the centre of the platform.



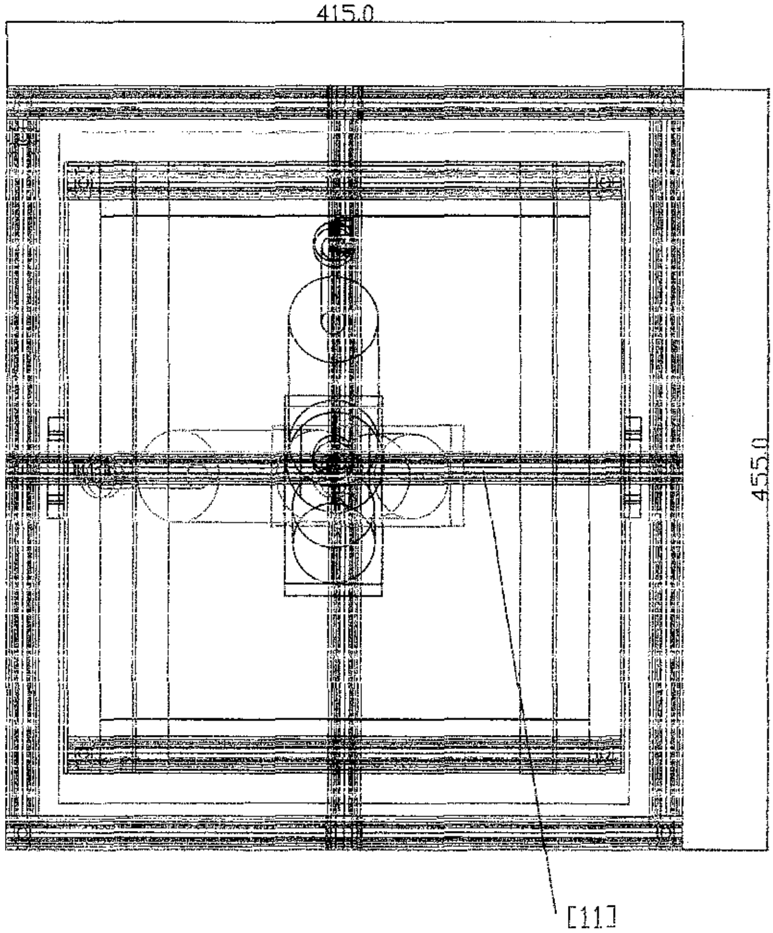
Drawn By	Anwarudin Saidu Mohd
ID	1423
Title	Levelling platform
Date	22/03/04
Reference	Fyp-Front View

Technical drawing showing the dimension of the structure in left view. Numbers are refered in table 5, pg 24.



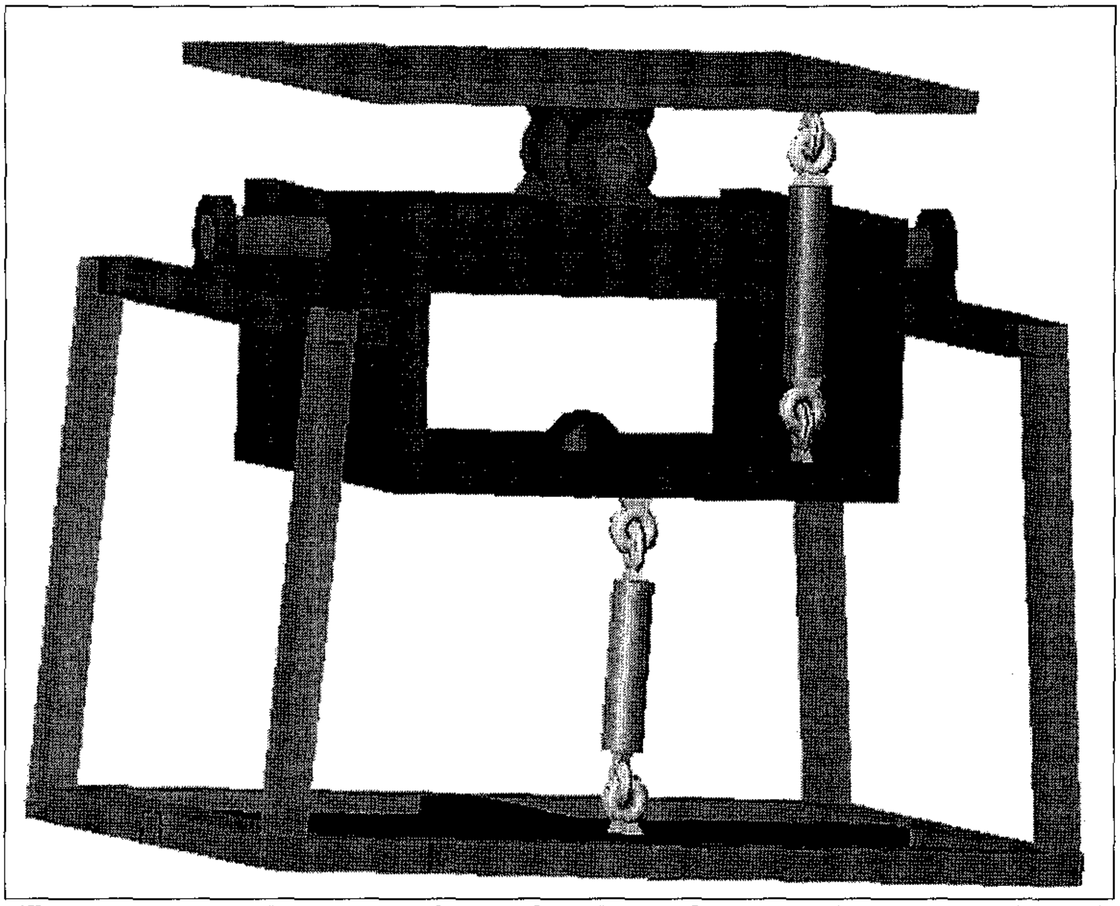
Drawn By	Anwarudin Saidu Mohd
ID	1423
Title	Levelling platform
Date	22/03/04
Reference	Fyp-Left View

Technical drawing showing the dimension of the structure in left view. Numbers are refered in table 5, pg 23.



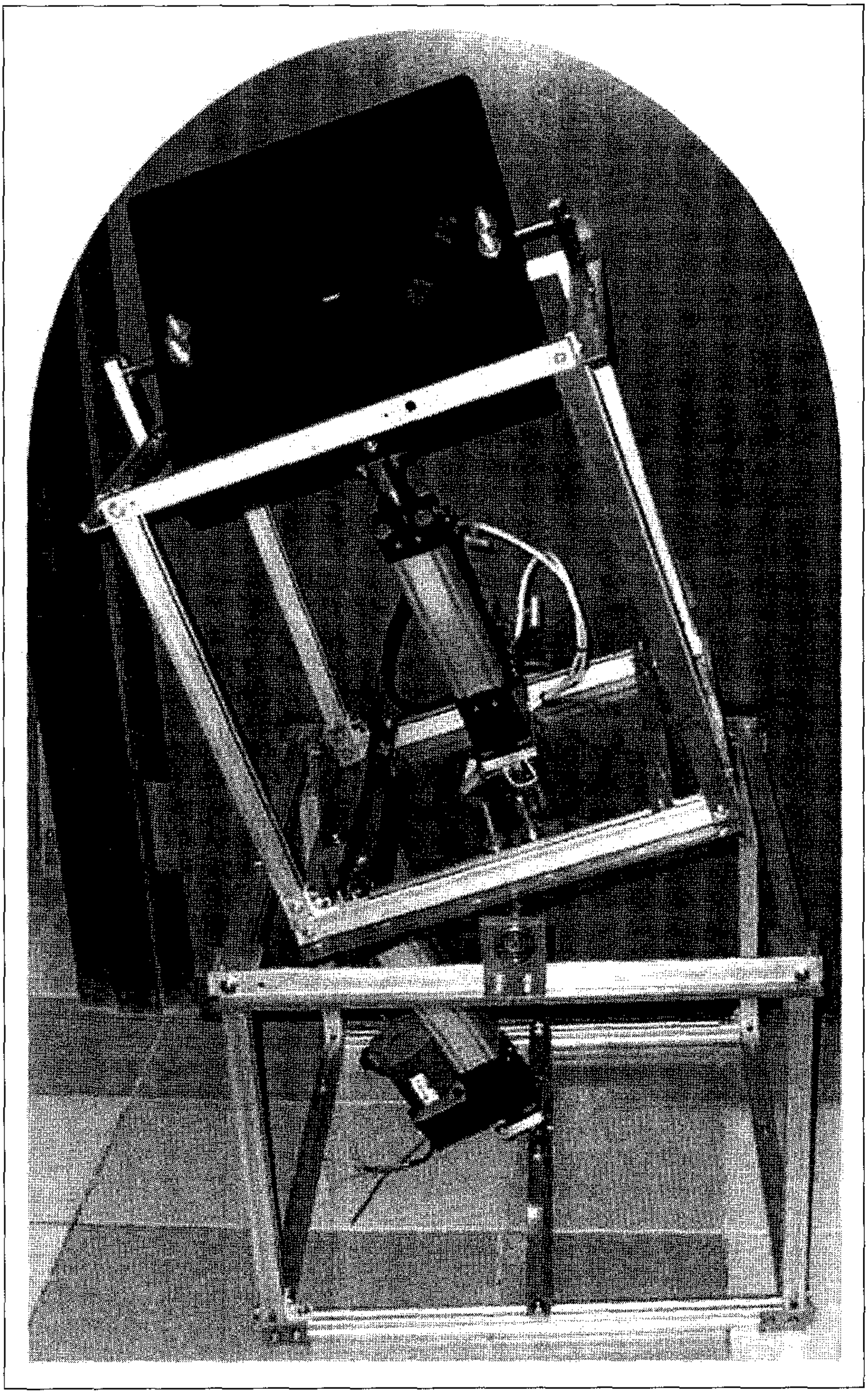
Drawn By	Anwarudin Saidu Mohd
ID	1423
Title	Levelling platform
Date	22/03/04
Reference	Fyp-Top View

Technical drawing showing the dimension of the structure in top view. Numbers are refered in table 5, pg 23.

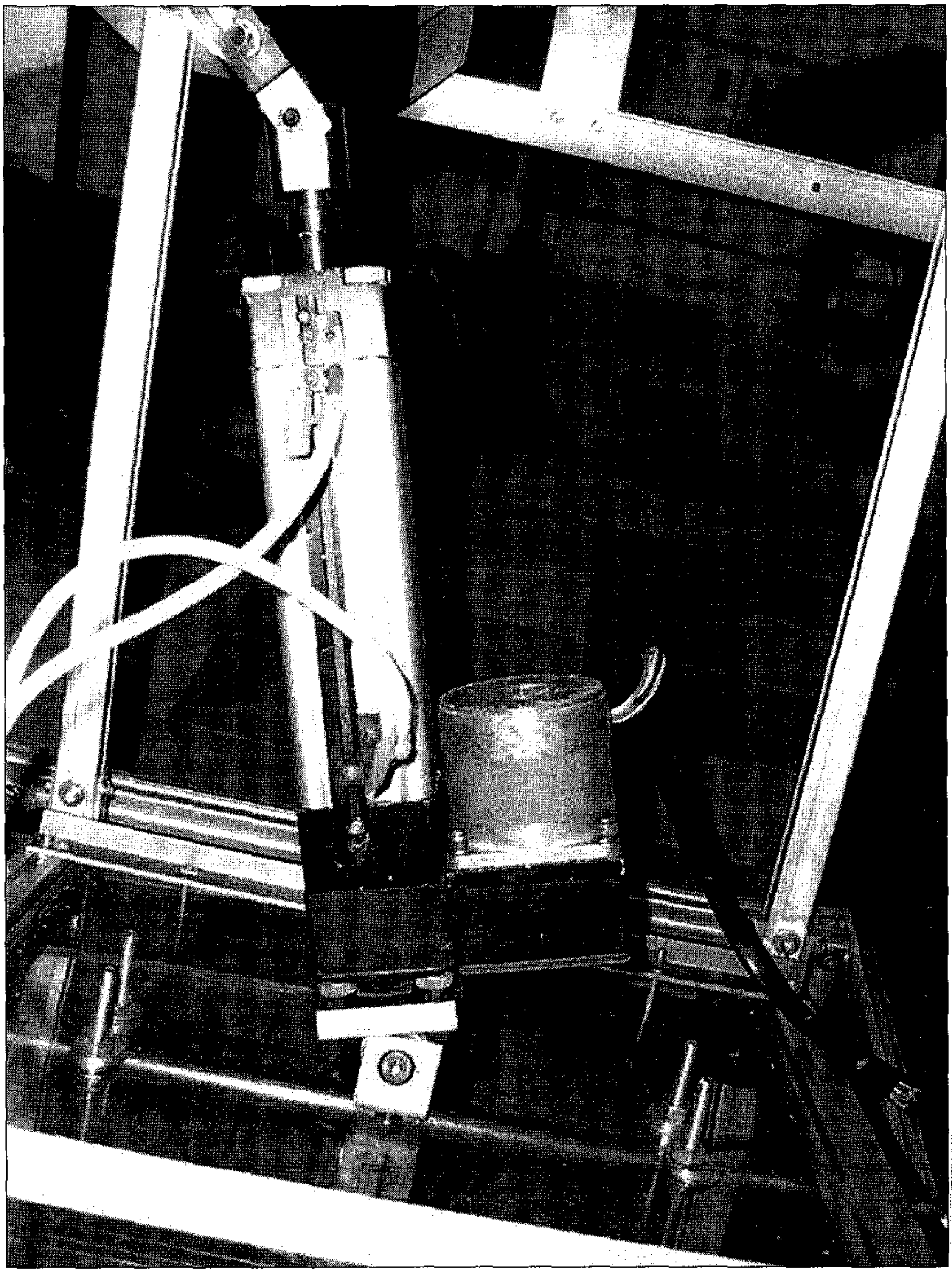


The picture showing the 3D model of the leveling platform before considering the actual size of the components.

Appendix I: The physical appearance of the model



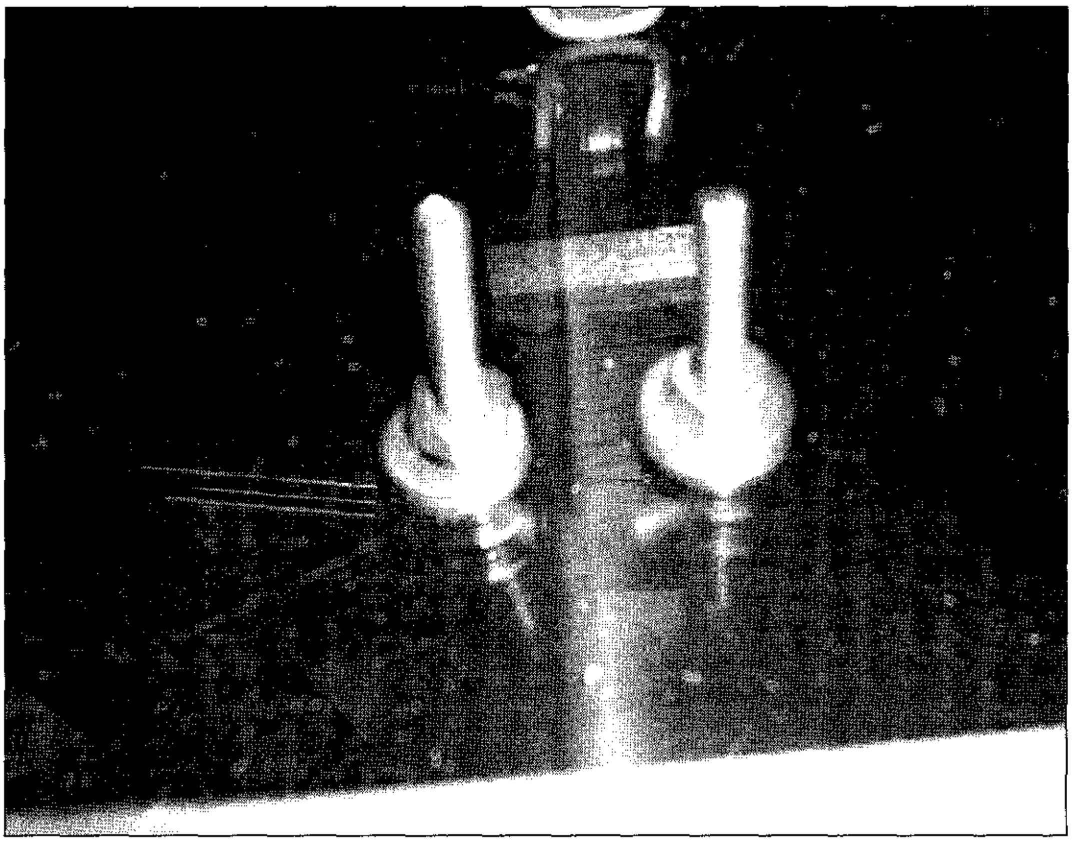
Overall picture the model



The physical appearance of the linear stepper motor


The picture showing the connection between alpha-iron beams

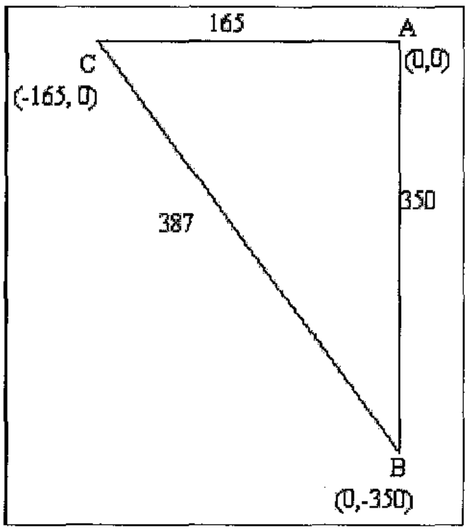

The picture showing how the bearing bracket made from standard size metal plates and the shaft fitted in the bearing.



The picture showing the connection using U-bolt and nut to attach the platform to the steel shaft.

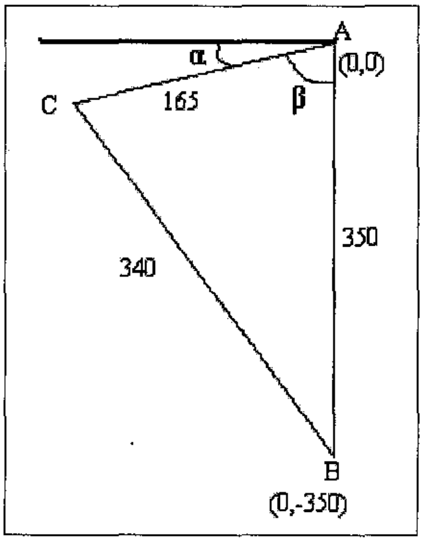
Appendix J: Allowable tilt-range calculation.

<u>Pitch angle range</u>



BC (Solid) = 387 mm

Actuator Solid Length (SL) condition



C $= (-165\cos\alpha, -165\sin\alpha)$

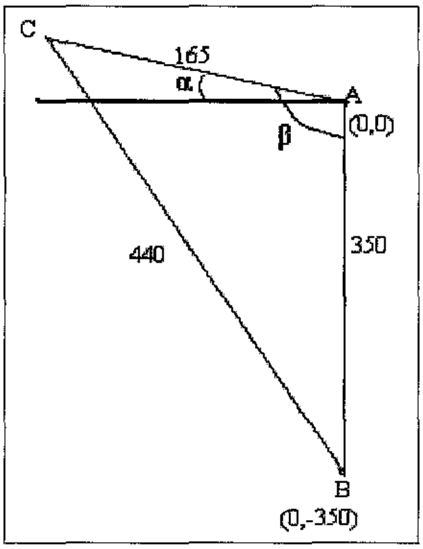
BC = Closed Length = 340mm

$340^2 = 165^2 + 350^2 - 2(165)(350)\cos\beta$

$\beta = 72.81^\circ$

$\alpha = 90 - 72.81 = 17.18^0$

Actuator Closed Length (CL) condition



C $= (-165\cos\alpha, 165\sin\alpha)$

BC = Open Length = 440mm

$440^2 = 165^2 + 350^2 - 2(165)(350)\cos\beta$

$\beta = 112.33^\circ$

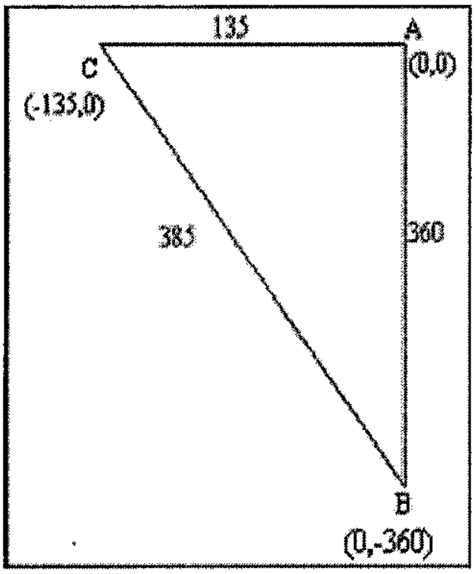
$\alpha = 112.33 - 90 = 22.32^0$

Actuator Closed Length (OL) condition

Therefore the allowable tilt angle range in pitch axis:

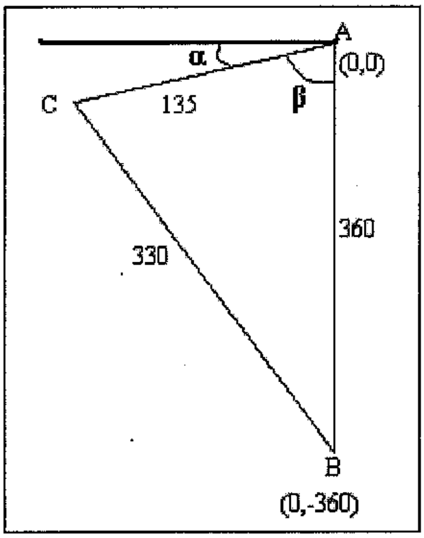
$-17.18^\circ < \mathbf{\Theta_{pitch}} < 22.32^\circ$

Roll angle range



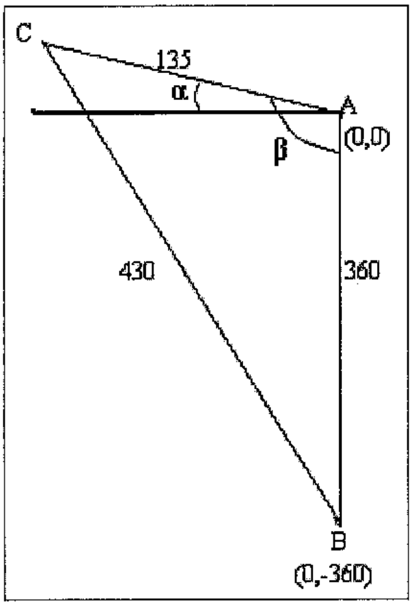
BC (Solid) = 385 mm

Actuator Solid Length (SL) condition



C $= (-135\cos\alpha, -135\sin\alpha)$
BC = Closed Length = 330mm
$330^2 = 135^2 + 360^2 - 2(135)(360)\cos\beta$
$\beta = 66.4^{\circ}$
$\alpha = 90 - 66.4 = 23.6^{0}$

Actuator Closed Length (CL) condition



C $= (-135\cos\alpha, 135\sin\alpha)$
BC = Open Length = 430mm
$430^2 = 135^2 + 360^2 - 2(135)(360)\cos\beta$
$\beta = 112.4^{\circ}$
$\alpha = 112.4 - 90 = 22.4^{0}$

Actuator Closed Length (OL) condition

Therefore the allowable tilt angle range in roll axis:

$-23.6^{\circ} < \boldsymbol{\theta}_{roll} < 22.4^{\circ}$

Appendix K: Performance characteristics of OEM linear motor with OS 22A FEMA 21 size stepper motor.

Parameters	OS22A
Static torque	196 (1.31)
Rotor inertia	1.39 (0.25)
Phase Inductance (mH)	2.8 NA
Detent Torque	7.0 (0.05)
Bearings Information	
Thrust Load	13 (5.9)
Radial Load	20 (9.1)
End Play (Reversing load equal to 1 lb)	0.001 (0.025)
Radial Play (Per 0.5 lb load)	0.0008 (0.02)
Motor Weight	2.5 (1.1)
Certifications	Pending Yes No
Drive Current	
E-AC	NA NA
E-DC	4.8 (3.39) NA

oz-in
(Nm)
oz-in^2
(kg-cm^2)
Series
Parallel
oz-in
(N-m)
lb
(kg)
lb
(kg)
in
(mm)
in
(mm)
lb
(kg)
UL recognized
CE (LVD)
CE (EMC & LVD)
Apk (Arms)
Series
Parallel
Series
Parallel
Series
Parallel
Series
Parallel
Series
Parallel

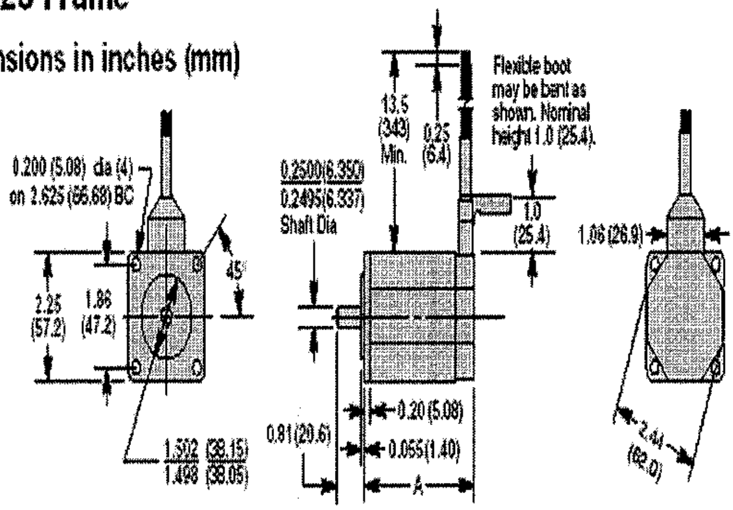
Appendix L: Dimensions of the OS 22A stepper motor.

Dimensions

OS Dimensional Drawings

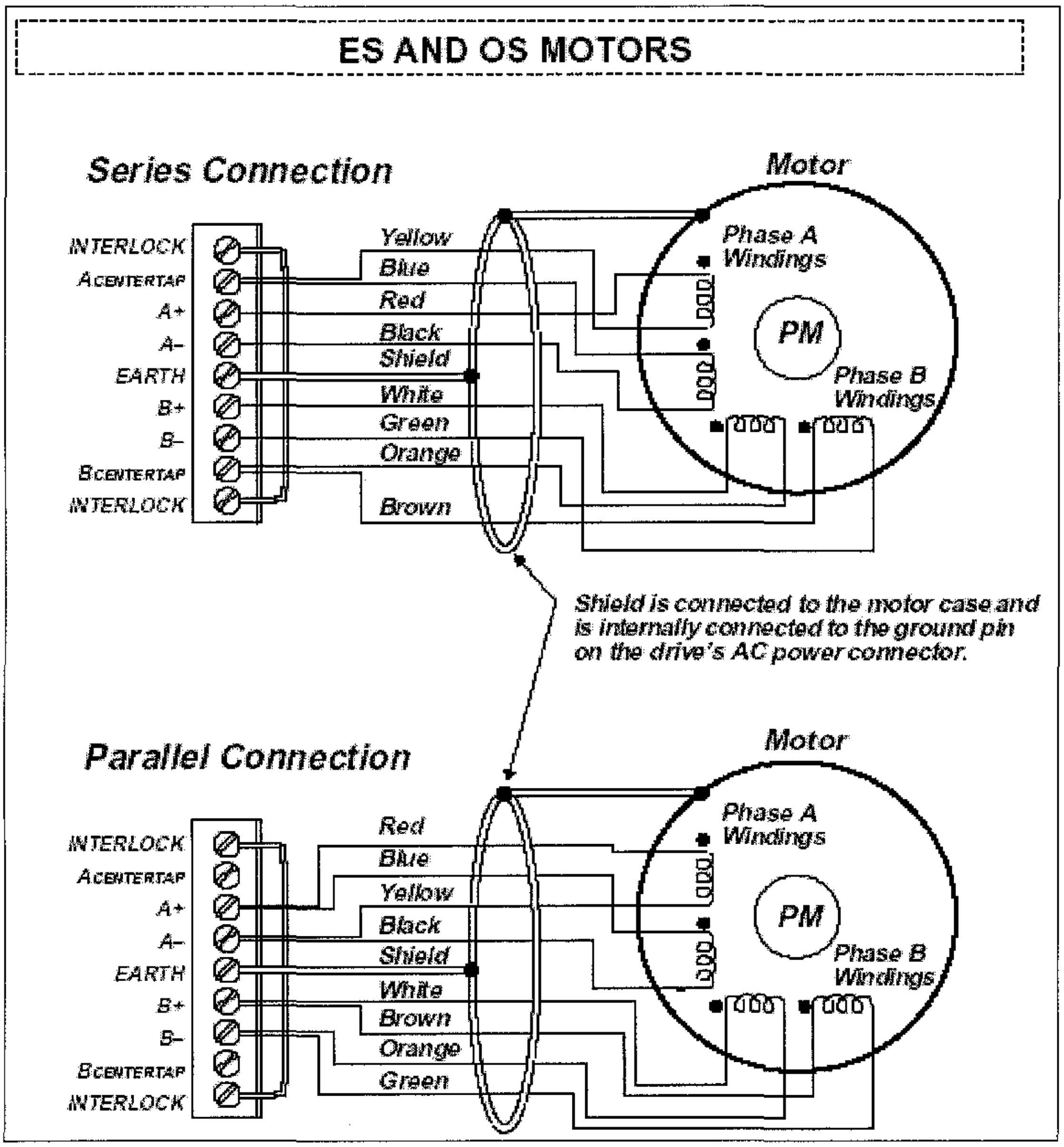
Size 23 Frame

Dimensions in inches (mm)



Model		A
OS2H	(OEM57-40)	1.60 (40.6)
OS21	(OEM57-51)	2.06 (52.3)
OS22	(OEM57-83)	3.10 (87.7)

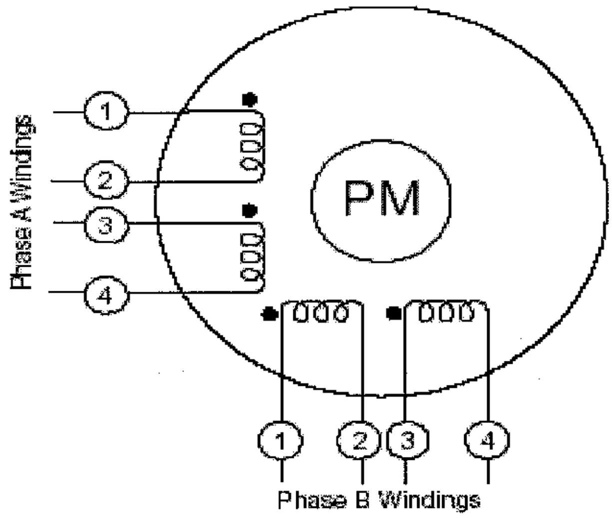
Appendix M: Connection diagram for OEM Linear series motor using 22A stepper motor



Appendix N: Instruction on connecting 8-lead motor (applicable for OS 22A, PARKER Compumotor)

8-Lead Motor

Because of the complexity involved in phasing an 8-lead motor, you must refer to the manufacturer's motor specification document. Using the manufacturer's specifications, label the motor leads as shown in the next drawing.



8-Lead Motor – Labeling the Leads

You can configure the 8-lead motor in series or parallel.

Series Configuration Use the following procedure for series configurations.

1. Connect A2 & A3 together and relabel this common point **A-CT**.
2. Connect B2 & B3 together and relabel this common point **B-CT**.
3. Relabel the A1 lead **A+**.
4. Relabel the A4 lead **A–**.
5. Relabel the B1 lead **B+**.
6. Relabel the B4 lead **B–**.
7. Proceed to the *Terminal Connections* section below.

Parallel Configuration Use the following procedure for parallel configurations.

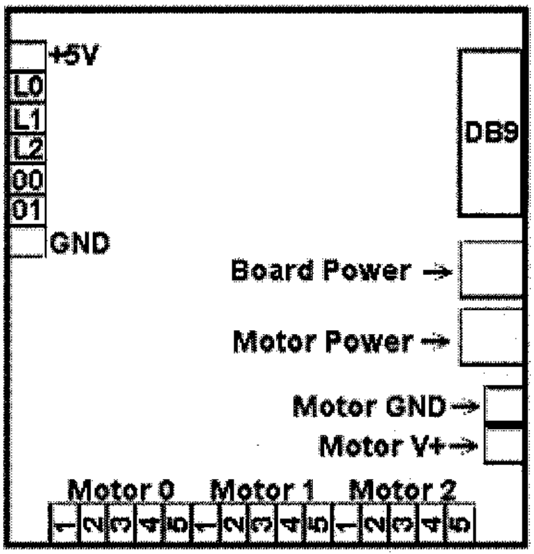
1. Connect motor leads A1 & A3 together and relabel this common point **A+**.
2. Connect motor leads A2 & A4 together and relabel this common point **A–**.
3. Connect motor leads B1 & B3 together and relabel this common point **B+**.
4. Connect motor leads B2 & B4 together and relabel this common point **B–**.
5. Proceed to the *Terminal Connections* section below.

Appendix O: Connection diagram for the controller board

S100SMC Setup Information

The following diagram shows the various connections found on the S100SMC, and below this image are instructions for connecting the power supplies, DB9 cable, and any inputs or outputs to the board.

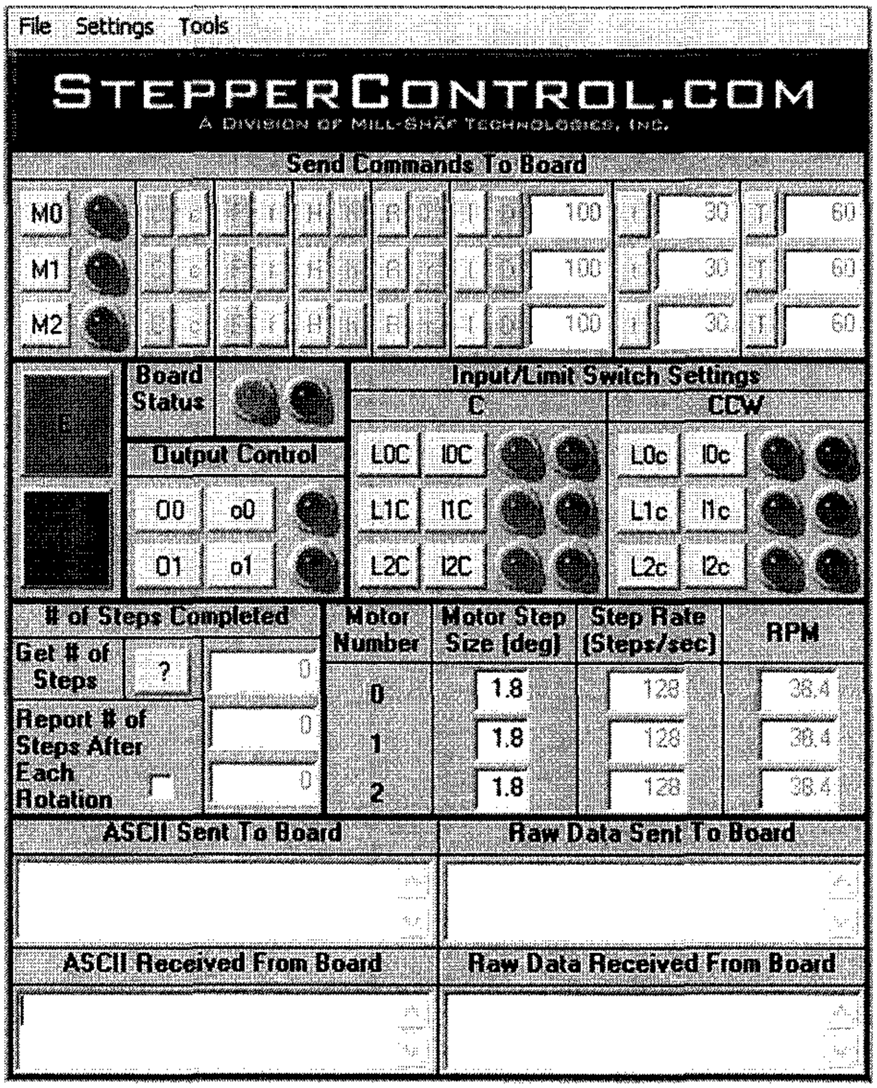
S100 Connections



Power and Cable Connections

The S100SMC connects to the computer or other controlling device through a DB-9 serial cable. Only the Rx, Tx, and Gnd pins are used, in case there is a need to develop own custom cable. Otherwise, any standard DB-9 male-female cable should work. The DB-9 connector (shown in Figure 1) on the S100SMC is located beside the two power jacks. Simply push the male end of the serial cable onto the DB-9 connector and connect the other end of the cable to the communications port of the computer. The standard control software for the S100SMC allows to select any comm. port from 1 - 15. The S100SMC operates off of two separate power supplies--the one that is included in order (shipping to countries with 120V, 60Hz power grids) powers the logic circuitry on the boards, and a second supply provides the power to the motors. The first power supply has a center-positive, .215" OD, .082" ID barrel plug. Simply push the barrel plug into the connector on the board closest to the DB-9 connector (labeled "Board Power" in Figure 1). The second power supply powers the motors, and it can be either of the barrel plug type and connect to the second plug connector (labeled "Motor Power" in Figure 1) or a power supply with terminal or lead wires that can be connected to the power terminal connector on the S100SMC. If the user is planning to connect the power supply to the power terminal on the corner of the board, connect the ground lead to the terminal labeled "Motor GND" in Figure 1, and connect the positive lead to the terminal labeled "Motor V+" in Figure 1. The S100SMC can handle up to 50V and 5A, but very few motors will require this much power. If user uses a motor that draws a large amount of current or operates at a high voltage, monitor the heat of the motor carefully (all motors run hot, but overheating could cause failure) and connect heat sinks to the TIP 120 chips for added heat dissipation.

Appendix P: Controller software provided by the www.steppercontrol.com that can control three motors simultaneously.

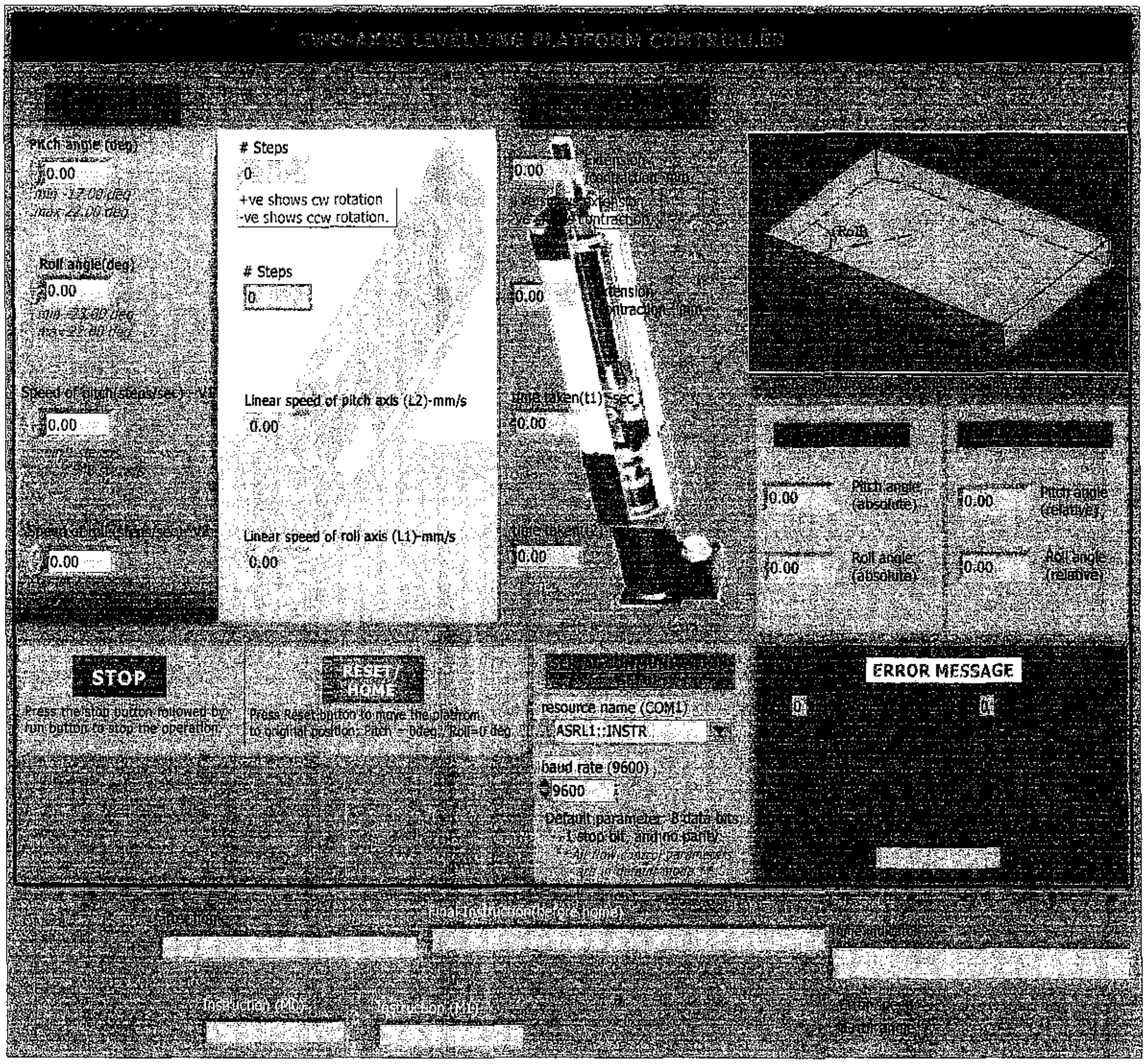


Connector Pane

Appendix G: Complete documentation of the LabVIEW controller program

Pich & Roll (sub-VI).vi

Front Panel



Controls and Indicators

Pitch angle (deg)
This is the tilt-angle in pitch direction measured in degrees angle.

The allowable range for the angle is btwn -17 to 22 degree.

Roll angle(deg)
This is the tilt-angle in pitch direction measured in degrees angle.

The allowable range for the angle is btwn -23 to 22 degree.

Speed of pitch(steps/sec) - V1
This is the speed of the pitch axis in term of the speed of the stepper motor in the embedded in the Linear Actuator.

Speed of roll (steps/sec)- V2
This is the speed of the roll axis in term of the speed of the stepper motor in the embedded in the Linear Actuator.

stop
This button will bring the actuator to halt. Please press the button only in emergency case because the tracing function of the software will be disabled. Therefore the platfrom have to be recalibrated for home

baud rate (9600)
The baud rate required to enable communication with the Steppercontroller card is 9600bps

resource name (COM1)
A string that uniquely identifies the resource to be opened and written to as well as read from. The grammar for the resource name is shown below. Optional string segments are shown in square brackets ([])

Interface	Syntax
VXI	VXI[board]::VXI logical address[::INSTR]
GPIB-VXI	GPIB-VXI[board]::VXI logical address[::INSTR]
GPIB	GPIB[board]::primary address[::secondary address][::INSTR]
Serial	ASRL[board][::INSTR]

The following table shows the default value for optional string segments.

Optional String Segments	Default Value
board	0
secondary address	none

The following table shows examples of address strings.

Address String	Description
VXI0::1	A VXI device at logical address 1 in VXI interface VXI0.
GPIB-VXI::9	A VXI device at logical address 9 in a GPIB-VXI controlled system.

Reset
Pressing this button will reset the platfrom to home position which is equivalent to 0 degree in pitch axis and 0 deg in roll axis.

 # Steps
This the number steps the stepper motor in the actuator has to move in order to cater for the tilt-angle input by the user in pitch axis.

Positive values indicate that, the stepper motor is rotating clockwise and vice vesa.

Extension /contraction- mm
Indication of the motion of the thrust in the actuator in Pitch axis. Wheter it's extending or retracting. Positive angle (pitch or roll) will led to the extension of the thrust whereelse the negative angle led to

 # Steps
This the number steps the stepper motor in the actuator has to move in order to cater for the tilt-angle input by the user in roll axis.

Positive values indicate that, the stepper motor is rotating clockwise and vice vesa

Extension /contraction - mm
Indication of the motion of the thrust in the actuator in roll axis. Wheter it's extending or retracting. Positive angle (pitch or roll) will led to the extension of the thrust whereelse the negative angle led to retraction.

time taken(t1) -sec
This is the time taken in seconds for the actuator to completete the motion in pitch axis as instructed by the software. It's depend on the speed of the actuator and tilt-angle.

Linear speed of pitch axis (L2)-mm/s
The spped of the actuation in pitch axis expresses as linear speed in mm/s. This is the speed after the rotational motion of the stepper motor converted to linear motion by timing belt-ball screw.

time taken(t2)-sec
This is the time taken in seconds for the actuator to completete the motion in roll axis as instructed by the software. It's depend on the speed of the actuator and tilt-angle.

Linear speed of roll axis (L1)-mm/s
The spped of the actuation in roll axis expresses as linear speed in mm/s. This is the speed after the rotational motion of the stepper motor converted to linear motion by timing belt-ball screw.

delay-high(M0)
Equivalent ASCII character "Null"

delay-low (M0)
Equivalent ASCII character "Null"

Direction(M0)

steps-high(M0)

Instruction (M0)

delay-high(M1)
Equivalent ASCII character "Null"

delay-low(M1)
Equivalent ASCII character "Null"

Direction(M1)

steps-high(M1)

steps-low(M1)

Instruction (M1)

Execute/Stop Status
This is the indication on wheter the actuator allowed to function. The alpahbet "S" will stop the actuator from moving. This letter indication might be due to error (i.e user input beyond the limit).

Alphabet 'E' will enable the motion of the actuator as per required.

Pitch angle (absolute)
This is the absolute angle in pitch axis expressed in degree. It should follow the limitation cited in red color text in teh input box. This is the absolute reference on the position of the platfrom.

Roll angle (absolute)
This is the absolute angle in roll axis expressed in degree. It should follow the limitation cited in red color text in teh input box. This is the absolute reference on the position of the platfrom.

Pitch angle (relative)
This is the relative angle in pitch axis expressed in degree. It is the incremental measures of the angle refering to the previoius cycle of instruction.

For example:

cycle 1 (user input) = 10 deg ; relative = 10 deg
cycle 2 (user input) = 20 deg ; relative = 10 deg (increment of 10 from the first cycle)

Roll angle (relative)
This is the relative angle in pitch axis expressed in degree. It is the incremental measures of the angle refering to the previoius cycle of instruction.

For example:

cycle 1 (user input) = 10 deg ; relative = 10 deg
cycle 2 (user input) = 20 deg ; relative = 10 deg (increment of 10 from the first cycle)

Error angle
Indication of the error status in case the of the limitation on the tilt-angle reached. Once the numerical value 1 is shown in the box, the actuator will not move. You may have to re-enter the required fields.

Error speed
Indication of the error status in case the of the limitation on the actaution/retraction speed reached. Once the numerical value 1 is shown in the box, the actuator will not move. You may have to re-enter the

steps-low(M0)

Msg-angle

DBL **Msg-speed**

abc **Final Instruction(before home)**

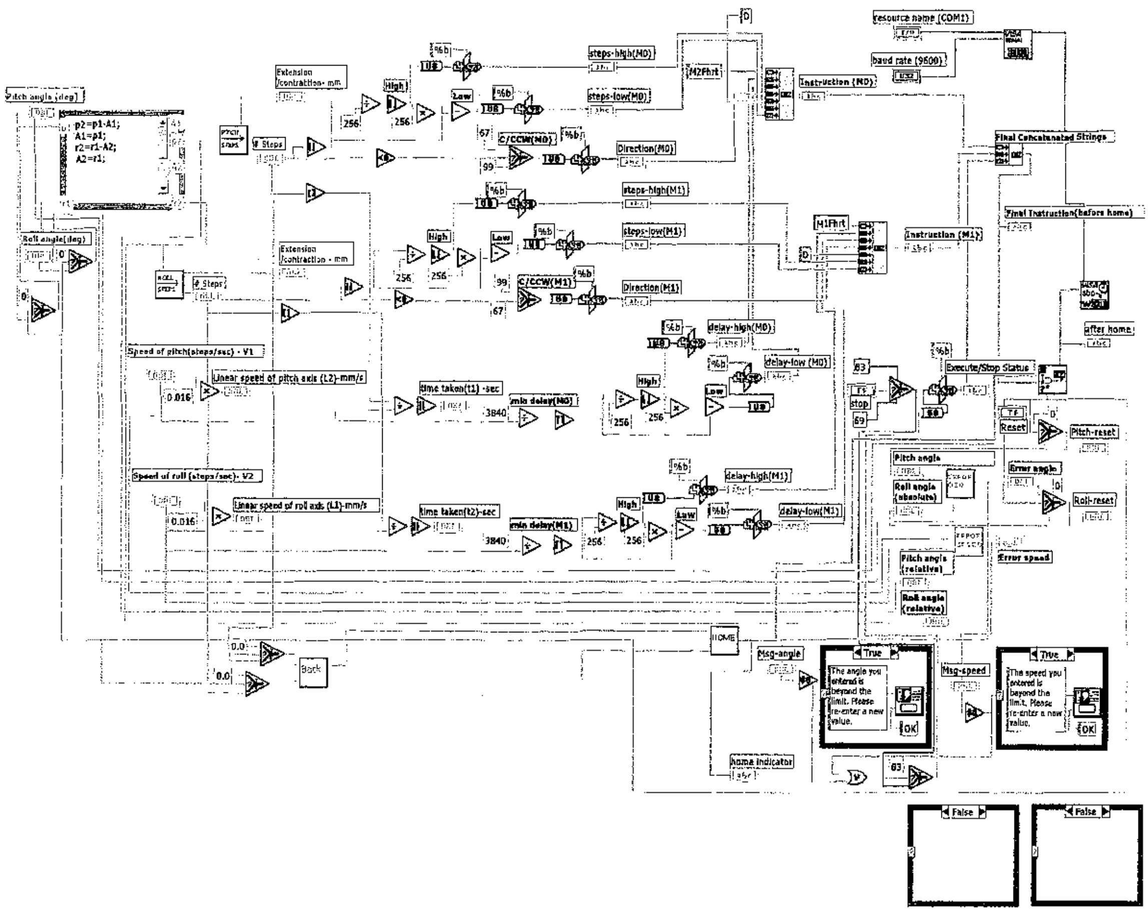
DBL **Pitch-reset**

DBL **Roll-reset**

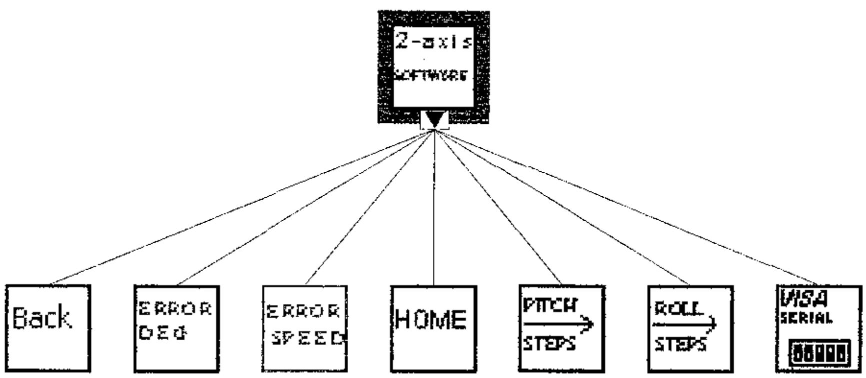
abc **home indicator**

abc **after home**

Block Diagram



Position in Hierarchy



List of SubVIs

roll.vi
D:\Subjects and collection\FYP\Oral Presentation\final\roll.vi

VISA Configure Serial Port
C:\Program Files\National Instruments\LabVIEW 6\vi.lib\Instr_visa.llb\
VISA Configure Serial Port

Error angle.vi
D:\Subjects and collection\FYP\Oral Presentation\final\Error angle.vi

Error speed.vi
D:\Subjects and collection\FYP\Oral Presentation\final\Error speed.vi

Home.vi
D:\Subjects and collection\FYP\Oral Presentation\final\Home.vi

back-home.vi
D:\Subjects and collection\FYP\Oral Presentation\final\back-home.vi

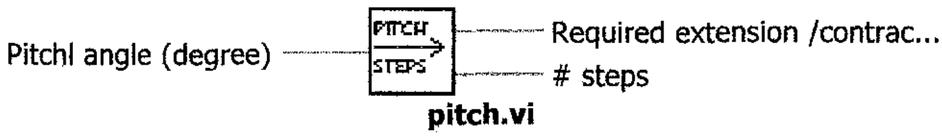
pitch.vi
D:\Subjects and collection\FYP\Oral Presentation\final\pitch.vi

---

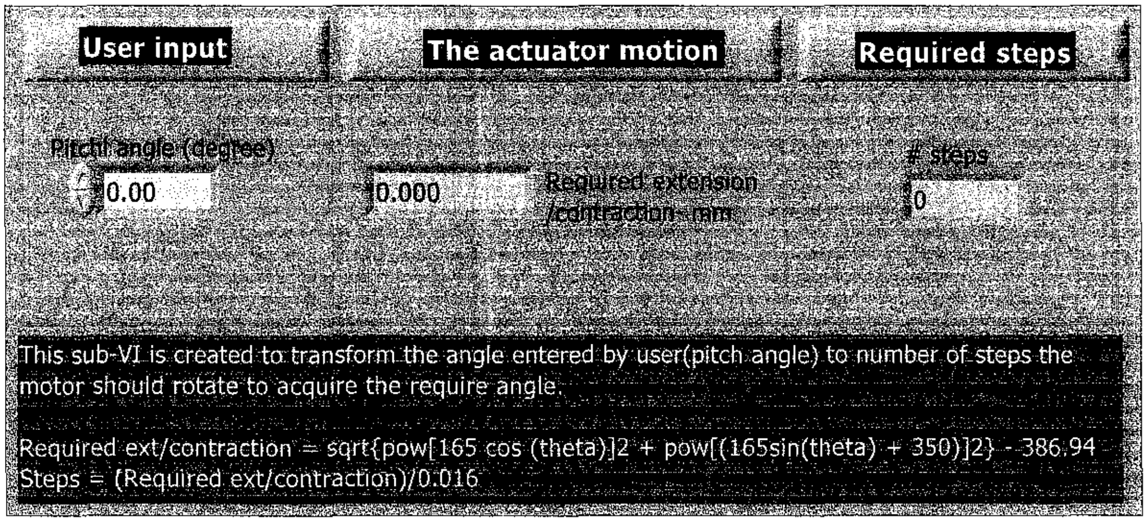
History

"Pich & Roll (sub-VI).vi History"
Current Revision: 92

Connector Pane



Front Panel



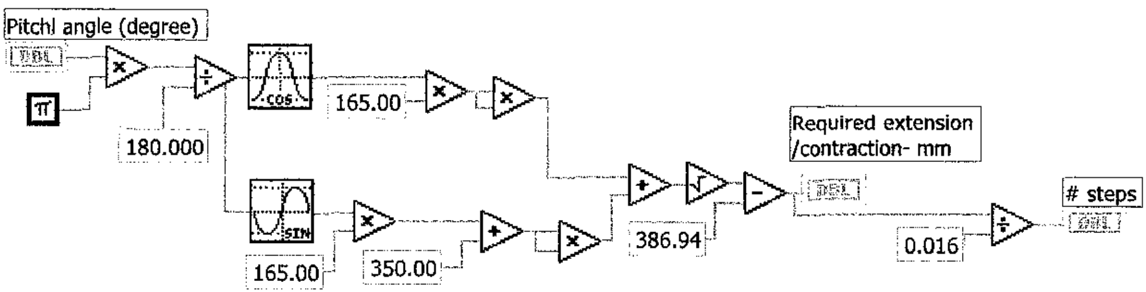
Controls and Indicators

Pitchl angle (degree)

Required extension /contraction- mm

 # steps

Block Diagram



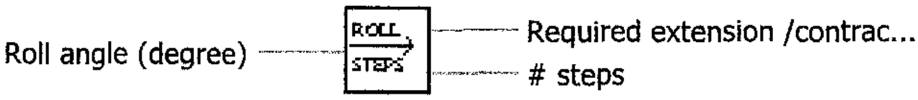
Position in Hierarchy

List of SubVIs

History

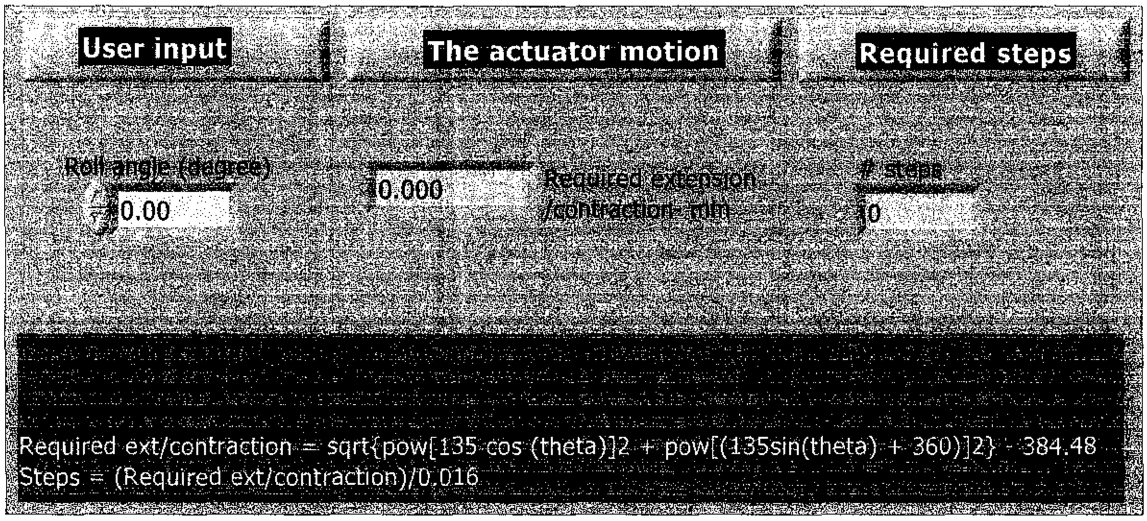
"pitch.vi History"
Current Revision: 6

Connector Pane



roll.vi

Front Panel



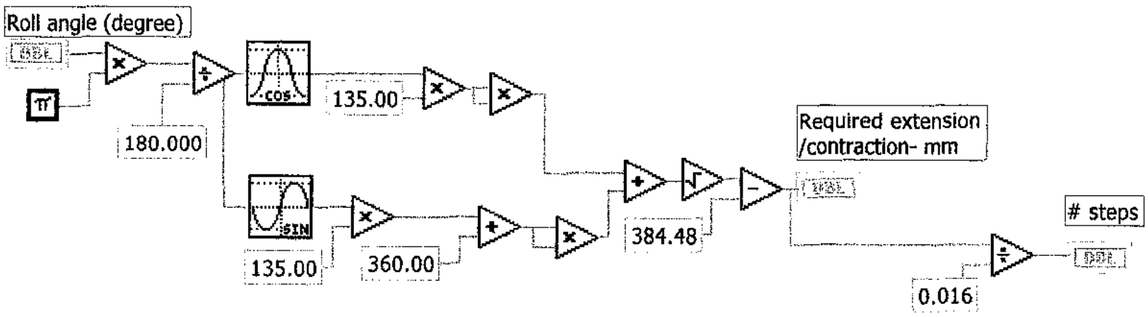
Controls and Indicators

Roll angle (degree)

Required extension /contraction- mm

 # steps

Block Diagram



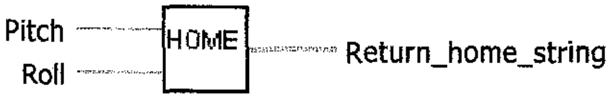
Position in Hierarchy

List of SubVIs

History

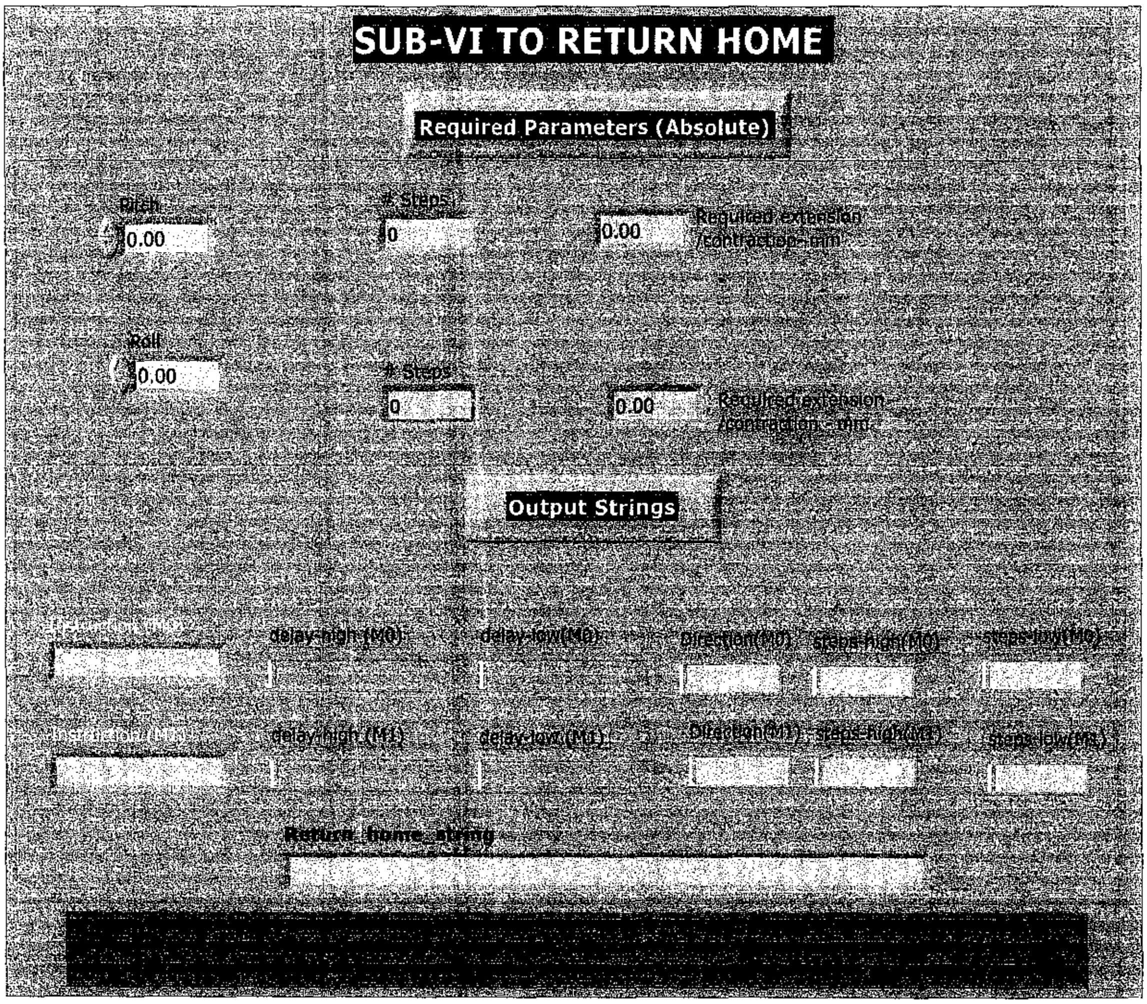
"roll.vi History"
Current Revision: 4

Connector Pane



Home.vi

Front Panel



Controls and Indicators

- **Pitch**
- **Roll**
- **steps-low(M0)**
- **Instruction (M0)**
- **steps-high(M0)**
- **Direction(M0)**
- **Required extension /contraction- mm**

 # Steps

delay-high (M0)
Equivalent ASCII character "Null"

Null Decimal Representation

delay-low(M0)
Equivalent ASCII character "Null"

Null Decimal Representation

Instruction (M1)

steps-low(M1)

steps-high(M1)

Direction(M1)

Required extension /contraction - mm

 # Steps

delay-low (M1)
Equivalent ASCII character "Null"

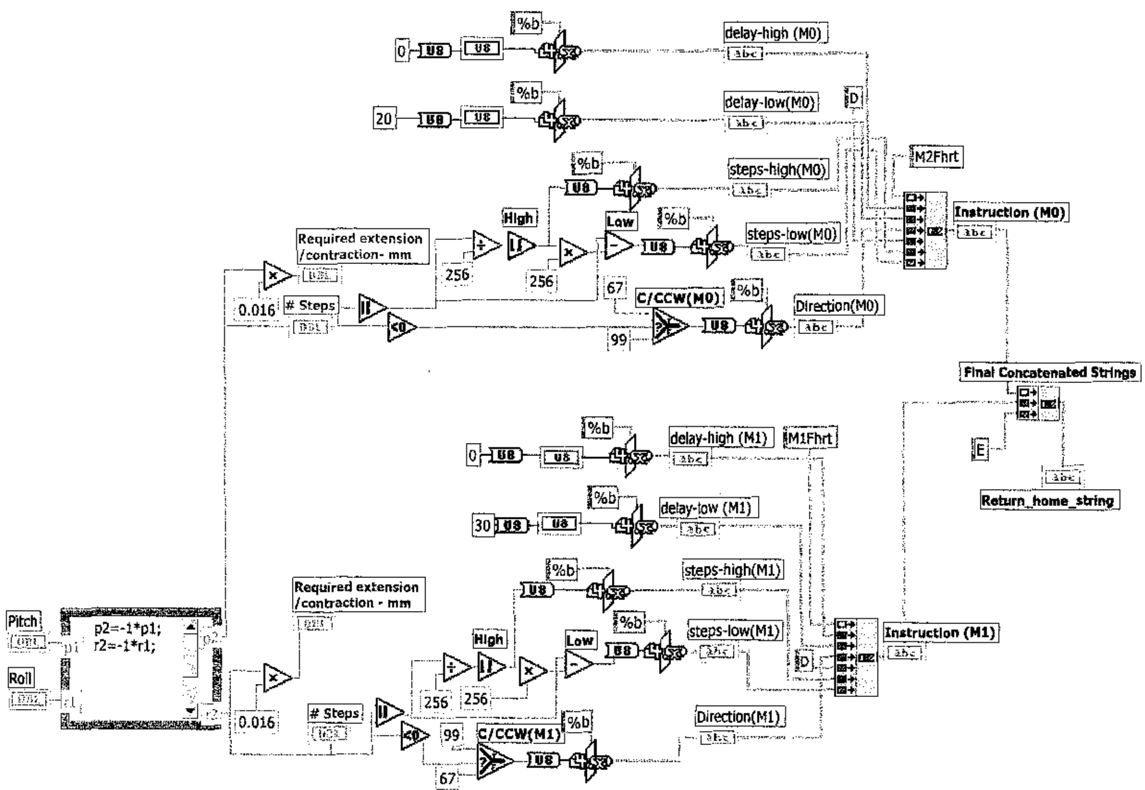
Null Decimal Representation

delay-high (M1)
Equivalent ASCII character "Null"

Null Decimal Representation

Return_home_string

Block Diagram



Position in Hierarchy

List of SubVIs

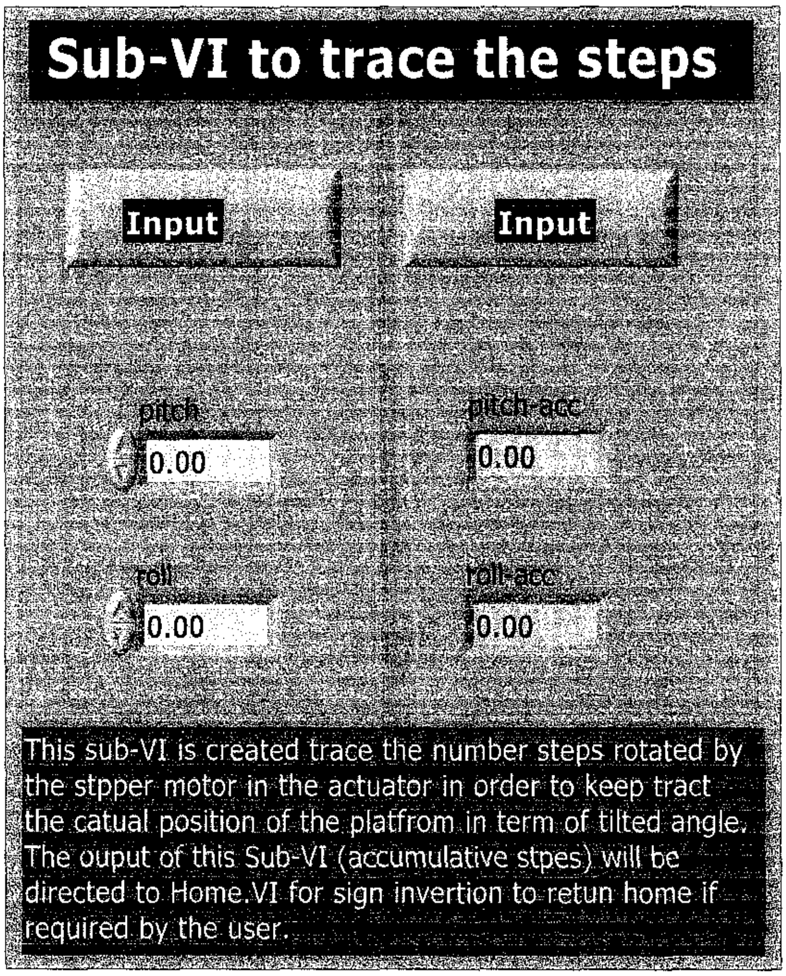
History

"Home.vi History"
Current Revision: 27

Connector Pane

back-home.vi

Front Panel



Controls and Indicators

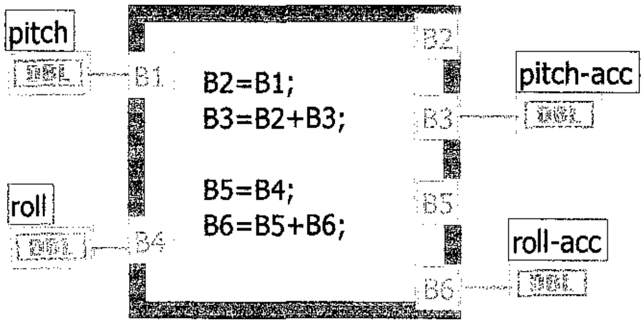
DBL **pitch**

DBL **roll**

DBL **pitch-acc**

DBL **roll-acc**

Block Diagram



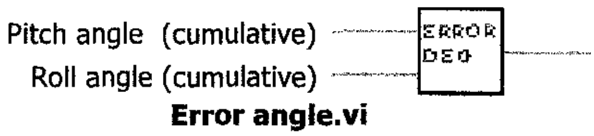
Position in Hierarchy

List of SubVIs

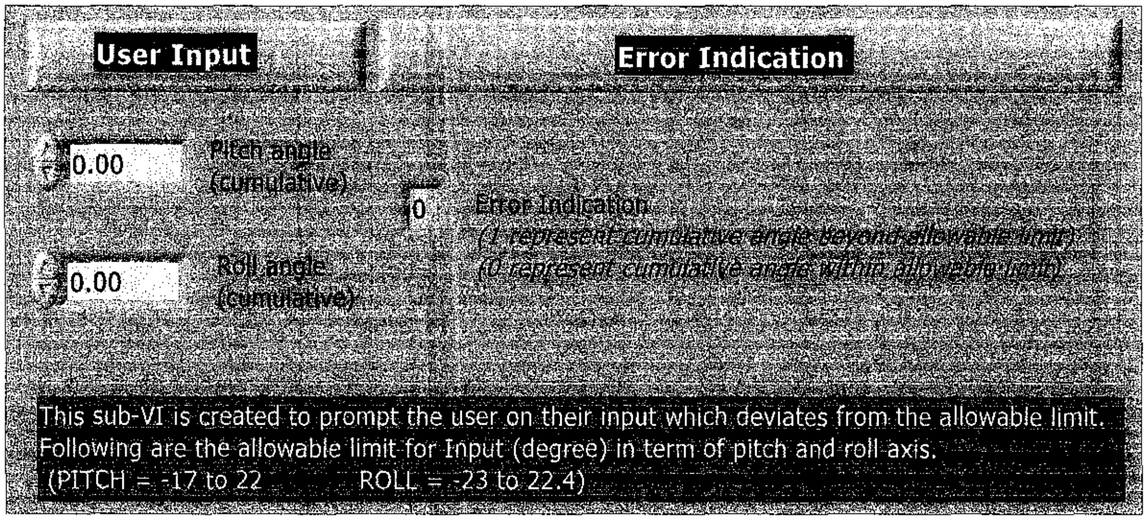
History

"back-home.vi History"
Current Revision: 7

Connector Pane



Front Panel

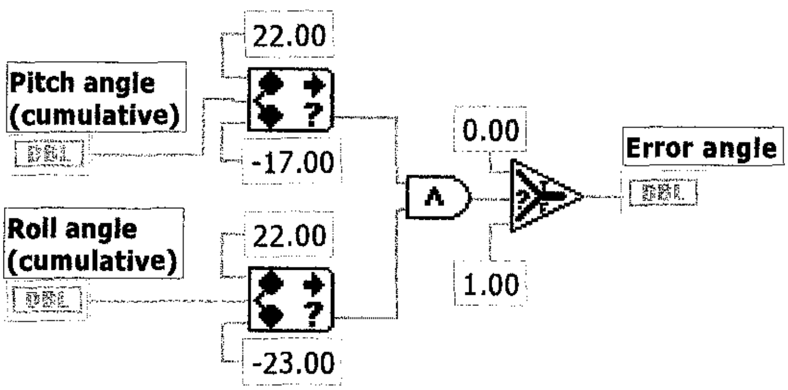


Controls and Indicators

Roll angle (cumulative)

Pitch angle (cumulative)

Block Diagram



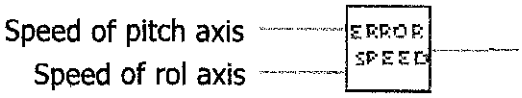
Position in Hierarchy

List of SubVIs

History

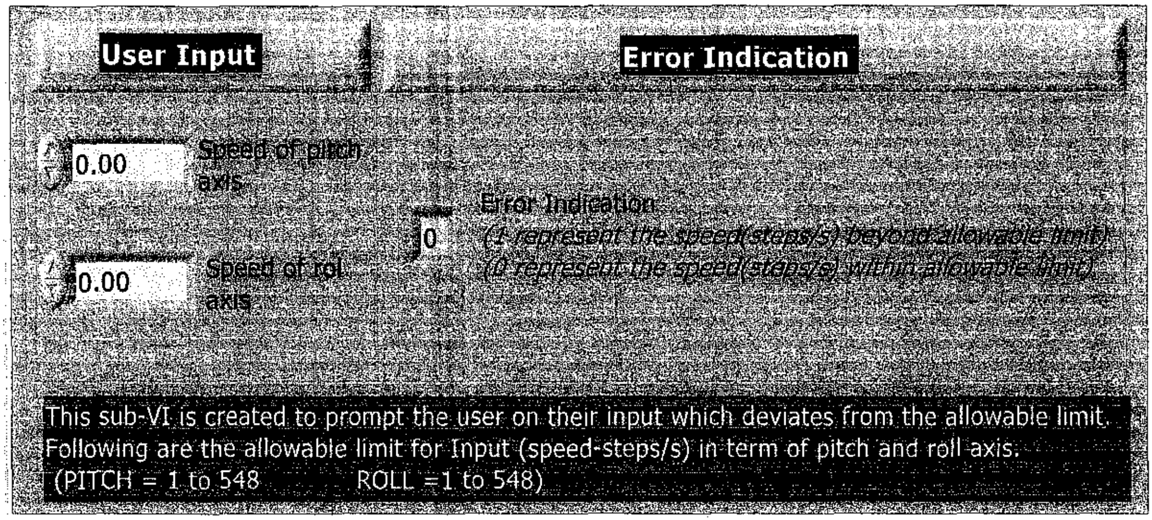
"Error angle.vi History"
Current Revision: 6

Connector Pane



Error speed.vi

Front Panel

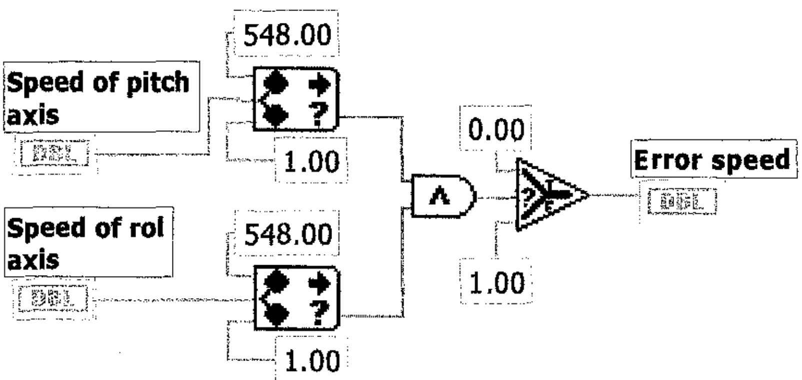


Controls and Indicators

Speed of rol axis

Speed of pitch axis

Block Diagram

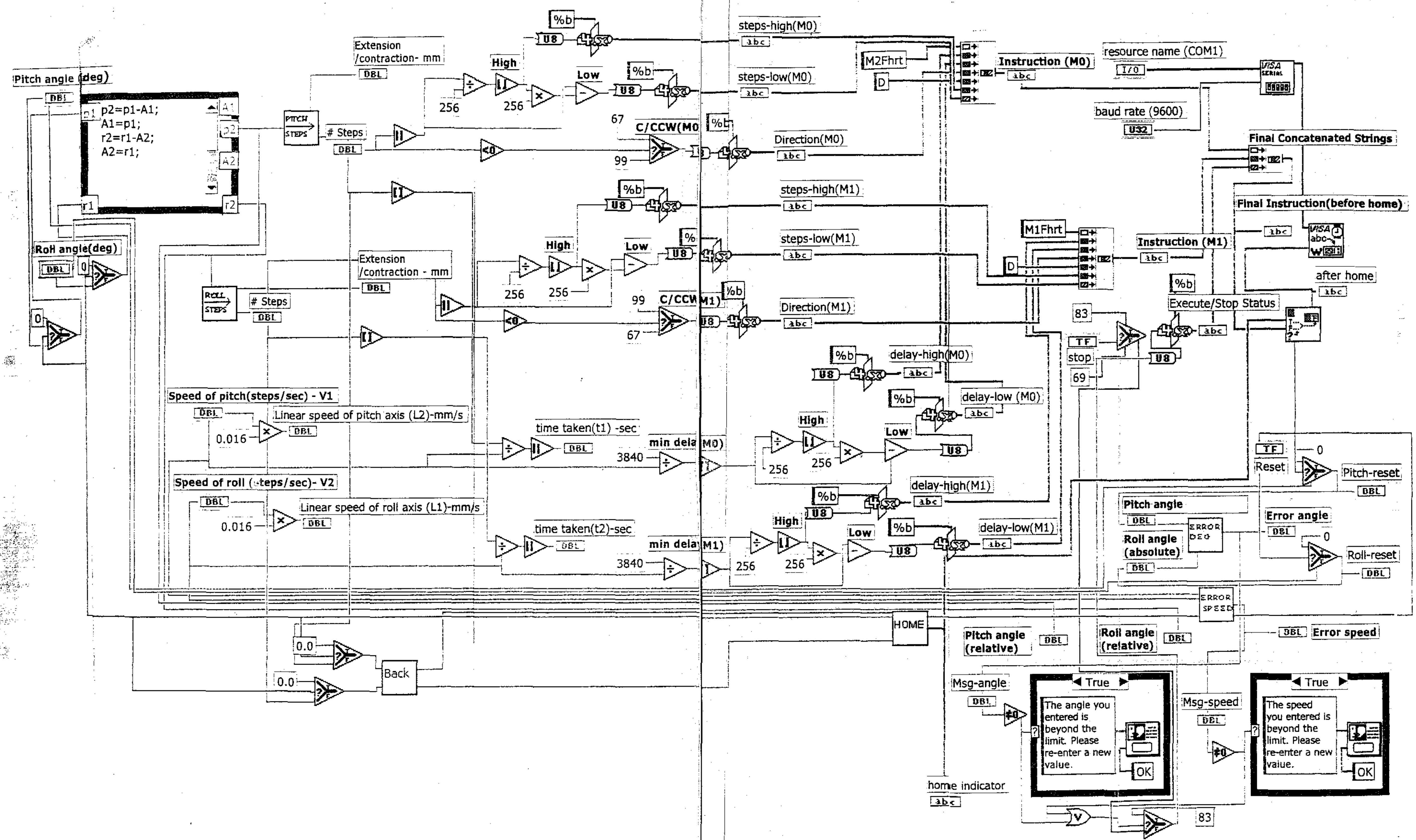


Position in Hierarchy

List of SubVIs

History

"Error speed.vi History"
Current Revision: 7


Pitch angle (deg)
p2=p1-A1;
A1=p1;
r2=r1-A2;
A2=r1;
Roll angle(deg)
Extension /contraction- mm
Steps
High
Low
256
67
99
C/CCW(M0)
C/CCW(M1)
steps-high(M0)
steps-low(M0)
Direction(M0)
steps-high(M1)
steps-low(M1)
Direction(M1)
M2Fhrt
M1Fhrt
Instruction (M0)
Instruction (M1)
resource name (COM1)
baud rate (9600)
Final Concatenated Strings
Final Instruction(before home)
after home
Execute/Stop Status
83
stop
69
delay-high(M0)
delay-low (M0)
delay-high(M1)
delay-low(M1)
Speed of pitch(steps/sec) - V1
Linear speed of pitch axis (L2)-mm/s
0.016
time taken(t1) -sec
min delay(M0)
3840
Speed of roll (steps/sec)- V2
Linear speed of roll axis (L1)-mm/s
time taken(t2)-sec
min delay(M1)
Reset
Pitch-reset
Pitch angle
Roll angle (absolute)
Error angle
Roll-reset
ERROR DEG
ERROR SPEED
HOME
Back
0.0
Pitch angle (relative)
Roll angle (relative)
Error speed
Msg-angle
Msg-speed
True
The angle you entered is beyond the limit. Please re-enter a new value.
The speed you entered is beyond the limit. Please re-enter a new value.
OK
home indicator

Appendix R: File attachments.

Old design (CAD file)
New Design (CAD file)
LabVIEW controller software; Pitch & Roll (sub-VI).vi
LabVIEW stand-alone program
Application example (Drinking Cup Stabilizer. peg)
Standard testing software for S100SMC controller board.
Power point presentation (oral-final.PPT)