University of Alberta

A 6-Degree-of-Freedom Testbed for Small Model Helicopters Control Evaluation

by

Jun Zhao

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Mechanical Engineering

Edmonton, Alberta
Spring 2008

Your file Votre référence
ISBN: 978-0-494-45916-4
Our file Notre référence
ISBN: 978-0-494-45916-4

Abstract

In this project, we want to develop a control system for the helicopter and test the control system. To confirm that our control algorithm can interpret the physical data correctly, we built a testbed which can perform experiments safely and give us raw data to check with the output of our algorithm. To meet the objective, we

- acquired the dynamical data of the helicopter while avoid unexpected crashes by designing the tesbed based on strength calculations which ensures sufficient strength and reduces the cost.

- derived the formulas for the dynamical parameters of the helicopter based on the knowledge in robotics.

- setup an encoder data system. By collecting the required data and transmitting them into a computer program, we are able to obtain position and orientation data of the helicopter.

- carried out some experiments to check the accuracy and repeatability of the encoder data system and reached satisfied results.

Acknowledgements

I would like to sincerely thank some individuals who gave me help while I was working through the project.

Dr. Farbod Fahimi---- For lifting up when I fell down, for being patient when I did it totally wrong, and for believing in me when I didn't believe in myself.

Dr. Carey Jason ---- For providing me useful information when I was confused and stuck.

Andrew Campbell ---- For teaching me and showing me wonderful machining tools when I was unfamiliar. Also thanks for helping me machine the Testbed.

Shop Machine Crew (Roger, Liana, Rick)---- For the uncountable little bits of help that you may not even remember.

Table of Contents

List of Tables

List of Figures

Chapter 1

Review

1.1 Introduction

1.1.1 Introduction and Literature Review

Autonomous aerial vehicles (UAVs) are well known for their advantages that they are tireless and can replace people to execute surveillance and reconnaissance missions in relatively harmful or dangerous environment. Initially, UAVs were mostly used to handle simple missions at dangerous places, but in the past decades, as the level of technology becomes more advanced, UAVs are increasingly being utilized in a wide range including both commercial and military areas [1, 2, 3, 4, 5, 6]. For example, they can be used to monitor roadway traffic, a high altitude UAV can even be a good assistant for weather forecast and disaster predict, etc. The usage of UAVs in space technology and military field are also impressive such as the famous project "Mars Exploration Rover" directed by NASA and the Global Hawk served in U.S.force.

However,the growth of the UAVs will not stop here, they are still at the leading edge place of the aviation. Many researchers are doing research about UAVs, some are talking about their commercial and industrial trend [4, 7, 8], some are conducting academic researches in the UAV area which involves dynamics, electrical applications or special experiments using UAV and so on [9, 10, 11, 12, 13, 14, 15].

Our project can be classified into the UAV practical application class. Among the past research projects done by other researchers about UAVs, unmanned helicopter is the most popular one due to the special characteristics of helicopters such as the vertical take-off and landing ability, nonlinear dynamics, etc [16]. In our project, we intend to build a testbed mounted with a small model helicopter which enables the helicopter to move in a limited space. The overall goal of the project for this phase is to use this testbed to allow safe autonomous controller test experiments for the helicopter.

Before we discuss the testbed project, it will be helpful to review what has been done before to gain some ideas and experiences. The related testbeds generated before are reviewed as followed:

1. The straight-forward application of UAV for sure is in the autonomous surveillance area. In [17], an Autonomous Scout Rotorcraft Testbed (ASRT) is described which was built in Georgia Institute of Technology. The purpose of ASRT project is to construct a vertical takeoff and landing (VTOL) helicopter system to carry out reconnaissance and surveillance assignments. It should be able to fly to an assigned area to search for a human-sized target according to the remote control signal from the operator. To satisfy the functionality, they put the electrical devices such as sensors and cameras onto the helicopter. The sensors can tell apart different conditions ongoing, and the cameras help the operator to locate the target. This system is a visual tracking system which may not be directly related to our project, but the algorithm they used to guide the behavior of the helicopter does intersect with our long-term goals.

2. The experiments that can extend the usage of the UAVs are also intensively conducted in the past decades. A remote controlled helicopter can land on man-made safe destination but what if we want the helicopter to land on some unknown area, for instance, the surface of planetary bodies? To solve this problem, a gas powered model helicopter was accomplished at University of Southern California (USC) in 2001 mounted with an inertial measurement unit, a GPS receiver/decoder, and a color video camera, etc. The image processing algorithm will search the image provided by the camera for a safe landing area which fulfils a pre-determined criterion hence the autonomous helicopter can land safely itself [16].

   Due to the need of precise landing by the next generation Mars project (2007 and beyond), USC developed another testbed based on the vision-toward landing to simulate the spacecraft using a emulator built around a autonomously controlled model helicopter [18]. Although the testbed did only simulate the planar landing scenario which only considered the horizontal and vertical position, roll and the other control inputs say the vertical thrusters, it somehow successfully simulated the motion of the spacecraft. The simulator convert thrusters inputs to stick inputs thus the model helicopter can track trajectories designed for the spacecraft. The simulation would be expanded to 3D scenario in the future work.

   Later in 2006, In the Jet Propulsion Lab (JPL) at California Institute of Technology, an Autonomous Helicopter Testbed (AHT) was constructed [6]. The research purpose of AHT is to support the development of the technologies for

the space area. The onboard avionics are quite similar with the former mentioned one. To complete the characteristics, they developed another testbed which is called Gantry Testbed (GT). By using them together, the research group is able to test more kinds of algorithms and system performance criteria than with only one. The platform comparisons between GT, AHT and other 4 testbeds were conducted. For the comparisons, 11 metrics including different aspects that would be important for the future experiments. The recent work is mainly about how to use a machine vision algorithm to enable safe and precise landing onto unknown planetary surface autonomously.

3. The testbed utilized to test the special algorithms and make sorts of control systems integrated can be classified into another category. The sample we want to introduce here is the interesting testbed developed by Software-Enabled Control (SEC) program in [19, 20].

   The SEC is started in 2000, and two different testbeds were developed in Georgia Institute of Technology (GIT) and University of California at Berkeley separately. The former testbed is called GTMax, a UAV platform with many avionics configured: Two embedded PCs, Inertial Measurement Unit (IMU), Differential GPS, etc. Since the responsibility of GIT is to work as a demonstration integrator, many control algorithm developed by the other universities under SEC program were first flown on the GTMax, many missions to test these algorithm were conducted as well such as delivery of supplies and reconnaissance mission. Actually, GIT has been working on the UAV flight testbed for a long time [21, 22]. In 1993, the first unmanned helicopter autonomous flight was demonstrated. Then, the U.S. Army Autonomous Scout Rotorcraft Testbed (ASRT) project was assigned to GIT from 1994-1997. From 1998-2002, GIT began the program SEC, lots of testbed were constructed not only the helicopter GTMax, but also the fixed-wing aircraft 1/4 scale Piper Cub and Gliders.

   The latter testbed is called Berkeley bear equipped with GPS/INS, a camera and on-board sensors. The mainly obligation of Bear is to support the integration of both control and sensing in a test. The sensing ability which enables the helicopter to look-ahead makes it able to avoid collision with the obstacles. Some experiments were performed to test the capability of the obstacle avoidance. For example, once a time, the group flight the helicopter in a given trajectory which was intersected with trees, power poles, power lines. The sensing ability allowed the helicopter to notice these obstacles, and the path replanning algorithm allowed it to avoid the obstacles.

4. Differ from the individual unmanned helicopter experiment, the leading insti-

tutions are also doing experiment with multi-UAV system, which is in the same direction with our long-term goals. In Stanford University, people constructed an outdoor testbed called STARMAC for multi-agent algorithms testing [23]. They chose a particular helicopter say X4 flyer which has four fixed-pitch rotor on the four ends of a cross frame. This X4 flyer has large availability in maneuverability and is not very expensive like normal model helicopter. Upon the finish of the project, each individual in the multi-agent system will be able to act at a team level, which means it can avoid collision with the other flyers and the obstacles as well. Furthermore, they will be able to fly in a specified formation.

The German Aerospace Center (DLR) at Institute of Flight Systems built a testbed for research in UAV autonomy [24]. The whole project is supposed to apply the manned and unmanned teaming and planning, high reliability and so on. They have established the experimental testbed and configured the necessary software environment, the onboard image processing system accompanied with the other sensors and onboard computer and the rest of the control system parts together make it possible to provide a reliable data source for the later research goals. The project was going to demonstrate the vision based functions when the testbed is flying with the manned helicopter. And it was also supposed to do the experiment of the manned-unmanned formation flight with collision avoidance in 2005-2006.

5. A distinct testbed which is quite the thing what we are going to build arise at University of Vigo, Spain in 1994 to perform a helicopter control experiment [25]. A model helicopter can move freely in the space except some springs used to limit the movement due to safety reasons. The cameras will collect and send the images of the helicopter to the vision computer through IEEE1394 interface. The position and orientation of the helicopter will be calculated on the vision computer using the image processing algorithm. These data are then transmitted to the Real-time control computer to generate the actuation signals, these control signals will then be used to control the motion of the helicopter by the radio system.

6. In Caltech, a testbed is built to perform the identification and control experiment to a model helicopter [26]. The helicopter is mounted on a 3 degrees of freedom (DOF) wrist which in turn is connected to a 3DOF stand. Each of these two joints allows 3 rotation DOFs. For the wrist, it allows pitch, yaw and roll; for the stand, it allows rotations about three axes. This testbed is the most similar to our design.

1.1.2 Project Objectives

We would like to design a testbed that can be used for evaluating developed helicopter control algorithms while the testbed ensures a limited and safe range of motion for the helicopter.

1.1.3 Design Requirements and Considerations

To build a testbed which can fully meet the required experiment functionality while providing some specified limits for the motion is very important. The brief description of the requirements that will guide our design is presented as following:

1. The fundamental requirement is the testbed has to allow all DOF of the real helicopter motion in a limited space. The testbed weight should be as low as possible. The joint frictions must be negligible. The testbed must be balanced to eliminate the effect of the testbed's weight on the helicopter.

2. Due to financial limitation, the instruments that are relatively inexpensive and have sufficient performance must be picked.

3. Since the testbed is supposed to operate in our lab, the space limitation should also be considered. A fenced and completely closed area must be designated for the operation of the testbed. All controls must be accessible outside of the closed area of the operation.

4. The testbed components must be designed such that they do not final under a crash load. This ensures the safety of the testbed operation.

5. The maintenance cost must be as low as possible, if a crash damages the component of the testbed.

Note that inertia is not a design requirement at this stage. The inertia of the testbed during an experiment depends on the acceleration defined for the motions of that experiment. Therefore, when a user designs an experiment, they will consider the effect of inertia and will compensate for it in the controller design.

1.1.4 Thesis Outline

Chapter 2 details what the design requirements and considerations are. This part is expanded according to the deductive method. It first sketches what the project objective is. Then considering the research goals, we select a helicopter which meets special requirements. Finally, the testbed design will be developed around the model

helicopter and the project objectives. Strength validation will be involved, and necessary revision of the testbed will also be introduced.

Chapter 3 concentrates on how to calculate the motion data of the helicopter such as position and velocity base on the arm joint positions by using robotics knowledge. The chapter presents how to derive position and velocity parameters with given data from joint positions and velocities, and how we implement these calculations in a MATLAB program to obtain real-time values of the helicopter's position and velocity.

Chapter 4 explains how to setup the real-time data collection environment. The chapter organizes the contents in hardware, software and experiment sequence. It details how the hardware are integrated, how the software works and how we make adjustments and how we perform our experiments.

Chapter 5 concludes the achievements and how we solve the problems encountered when we are trying to reach our goals.

Appendices come at the end of this thesis, they contain drawing for all of the testbed parts and MATLAB programs developed in this thesis for real-time data acquisition.

Chapter 2

Testbed Design

2.1 Project Objectives and Helicopter Selection

The long term goal for our project is to design a formation control system for helicopters. The project will be done in two steps. First, we build an encoder data system to validate our control algorithm. In this pace, the helicopter can only move in scale range. Second, the helicopter will be able to fly in the sky, the sensor data system will replace the encoder data system to provide helicopter states. Sine the controller is decentralized, when one helicopter can fly on its own, it is easy to extend the results to multiple helicopters.

Based on long-term goals of our project, for this phase, we are going to develop a control system for the helicopter and test the control system. Obviously, the control algorithm will be the core of the whole control system, and we need to guarantee the accuracy and correctness of the control algorithm. To confirm that the control algorithm can interpret the physical data correctly, we plan to build a testbed which can perform experiments safely and give us raw physical data output and in addition allow us to check the errors of the data. The intended data system is divided into 2 systems, one is sensory data system, the other is the encoder data system. Figure 2.1 shows how these data streams work with the controller and different units.

2.1.1 Sensor and Controller

To achieve raw data, we intend to employ the sensory system. Since this part is being done by another group member, here, I will only briefly introduce it. The sensory system consists of two parts: sensors and controllers, both of them will be mounted on the helicopter. These sensors will collect position and orientation data of the helicopter and transmit them to controller. The controller will then send out control signals to aid controlling helicopter's motion. Here, the selected Attitude, Heading

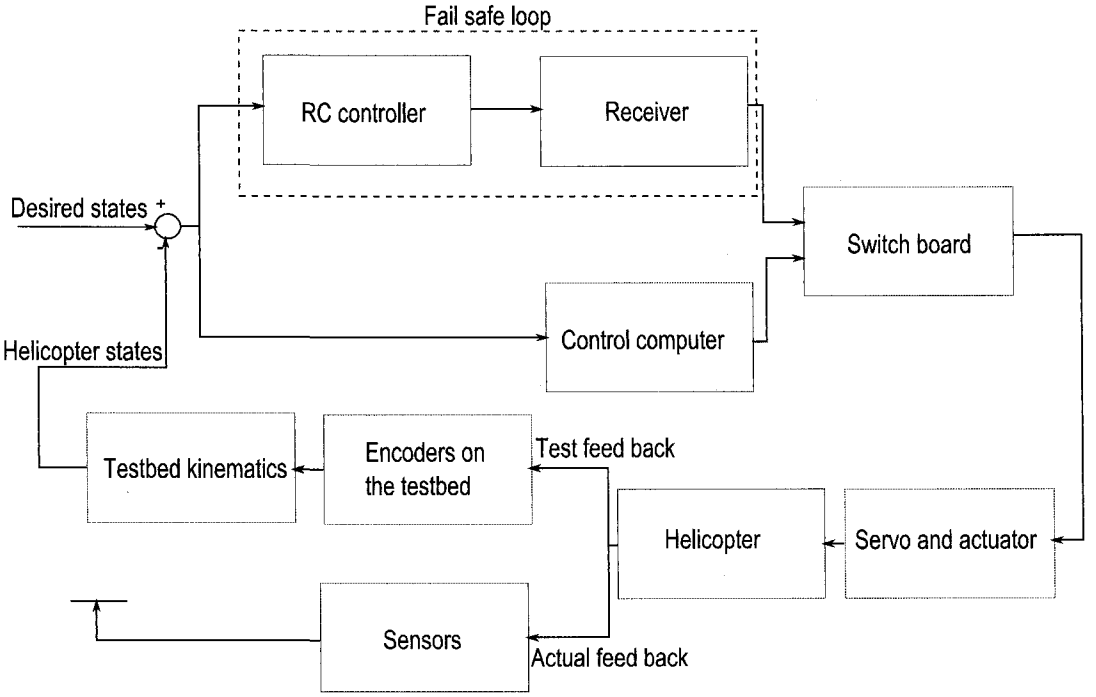

Figure 2.1: The control loop of the intended test system

and Position Reference sensor (AHPRS) is from Rotomotion, LLC, which includes 3 gyroscopes, 3 accelerometers, 1 magnetometer, and integrated GPS. The AHPRS is able to collect the position, orientation, linear and angular velocities and accelerations of the helicopter. The controller is a PC 104 onboard computer. It receives data from AHPRS and uses the control signals generated by the control algorithm to drive servo motors and actuator that actuate the rotor mechanism to control the motion of the helicopter.

2.1.2 Encoder and controller

To confirm the accuracy of the data, another data collecting system is developed. This systems consists of 6 Encoders that can be used to calculate the angular and linear displacement of the helicopter. The collected data from the encoders are required to be transformed into position, orientation, and velocity information to be usable by the control algorithm. Also, the information will be logged by the controller computer to be compared with the data from the sensory system.

2.1.3 Helicopter Selection

Before we decide the details of the testbed design, we need to choose a good model helicopter. Here, we decided to work with an electric rather than a gas powered helicopter because they are clean to run and more suitable for indoor experiment. The payload of the helicopter was also a top priority.

The model helicopter we work with was finally decided to be the Maxi-Joker 2 from Joker-USA. Maxi-Joker 2 is designed to carry a payload of up to 7 lb. This payload capacity is enough for us to mount necessary sensors and controllers on the helicopter, which is important for our future project. Furthermore, the Maxi-Joker 2 is offered on the product lists of several reputable RC helicopter vendors and is in a reasonable price range for a RC electric helicopter of its size. The Maxi-Joker 2 has a strong composite construction. Most other mini-helicopters that have plastic ones. The helicopter also has a good size, so that we will be able to attach the PC 104 autonomous control computer and the AHPRS (IMU) that is needed onboard to calculate the helicopters 3D position and orientation and implement the changes necessary to maintain the helicopter's intended flight path. The Manufacturers of the Maxi-Joker 2 are based in the USA as opposed to the U.K., as are most of the vendors. So we do not have to worry about long shipping waits and inaccessible replacement parts. The Maxi-Joker 2 has an enlarged main rotor diameter of 1.8 m. It can lift a weight of 2 kg. The gear reduction depends on the motor used, between 10.8 and 11.6:1. A rotorspeed of 1200-1300 rpm can be achieved. With a payload of

7 lb the all-up weight is about 16 lb. Equipped with two 10s3p batteries a flight time of up to 20 minutes can be reached, depending on the payload [27].

2.2 Problem Definition

As mentioned in the last section, we need to test the helicopter safely, and we need two data streams to compare. Therefore, the testbed design goal is to acquire the position and orientation data by setting up the encoders properly and making them to work well. Also as a first step for verifying the controller. We want to safely test the helicopter when it is hovering on the testbed autonomously, we want to confine the motion of the helicopter to a relatively small scale so that we can avoid any unexpected crashes or incidents in case the helicopter is out of control.

2.2.1 Overall Conceptual Design

To control the motion of an object, at first, we need to know how many degrees of freedom (DOFs) the object has, and for a successful control, we have to guarantee that the sensors can provide the same number of DOFs and their rates at all times.

For a free helicopter without constraints, there are 6 DOF: three cartesian position components, X, Y, Z and three orientation components, pitch, yaw and roll as shown in Figure 2.2. To make it simpler, we divide the DOFs into 2 sets, one for position the other for orientation.

To generate the set of three positional degrees of freedom, we could use three mutually perpendicular prismatic joints. However, since that design would have been too bulky, we decided to use two revolute joints with perpendicular axes and one prismatic joint. To generate the set of three rotational degrees of freedom, we use three revolute joints with mutually perpendicular and intersecting axes. Based on the above, the design concept of the testbed arm was sketched as shown in Figure 2.3. The XYZ coordinate system determines the position of the testbed's end-effector, whicle the $X'Y'Z'$ system determines the orientation of the testbed's end-effector. The number 1 to 6 stands for joint 1 to joint 6.

2.2.2 Motion Confine

Now, we have a briefly sketched testbed, however it is still far from being complete. The helicopter should finally be allowed to hover autonomously on the testbed. Although the motion of the helicopter is restricted by the range of motion of the joints, for the first experiment phase, we should confine the motion of the helicopter to an even smaller range, and after we make sure everything goes well, we will be able

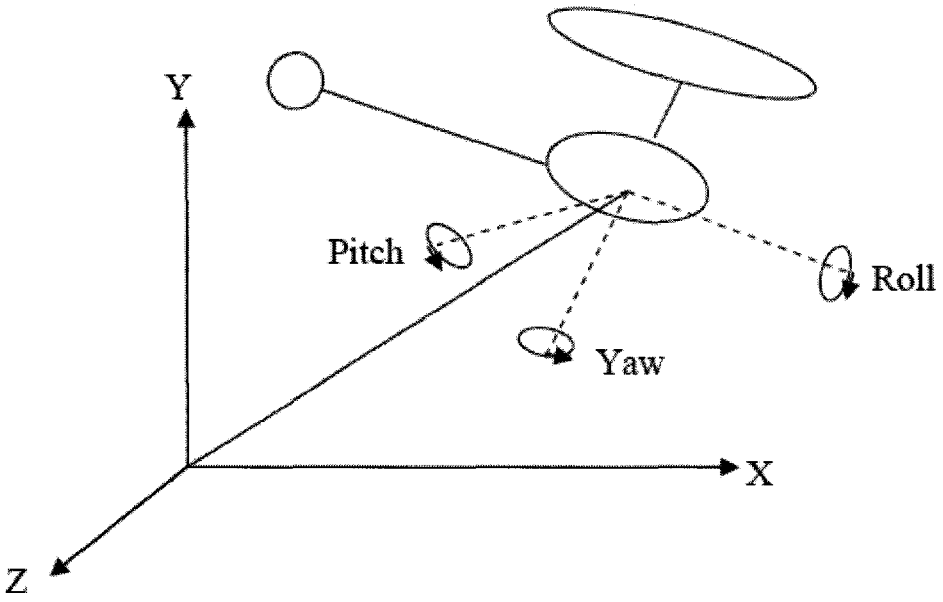

Figure 2.2: The DOFs of a helicopter in space

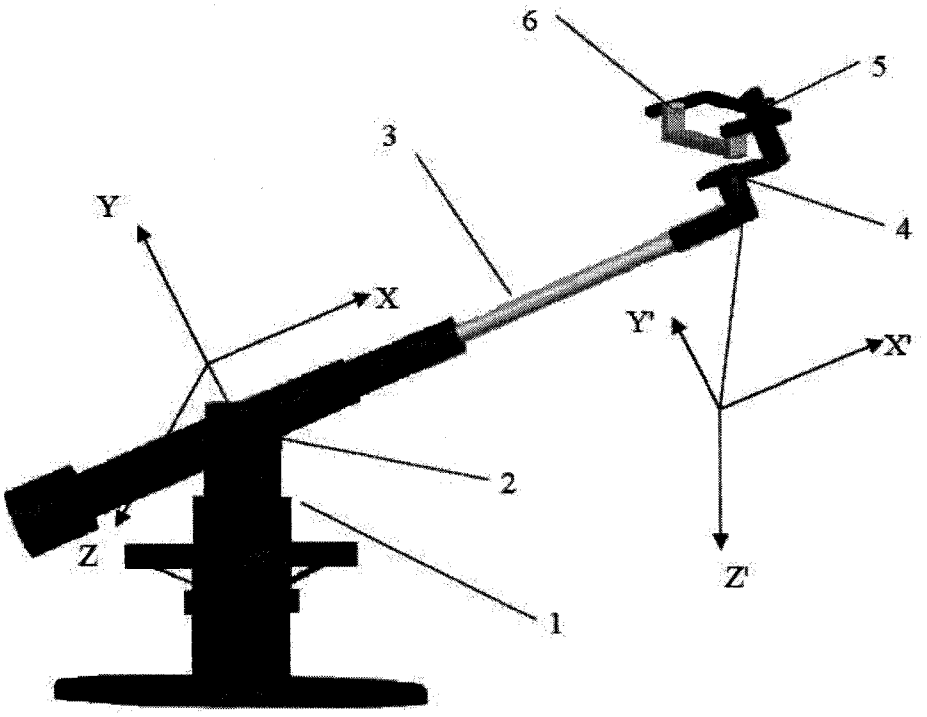

Figure 2.3: The concept image of the helicopter testbed

to somewhat loose the restrictions. For this reason, we intend to build the testbed according to the final motion, however we will add some extra parts as stoppers to be able to limit the motion. Let us assume that the helicopter should be able to fly on a cricle with 2 meters radius and move in the vertical direction with the total range of approximately 2 meters.

A simple graph helps understand the range of the vertical motion requirement as shown in Figure 2.4. The ideal motion would be: the helicopter should be able to move about the axis OO' while moving up and down. The maximum range of the vertical motion is from A to B, and the ideal path would be the line AB, not the arc ADB. As we require 2 meters vertical motion, the length of AC and CB are each equal to 1 meter. The maximum angle for the maximum vertical range is 30 degrees, but, we want to be able to limit that to 10 degrees via a stopper, which results in the vertical range of 0.75 m. Also, since we want the helicopter to be able to move along the path AB, which means OD should be able to be shorten to OC and recovered to the original length freely.

Having considered the size of the helicopter, to avoid the helicopter body and rotor collision with the groud and the testbed and to achieve the required motion range, we computed the approximate range for each joint as shown below. The first numbers show the range of each joint confined by optional stoppers. The numbers in parentheses show the maximum range of the joints:

- Joint 1: $\pm 15°$ ($360°$)
- Joint 2: $\pm 5°$ $\pm 10°$ or $\pm 15°(\pm 30°)$
- Joint 3: 25cm (25cm)
- Joint 4: $\pm 30°$ ($360°$)
- Joint 5: $\pm 30°$ ($\pm 30°$)
- Joint 6: $\pm 10°$ ($\pm 30°$)

Note that the multiple ranges for θ_2 will be used for different experiment scenarios. At the first control validation experiment, the smallest range will be used, because it provides the safest range. When the validation is satisfactory in the small range of motion, the other ranges will be used for further evaluation experiments.

The approximate length of the arm OD was calculated to be 2 meters and the length of the straight bar OO' is also 2 meters. The detailed procedure of how we build each joint will be described later.

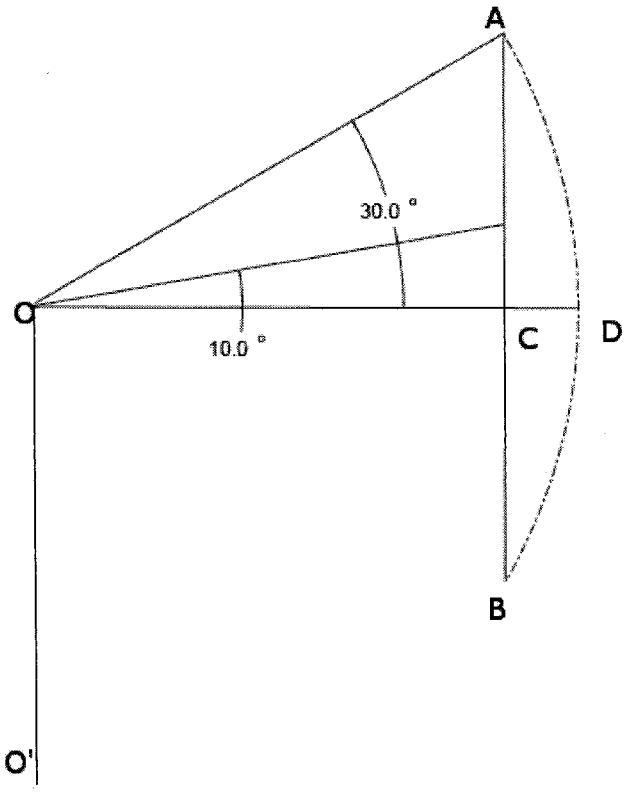

Figure 2.4: The vertical motion range

Table 2.1: Resolution Requirement

Joint	Res Required	Quadrature Res	Res	Displacement Res
1	2 mm	6400PPR	1600CPR	1.96 mm
2	2 mm	6400PPR	1600CPR	1.96 mm
3	0.2 mm	200PPI	50CPI	0.13 mm
4	0.2°	3200PPR	800CPR	0.11°
5	0.2°	3200PPR	800CPR	0.11°
6	0.2°	3200PPR	800CPR	0.11°

2.2.3 Encoder Selection

An encoder is a device that allow us to measure angles and distances accurately. The encoder consists of two pieces, a hub and a disk or strip. One of these two pieces is fixed, the other is moving. One of them is mounted with an optical detector. The other is marked with lines. The detector counts the lines and outputs the count number, which is a representative of an angle or a distance.

To determine the resolution of the encoders based on the precision of position measurements provided by the arm, some simple calculations are needed. Since the helicopter will be mounted at the tip of the arm OD, the helicopter will move about the axis OO', the length of the arm will affect the desired resolution of the encoders for joint 1 and joint 2, the relation between the rotation angle of joint 1 and the horizontal displacement at the helicopter is:

$$D = OD \cdot \sin\theta_1 \approx OD \cdot \theta_1 \tag{2.1}$$

Where D is the horizontal displacement, OD is the length of the arm. Here, OD is equal to 2 meters. The above formulation is correct when θ_1 is quite small. We want the linear displacement resolution to be as small as possible, since this affects the resolution of the whole system. The requirement displacement resolution we pick is 2 mm. By using the formula above, we can calculate the corresponding resolution for θ_1 is 10^{-3} rad. Then, the minimum encoder pulse counts for each revolution on joint 1 is $2\pi \div (10^{-3}) = 6294$ PPR (pulses per revolution). For a quadrature encoder, the corresponding minimum resolution will be 6294/4=1571 CPR (counts per revolution)

The same resolution requirement applies to joint 2 as well. For joint 3, 4, 5, 6 the requirements are rather lower. The resolution requirements for each joint including displacement and corresponding encoder resolution are given in Table 2.1, where Res means resolution, the first column is the resolutions required in mm or degrees. The second column gives us the quadrature resolution in pulse per revolution (PPR) or pulse per inch (PPI) that will meet the needs. The third column is the encoder resolution in count per revolution (CPR) or count per inch (CPI) corresponding to the

second column. The forth column is the resolutions in mm or degrees corresponding to the second column.

We need to notice that, since we care about the displacement changed between adjacent pulses measured along vertical and horizontal axes, the above required resolution should indicate the minimum resolution in worst states. For joints 1 and 2, the worst displacement measurement resolution occurs when the arm is horizontal.

Another design criteria besides the resolution is the size of the encoders. They should be as small as possible, so that we can reduce the size of the joints. We first considered the metal encoders, however, we did not choose them due to their extremely high price, large size, and relative higher weight. We later found better encoders from USdigital [28]. Their encoders are quite inexpensive. The part used for counting is made of polymer. The flat and thin polymer plate can significantly decrease the space of encoder. The detection head of encoder is also small and easy to mount.

The encoders we chose are the follows:

- Joint 1: HUBDISK-2048-394-2-I
- Joint 2: HUBDISK-2048-394-2-I
- Joint 3: LIN-200-12-I
- Joint 4: HUBDISK-1250-394-1-I
- Joint 5: HUBDISK-1250-394-1-I
- Joint 6: HUBDISK-1250-394-1-I

The rotary encoder was expressed in the form: part-CPR-shaft diameter (0.XXX inch)-outside diameter-options ('I' stands for index); the linear encoder was expressed as: part-CPI-length-options.

The resolutions for these encoders are as shown in Table 2.2. By comparing this with Table 2.1, we can see the encoders selected can meet the resolution requirements.

2.3 Design Details

Here the testbed is divided into several sections, each section is introduced one by one. Strength validation, and revisions of the parts will be followed at the end, MATLAB program that has been written for design calculations will be presented as well.

Table 2.2: Encoder Resolutions

Joint	Res	Quadrature Res	Displacement Res
1	2048CPR	8192PPR	1.53 mm
2	2048CPR	8192PPR	1.53 mm
3	200CPI	800PPI	0.03 mm
4	1250CPR	5000PPR	0.072°
5	1250CPR	5000PPR	0.072°
6	1250CPR	5000PPR	0.072°

2.3.1 Features of Distinct Sections

Base Section

As shown in Figure 2.5, this part is quite simple and is the base of the whole system. Its main function is to support the total weight of the testbed.

The base plate is made to a reasonable size and will be bolted to the ground. The straight bar is hollow. It is used to lift up the helicopter to a specified height. To make the experiment for the current phase easier, we designed the bar to be 139.7 mm (55 in) long. In order to avoid that tail of the helicopter hitting the floor, we designed an extension bar. By adding the extension to the straight bar, sufficient length was reached. The straight bar, the extension bar, and the base plate are connected by the connectors.

To constrain the moving range of joint 2, a ring is connected to the base via track roller guides, the ring can stop the arm from moving once the arm hits it. Since the collision energy could be large and might damage the parts, we mount some rubber material on the upper surface of the ring to avoid hard metal-to-metal contact. However, the energy that can be absorbed by the rubber is very limited. So we need another mechanism to absorb the collision energy. The solution is to install springs to the ring, which could cause another problem: The arm will hit one side of the ring, hence the force would not be distributed equally. Different movement of the spring could cause the harmful imbalance. So, we use 4 track roller guides (product number: 60135K71 and 60135K13) from McMASTER-CARR on each surface of the straight bar, and the ring support ribs are mounted to the track roller, the extension springs are also mounted to the track rollers and another fixed bolts. If the arm hits the ring, the extension spring will be extended to absorb the energy, and all the springs will be extended by the same length due to the existence of the track roller guides. Ring, its support ribs, and the track roller guides together enable the desired functionality of this part. Furthermore, we can adjust the allowed moving range of joint 2 by simply changing the position of track roller guides. The 3 allowed moving ranges of joint 2 we designed are $\pm 5°, \pm 10°$ and $\pm 15°$.

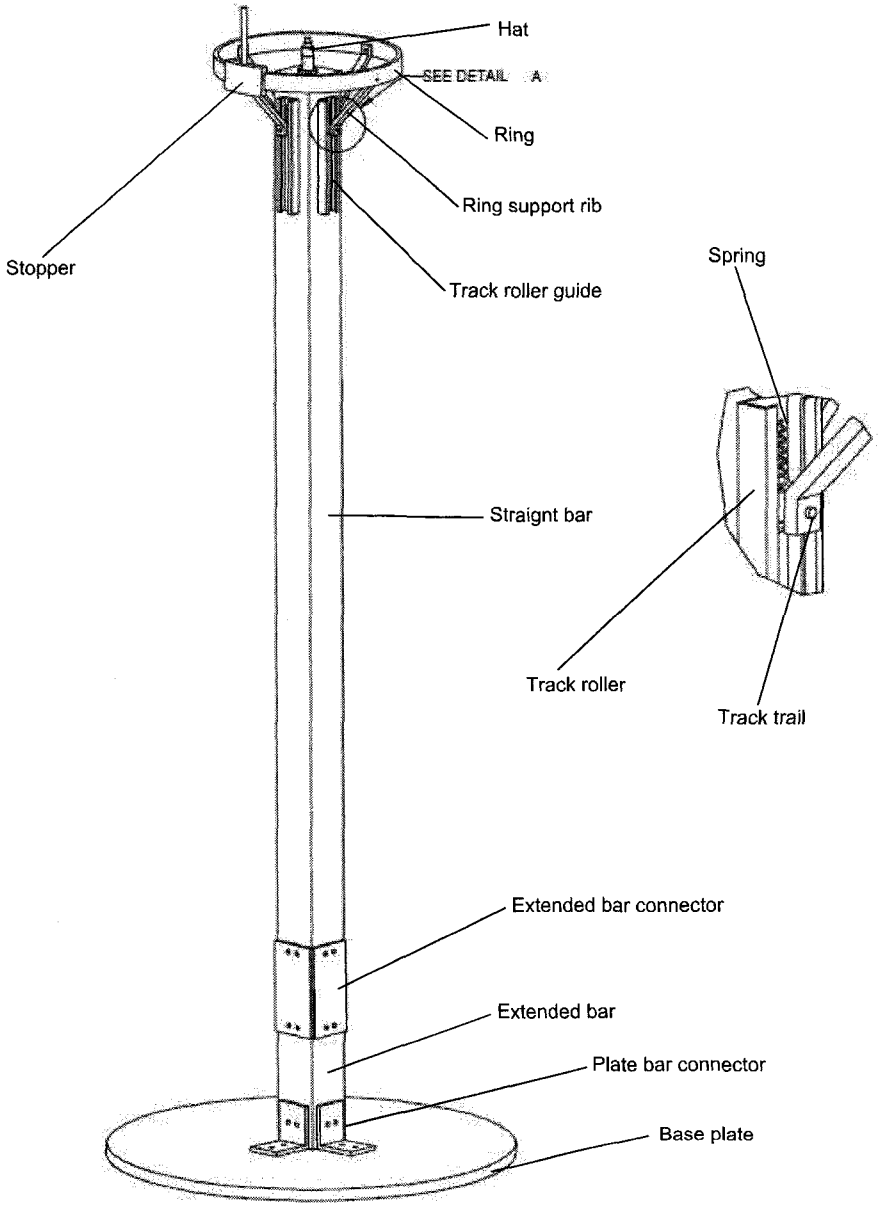

Figure 2.5: Schematic diagram of the base section

The stopper is used to limit the moving range of joint 1, we actually have 2 stoppers, and the arm can only move inside the stoppers. The hat is the connection part with the next section, the taper roller bearing, thrust needle bearing, nut and encoder disk are mounted to the hat.

Base-arm Joint Section

As shown in Figure 2.6, this section contains 2 joints, joint 1 and joint 2. The encoder for joint 1 is omitted to make the figure clearer.

Here, we have a yoke shape constructed by 3 leg parts, the middle part of the yoke is connected to the base group, the side parts of the yoke are used to install encoder and tube connectors and corresponding bearings. The encoder for joint 2 is shown in the figure, we can see the encoder holder for joint 2 lifts the encoder hub to a required height, so the tolerance requested by the encoder mounting can be satisfied. Also, the distance between the encoder hub and the disk tip is measured with high accuracy. The sleeve coat of the next section inserts into the hole on the tube connector and they are fixed by a set screw.

Arm Joint Section

As shown in Figure 2.7, this section can provide the sliding motion and constructs the 3rd joint. This joint is the most complicated one in the whole system. The only allowed motion is a slide in a specified range.

To allow the sliding motion, we developed a tube-shaft system. The shaft can slide inside the outer tube, at each end of the outer tube. There is a linear ball bearing (product number: 8974T3) from McMASTER-CARR which separates the shaft and the outer tube to significantly decrease the friction force. However, the tube-shaft system allows not only slide but also rotation.

The shaft that we used for the tube-shaft system is a hardened precision shaft from McMASTER-CARR (product number: 6061K646) as requested by the linear bearings [29]. And since the hardened precision shaft is expensive and hard to machine, the arm was designed to consist of two pieces, one is the shaft, the other is the extension tube. They were clamped together by a tube connector part. The reason we bent the inner tube to the shown shape was because we need to align the center of mass of the helicopter with the centerline of joint 3. This helps with simplifying the calculations of position of the helicopter.

To stop the revolution, a trail was added on the top of the sliding shaft. As we can see, the two trail support blocks were clamped onto the shaft, which hold the trail together. The other two strip support blocks are bolted onto the outer tube. The trail was constrained to slide along a notch machined to the strip support block.

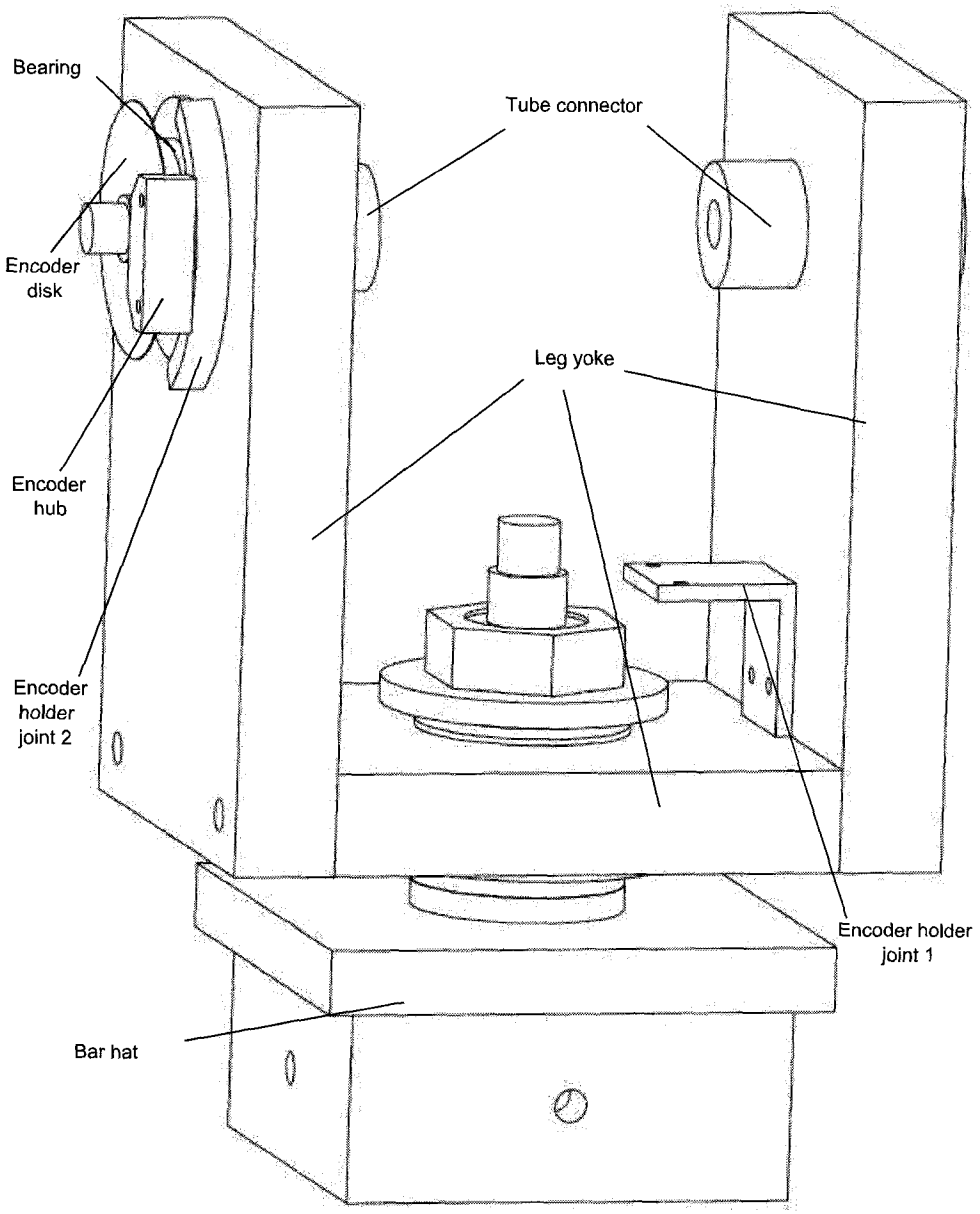

Figure 2.6: Schematic diagram of the base-arm joint section

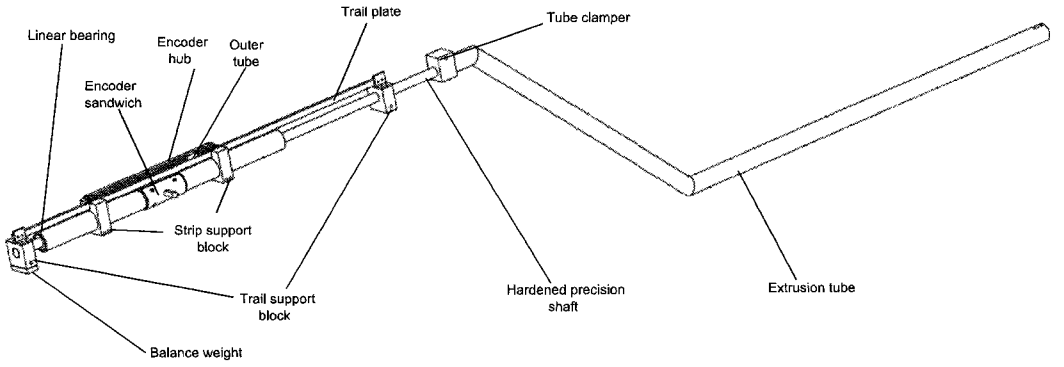

Figure 2.7: Schematic diagram of the arm joint section

Hence, the shaft can not rotate relative to the outer tube.

The trail system also holds the polymer strip of the linear encoder, which was sandwiched by two pieces of thin aluminum plates. The whole encoder sandwich assembly is bolted to the strip support block. The other part–the encoder hub– is mounted to the trail and hence the relative motion between strip and hub of the linear encoder is generated. The tolerance required by the encoder mounting was also considered during the design process.

Wrist Section

As shown in Figure 2.8, this section is actually the hand of the testbed including 3 joints, and it plays the role of holding the helicopter and aligning the center of gravity (C.G.) of the helicopter with the axis of the third joint. It is also in charge of giving the helicopter 3 rotational DOFs to enable the helicopter to rotate in the motion of pitch, roll and yaw.

The C.G. of the helicopter must be exactly in the middle cross section plane of the ski adjustor. The C.G. of the helicopter must also be aligned with the axis of the third joint. With these alignments, the weight of the helicopter will not tip it to any sides, and the position calculation are easier. The reason will be detailed in chapter 3. To make the C.G. to sit in the middle cross section of the ski adjustor is rather easy. The vertical position of the helicopter C.G. is harder to estimate. So here, two adjusting systems are embedded, one of them is used to adjust the holder of the helicopter say the ski adjustor, mainly in the vertical direction. The other system is

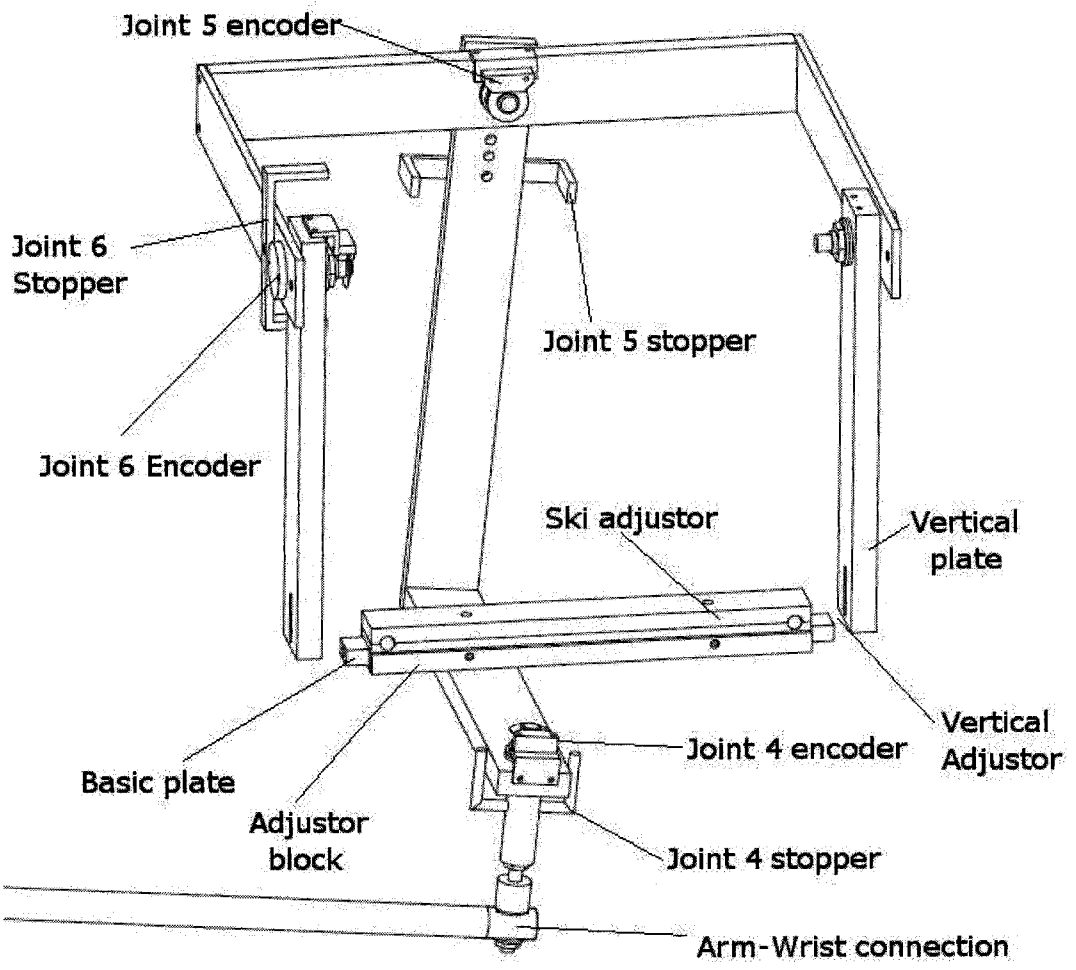

Figure 2.8: Schematic diagram of the wrist section

used to adjust the lateral position of the helicopter on the ski adjustor.

Let us begin from the bottom of this section. The bottom, labeled arm-wrist connection, consists of washers, nut and shaft. Both ends of the shaft are screwed, one end goes into the inner tube and is fixed, the other goes into the upper shaft. The end of the inner arm is flattened hence the washer can be attached to it. The down end of the shaft can be perfectly tightened. The upper shaft is screwed inside so that the lower shaft can move relative to it. If we want to change the gap between the two shafts, both of the two nuts will be loosen. The lower shaft will move relative to the upper one, while moving relative to the inner tube in the opposite direction. This structure makes it possible for us to adjust the vertical position of the ski adjustor without changing the required initial setup of the system.

If we look at the upside portion of this section, we can see the encoder and its corresponding stopper. The encoder takes the angle signal and the stopper constrains the motion of joint 4. This structure is actually the same for these 3 joints.

There is a second adjust system located at joint 6. The vertical adjustor in joint 6 is utilized to moderate the vertical position of the helicopter. This part does its job by two plates. As we can see, a groove was made on the vertical plate, which implies the adjustable range. A threaded hole was made to the basic plate. We will use a bolt, a nut and a washer to connect these two parts.

The two horizontal positions can be adjusted by the ski adjustor. The ski of the helicopter are clamped by the adjustor block, and we can adjust the back and forth position once we loose the bolts on the top of the adjustor block. And the block itself can move left and right to adjust the left and right position.

Now, all the sections of the testbed and how the basic mechanism works have been introduced. All the dimensions are only based on the functionality of the testbed. The dimensions will be further modified if we take the strength checking in consideration. That is discussed in the next section.

2.3.2 Strength Calculation and Optimization

Wrist Strength Validation and Optimization

The outline of our testbed is modeled in Pro/Engineer as a 3D model. However, before manufacturing, we must make sure all the parts are strong enough and would not be damaged under critical situations. For the static state, all the parts might be strong enough to take the weight. However, we need to know how the arm survives the most dangerous state. We need to think about what would happen if the helicopter was hovering and suddenly the power is cut. The arm should not break or get damaged.

Since the shear force and bending moment largely depend on the weight of the wrist section and the helicopter. It will be our first assignment to figure out how we

can lower down the weight, and guarantee sufficient strength. For this purpose, we wrote a MATLAB program. The program will be described later in this chapter, here, the approximate assessment is presented. The MATLAB program can give us the angular acceleration of the arm, while being stopped by safety springs after a sudden drop, which is important for the calculations. The maximum angular acceleration we used based on our program is 25 rad/s^2 (see Figure 2.18). Also, the material we used is Aluminum 6061 T6 whose yield stress is 275 MPa. We can get the approximate force by the following equation where m_h is the mass of helicopter, L is the length of the extruded arm, α is the angular acceleration of the arm . m_h was measured to be 5.44 Kg. The reason we did not include the mass of the arm is because the arm is supposed to be fully balanced. The mass of the arm will not affect force.

$$F = m_h(L\alpha + g) \approx 325 \text{ N} \tag{2.2}$$

Note that, we did not include the mass of the arm in the above equation because the arm is fully balanced. The mass of the arm will not affect the impact force.

As shown in Figure 2.9, we simplify the wrist into 6 models, and we investigate their stresses and see if they have reached the critical point of failure. For model 1, 3, and 6, we handle them as shear bending scenario. For model 2 and 5, they are computed as pure axial force scenario. Finally, for model 4, we use distortion energy theory to solve this integrated loading case. Basically, we could get the required relations between the dimensions of the cross section of these small pieces and a specified value based on the knowledge in mechanics of materials [30].

The calculation process is tedious and here we prefer to give only these formulas, results and how we decide the dimensions.

Model 1:

$$\frac{F/2 \times 2.975}{bh^2/6} \leq \sigma_{max} = 275 \text{ MPa} \Rightarrow bh^2 \geq 0.2622 \text{ cm}^3 \tag{2.3}$$

In the above formula, b and h are the width and height for cross section of the plate, and these labels will be applied to following formulas. For this plate, dimensions of cross section stay with no modifications

Model 2:

$$\frac{F/2}{bh} \leq \sigma_{max} = 275 \text{ MPa} \Rightarrow bh \geq 0.0059 \text{ cm}^2 \tag{2.4}$$

Selected dimensions: b: 7.62 mm (0.3 in); h: 19.05 mm (0.75 in);

Model 3:

$$\frac{F/2 \times 13}{bh^2/6} \leq \sigma_{max} = 275 \text{ MPa} \Rightarrow bh^2 \geq 1.147 \text{ cm}^3 \tag{2.5}$$

Selected dimensions: b: 5.08 mm (0.2 in); h: 19.05 mm (0.75 in);

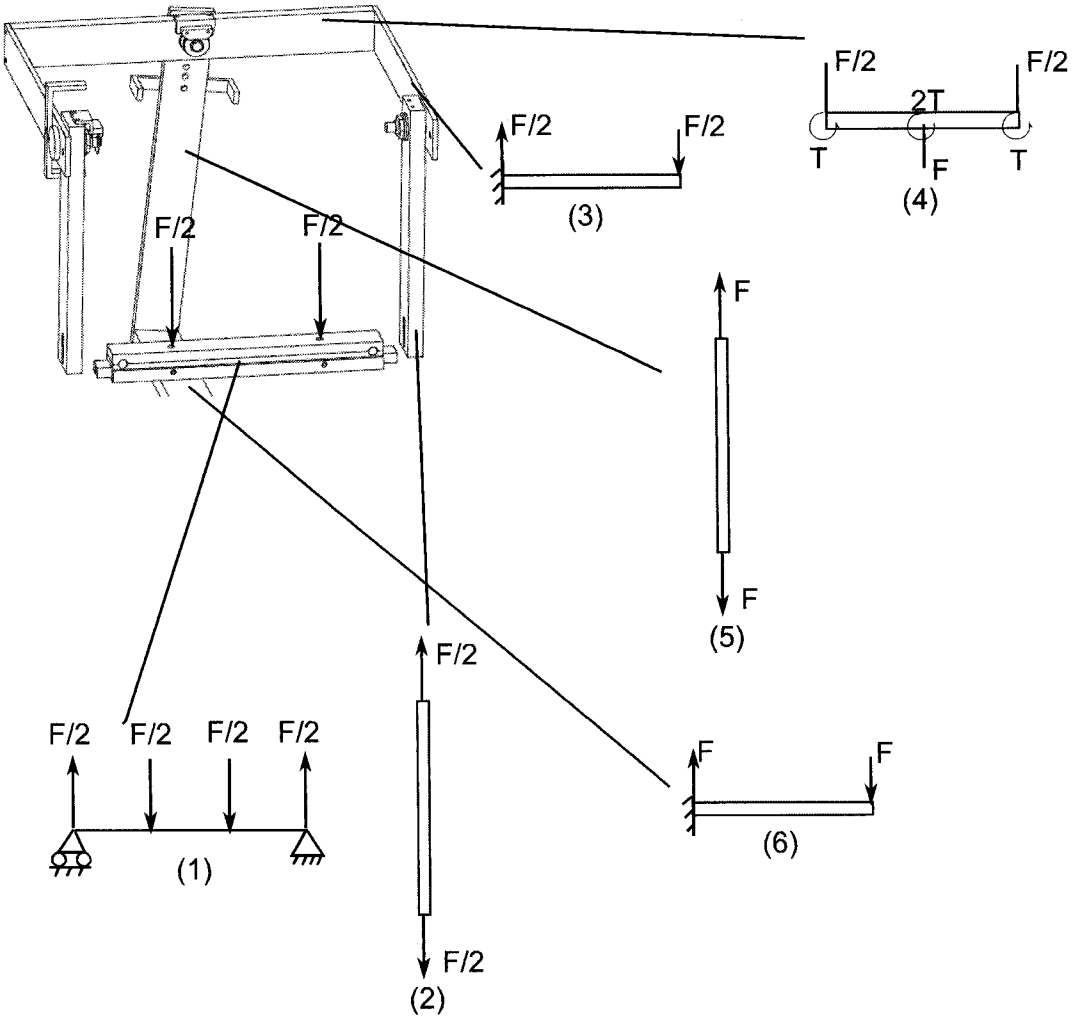

Figure 2.9: The Simplified Force Model of the Wrist

Model 4:

For this plate, we selected b to be 12.7 mm (0.5 in), h to be 25.4 mm (1 in), and calculated to see if they are strong enough.

At danger point A (middle up point of the danger cross section)

$$\sigma_x = \frac{Mc}{I} = \frac{M \times 0.5inch}{bh^3/12} = 21.4 \text{ MPa} \tag{2.6}$$

$$\tau_{xz} = \frac{Tc}{J} = \frac{T \times 0.5inch}{bh^3/12 + hb^3/12} = 28.96 \text{ MPa} \tag{2.7}$$

$$\tau_{max} = \sqrt{(\frac{\sigma_x - \sigma_z}{2})^2 + \tau_{xz}^2} = 30.87 \text{ MPa} \tag{2.8}$$

$$\sigma_1 = \frac{\sigma_x + \sigma_z}{2} + \tau_{xz} = 41.57 \text{ MPa} \tag{2.9}$$

$$\sigma_2 = 0 \text{ MPa} \tag{2.10}$$

$$\sigma_x = \frac{\sigma_x + \sigma_z}{2} - \tau_{xz} = -18.26 \text{ MPa} \tag{2.11}$$

At danger point B (Middle side point of the danger cross section)

$$\tau_{bending} = \frac{3 \times F/2}{A} = 0.754 \text{ MPa} \tag{2.12}$$

$$\tau_{max} = \tau_{bending} + \tau_{torsion} = 32.624 \text{ MPa} \tag{2.13}$$

Then, we have

$$\sigma\prime = \sqrt{\sigma_1^2 - \sigma_1\sigma_3 + \sigma_3^2} = 53.1 \text{ MPa} \tag{2.14}$$

For axial stress, the safety factor is

$$N = \frac{0.5\sigma_{max}}{\sigma\prime} = 2.6 \tag{2.15}$$

For shear stress, the safety factor is

$$N = \frac{0.577\tau_{max}}{\sigma\prime} = 4.9 \tag{2.16}$$

Since we for this size plate, the weight is sufficient small, we would like to keep the cross section dimensions we selected.

model 5:

$$\frac{F}{bh} \leq \sigma_{max} = 275 \text{ MPa} \Rightarrow bh \geq 0.0116 \text{ cm}^2 \tag{2.17}$$

Selected dimensions: b: 5.08 mm (0.2 in); h: 19.05 mm (0.75 in);
model 6:

$$\frac{F \times 12.325}{bh^2/6} \leq \sigma_{max} = 275 \text{ MPa} \Rightarrow bh^2 \geq 2.2942 \text{ cm}^3 \tag{2.18}$$

Selected dimensions: b: 7.62 mm (0.3 in); h: 19.05 mm (0.75 in);

The revised wrist section is shown in Figure 2.10. As we can see, the areas for bearing and other important connections are kept larger. We only slimed down other areas to reduce the weight. The other parts such as the bolts and bearings were validated as well. Some structures are also revised, we used a clamper and slider structure to replace the ski adjustor. When the clamper is loose, the helicopter can be adjusted in back and forth direction, when the slider is loose, the helicopter can be adjusted in left and right direction. This structure can not only meet the adjusting requirements as the ski adjustor but also has small weight.

Arm Strength Validation

To avoid metal-to-metal contact, we designed the testbed with the springs, which can absorb most of the harmful energy. When the helicopter is in controlled state, no large force is generated. If the helicopter acts strangely, a user must cut the power to the helicopter for safety. When the helicopter loses power suddenly, it will fall and the arm collides with the ring and compress the springs. When the springs are compressed to their limit, the forces generated in the system also reach their maximums. This state is the most dangerous state, we are considering the stresses in this state as design stresses. Based on this state, we made a program in MATLAB and did necessary theoretical calculations to determine the coefficient of stiffness of the spring and to reach an optimized solution for the allowed range of the motion.

As shown in Figure 2.11, L is the total length of the arm, X is the radius of the upper ring, K is the coefficient of stiffness of the spring. The arm could move freely for the range $\pm\theta_1$, and if the helicopter loses power, the arm will compress the springs of the ring and when the angle between the arm and the horizontal turns to be θ_2, the spring force reaches maximum and the arm stops.

The forces labeled in this figure are all the forces applied on arm joint section. The important forces that will be used in the later calculation are: B_x, joint force in X direction; B_y, joint force in Y direction and also α, angular acceleration with which the arm bounces back. We can simply list the formulations as followed based on basic knowledge of mathematics. In the formulations, m_h is the mass of the helicopter and wrist together. m_t is the weight of arm including balance weight. F_a is the force due to acceleration. a_t is the acceleration of the arm. a_x is the component of the

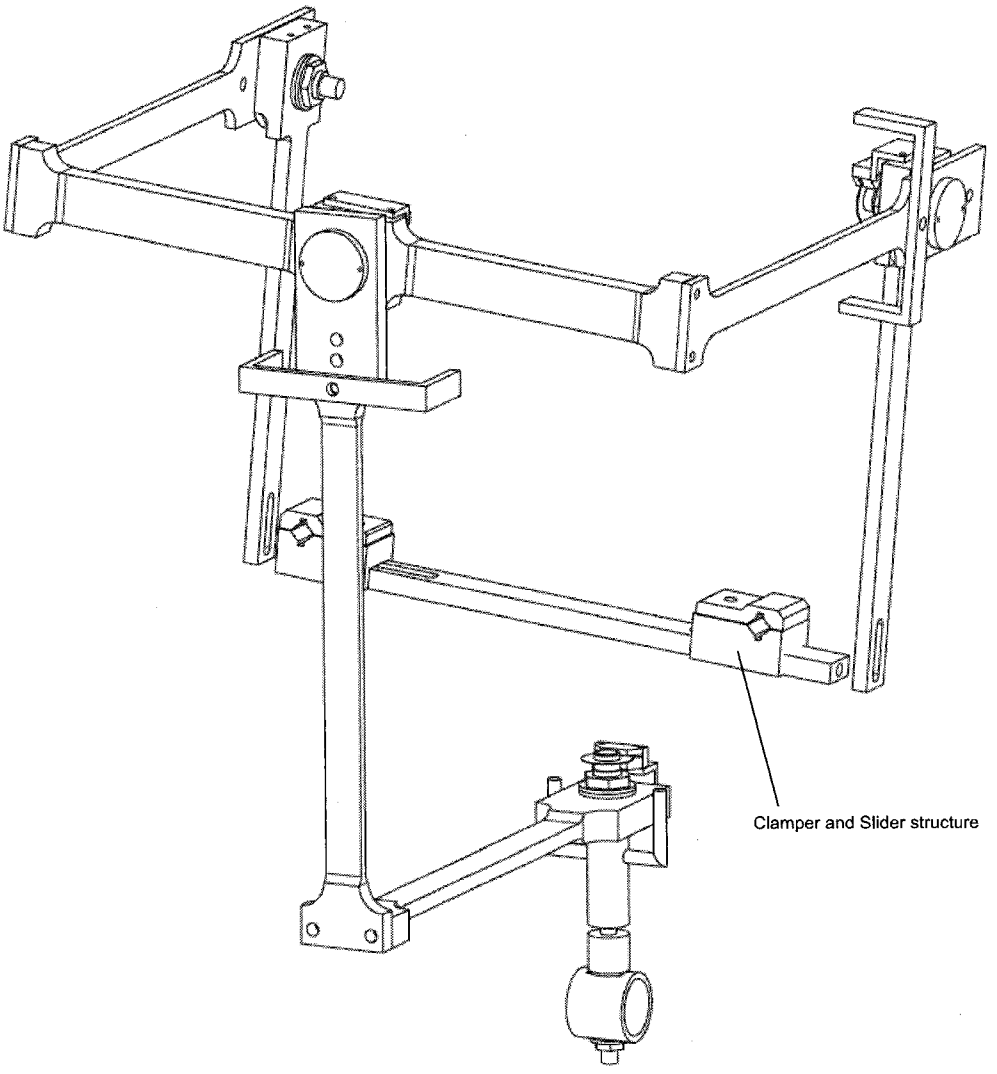

Figure 2.10: The Revised Wrist Section

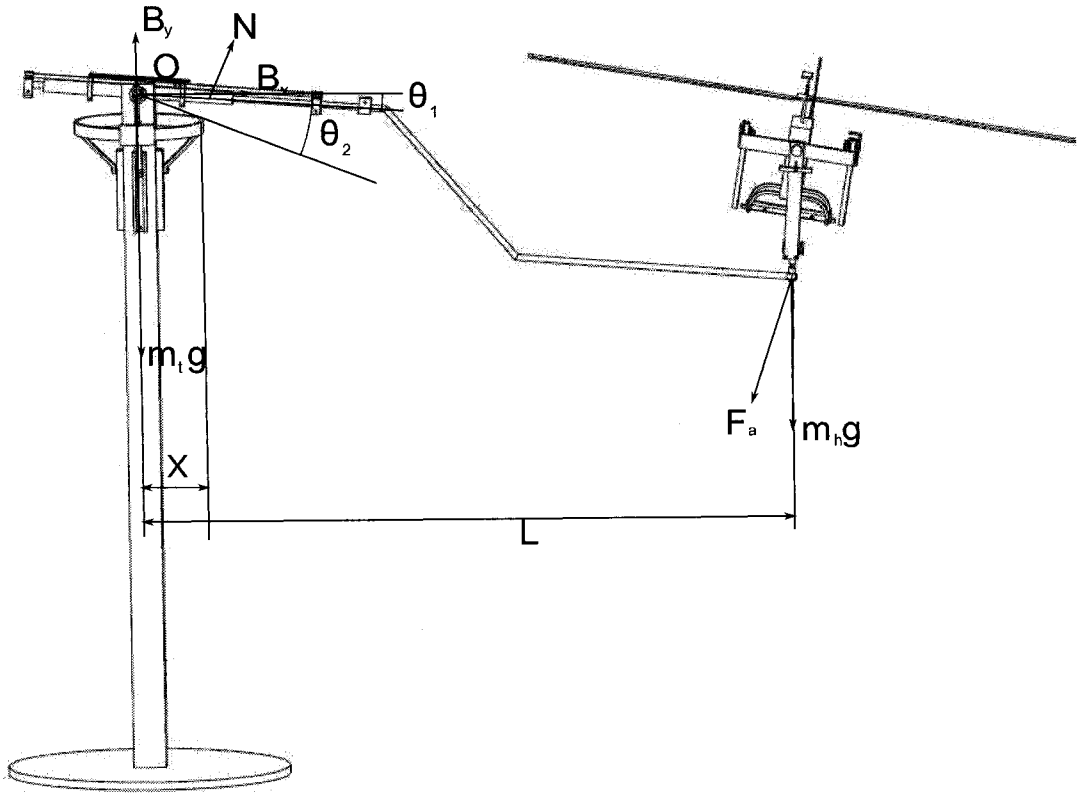

Figure 2.11: The forces on the arm-joint

acceleration in X direction. a_y is the component of acceleration in Y direction. N is the force acting on arm by the ring, and F_s is the spring force.

$$m_h g h = \frac{1}{2} \times 4Kx^2 \Rightarrow K = \frac{m_h g L(\sin\theta_2 + \sin\theta_1)}{2X^2(\tan\theta_2 - \tan\theta_1)^2} \tag{2.19}$$

$$F_s = Kx \Rightarrow F_s = KX(\tan\theta_2 - \tan\theta_1) \tag{2.20}$$

$$N = 4F_s cos\theta_2 \tag{2.21}$$

$$\begin{aligned} a_t &= L\alpha \\ a_x &= a_t \sin\theta_2 = L\alpha \sin\theta_2 \\ a_y &= a_t \cos\theta_2 = L\alpha \cos\theta_2 \end{aligned} \tag{2.22}$$

$$\begin{aligned} \Sigma F_x = 0 &\Rightarrow B_x + N\sin\theta_2 - m_h a_x = 0 \\ B_x &= m_h L\alpha \sin\theta_2 - 4F_s \cos\theta_2 \sin\theta_2 \end{aligned} \tag{2.23}$$

$$\begin{aligned} &\Sigma F_y = 0 \\ &\Rightarrow B_y + N\cos\theta_2 - m_h g - m_h L\alpha\cos\theta_2 - m_t g = 0 \\ &B_y = m_h L\alpha\cos\theta_2 + (m_h + m_t)g - 4F_s\cos^2\theta_2 \end{aligned} \tag{2.24}$$

$$\begin{aligned} &\Sigma M_o = 0 \\ &\Rightarrow m_h(g + a_y)L\cos\theta_2 + m_h a_x L\sin\theta_2 - 4F_s X + I\alpha = 0 \\ \text{where}\quad &I \approx m_{shaft}L_{shaft}^2/12 + m_{tube}L_{tube}^2/12 + m_{tube}L_1^2 = 6\text{Kg}\cdot\text{m}^2 \\ &\alpha = \tfrac{4F_s X - m_h g L\cos\theta_2}{m_h L^2 + 6} \end{aligned} \tag{2.25}$$

In addition to the joint force and spring force, the bending moment in the arm must be considered. The shear force applied on the arm is as shown in Figure 2.12, axial forces are ignored. Based on this, we plotted moment diagram and located danger points A, B as labeled. Point A is located at the maximum bending moment point. Point B is located at the tube connector. The moment at these points can be found from the following equations. The constant 0.62 m is the length of OB at this state:

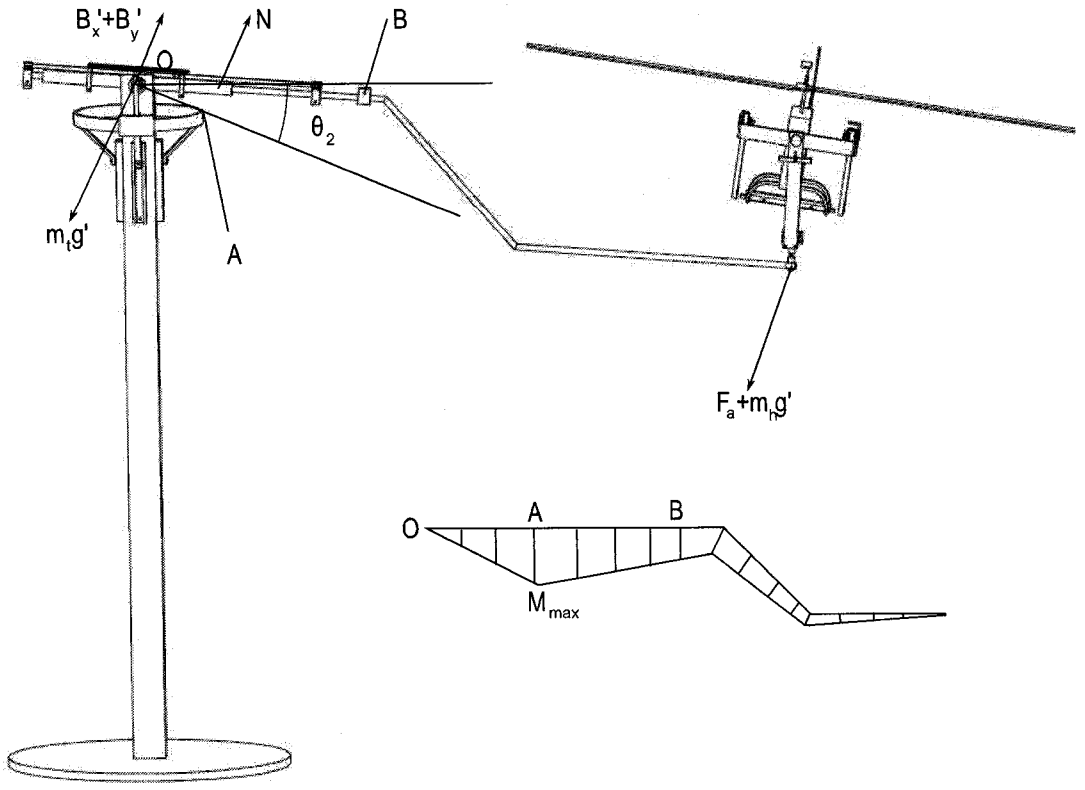

Figure 2.12: Shear Force Deploy and Moment Diagram

At point A:

$$M_{max} = (B_x \sin\theta_2 + B_y \cos\theta_2 - m_t g \cos\theta_2) \times \frac{X}{\cos\theta_2} \tag{2.26}$$

At point B:

$$M_b = (B_x \sin\theta_2 + B_y \cos\theta_2 - m_t g \cos\theta_2) \times 0.62 - N \times (0.62 - \frac{X}{\cos\theta_2}) \tag{2.27}$$

The parameters X and L are assumed to be 8 in and 78.44 in, which in metric are 0.2032 m and 1.9924 m, respectively. The weight of the helicopter is assumed to be full-loaded 8 kg, the weight of the arm including the balance weight is 17.5 kg assuming the density of aluminum is 2.7 Kg/m^3. The above parameters were used with the MATLAB program to solve for an optimized solution for θ_2 and K. The results of the program are shown in Figures 2.13 to 2.19, which demonstrate how the parameters vary with the angles.

In these figures, the ranges for θ_1 and θ_2 have been chosen such that the helicopter was an acceptable spatial range of motion. According to Figures 2.13 and 2.14, the bending moments at points A and B are the lowest when $\theta_1 = 5°$ and $\theta_2 = 20°$. We pick this point as the design point, because the low moments allow us to select smaller cross sections of the arm and reduce the weight of the arm. In figure 2.15, the spring force for the design point ($\theta_1 = 5°$, $\theta_2 = 20°$) is picked and use for stress analysis for the other parts of the arm. Figures 2.16 and 2.17 show the X and Y components of the joint force. Although B_X and B_Y are high at the design point, the stress generated by them is smaller than the strength of the parts. Figure 2.18 shows the upward angular acceleration of the arm just after it is stopped by the springs. Due to the nonliear nature of the equation derived for α, this figure has a minimum. The maximum value of α as a conservative selection of 25 rad/s^2 was selected from this graph to design the arm wrist in the previous section. Figure 2.19 shows the required spring stiffness for the whole range of θ_1 and θ_2. At the design point of $\theta_1 = 5°$ and $\theta_2 = 20°$, the spring constant is 9000 N/m. This value will be used to pick the correct springs.

Here, as shown in Figure 2.20, the optimized value of θ_1, and θ_2 are selected to be 5° and 20° separately.

With the similar process as we discussed for the wrist section, we investigated the strength of the arm (Point A and B) and all the bolts, pins, and bearings, and the results were satisfying. The ring support rib of base section and the inner tube and shaft of arm joint section need to be reinforced. We changed the cross section size of the ring support rib from 12 mm×8 mm to 12 mm×15 mm and changed the outside diameter of the extrusion tube from 19.05 mm (0.75 in) to 34.925 mm (1.375 in). The results of the validated parts are as shown in Table 2.3.

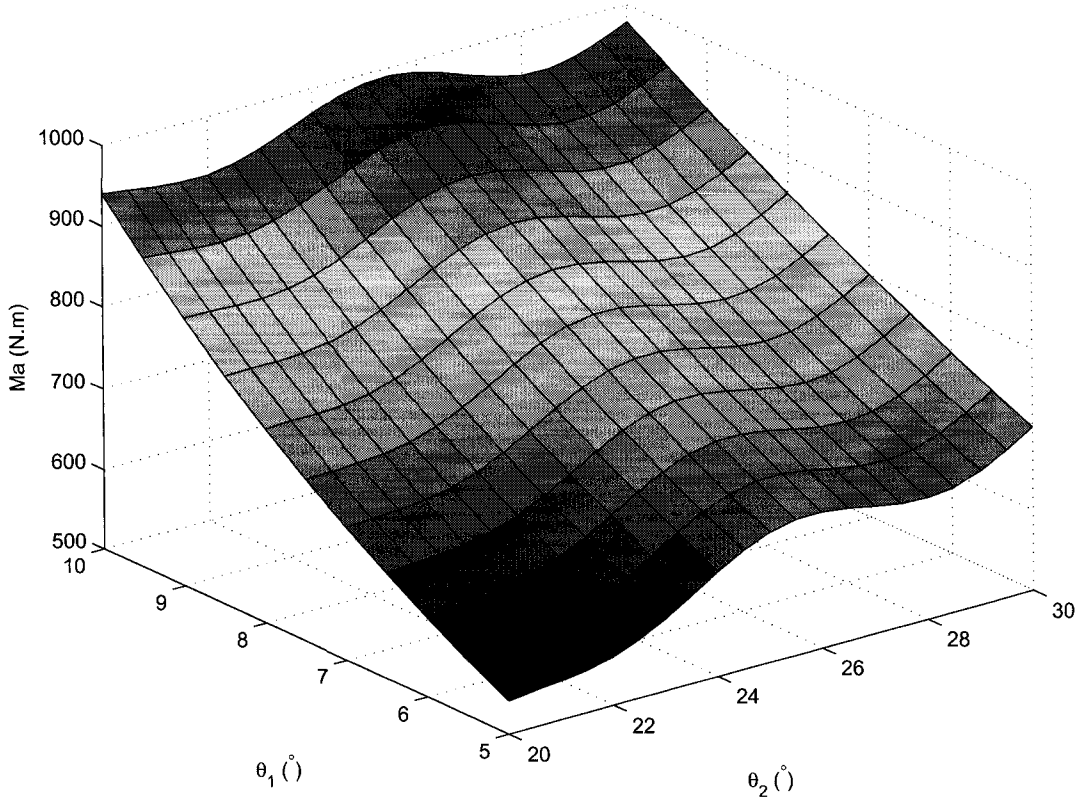

Figure 2.13: The Relation between Moment at Point A and the Angles

Table 2.3: Wrist Strength Validation Results

Part	Force/Stress	Critical Force/Stress	Safety Factor
Bolts on the Trail	33 MPa	400 MPa	12
Bolts on the Ring	33 MPa	400 MPa	12
Ring Support Rib 6	202 MPa	275 MPa	1.36
Shaft of the Tube Coat	235 MPa	275 MPa	1.17
Bolts on the legs	13 MPa	275 MPa	21.15
Point A of the Arm	88 MPa	275 MPa	3.13
Point B of the Arm	239 Mpa	275 MPa	1.15

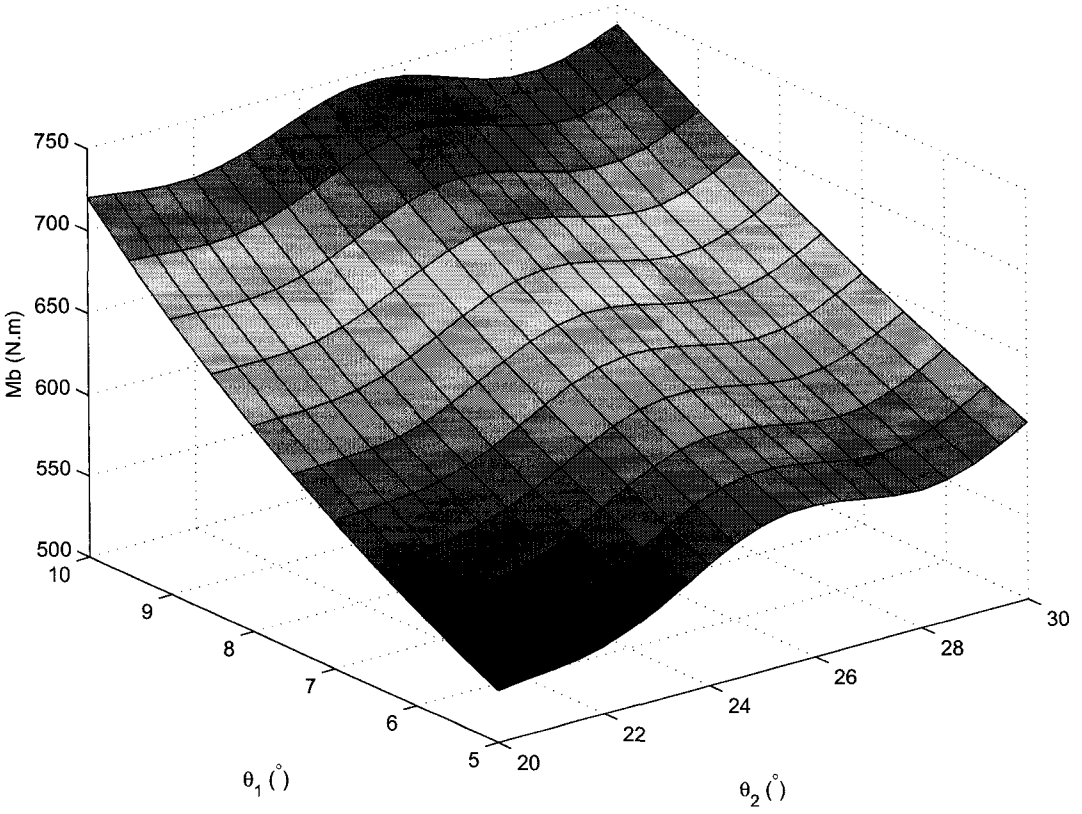

Figure 2.14: The Relation between Moment at Point B and the Angles

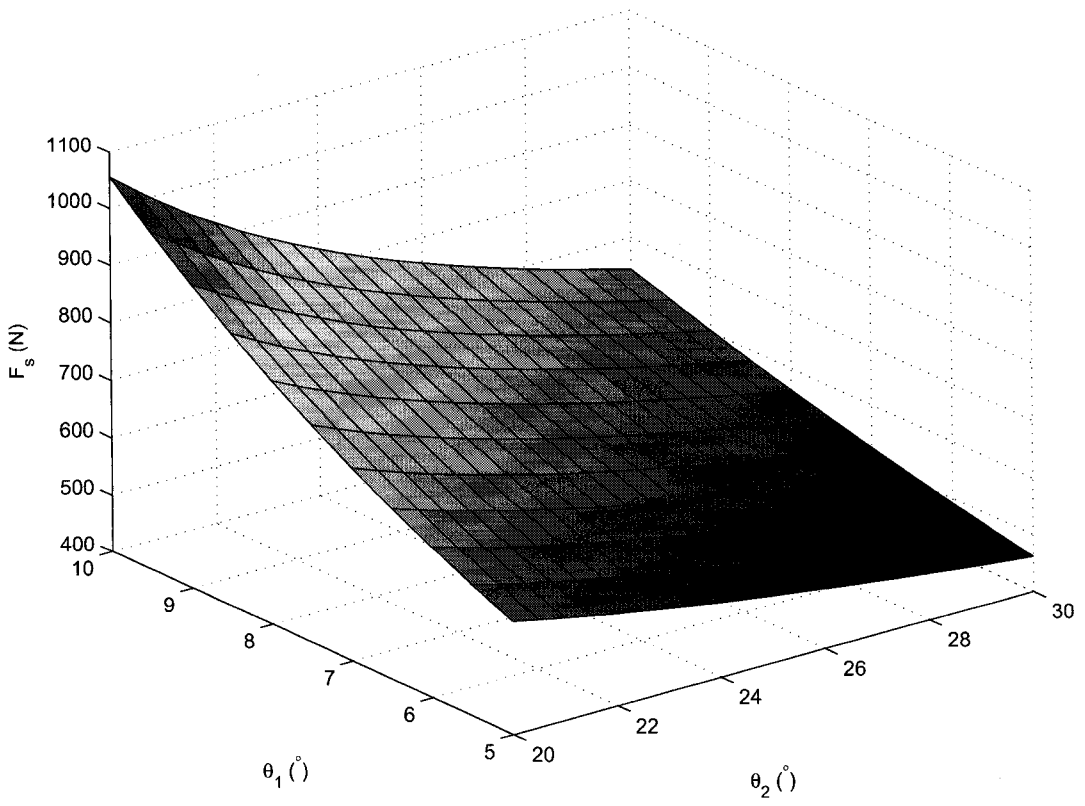

Figure 2.15: The Relation between Maximum Spring Force and the Angles

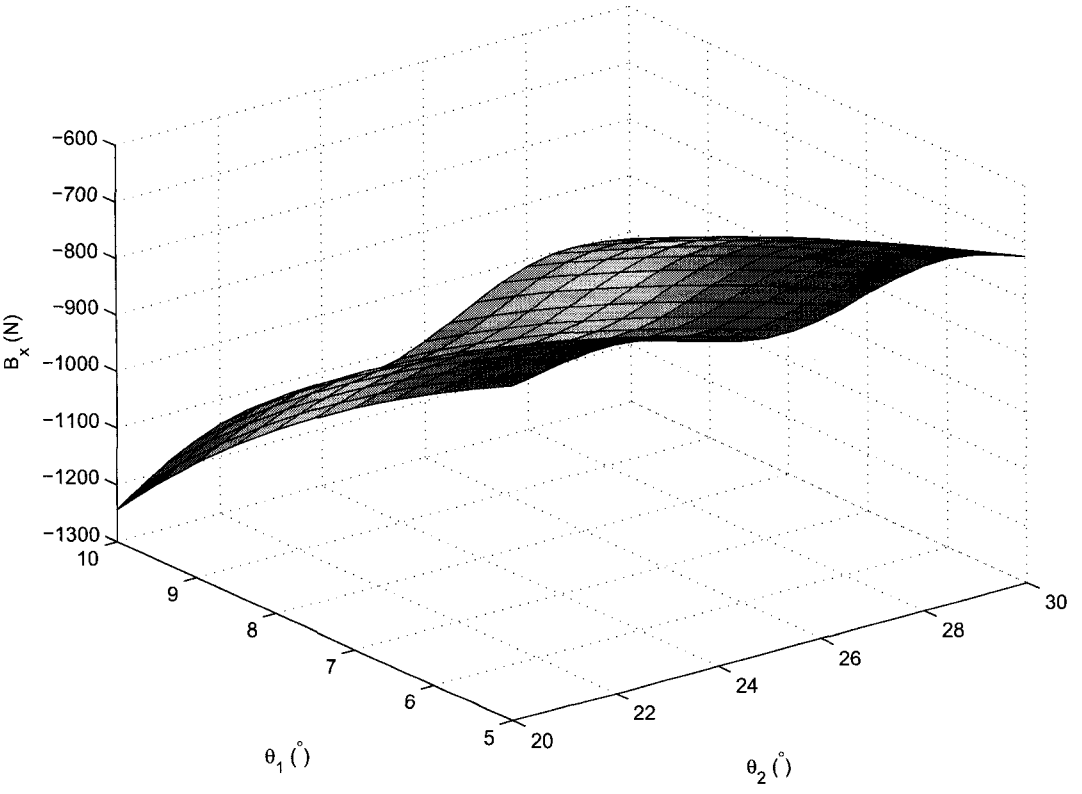

Figure 2.16: The Relation between Joint Force in X Direction and the Angles

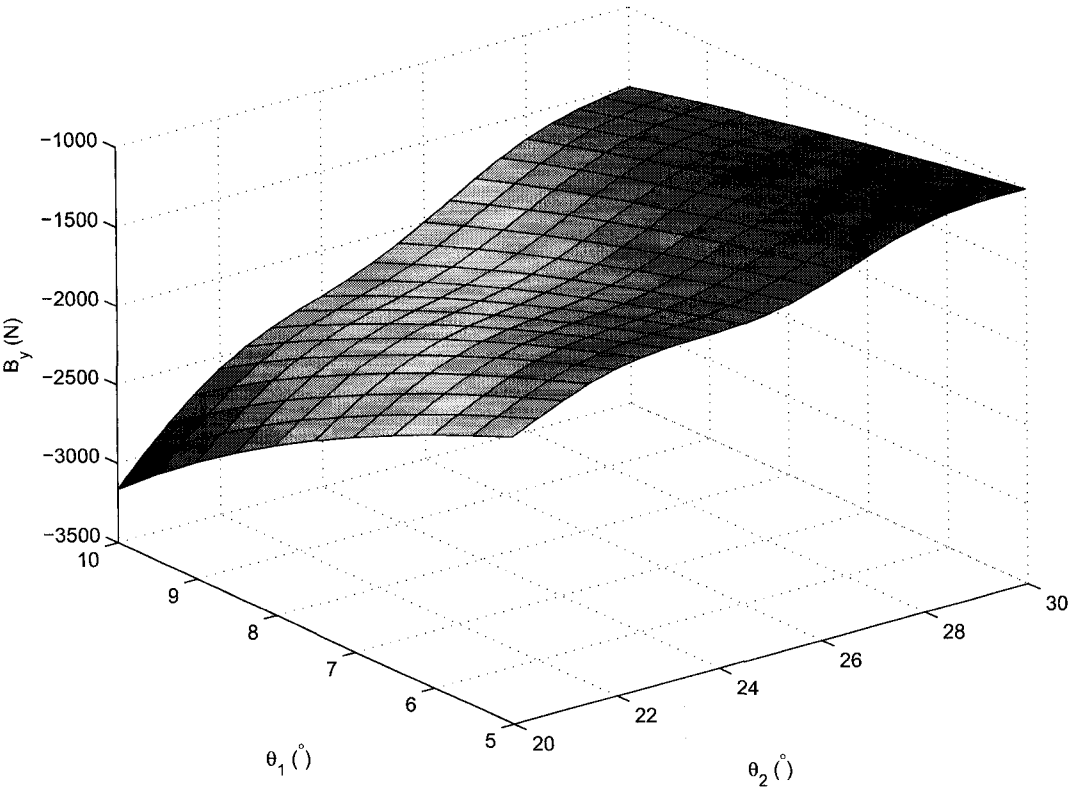

Figure 2.17: The Relation between Joint Force in Y Direction and the Angles

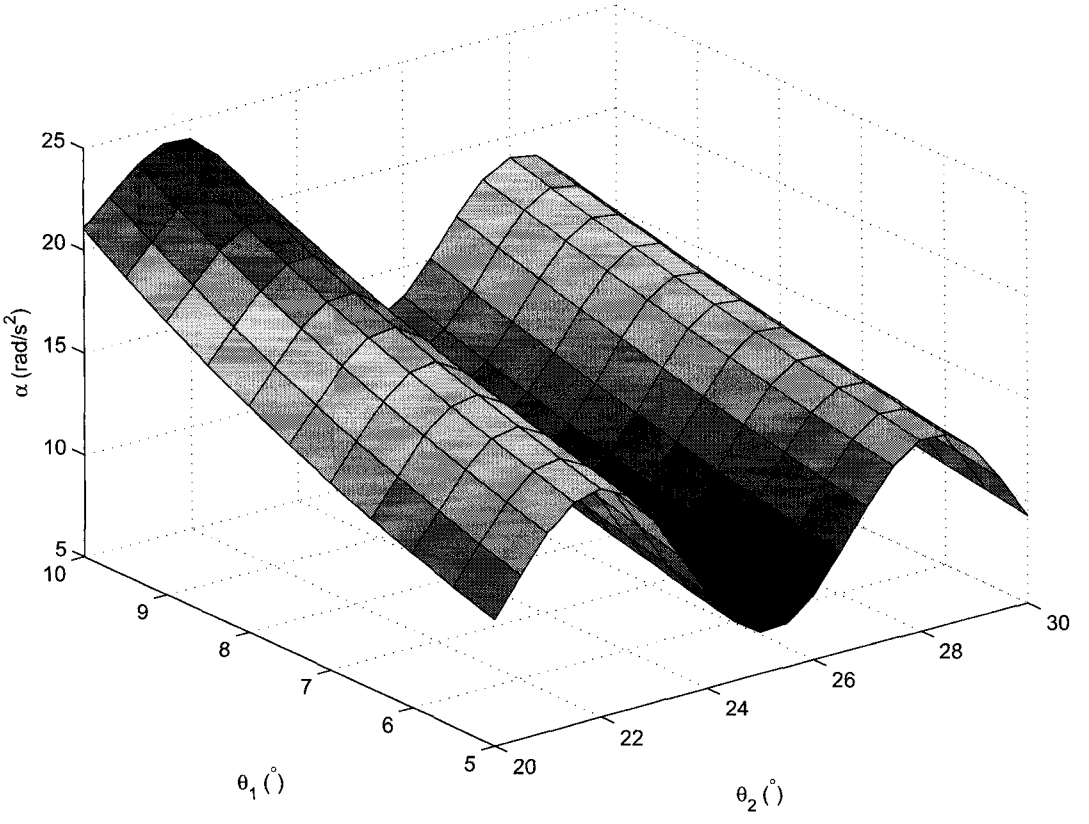

Figure 2.18: The Relation between Angular Acceleration and the Angles

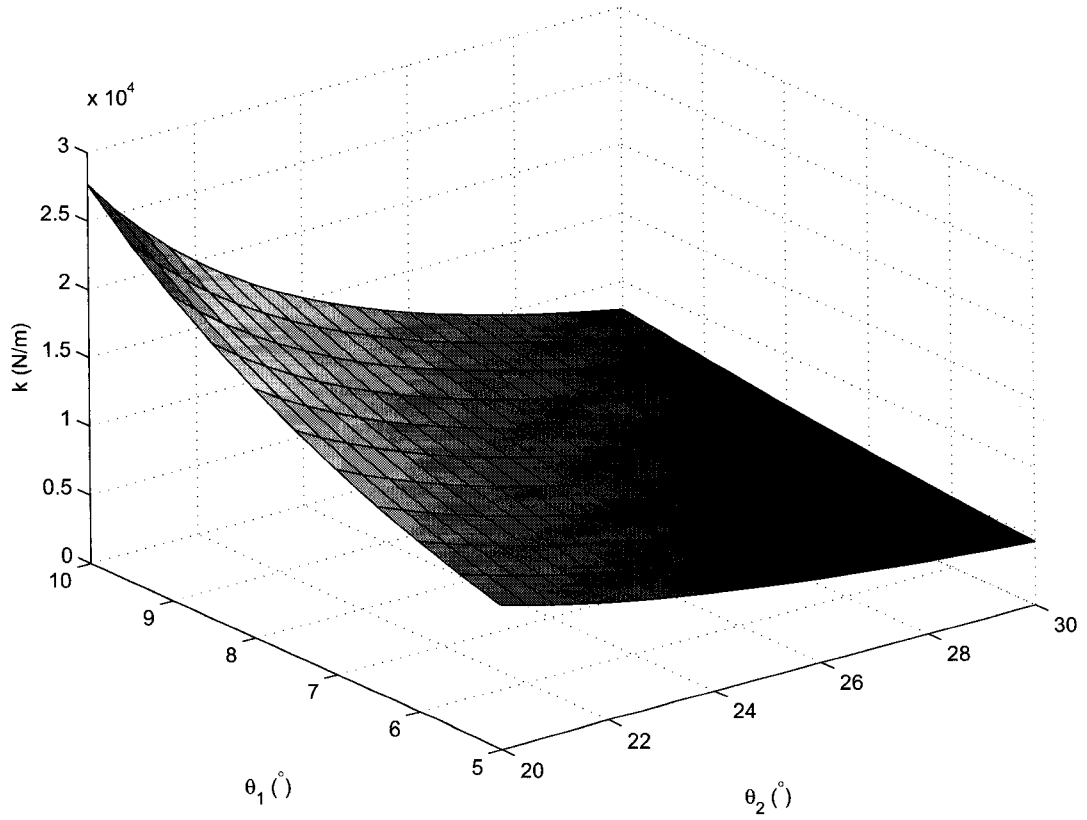

Figure 2.19: The Relation between Spring Stiffness and the Angles

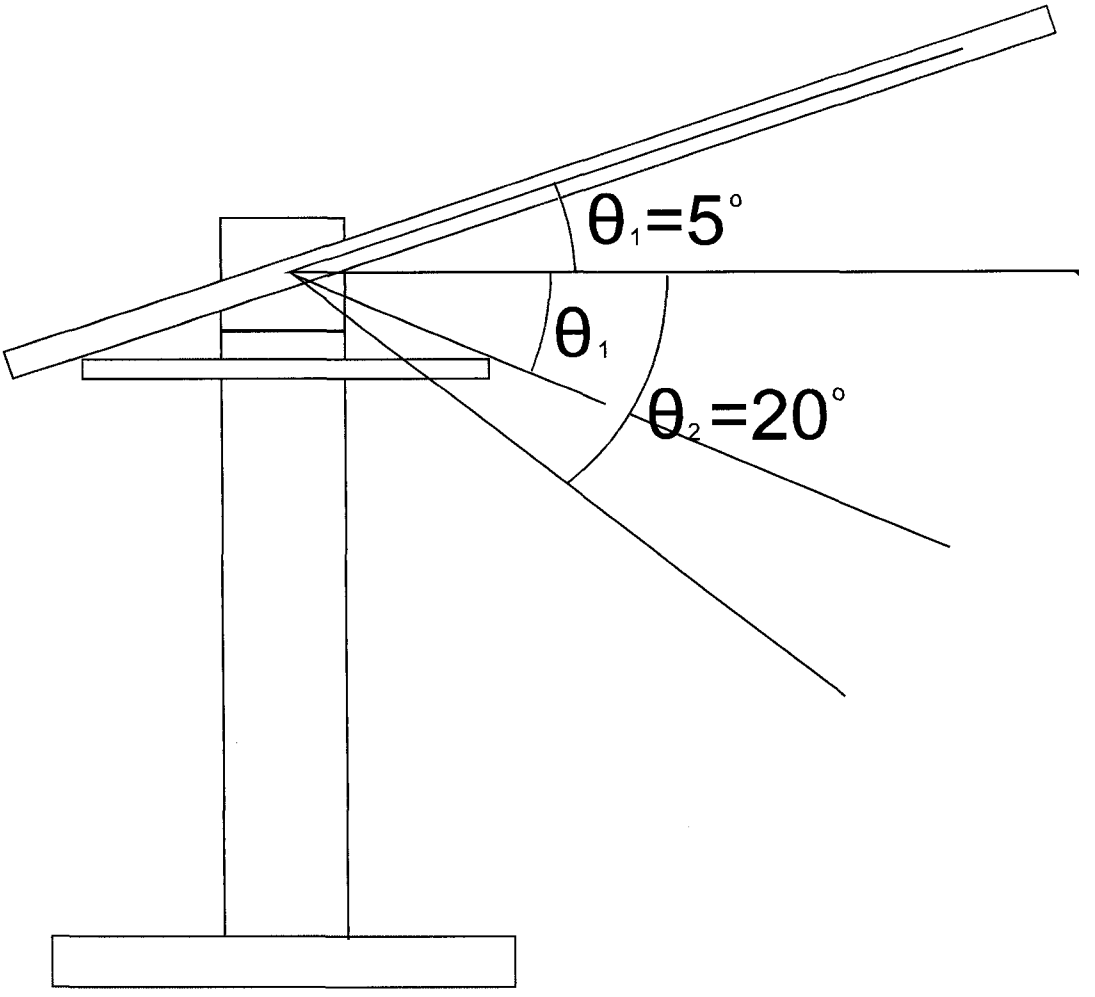

Figure 2.20: Optimized θ_1 and θ_2

2.4 Balancing Counter Weight

Because the weight of the arm is relatively large, when the arm slides along the outer tube, the position of the C.G. of the testbed will change. This will cause the testbed to be unstable at any position except the farthest points. A structure that can always keep the C.G. of the testbed stay coincide of the axis of the straight bar turns to be important.

For this reason, we designed the structure as shown in Figure 2.21. The extension part is attached to the shaft. When shaft slides inside the outer tube, the extension part will move along with it. The balance weight combo is hinged to the extension part. The 3rd part added here is a hinged bar. The bar is hinged to the middle of the straight bar of the balance weight combo. For this structure, when the shaft slides inside the outer arm, the C.G. of the arm $X'_{C.G.}$ will change, while the C.G. of the balance weight $X_{C.G.}$ will move in opposite direction. Hence, the C.G. of the testbed will be always coincide with the axes of the straight bar.

So far, the design of the testbed has been discussed. For final version of all the parts utilized, please refer to A. We are going to construct the data acquisition system in the following chapter.

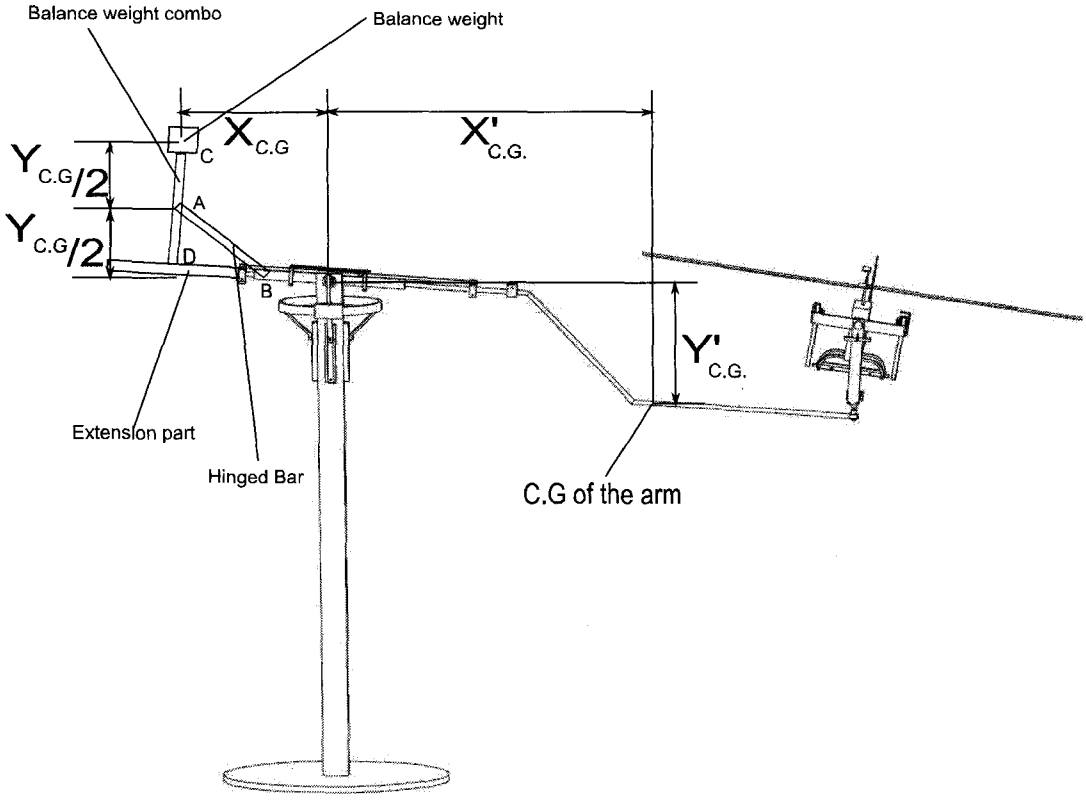

Figure 2.21: Counter Weight Balancing Structure

Chapter 3

Displacement and Velocity Calculation

3.1 Displacement Calculation

As we mentioned in chapter 2, we need to determine the exact motion of the helicopter on the 6 DOFs testbed, which has 5 rotational DOFs and 1 linear DOF.

Because all our calculations are based on the knowledge of robotics, some basic concepts of robotics are introduced here, which could help the reader understand how the formulas for these calculations are derived. For more detailed information about robotics, refer to Figure 3.1. The testbed is regarded to be a manipulator consisting of a set of bodies connected in a chain by joints. The bodies are called links. Each joint usually exhibits one DOF, and the joints can be divided into two categories. The first category is revolute joints, which allow revolution DOFs just like the shaft and accompanied bearings of our testbed. The other category is prismatic joints, which allow translation DOF just like the arm and its sleeve we utilized.

To solve for the needed parameters, the main steps are: First, coordinate frames are affixed to each joint, and the direction of the coordinate axis will be determined according to Denavit-Hartenburg's convention. The second step is to determine the link parameters for each joint, such that, the link transformations can be derived. The forth step is to concatenate link transformations. At last, based on the concatenated transformation, the required 6 DOFs for the helicopter are determined. In the next section, the detail of these steps and the results of the derivations are presented.

3.1.1 Frame Affixing

According to [31], the Denavit-Hartenburg's rules for affixing frames to the joints are: Label Z_i stands for the axis that is coincident with the joint axis. Z_i denotes the

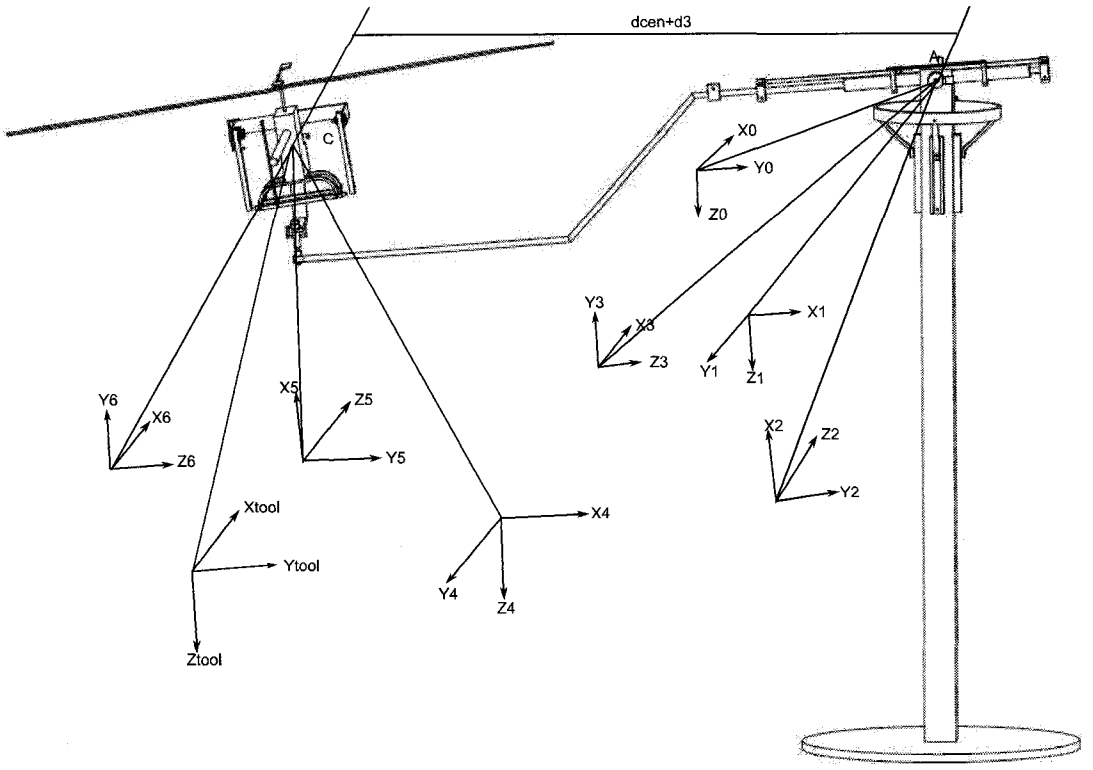

Figure 3.1: Coordinate frame mapping (d_{cen}=2 m)

rotation axis for revolute joint or the trail of the linear motion for a prismatic joint. X_i stands for the axis that is normal to the plane determined by axes Z_i and Z_{i+1}.

To comply with standard axis conventions for aerial vehicles, we added another requirement for the axes of the inertial, frame 0 and the tool frame. They should be at the same orientation at the initial state of the arm, where all joints variables are zero. This rule could let us define the displacement of the tool frame relative to the inertial frame more easily. Figure 3.1 shows the finished mapping of the coordinate frames.

3.1.2 Link Parameters and Transformations

Now, the link parameters for each link must be determined based on the Denavit-Hartenburg's convention, as refer to [31]. The link parameters are shown in Table 3.1.

Table 3.1: Link parameters

i	$a_{i-1}(m)$	$\alpha_{i-1}(deg.)$	$d_i(m)$	$\theta_i(deg.)$
1	0	0°	0	$\theta_1 + 90°$
2	0	90°	0	$\theta_2 - 90°$
3	0	−90°	$-2 - d_3$	−90°
4	0	90°	0	$\theta_4 + 90°$
5	0	90°	0	$\theta_5 - 90°$
6	0	−90°	0	$\theta_6 - 90°$
tool	0	90°	0	0

In this table, i is the joint number; a_i is equal to the distance from axis Z_i to Z_{i+1} measured along X_i; α_i is equal to the angle between axis Z_i and Z_{i+1} measured about X_i; d_i is the distance from X_{i-1} to X_i measured along Z_i and θ_i is the angle between X_{i-1} and X_i measured about Z_i.

Here, the reason why we want the C.G. of the helicopter to be exactly aligned with the axis of the third joint can be explained more clearly. By aligning the C.G. with the axis of the third joint, the origin of the joint frames 4, 5 ,6, and the tool frame are coincide, which makes the value of a_i and d_i to be 0 for all these joints. This significantly simplifies our calculations of the transformations, because the position of the helicopter's C.G. becomes independent of the angles of the last three joints.

The link transformations can be expressed as [31]:

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i \cos\alpha_{i-1} & \cos\theta_i \cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1} d_i \\ \sin\theta_i \sin\alpha_{i-1} & \cos\theta_i \sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1} $$

We can simply substitute the link parameters for each link into the above equation to obtain the link transformation matrices:

$$ {}^{0}_{1}T = \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 & 0 & 0 \\ \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.2} $$

$$ {}^{1}_{2}T = \begin{bmatrix} \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\cos\theta_2 & \sin\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.3} $$

$$ {}^{2}_{3}T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2-d_3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.4} $$

$$ {}^{3}_{4}T = \begin{bmatrix} -\sin\theta_4 & -\cos\theta_4 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.5} $$

$$ {}^{4}_{5}T = \begin{bmatrix} \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\cos\theta_5 & \sin\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6} $$

$$ {}^{5}_{6}T = \begin{bmatrix} \sin\theta_6 & \cos\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \cos\theta_6 & -\sin\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.7} $$

$$ {}^{6}_{tool}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.8} $$

By concatenating link transformations we can find the orientation and position of the tool frame relative to the inertial frame.

$$ {}^{0}_{t}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T\,{}^{3}_{4}T\,{}^{4}_{5}T\,{}^{5}_{6}T\,{}^{6}_{t}T \tag{3.9} $$

where ${}^{6}_{t}T$ is the transformation matrix between frame 6 and the tool frame, and ${}^{0}_{t}T$ is the transformation matrix between the inertial frame, 0, and the tool frame. For more details, see Reference [31]

3.1.3 Displacement Parameters

Now that the transformations are found, the position (X, Y, Z) and orientation (ϕ, θ, ψ) parameters of the helicopter as the end-effector should be derived. In robotics, the orientation of a frame can be defined by using Euler angles. A convention for describing the orientation of a helicopter is using the three orientations: roll, pitch, and yaw angles. The rotation about X is defined by γ, the rotation about Y is defined by β, and about Z is defined by α. The orientation of a frame B relative

to another frame A can be generated by three successive rotations starting from the position at which the two frames are coincident:

$$
\begin{aligned}
{}^{A}_{B}R(\phi,\theta,\psi) &= ROT({}^{A}Z_A,\psi)ROT({}^{A}Y_A,\theta)ROT({}^{A}X_A,\phi) \\
&= \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}
\end{aligned} \tag{3.10}
$$

Hence,

$$
{}^{A}_{B}R(\phi,\theta,\psi) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3.11}
$$

where:

$$
\begin{aligned}
r_{12} &= \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi \\
r_{13} &= \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\
r_{21} &= \sin\psi\cos\theta \\
r_{22} &= \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi \\
r_{23} &= \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\
r_{31} &= -\sin\theta \\
r_{32} &= \cos\theta\sin\phi \\
r_{33} &= \cos\theta\cos\phi
\end{aligned} \tag{3.12}
$$

And the Euler angles can be computed as:

$$
\begin{aligned}
\phi &= \tan^{-1}(r_{32}/r_{33}) \\
\theta &= \tan^{-1}(-r_{31}/\sqrt{r_{11}^2 + r_{21}^2}) \\
\psi &= \tan^{-1}(r_{21}/r_{11})
\end{aligned} \tag{3.13}
$$

Note that in the MATLAB program the atan2 function must be used so that the correct quadrant of the angle is found.

As a transformation, the matrix ${}^{0}_{t}T$ we got is a 4×4 matrix, and this matrix can also be expressed as:

$$ {}^{0}_{t}T = \begin{bmatrix} {}^{0}_{t}R & {}^{0}_{t}P \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.14} $$

where ${}^{0}_{t}R$ is the orientation of tool frame relative to the inertial frame, and ${}^{0}_{t}P$ is a $3\times$ 1 position vector. Combining equations (3.11), (3.13), and (3.14), we can get the displacement parameters:

$$ \begin{bmatrix} X \\ Y \\ Z \\ \gamma \\ \beta \\ \alpha \end{bmatrix} = \begin{bmatrix} t_{14} \\ t_{24} \\ t_{34} \\ \tan^{-1}(t_{32}/t_{33}) \\ \tan^{-1}(-t_{31}/\sqrt{t_{11}^2 + t_{21}^2}) \\ \tan^{-1}(t_{21}/t_{11}) \end{bmatrix} \tag{3.15} $$

3.2 Velocity Calculation

The other parameters that have to be calculated besides the displacements are the linear and angular velocities of the helicopter. These velocities must be calculated in terms of the testbed's joint rates.

3.2.1 Velocity Formulation Derivation

As shown in Figure 3.2, ω_i is the angular velocity vector of the i^{th} joint, u_i is the linear velocity vector of the i^{th} joint. Based on kinematics, for revolute joints, the relation of linear and angular velocities of neighboring joints can be formulated as:

$$ {}^{i}\omega_{i+1} = {}^{i}\omega_i + {}^{i}_{i+1}R\dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{3.16} $$

$$ {}^{i}u_{i+1} = {}^{i}u_i + {}^{i}\omega_i \times {}^{i}P_{i+1} \tag{3.17} $$

where i on the left top of the velocity means the velocity is measured in the i^{th} frame. By premultiplying ${}^{i+1}_{i}R$ to both sides of the equations (3.14), (3.17), the desired form is obtained as below:

$$ {}^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega_i + \dot{\theta}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{3.18} $$

$$ {}^{i+1}u_{i+1} = {}^{i+1}_{i}R({}^{i}u_i + {}^{i}\omega_i \times {}^{i}P_{i+1}) \tag{3.19} $$

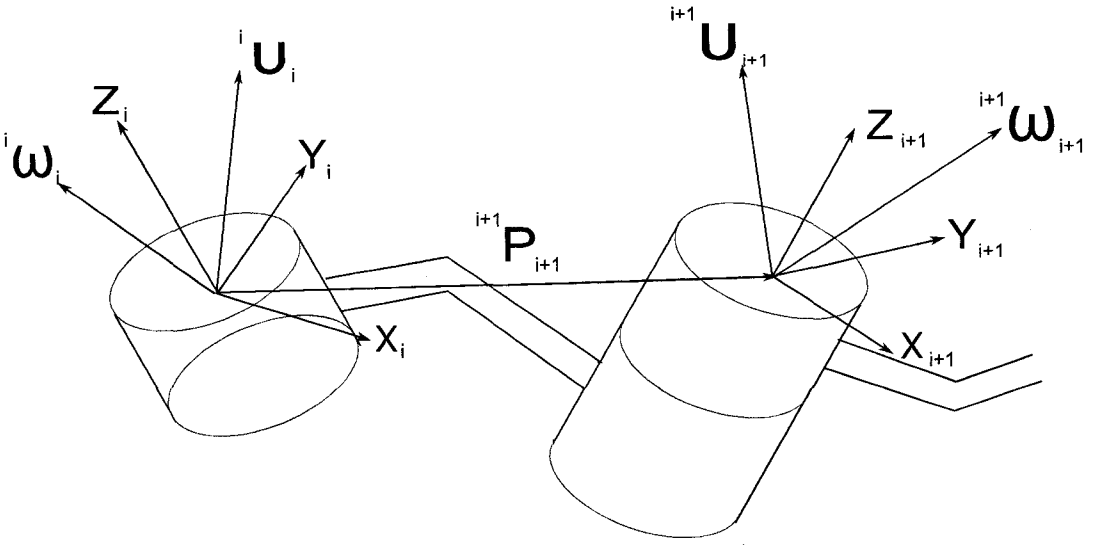

Figure 3.2: Velocity of Connected Joints

Similarly, we can derive the formulations of a prismatic joint as shown below:

$$^{i+1}\omega_{i+1} = {}^{i+1}_{i}R\,{}^{i}\omega_{i} \tag{3.20}$$

$$^{i+1}u_{i+1} = {}^{i+1}_{i}R(^{i}u_{i} + {}^{i}\omega_{i} \times {}^{i}P_{i+1}) + \dot{d}_{i+1}\,{}^{i+1}\hat{Z}_{i+1} \tag{3.21}$$

3.2.2 Velocity Calculation

As required by the formulations (3.18), (3.19), (3.20),and (3.21), ${}^{i}_{i+1}R$ is needed for the velocity calculations. It is known that in ${}^{i+1}_{i}T$, the first 3 rows and 3 columns construct the matrix ${}^{i+1}_{i}R$. Here, the inverse matrix of each ${}^{i+1}_{i}R$ is used to get ${}^{i}_{i+1}R$. The other required parameters are $\dot{\theta}_i$ and $\dot{d}$. These values are determined by differentiating the signals acquired from encoders.

Thus, following formulations are obtained

$$^{0}\omega_{0} = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \tag{3.22}$$

$$^{0}v_{0} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.23}$$

$$ {}^{1}\omega_1 = \begin{bmatrix} -\sin\theta_1 & \cos\theta_1 & 0 \\ -\cos\theta_1 & -\sin\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot {}^{0}\omega_0 + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_1 \end{bmatrix} \tag{3.24} $$

$$ {}^{1}v_1 = {}^{0}v_0 + {}^{0}\omega_0 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.25} $$

$$ {}^{2}\omega_2 = \begin{bmatrix} \sin\theta_2 & 0 & -\cos\theta_2 \\ \cos\theta_2 & 0 & \sin\theta_2 \\ 0 & -1 & 0 \end{bmatrix} \cdot {}^{1}\omega_1 + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_2 \end{bmatrix} \tag{3.26} $$

$$ {}^{2}v_2 = {}^{1}v_0 + {}^{1}\omega_1 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.27} $$

$$ {}^{3}\omega_3 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot {}^{2}\omega_2 \tag{3.28} $$

$$ {}^{3}v_3 = {}^{2}v_2 + {}^{2}\omega_0 \cdot \begin{bmatrix} 0 \\ 0 \\ -2 - d_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{d}_3 \end{bmatrix} \tag{3.29} $$

$$ {}^{4}\omega_4 = \begin{bmatrix} -\sin\theta_4 & 0 & \cos\theta_4 \\ -\cos\theta_4 & 0 & -\sin\theta_4 \\ 0 & -1 & 0 \end{bmatrix} \cdot {}^{3}\omega_3 + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_4 \end{bmatrix} \tag{3.30} $$

$$ {}^{4}v_4 = {}^{3}v_3 + {}^{3}\omega_3 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.31} $$

$$ {}^{5}\omega_5 = \begin{bmatrix} \sin\theta_5 & 0 & -\cos\theta_5 \\ \cos\theta_5 & 0 & \sin\theta_5 \\ 0 & -1 & 0 \end{bmatrix} \cdot {}^{4}\omega_4 + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_5 \end{bmatrix} \tag{3.32} $$

$$ {}^{5}v_5 = {}^{4}v_4 + {}^{4}\omega_4 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.33} $$

$$ {}^{6}\omega_6 = \begin{bmatrix} \sin\theta_6 & 0 & \cos\theta_6 \\ \cos\theta_6 & 0 & -\sin\theta_6 \\ 0 & 1 & 0 \end{bmatrix} \cdot {}^{5}\omega_5 + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_6 \end{bmatrix} \tag{3.34} $$

$$^{6}v_6 = {}^{5}v_5 + {}^{5}\omega_5 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.35}$$

$$^{t}\omega_t = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \cdot {}^{6}\omega_6 \tag{3.36}$$

$$^{t}v_t = {}^{6}v_6 + {}^{6}\omega_6 \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{3.37}$$

Where $^{t}\omega_t$ and $^{t}v_t$ are the angular velocity and linear velocity of the helicopter (end-effector) described in the tool frame.

Now, the velocities of the tool frame expressed in the tool frame are determined, however, the final data needed are the linear velocities expressed in the inertial frame. Refer to [32], based on the already known Euler angles of the tool frame, one can construct the following matrix A:

$$A = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{3.38}$$

So the angular velocity of tool the frame expressed in the inertial frame turns to be:

$$^{0}\omega_t = A^{-1} \cdot {}^{6}\omega_6 \tag{3.39}$$

For the linear velocity, we can simply use the following equations:

$$^{0}_{t}R = {}^{0}_{1}R^{1}_{2}R^{2}_{3}R^{3}_{4}R^{4}_{5}R^{5}_{6}R^{6}_{t}R \tag{3.40}$$

$$^{0}v_t = {}^{0}_{t}R \cdot {}^{t}v_t \tag{3.41}$$

3.3 Programming and Adjustment in MATLAB

To check the correctness of the displacement and velocity calculations, we coded a program in MATLAB, which can reach the results very fast and can be conveniently adjusted for newer designs. To make sure the program provides the correct results, we built a Simulink model as shown in Figure 3.3. Our program is embedded into a simulink block as a function. The 6 sine wave generators are used to simulate the

encoder inputs. The input signals are sent directly into the embedded function. The outputs are the three helicopter position component X, Y, Z, the three Euler angles ϕ, θ, ψ, the three linear other velocity components $\dot{X}$, $\dot{Y}$, $\dot{Z}$, and the three body angular velocity components ${}^0\omega_{tx}$, ${}^0\omega_{ty}$, ${}^0\omega_{tz}$.

We run the model while enabling only one of the generators at a time and disabling the others by setting their inputs to 0. Then we observe how the results are affected when the input signal changes. Since the inputs are isolated, we can easily interpret the results and make sure they are correct. As an example, the affected plots of the outputs when only the first function generator is enabled is shown as in Figures 3.4 to 3.7.

Since the input signal is a sine wave that ranges from -1 rad to 1 rad, the input variable θ_1 of our model is thus changing from -1 to 1 according to a sine wave.

We should investigate the position parameters as shown in Figure 3.4 and 3.5. We should note that: the position and velocity components are not changing as a sine wave. They follow these rules:

$$X = 2 \times \sin\theta_1 = 2 \times \sin(\theta_{1max} \sin t)$$
$$Y = 2 \times \cos\theta_1 = 2 \times \cos(\theta_{1max} \sin t)$$
$$\dot{X} = 2\theta_{1max} \times \cos(\theta_{1max} \sin t) \cos t \tag{3.42}$$
$$\dot{Y} = -2\theta_{1max} \times \sin(\theta_{1max} \sin t) \cos t \tag{3.43}$$

Where θ_{1max} is equal to 1. We can see the plot is essentially complicated. To make the interpretation easier, we simply replace the input sine wave by a ramp signal, and we can reach the desired sine and cosine wave as shown in Figures 3.8 and 3.9.

Now, by looking at Figure 3.6 and 3.7, we can see the Euler angle indicating the rotation about Z axis is the only angle affected by the changing of θ_1, which was expected, also when the angle is changing like a sine wave, the angular velocity is varying as a cosine wave, which also agrees with our model. Until now, we have finished the theoretical calculation part, and we are going to present the experimental phase in the next chapter.

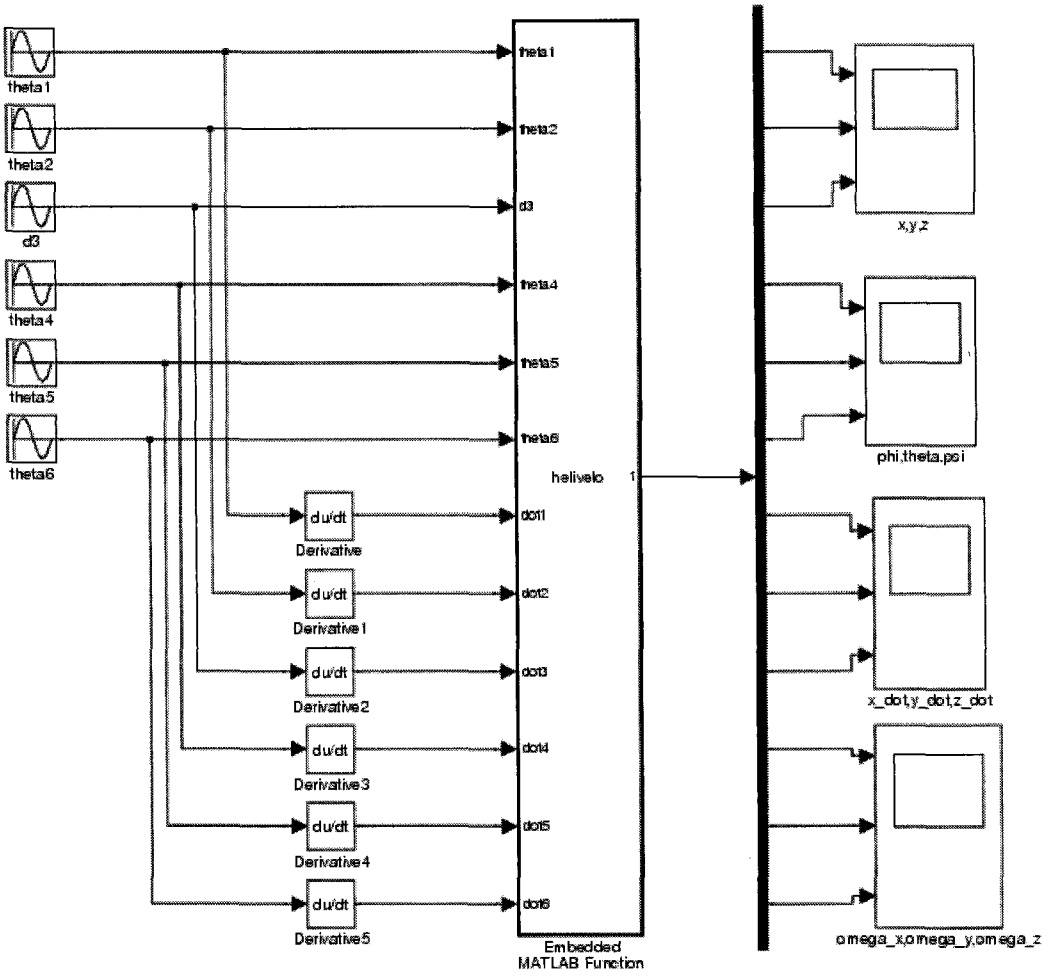

Figure 3.3: Simulink Model Layout

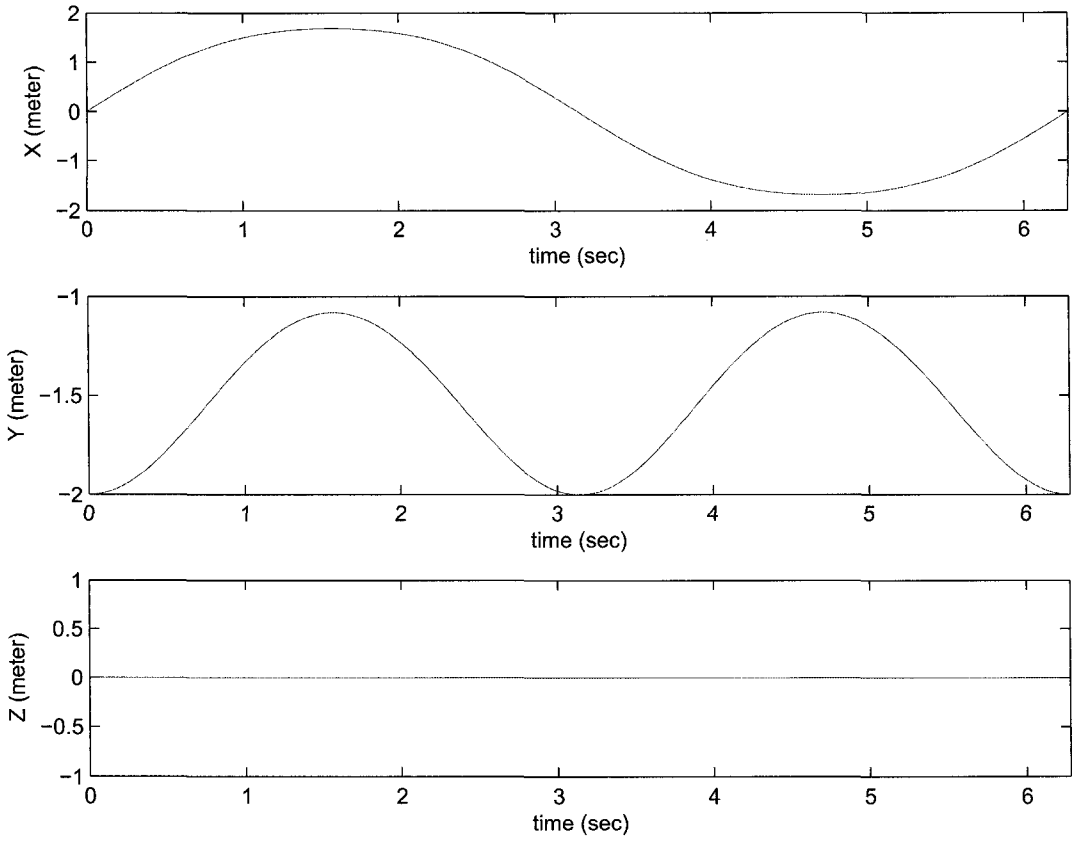

Figure 3.4: The position of the end-effector when the first joint rotates with a sine profile

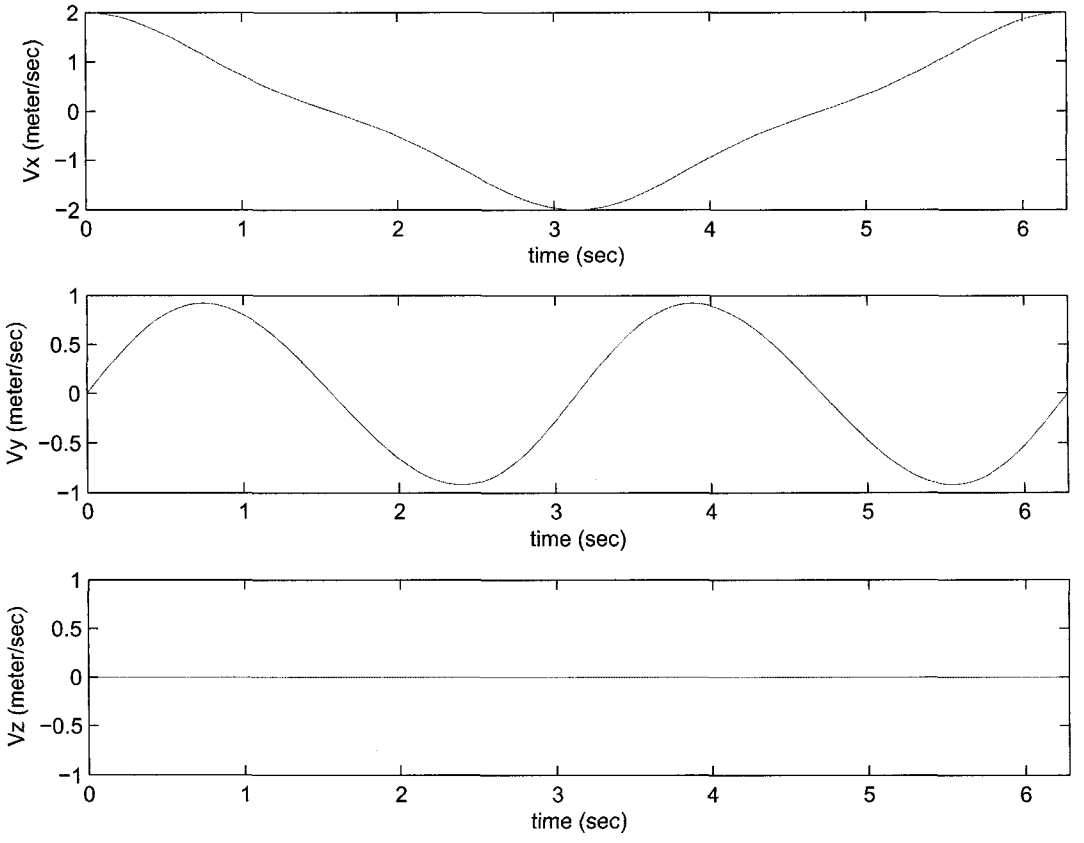

Figure 3.5: The linear velocity of the end-effector when the first joint rotates with a sine profile

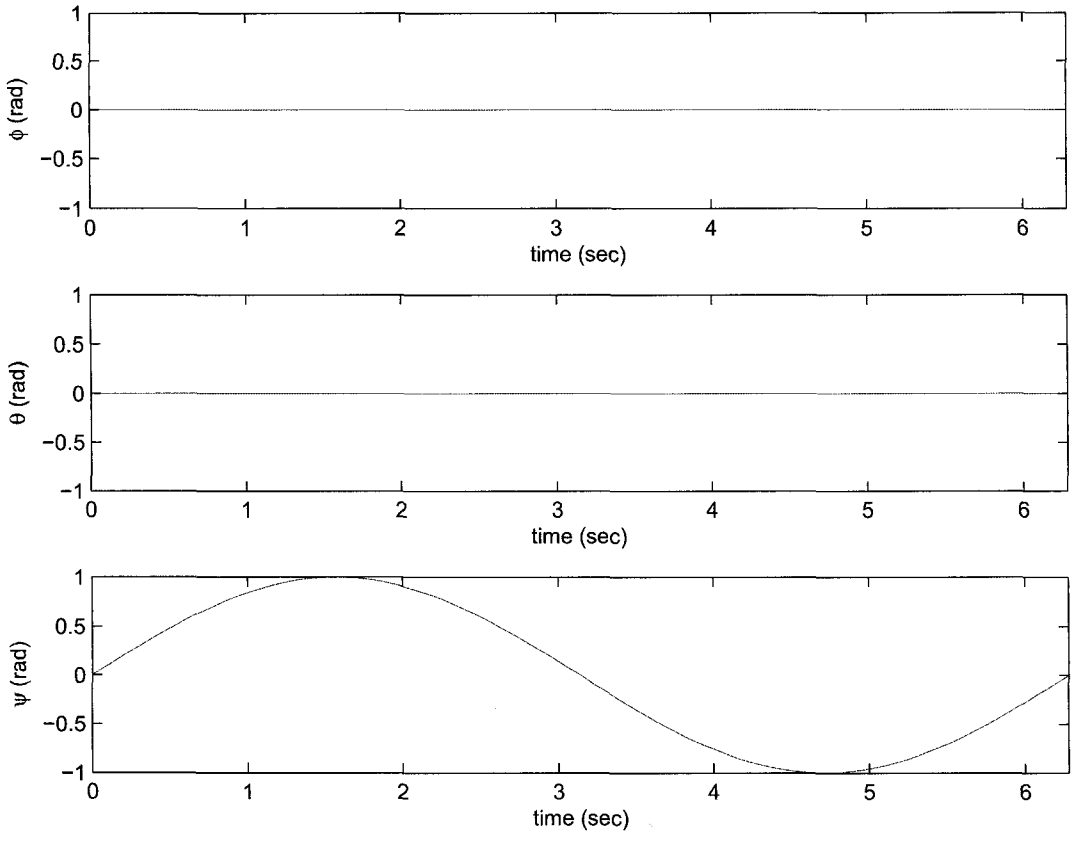

Figure 3.6: The Euler angles of the end-effector when the first joint rotates with a sine profile

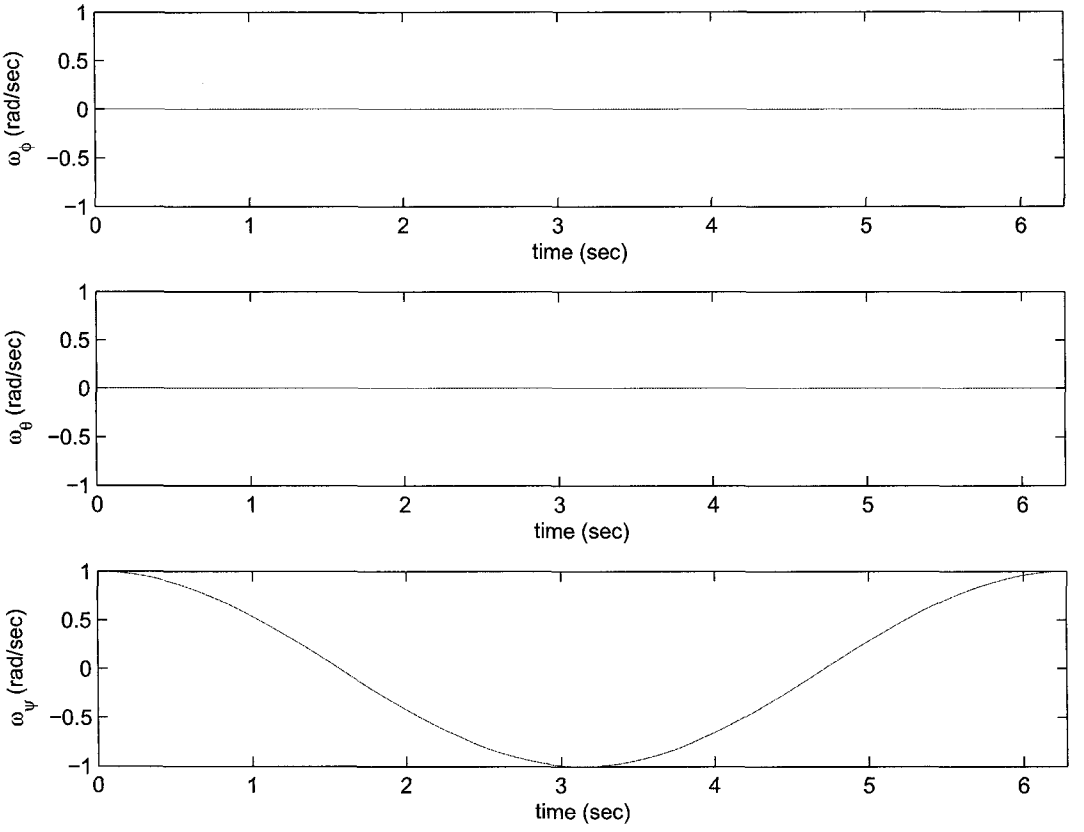

Figure 3.7: The Angular velocity of the end-effector when the first joint rotates with a sine profile

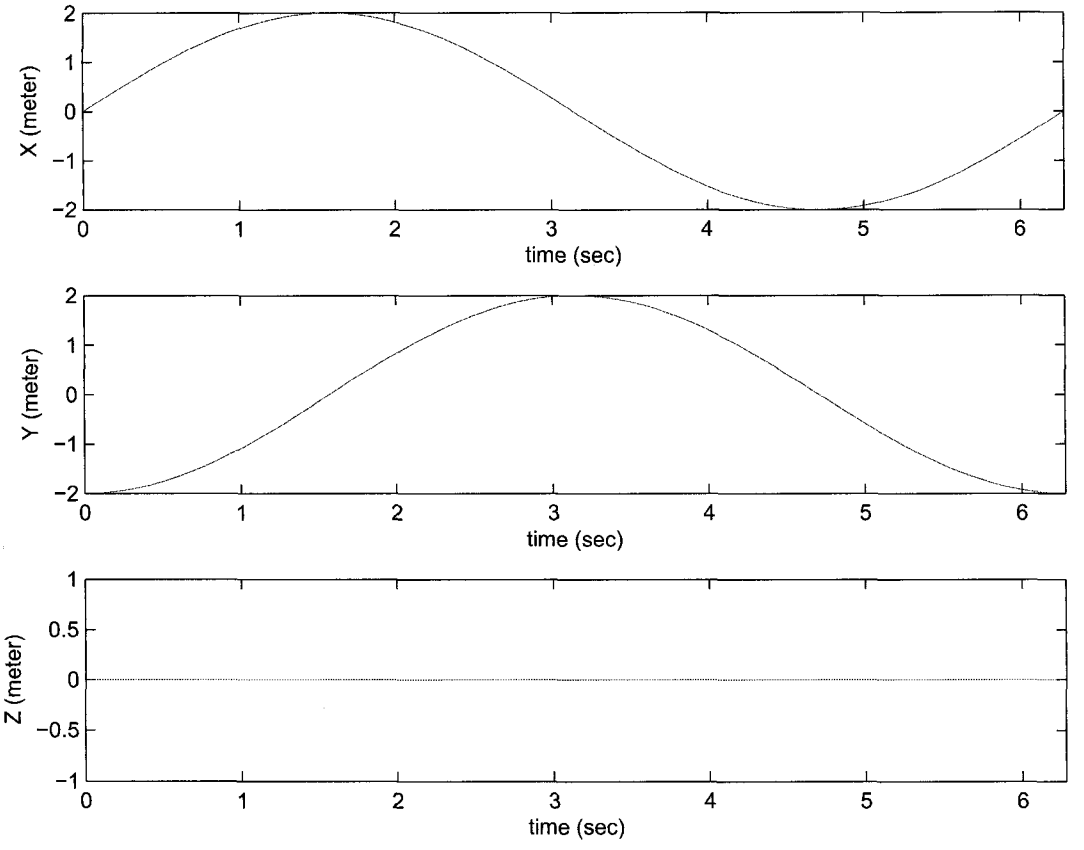

Figure 3.8: The position of the end-effector when the first joint rotates with a ramp profile

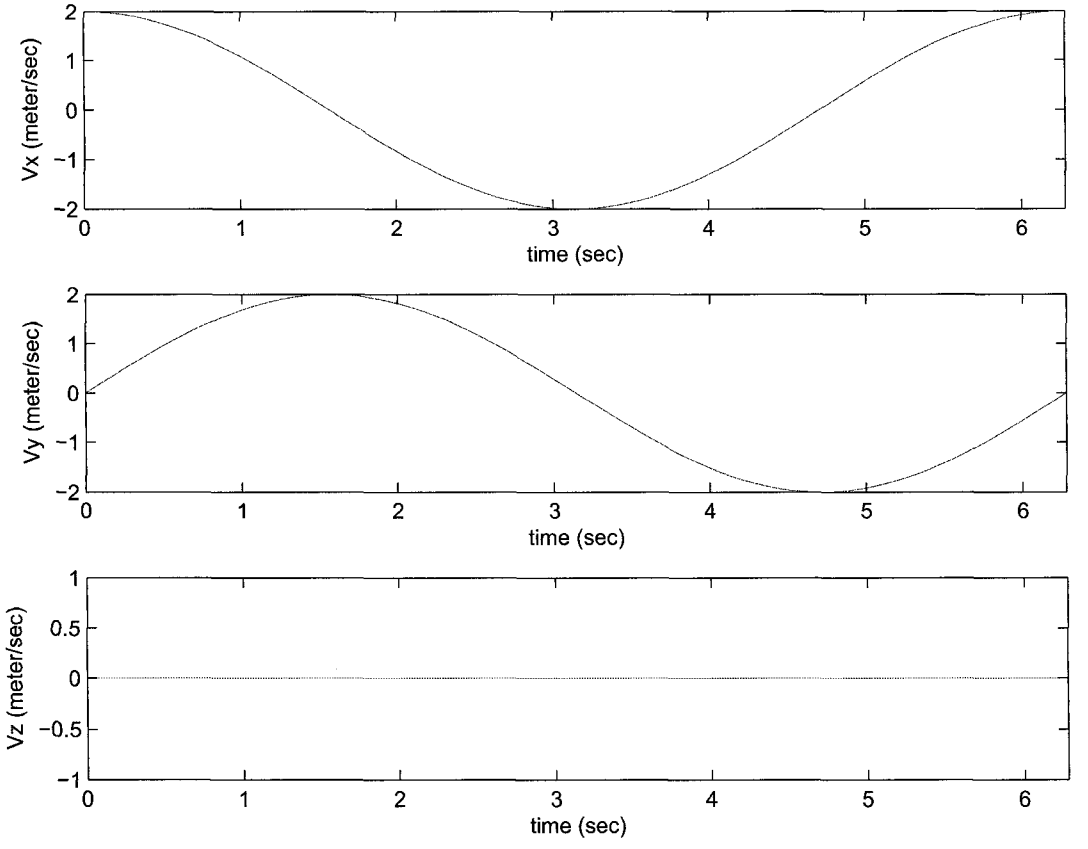

Figure 3.9: The linear velocity of the end-effector when the first joint rotates with a ramp profile

Chapter 4

xPC Target Hardware and Software

To test how our data acquisition system works, we intend to develop a data system according to Figure 4.1. The encoders are mounted on the testbed. They will generate pulse signals and transmit them to the counter board PCI6602 through the terminal board CB-68LP. PCI6602 is installed on the target PC, and the target PC can communicate with the host PC by local area network (LAN).

4.1 Hardware Parts Introduction

Generally speaking, the encoders should be able to collect the signals and send them to control computer. Then a program will calculate the displacements and velocitics.

Our hardware utilized to carry out this assignment contains: A host PC which runs MATLAB connected to LAN; an xPC target PC which can be booted by a MATLAB xPC target boot disk and is also connected to LAN; a PCI6602 board from National instrument; a CB-68LP terminal board from National Instrument; and 6 Encoders from USdigital.

4.1.1 xPC Host and Target PC

Our control system is based on the utilization of MATLAB xPC target function. xPC Target is a solution for prototyping, testing, and deploying real-time systems. By constructing a host/target computer environment, we can setup variety of boards on the target PC and do sorts of tasks such as data acquisition, signal generation and so on.

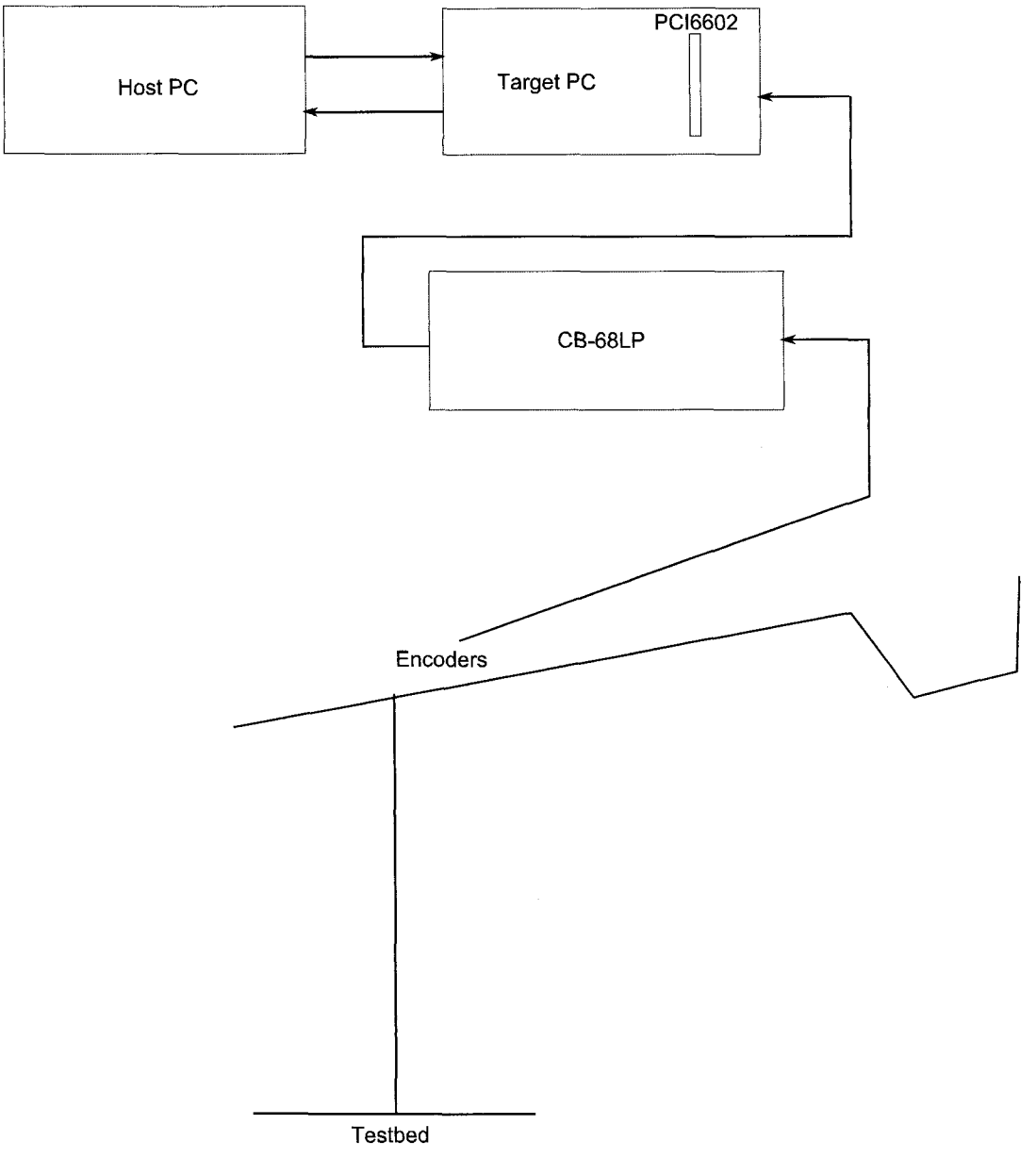

Figure 4.1: Encoder Data System Outline

MATLAB is installed on the host PC, and can fulfil all the MATLAB functionalities. We make the program and simulation model on the host PC. The target PC is basically a computer without operating system installed, the necessary boot information is from a boot disk made on the host PC. Then, once we power on the target PC, we could reach it by xPC explorer from the host PC, and download the models to it. Then we can run the model on the target PC to verify the program abd the modeling. Real-time monitoring and data collection are available.

4.1.2 Boards

To do a specific job, we need specific board. Here for data acquisition purpose, we purchased the board PCI6602 and CB-68LP from National Instrument.

PCI6602 is a counter/timer device which can receive the pulse signals from the encoders. This board consists of 8 channels and can perform 8 operations simultaneously. It can not only receive pulse inputs but also generate output pulses itself. PCI6602 has 4 pins for each channel accompanying with lots of ground pins and some reserved pins making the total number of the pins to 68.

The functionality of CB-68LP is to make the connection of field I/O signals to the counter/timer device easier. We could easily insert our wires directly into these vertically mounted 68-pin connector on the so called connector block CB-68LP

4.1.3 Encoders

As we have mentioned in chapter 2, we need 6 encoders to measure the parameters of the 6 joints separately. All these encoders are from USdigital and have different resolutions due to their usage. All the encoders are composed of two parts, the disk/strip and the hub. The scales are on the disk/strip. Once the disk/strip moves relative to the hub, pulse output is generated and transmitted to PCI6602 board by the pins on the hub. Each of the encoders will be connected to one of the 8 channels of PCI6602.

4.1.4 Wiring the Encoders

To connect the encoders, we need correct connection of pins. Refer to Figure 4.2 and 4.3, we have the led side view of the pin deployment and the terminal board, the pin wiring tables are shown as in Table 4.1 and 4.2, it gives us how the encoders can be connected to the pins on the terminal board for the 8 channels respectively, we only need 6 of them in our experiment.

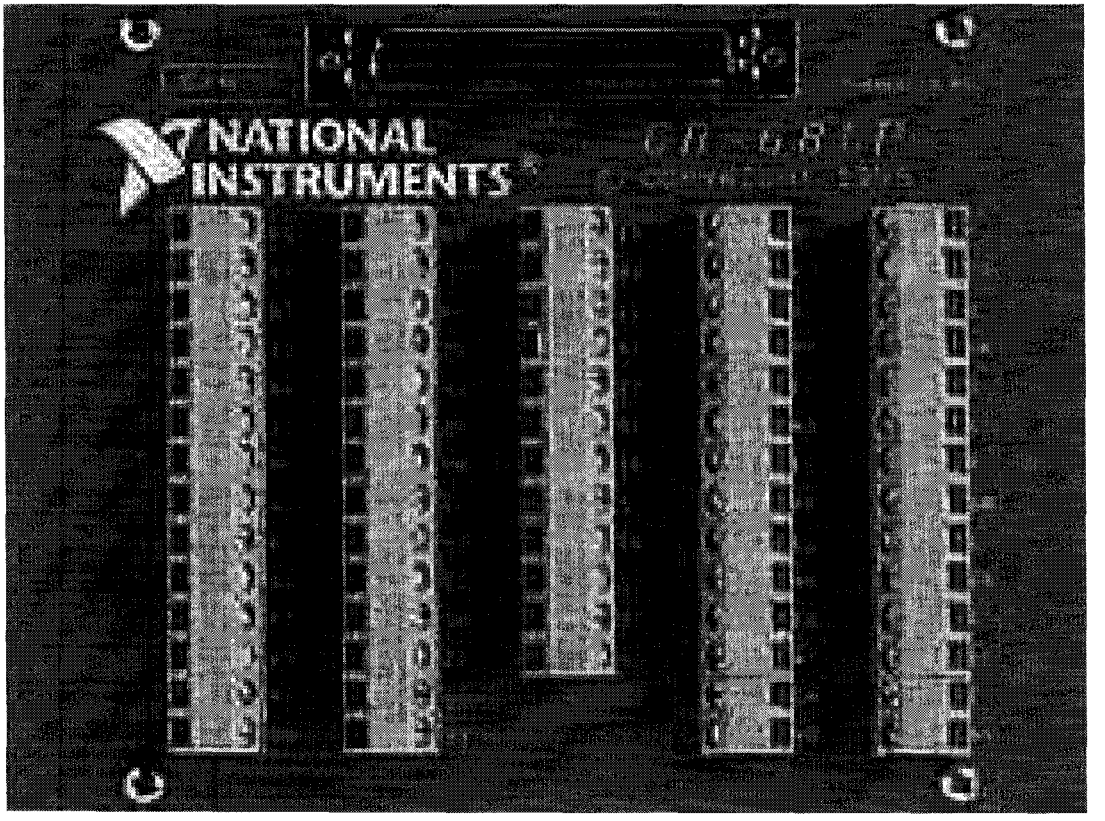

Figure 4.2: Terminal Board

Table 4.1: Wiring Table for Encoder Connection

Pins on encoder	Wire color	channel 0	channel 1	channel 2	channel 3
Gnd	Silver	36	11	68	30
Index	Green	3	8	67	64
Ch A	Black	2	7	34	31
+5V	White	1	1	1	1
Ch B	Red	40	6	66	63

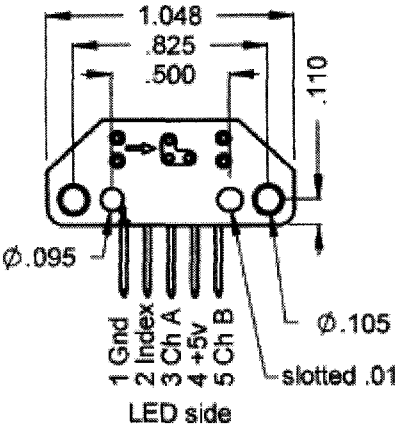

Figure 4.3: Encoder Pin Deployment

Table 4.2: Wiring Table for Encoder Connection Continued

Pins on encoder	Wire color	channel 4	channel 5	channel 6	channel 7
Gnd	Silver	27	24	55	46
Index	Green	61	58	21	10
Ch A	Black	28	25	22	11
+5V	White	1	1	1	1
Ch B	Red	60	57	12	19

4.2 xPC Target Software

The software we are using here is MATLAB. A model is made based on the one we have seen in chapter 3. The new one is suitable for real-time modeling with PCI6602 count inputs. The model is shown in Figure 4.4. The 6 blocks labled PCI6602 are the 6 channels of our PCI6602 board, since the input is pulse, we have to convert the pulse into degrees for angles and meter for distance. The 6 constants are conversion parameters for the encoders. The product blocks are used to perform multiplications between the input pulse signals and the conversion parameters. The tall block in the middle labeled *helivelo* is our main program. We input the encoder signals and corresponding differentials. The code block calculates position, Euler angles and velocities. We also add multiplications to the code block output so as to label the outputs.

4.2.1 Initialization of the Testbed

Some precautions are necessary to make sure the testbed works well with the program.

- Data Source Validation

  After connecting the encoders to the CB-68LP board, we run the xPC explorer in MATLAB, and add host scopes to the model. This allows us to monitor the values of the encoders outputs on the host PC.

  Since the encoder outputs are counts, and are difficult to interpret, we must monitor the products that convert the counts into angles or positions. Then, by observing the value when the testbed is moving, we can make sure the data source (the encoders) working correctly.

- Output Validation

  After we have adjusted the data source (the encoders), we know all the inputs to the program block are correct. Then, we add scopes to detect the outputs separately in position and velocity groups. This phase help us to ensure the correct results based on the encoder signal inputs. For example, when encoder signal indicates a increase in one of the euler angles, the particular output should act as it is supposed to.

- Index Validation

  To agree with our program, we have to make certain index points for each of these 6 encoders. We adjust the encoders to implement the requirements. But we notice that, the encoder will not take the index position as 0 point unless

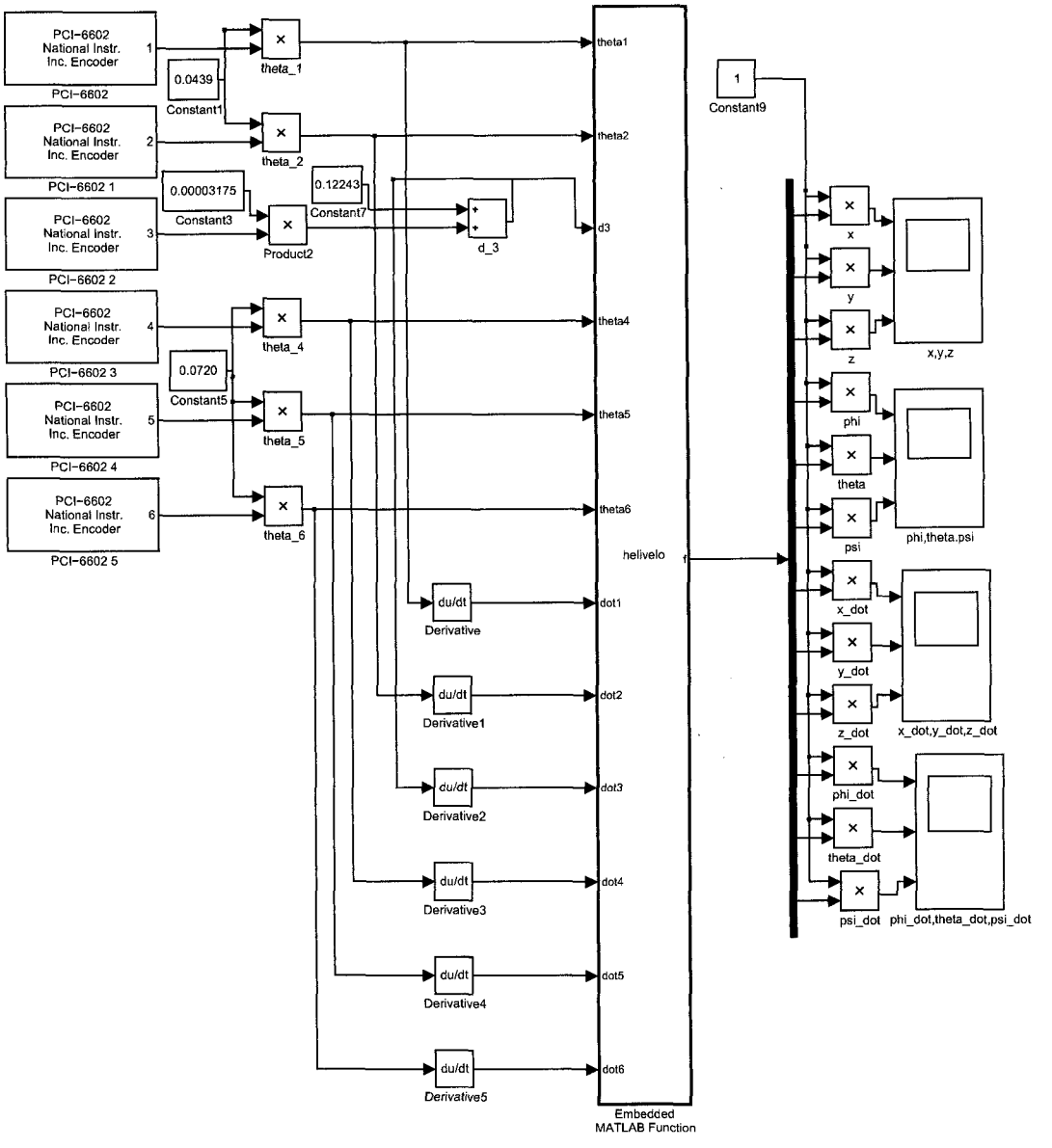

Figure 4.4: Real-time MATLAB Model

the detector sweeps the index point. To solve this problem, we have to make sure all the index points have been swept before we start data collection.

The other problem we have is the index of the linear joint was damaged, since we have to drill some holes on it, the damage is hard to avoid. To solve this problem, we adjust the output in our model by adding a constant which makes the center point of the linear motion range to be index point electrically. To do this, we have to make sure that before the program is started , the slider joint must be at the farthest out position, meaning that the helicopter should be at the farthest position from the base joint of the arm. This position is chosen as zero point for the linear encoder. The correction that shifts this to the center of the encoder is programmed into the model.

- Index Correct Positions

  To make the testbed ready for experiment, we need to set the index to correct position, the intended position for the encoder indexes are shown as in Figure 4.5

4.3 Experiment Results

After we initilized the testbed correctly, we are ready to do the experiments. For the testbed, the accuracy and repeatability are very important which can affect the correctness of our results when being used. Therefore, we did the calibration experiment as followed to test how accurate the testbed is.

4.3.1 Calibration, Validation and Repeatability Experiments

As mentioned in Chapter 2, we designed the first 3 joints of the testbed to allow the helicopter to move along either a vertical or a horizontal line. Here, we take advantage of this feature and do the following calibration experiment. Since the testbed calculates the coordinates of the center of the hand, a calibration bar had to be made that replaced the hand and pointed to the center of the hand.

Vertical Calibration and Validation

In this experiment, we adjusted the testbed to the configuration as shown in Figure 4.6: The distance from the origin of the first joints to the wall is about 2 m where is also approximately the middle of the joint 3's moving range. When the arm is horizontal, the tip of the arm will be at a right contact with the wall. When the tip of the arm moves along the wall, the arm will slide in and out by the motion of joint 3.

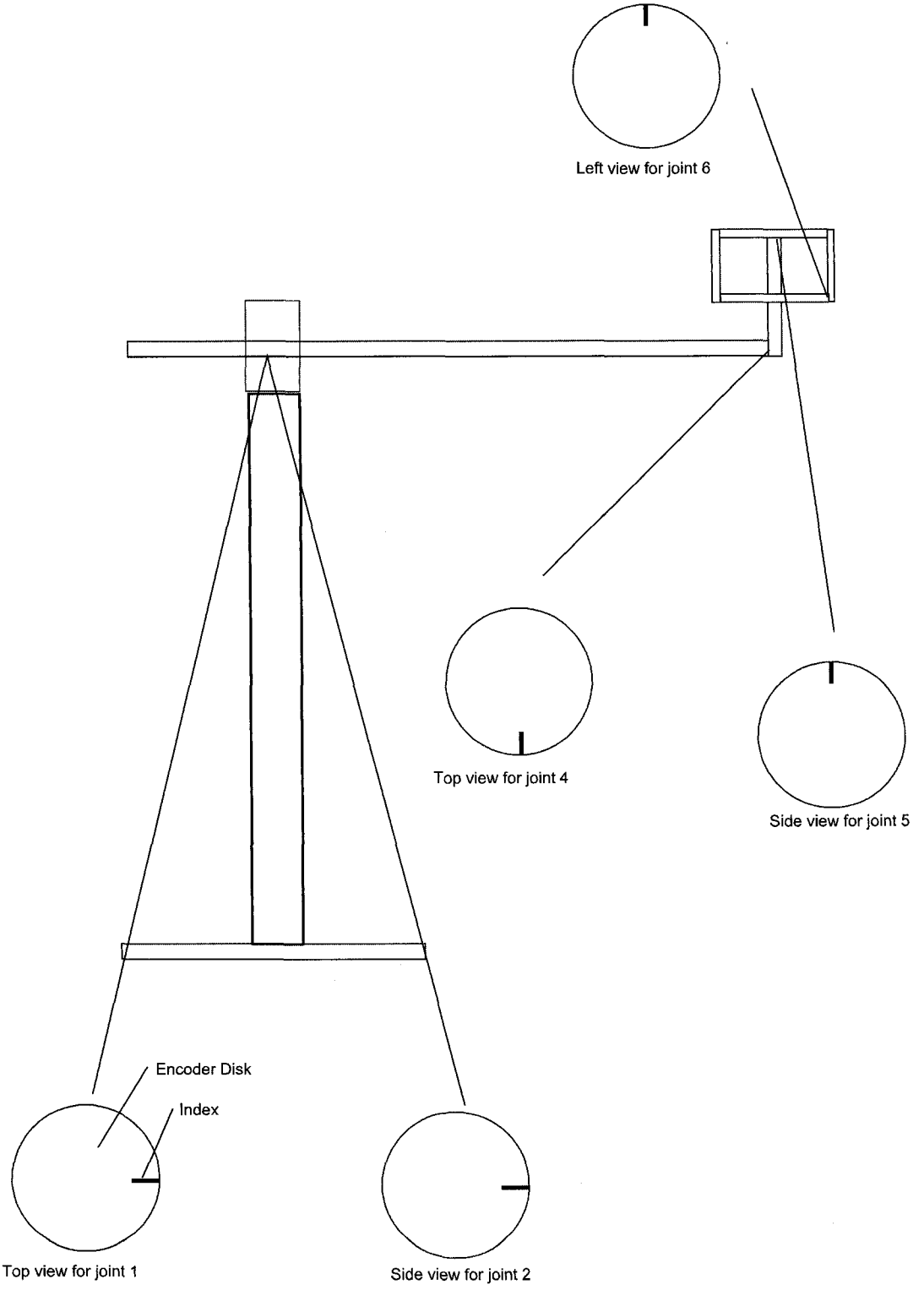

Figure 4.5: Intended Encoder Index Position

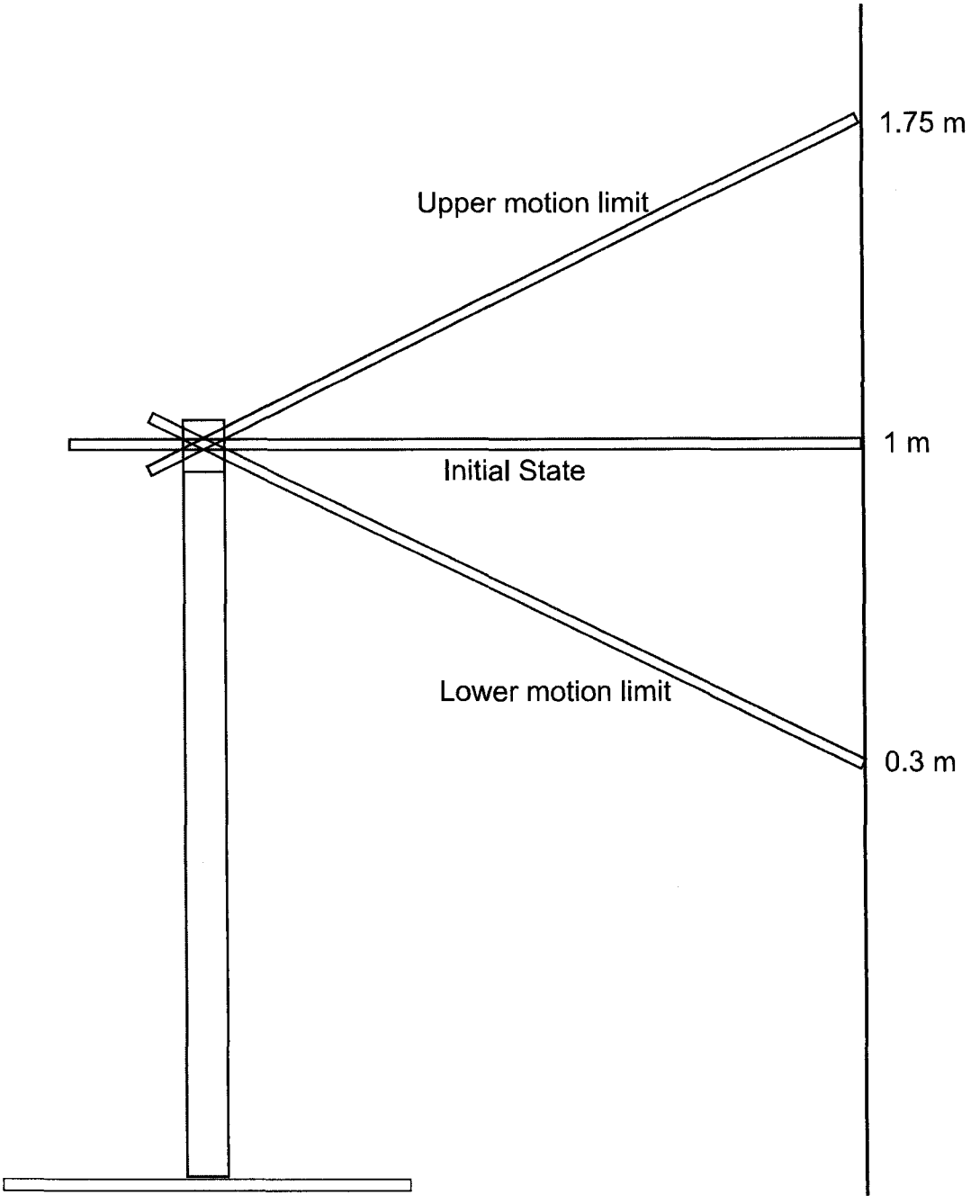

Figure 4.6: Vertical Calibration and Validation Setup and Scheduled Motion

Then, we attached a tape measure vertically to the wall to make the point displays 1 m on the tape measure to be coincident with the bottom point of the arm when the arm is horizontal.

After that, we made sure the slider joint was at the farthest position, we launched the xPC model, swept the index points of the encoders and made it ready to work.

Now, we started the experiment. We let the bottom point of the arm's calibration tube move to specified points, made records of the tape measure values in the Z direction, and also recorded the values from our program. The results of this experiment is shown in Table 4.3.

Horizontal Calibration and Validation

After the vertical calibration experiment, we simply adjusted the testbed to the configuration the same as the former experiment, and attached the tape measure horizontally to the wall.

This time, we moved the left side point of the arm's calibration tube to specified points on the tape, made records of the values in the X direction, and recorded the values from our program. The results of this experiment is shown in Table 4.4.

Repeatability Experiment

To make sure the results are repeatable, we recorded values for a specified point: X: 0 m, Y: 2 m, Z: 0 m several times and calculated the repeatability for it. The results are shown in Table 4.5.

4.3.2 Result Analysis

Due to the procedure of the experiment, some adjustments must be made to the tape measurement readings. They follow.

Error Source Analysis

As we know, for vertical and horizontal calibration, we used the bottom point and left side point of the arm respectively as our standards. These standard points are not the point indicated by our program. As shown in Figure 4.7. P_m is the standard point used by the program, P_t is the standard point used by our manual tape measurement. Based on this figure, we can derive the correction formula for the experiment results. For the vertical calibration, the following adjustment is required for the tape measure results.

$$Z_{tc} = Z_t - r * \cos(\theta) - 1 \tag{4.1}$$

Table 4.3: Results for Vertical Calibration and Validation

Z_t (m)	Y_m (m)	Z_m (m)	θ (degree)	Z_{tc} (m)	Z_{error} (m)
1.75	-1.986	0.767	-21.1	0.741	-0.026
1.70	-1.988	0.719	-19.9	0.691	-0.023
1.65	-1.990	0.671	-18.6	0.641	-0.030
1.60	-1.991	0.616	-17.2	0.591	-0.025
1.55	-1.992	0.554	-15.5	0.541	-0.013
1.50	-1.993	0.517	-14.5	0.491	-0.026
1.45	-1.995	0.464	-13.1	0.441	-0.023
1.40	-1.996	0.416	-11.8	0.391	-0.025
1.35	-1.997	0.364	-10.3	0.341	-0.023
1.30	-1.998	0.314	-8.9	0.291	-0.023
1.25	-1.999	0.263	-7.5	0.241	-0.023
1.20	-2.000	0.213	-6.1	0.191	-0.023
1.15	-2.001	0.163	-4.7	0.141	-0.023
1.10	-2.002	0.113	-3.2	0.091	-0.023
1.05	-2.002	0.063	-1.8	0.041	-0.023
1.00	-2.003	0.014	-0.4	-0.009	-0.024
0.95	-2.003	-0.039	1.1	-0.059	-0.021
0.90	-2.003	-0.086	2.5	-0.109	-0.024
0.85	-2.004	-0.136	3.9	-0.159	-0.024
0.80	-2.004	-0.185	5.3	-0.209	-0.025
0.75	-2.005	-0.230	6.5	-0.259	-0.030
0.70	-2.006	-0.282	8.0	-0.309	-0.028
0.65	-2.001	-0.332	9.4	-0.359	-0.027
0.60	-2.009	-0.380	10.7	-0.409	-0.029
0.55	-2.011	-0.430	12.1	-0.459	-0.029
0.50	-2.012	-0.479	13.4	-0.509	-0.030
0.45	-2.014	-0.527	14.7	-0.559	-0.032
0.40	-2.016	-0.575	15.9	-0.609	-0.034
0.35	-2.018	-0.625	17.2	-0.659	-0.035
0.30	-2.016	-0.676	18.5	-0.709	-0.033

Table 4.4: Results for Horizontal Calibration and Validation

X_t (m)	Y_m (m)	X_m (m)	θ (degree)	X_{tc} (m)	X_{error} (m)
1.70	-2.003	0.695	-19.1	0.700	0.004
1.65	-2.002	0.645	-17.9	0.645	0.001
1.60	-2.000	0.596	-16.6	0.584	-0.011
1.55	-2.000	0.545	-15.2	0.532	-0.013
1.50	-2.001	0.496	-13.9	0.493	-0.003
1.45	-2.000	0.445	-12.6	0.450	-0.005
1.40	-2.001	0.396	-11.2	0.392	-0.005
1.35	-2.000	0.347	-9.8	0.332	-0.015
1.30	-2.001	0.296	-8.4	0.285	-0.011
1.25	-2.000	0.248	-7.1	0.247	-0.001
1.20	-2.001	0.198	-5.7	0.198	-0.000
1.15	-2.001	0.149	-4.3	0.136	-0.013
1.10	-2.002	0.097	-2.8	0.082	-0.015
1.05	-2.002	0.051	-1.5	0.042	-0.010
1.00	-2.003	-0.002	0.0	0.000	-0.002
0.95	-2.003	-0.049	1.4	-0.058	-0.009
0.90	-2.004	-0.100	2.9	-0.119	-0.019
0.85	-2.005	-0.148	4.2	-0.164	-0.016
0.80	-2.006	-0.199	5.7	-0.202	-0.003
0.75	-2.007	-0.249	7.1	-0.253	-0.004
0.70	-2.007	-0.297	8.4	-0.315	-0.017
0.65	-2.008	-0.346	9.8	-0.368	-0.022
0.60	-2.008	-0.395	11.1	-0.408	-0.013
0.55	-2.009	-0.447	12.6	-0.450	-0.003
0.50	-2.010	-0.497	13.9	-0.507	-0.010
0.45	-2.010	-0.546	15.2	-0.568	-0.022
0.40	-2.010	-0.596	16.5	-0.616	-0.020
0.35	-2.010	-0.646	17.8	-0.655	-0.008
0.30	-2.011	-0.691	19.0	-0.700	-0.009

Table 4.5: Results for Repeatability Experiment

	1	2	3	4	5
X	-0.003	-0.0015	-0.0015	-0.0015	-0.0015
Y	-2.003	-2.003	-2.003	-2.003	-2.003
Z	0.0135	0.0135	0.0135	0.0135	0.0120

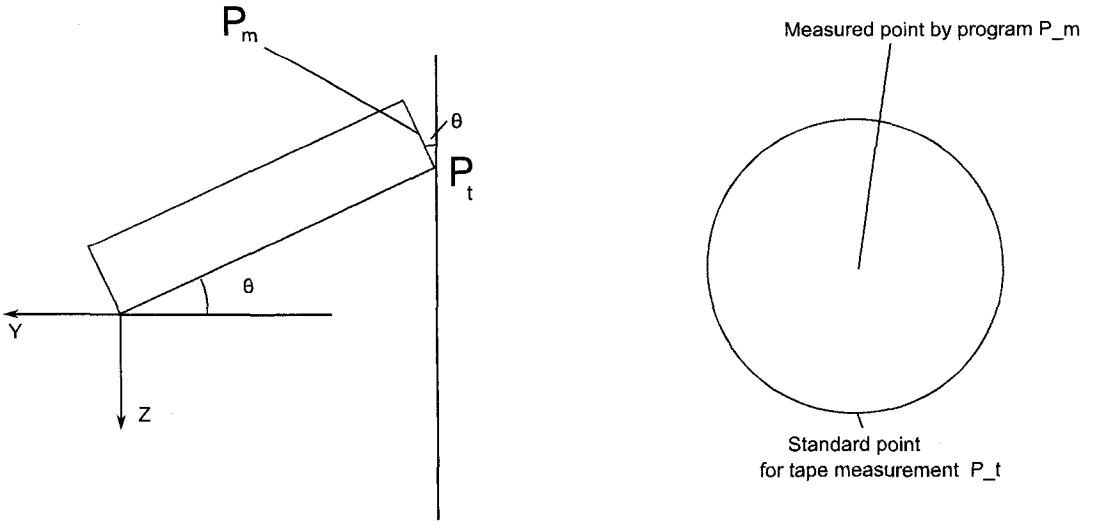

Figure 4.7: Error Source Diagram

where, Z_{tc} is the corrected Z tape value. Z_t is the value before correction. r is the radius of the arm's calibration bar which is 0.0095 m.

Following the same way, we can find the correction formula for the horizontal calibration as follows:

$$X_{tc} = X_t + r * \cos(\theta) - 1 \tag{4.2}$$

Calculation and Conclusion

According to Table 4.3, 4.4 and 4.5. We can generates the plots of the relationship between the tape measured results and the program measured results as shown in Figure 4.8 and 4.9. We can also calculate the accuracies and repeatability as followed:

For Z, accuracy is:

$$A_Z = \pm\frac{|e_{max} - e_{min}|}{2} = \pm 0.011 \text{ m} \tag{4.3}$$

For X, accuracy is:

$$A_X = \pm\frac{|e_{max} - e_{min}|}{2} = \pm 0.013 \text{ m} \tag{4.4}$$

By the formulas:

$$\bar{x} = \frac{\sum x}{n} \tag{4.5}$$

$$x_d = \pm 2\frac{\sum (x_i - \bar{x})^2}{n-1} \tag{4.6}$$

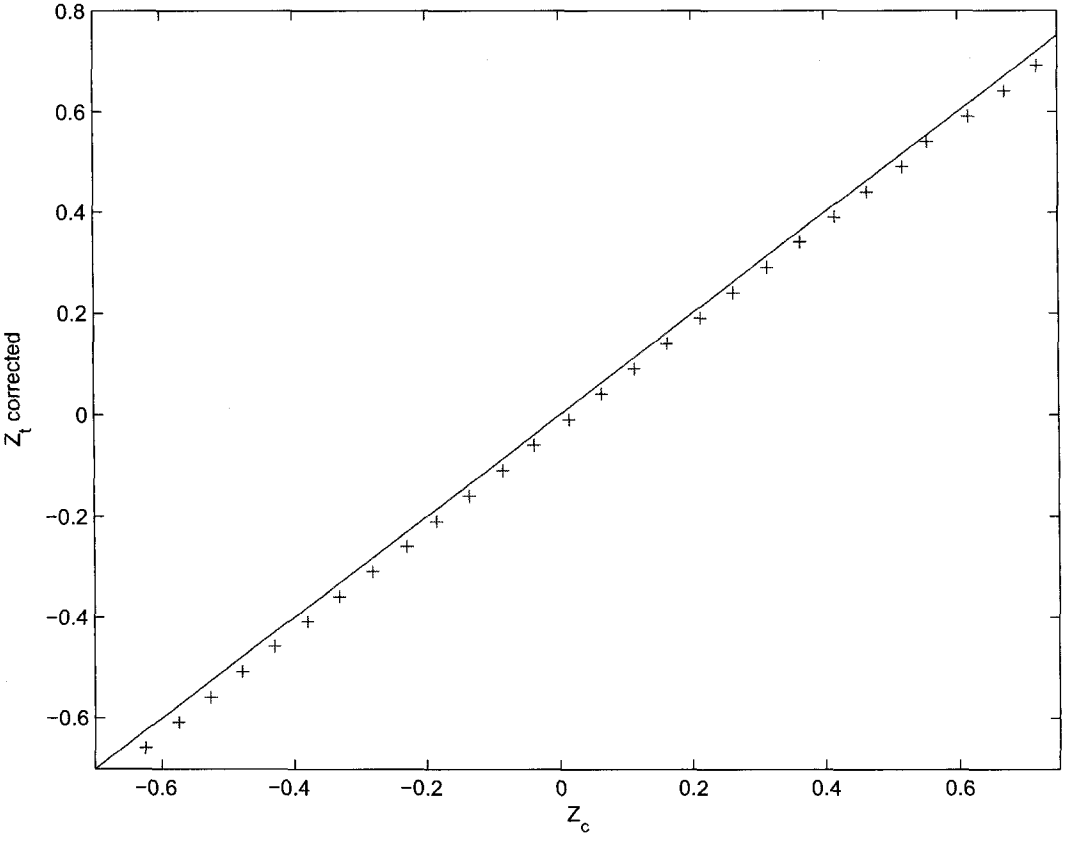

Figure 4.8: Relation between Tape Measured Z and Program Measured Z

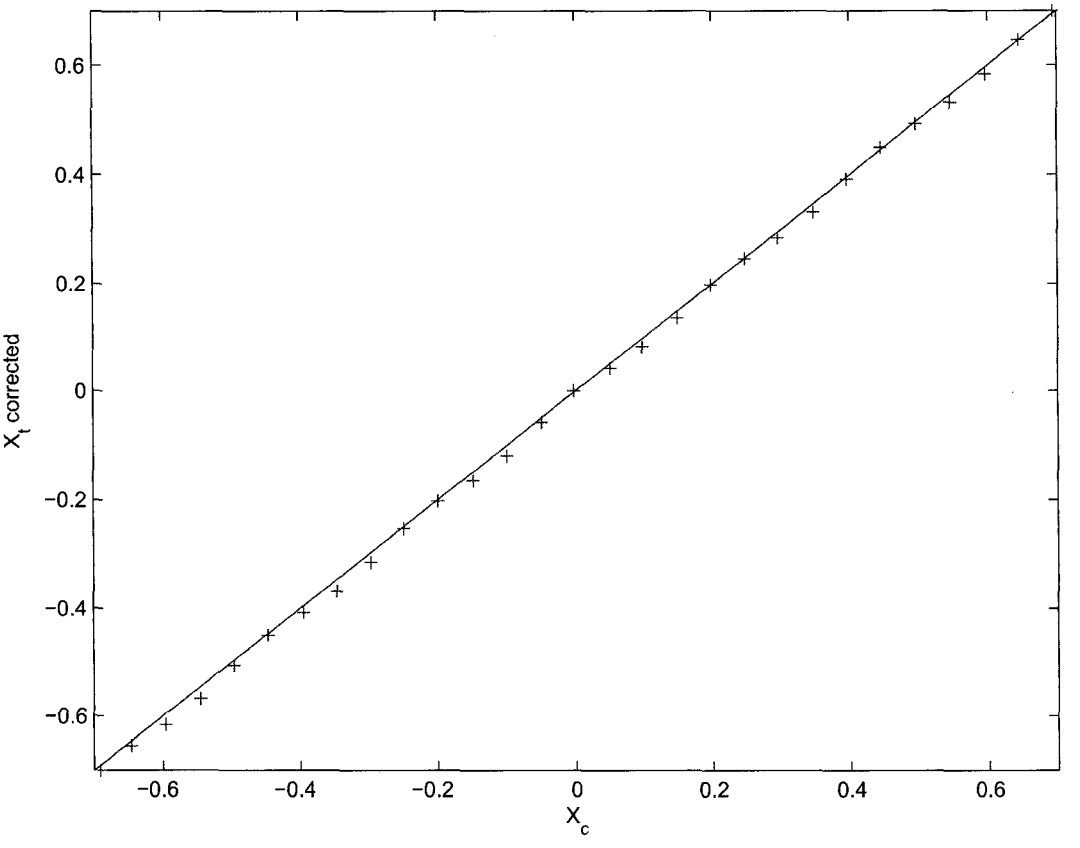

Figure 4.9: Relation between Tape Measured X and Program Measured X

We have:

$$\begin{aligned} X_d &= \pm 5 \times 10^{-7} \text{ m} \\ Y_d &= \pm 0 \text{ m} \\ Z_d &= \pm 5 \times 10^{-7} \text{ m} \end{aligned} \tag{4.7}$$

Based on the Figures 4.8 and 4.9 , we found that, for the horizontal calibration, we obtained good accuracy and the plot is good, the points are around the $Y = X$ straight line. But for the vertical calibration, accuracy is good, but all the points are lower than the expected place. It's supposed to because of the bending of the arm. The arm we used to do the experiment is 0.75 in diameter. For the final testbed, the arm we used is 1.375 in diameter, that stronger arm will reduce the affection by the bending. At last, we can conclude the repeatability of the testbed is good based on the calculations.

4.3.3 Wrist Angle Calibration and Validation Experiment

In the above experiments, we calibrated the first 3 joints. Now, we are going to calibrate two of the wrist joints. We clamped the wrist to a leveled table to make it stationary. We set the zero point of the inclinometer while it is on a level table. Then, we mount a inclinometer to a plate belongs to joint 5. When we move the plate, the inclinometer will give us an angle indicates the amount the rotating motion. We also can read the result from our program running in real-time on a target PC. First, we added an angle offset to the program to make sure both the program and the inclinometer show zero angles when the link connected to the joint is horizontal. We rotate the joint 5 with 2-degree intervals according to the inclinometer and we record the results from the real-time xPC-target program. After we calibrate joint 5, we repeated the same procedure for joint 6. The calibration results are as shown in Table 4.6.

We can plot the results out and show them as in Figure 4.10 and 4.11. The accuracy for these angles can also be calculated as in Equation 4.8 and 4.9.

From Figure 4.10 and 4.11, we can see the results are not perfectly match the expected values. That's because when the plate moves relative to each other, the shaft through them suffers some warping. This caused the center of the encoder disk to depart from the point it supposed to be. However, The accuracy of the encoders is satisfying.

$$\theta_5 = \pm \frac{|e_{max} - e_{min}|}{2} = \pm 1.1^{\circ} \tag{4.8}$$

Table 4.6: Results for Wrist Angle Calibration and Validation

θ_5 deg.(inclinometer)	θ_5 deg.(encoder)		θ_6 deg.(inclinometer)	θ_6 deg.(encoder)
-18.3	-19.3		-20.0	-20.6
-16.0	-17.0		-18.0	-18.7
-14.0	-14.8		-16.0	-16.7
-12.0	-12.2		-14.0	-14.8
-10.0	-10.5		-12.0	-12.7
-8.0	-8.2		-10.0	-10.5
-6.0	-6.1		-8.0	-8.4
-4.0	-4.0		-6.0	-6.3
-2.0	-1.9		7-4.0	-4.3
0.0	0.0		-2.0	-2.1
2.0	2.3		0.0	-0.1
4.0	4.4		2.0	2.1
6.0	6.6		4.0	4.1
8.0	8.8		6.0	6.3
10.0	10.8		8.0	8.3
12.0	12.9		10.0	10.2
14.0	15.2		12.0	12.2
16.0	17.0		14.0	14.3
18.0	19.0		16.0	16.3
19.5	20.1		18.0	18.4
			20.0	20.1

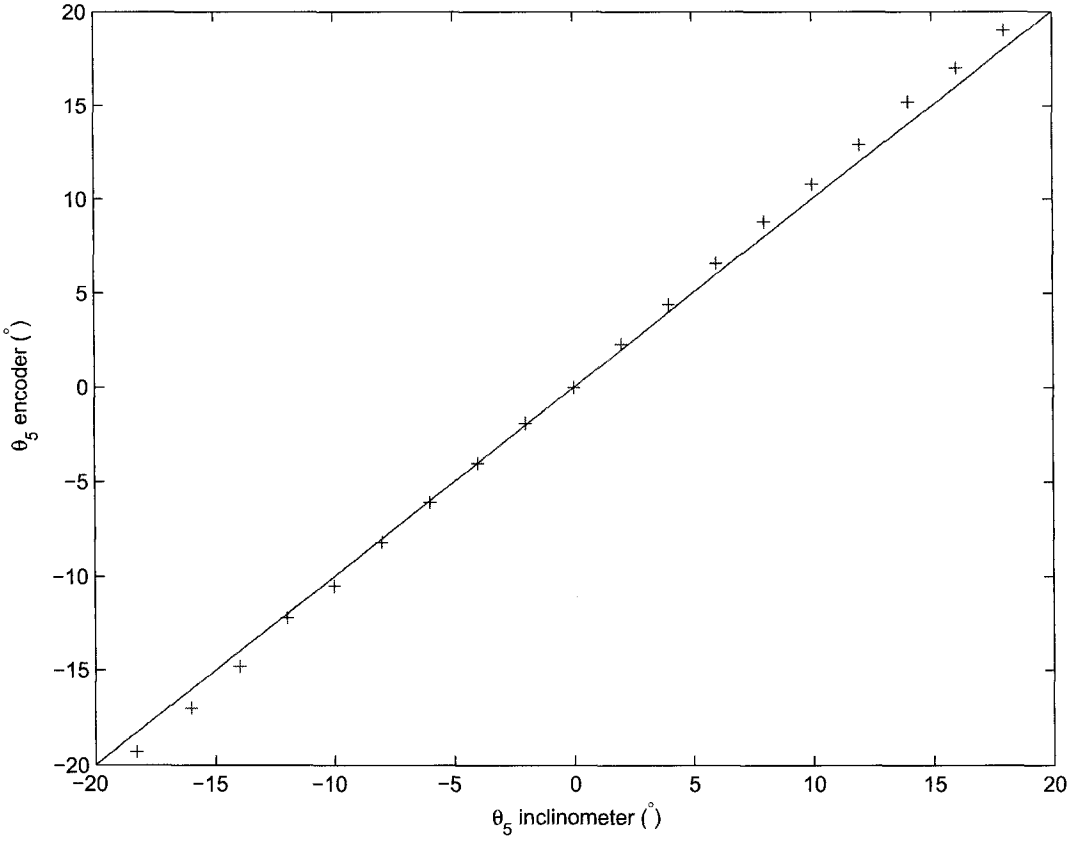

Figure 4.10: Relation between Inclinometer Measured Angle and Encoder Measured Angle for Joint 5

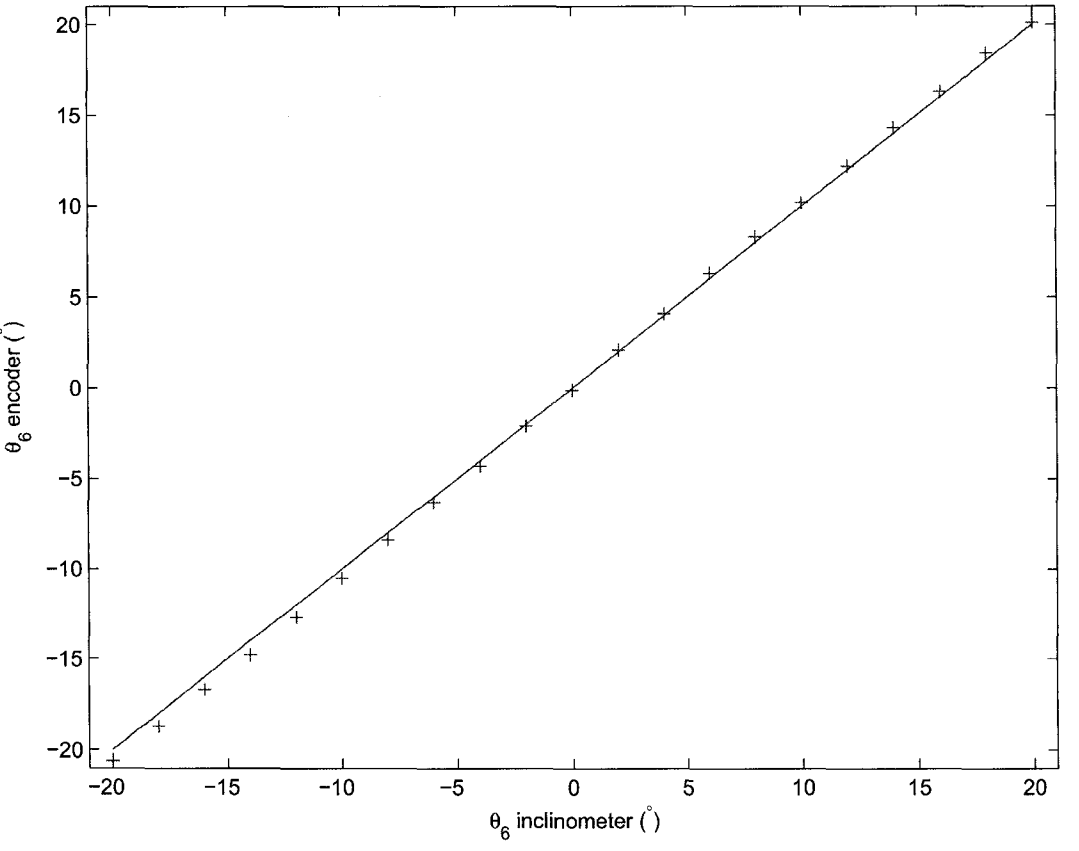

Figure 4.11: Relation between Inclinometer Measured Angle and Encoder Measured Angle for Joint 6

$$\theta_6 = \pm\frac{|e_{max} - e_{min}|}{2} = \pm 0.6\,^{\circ} \tag{4.9}$$

Chapter 5

Conclusions

In this project, the objective was to develop a control system for the helicopter and test the control system. To confirm that our control algorithm can interpret the physical data correctly, we built a testbed which can perform experiments safely and give us raw data to check with the output of our algorithm. By this testbed, we are able to acquire the position and orientation data of the helicopter without any unexpected crashes or incidents.

At first, we designed the testbed in Pro/Engineer based on the required functions. Then, we revised the designs due to the strength validation results. Furthermore, some better designs that could do a better job were raised and replaced the original ones when we were working on the design.

Second, we derived the formulas used to calculate the position and orientation of the helicopter based on the knowledge in robotics. We made a program in MATLAB to aid us as well.

Finally, we built a encoder data system. We mounted 6 encoders to the testbed which can give us required outputs. Then, by an xPC target and host PC system, we were able to transmit the collected data to target PC and thus received by the control computer. The position and orientation were calculated by the control computer once inputs were obtained. Some experiments were carried out to check the accuracy and repeatability of the system. Satisfying results were obtained and the project objectives were met successfully.

Bibliography

[1] F. Zhang, Y. Tao, X. Chen, and D. Lu, "The system framework of earth observation information captures based on high altitude uav," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 3, pp. 2074 – 2077, 2005.

[2] B. Coifman, M. McCord, R. Mishalani, M. Iswalt, and Y. Ji, "Roadway traffic monitoring from an unmanned aerial vehicle," *IEE Proceedings: Intelligent Transport Systems*, vol. 153, no. 1, pp. 11 – 20, 2006.

[3] C. Edwards and C. Robinson, "Remote deployment and monitoring of ground sensors from uavs," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 5090, pp. 301 – 310, 2003.

[4] D. Cerchie, G. Dockter, M. Hardesty, and S. Kasprzyk, "Rapid development of a rotorcraft uav system," *AHS International Specialists' Meeting - Unmanned Rotorcraft: Design, Control and Testing, Proceedings*, pp. 105 – 112, 2005.

[5] A. D. J. C. Avila Vilchis, B. Brogliato and R. Lozano, "Nonlinear modelling and control of helicopters," *Automatica*, vol. 39, pp. 1583–1596, 2003.

[6] J. F. Montgomery, A. E. Johnson, S. I. Roumeliotis, and L. H. Matthies, "The jet propulsion laboratory autonomous helicopter testbed: A platform for planetary exploration technology research and development," *Journal of Field Robotics*, vol. 23, no. 3-4, pp. 245 – 267, 2006.

[7] K. Wong and C. Bill, "Uavs over australia - market and capabilities," *RPVs Thirteenth International Conference. RPVs/UAVs. Remotely Piloted Vehicles*, pp. 4 – 1, 1998//.

[8] D. A. McCarville, "Unmanned aerial systems: Industry overview and material usage trends," *SAMPE Journal*, vol. 42, no. 3, pp. 7 – 15, 2006.

[9] F.-B. Hsiao, M.-T. Lee, Y.-H. Chien, W.-Y. Chang, T.-L. Liu, and U.-J. Payne, "Development of a low cost autonomous surveillance unmanned aerial vehicle

system," *Zhongguo Hangkong Taikong Xuehui Huikan/Transactions of the Aeronautical and Astronautical Society of the Republic of China*, vol. 35, no. 4, pp. 307 – 316, 2003.

[10] X. Zhang, A. Myklebust, and P. Gelhausen, "A geometric modeler for the conceptual design of ducted fan uavs," *Collection of Technical Papers - AIAA 3rd "Unmanned-Unlimited" Technical Conference, Workshop, and Exhibit*, vol. 2, pp. 705 – 714, 2004.

[11] T. D. Rawlins, "High-incidence stabilator as an out-of-control recovery device for a fixed-wing subscale transport unmanned air vehicle," *43rd AIAA Aerospace Sciences Meeting and Exhibit - Meeting Papers*, pp. 3821 – 3851, 2005.

[12] S. Winkler, H.-W. Schulz, M. Buschmann, and P. Vorsmann, "Testing gps/ins integration for autonomous mini and micro aerial vehicles," *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation, ION GNSS 2005*, vol. 2005, pp. 999 – 1006, 2005.

[13] M. Darbandi, A. Nazari, and G. E. Schneider, "Drag reduction of light uav wing with deflectable surface in low reynolds number flows," *Collection of Technical Papers - 3rd AIAA Flow Control Conference*, vol. 3, pp. 1551 – 1558, 2006.

[14] S. Rasmussen and P. Chandler, "Multiuav: a multiple uav simulation for investigation of cooperative control," *Proceedings of the 2002 Winter Simulation Conference (Cat. No.02CH37393)*, vol. vol.1, pp. 869 – 77, 2002//.

[15] J. Ramage, "Emerging military unmanned air vehicle system concepts," *Air Traffic Management: Support for Decision Making Optimisation - Automation (AGARD-R-825)*, pp. 7 – 1, 1997//.

[16] P. Garcia-Pardo, G. Sukhatme, and J. Montgomery, "Towards vision-based safe landing for an autonomous helicopter," *Robotics and Autonomous Systems*, vol. 38, no. 1, pp. 19 – 29, 2002/01/31.

[17] D. E. Cardoze and R. C. Arkin, "Development of visual tracking algorithms for an autonomous helicopter," *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 2591, pp. 145 – 156, 1995.

[18] S. Saripalli and G. Sukhatme, "A testbed for mars precision landing experiments by emulating spacecraft dynamics on a model helicopter," *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (Cat. No.02CH37332C)*, vol. vol.3, pp. 2097 – 102, 2002//.

[19] J. S. Bay, E. N. Johnson, and P. Ray, "University testbeds demonstrate major advances in autonomous rotorcraft control software," *Vertiflite*, vol. 51, no. 3, pp. 22 – 27, 2005.

[20] J. S. Dittrich and E. N. Johnson, "Multi-sensor navigation system for an autonomous helicopter," *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, vol. 2, pp. 8 – 11, 2002.

[21] E. N. Johnson, D. P. Schrage, J. Prasad, and G. J. Vachtsevanos, "Uav flight test programs at georgia tech," *Collection of Technical Papers - AIAA 3rd "Unmanned-Unlimited" Technical Conference, Workshop, and Exhibit*, vol. 1, pp. 527 – 539, 2004.

[22] E. N. Johnson, N. Rooz, J. Hur, and W. Pickell, "A concurrent testing process for research unmanned aerial vehicles," vol. 2, (San Francisco, CA, United States), pp. 723 – 733, 2006.

[23] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. J. Tomlin, "The stanford testbed of autonomous rotorcraft for multi agent control (starmac)," *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, vol. 2, pp. 12 – 4, 2004.

[24] J. S. Dittrich, F. Thielecke, and H.-P. Schwaneck, "Unmanned rotorcraft demonstrator artis: Challenges in autonomous control and teaming," *Annual Forum Proceedings - American Helicopter Society*, vol. 2, pp. 1404 – 1411, 2004.

[25] J. Camano, J. Baltar, and J. Raimundez, "Laboratory testbed for nonlinear control of an autonomous model helicopter," *Proceedings of the World Automation Congress (IEEE Cat. No.04EX832C)*, vol. vol.15, pp. 547 – 52, 2004//.

[26] J. Morris, M. van Nieuwstadt, and P. Bendotti, "Identification and control of a model helicopter in hover," *American Control Conference*, vol. 2, pp. 1238–1242, June 29-July 1 1994.

[27] "Model helicopter specifications." http://www.joker-usa.com/maxijoker2.htm, 2000.

[28] "Encoder features." www.usdigital.com, 2007.

[29] "Product from mcmaster-carr." http://www.mcmaster.com/, 2007.

[30] R. L. Norton, *Machine Design*. Person Education Ltd., 2006.

[31] J. J. Craig, *Introduction to Robotics Mechanics and Controls.* Addison-Wesley Publishing Company, Inc, 1986.

[32] S. Kim and D. Tilbury, "Mathematical modeling and experimental identification of an unmanned helicopter robot with flybar dynamics," *Journal of Robotic Systems*, vol. 21, no. 3, pp. 95 – 116, 2004. Unmanned helicopter robot;Flybar;.

Appendix A

Testbed Drawing

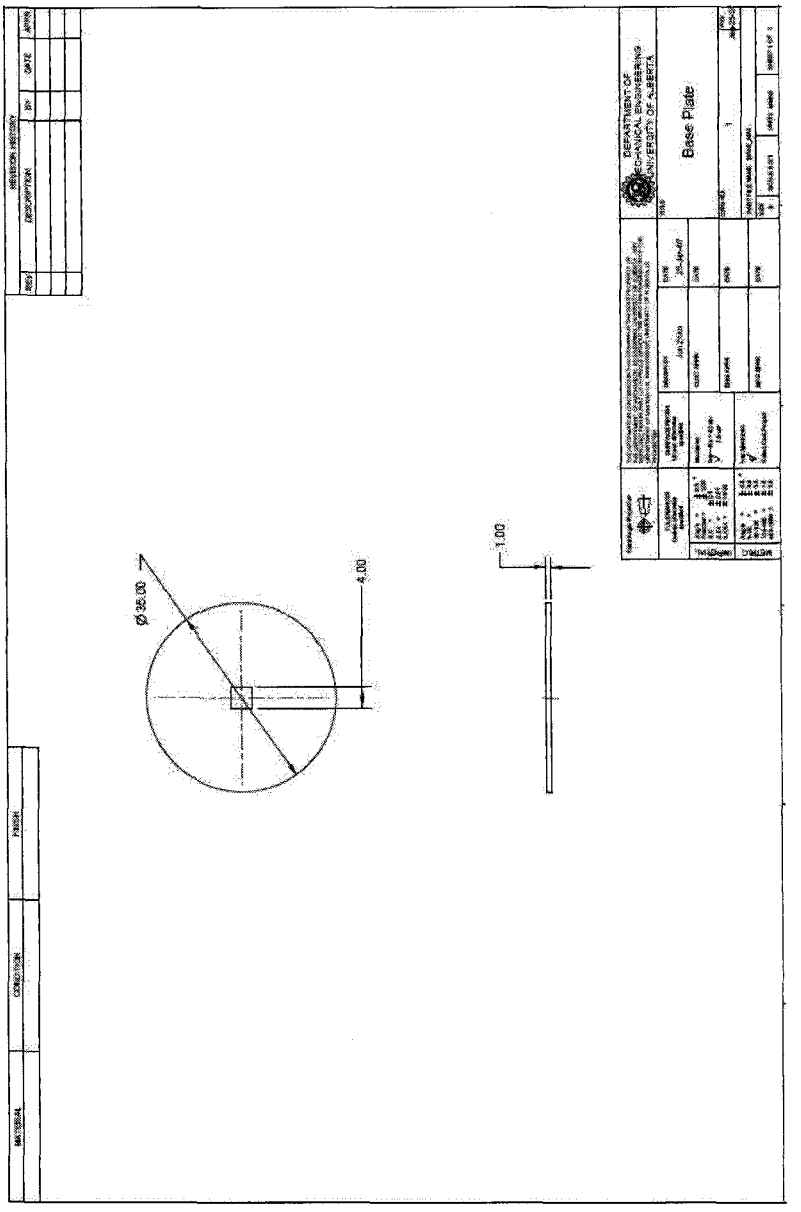

Figure A.1: Base Plate

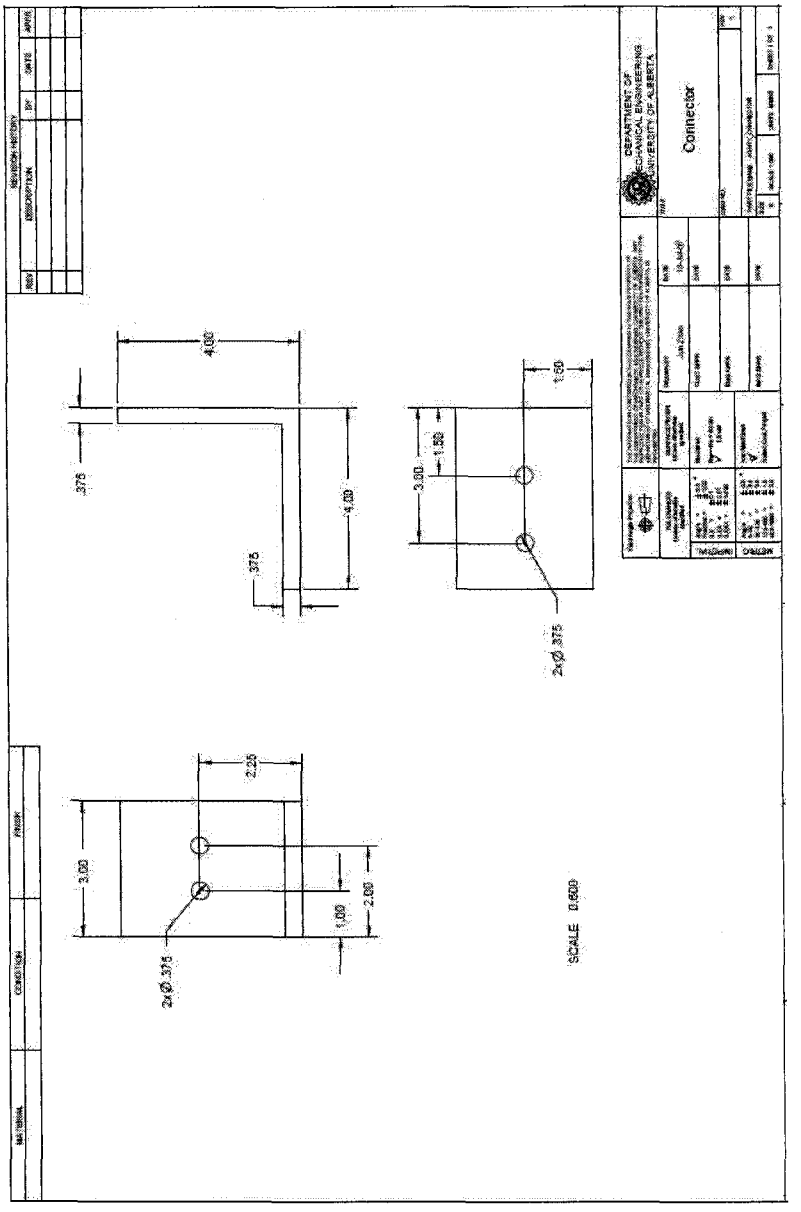

Figure A.2: Connector between Plate and Bar

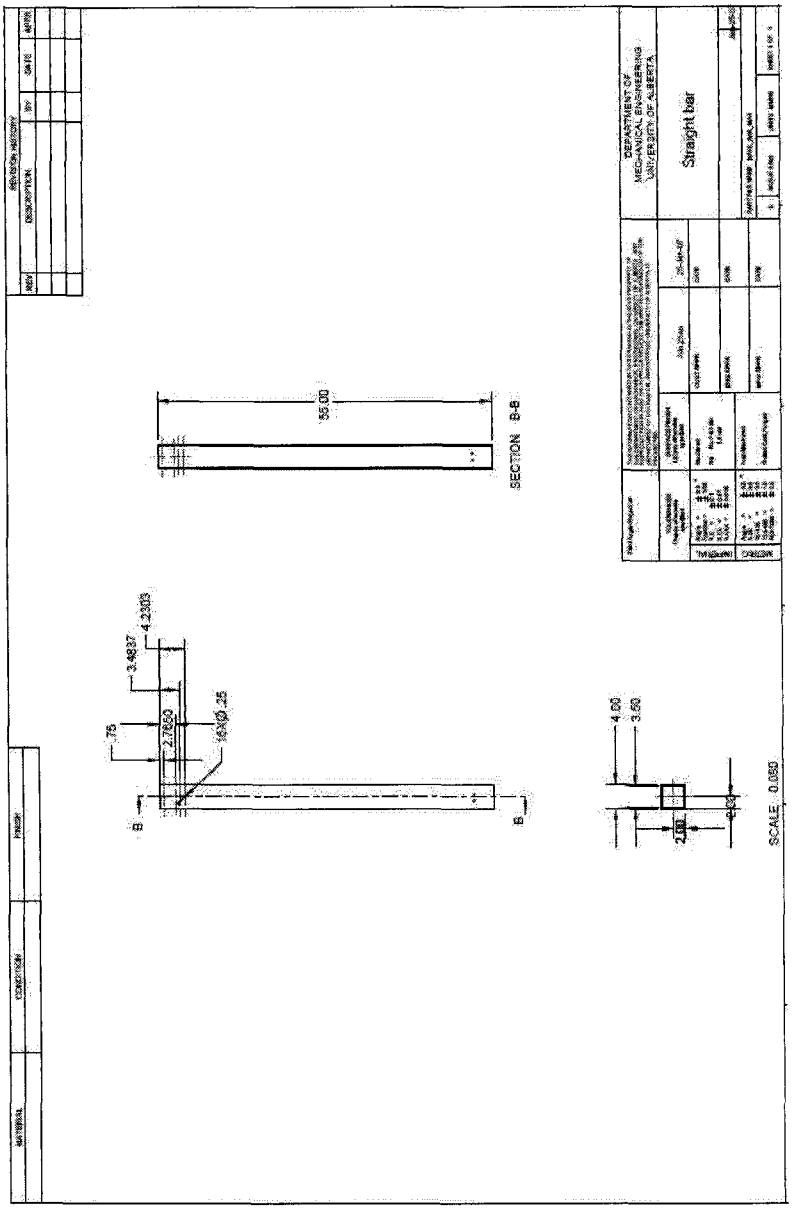

Figure A.3: Straight Bar

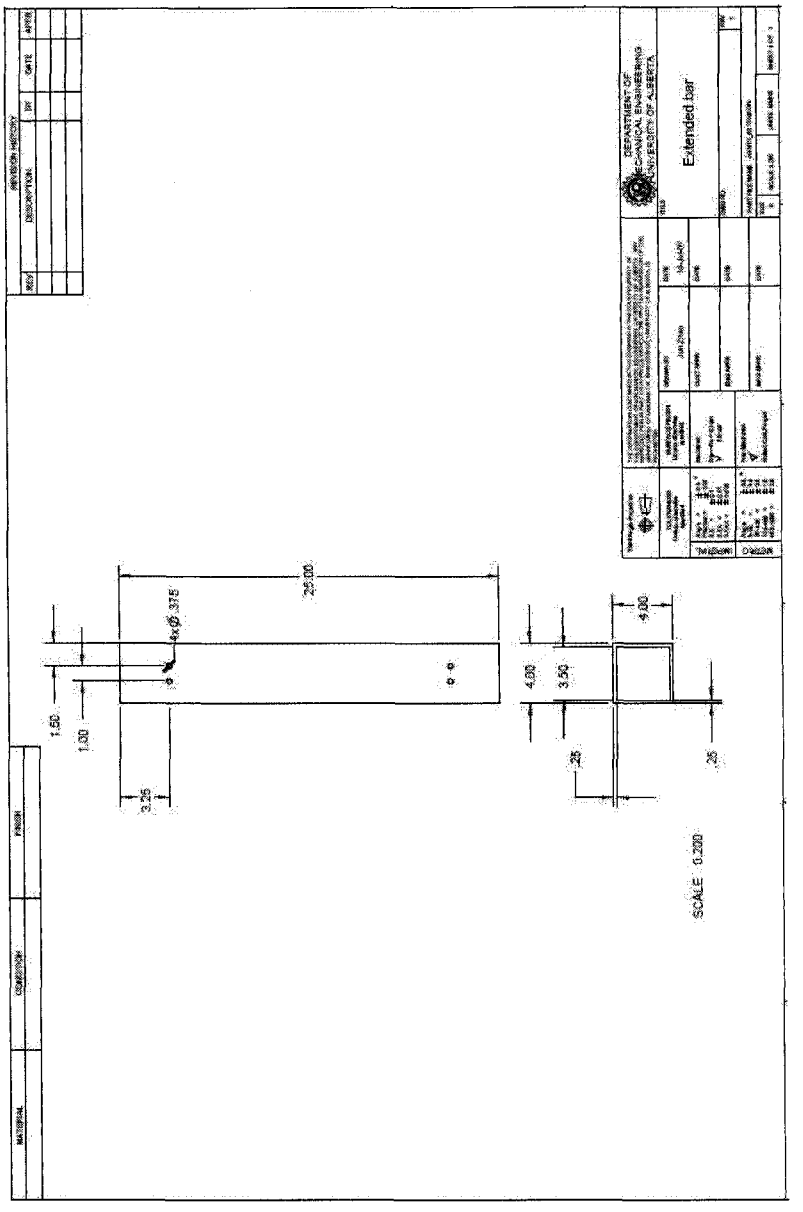

Figure A.4: Extended Bar

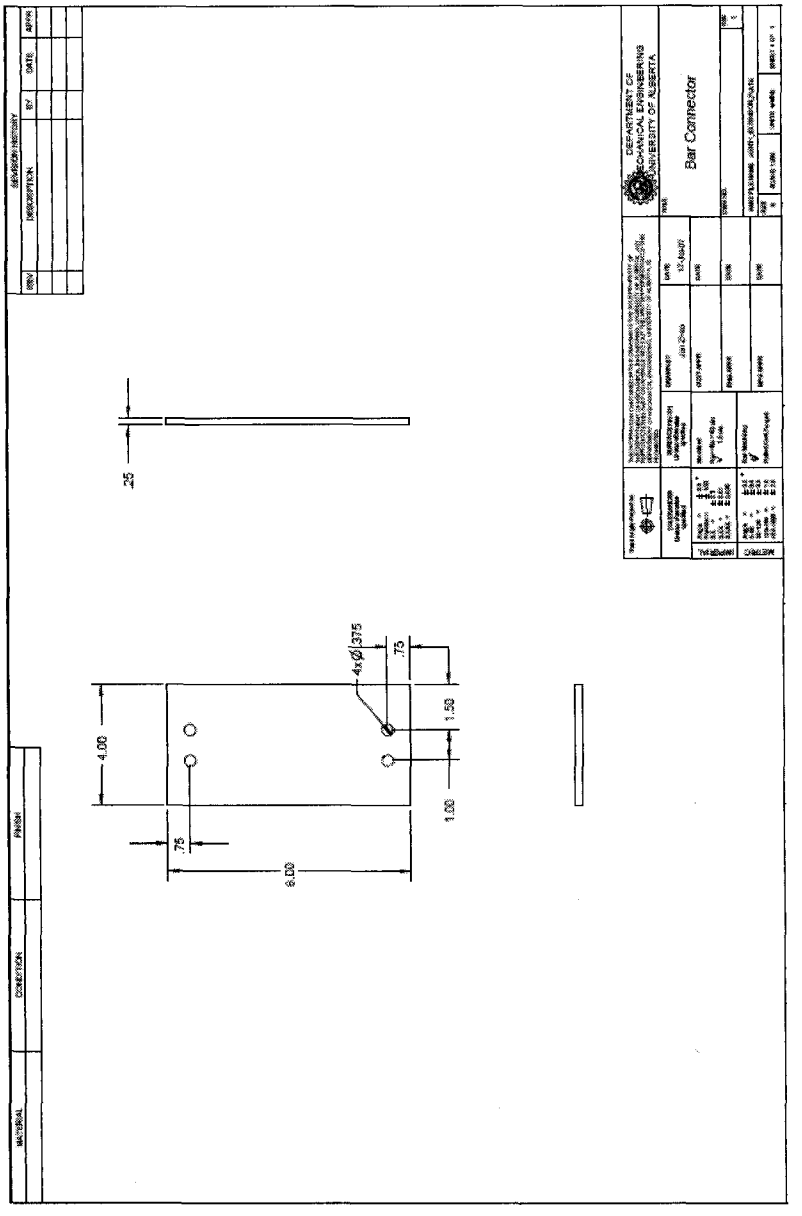

Figure A.5: Connector between straight bar and extended bar

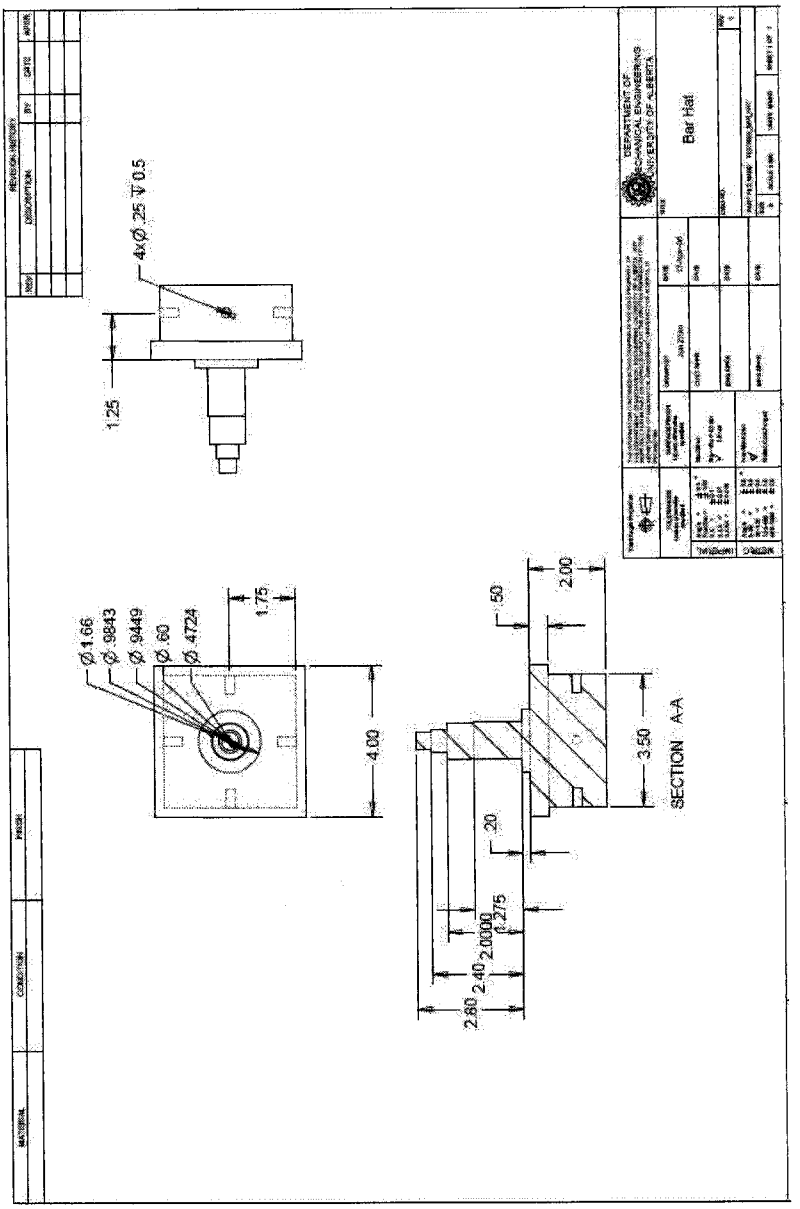

Figure A.6: Hat

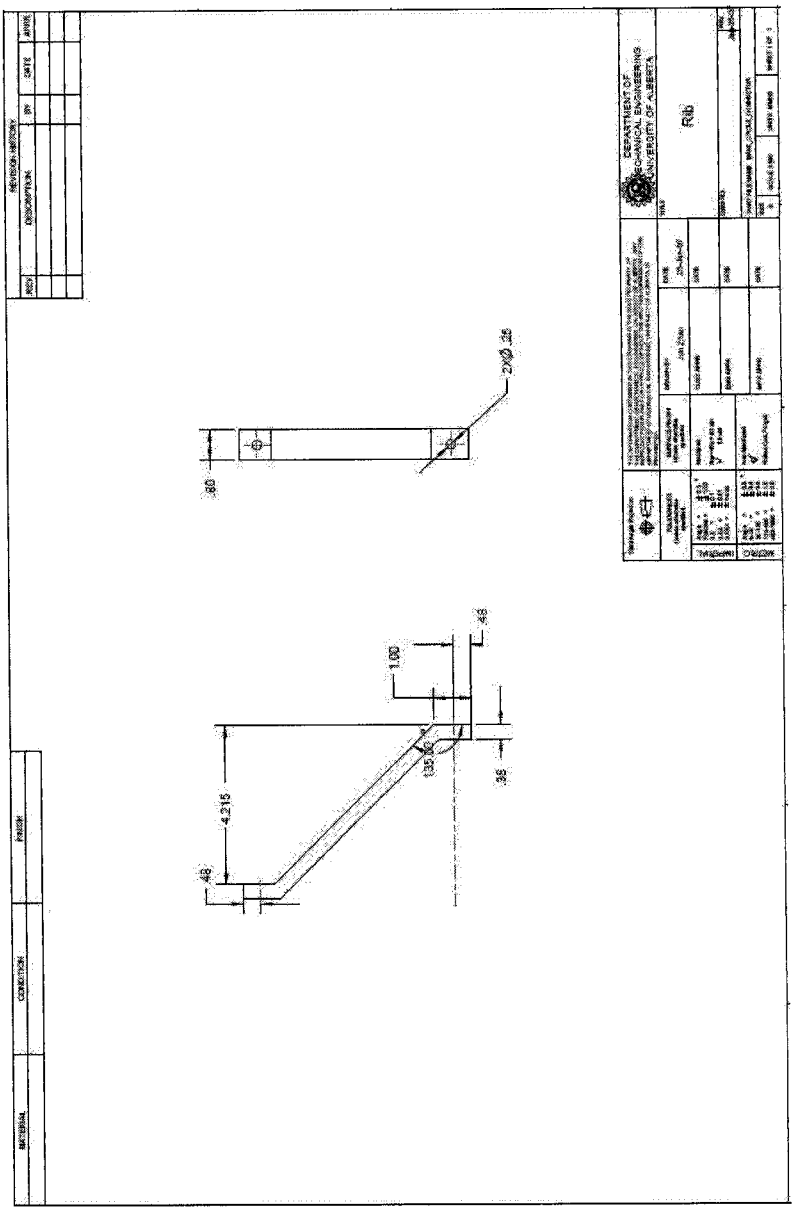

Figure A.7: Ring Support Rib

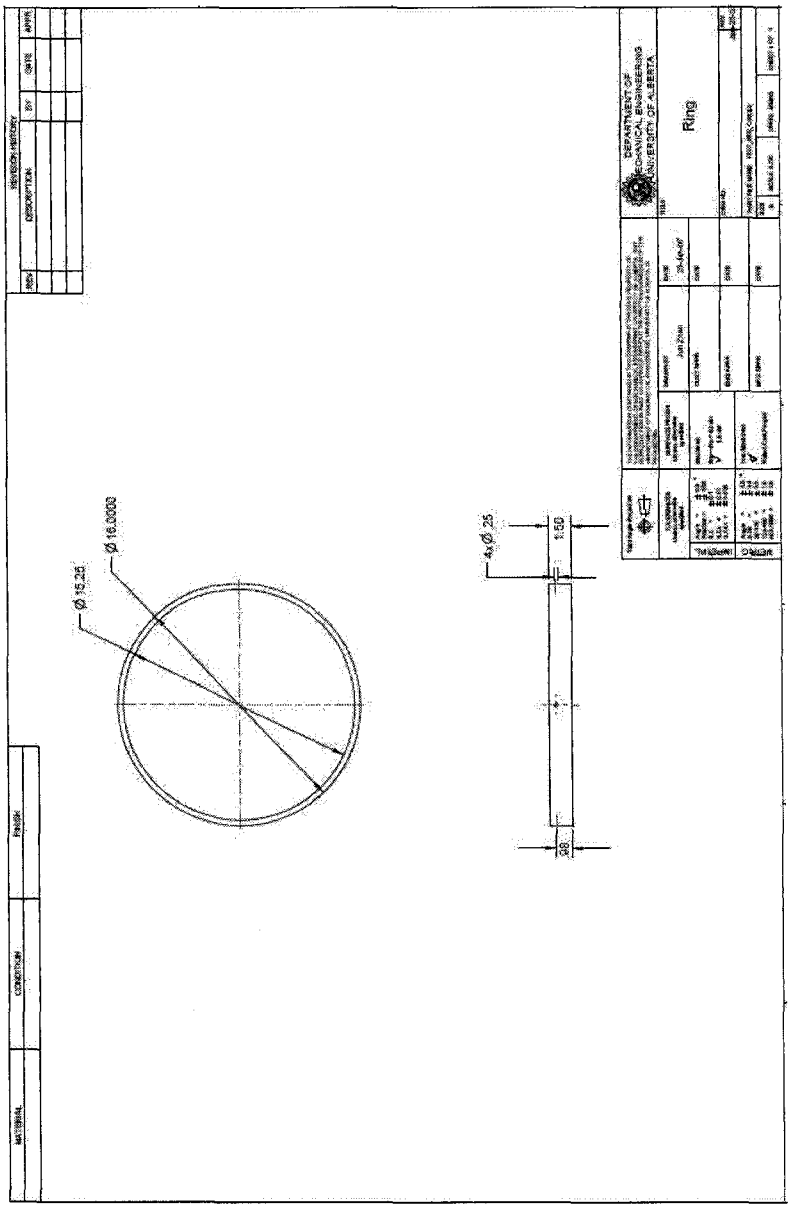

Figure A.8: Ring

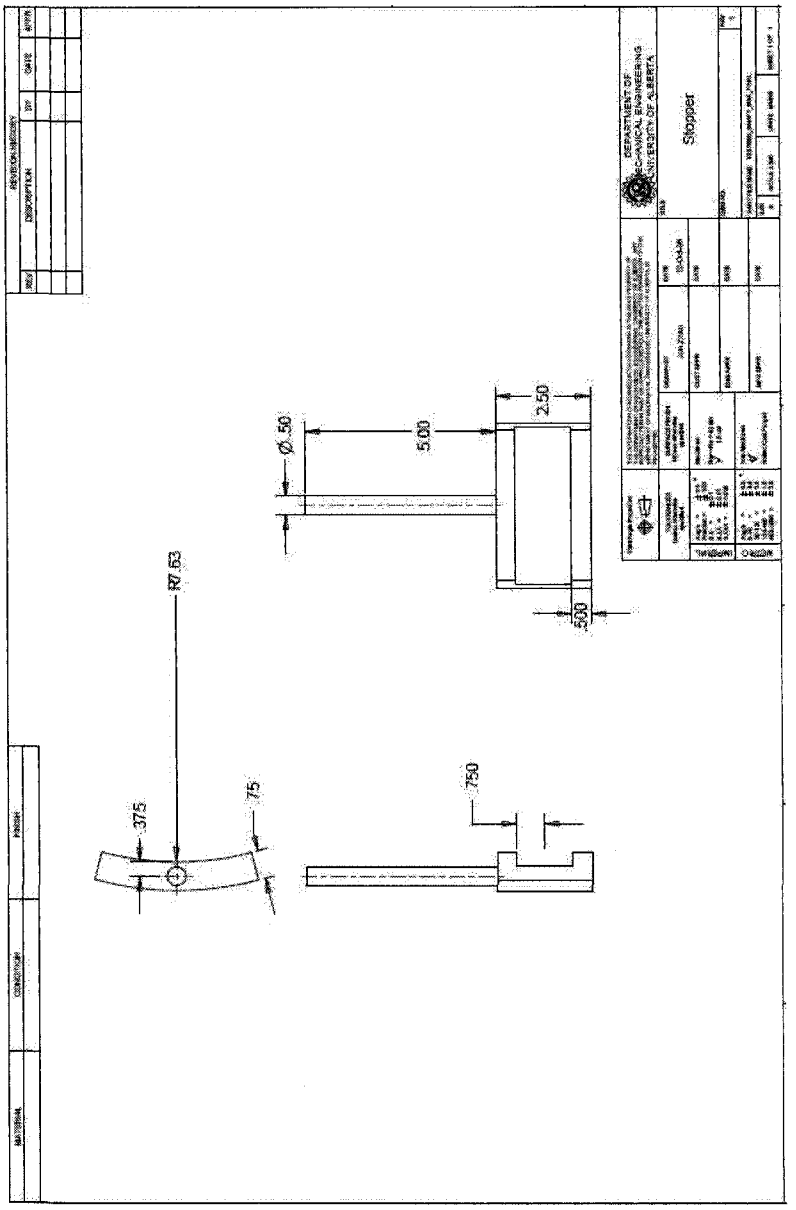

Figure A.9: Stopper

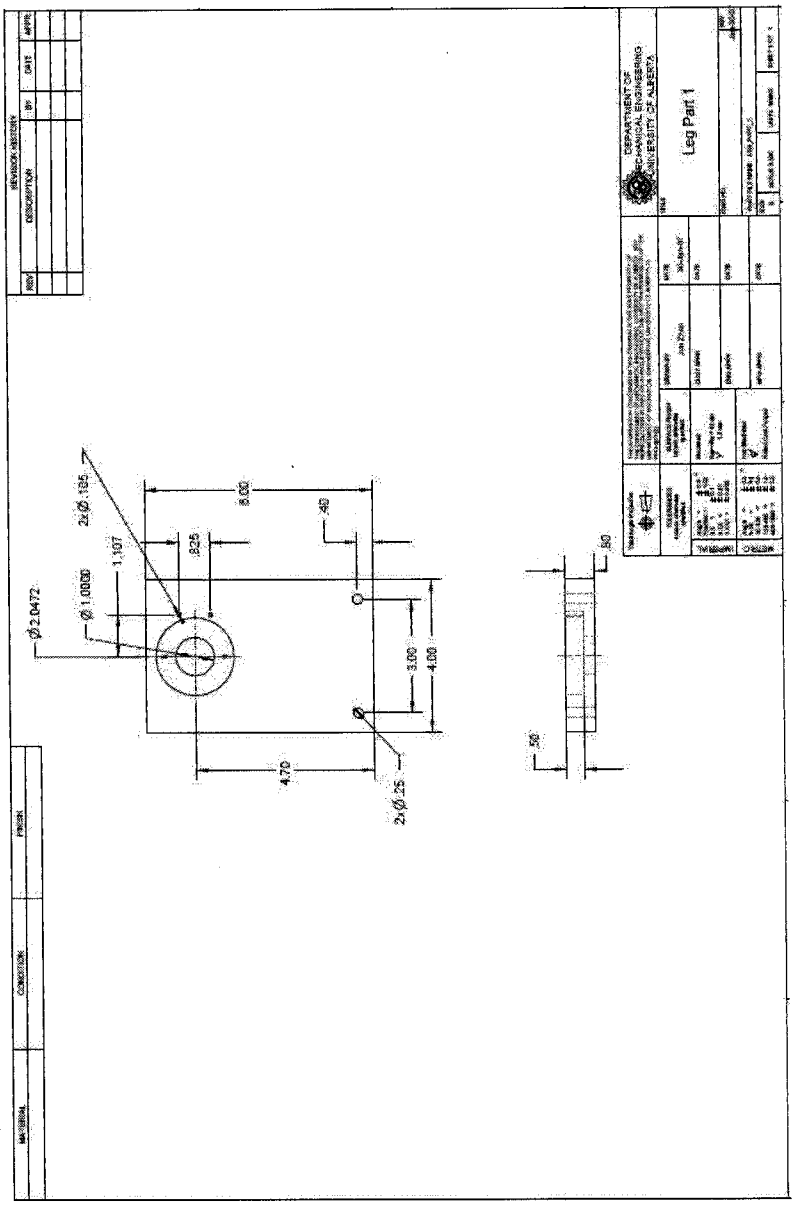

Figure A.10 Leg Part 1

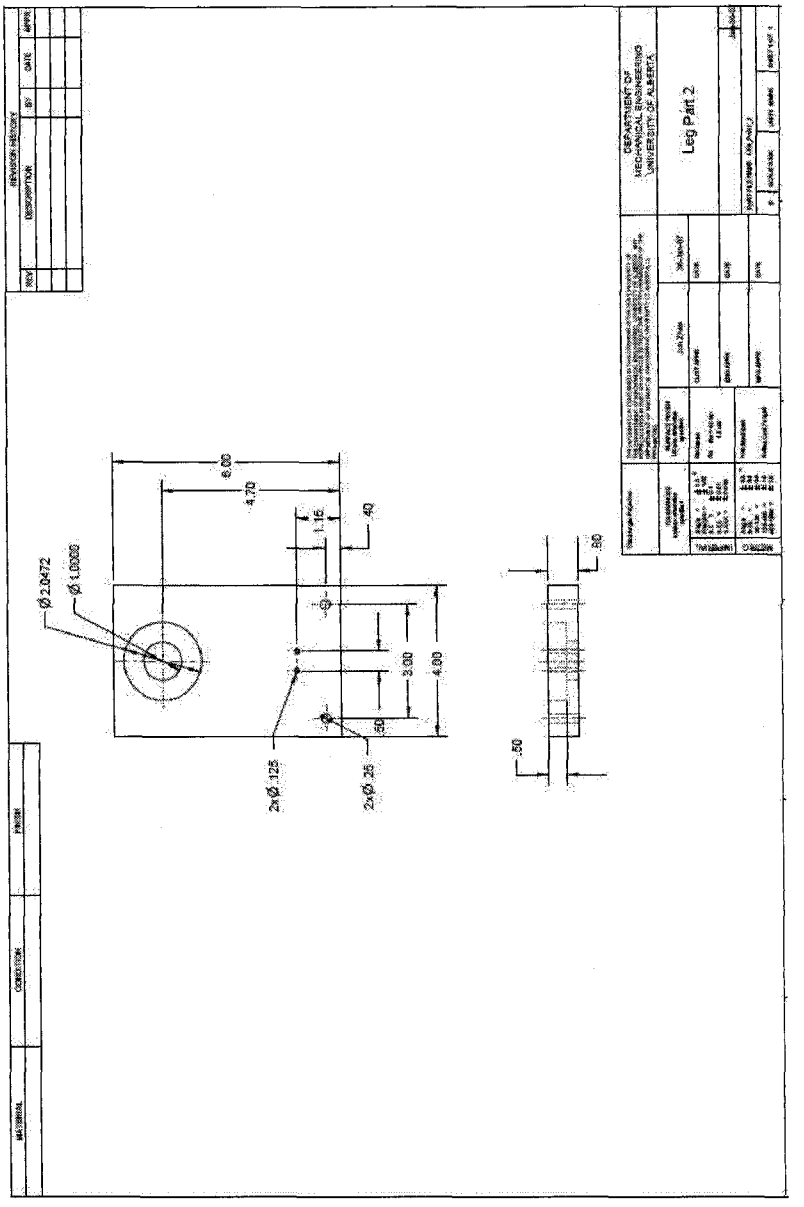

Figure A.11 Leg Part 2

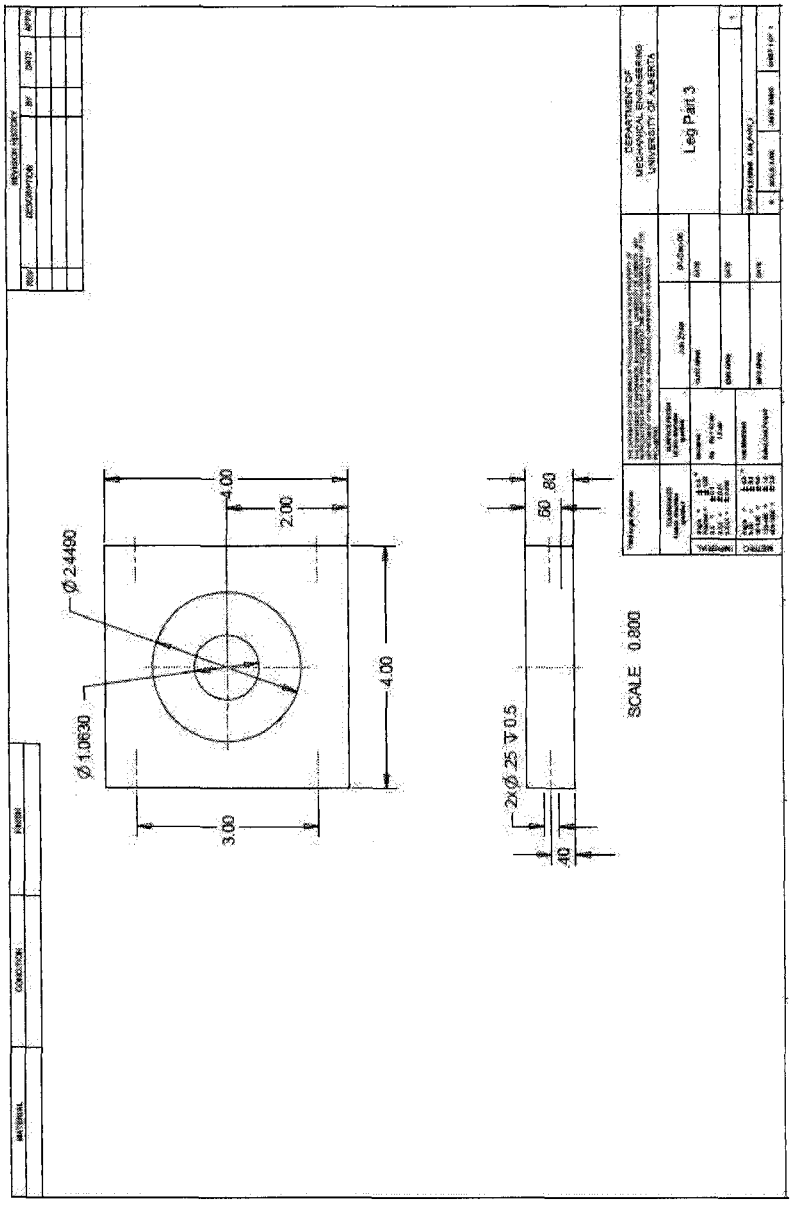

Figure A.12 Leg Part 3

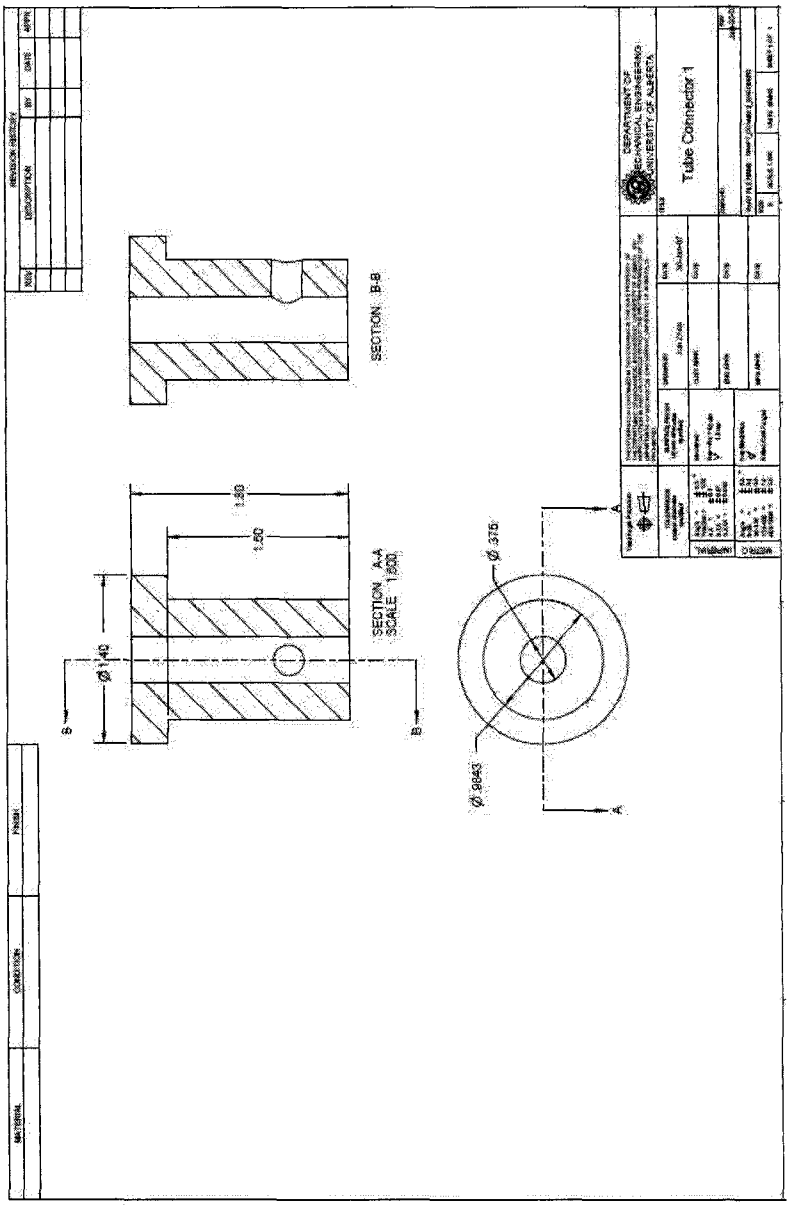

Figure A.13 Tube Connector 1

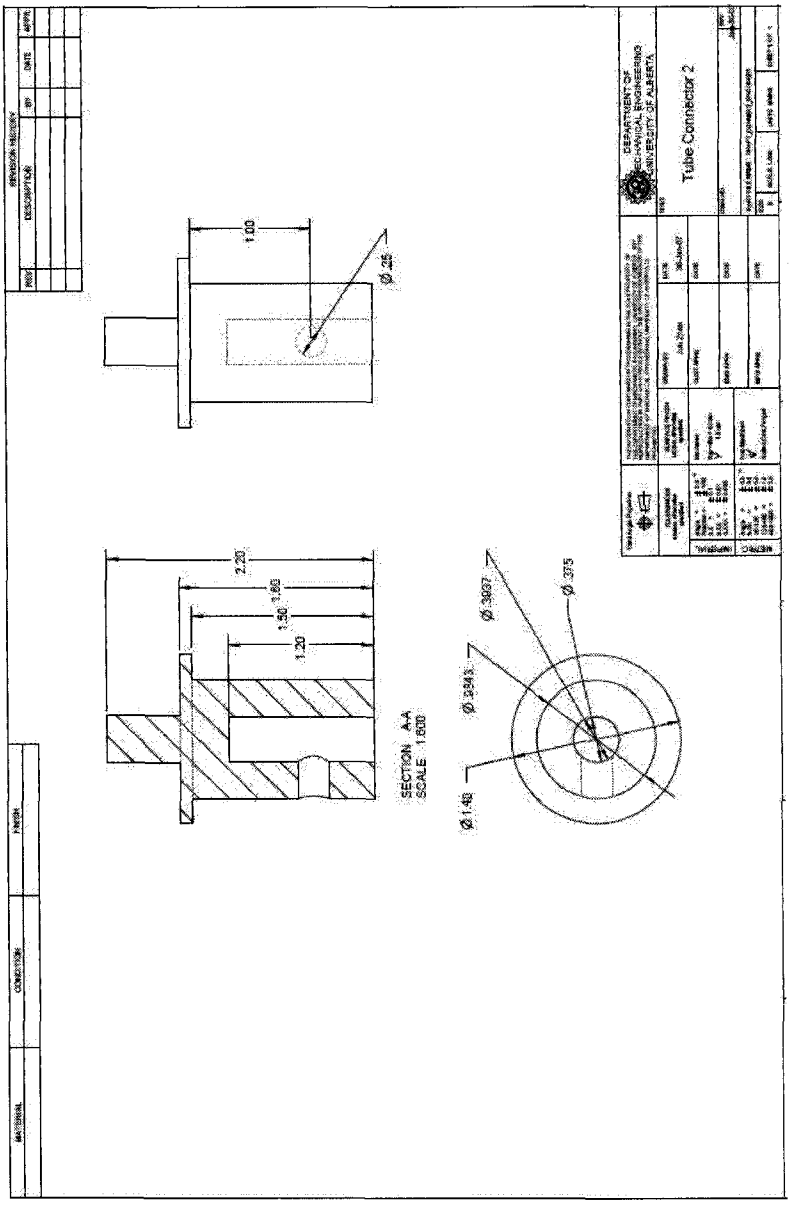

Figure A.14 Tube Connector 2

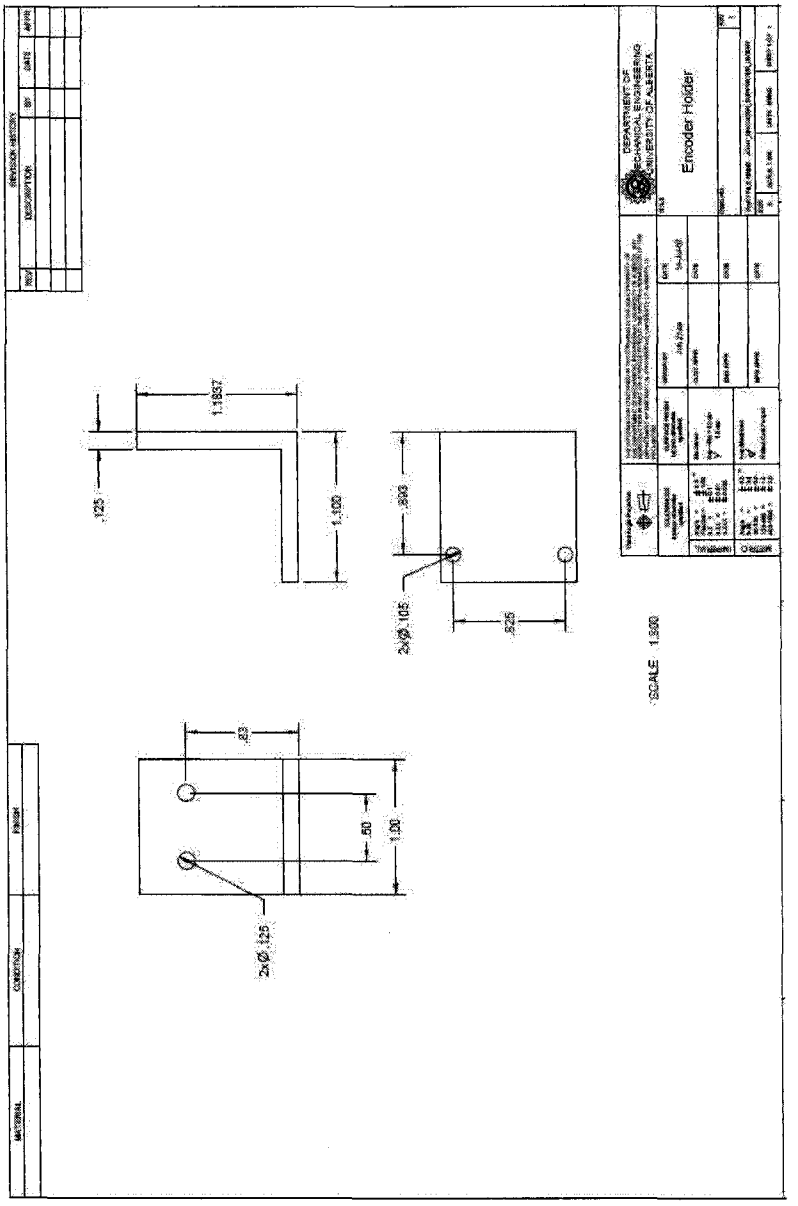

Figure A.15 Encoder Holder Joint 1

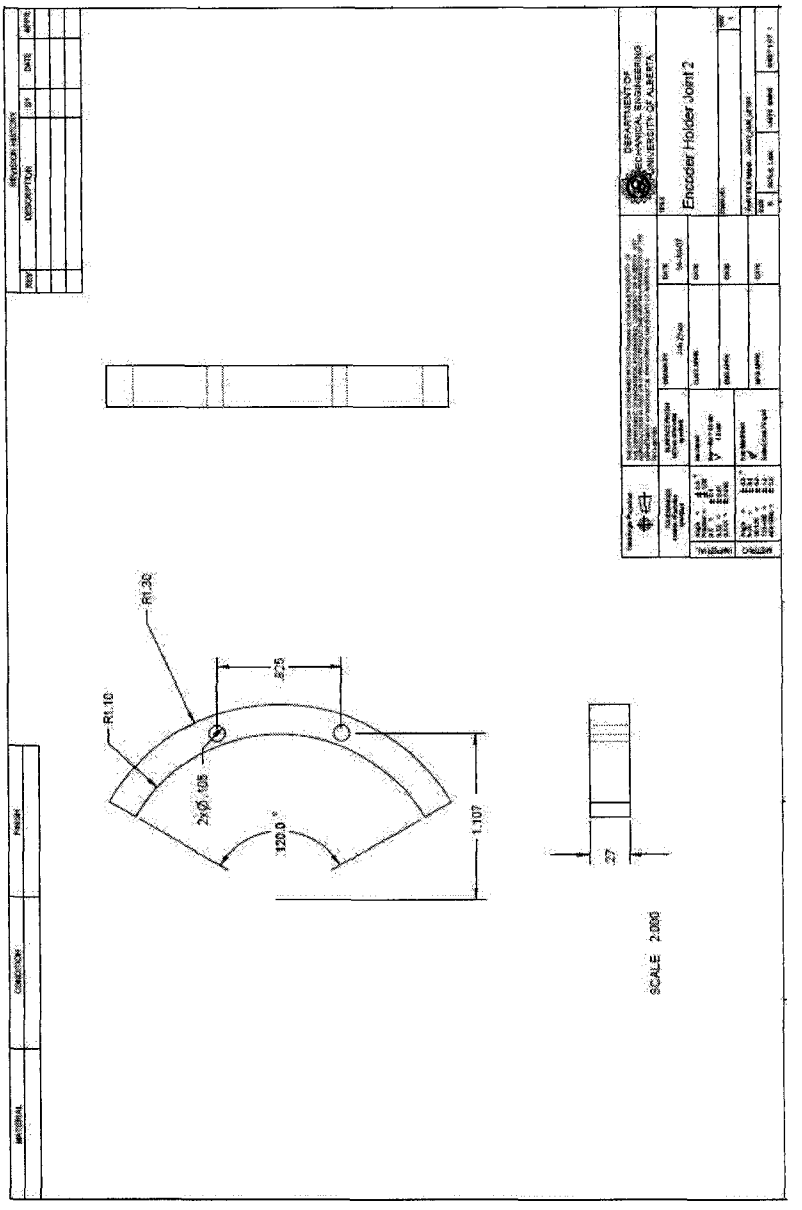

Figure A.16 Encoder Holder Joint 2

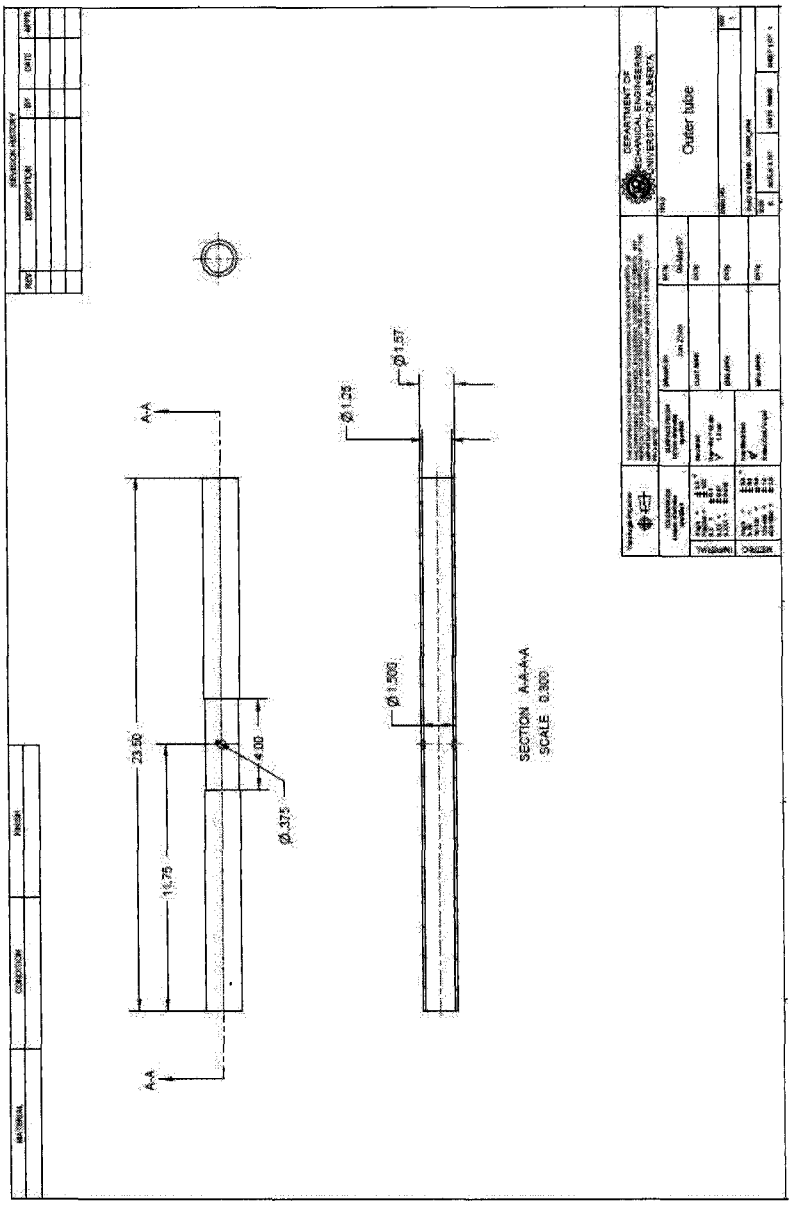

Figure A.17 Outer Tube

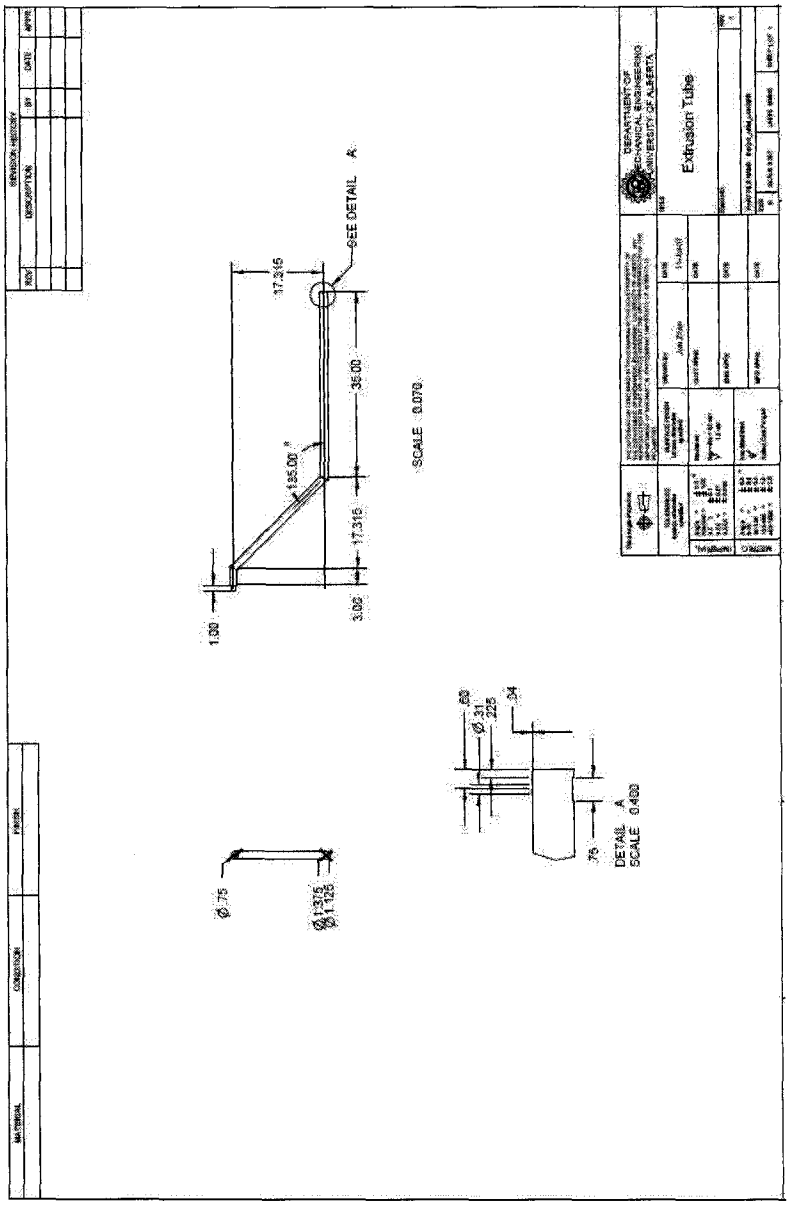

Figure A.18 Extrusion Tube

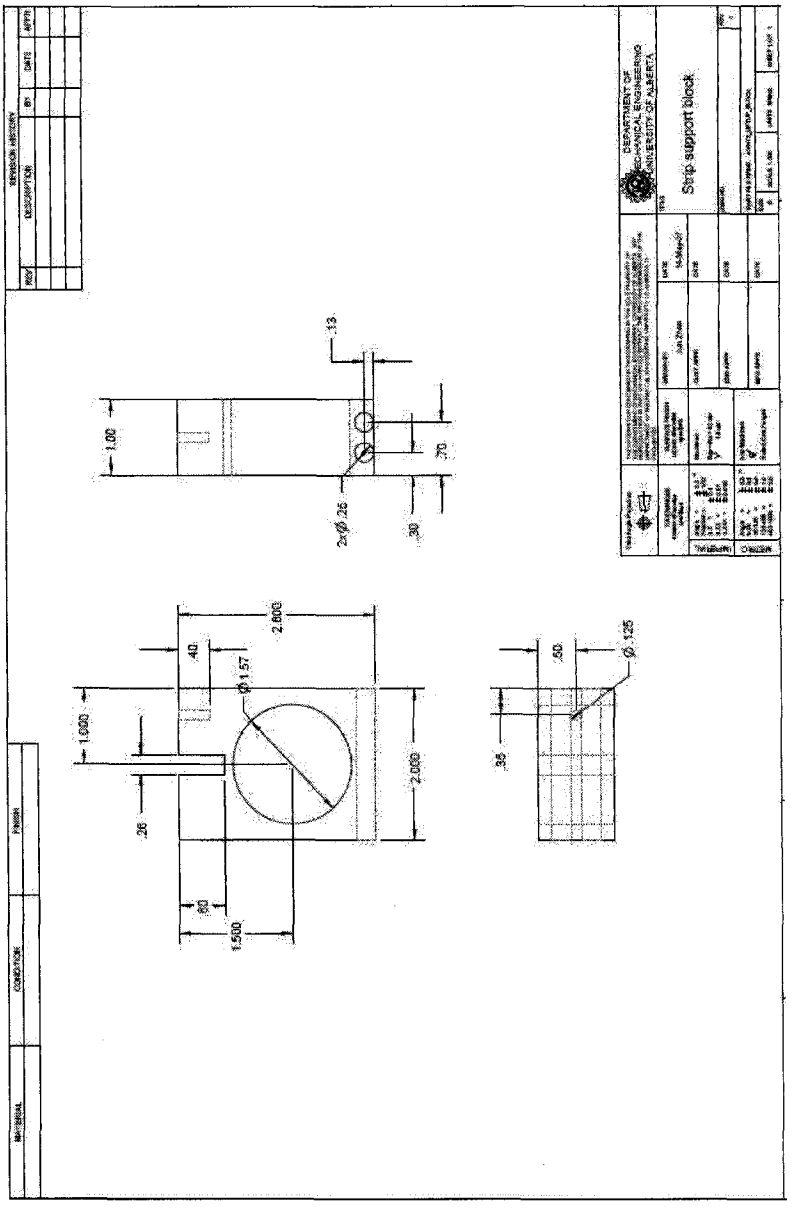

Figure A.19 Strip Support Block

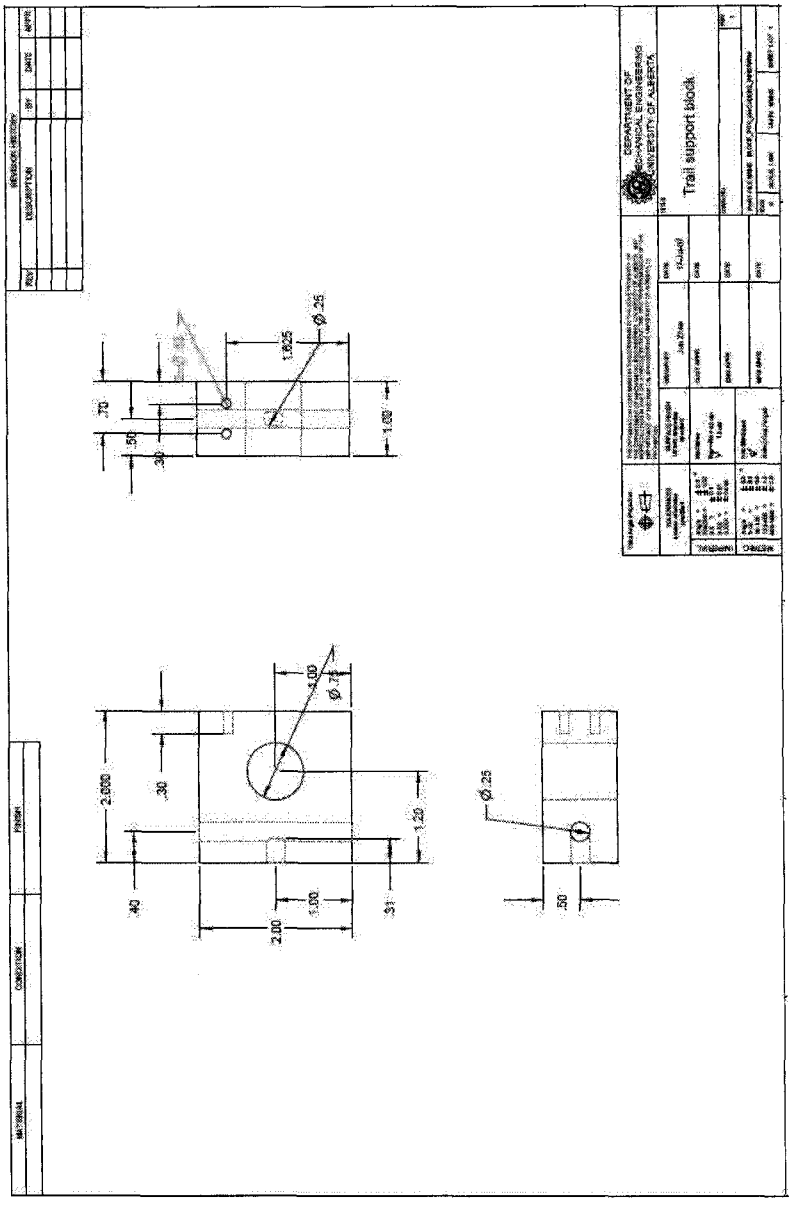

Figure A.20 Trail Support Block

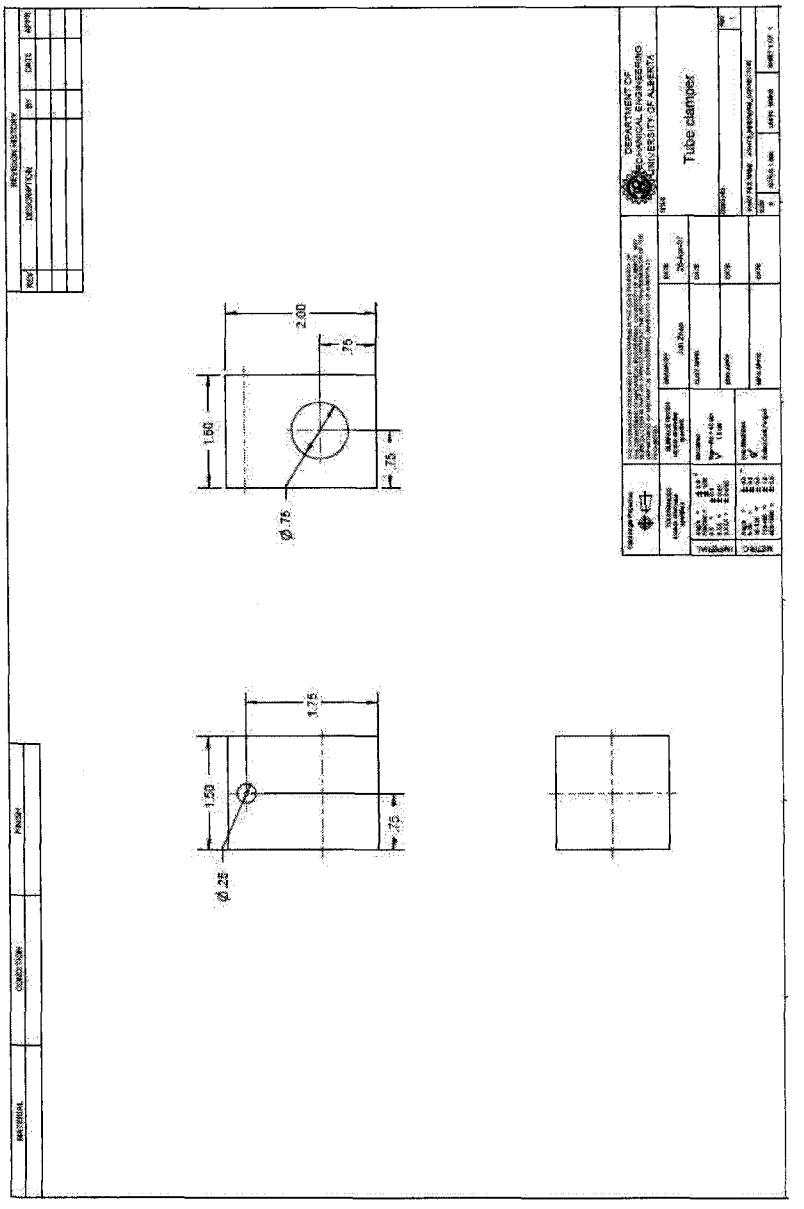

Figure A.21 Tube Clamper

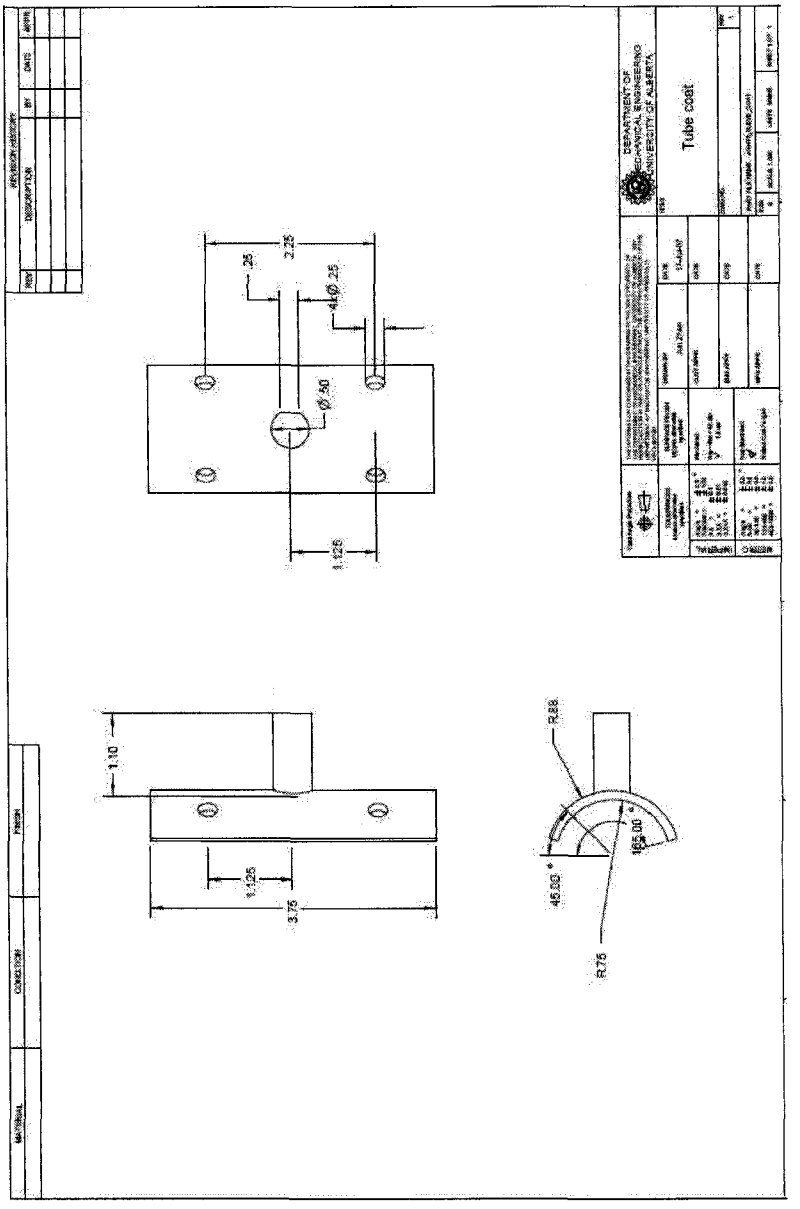

Figure A.22 Tube Coat

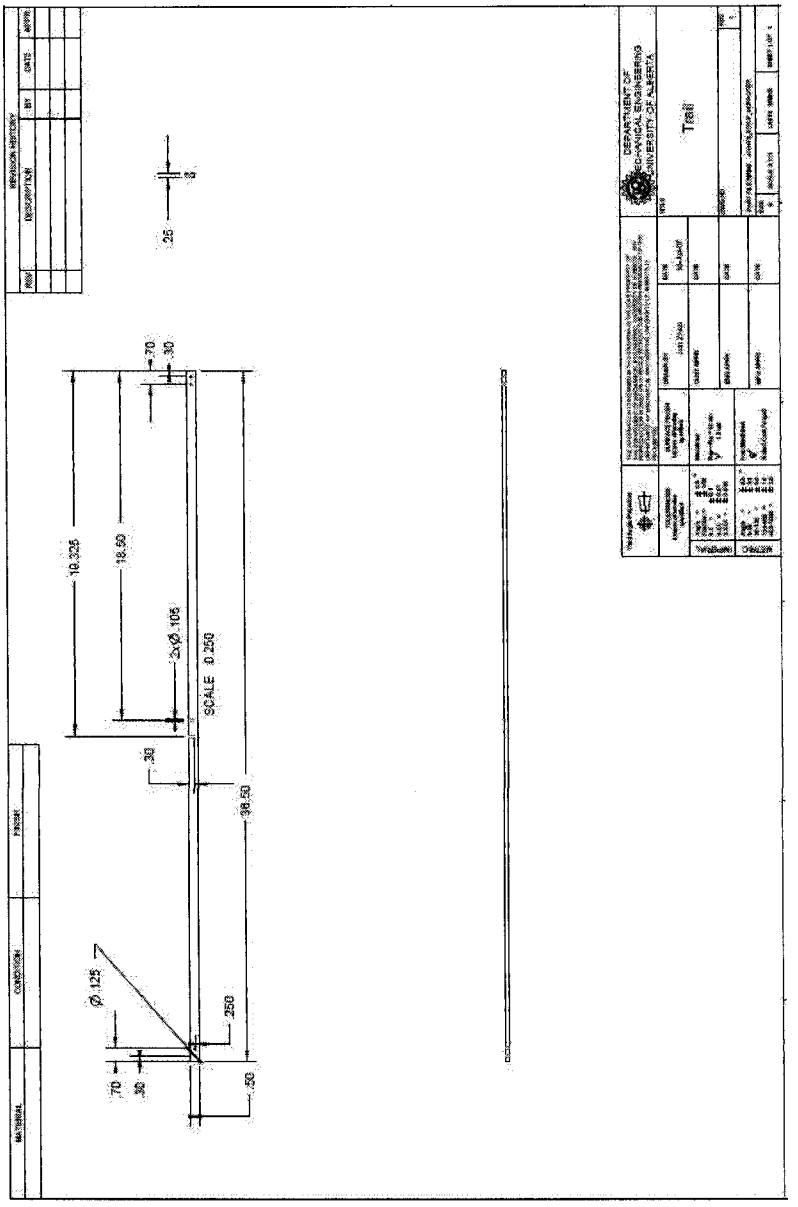

Figure A.23 Trail

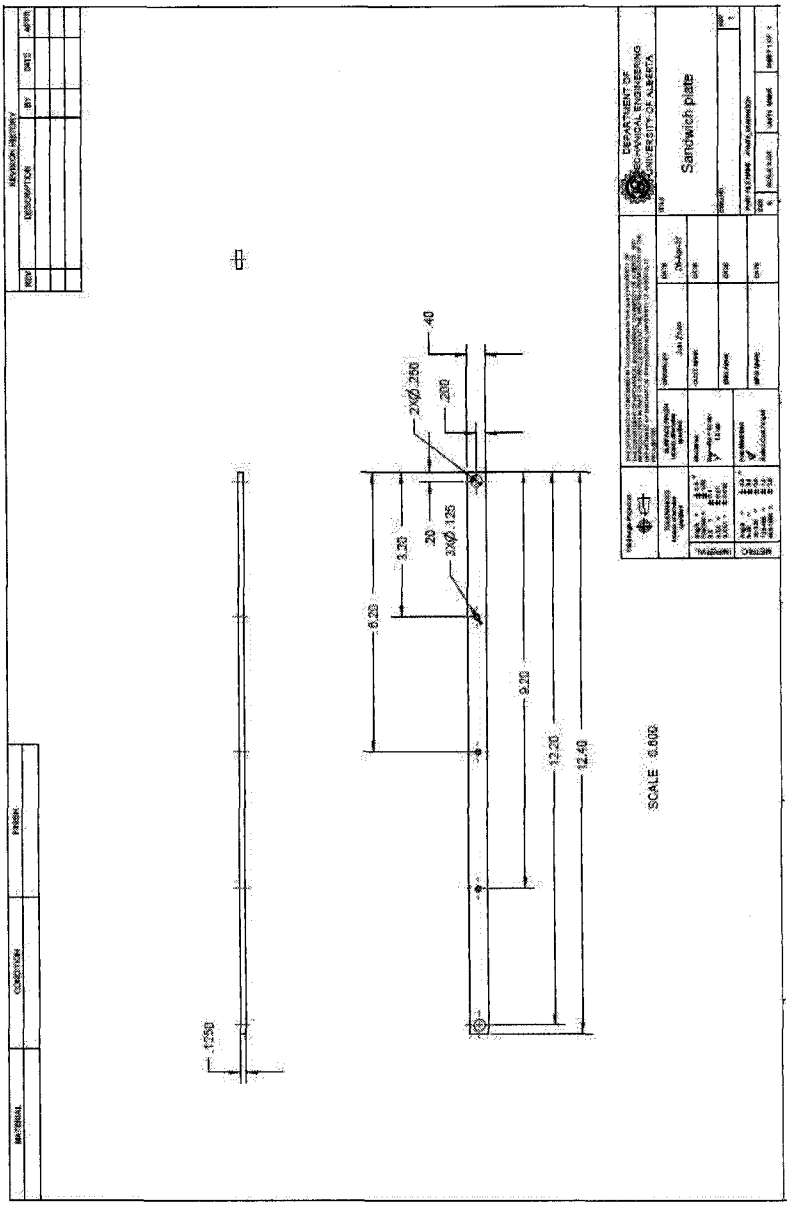

Figure A.24 Sandwich Plate

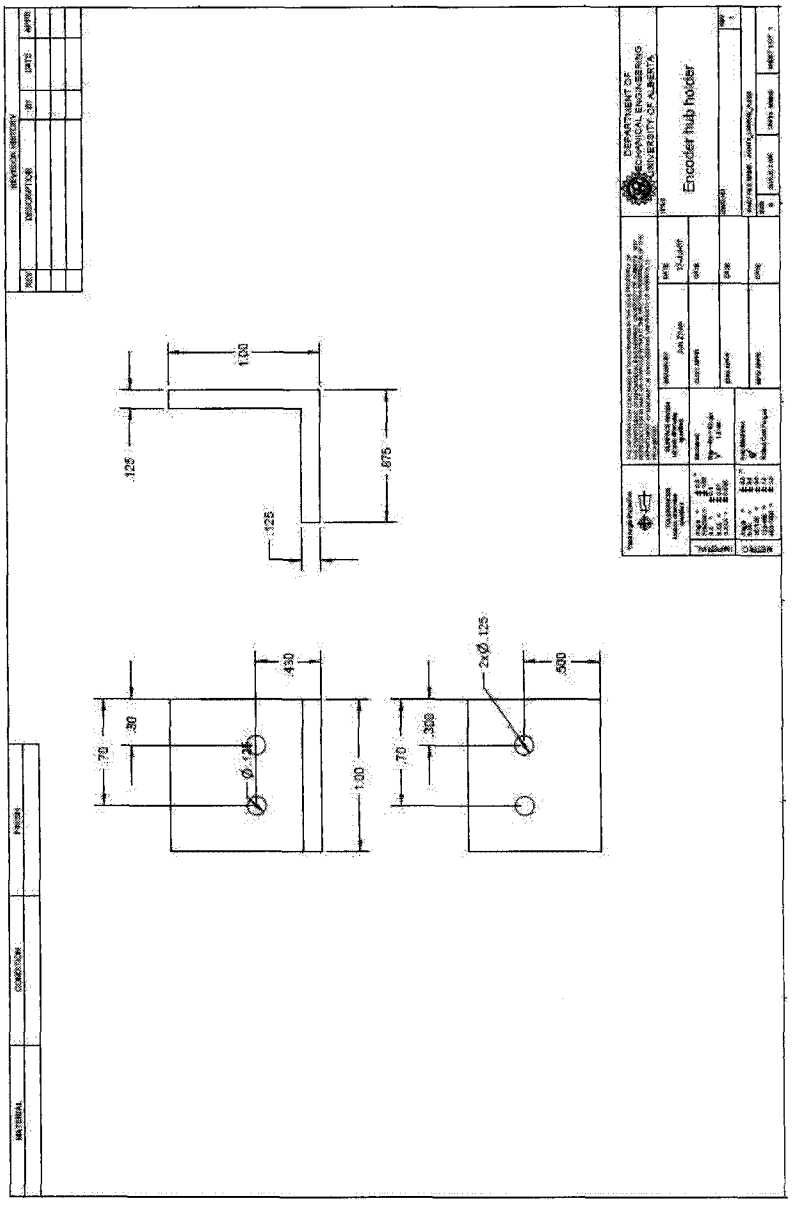

Figure A.25 Encoder Hub Holder

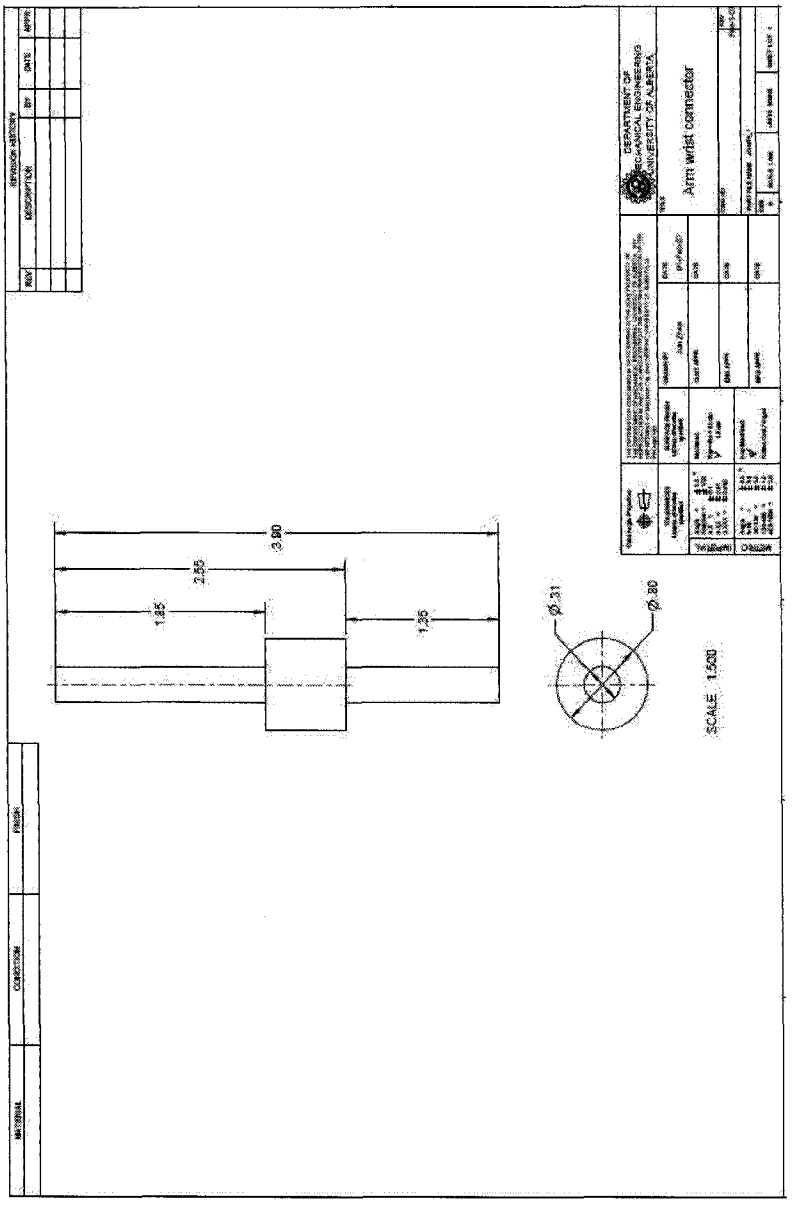

Figure A.26 Arm Wrist Connector

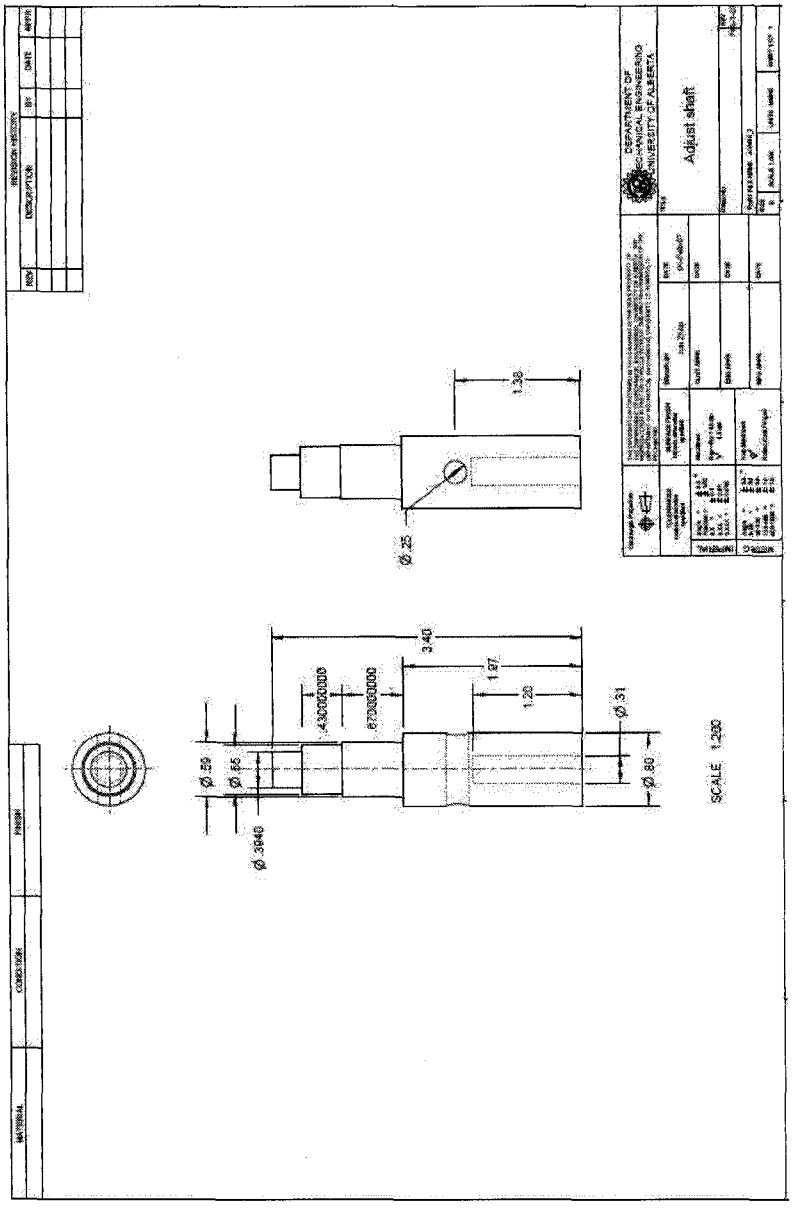

Figure A.27 Adjust Shaft

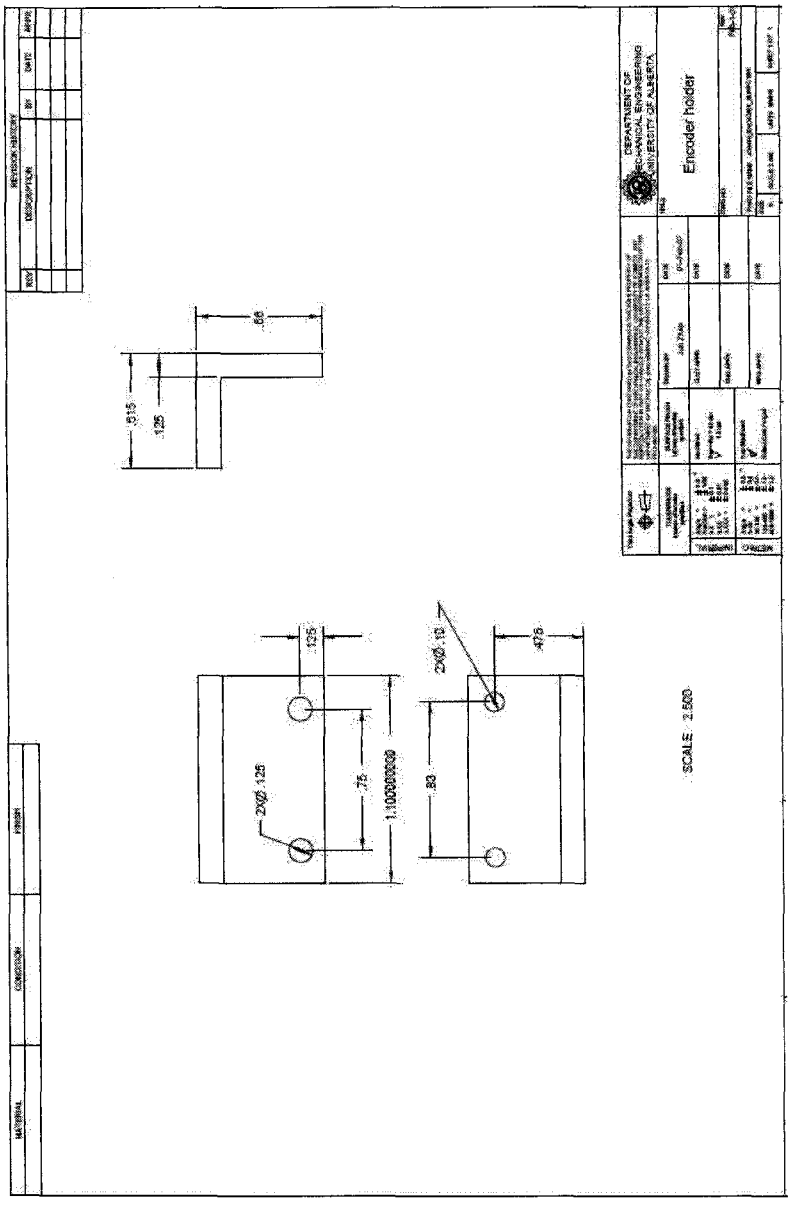

Figure A.28 Encoder Hub Holder

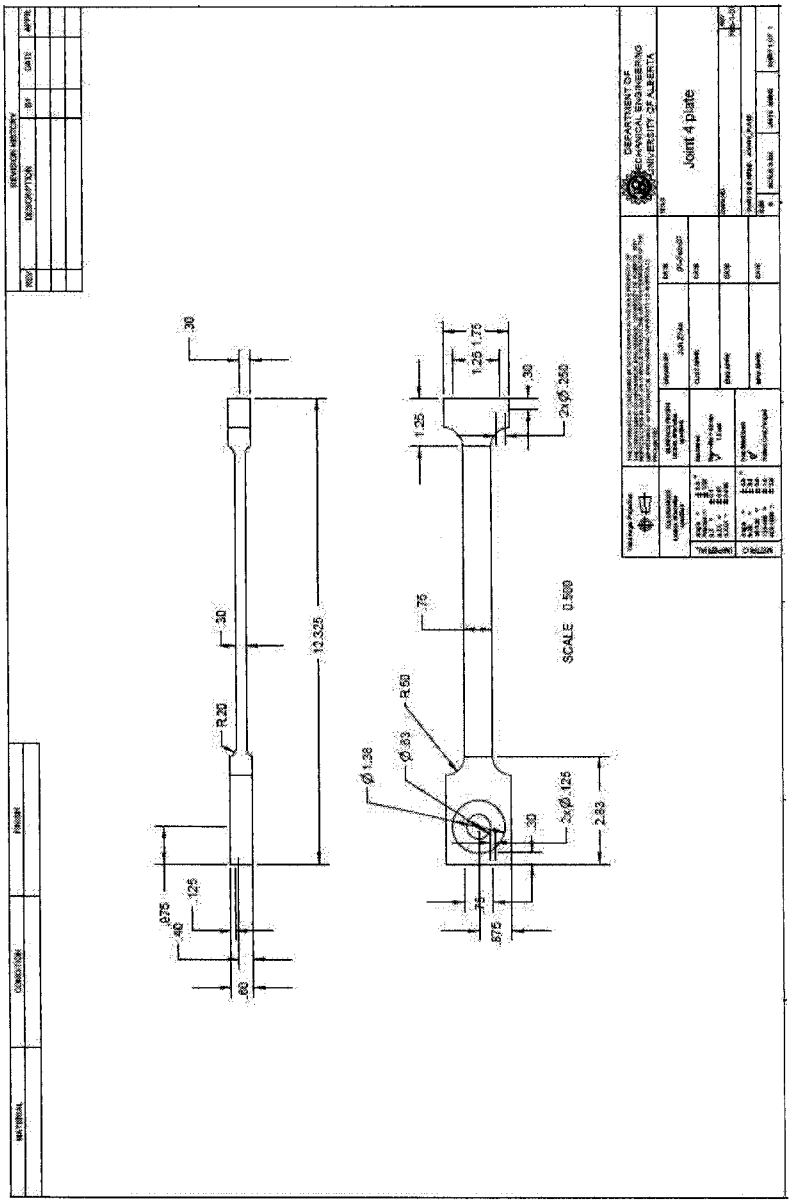

Figure A.29 Joint 4 Plate

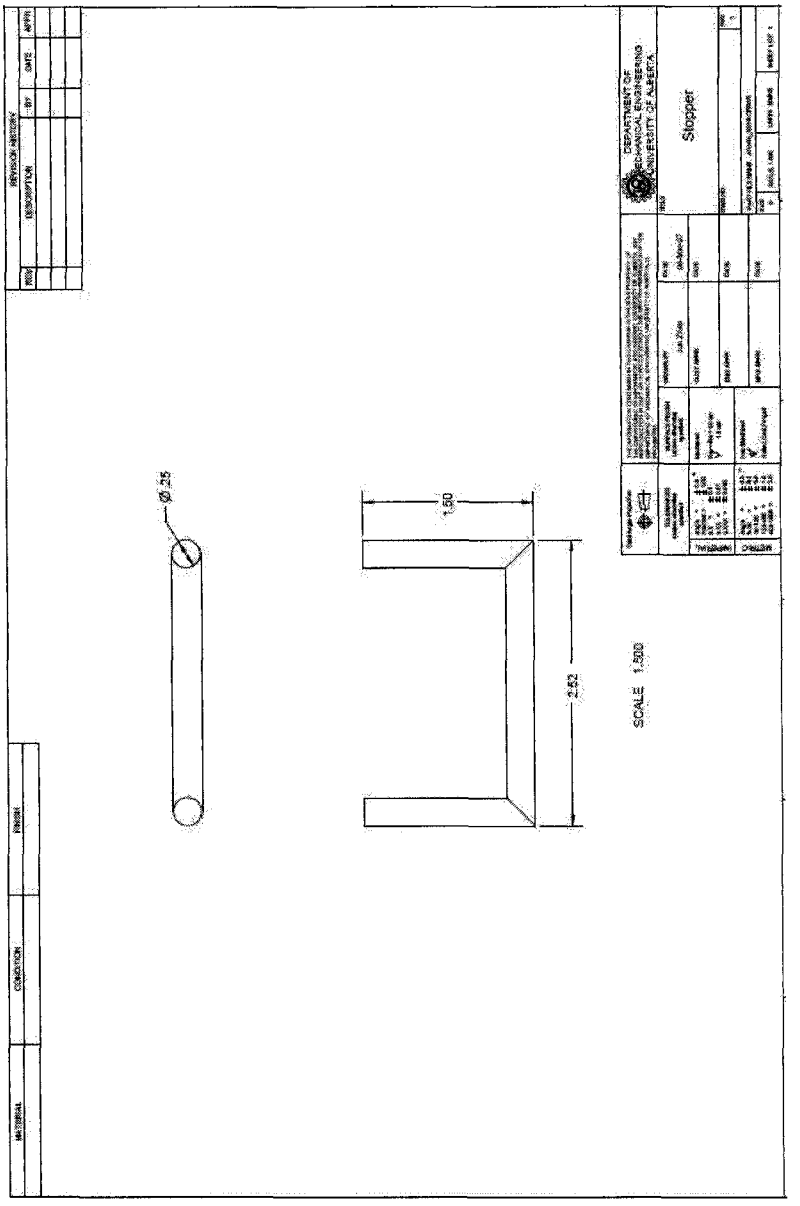

Figure A.30 Stopper

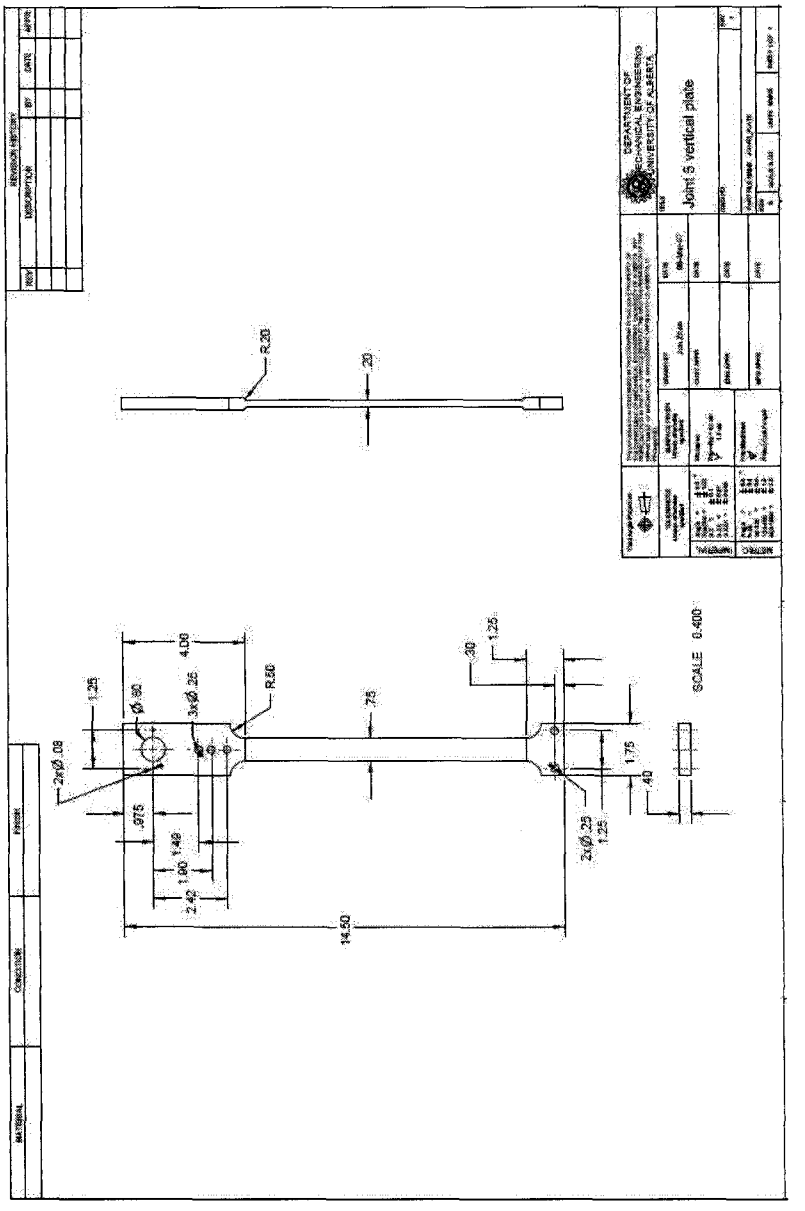

Figure A.31 Joint 5 Vertical Plate

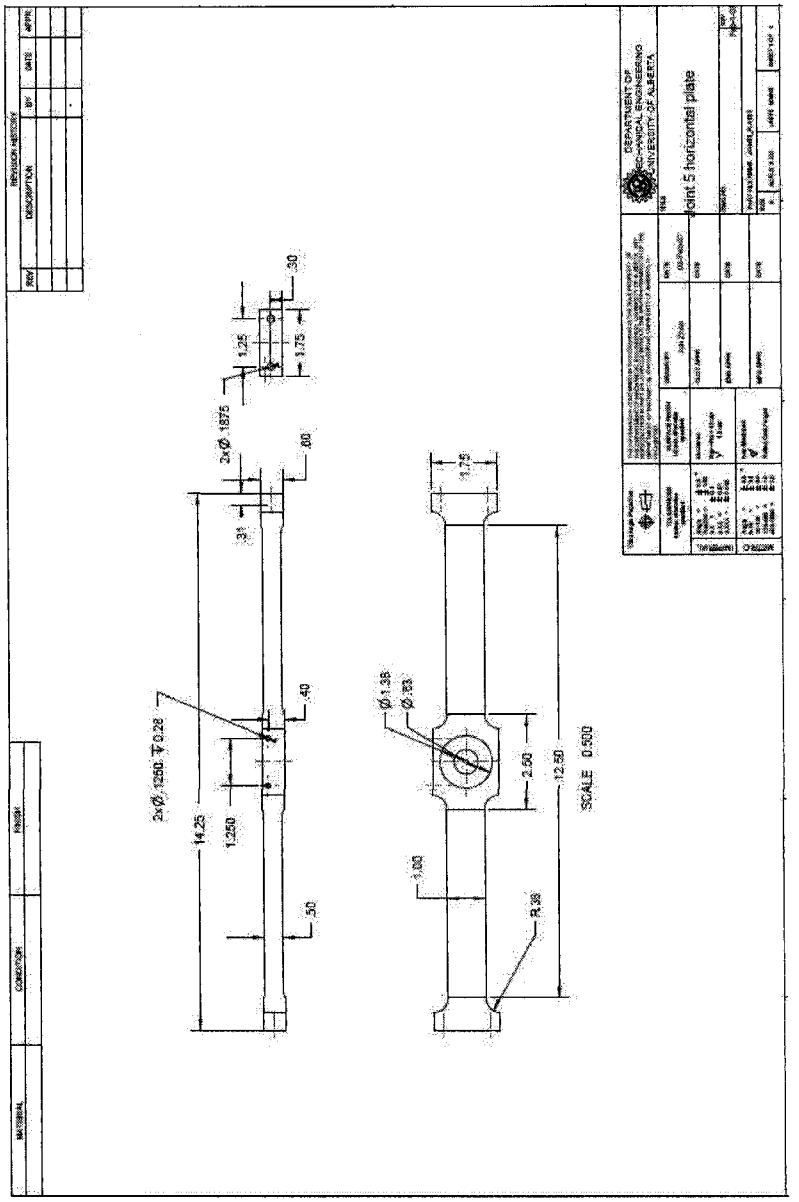

Figure A.32 Joint 5 Horizontal Plate

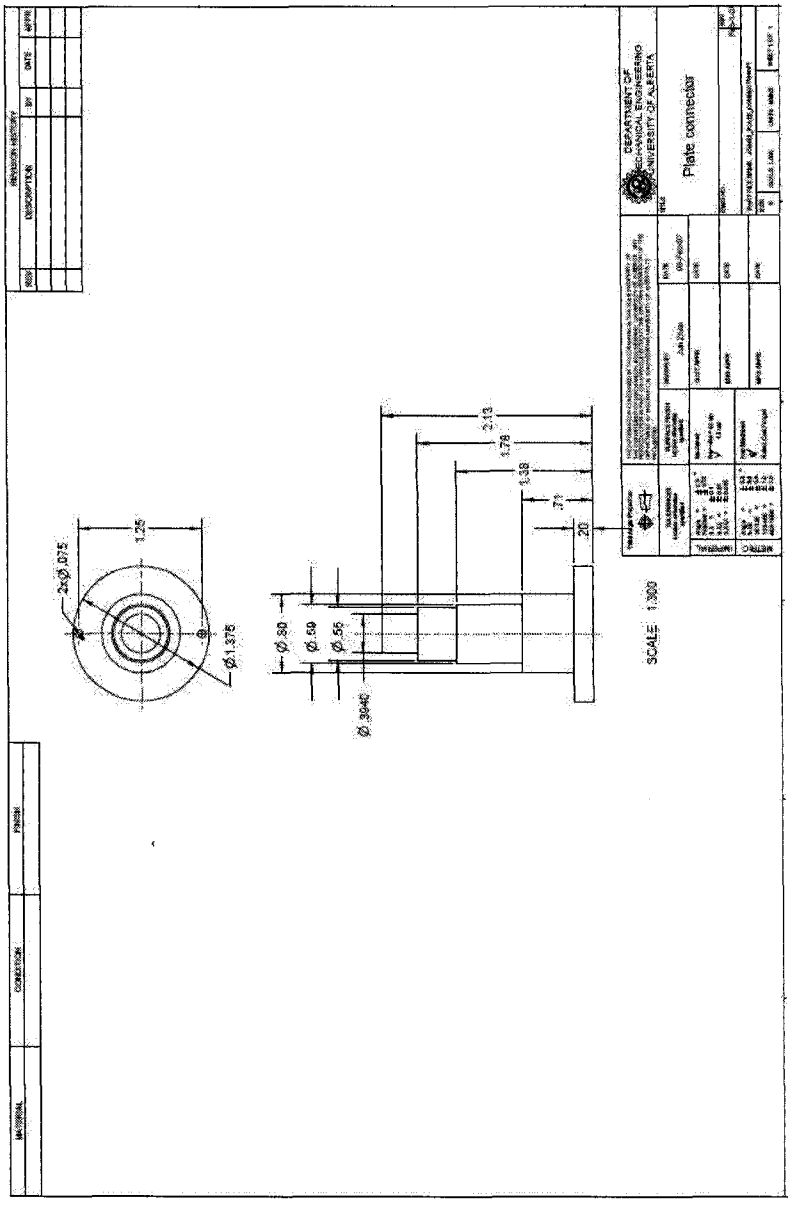

Figure A.33 Joint 5 and 6 Plate Connector

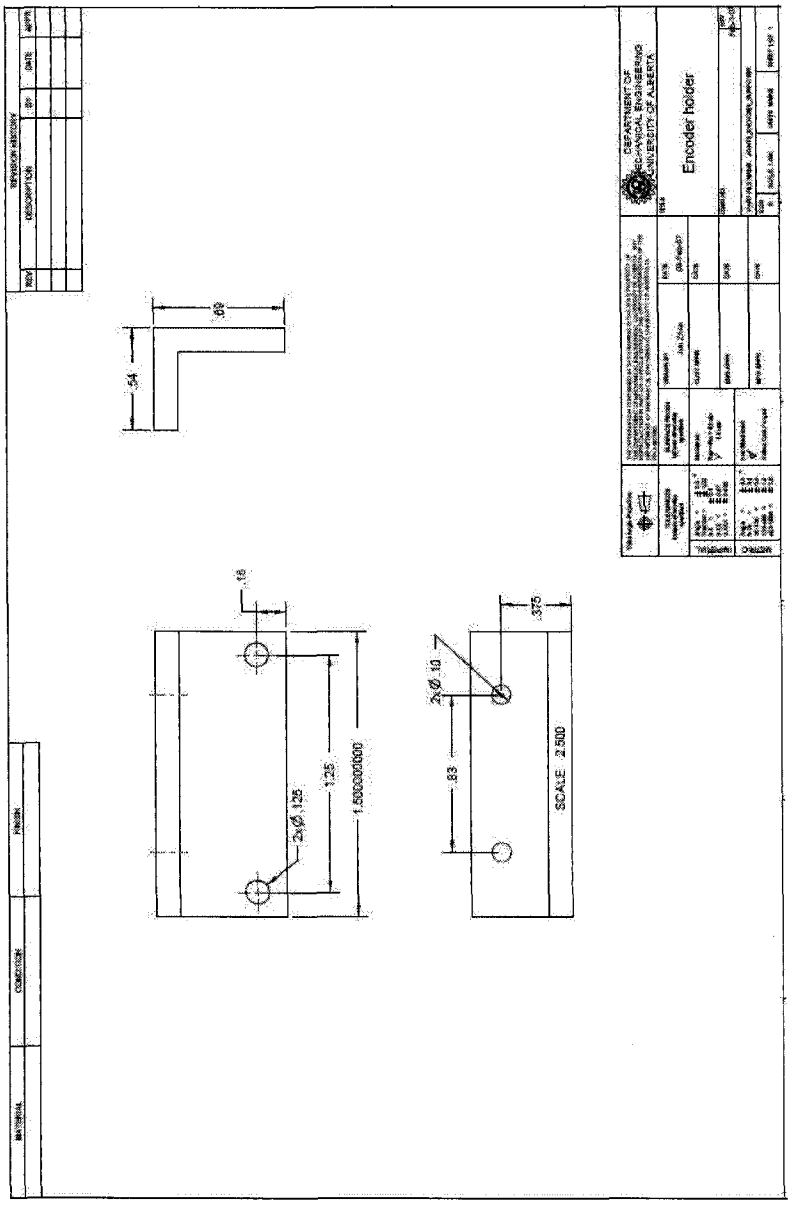

Figure A.34 Joint 5 and 6 Encoder Holder

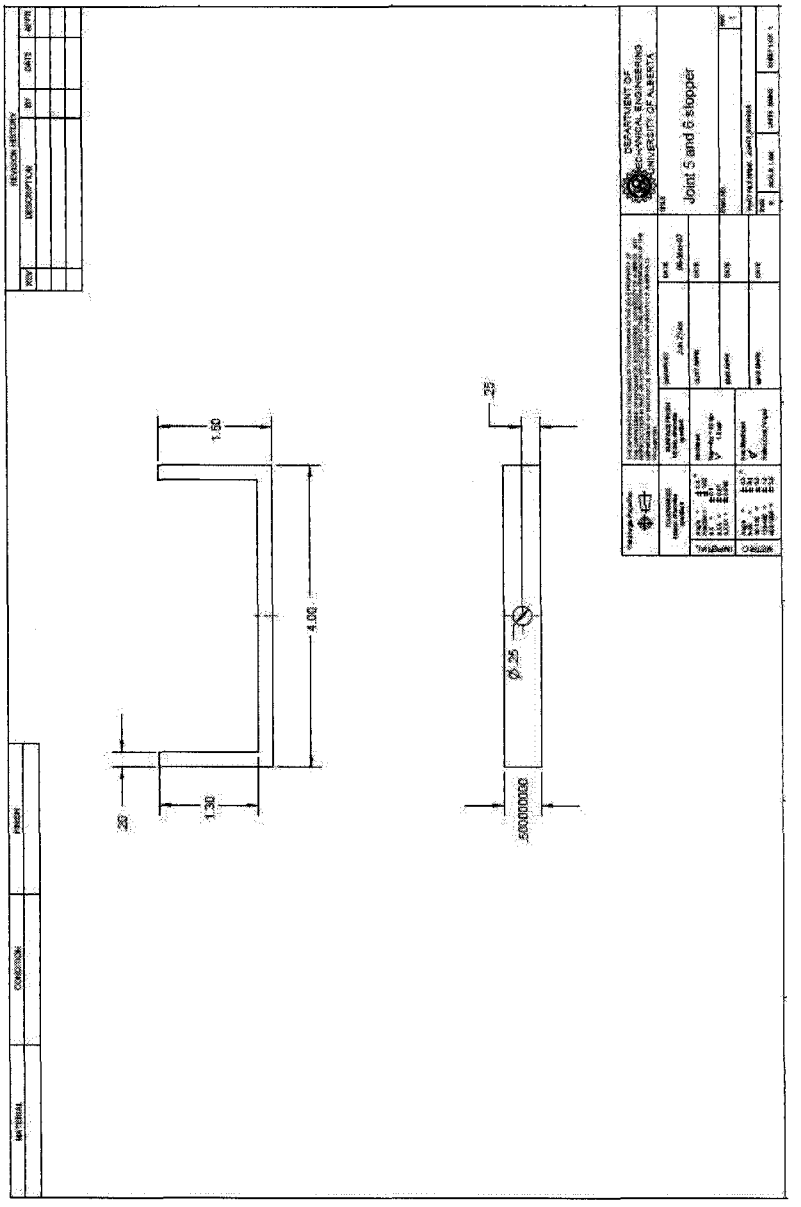

Figure A.35 Joint 5 and 6 Stopper

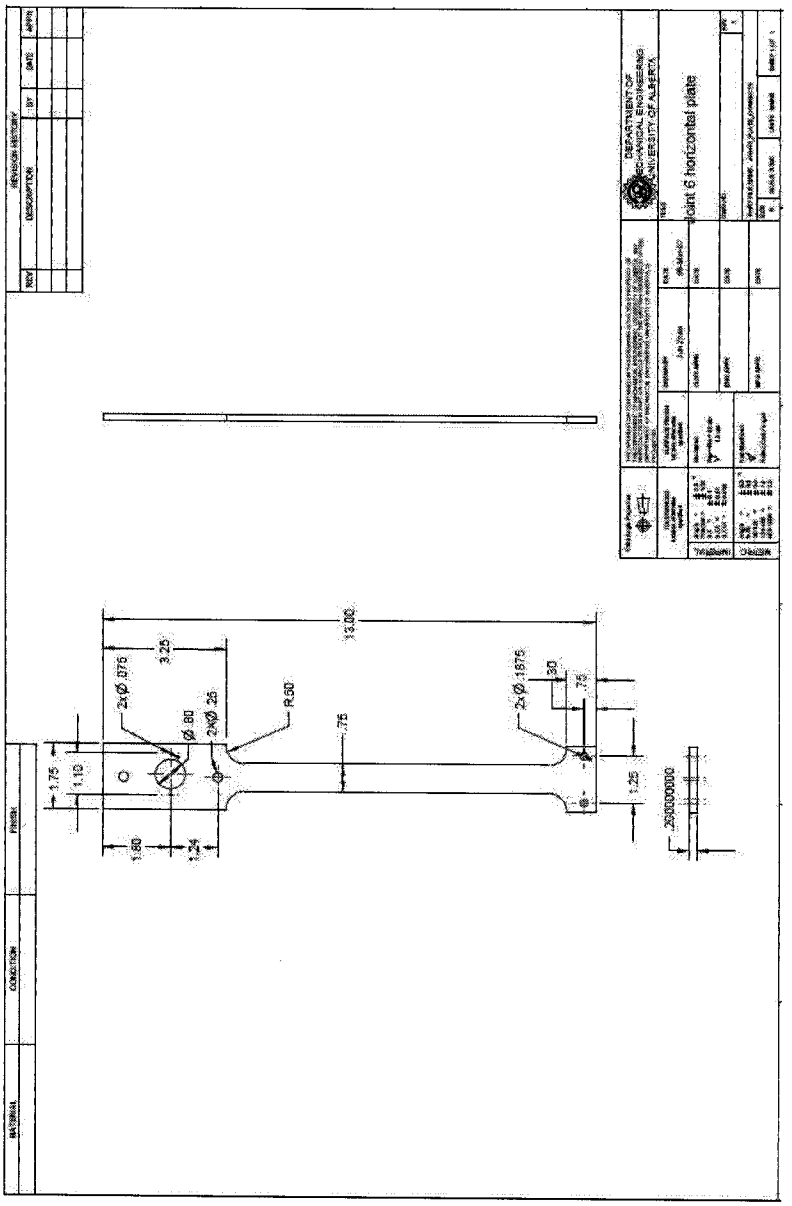

Figure A.36 Joint 6 Horizontal Plate

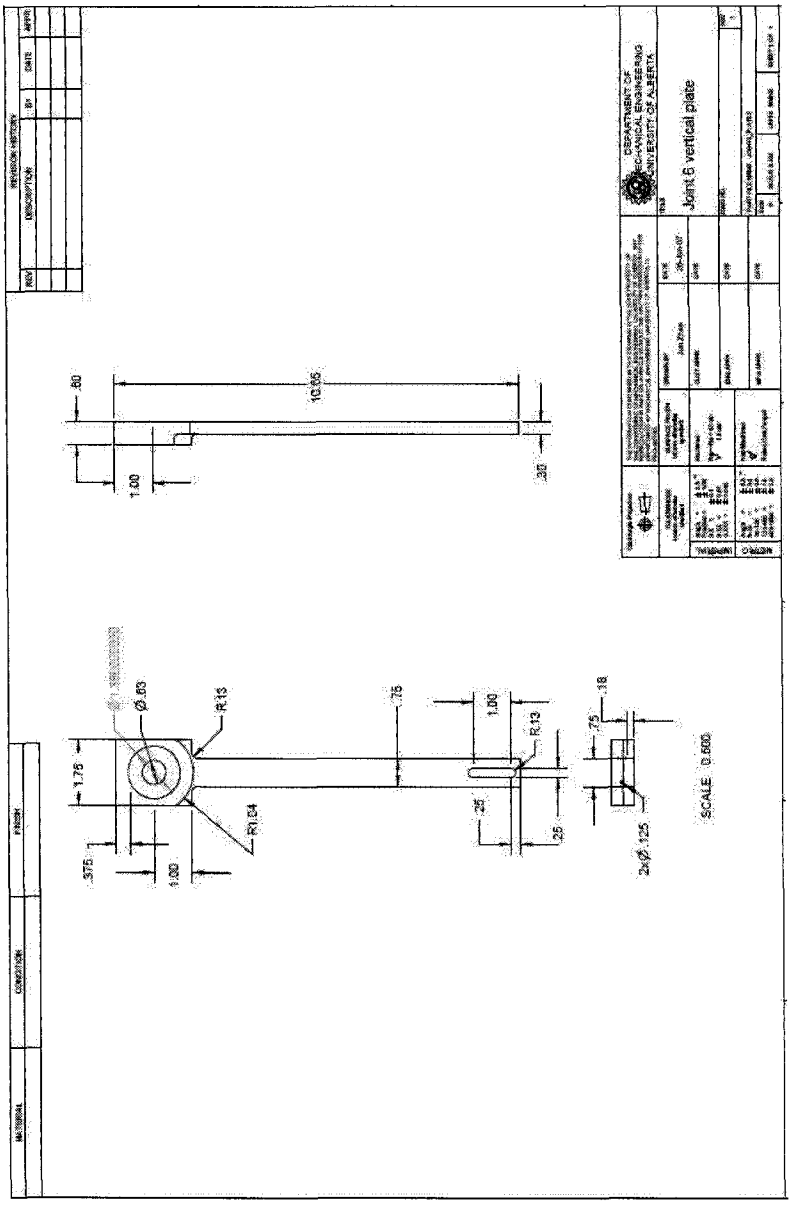

Figure A.37 Joint 6 Vertical Plate

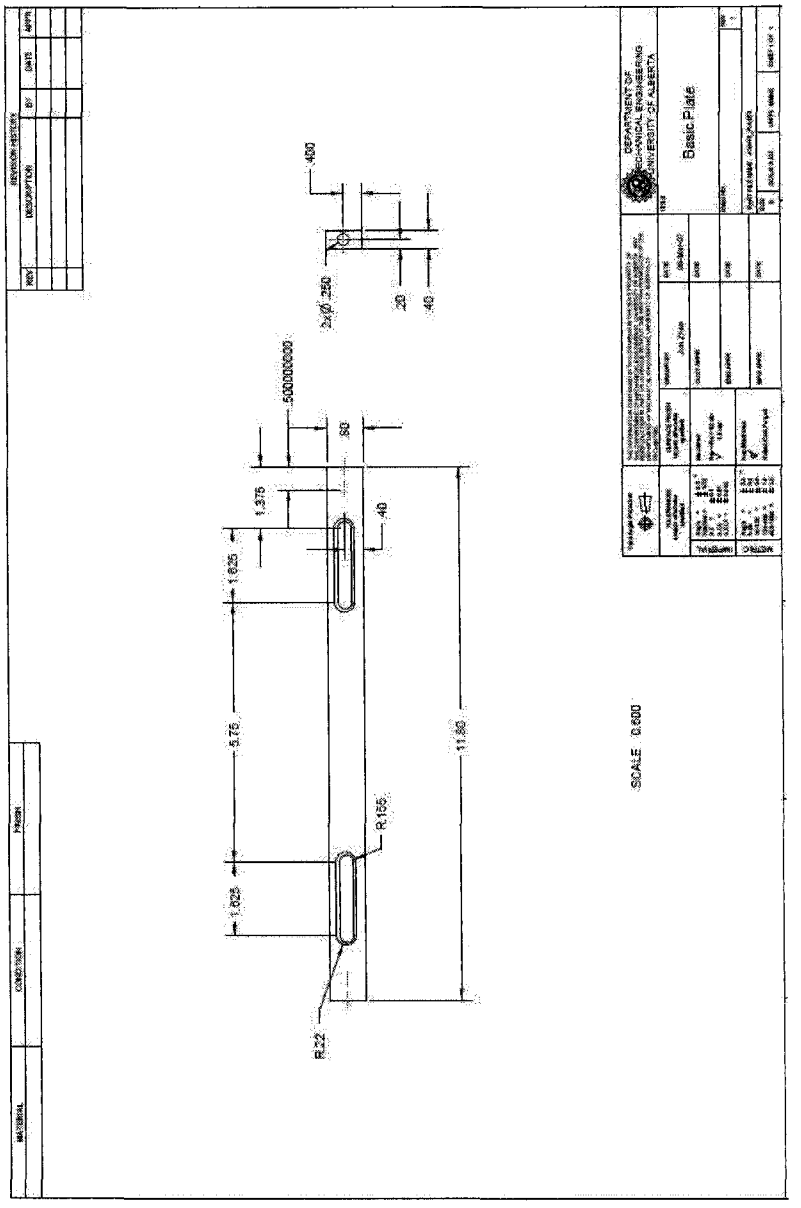

Figure A.38 Joint 6 Basic Plate

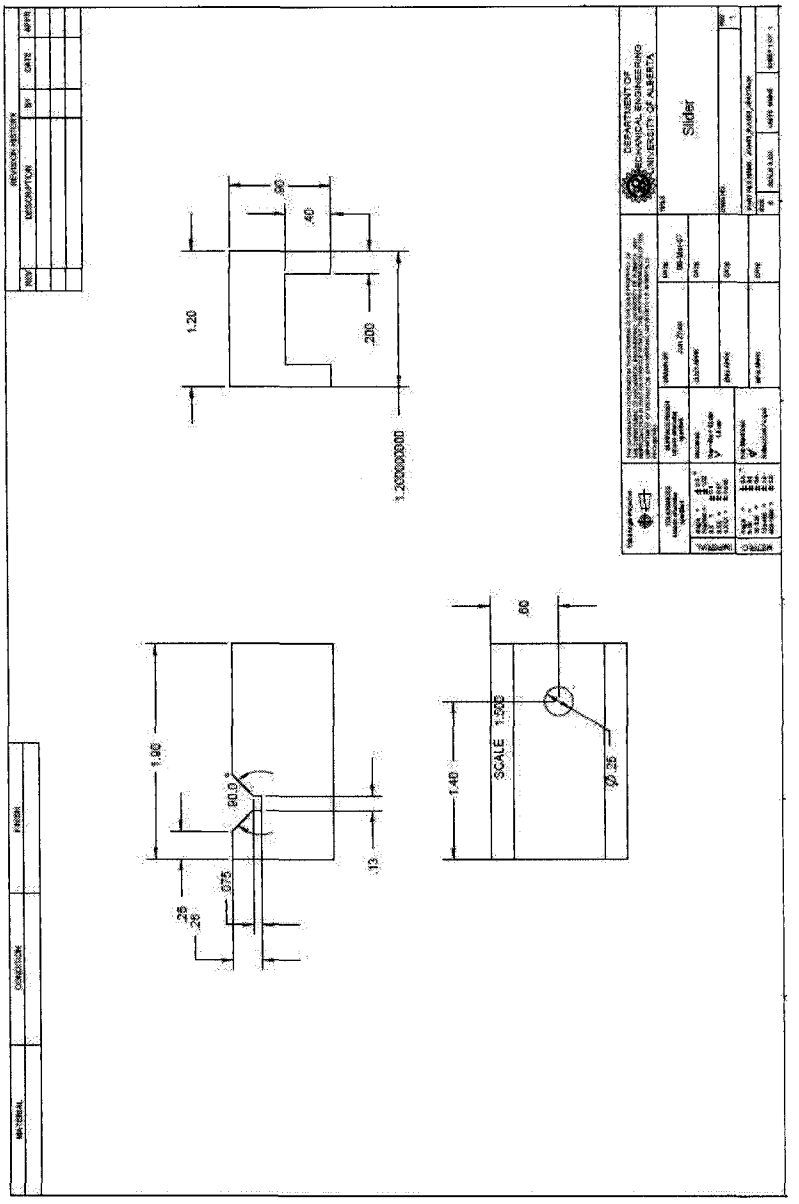

Figure A.39 Joint 6 Slider

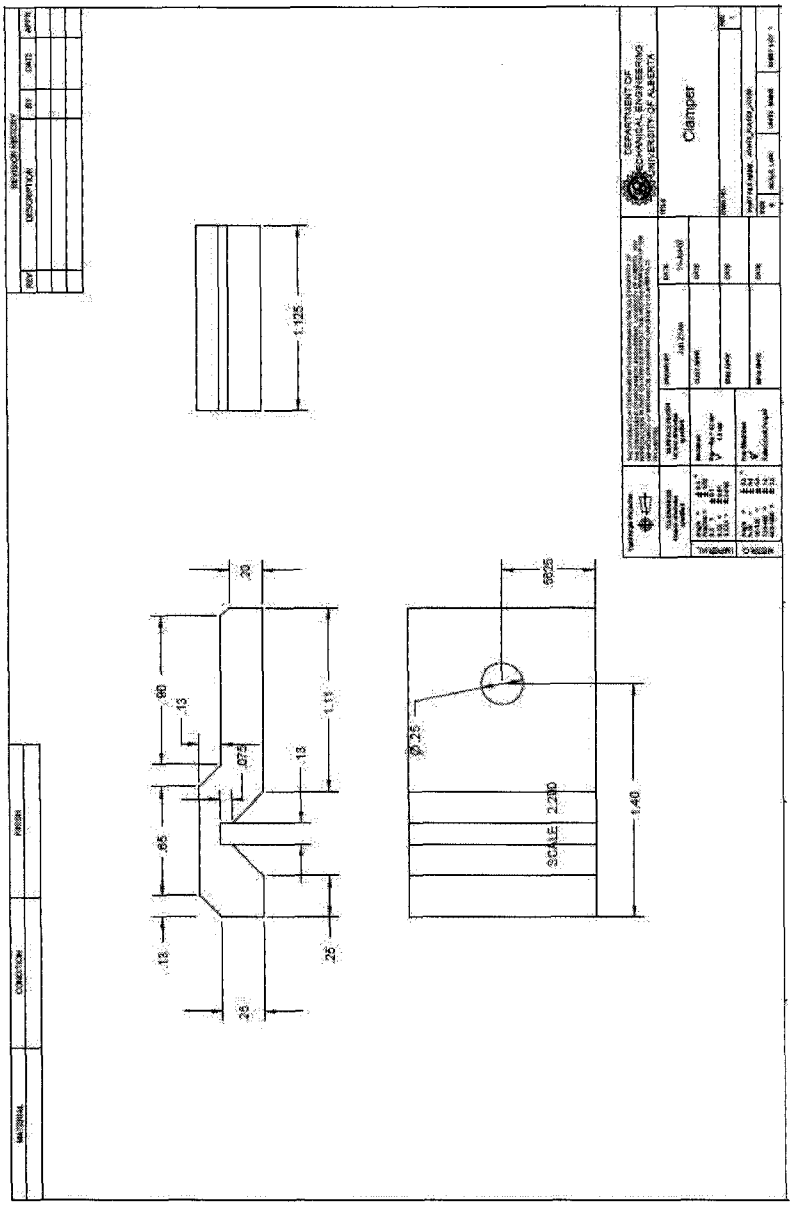

Figure A.40 Joint 6 Clamper