

Project Number: <ME-KZS-0602>

Design of a Roof Inspection Robot

A Major Qualifying Project Report

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

in Mechanical Engineering

by

Nicholas McMahon

Samuel Feller

Nathan Malatesta

April 26th, 2007

Approved:

Prof. Kenneth Stafford, Advisor

Acknowledgements

We would like to thank the following people for their help and support throughout the course of this project.

St. Paul Travelers

Jared Krechko

Kenneth Stafford

Brad Miller

Mike O'Donnell

Neal Whitehouse

Abstract

St Paul Travelers is an insurance company that performs over 35,000 roof inspections per year. The goal of this project was to design and build a robot inspection platform in order to limit risk to the human inspector and lower inspection time. The team developed an all-wheel drive robot capable of traversing a variety of roof geometries while visually recording data. The final deliverable also included an ascender system to deliver the robot to the roof. The robot will serve as a platform for future MQPs to further develop sensor systems for roof inspection.

Executive Summary

St. Paul Travelers conducted over 35,000 roof inspections last year, and with inspections lasting up to two hours and costing as much as \$1500 dollars each, they represent a significant expenditure of time and money. Furthermore, each inspection carries a degree of risk for the inspector, be it damaging the roof during the inspection process, falling off, or even falling through the roof.

Robots have long been used to explore and investigate places where it is too difficult or dangerous for a person to go. A roof inspection robot would face a number of challenges. There are many types of roof surfaces, ranging from clay tile, to slate, to metal, to composition. Some roofs are relatively flat, while others have pitches as steep as $\frac{24}{12}$. A robot able to replace a human in roof inspections needs to navigate those pitches, and be able to traverse the crests and valleys of the roof.

Project Task Specifications

This project's goals were developed in collaboration with St. Paul Travelers, reflecting both their needs and our limited timeframe. Because of the team's expertise in mechanical engineering, it was determined that the best use of time would be to develop a roof-ascending system as well as a remotely-controlled vehicle capable of performing visual inspections. Future projects could be developed to put together a sensing package for physical inspections and to further refine our designs. The formal task specifications for the inspection robot and ascender system are described below.

The robot must be able to traverse a roof, defined as:

- Maneuverability over composition tile with $\frac{12}{12}$ pitch
- Able to crest rooftop and valley
- Remote control (untethered)

The robot must have the following autonomous features:

- Can sense roof edge to prevent operator from driving over the edge
- Tilt warnings to prevent operator-induced flip-over

- Manual overrides to allow operability in the event of a false sensor warning

The robot must be able to conduct a visual inspection of the roof:

- Incorporates a camera with transmittable feed
- Is able to produce a record of visual inspection

The robot “prototype” must last through 50 hours of operation

The robot must be able to get on and off of a roof with a provided ascender mechanism; a ten foot (one story) prototype will serve as a proof of concept

Results

We met the project specifications through the design and fabrication of robot and ascender mechanism. The robot has a unique chassis design that allows it to traverse the peaks and valleys of a roof while avoiding any skidding that a tank style robot would be subject to.

The robot uses a Vex microcontroller to receive directions from the operator. Control algorithms take

sensor input from the wheels to provide all wheel drive and traction control. Additional sensors provide edge detection and a camera allows the operator to see the roof from the robots point of view.

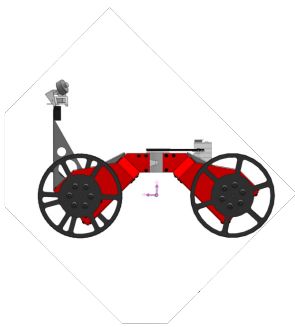


Figure 2: Robot CAD Model

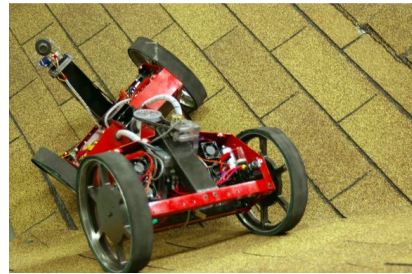


Figure 1: Robot Driving on Roof

The ascender mechanism transports the robot to the top of the roof without putting a person in danger. The final ascender design is an attachment for a standard extension ladder. The robot is placed inside of a carriage which is pulled up the ladder by means of a pulley. Once the carriage reaches the top it tilts to allow the robot to drive on to the roof.

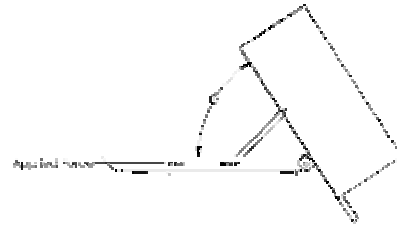


Figure 3: Ascender Carriage Operation

Conclusions and Recommendations

Through this Major Qualifying Project we have shown that it is possible to build a robot that can operate on a 45° roof with an asphalt shingle surface. We have concluded that it is feasible to use a robot to conduct roof inspections and we have further recommendations on how to continue to develop our work.

Conduct Further Research into Friction Materials

The major design challenges that we faced were centered around the frictional coefficient between the roof and the wheels. The robot is stable and balanced on a 45° slope, but its performance is traction limited. The best material combination we found was a EPDM foam over a Scotchbrite substrate, but there is not a large margin of safety. The robot occasionally loses traction and skids, but catches itself. We recommend future research into finding friction materials to help the robot navigate the roof without risk of slippage.



Figure 4: Robot Wheel on Roof

Continue to Develop and Optimize the Current Design

We have met nearly all of the original task specifications but were limited by time constraints. All the systems can be improved and optimized. Because all the team members working on this MQP were mechanical engineers, there is particular room for improvement in the electrical and control systems. The Vex microcontroller used on the

robot, for example, is a very easy to use system and serves as a proof of concept, but its lack of two way wireless data communication makes it a poor choice for a remotely operated vehicle. We recommend that future project teams include an electrical engineer and computer scientist to develop the systems on the robot that are outside the expertise of a mechanical engineer.

Develop a New, Roof Specific Sensor Package

One of the intentions of this project was to test whether a robot could even navigate the steep inclines and geometries of a roof. To fit within the time constraints of this project, the robot was only required to conduct a visual inspection. Having concluded that it is feasible for a robot to conduct a roof inspection, the next step is to develop a roof specific sensor package.

Our work can serve as a basis for future projects based on a roof inspection robot. We have laid the groundwork for the development of a sensor package, a more advanced operator interface, and an improved ascender mechanism through our work with this prototype.

Table of Contents

Acknowledgements.....	i
Abstract.....	ii
Executive Summary.....	iii
Introduction.....	1
1 Background.....	2
1.1 Commercially Available Robots.....	2
1.1.1 iRobot Packbot.....	2
1.1.2 Mesa Robotics, MATILDA.....	3
1.2 Lift Mechanisms.....	3
1.3 Robot Design.....	5
2 Methodology.....	6
2.1 Robot Task Specifications.....	6
2.2 Testing Procedure.....	7
2.3 Preliminary Design with Vex Kit.....	7
2.4 Chassis Design.....	9
2.5 Joint.....	13
2.6 Drive Components.....	17
2.7 Battery Selection.....	18
2.8 Wheels and Friction Materials.....	19
2.9 Articulated Chassis Kinematics.....	24
2.10 Microcontroller Selection.....	27
2.11 Overall Code Architecture.....	28
2.12 User Interface.....	28
2.13 Wheel Speed Algorithm.....	31
2.14 Feedback Algorithm.....	32
2.15 Sensor Package.....	33
2.16 Ascender System.....	34
3 Results.....	43

3.1	Navigation and Maneuverability.....	43
3.2	Autonomous Features	44
3.3	Visual Inspection	44
3.4	Durability	45
3.5	Ascender Mechanism.....	45
4	Recommendations.....	46
5	Conclusions.....	52
	Appendix A- CAD Drawings.....	53
	Appendix B- Robot Code.....	86
	Appendix C- Weight Breakdown.....	97
	Appendix D- Bill of Materials	98

Table of Figures

Figure 1: Robot Driving on Roof.....	iv
Figure 2: Robot CAD Model	iv
Figure 3: Ascender Carriage Operation	v
Figure 4: Robot Wheel on Roof.....	v
Figure 5 – Packbot Robot	2
Figure 6 – MATILDA robot	3
Figure 7 - Man Portable	4
Figure 8 - Trailer Towable.....	4
Figure 9 - Truck Mounted.....	4
Figure 10 - Peak and Valley Roof.....	7
Figure 11: Vex Robotics kit robots.....	7
Figure 12: First Prototype	7
Figure 13: Active stabilization.....	9
Figure 14: Roof Robot Final Design.....	9
Figure 15: Joint Rotation	9
Figure 16: Robot Cresting Peak	10
Figure 17: Center of Gravity Representation.....	10
Figure 18: CAD Assembly.....	11
Figure 19: Cracked Panel.....	12
Figure 20: Robot joint forces	13
Figure 21: Joint Housing Disassembled	14
Figure 22: Joint Housing Assembled.....	15
Figure 23: Joint front view 3.....	15
Figure 24: Joint front view 2.....	15
Figure 25: Joint front view 1.....	15
Figure 26: Joint vertical and horizontal pin assembly	15
Figure 27: Joint Assembled	16
Figure 28: Joint Disassembled.....	16
Figure 29: Torque vs Current Draw.....	18

Figure 30: Friction Test System.....	20
Figure 31: Static Friction Testing Results	21
Figure 32: Pressure Rig.....	22
Figure 33: Substrate Testing on EPDM Foam.....	23
Figure 34: Wheel Compression Under Load	24
Figure 35 - Geometry and Kinematics for Driving.....	25
Figure 36 - Geometry and Kinematics for Steering.....	26
Figure 37: User Interface Version One.....	29
Figure 38: User Interface Version 2.....	30
Figure 39: Laddervator, Power Ladder.....	34
Figure 40: Telescope Diagram.....	35
Figure 41: First Ascender Prototype CAD Model	35
Figure 42: First Ascender Prototype.....	35
Figure 43: Geo Data Systems Telescoping Pole.....	36
Figure 44: Ascender Flow Chart.....	37
Figure 45: Ascender Schematic Diagram	38
Figure 46: Pulley Free Body Diagram (Not To Scale)	39
Figure 47: Carriage Tilt vs Force Applied at Handle	40
Figure 48: Carriage Operation Schematic Diagram.....	41
Figure 49: Applied Force Free Body Diagram	41
Figure 50: Carriage Tilt vs Applied Force Equations.....	42

Introduction

St. Paul Travelers conducted over 35,000 roof inspections last year, and with inspections lasting up to two hours and costing as much as \$1500 dollars each, they represent a significant expenditure of time and money. Furthermore, each inspection carries a degree of risk for the inspector, be it damaging the roof during the inspection process, or falling off or even through the roof.

Robots have long been used to explore and investigate places where it is too difficult or dangerous for a person to go. A roof inspection robot would face a number of challenges. There are many types of roof surfaces, ranging from clay tile, to slate, to metal, to composition. Some roofs are relatively flat, while others have pitches as steep as $\frac{24}{12}$. A robot able to replace a human in roof inspections needs to navigate those pitches, and be able to traverse the crests and valleys of the roof. The robot needs a vision system so the inspector can drive the robot, and sensors to check the roof for non-visual damage. On top of that, the robot needs a deployment system to reach roofs which may be as high as forty-two feet.

The goal of this project is to deliver a robotic chassis capable of navigating a composition tile roof at a pitch of $\frac{12}{12}$. The robot will be able to cross both peaks and valleys, and will carry a simple camera for navigation and a visual inspection. The project will include a delivery mechanism capable of placing the robot on and off a roof from ground level.

1 Background

St. Paul Travelers insurance company has presented us with the task of developing of a robot for the purpose of inspecting household roofs. This is an operation that is normally carried out by a human inspector that must climb to the roof and then carry out a visual and physical inspection. The following section contains background research on commercially available robots, lifting mechanisms, and sensors.

1.1 Commercially Available Robots

The standards for the platform as specified by the sponsor of this project state that the robot must:

- Be maneuverable by someone on the ground
- Be driven on roofs of various surfaces
- Be driven on roofs with a pitch of up to 45 degrees
- Fit into the trunk of a minivan or back of a pickup truck

We have investigated several different commercially available robots regarding their use for roof inspection purposes. Remote-controlled all-terrain platforms meet most of the design specifications and flying machines have been ruled out by the sponsor. Of particular interest are remote-controlled platforms that have been contracted by the military as all-terrain vehicles used in applications where it would be dangerous to send a person. The following is a list of potential robots.

1.1.1 iRobot Packbot

The Packbot, shown in Figure 5, is designed for portability and survivability, both good qualities for a roof inspection robot. It is small enough to fit in the back of almost any consumer vehicle and can be lifted by one person. Its low center of gravity allows it to traverse slopes of up to 60 degrees



and its unique articulating tread system allows it to climb stairs and

Figure 5 – Packbot Robot

other obstacles. The Packbot can also survive a two meter drop; while this is not equivalent to a fall from a three-story building, it is an advantage. The Packbot is designed to have a payload. This payload could be optimized for roof inspections with a combination of sensors and actuators. iRobot can package the Packbot with a two-meter, remote-controlled, extendable arm. Operator interface for the Packbot is based around joystick controls and an LCD display of the robot's vision system, a simple effective way to control the robot for an insurance company agent. Packbots are sold for \$50k-\$115k, depending on the payload features supplied by iRobot. This is relatively cost-prohibitive to the sponsor. Additional development costs would be incurred for research and development of payload sensors and a roof deployment system.

1.1.2 Mesa Robotics, MATILDA

MATILDA (see Figure 6) is similar to the Packbot in that it can fit in the trunk of a car, be carried by one or two people, and is designed for rough use. It is controlled by a



Figure 6 – MATILDA robot

briefcase operator system similar to Packbot's. A low center of gravity allows MATILDA to drive on slopes of up to 55 degrees and which is suitable for the roof pitch requirement. It has a payload bay measuring 13.5x16.5 inches which would have to house all of the rooftop inspection sensors.

MATILDA has an approximate cost of \$55K, making it also too expensive for the sponsor's

budget.

1.2 Lift Mechanisms

In the event that there is no convenient level access point to the roof, it will be necessary to build or custom modify an existing device to place the robot on and off the roof. In this section, various commercially available roofing, personnel, and lighting lift mechanisms are discussed. Each mechanism is considered in terms of cost,

transportability, and effective lift height. Furthermore, the ease of operation and need for modification is also considered.



Figure 7 - Man Portable



Figure 8 - Trailer Towable



Figure 9 - Truck Mounted

Portable roofing lifts can be categorized by their means of transport. There are hand portable units, towed trailer units, and truck mounted units. (See Figure 7, Figure 8, and Figure 9, respectively.) Price and effective height go up as the units get larger. All the lifts are designed as general purpose lifts and a platform with a landing ramp might need to be built to carry the robot up.

Personnel lifts come in a variety of heights, again, usually varying with price. Most units are hand portable and can be carried by one or two people. They already have a wide, flat platform to carry the robot, and would need simple modifications to get the robot onto a roof.

Lighting lifts come as both hydraulic and hand-powered systems. Systems intended for indoor use are hand portable by one or two people. Larger outdoor systems are truck towable.

The sponsor's need for a compact and portable system is the ultimate limiting factor to commercially available lifting devices. Ideally they would like a telescoping device that can fit in the back of a car and deliver the robot to the roof. In our background research we found no such device.

1.3 Robot Design

The sponsor would like to have a complete robotic system capable of lifting independent shingles for damage inspection, using sensors for semi-autonomous functions, with an ascender system to move the robot to the roof. However, this is outside the scope of our three-man team in the allotted timeframe. Because of the team's expertise in mechanical engineering, it has been determined the best use of time will be to develop a roof-ascending system as well as the remotely-controlled vehicle that will do the inspections. A future project should be developed to put together a sensing package for physical inspections and to further refine our designs.

2 Methodology

2.1 Robot Task Specifications

In this section we state the task specifications for this project. To determine these task specifications we collaborated with our liaison from Travelers and our project advisors at WPI. Our goal was to set achievable specifications for our technical abilities and timeframe while leaving room for future projects to improve and expand on our designs. The resulting task specifications are as follows.

The robot must be able to traverse a roof:

- Maneuverability over composition tile with $\frac{12}{12}$ pitch
- Able to crest rooftop and valley
- Remote control (untethered)

The robot must have the following autonomous features:

- Can sense roof edge to prevent operator from driving over the edge
- Tilt warnings to prevent operator-induced flip-over
- Manual overrides to allow operability in the event of a false sensor warning

The robot must be able to conduct a visual inspection of the roof:

- Incorporates a camera with transmittable feed
- Is able to produce a record of visual inspection

The robot “prototype” must last through 50 hours of operation

The robot must be able to get on and off of a roof with a provided ascender mechanism

- A ten foot (one story) prototype will serve as a proof of concept

2.2 Testing Procedure

Before the finished robot will be designated as capable of doing rooftop inspections it will first have to qualify itself in a series of tests developed to determine its safety and reliability on simulated roof surfaces. The test fixture will be a simulated roof similar to that shown in Figure 10. It will consist of a peak and a corner valley covered in composition shingles at a pitch of 45 degrees. The robot must demonstrate that it is stable in all possible orientations and that the autonomous failsafe features prevent it from falling off.



Figure 10 - Peak and Valley Roof

2.3 Preliminary Design with Vex Kit

In order to test out concepts for chassis design we used the Vex robotics kit, seen in Figure 11 and Figure 12. It uses modular components that can be quickly assembled and disassembled to try out new ideas. The kit includes a six channel remote control and microcontroller. A system like this is very useful for characterizing the way different designs will behave on a simulated roof environment.

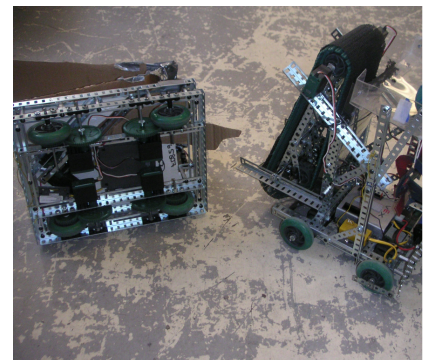


Figure 11: Vex Robotics kit robots

Our first prototype, seen in Figure 12, was made using the Vex kit. Due to the Vex kits ability to prototype rapidly several designs were tested to see how different joints, wheel bases, and centers of gravity would affect the final design. Since we ended up using the Vex kit controller to program our final robot it was beneficial that we gained programming experience with this controller early on in the project.

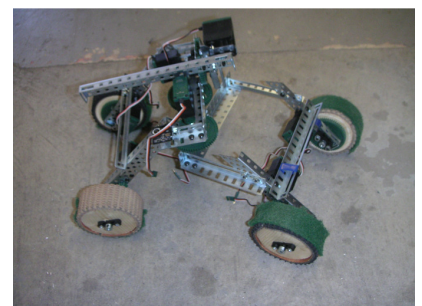


Figure 12: First Prototype

The initial designs showed that we needed a drive system that did not require loss of friction to turn, such as that of a tank

steering robot where the wheels must skid. It was found that when the wheels lose static friction it is very difficult to regain control of the robot and keep it from sliding on the roof. Thus we concluded that we would need a larger turning radius and a way for the wheels to move perpendicular to that radius

To work through this problem several joints were tested that articulated on two different axes. The purpose of a joint with two degrees of freedom was so that the robot could have four wheels in contact with the roof while cresting any valley or peak. To do this, one axis controlled the turning of the robot “yaw” and the other controlled the rotation between the front and the back of the robot or the “roll”. Roll rotation is used when the robot drives over a valley and must operate on two different planes. The joints tested compared the effects of using active or passive rotation in each configuration. An active joint is powered by a servo while a passive joint is not powered. The joint that we found to work the best is a configuration where there is active rotation in the turning “yaw” axis and passive in the “roll” axis. This allows the operator to control turning while not having to control roll. Two gearing combinations were experimented with to find a controllable rate of turn and it was found that about 30° per second would be appropriate. This joint design seemed to work well during this stage and was incorporated into the final design with some modifications discussed in section 2.5.

Another design problem solved while prototyping with the vex kit was how to crest the roof at the peak. Designs discussed were similar to that of iRobot’s Packbot (see section 1.1.1) where the joint would have active rotation in a third axis to essentially pull the robot over the peak and onto the other side. This was a benefit because it would allow our design to have a lower center of gravity when it returned to a flat surface after cresting the peak. The con of this system was the complexity. It was thought that by either keeping our center of gravity low or creating an active stabilization system we could eliminate this problem.

A prototype of the active stabilization system can be seen in Figure 13. To use this system the robot used an accelerometer to measure the angle of the roof and calculate where the counterweight of the robot would be above it to keep it stable. This system, while workable, was excessively complicated considering that a properly placed center of gravity would eliminate the need for this feature.

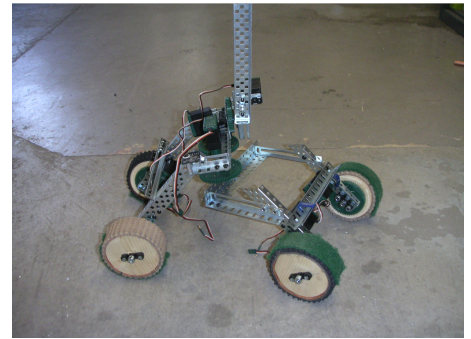


Figure 13: Active stabilization

One final consideration that was brought into the next design stage was that of wheel base selection. Since we knew the robot might need to crest over a vent and would need enough room to house additional sensor components it was found that a wheel base of at least twelve inches would be needed.

The ideas brought into the next phase of the robot were driving configuration, joint selection, concept of active stabilization, and a general concept of the final wheel base measurements.

2.4 Chassis Design

The design of the chassis was facilitated by two main factors; the shape of the roofs that the robot must navigate and the size of the payload. In order to traverse the unique peaks and valleys of roofs we decided to use an articulated design that uses a powered joint in the center of the chassis. The joint allows freedom of rotation in the yaw and roll as shown in Figure 15. This allows the robot to turn without using skid steering techniques and to operate on two unique planes as it crosses a 90° corner. The joint will be discussed further in section 2.5 as well as the unique control features associated with it in section 2.9. In order to cross the peak of a roof the underside of the chassis was left open so that

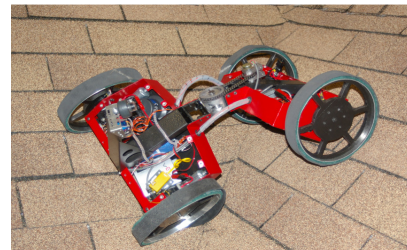


Figure 14: Roof Robot Final

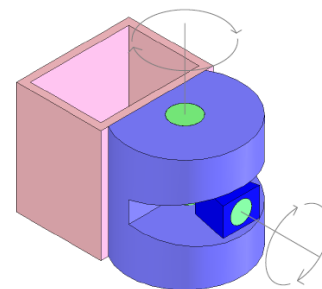


Figure 15: Joint Rotation

as the wheels cross from one side of the peak to the other there is clearance for the chassis to pass. (See Figure 16)

The size of the payload was determined by the components that were needed for the robot to operate. These components included; motors, batteries, microcontroller, wireless camera, speed controllers and various wires. The components were modeled in CAD and placed in various configurations as the chassis was designed. The placement of each component was chosen to ensure that a proper center of gravity (CG) was maintained in all operable conditions of the robot. To maintain stability a body's center of gravity must remain over its contact points with the ground, in this case the wheelbase of the robot. The greater the pitch the robot encounters the smaller the effective wheelbase becomes. The CAD package Solidworks allows for the calculation of CG by inputting the weights of each assembly item. By running a simple software analysis the CG of the assembly is calculated and outputted visually on the model as well as in XYZ coordinates.

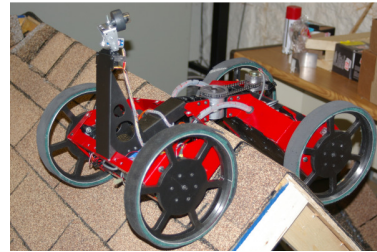


Figure 16: Robot Cresting Peak

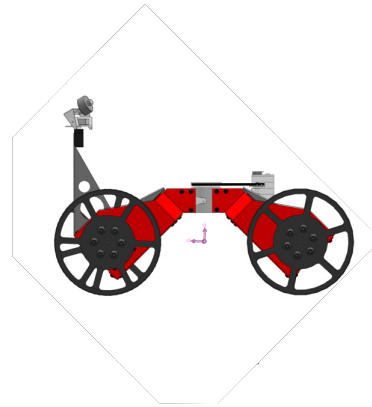


Figure 17: Center of Gravity Representation

The initial body design was prototyped using the Vex kit discussed in section 2.3. Working from this basic geometry and the CAD models of the components that were chosen we positioned each component as low to the wheels as practically possible keeping in mind that the heavier components such as batteries would have a more significant effect on CG placement. Once an acceptable configuration was determined we ran a mass properties analysis and the CG was displayed in context of the CAD model. We further analyzed the results to ensure that in all feasible body configurations (i.e. joint angles and rotations) the CG was acceptably placed within the wheel base. The calculated CG for the robot shows that it is able to maintain stability on slopes up to 60°. This is feasible although we were not able to find a friction material for the wheels that will provide the necessary coefficient of friction to maintain traction on these angles.

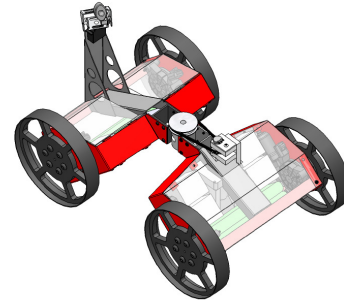


Figure 18: CAD Assembly

The CAD model also served as the source of dimensions and geometry of all the parts that needed to be made for the robot in the machine shop. A complete assembly and bill of materials can be found in Appendix A.

The materials used for the chassis side walls were 1/16 inch aluminum panels. The panels were cut using Haas CNC Vertical Machining Centers (VMC) and bent on a sheet metal break. The panels were first made of Aluminum, alloy 6061 T6. The 6061 alloy was chosen because of its high strength. Initial attempts to bend these panels resulted in cracking and breaking in several areas. After some research we determined that 5052 aluminum would be a better alloy for bending due to its greater percent elongation (12% for 6061 versus 25% for 5052¹). The resulting panels turned out much better after bending and showed very little stretch. The bottom of the chassis was made from 3/16" polycarbonate sheet. Polycarbonate was chosen for its low density and non

¹ <http://www.matweb.com/search/SpecificMaterial.asp?bassnum=MA6061AT6>

conductive properties. Conductivity was an important issue since all the electrical components of the robot were to be mounted to this surface.

The chassis design translated from CAD to physical prototype relatively well. With the aid of the CNC milling machines we were able to produce consistent and accurate parts. The materials have held up well to their application. Further results and suggested modifications to the chassis will be detailed in Section 4.

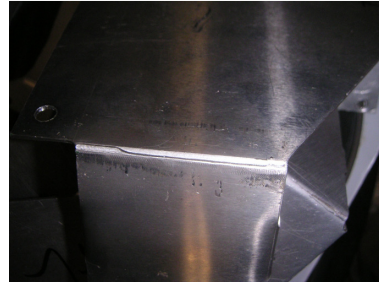


Figure 19: Cracked Panel

2.5 Joint

The joint that was developed during the prototyping phase of the Vex kit needed to be produced on a larger scale and needed to be more reliable for our final design. The Vex kit components only included plastic gears and small servo motors which would not withstand the torsion forces required to turn the full size robot.

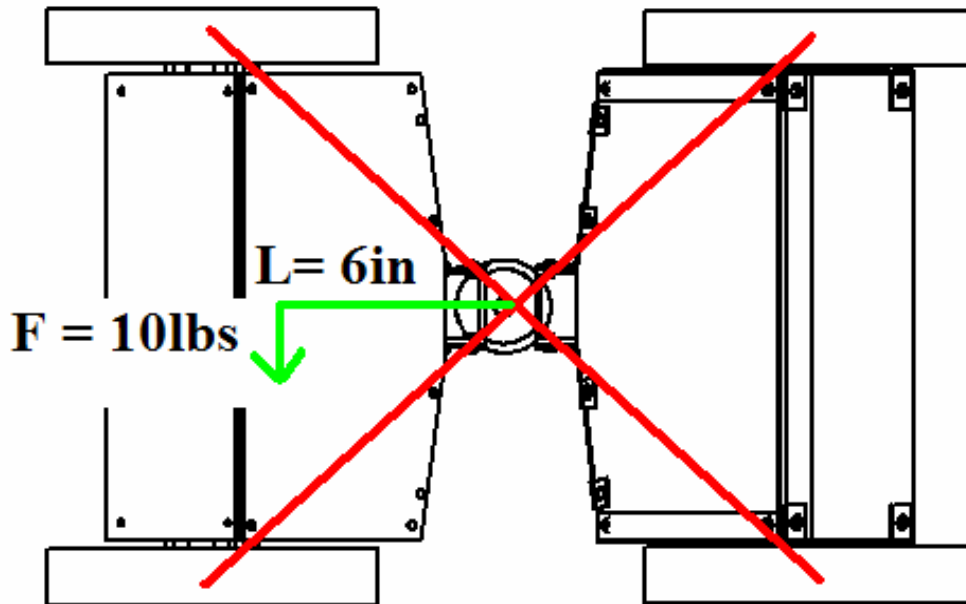


Figure 20: Robot joint forces

If the robot were to ever roll in an odd way there the wheels were not in contact to drive it a powered joint would be needed to point the wheels in the desired direction. We selected the power needed in the following way.

The center of gravity of half of the robot was approximately six inches from center. To lift half the weight of the robot (10 lbs) we would need 60inlbs of torque.

Our specifications were to find a motor that would allow the joint to turn at about six RPM so that we could make a 45 degree pivot in 1.25 seconds and would have enough torque to turn the joint in the worst case scenario.

We found a small DC motor from McMaster Carr that turned at 12 RPM and had a stall torque of 40 in lbs. We used a one stage reduction through a sprocket and capstan to give the joint a speed of six RPM and a torque of 80 in lbs giving us a safety factor of 1.33.

The problem with the Vex kit joint was that there were no limitations to the travel of either axis. The actively controlled “yaw” axis would allow the robot to turn into itself unless the operator realized his/her error and the passive “roll” axis had no limitation either allowing the robot to have one half upright and the other half inverted. To correct this from happening mechanical and software stops were incorporated into the design.

To mechanically stop the joint from turning too far, a housing was created (Figure 21) to allow only 45° of turn to the left and 45° to the right. This angle was chosen as a compromise between a tank style zero turning radius which requires a complete loss of friction, and driving in a straight line for which there is no loss of friction because there is no turn. Testing with the Vex kit also showed that the turning radius was small enough to navigate tight corners when inspecting a roof.



Figure 21: Joint Housing Disassembled

The more difficult part of the joint design was limiting the roll. However, we did not want to limit the roll to one angle of rotation for all yaw positions. The problem was that the robot required more roll articulation when driving straight and less when in a full 45° turn. If the joint was given more rotational allowance in a 45° turn than necessary, we found that the robot would collapse on itself and one section of the robot would become inverted.

To solve this problem we designed a joint similar to that of a skid steer logger. In this design the joint allows for full rotation while driving straight and limits the rotation as a linear function down to zero in a full 45 degree turn.

To do this a custom joint was fabricated with these specifications in mind and sized to fit the proposed chassis design. It was required that it be strong enough to withstand the bending moment forces of a 20 pound robot.

To manufacture this joint the aforementioned housing was modeled in CAD and then built out of two aluminum pieces as seen in Figure 21 and Figure 22. The angled cut was matched to that of a toggle piece seen in Figure 25, Figure 23, and Figure 24 This piece was welded onto a shaft that holds the bending moment of the chassis and is the piece that connects the two halves.



Figure 22: Joint Housing Assembled

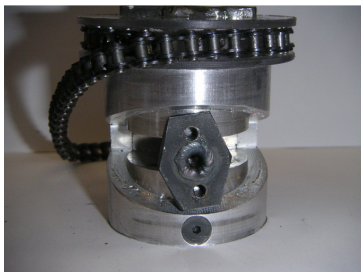


Figure 25: Joint front view 1



Figure 24: Joint front view 2

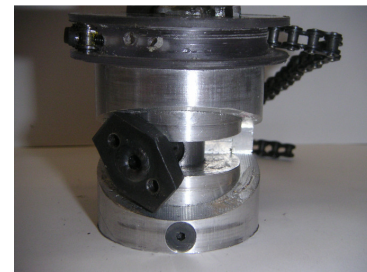


Figure 23: Joint front view 3

This piece was then inserted into a vertical shaft seen in Figure 26, which allowed for the active turning rotation.

This shaft was integrated into a capstan seen on the top of Figure 27 which ran 1/4" chain to the drive motor which powers the active portion of the joint.



Figure 26: Joint vertical and horizontal pin assembly

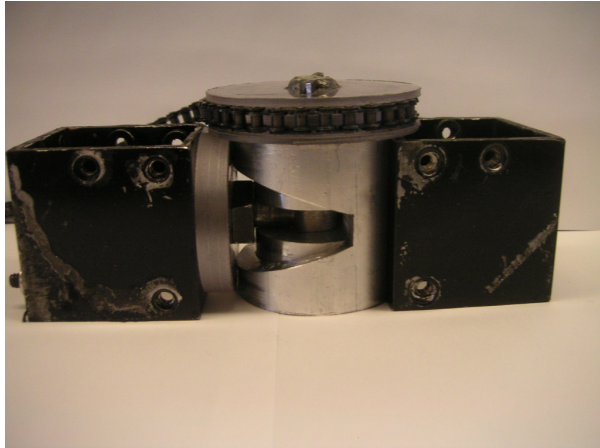


Figure 27: Joint Assembled

A full disassembly can be seen in Figure 28.



Figure 28: Joint Disassembled

2.6 Drive Components

The choice of drive components became the starting point of our design. Drive motor selection ultimately dictated the size of our chassis and also accounts for approximately one fourth of the robots weight.

Our drive system needed to meet the following specifications:

- Non back-drivable
- Driving speed of approximately three feet per second
- One motor per wheel
- Drive time of at least one half hour

To have a non back-driving system we had two options. Design a ratcheting system into our driveline to prevent back-drive or use worm gear transmissions which are naturally back-drive resistant. Worm gear drive motors are commercially available prepackaged making them very attractive to our application.

From experience with other robotic projects we thought that a speed of approximately three feet per second would be appropriate for driving on the roof. This would allow the robot to scan over 10,000 square feet assuming a field of view of two feet. The speed calculation is derived from the output speed of the motor and the circumference of the chosen wheel. The motors we chose for the drive line are Nippon-Denso window motor that are used in many General Motors vehicles. These motors were chosen because they are readily available and had a low max output speed of 85 RPM and a stall torque of 106 inch lbs. Using these motors in conjunction with an 8.2” wheel diameter we were able to achieve our desired speed of three feet per second.

As we moved further into the design of the robot chassis it was discovered that a wheel diameter of nine inches would be needed for our robot to crest the peak of the roof, giving it a driving speed of 3.33 feet per second. This was deemed to be a speed which the operator should still be able to control.

$$\text{diameter} = 9''$$

$$\text{Speed} = \text{diameter} * \pi * 85\text{rpm} / 60\text{sec} = 3.33 \text{ ft/sec}$$

With a 4.5” arm on each wheel and our estimated 20 pound robot we would be in need of 64 inch pounds of torque assuming the robot is climbing a 45 degree angle slope. Therefore assuming we use half the stall torque because we are moving and we are climbing a 45 degree slope we will still have a safety factor of 3.3.

2.7 Battery Selection

This system needed to last at least one half hour at full draw from the drive motors to be considered acceptable. A graph of the current draw of the drive motors can be seen below in Figure 29.

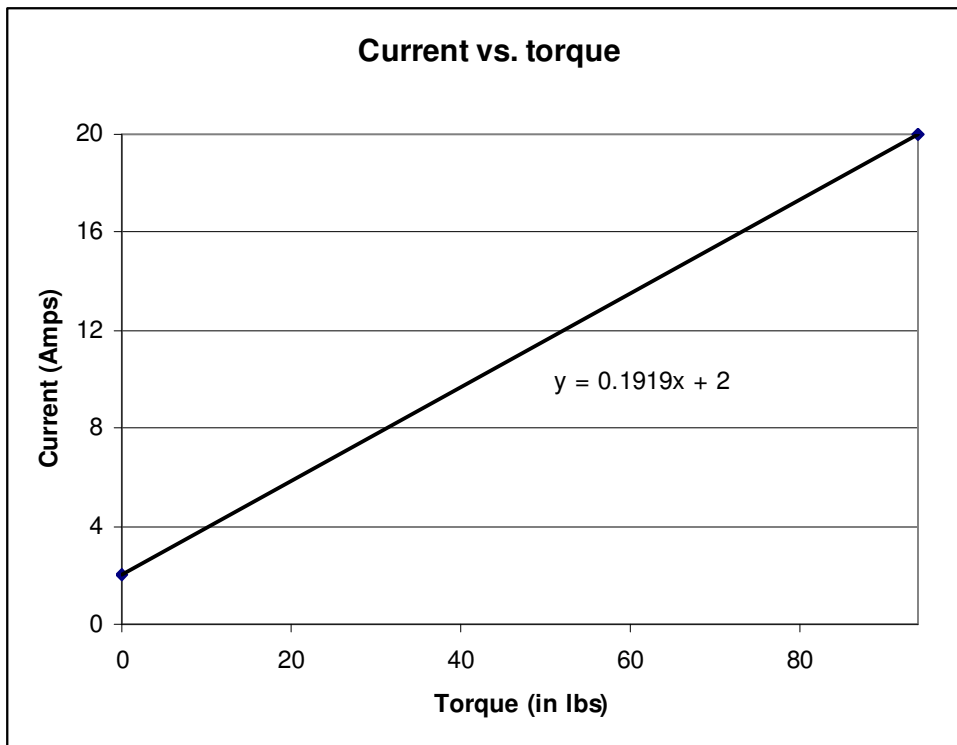


Figure 29: Torque vs Current Draw

The motor draw was calculated for each motor giving equal power while traveling up a 45° slope.

$$M_{robot} = weight = 20lbs$$

$$R = radius = 4.5"$$

$$\beta = slope = 45^\circ$$

$$\sum Torque = M_{robot} * \sin(\beta) * R$$

$$\sum Torque = 64inlbs$$

When driving up a 45 degree slope there will be approximately 32 inlbs of torque on the rear two wheels.

$$Current = (.1919 * Torque + 2) * 2 \quad \text{We multiply by two to account for each wheel}$$

This is given from the graph in Figure 29.

$$Current = (.1919 * 32 + 2) * 2 = 16.28 \text{ Amps}$$

(The batteries are rated for a peak draw of 36Amps and a recommended draw of 12Amps.)

Therefore to drive at this peak draw for ½ hour we would need a source which can supply at least 8.14 Amp hours.

We used a safety factor of 1.5 and bought three 12 volt batteries supplying 4.2 Amp hours each totaling 12.6 Amp hours, giving us a total driving time of 45 min at max draw.

2.8 Wheels and Friction Materials

Maintaining static friction on a 45° slope requires that the coefficient of friction be at least 1. In order to find a suitable material to interface with the roof we had to test many different materials on asphalt shingles. To do this friction testing we set up a measuring rig using a force gauge, data logging system and multiple weights. Each material was fastened to the underside of a block of known mass. This block was then fixed to the end of a force gauge using a length of line. A pulley was used to ensure that the force was directed in the same direction at all times.

The block was loaded with weights and the force required to move the block was determined through a data acquisition system reading the force gauge. The coefficient of friction was determined by dividing the mass on the block by the force it took to move the block. The test was repeated with weights measuring 5, 10, 15 and 20 Newtons. All materials with a coefficient of friction less than one were eliminated. The top four materials were selected for further testing.

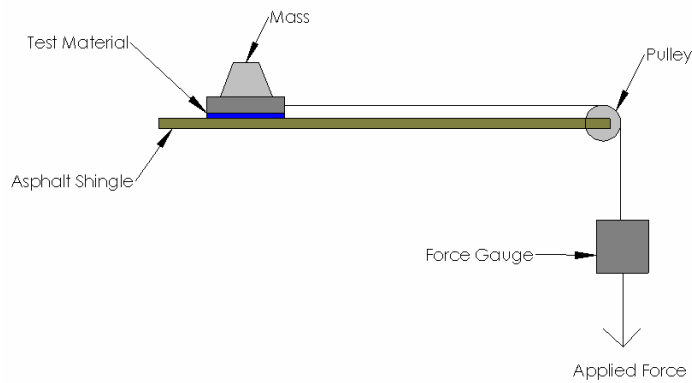


Figure 30: Friction Test System

The results for the top four of the thirteen tested materials can be seen in the chart below, Figure 31. EPDM foam was the clear winner and we decided to proceed in using it as the friction material on the robot's wheels.

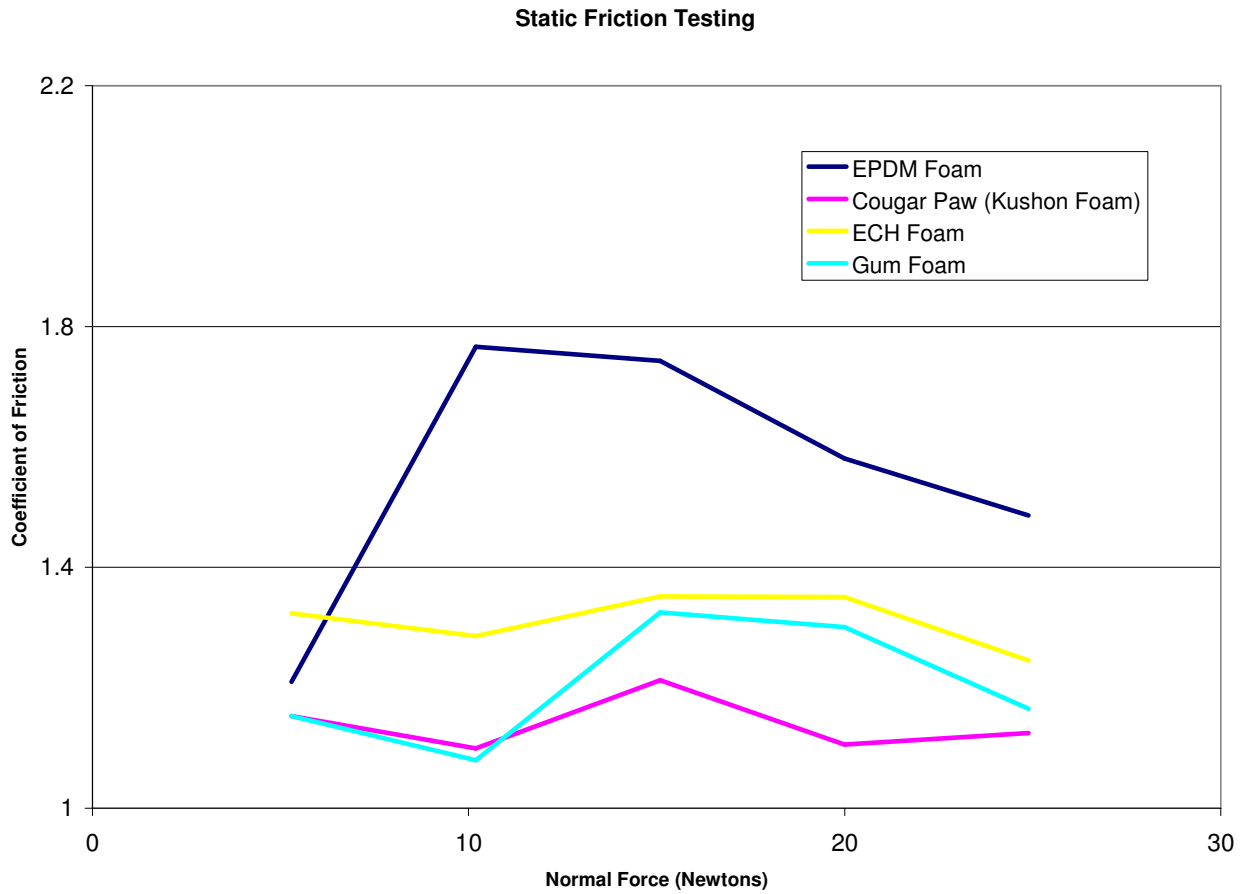


Figure 31: Static Friction Testing Results

The equation relating friction to downward force is expressed as:

$$F_{friction} = \mu N$$

$$\mu = FrictionCoefficient .$$

$$N = NormalForce$$

This equation implies that there is no relation between friction and surface area. However, we observed a phenomenon when using these materials on the roof that shows this is not the case on asphalt shingles. Because an asphalt shingle is composed of grit adhered to a tarpaper backing, it is natural that with the correct amount of force the grit will come off of its backing. We observed this happening as the robot drove on the shingles. The theory is that when the normal force per unit area becomes too high the grit lets go of the backing, thereby inducing slip between the wheel and the shingle. There is evidence of this in our static friction testing graph which shows that once the applied weight becomes too great the frictional coefficient goes down. To counteract this problem we decided to try decreasing the pressure per unit area of the wheel on the roof. In a pneumatic tire this would be accomplished by letting air out. Since we were not using pneumatic tires we experimented with different substrates between the hard rim and the friction material of the wheel. The idea is that the substrate will compress increasing the contact patch between the friction material and the roof lowering the pressure per unit area. The substrates we experimented with included; open cell sponge, Scotchbrite, and insulating foam. To test this setup a rig was constructed as pictured in Figure 32. This rig was used on the same fixture as described for the initial friction testing.

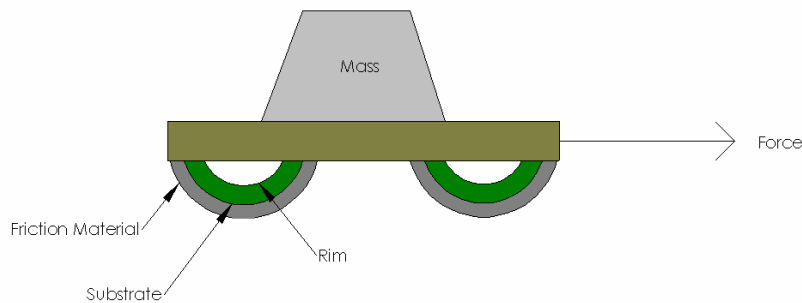


Figure 32: Pressure Rig

We found that using a substrate with the selected materials caused the coefficient of friction to go up. However, it was inconclusive as to which substrate worked better than another. The results of this testing can be found in Figure 33. We decided to use Scotchbrite as the substrate because it scored well in the testing but also for its shear resistance. Scotchbrite, unlike the rest of the substrates tested resists shear loads while maintaining low resistance to compressibility. The ability to resist these shear loads was important because when the robot is driving parallel to the peak of the roof there is a high axial load on the substrate which, under the right circumstances may fold and cause the robot to lose traction.

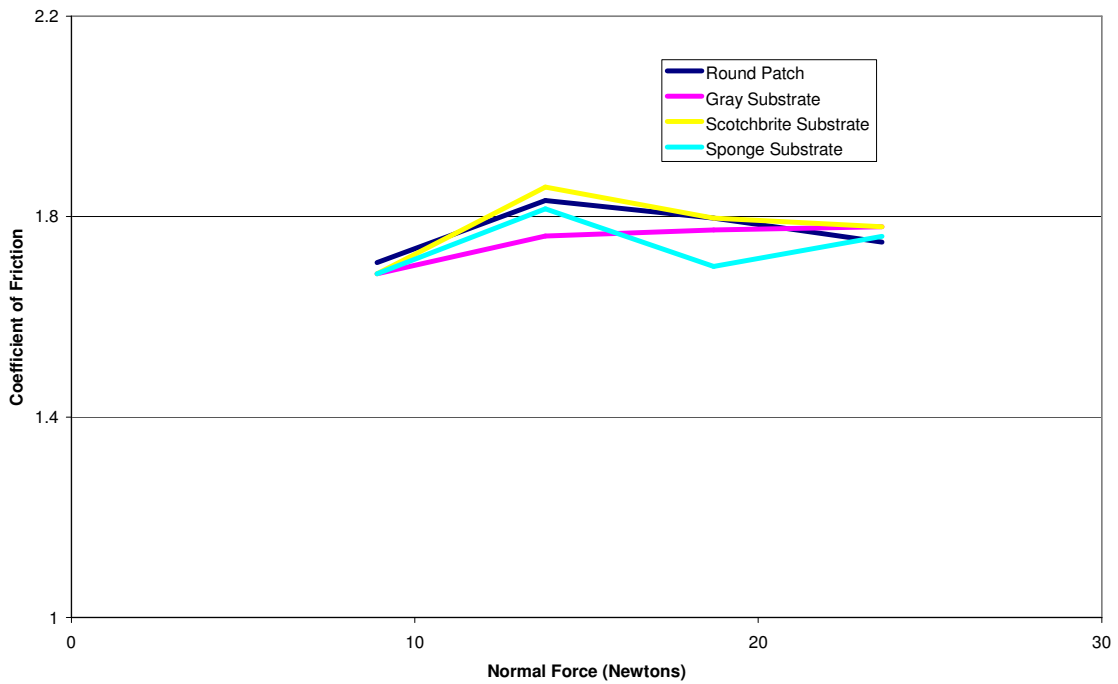


Figure 33: Substrate Testing on EPDM Foam

The use of EPDM with a Scotchbrite substrate allows the robot to maintain static stability on 45° slopes. Figure 34 shows how the substrate compresses under the load of the robot increasing the contact patch of the wheel. The robot is also able to drive on 45° slopes. However if a slip is induced it is possible for the robot to lose its adherence to the roof and continue to



Figure 34: Wheel Compression Under Load

slide. To reduce the possibility of induced slip an electronically controlled all wheel drive system was developed to control the speeds and power of each wheel.

2.9 Articulated Chassis Kinematics

The articulated chassis was designed with the intent of maintaining the wheels in rolling contact with the roof surface at all times. A design choice was made early on to have four independent motors powering the wheels, rather than a single motor and a set of mechanical differentials. Therefore, it was necessary to describe the kinematics of the articulated chassis so that electronic means of controlling the wheel speed could be implemented using a microcontroller later on. The kinematics for the two major navigation motions, driving and steering, are described below.

The wheel speeds during driving are a function of the angle between the two halves of the chassis. As the robot articulates and drives through a turn, the wheels travel along two different radii of curvature. The inner wheels have a shorter distance to travel than the outer wheels and must travel slower to remain in contact. As the chassis straightens, the radius of curvature of the turn approaches infinity, and the inner and outer wheels drive closer and closer to the same speed.

Figure 35 below shows the relevant geometry, where

θ is the measurement of the relative angle between the chassis halves

l is half the length of the wheel base

w is half the width of the wheel base

C is the center of rotation for driving

d is the distance from the centerline of the chassis to the center of rotation C

Lines d and l make up two sides of a perpendicular triangle, with angle $\theta/2$. The

unknown distance d to the center of rotation C can be calculated using the known constant l .

$$d = l / \tan (\theta / 2)$$

Given a driving velocity V , the velocities V_{inner} and V_{outer} can be calculated using the equations:

$$V_{inner} = V * (d - w) / d$$

$$V_{outer} = V * (d + w) / d$$

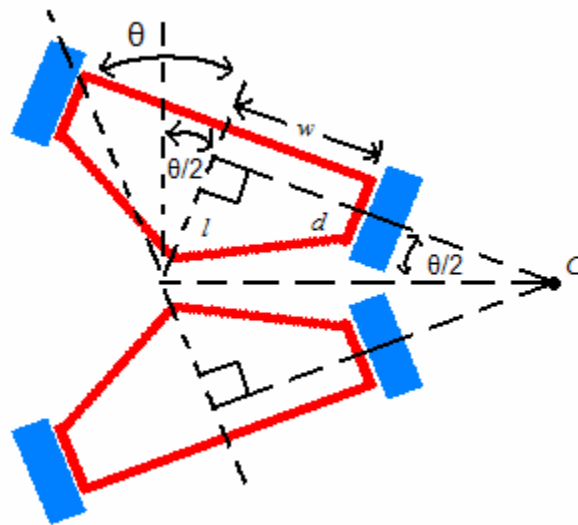


Figure 35 - Geometry and Kinematics for Driving

As the chassis articulates, one pair of wheels moves closer to each other, while the other pair moves farther apart. To keep the wheels in pure rolling, a center of rotation is calculated that lies exactly between the wheels. This location is also a function of the relative angle of the two chassis halves.

Figure 36 below shows the relevant geometry, where

θ , l , w , are the same as above

C is the center of rotation for steering

r is the distance from the centerline of the chassis to the center of rotation

Lines r and l make up two sides of a perpendicular triangle, with angle $\theta/2$. Again, the unknown distance r to the center of rotation C can be calculated using the known constant l .

$$r = \tan(\theta/2) * l$$

As before, if given a steering velocity V , the velocities V_{inner} and V_{outer} can be calculated using the equations below. The one addition being that the wheels on the opposite halves of the chassis spin in the opposite direction and the equation describing their speed is simply multiplied by -1.

$$V_{inner} = V * (w + r) / w$$

$$V_{outer} = -V * (w - r) / w$$

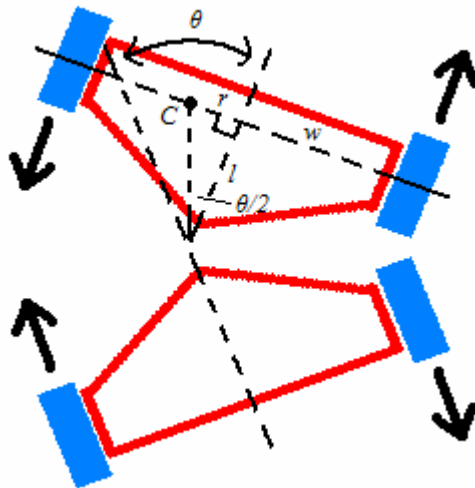


Figure 36 - Geometry and Kinematics for Steering

2.10 Microcontroller Selection

The Qwerk microcontroller, produced by Charmed Labs, was our first choice for use as a robot controller. Its chief advantage was that it had support for a web cam and a wireless internet adapter, so that we could drive the robot over a wireless internet connection and get integrated video feedback. The robot was also very adaptable for future upgrades, with a wide range of inputs and outputs including 16 PWM outputs, 4 motor outputs with integrated current sensing feedback, support for 4 quadrature encoders, 16 digital inputs and outputs, and 12 analog inputs with 12 bit resolution. The Qwerk was intended to act as a webserver, and could be accessed over the internet using a Java-based client. An out of the box solution had support for a tank style robot with video navigation. Unfortunately, our lack of java programming ability and the lack of documentation and poor support (the Qwerk was just out of beta testing), made it impossible for us to modify the client for use with an articulated robot.

After ruling out the Qwerk, we switched to the Vex microcontroller. Although the Vex did not have video support or the wide range of inputs that the Qwerk did, it had enough PWM outputs to drive the motors and enough inputs to gather data from the sensors we incorporated into our design. The Vex also has a well documented, easy to use API, which is written in C, a programming language we are familiar with. It was also possible to add video on a wireless home security system, which is completely independent of the Vex microcontroller.

2.11 Overall Code Architecture

The overall architecture for the code was fairly simple, as seen below. A continuously running loop cycles through and polls the inputs from the radio controller, collects feedback from the sensors, and updates the motors appropriately.

```
while (true)
{
    getInput();           //gets joystick positions from the radio controller

    getFeedback();       //gets feedback from
                        //          chassis articulation angle
                        //          wheel encoders
                        //          timers
                        //          IR sensors

    calculateOutputs();  //adjusts outputs appropriately based on input and
                        //feedback

    driveMotors();       //sends signal to the motors
}
```

2.12 User Interface

The user interface makes use of the six channel radio controller, supplied with the Vex microcontroller. The first four channels are defined by two joysticks on the front of the controller and the remaining two are controlled by buttons on the back.

In the first iteration, the left joystick was used to pan and tilt the camera and the right joystick was used for navigation. It was impossible to drive and turn at the same time and the operator would have to pause, adjust course, and continue driving. At best, it was difficult and at worst it caused violent shaking in the robot as the joystick moved in and out of its dead zone.

The camera was located above the joint at the highest point on the robot. The operator had no sense of proportion and because the camera pan/tilt platform didn't auto-center, the operator had no sense of which direction the robot was going.

Channel	Mode	Description
Navigation Forward/Reverse	Linear Proportional Velocity Control	The joystick is linearly correlated with the centerline speed of the robot. During turns, the outer wheel might spin twice as fast as the centerline, so maximum power is cut in half to avoid clipping.
Navigation Left/Right	Non Linear Velocity Control	The axis pans the robot through turns at a constant speed if the joystick is out of the deadzone.
Camera Up/Down	Non Linear Velocity Control	Tilts the camera.
Camera Left/Right	Non Linear Velocity Control	Pans the camera
Camera Location	N/A	Camera is located over the joint at the highest point on the robot.

Figure 37: User Interface Version One

After driving the robot and gaining experience as an operator, we made several changes to improve the interface. First, the forward/reverse navigation axis was changed so that the amount of power available would scale depending on the angle of articulation of the chassis, i.e. during a full turn, the maximum joystick position would correspond to half-speed along the centerline, so the outer wheels could spin at the correct speed without maxing out power. While driving straight, however, the joystick controlled the full range of power.

Second, the navigation left/right channel was changed so that it controlled the angle of the chassis articulation, much like a steering wheel on a car controls the angle of the wheels. This gave the operator a much better sense of where the robot was, as a joystick slightly to the left would mean the robot chassis was angled slightly to the left, rather than meaning the robot was simply turning more and more left. Additionally, the code was revised making it possible to drive and turn at the same time.

Third, the camera location was changed to sit on a mast on the back half of the robot. The operator now had a third person view of the robot, and much better sense of where it was going. The operator could use the camera to look down at the robot and see if it were stuck, if it were articulated appropriately, or if the wheels were spinning like they should.

Fourth, the camera pan/tilt platform was improved to include an auto-centering feature. In addition, as the robot turned, the camera would turn in that direction as well. By keeping a point of interest centered in the field of view, the operator could easily drive towards it. The left/right of the camera control was changed to an adaptive, position control system, so if the robot, and therefore the camera, were pointed slightly left, pushing the joystick right could still allow you to look all the way to the right. Pushing the joystick all the way to the left would allow you to look all the way to the left without going to far. Letting go of the joystick and centering it would cause the camera to snap back to its original position

Finally, a sort of “artificial horizon” in the form of a simple vertical pole was added to the robot to aid the operator in navigation. The pole was placed at the joint, directly on the centerline of the robot. The operator can see the pole in the field of view and get a much better sense of which way is straight forward.

Channel	Mode	Description
Navigation Forward/Reverse	Adaptive, Linear Proportional Velocity Control	Joystick has linear control over power. The available power scales so that the outer wheels never spin too fast during a turn, but the robot can still go full speed while driving straight forward.
Navigation Left/Right	Position Control	The axis pans the robot through turns at a constant speed if the joystick is out of the deadzone.
Camera Up/Down	Non Linear Velocity Control	Tilts the camera.
Camera Left/Right	Adaptive, Proportional Control	Camera looks in the direction the robot is going. Joystick looks left and right, but snaps back to center
Camera Location	N/A	Camera is located on a mast on the back of the robot

Figure 38: User Interface Version 2

2.13 Wheel Speed Algorithm

One of the biggest problems driving the robot was caused by the Victor Speed Controllers, which take a low power PWM signal to control the high power current loads to the motors. The Victors have a built in deadband, which cause the motors to freeze when they should be moving. In the worst case scenario, the Victors are not calibrated to the same settings and one wheel will spin while the other is stuck, causing a loss of traction.

The original algorithm to calculate wheel speed used the kinematic equations described above. Each wheel speed was calculated relative to the centerline velocity. The centerline velocity was calculated from the forward/reverse axis of the controller. There were two problems with this algorithm. The first was the problem with the deadband , as described above. The second was that during a full turn, the outer wheels of the robot spin nearly twice as fast as the inner wheels. To avoid maxing out the motors, the top speed corresponding to the full forward and reverse joystick positions was simply cut in half. This limited the robot's top speed while driving straight forward as well.

The solution to both of these problems was to calculate the wheel speeds relative to each other, rather than to the centerline of the robot. Specifically, the wheel speed of the right half of the robot was compared to the left half. There is a deadband on the joystick, and it was set so that if the robot was supposed to be moving, whichever side was slower, right or left, would receive a signal to make sure it was out of the Victors' deadband. The wheel speed for the faster side was then calculated as a ratio of the slow wheel speed. To avoid maxing out the motors, without giving up top speed, the forward/reverse joystick channel was scaled as a function of chassis angle, so that while driving straight forward, all the motors could run at top speed and during a turn, only the outer wheels could run at top speed.

2.14 Feedback Algorithm

The feedback algorithms were implemented in response to traction control problems that arose during testing. The outputs of the wheel speed algorithms were sent to directly to the motors as a “power level” signal, meaning that if the wheel was meant to go at 20% of its top speed, it received 20% power. On flat ground, with an equal load on all the motors, this worked very well. Driving up a 45° slope, the robot’s weight shifts to the back wheels. At 20% power, the front wheels would spin in place, while the back wheels would be stalled.

Optical encoders were placed on all the wheels so that control loops could be implemented to control the actual speed of the wheels, rather than power levels. Because the Victor speed controllers had a deadband and because the motors had worm gears and did not back drive while under load on the roof, it was possible to use simple, single channel optical encoders and keep track of the wheel direction in the computer code, rather than use quadrature encoders.

Because of limited resolution of the Vex output signals and the backlash in the drive train, programming a control loop that could respond quickly to errors in wheel speed without overshooting and creating oscillations was extremely difficult.

The solution was to use two Proportional inputs simultaneously. Each wheel’s speed was calculated independently and used as part of the feedback loop. This was good enough to work under most driving conditions, but when the robot was starting from a dead stop on a 45° slope, the back wheels did not respond quickly enough to avoid significant loss of traction in the front wheels. The second proportional input to the control loop was the speed of both wheels on each half of the robot. If a wheel was traveling too slowly and it was traveling slower than its respective wheel on the other half of the chassis, it got an extra speed boost.

2.15 Sensor Package

In order to make the robot safer for the operator to use, autonomous edge detection and lockout was added. Four Sharp IR sensors were added, pointing slightly outwards at all four corners of the robot. The IR sensors have a transmitter and receiver and look for the reflection of the IR beam. The sensor is very robust and tolerant to noise because it uses a lens and a strip of photocells to triangulate the distance to the target, rather than measuring the amount of reflected light.

In operation, the computer code polls the IR sensors during each loop of the main code. If the sensors return a low enough value, indicating that it is looking into open space beyond the edge of the roof, the robot is immediately halted and the operator is locked out from the controls. A manual override button may be pressed for the operator to resume control of the robot.

2.16 Ascender System

The ascender was viewed as a system independent of the robot. Initial technical specifications for the ascender stated that it must: raise the robot to a three story roof, be transportable by one to two people, and fit inside of a car. These specifications changed as the design of the ascender progressed. This section will discuss initial design concepts and the progression of the design to its current working state.

The first step in designing the ascender was to research current means of lifting loads to a roof. There are a number of different cranes and elevator like devices on the market for such purposes. These devices are used primarily by the construction industry for moving materials such as shingles to the top of a roof. The commercially available products we identified were impractical for delivering a twenty pound robot to the desired height of three stories. A typical ladder lift for instance consists of individual eight foot sections which must be bolted together. The assembled ladder must then be raised to the rooftop. At this point the payload is placed on a platform which rides on the ladders rails and a motor, gas or electric, is used to drag the platform up the ladder. This at first seemed like a perfect solution to lifting the robot. In a system like this the ladder must be assembled to full length on the ground and then positioned against the side of the building. We decided that positioning a system like this was undesirable because of the large moments involved in lifting a thirty foot ladder to vertical from horizontal. Furthermore the commercially available systems are designed to carry a maximum weight of at least 250 pounds which is significantly more than the twenty pounds of the robot.



Figure 39:
Laddervator,
Power Ladder

We decided against using any form of ladder that must be assembled to length on the ground. This meant that the ascender must be some form of telescoping mechanism. The most recognizable form of a telescoping mechanism is an extension ladder. By nature, a telescope device must have profiles that nest within each other. An actuator, usually a cable on a pulley, then pulls one profile along the other until the device has extended to its full height. We also had to design a carriage which would hold the robot as it was brought up telescope. This carriage needed to have some form of linear bearing on it to keep it on the telescope. The inherent problem with telescoping devices is that they do not provide a constant profile for a linear bearing to follow. This means that in order to have a carriage follow a telescope device it must have loose tolerances or have an adaptable mechanism capable of

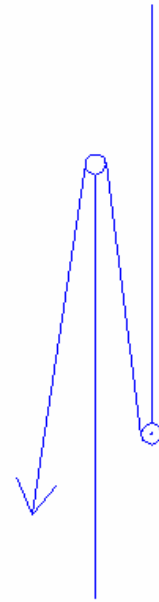


Figure 40:
Telescope
Diagram

following the profiles of the extension.

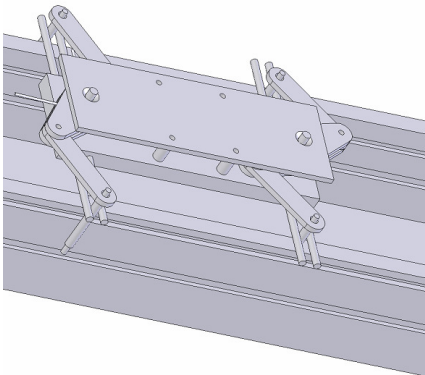


Figure 41: First Ascender Prototype
CAD Model

The first device that we prototyped was a four stage telescoping extension with a spring loaded four-bar linkage used to follow the telescope profiles. The design was modeled in CAD (Figure 41) and a prototype was fabricated from wood and metal to test the concept (Figure 42). Because of tolerance issues with the wood the carriage never

properly mated with profiles.

This was a significant problem because as we looked forward to a full size prototype we realized that constructing the necessary profiles would not be feasible since we did not have the manufacturing facilities

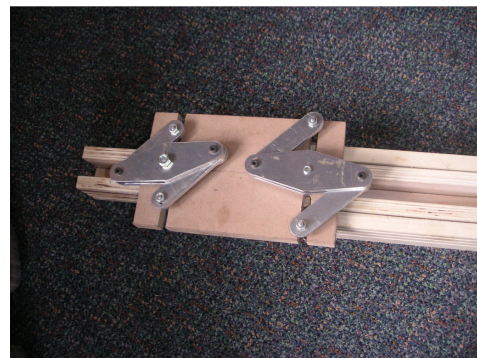


Figure 42: First Ascender Prototype

necessary nor would it be economical to outsource the design. This concept was a reality check that showed us even though something works great in CAD it will not necessarily translate to the real world.

We contacted Travelers again to go over the technical specifications. We found that the inspectors currently use ladders that bolt together from four foot sections to a height of sixteen feet and when access to a taller roof is needed a twenty-eight foot fiberglass extension ladder is used. When this was determined we proposed the idea of using a sled that rides in the existing extension ladder to transport the robot to the roof. Our liaisons at Travelers expressed that while this was not ideal it would be sufficient. We continued to press on looking for a means of creating our own telescoping device. The use of a commercially available telescoping pole was explored but when the manufacturing company (Geo Data Systems) provided us with the pole's specifications we found the deflection with the weight of the robot to be excessive. (A thirty foot fiberglass pole available from Geo Data Systems will deflect fourteen feet with a fifteen pound load on the end).² Because time was running out we decided to go back to the design involving a sled on a commercially available extension ladder.

An extension ladder is essentially a prepackaged two stage telescoping platform. We purchased a twenty foot fiberglass extension ladder made by Werner, the preferred brand of Travelers. The twenty foot length was selected because moving and storing a twenty-eight foot ladder was not practical for us and upon inspection we found that all Werner fiberglass ladders have the same profile dimensions, the only thing that varies is the length. This meant that we could design a sled system for the twenty foot ladder that could then be bolted onto a twenty-eight foot ladder later. The sled was modeled in CAD and a prototype built from lexan for testing. The design uses a sled to carry the robot to the top of the ladder. The robot sits on a platform enclosed in a lexan box preventing the robot from falling out. The platform pivots about the end of the sled.



Figure 43:
Geo Data
Systems
Telescoping
Pole

² <http://www.geodatasys.com/pole3.htm>

As the sled approaches the end of the ladder the tilting platform engages with an angled profile which guides the tilt of the platform as it rotates at the end of the ladder. The roof itself is used as a positive stop. The angled profile also guides the platform back into its original position for lowering the robot back to the ground. The entire assembly is operated by a hand operated drum at chest height fixed to the ladder.

Once this prototype was made it was apparent that there were certain situations where the stability of the sled on the ladder became precarious. To remedy this, the addition of an 8020 extruded profile was made. The extruded profile is fastened to the upper portion of the extension ladder. Linear bearings were secured to the sled which interfaces with the 8020 extrusion; this locks the sled to the ladder, preventing it from falling off. The sled cannot leave the upper portion of the extension ladder or it will come off of the extrusion. Because of this the robot must be loaded before the ladder is extended. The process for raising the robot to the roof can be seen in Figure 44.

To reduce the force required to tilt the platform over the sled a two bar linkage was designed. The linkage remains locked by spring loaded latches as the sled is pulled up the ladder; once the sled reaches the top the latches are depressed allowing the platform to be actuated. The platform can actuate to 90°, however, the roof will stop the platform at the appropriate angle for the robot to drive off at.

The forces involved in raising and actuating the ascender are not insignificant. The following is a compilation of free body diagrams and force calculations that were carried out to ensure that the ascender would be operable as designed. The forces were calculated for what we perceived was worse than the worst case scenario, where the ladder is setup at 90° to the ground. This situation is not possible and the ladder manufacturer only recommends up to a 75° angle. Therefore there is a safety factor of 15°.

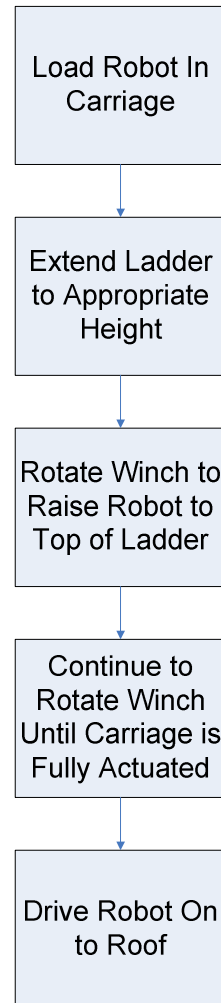


Figure 44:
Ascender Flow
Chart

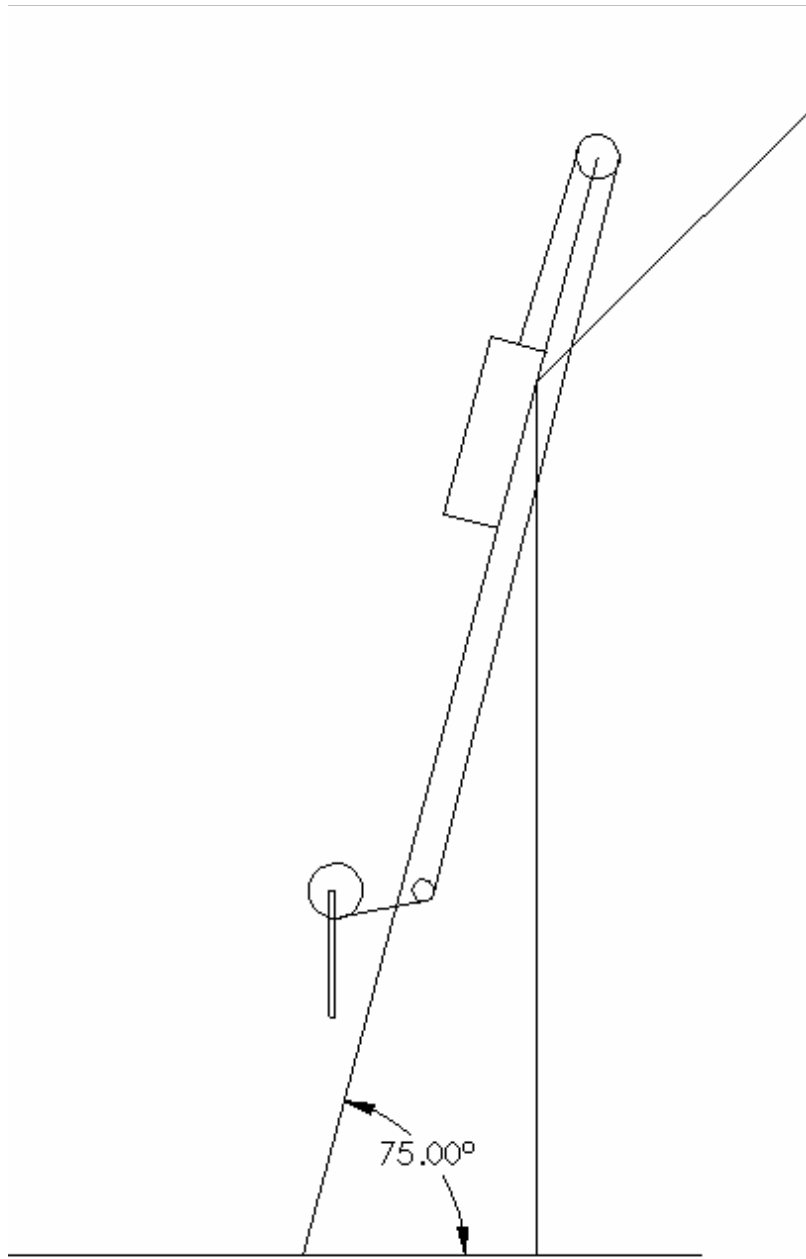


Figure 45: Ascender Schematic Diagram

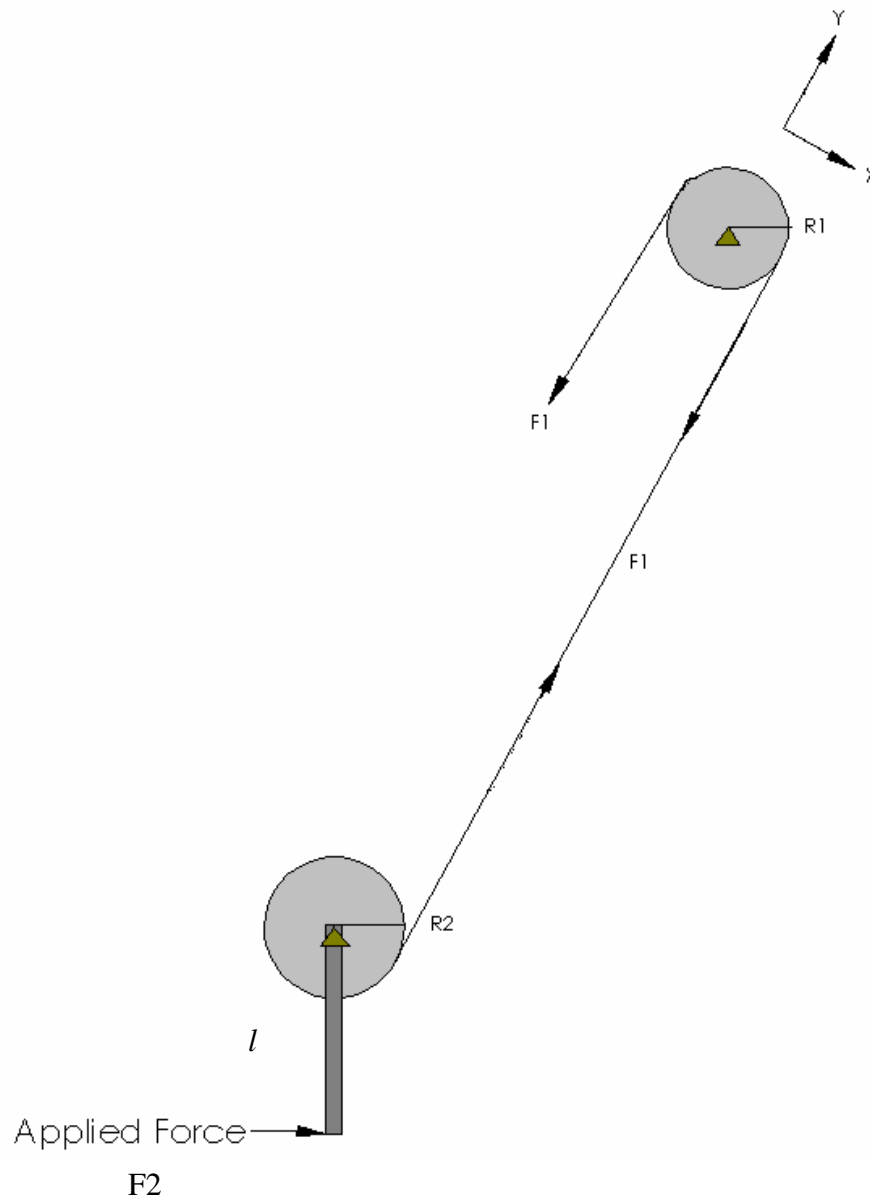


Figure 46: Pulley Free Body Diagram (Not To Scale)

$$\sum M_A = F1 * R2 - F2 * l = 0$$

$$\sum F_y = F1 - F1 = 0$$

$$\sum F_x = 0$$

$$F1 = 30lbs$$

$$R1 = 0.75"$$

$$R2 = 1.25"$$

Equation 1: Force at Handle

Using the winch that we built the required applied force to raise the robot in its carriage is 3.1 lbs. After the sled reaches the top of the ladder the carriage is actuated tilting the robot into a position from which it can be driven onto the roof. The force required to actuate the linkage is a function of the angle of tilt of the carriage. The applied force at the handle is a maximum of 18.8lbs. Figure 47 plots the force required to actuate the carriage linkage at the handle versus the angle of carriage tilt.

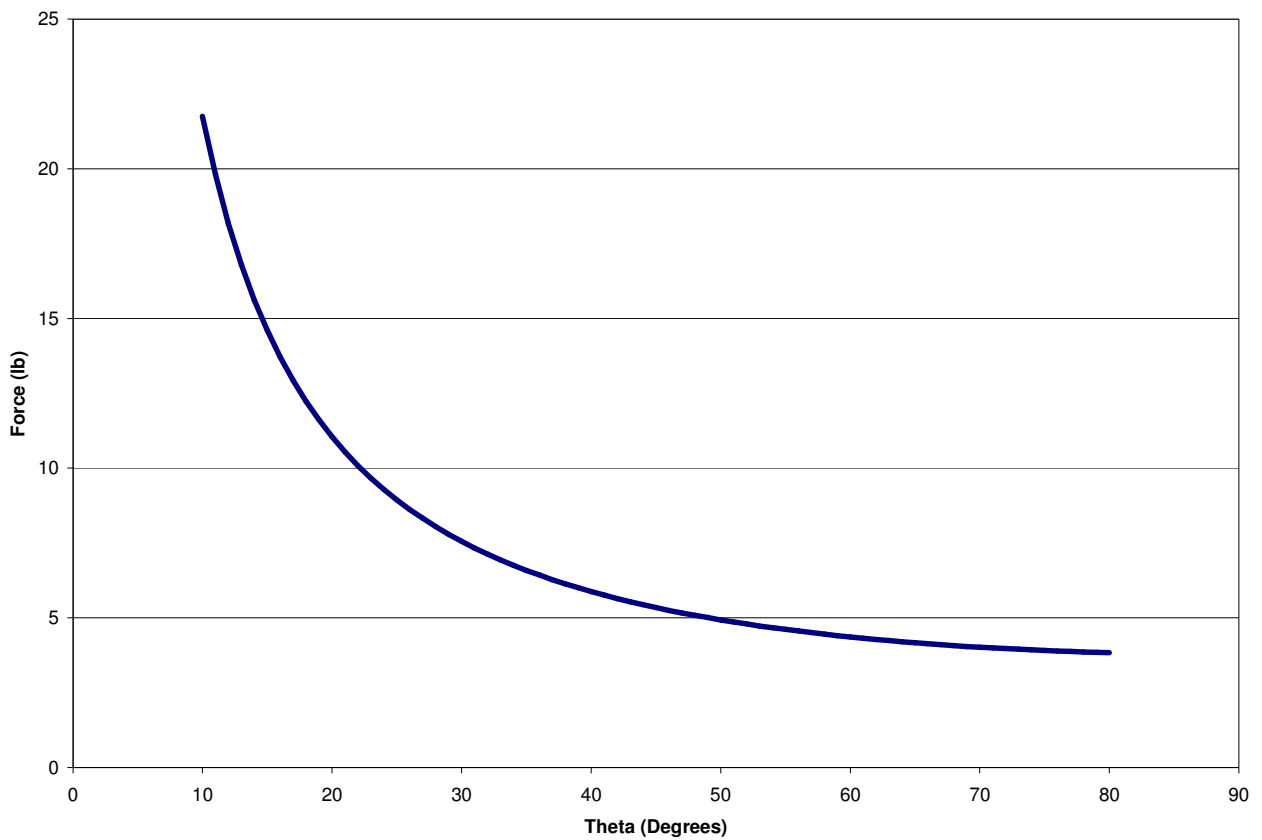


Figure 47: Carriage Tilt vs Force Applied at Handle

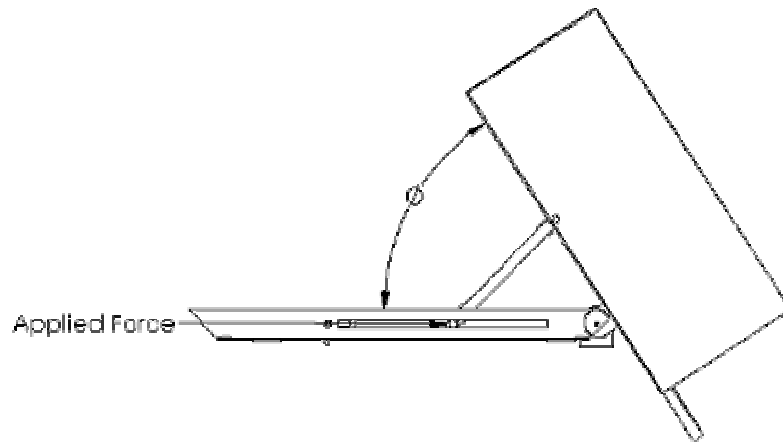


Figure 48: Carriage Operation Schematic Diagram

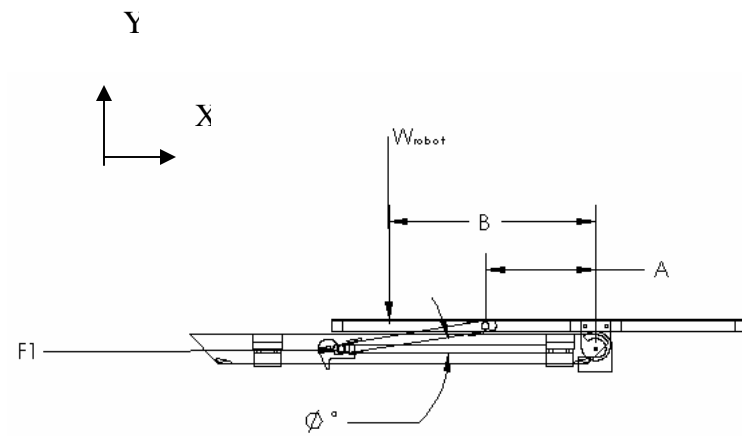


Figure 49: Applied Force Free Body Diagram

$$\sum M_A = A * F_1 * \sin(\theta) - W_{robot} * B = 0$$

$$\sum F_x = 0$$

$$\sum F_y = F_R - W_{robot} = 0$$

$$A = 8"$$

$$B = 14.5"$$

$$W_{robot} = 20lbs$$

Figure 50: Carriage Tilt vs Applied Force Equations

3 Results

This section describes the robot's performance as defined by our original task specifications. Once we finished our first prototype of the robot we went through several design iterations to improve performance. Our results come from testing performed on the mock-up roof located in Washburn Shops at WPI.

3.1 Navigation and Maneuverability

The robot is capable of traversing a roof, with some limitation on performance. It should be noted that to date, all testing has been conducted indoors on a mockup roof.

Specification: Maneuverability over composition tile at a 12/12 pitch

This specification has been partially met. The robot is capable of driving up 12/12 sloped composition tile roofs, but there is a very limited margin of safety. Conditions such as loose surface grains or dirty or worn out treads may cause the robot to slip. In testing, the robot would occasionally slide, but would still regain traction. The robot is capable of driving up 35° slopes without slipping.

Specification: Able to traverse rooftop and valley

This specification has been met, but with some limitations. The robot has enough ground clearance to clear the rooftop, however, the operator must take the rooftop straight on. The robot is also able to traverse valleys, but if the operator drives at too shallow an angle relative to the fold of the valley, a wheel may get caught, inducing a tipover. Driving at a perpendicular angle to the fold avoids any problems while traversing it.

Specification: Remote Control (Untethered)

This specification has been fully met. Again, it should be noted that indoor testing has prevented us from testing radio control at ranges likely to be found in outdoor conditions.

3.2 Autonomous Features

The robot's autonomous features respond very quickly. Limitations of the VEX controller make it difficult to send feedback to the operator when an override has triggered.

Specification: Can sense roof edge to prevent operator error

This specification has been fully met. The IR sensors lock the operator out of the controls between 12" – 18" from the roof edge and immediately stop the robot. The IR sensors are reliable and have an excellent signal to noise ratio, but have not been tested under outdoor lighting conditions.

Specification: Tilt warning

We were not able to meet this specification. We would have integrated accelerometer based tilt warnings into the system but because the Vex controller will not send information back to the operator we had no way of reading the warnings. One option we explored was to have the controller light up an LED on the robot which would be visible through the robot's camera. We deemed this to not be practical since the next version of the robot will not use the Vex controller and will be able to send the warning to the operator.

Specification: Manual overrides to allow operability in the event of a false sensor warning

This specification has been fully met.

3.3 Visual Inspection

As a proof of concept, the robot has shown that it is possible to use a camera to conduct a visual inspection, but the current prototype does not have enough resolution to produce a high quality image.

Specification: Robot incorporates a camera with a transmissible feed

This specification has been fully met.

Specification: Robot is able to produce a record of visual inspection

This specification has been fully met.

3.4 Durability

Without further testing, we do not have enough data to be able to evaluate the robot's durability or lifecycle. We estimate that the robot has been through 20 hours of operation so far, but a 50 hour lifecycle was specified. During this time, we have noted the following issues.

- The treads are subject to wear and may need to be replaced regularly. (We are unable to specify how often without further testing)
- Occasionally, wires came loose during operation, which cause unpredictable behavior.
- The joint which holds together the two chassis halves broke with two different failure modes. The first time, a screw which held the pin in the roll axis failed. The second time, a lost signal from the potentiometer caused the robot to drive against itself, breaking the key stock which holds the pin on the capstan. See Section 4 for further recommendations.

3.5 Ascender Mechanism

We have built an ascender mechanism that will transport the robot to a height of twenty feet. This meets the specification that we originally stated; however, the packaging of the ascender is not consistent with the desired state. Travelers would prefer that the ascender be small enough to fit in the back of a car and light enough for one person to setup. This is not the current case and we ran into trouble with time resources. See Section 4 for further recommendations.

4 Recommendations

This Major Qualifying Project has significant potential for follow on projects at WPI. For this reason this section is geared towards future improvements to the design of the robot and ascender by follow on MQP teams.

Redesign the driveline and control system:

The current driveline and control system is limited by its components. A redesign of the driveline and control system would provide more efficient power consumption, better handling, and simpler computer code.

Replace the window motors to improve power consumption and handling:

The window motors that drive the robot are not optimal because they run too fast and have a significant amount of backlash. The motors are run very close to their minimum speed, resulting in poor power efficiency. The backlash limits the response time of the control loop and makes it difficult to avoid control loop oscillations.

Replace the current optical encoders with quadrature encoders to simplify code:

The optical encoders on the robot are non-directional, meaning they can only tell the speed of the wheels, but not the velocity. A quadrature encoder can sense both speed and velocity and would greatly reduce the complexity of the computer code.

Move the optical encoders closer to the motor in the gear train to improve handling:

The optical encoders are currently mounted near the rims of wheels, outside of the chassis, where they are exposed and unprotected. Furthermore, it is very important for the robot to be able to control wheel speed very precisely,

especially at low speeds. If the encoder was mounted on the motor before the reduction through the worm gear, it would improve the resolution of the encoder, thereby shortening the response time of the control loop and improving handling.

Replace the Vex and Victor Speed Controllers to improve handling and simplify code:

The pulse width modulated (PWM) signal sent to the Victor Speed Controllers contains analog information on both direction and duty cycle for the signal that is sent to the motors. To avoid crossover issues with the input signal changing from forward to reverse, the Victor has a deadband. The deadband reduces handling performance at slow speeds, and introduces incredible complexity to finding workarounds in the computer code.

All speed controllers will have the same problem with the analog PWM signal sent by the Vex. The only solution is to replace the Vex with another microcontroller that can have tighter integration between the processor and the speed controller, avoiding the issue altogether.

Replace the Vex microcontroller with something more powerful:

The original intent of this project was to use the Qwerk microcontroller available from Charmed labs to control the robot. After spending several months trying to adapt it to our uses we decided it would be more productive to use the Vex microcontroller that we had from the early prototype despite its limitations. The Vex, while it is able to support all of our signal processing and controls has a major limitation in that it cannot log or transmit any data back to the operator. Because the intent of the roof robot is to inspect the roof it will be necessary to incorporate a more advanced controller, such as the Qwerk, into the next iteration of the roof robot to read and transmit sensor data.

Replace the X10 camera with a higher resolution device:

One of the advantages to the Qwerk controller was its support for transmitting a webcam feed to a laptop. Webcams are now capable of providing feeds with video over one mega pixel in resolution. When we decided not to use the Qwerk controller we had to provide an alternative camera with an independent transmitter. The simplest way to provide video feedback was by incorporating an X10 wireless home security camera into the design. The X10 is an all inclusive package has a transmitter and battery pack. Unfortunately the resolution of this camera is not very clear and a better camera should be identified.

Have separate navigation and inspection cameras:

The camera, at it present location on a mast on the back of the robot, is not suited for close up visual inspections. We recommend keeping a camera there for the purpose of navigation, where it is possible to see the robot in the field of view and have a better sense of the surroundings, but for the purpose of visual inspection, we recommend having a dedicated camera on the front of the robot. The camera would be in a more ideal position to look at the roof and could send back high resolution photos of the target areas.

Replace X10 camera transmitter with a more powerful transmitter:

The transmission of video from the X10 to the laptop is accomplished through a radio channel. The provided antenna for the X10 is a directional antenna. The signal becomes fuzzy or non-existent under certain conditions where the transmitter is not facing the receiver. The X10 has been modified by other people to use an omni-directional antenna; however we did not have time to incorporate this into the roof robot. We therefore recommend that more suitable transmitter be found, or the camera be replaced by a more appropriate one.

Add additional autonomous sensors to prevent operator induced flip-over:

Currently, the robot has no features to prevent the operator from inducing an unstable condition where the robot may flip over. Additional sensors, such as an accelerometer, can be used to detect this condition. Other situations may be induced when a wheel gets stuck, causing the robot to pivot about the stuck wheel and flip the robot over. To prevent this condition, we would recommend additional sensors to detect whenever a wheel loses contact with the ground.

Re-evaluate battery requirements and alter individual battery packs to function as one:

The battery packs currently supplying the robot with power run at twelve volts. There are three of these battery packs wired in parallel to supply power necessary for our specified operating time of one half hour at full power. These battery packs are not wired to charge through one cable. The current setup requires that each battery pack be charged individually until full. Rewiring the charge terminals would simplify this procedure. Additionally the Vex controller runs off of a nine-volt battery. This has its own separate battery pack. The X10 camera also runs off of its own 24-volt battery pack. Through some relatively simple circuitry these three systems; drive train, controller and camera could be powered off of one battery pack. This would also simplify the procedure to turn on the robot which currently requires activating three separate switches to turn on power to all the systems.

Improve the durability of the joint components:

The construction of the joint has two observed failure modes that need to be corrected. The first is the welded key-stock bar connecting the rotating shaft A to the capstan. Under high torque loads the shaft will break causing the robot to not have the ability to turn. Furthermore the bolt holding shaft A to shaft B will also snap. This is due to tolerances between joint A and B being too loose. The resulting gap means that all forces associated with the weight of the robot may be directed through this .195" bolt. If this bolt breaks then the joint falls apart and the robot is left in two pieces. To fix this

shaft A and B should be fabricated out of a single piece of stock eliminating the bolt completely.

Further research into friction materials for the wheels:

Although the robot maintains static frictional stability with roof on the specified angle of 45°, if slip is induced it does not have a satisfactory margin of safety to arrest sliding. Despite our testing of materials we were not able to find anything with better frictional coefficients than EPDM foam. To prevent slip we implemented the electronic traction control system, this improves traction capabilities, but it is not fool proof. Further testing should also be done on weathered shingles and wet shingles to see what the frictional properties under these conditions are. One proposed solution would be to develop an active friction material that interfaces with the roof surface the same way a gecko's toe is able to hold on to a vertical wall.

Redesign of the ascender system:

The largest design challenge the ascender presented was how to deliver the robot to the roof while staying small and lightweight enough to be easily transportable. Although the robot and the ascender were being developed concurrently the design of the robot heavily influenced the design of the ascender. The robot's size and weight made it necessary to have a robust ascender mechanism which in turn became large and heavy. If the robot had been designed to fit a compact ascender mechanism it is possible it would have worked out much more efficiently. The concurrent design of the robot and the ascender was a large work order. We think that should one have preceded the other entirely the results would have been more acceptable. This way each system would receive everyone's full attention. Future considerations for the ascender should include the use of a telescoping device that is only as big as it absolutely needs to be, furthermore the size of the robot should not be the ultimate dictator of the ascenders design specifications. Through the use of a more customized drive train and electronics the size of the robot could easily be reduced making the load on the ascender much less and

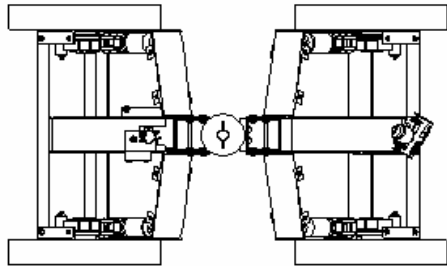
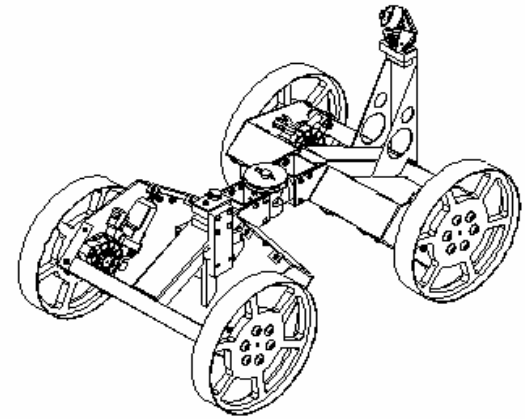
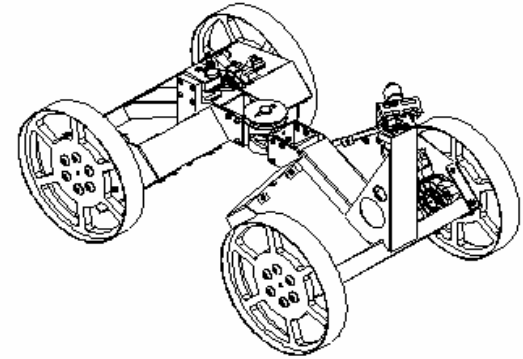
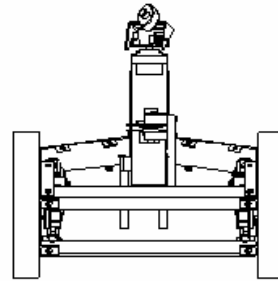
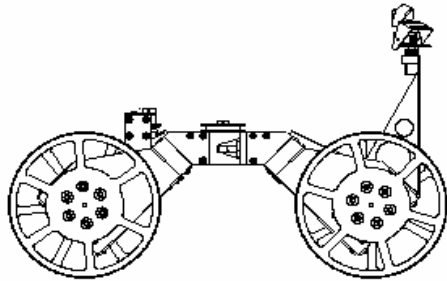
opening up the possibilities for further innovation. For instance we had to rule out using a telescoping pole to place the robot on the roof because it would not support the size of the robot. If the robot were one third of its current weight (approximately six pounds), it would have been possible to use a commercially available telescoping pole as a means of placing the robot on the roof.

5 Conclusions

Through this Major Qualifying Project we have shown that it is possible to build a robot that can operate on a 45° roof with an asphalt shingle surface. We have met nearly all of the original task specifications but were limited by time constraints and complications with programming. The prototype that we built has room for improvement and we have detailed these areas in our recommendations section of this report. This project lays the groundwork for future projects that will go on to improve on our designs and develop further sensor packages for the St Paul Travelers insurance company.

Appendix A- CAD Drawings

This appendix is a compilation of CAD drawings relevant to the design of the roof robot. All CAD files were made using Solidworks Design Studio. The appropriate electronic files are included on the CD attached to the hardcopy of this report. Solidworks was used as a design aid in determining the dimensions and geometries of all of the components that we machined. Some of the components were made using HAAS CNC mills while others were made using manual mills and lathes. Due to the iterative design process of this project the CAD model does not completely reflect the ultimate state of the robot. For this reason any parts needing to be reproduced should be referenced against the physical part on the robot to determine what liberties were taken during the machining of the part.



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WORCESTER POLYTECHNIC INSTITUTE. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WORCESTER POLYTECHNIC INSTITUTE IS PROHIBITED.

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL

FINISH

DO NOT SCALE DRAWING

UNLESS OTHERWISE SPECIFIED:	NAME	DATE
DRAWN	NCM	4/24/07
CHECKED		

COMMENTS:

WORCESTER POLYTECHNIC INSTITUTE

TITLE:
 BUILT PARTS ASSEMBLY

SIZE	DWG. NO.	REV
A	ROBOT	1

SCALE: 1:10 WEIGHT: 1 SHEET 1 OF 2

5

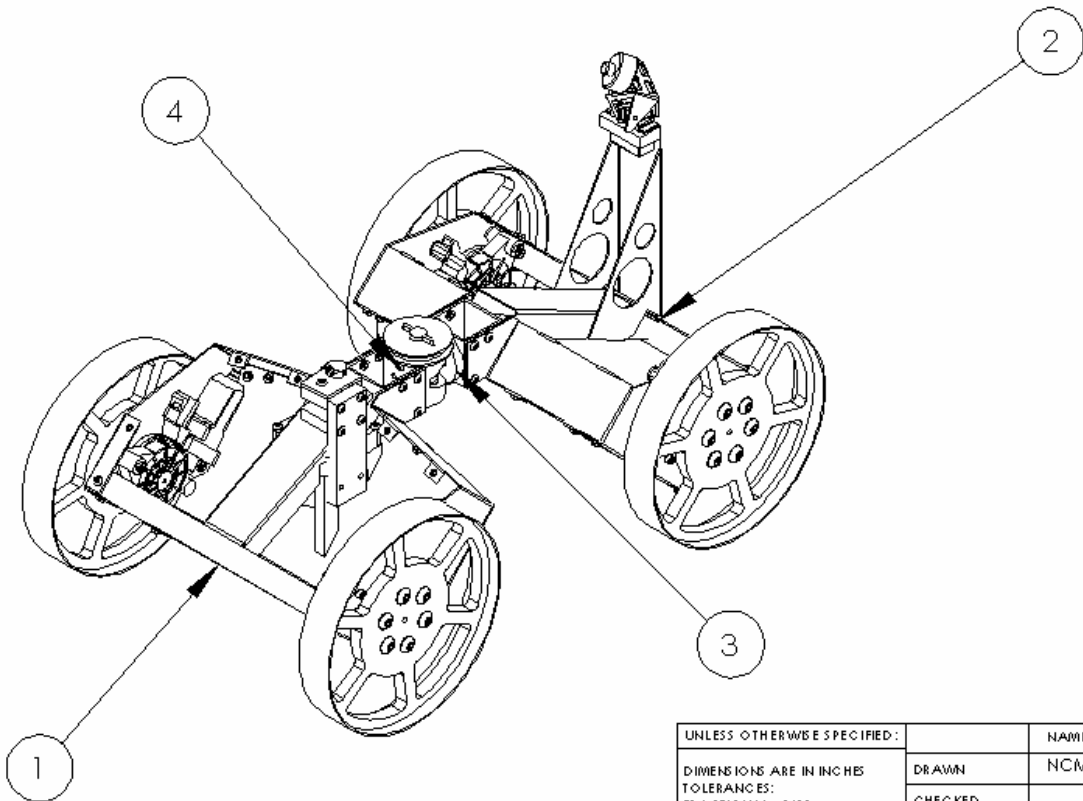
4

3

2

1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	CH6	CHASSIS ASSEMBLY 6	1
2	CH7	CHASSIS CAMERA SIDE	1
3	JO3	JOINT SUB ASSEMBLY 3	1
4	91306A341	BUTTONHEAD BOLT 10-24 X 3/8	4

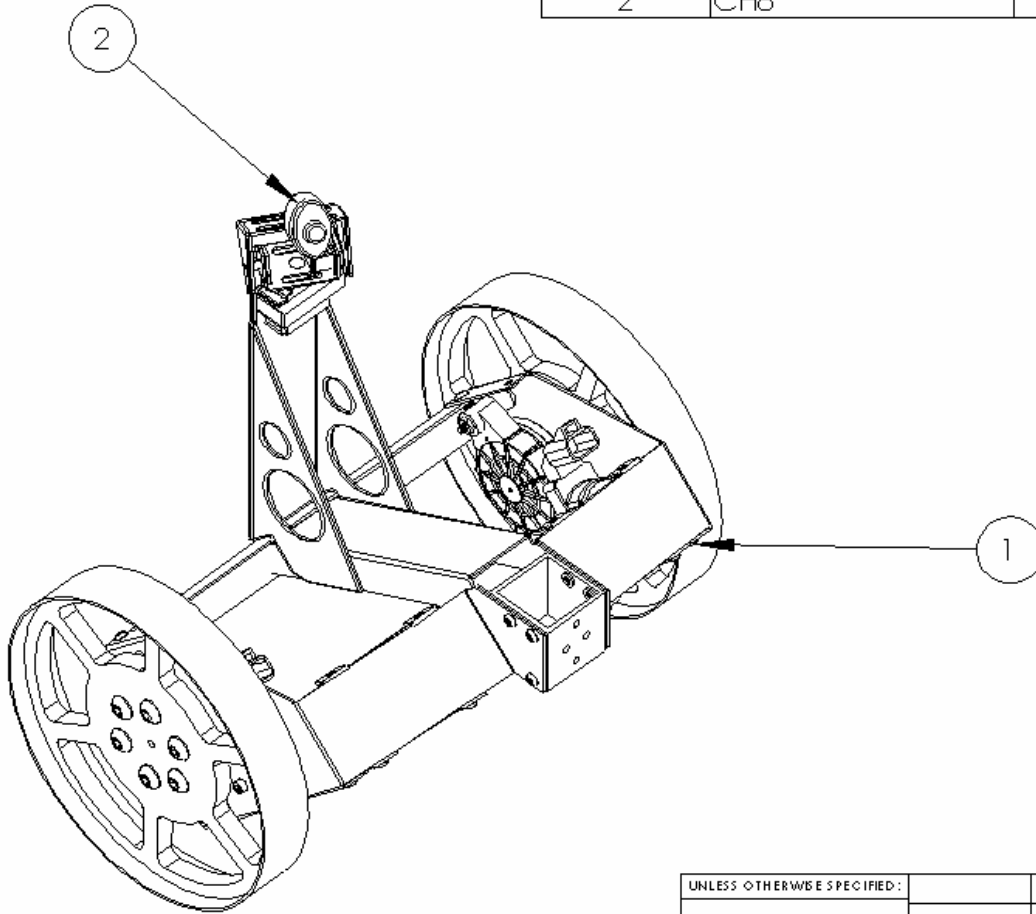


UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: BUILT PARTS ASSEMBLY
TOLERANCES:		CHECKED			
FRACTIONAL: $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL: ± 0.01					SIZE DWG. NO. REV A ROBOT 1
THREE PLACE DECIMAL: ± 0.005					
MATERIAL		COMMENTS:			SCALE: 1:6
FINISH					WEIGHT: 1
DO NOT SCALE DRAWING					SHEET 2 OF 2

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPI AND ST PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPI AND ST PAUL TRAVELERS IS PROHIBITED.

5 4 3 2 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	CH7	CHASSIS CAMERA SIDE	1
2	CH8	CAMERA MAST ASSEMBLY	1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE			
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE:		
TOLERANCES:		CHECKED					
FRACTIONAL ± 1/32					CHASSIS ASSEMBLY 6		
ANGULAR ± 3							
TWO PLACE DECIMAL ±0.01							
THREE PLACE DECIMAL ±0.005							
<p>PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPI AND ST PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPI AND ST PAUL TRAVELERS IS PROHIBITED.</p>		COMMENTS:		SIZE	DWG. NO.	REV	
		MATERIAL			A	CH9	1
FINISH				SCALE: 1:4		WEIGHT: 1	SHEET 1 OF 1
DO NOT SCALE DRAWING							

5

4

4

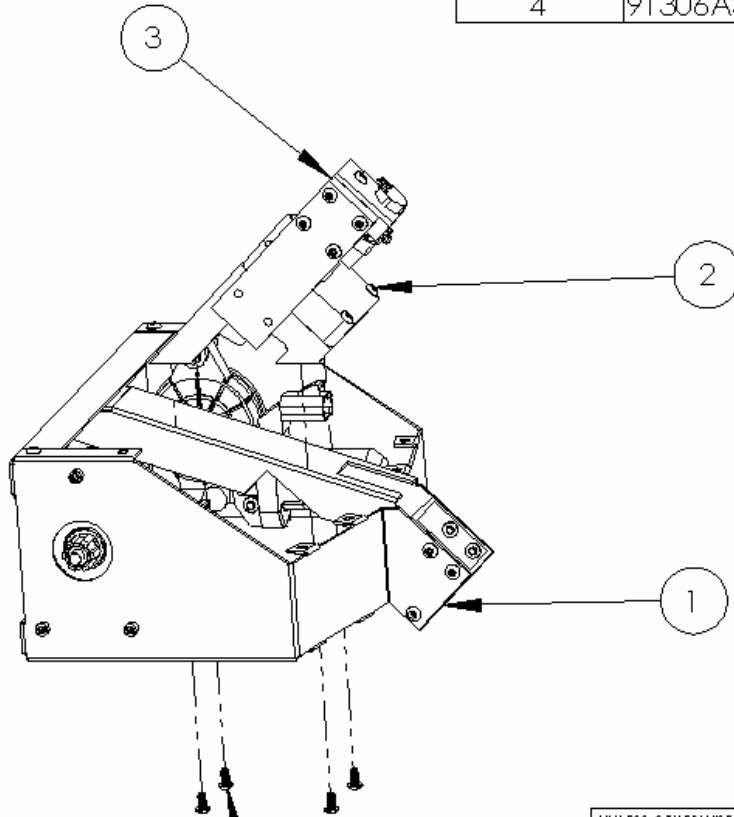
3

1

2

1

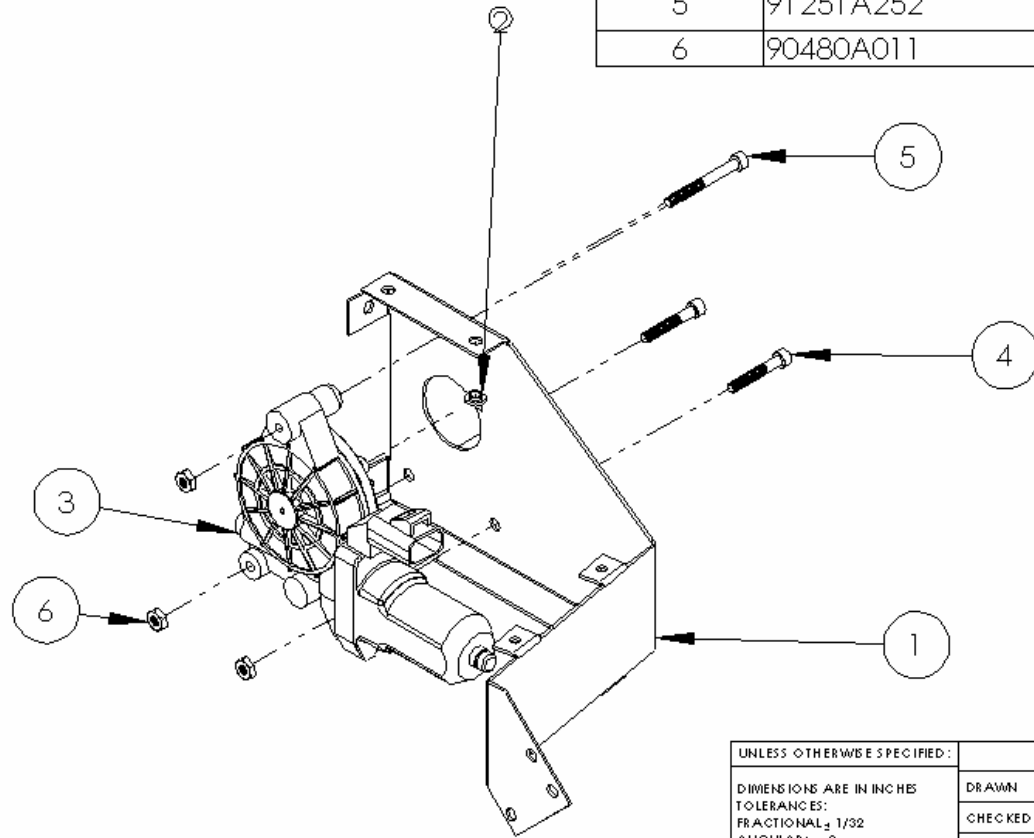
ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	CH4	CHASSIS	1
2	JO2	JOINT SUB ASSEMBLY B	1
3	JO01	JOINT SUB-ASSEMBLY A	1
4	91306A341	BUTTONHEAD BOLT 10-24 X 3/8	4



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: CHASSIS ASSEMBLY 5
TOLERANCES:		CHECKED			
FRACTIONAL: $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL: ± 0.01					SIZE DWG. NO. REV
THREE PLACE DECIMAL: ± 0.005		COMMENTS:		A C06 1	SCALE: 1:4 WEIGHT: 1 SHEET 1 OF 1
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPICA AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPICA AND ST. PAUL TRAVELERS IS PROHIBITED.		MATERIAL			
		FINISH			
		DO NOT SCALE DRAWING			

5 4 3 2 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	CHASSIS01	CHASSIS SIDE PANEL LEFT	1
2	pem	10-24 THREADED INSERT	3
3	ND Motor Left	NIPPON DENSO MOTOR LEFT	1
4	91 251 A252	SOCKETHEAD CAP SCREW 10-24X1.25	2
5	91 251 A252	SOCKETHEAD CAP SCREW 10-24 X 1.75	1
6	90480A011	10-24 NUT	3

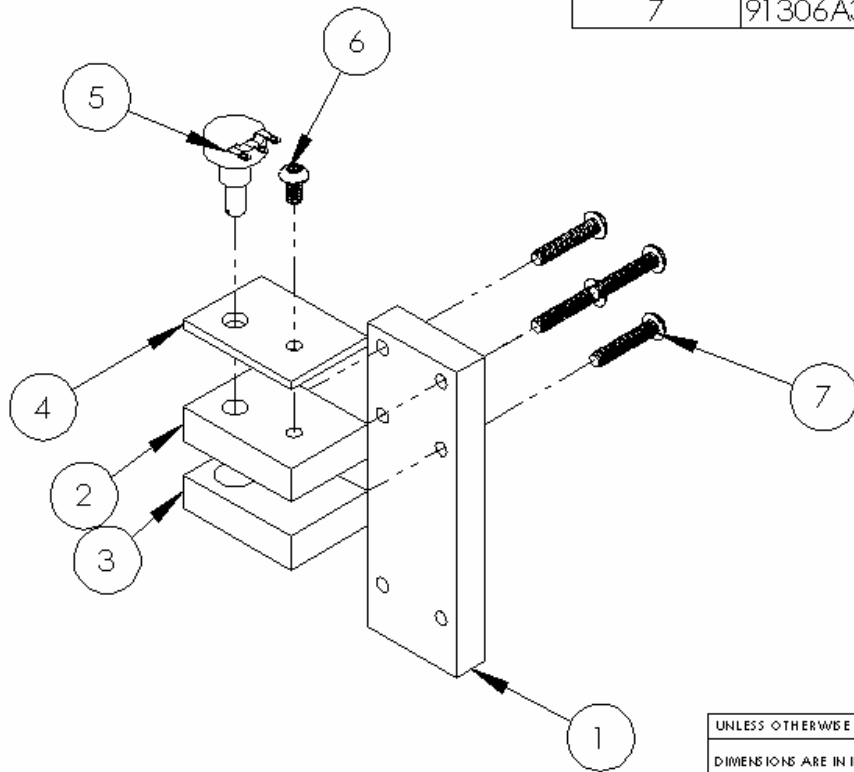


UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: CHASSIS ASSEMBLY LEFT
TOLERANCES:		CHECKED			
FRACTIONAL: $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL: ± 0.01					SIZE DWG. NO. REV A CH3L 1
THREE PLACE DECIMAL: ± 0.005					
MATERIAL		COMMENTS:		SCALE: 1:3 WEIGHT: 1 SHEET 1 OF 1	
FINISH		CHASSIS ASSEMBLY RIGHT IS MIRROR IMAGE OF LEFT			
DO NOT SCALE DRAWING					

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPIC AND ST PALL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPIC AND ST PALL TRAVELERS IS PROHIBITED.

5 4 3 2 1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	JOINT05	JOINT SIDE BLOCK	1
2	JOINT07	JOINT BOTTOM BLOCK	1
3	JOINT06	JOINT BOTTOM BLOCK	1
4	JOINT08	JOINT TOP PLATE	1
5	POT	POTENTIOMETER	1
6	91306A341	BUTTONHEAD BOLT 10-24 X 3/8	1
7	91306A341	BUTTONHEAD BOLT 10-24 X 1	4

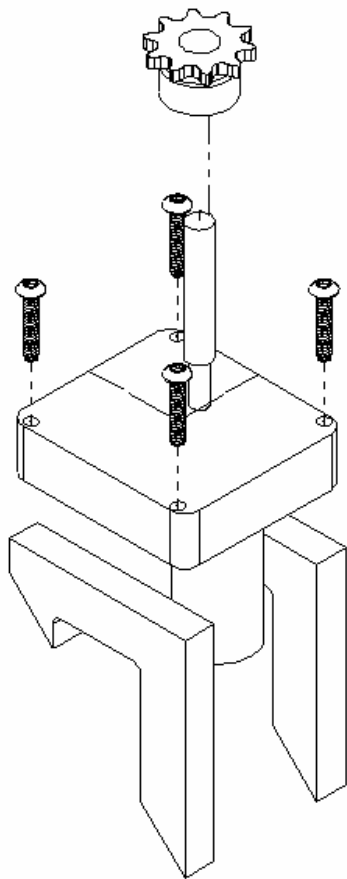


UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: JOINT SUB-ASSEMBLY A
TOLERANCES:		CHECKED			
FRACTIONAL: $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL ± 0.01					SIZE DWG. NO. REV A JA1 1
THREE PLACE DECIMAL ± 0.005					
MATERIAL		COMMENTS:			SCALE: 1/2 WEIGHT: 1 SHEET 1 OF 1
FINISH					
DO NOT SCALE DRAWING					

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPPI AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPPI AND ST. PAUL TRAVELERS IS PROHIBITED.

5 4 3 2 1

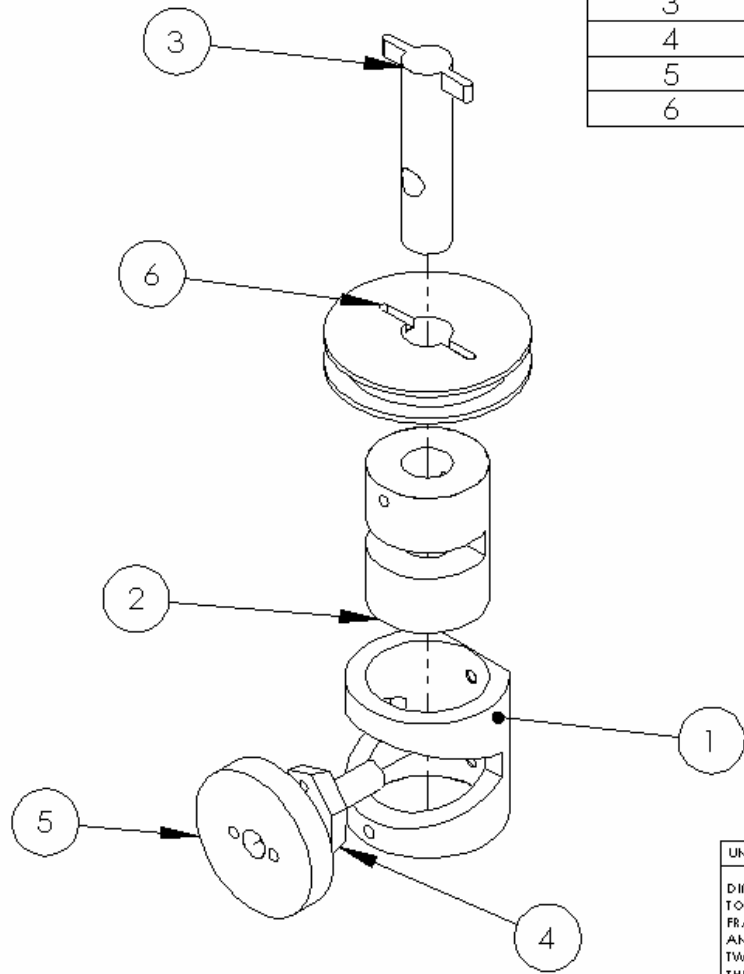
ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	JOINT01	JOINT MOTOR MOUNTING BRACKET	2
2	JOINT02	JOINT MOTOR	1
3	JOINT13	JOINT MOTOR SHAFT EXTENSION	1
4	91306A341	BUTTONHEAD BOLT 10-24 X 1	4
5	JOINT14	JOINT DRIVE SPROCKET 9 TOOTH	1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	
TOLERANCES:		CHECKED		TITLE:	
FRACTIONAL $\pm 1/32$				JOINT SUB ASSEMBLY B	
ANGULAR: ± 3				SIZE DWG. NO. REV	
TWO PLACE DECIMAL ± 0.01				A JO2 1	
THREE PLACE DECIMAL ± 0.005				SCALE: 1:2 WEIGHT: 1 SHEET 1 OF 1	
PROPRIETARY AND CONFIDENTIAL		COMMENTS:			
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPI AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPI AND ST. PAUL TRAVELERS IS PROHIBITED.		MATERIAL			
		FINISH			
		DO NOT SCALE DRAWING			

5 4 3 2 1

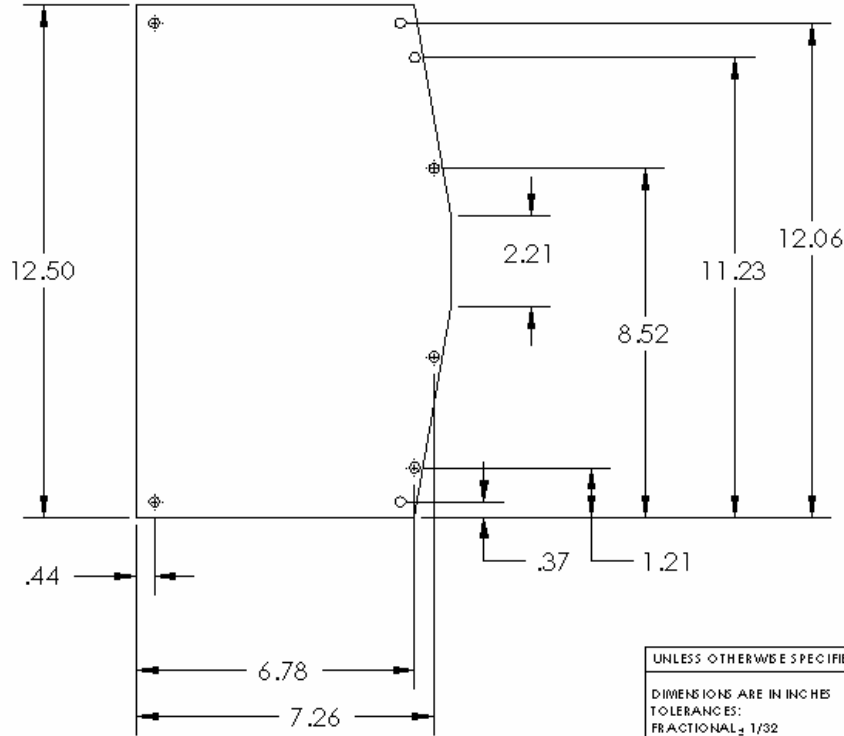
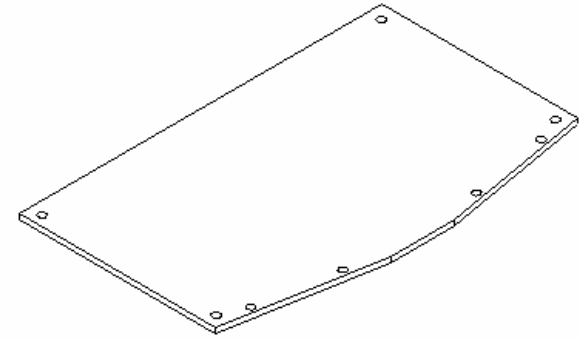
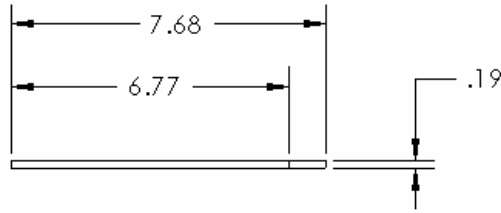
ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	JOINT10	JOINT OUTER HOUSING	1
2	JOINT16	JOINT INNER HOUSING	1
3	JOINT12	JOINT VERTICAL SHAFT	1
4	JOINT11	JOINT TOGGLE	1
5	JOINT09	JOINT SPACER	1
6	JOINT15	JOINT CAPSTAN	1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: JOINT SUB ASSEMBLY 3
TOLERANCES:		CHECKED			
FRACTIONAL $\pm 1/32$					
ANGULAR ± 3					
TWO PLACE DECIMAL ± 0.01					SIZE DWG. NO. REV A JO3 1
THREE PLACE DECIMAL ± 0.005					
MATERIAL		COMMENTS:			SCALE: 1:2
FINISH					WEIGHT: 1
DO NOT SCALE DRAWING					SHEET 1 OF 1

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WORCESTER POLYTECHNIC INSTITUTE. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WORCESTER POLYTECHNIC INSTITUTE IS PROHIBITED.

5 4 4 3 2 1



UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL ± 1/32
 ANGULAR: ± 3
 TWO PLACE DECIMAL ±0.01
 THREE PLACE DECIMAL ±0.001

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WPI AND ST PAUL TRAVELERS. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WPI AND ST PAUL TRAVELERS IS
 PROHIBITED.

MATERIAL
 POLYCARBONATE

FINISH

DO NOT SCALE DRAWING

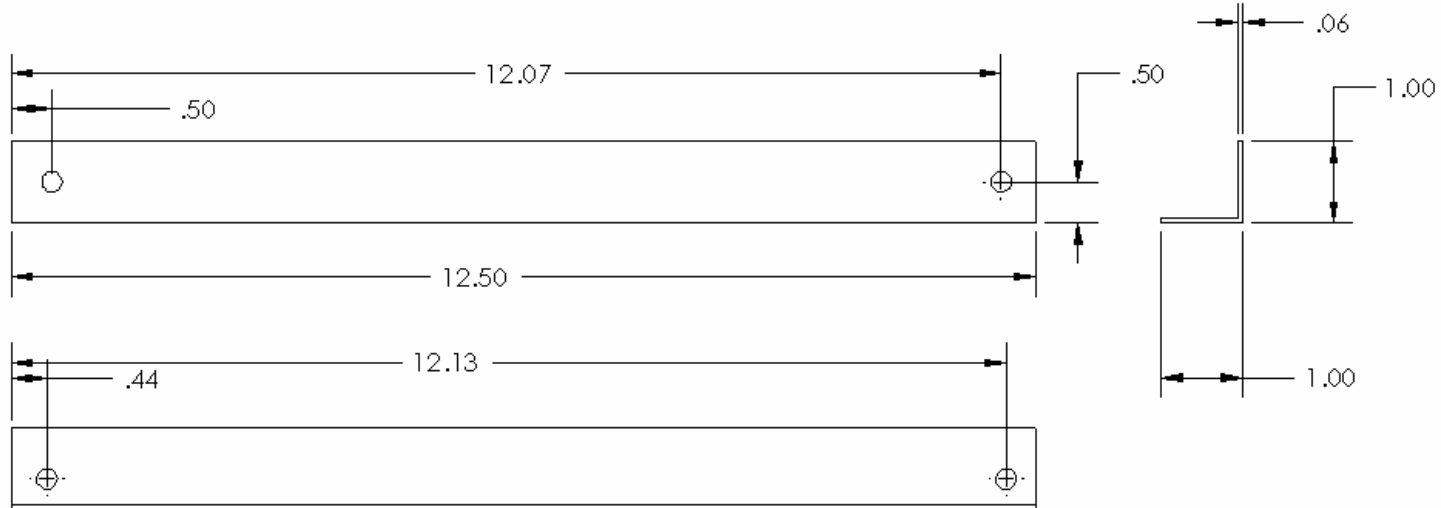
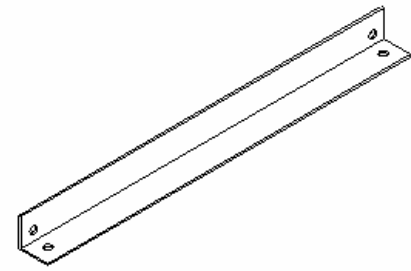
	NAME	DATE
DRAWN	NCM	4/27/07
CHECKED		

TITLE:
BOTTOM PLATE

COMMENTS:

SIZE	DWG. NO.	REV
A	C01	1
SCALE: 1:4	WEIGHT:	SHEET 1 OF 1

5 4 3 2 1

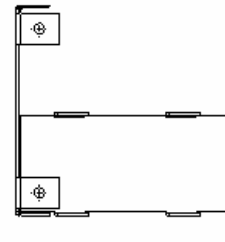
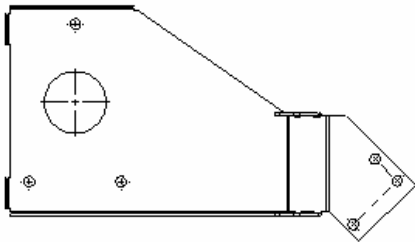
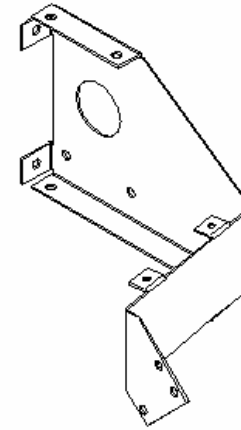
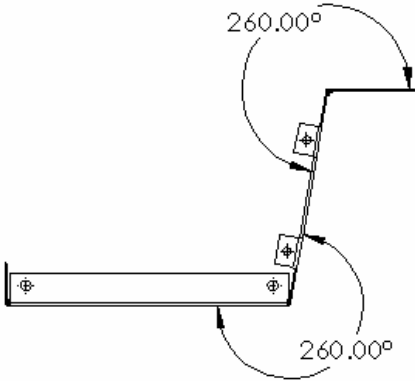


UNLESS OTHERWISE SPECIFIED:	NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES	DRAWN	NCM	4/24/07	TITLE: ANGLE BRACKET
TOLERANCES:	CHECKED			
FRACTIONAL: 1/32				
ANGULAR: ±3				
TWO PLACE DECIMAL: ±0.01				SIZE DWG. NO. REV
THREE PLACE DECIMAL: ±0.005				A C02 1
	COMMENTS:		SCALE: 1:2	WEIGHT:
				SHEET 1 OF 1

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

MATERIAL **ALUMINUM**
 FINISH
 DO NOT SCALE DRAWING

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WORCESTER POLYTECHNIC INSTITUTE. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WORCESTER POLYTECHNIC INSTITUTE IS PROHIBITED.

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR: ± 3 '
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL: ALUMINUM

FINISH:

DO NOT SCALE DRAWING

NAME: NCM
 DATE: 4/24/07

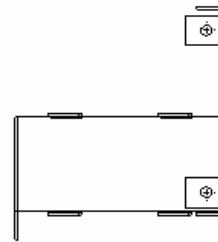
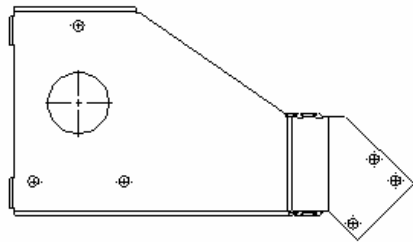
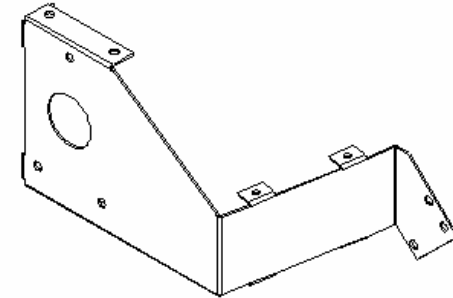
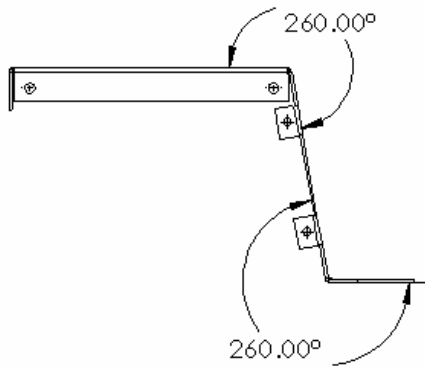
WORCESTER POLYTECHNIC INSTITUTE

TITLE:
 CHASSIS SIDE PANEL LEFT

SIZE: A DWG. NO.: C04 REV: 1

SCALE: 1:4 WEIGHT: 0.33 SHEET 1 OF 5

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE		
DIMENSIONS ARE IN INCHES	DRAWN	NCM	4/24/07	TITLE: CHASSIS BRACKET RIGHT		
TOLERANCES:	CHECKED					
FRACTIONAL $\pm 1/32$						
ANGULAR ± 3						
TWO PLACE DECIMAL ± 0.01				SIZE	DWG. NO.	REV
THREE PLACE DECIMAL ± 0.005				A	C04	1
	COMMENTS:			SCALE: 1:4	WEIGHT: 0.33	SHEET 2 OF 5
MATERIAL	ALUMINUM					
FINISH						
	DO NOT SCALE DRAWING					

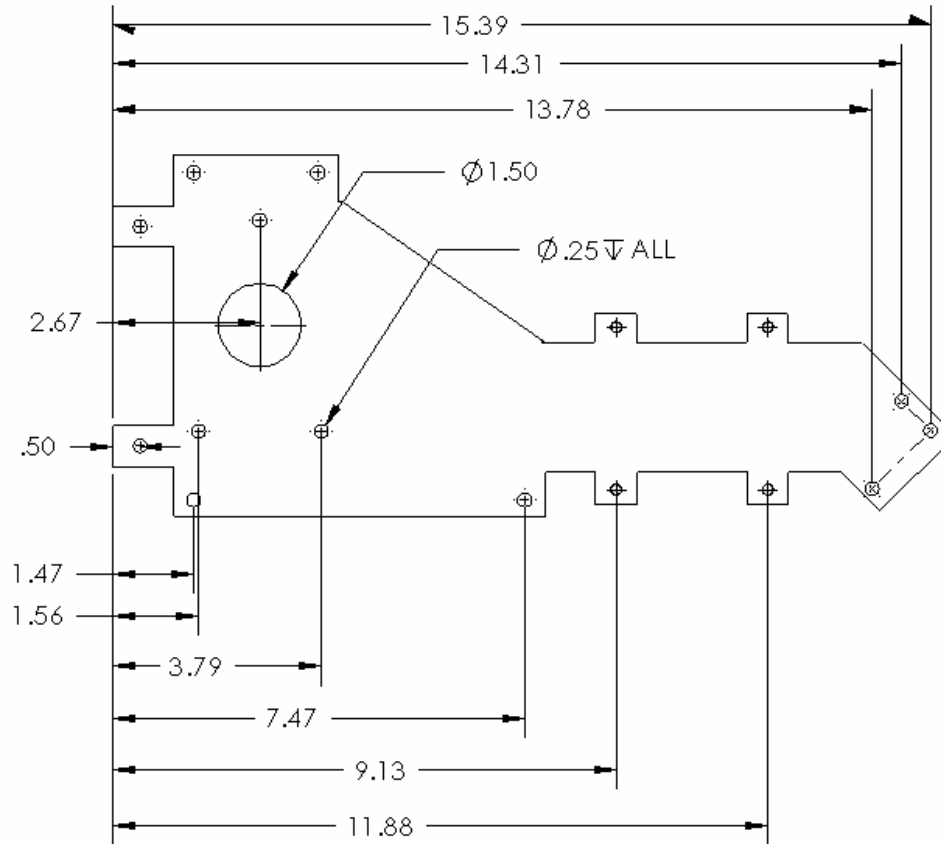
5

4

3

2

1



UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR: ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

MATERIAL ALUMINUM
 FINISH
 DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	NCM	4/24/07
CHECKED		

COMMENTS:

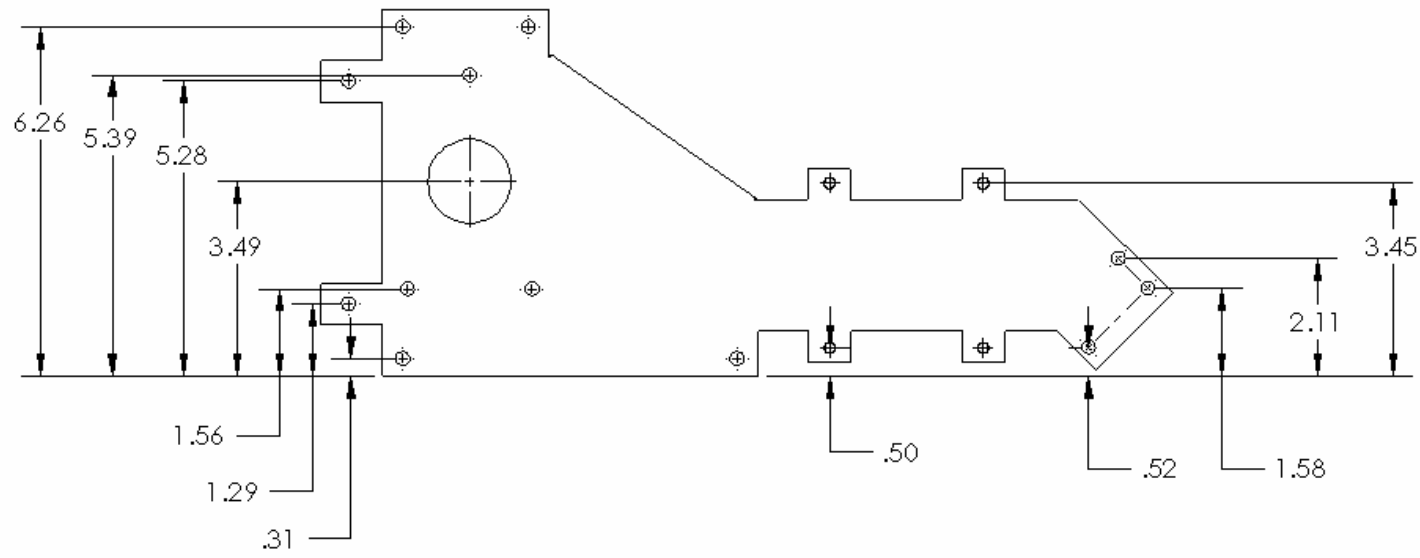
WORCESTER POLYTECHNIC INSTITUTE

TITLE:
 CHASSIS PANEL

SIZE	DWG. NO.	REV
A	C04	1

SCALE: 1:3 WEIGHT: 0.33 SHEET 3 OF 5

5 4 3 2 1

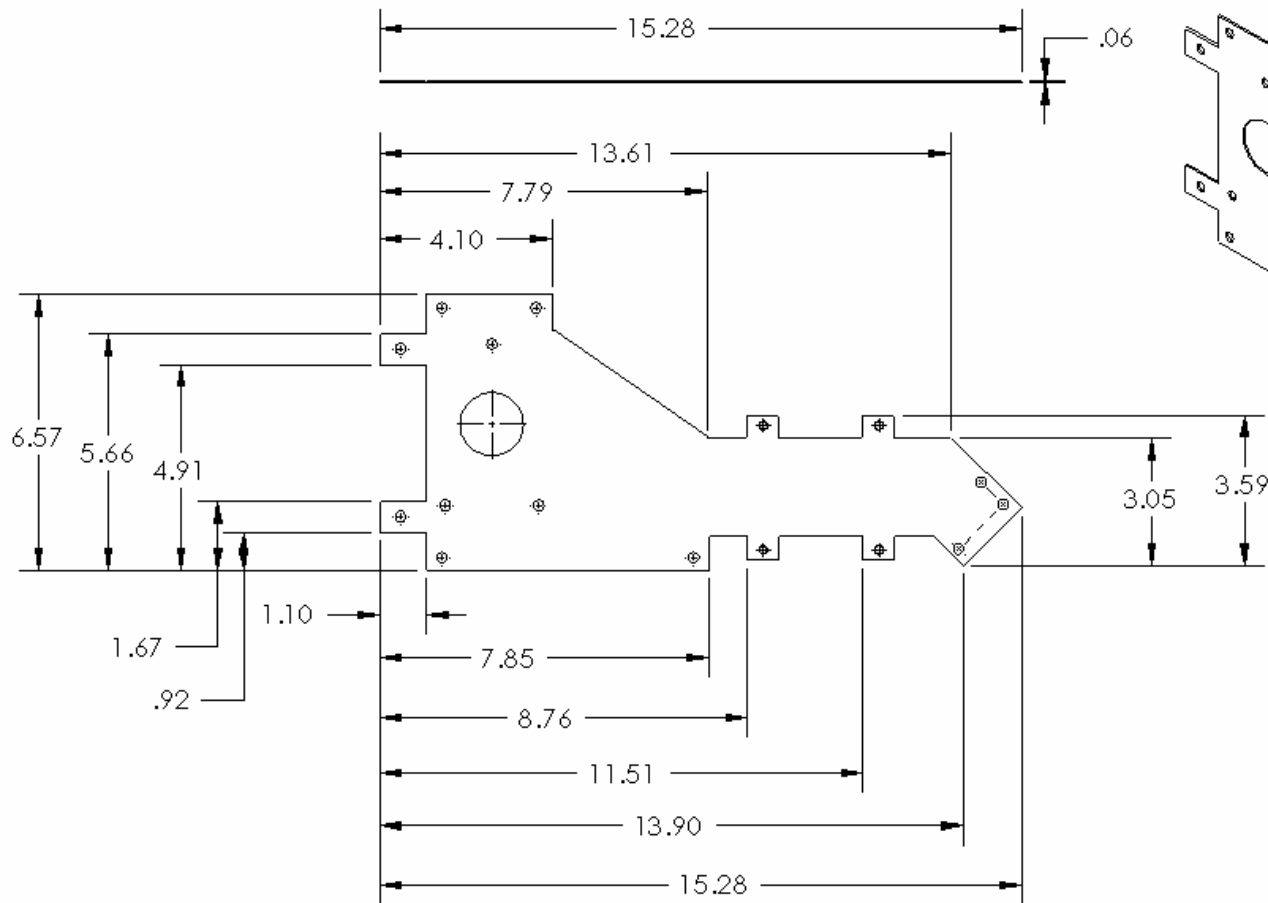


UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	
TOLERANCES:		CHECKED		TITLE:	
FRACTIONAL ± 1/32				CHASSIS PANEL	
ANGULAR: ± 3				SIZE DWG. NO. REV	
TWO PLACE DECIMAL ±0.01				A C04 1	
THREE PLACE DECIMAL ±0.005				SCALE: 1:3 WEIGHT: 0.33 SHEET 4 OF 5	
COMMENTS:					

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPINC AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPINC AND ST. PAUL TRAVELERS IS PROHIBITED.

MATERIAL: ALUMINUM
 FINISH:
 DO NOT SCALE DRAWING

5 4 3 2 1

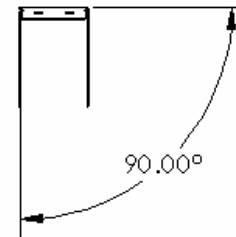
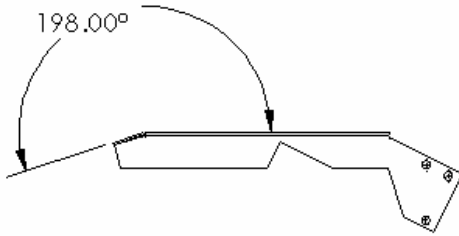
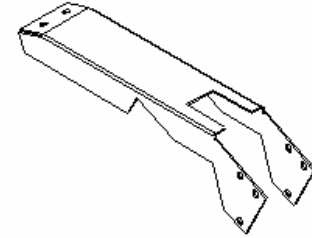


UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	
TOLERANCES:		CHECKED		TITLE:	
FRACTIONAL: 1/32				CHASSIS PANEL	
ANGULAR: ± 3°				SIZE	
TWO PLACE DECIMAL ±0.01				DWG. NO.	
THREE PLACE DECIMAL ±0.005				C04	
				REV	
				1	
		COMMENTS:		SCALE: 1:4	
				WEIGHT: 0.33	
				SHEET 5 OF 5	

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPINC AND SHALL REMAIN THE PROPERTY OF WPINC WHETHER REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS, WITHOUT THE WRITTEN PERMISSION OF WPINC AND SHALL REMAIN THE PROPERTY OF WPINC.

DO NOT SCALE DRAWING

5 4 3 2 1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE			
DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL $\pm 1/32$ ANGULAR: ± 3 TWO PLACE DECIMAL ± 0.01 THREE PLACE DECIMAL ± 0.005		DRAWN	NCM	4/24/07	TITLE: CHASSIS BRACKET SIDE A		
		CHECKED					
PROPRIETARY AND CONFIDENTIAL THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPPI AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPPI AND ST. PAUL TRAVELERS IS PROHIBITED.		COMMENTS:		SIZE	DWG. NO.	REV	
		MATERIAL ALUMINUM			A	C05	1
FINISH				SCALE: 1:5		WEIGHT: 0.23	SHEET 1 OF 2
DO NOT SCALE DRAWING							

5

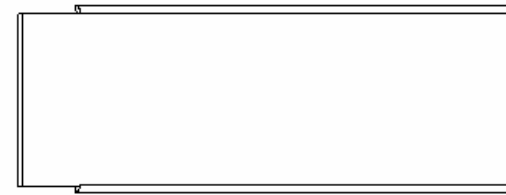
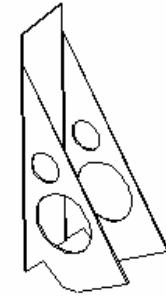
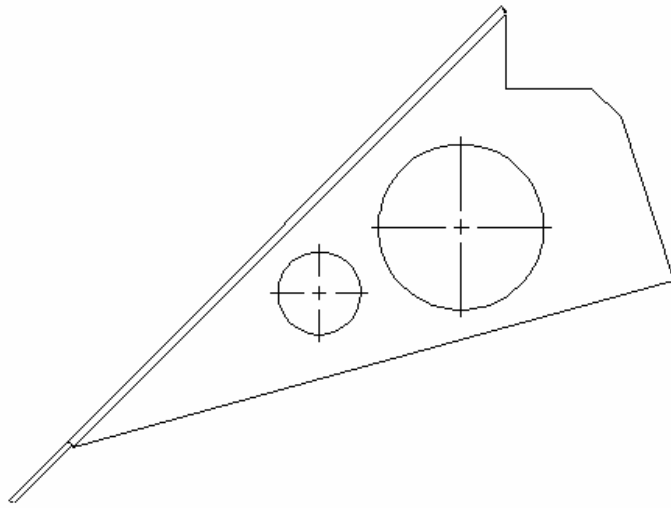
4

4

3

2

1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPIA AND ST PALL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPIA AND ST PALL TRAVELERS IS PROHIBITED.

UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE			
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: CAMERA MAST		
TOLERANCES:		CHECKED					
FRACTIONAL: $\pm 1/32$							
ANGULAR: ± 3							
TWO PLACE DECIMAL ± 0.01					SIZE	DWG. NO.	REV
THREE PLACE DECIMAL ± 0.005		COMMENTS:			A	C07	1
MATERIAL ALUMINUM					SCALE: 1:2	WEIGHT: 0.24	SHEET 1 OF 2
FINISH							
D= NOT SCALE DRAWING							

5

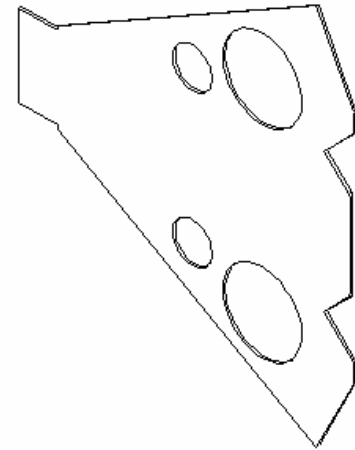
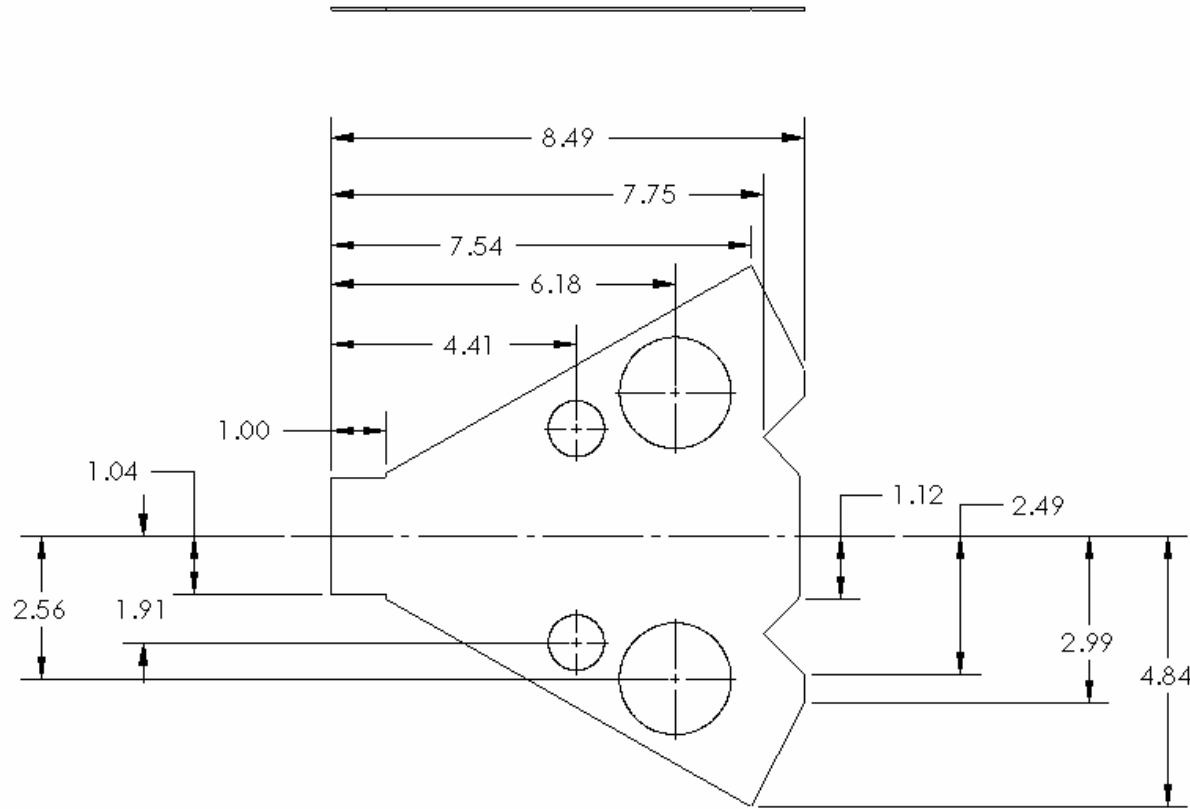
4

4

3

2

1



UNLESS OTHERWISE SPECIFIED:	NAME	DATE
DIMENSIONS ARE IN INCHES	NCM	4/24/07
TOLERANCES:		
FRACTIONAL $\pm 1/32$		
ANGULAR ± 3		
TWO PLACE DECIMAL ± 0.01		
THREE PLACE DECIMAL ± 0.005		

WORCESTER POLYTECHNIC INSTITUTE

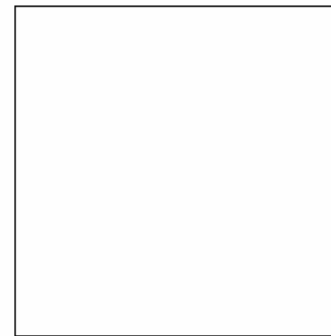
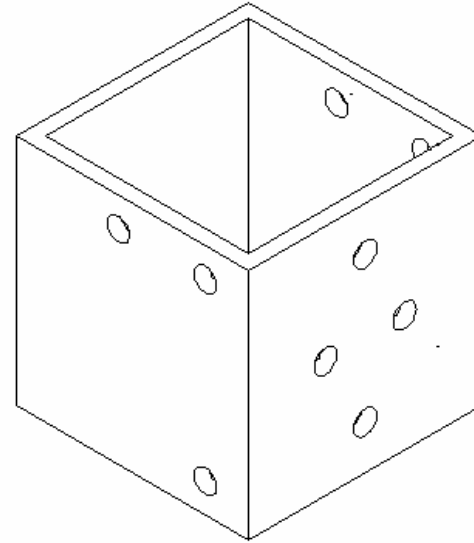
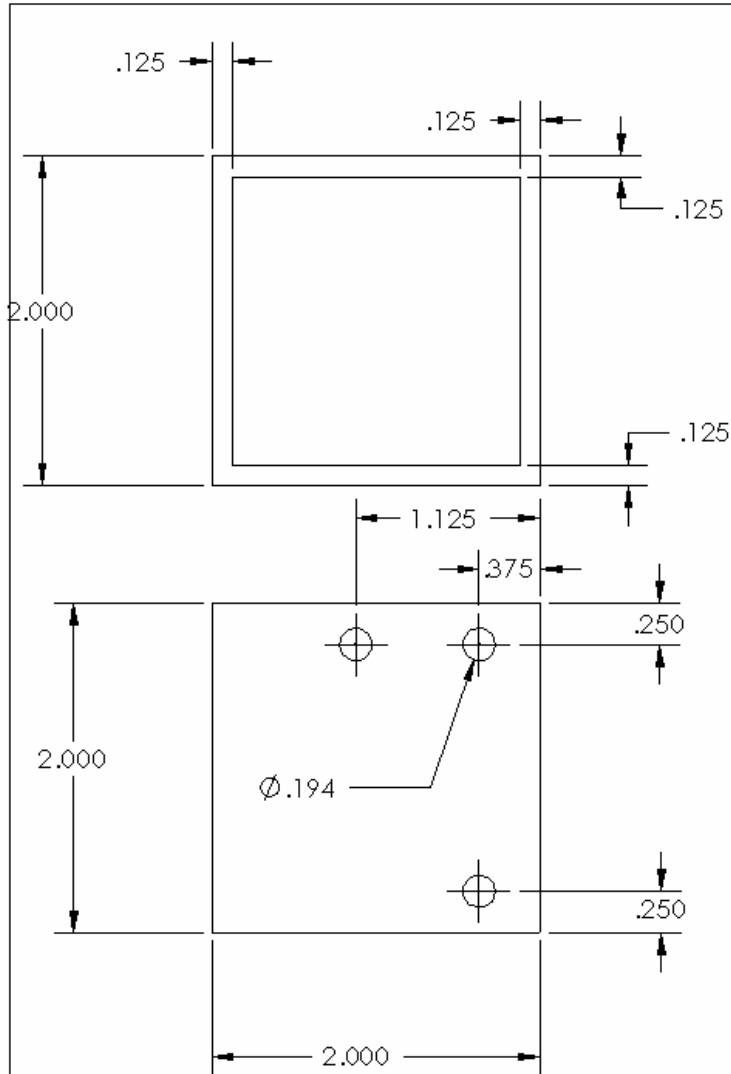
TITLE:
CAMERA MAST

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPPIA. NO PART SHALL BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF WPPIA. NO PART SHALL BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF WPPIA. NO PART SHALL BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE WRITTEN PERMISSION OF WPPIA.

MATERIAL	COMMENTS:
ALUMINUM	
FINISH	
DO NOT SCALE DRAWING	

SIZE	DWG. NO.	REV
A	C07	1
SCALE: 1:3	WEIGHT: 0.24	SHEET 2 OF 2

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WPI AND ST PALL TRAVELERS. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WPI AND ST PALL TRAVELERS IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL ± 1/32
 ANGULAR: ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL ALUMINUM

FINISH

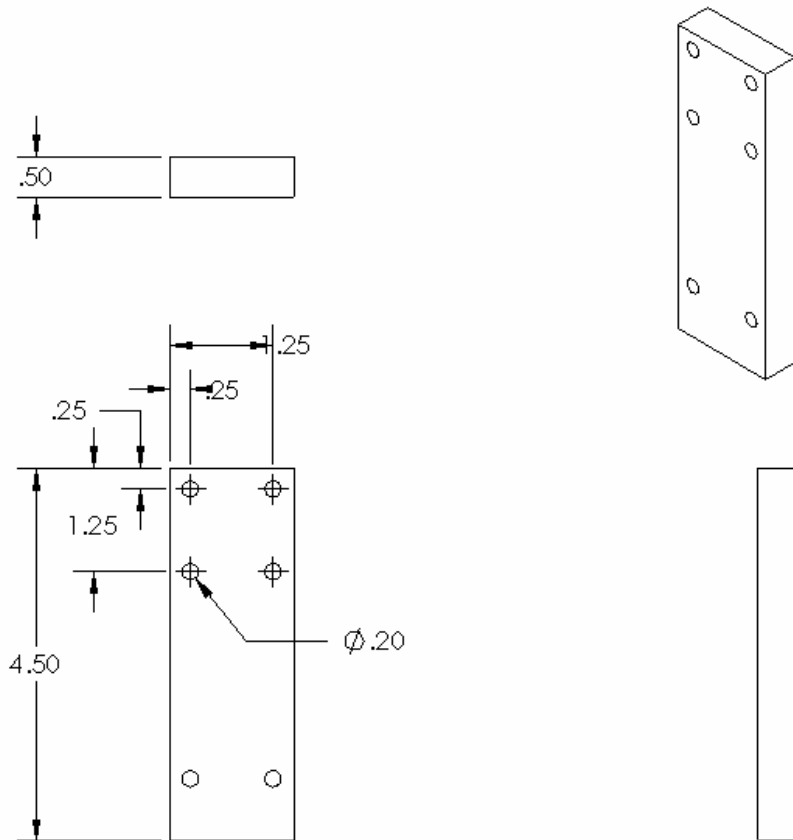
DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	SFF	4/24/07
CHECKED		

COMMENTS:

WORCESTER POLYTECHNIC INSTITUTE		
TITLE: JOINT MOUNTING BLOCK		
SIZE A	DWG. NO. J01	REV 1
SCALE: 1:1	WEIGHT: 0.07 LB	SHEET 1 OF 1

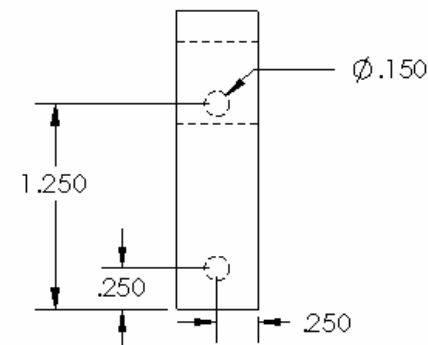
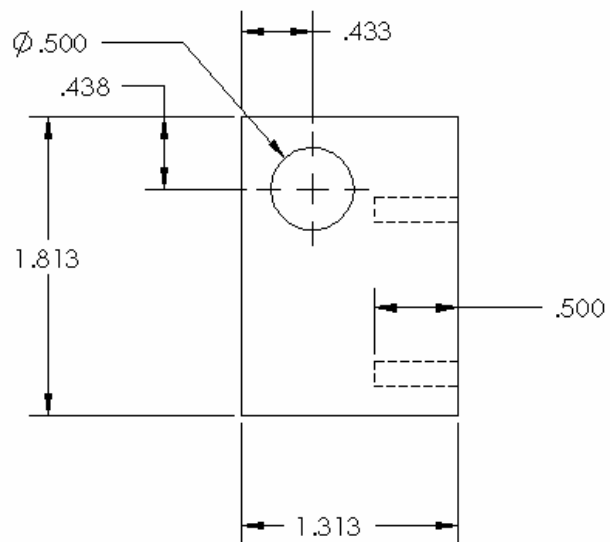
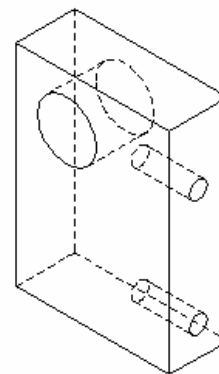
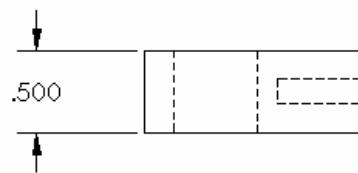
5 4 3 2 1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	SFF	4/24/07	TITLE: JOINT SIDE BLOCK
TOLERANCES:		CHECKED			
FRACTIONAL $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL ± 0.01					SIZE DWG. NO. REV
THREE PLACE DECIMAL ± 0.005					A J02 1
MATERIAL POLYCARBONATE		COMMENTS:			SCALE: 1:2 WEIGHT: 0.14 SHEET 1 OF 1
FINISH					
DO NOT SCALE DRAWING					

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPI AND ST. PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPI AND ST. PAUL TRAVELERS IS PROHIBITED.

5 4 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE		
DIMENSIONS ARE IN INCHES	DRAWN	SFF	4/24/07	TITLE:		
TOLERANCES:	CHECKED			JOINT BOTTOM BLOCK		
FRACTIONAL $\pm 1/32$				SIZE DWG. NO. REV		
ANGULAR ± 3				A	J03	1
TWO PLACE DECIMAL ± 0.01				SCALE: 1:1	WEIGHT: 0.05 LBS	SHEET 1 OF 1
THREE PLACE DECIMAL ± 0.005	COMMENTS:					
MATERIAL POLYCARBONATE						
FINISH						
DO NOT SCALE DRAWING						

5

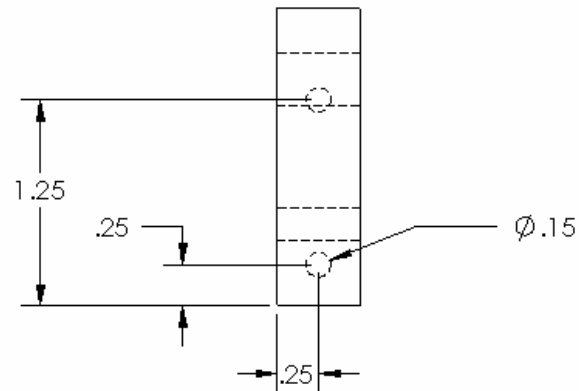
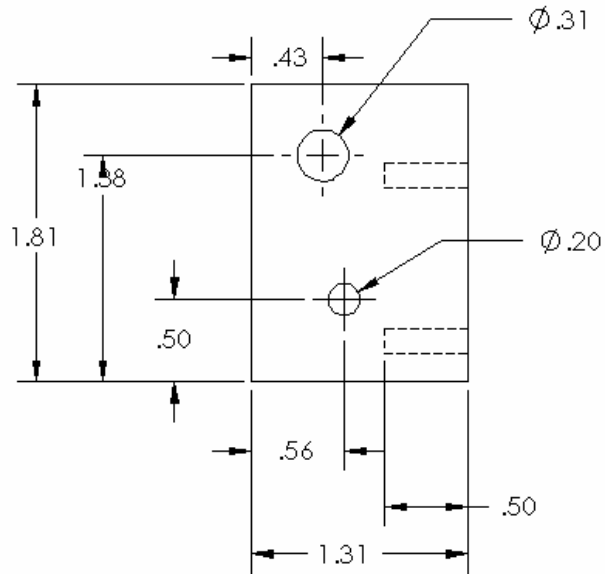
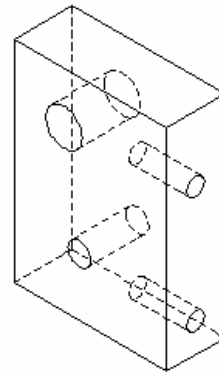
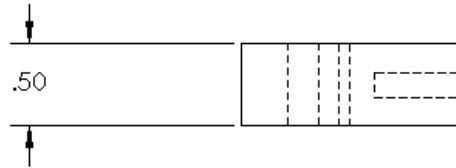
4

4

3

2

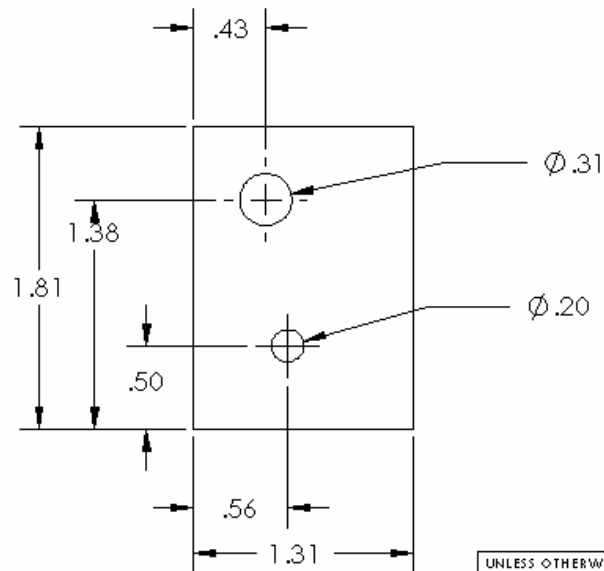
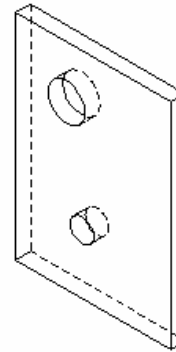
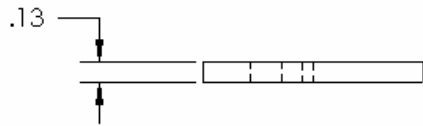
1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES	DRAWN	SFF	4/24/07	TITLE:	
TOLERANCES:	CHECKED			JOINT BOTTOM BLOCK	
FRACTIONAL: $\pm 1/32$				SIZE DWG. NO. REV	
ANGULAR: ± 3				A J04 1	
TWO PLACE DECIMAL: ± 0.01				SCALE: 1:1 WEIGHT: 0.05 LBS SHEET 1 OF 1	
THREE PLACE DECIMAL: ± 0.005					
	COMMENTS:				
	MATERIAL				
	POLYCARBONATE				
	FINISH				
	DO NOT SCALE DRAWING				

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WORCESTER POLYTECHNIC INSTITUTE. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WORCESTER POLYTECHNIC INSTITUTE IS PROHIBITED.

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR: ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL
 POLYCARBONATE

FINISH

DO NOT SCALE DRAWING

UNLESS OTHERWISE SPECIFIED:	NAME	DATE
DRAWN	SFF	4/24/07
CHECKED		

COMMENTS:

WORCESTER POLYTECHNIC INSTITUTE		
TITLE: JOINT TOP PLATE		
SIZE A	DWG. NO. J05	REV 1
SCALE: 1:1	WEIGHT: 0.01 LBS	SHEET 1 OF 1

5

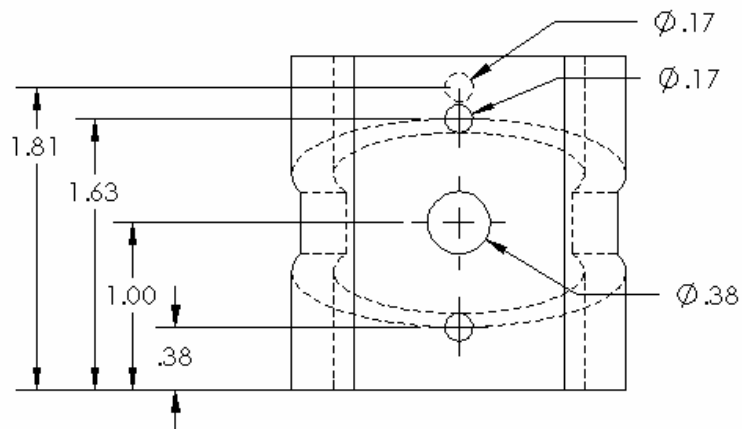
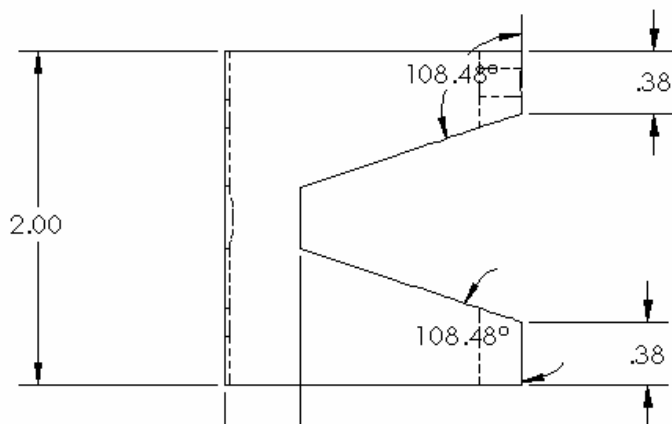
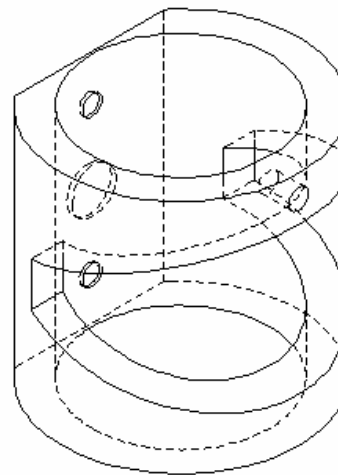
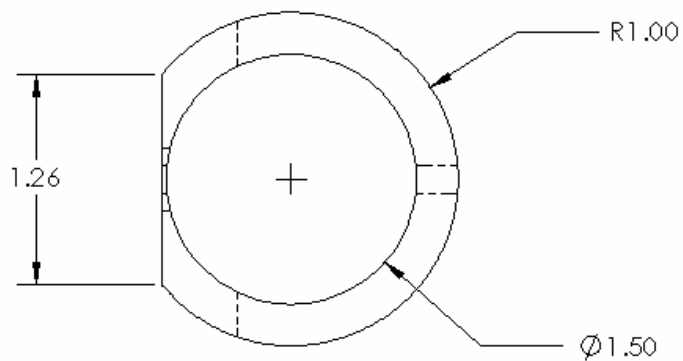
4

4

3

2

1



.46

PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WP&A AND ST PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WP&A AND ST PAUL TRAVELERS IS PROHIBITED.

UNLESS OTHERWISE SPECIFIED:

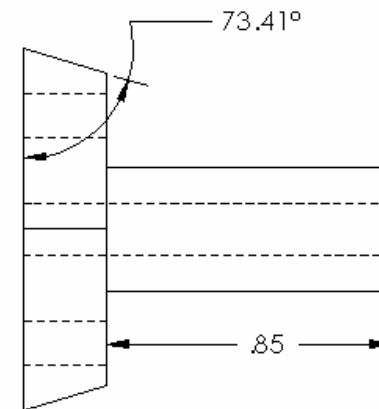
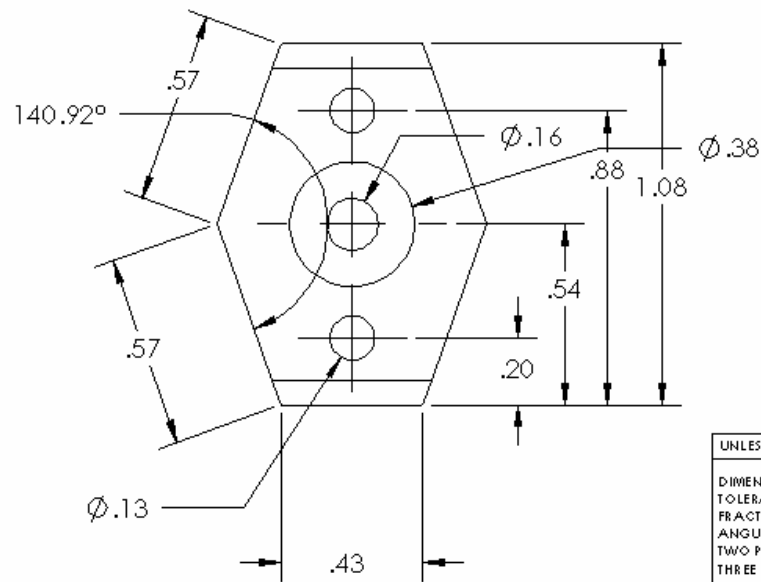
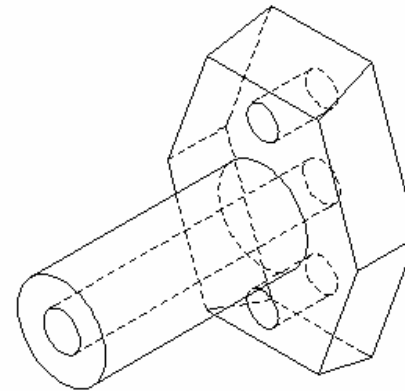
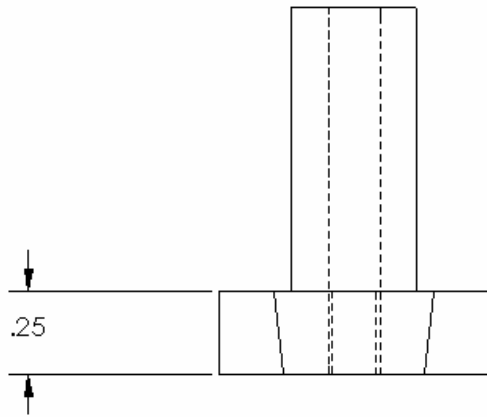
DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL: 1/32
 ANGULAR: ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL
ALUMINUM
 FINISH
 DO NOT SCALE DRAWING

NAME	DATE
SFF	4/24/07
CHECKED	
COMMENTS:	

WORCESTER POLYTECHNIC INSTITUTE		
TITLE: JOINT OUTER HOUSING		
SIZE A	DWG. NO. JJ01	REV 1
SCALE: 1:1	WEIGHT: 0.16 LBS SHEET 1 OF 1	

5 4 3 2 1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:

DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL
 STEEL

FINISH
 DO NOT SCALE DRAWING

NAME	DATE
DRAWN SFF	4/24/07
CHECKED	

COMMENTS:

WORCESTER POLYTECHNIC INSTITUTE		
TITLE: JOINT TOGGLE		
SIZE A	DWG. NO. JJ02	REV 1
SCALE: 2:1	WEIGHT: 0.06 LBS SHEET 1 OF 1	

5

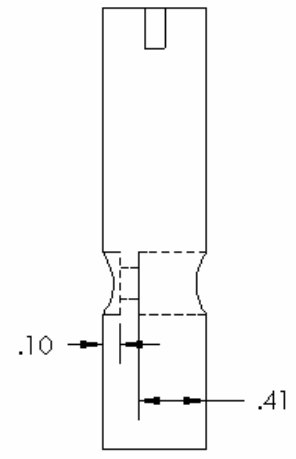
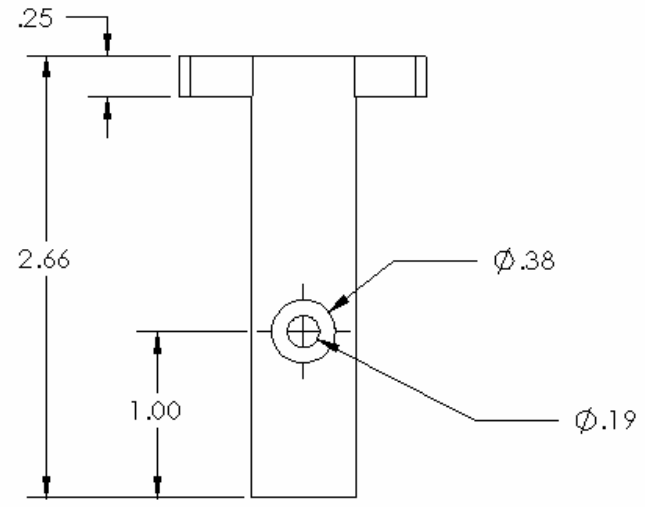
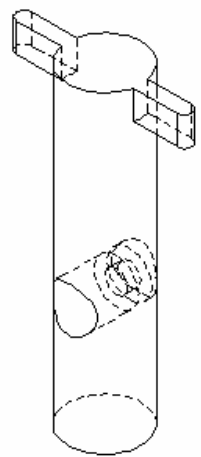
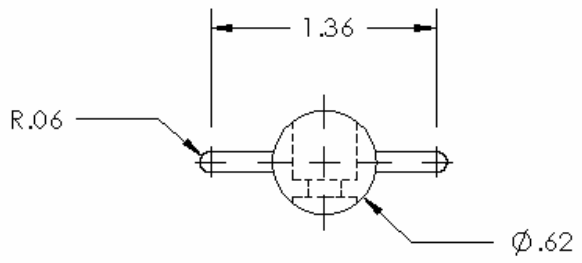
4

4

3

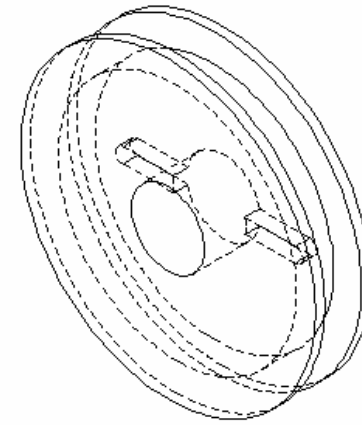
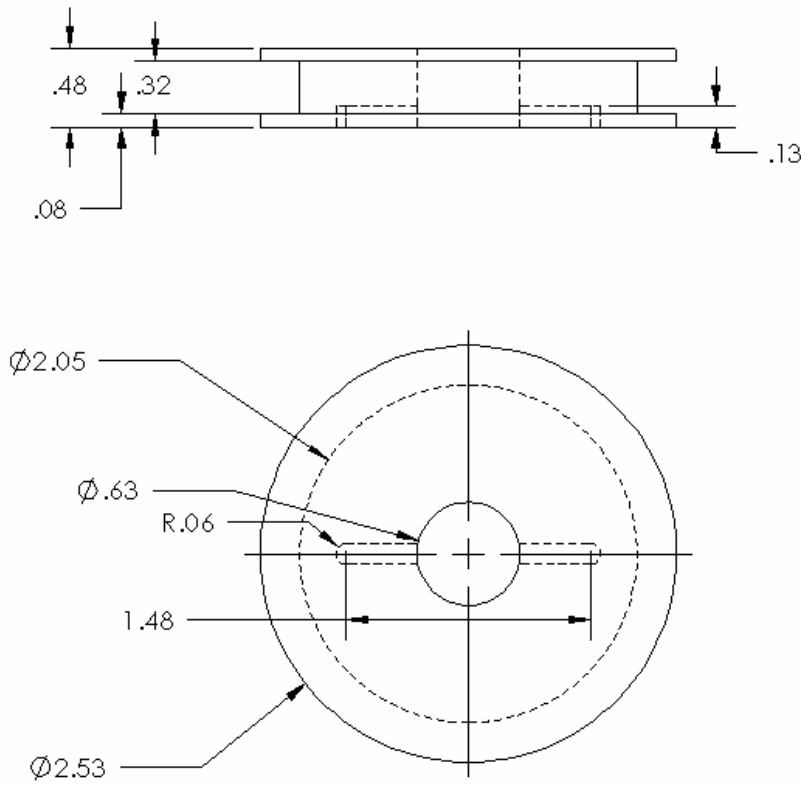
2

1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE		
DIMENSIONS ARE IN INCHES		DRAWN	SFF	4/25/07	TITLE: JOINT VERTICAL SHAFT	
TOLERANCES:		CHECKED				
FRACTIONAL: ± 1/32						
ANGULAR: ± 3						
TWO PLACE DECIMAL ±0.01					SIZE DWG. NO. REV	
THREE PLACE DECIMAL ±0.005					A JJ03 1	
<p>PROPRIETARY AND CONFIDENTIAL</p> <p>THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPJ AND ST PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPJ AND ST PAUL TRAVELERS IS PROHIBITED.</p>		COMMENTS:		SCALE: 1:1 WEIGHT: 0.22 LBS SHEET 1 OF 1		
		MATERIAL	STEEL			
		FINISH				
		DO NOT SCALE DRAWING				

5 4 3 2 1



UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	SFF	4/25/07	TITLE: JOINT CAPSTAN
TOLERANCES:		CHECKED			
FRACTIONAL $\pm 1/32$					
ANGULAR: ± 3					
TWO PLACE DECIMAL ± 0.01					SIZE DWG. NO.
THREE PLACE DECIMAL ± 0.005					A JJ04
MATERIAL POLYCARBONATE		COMMENTS:		REV	1
FINISH				SCALE: 1:1 WEIGHT: 0.07 LBSSHEET 1 OF 1	
DO NOT SCALE DRAWING					

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF WPI AND ST PAUL TRAVELERS. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF WPI AND ST PAUL TRAVELERS IS PROHIBITED.

5

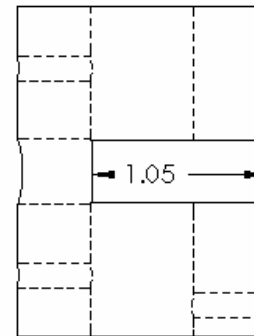
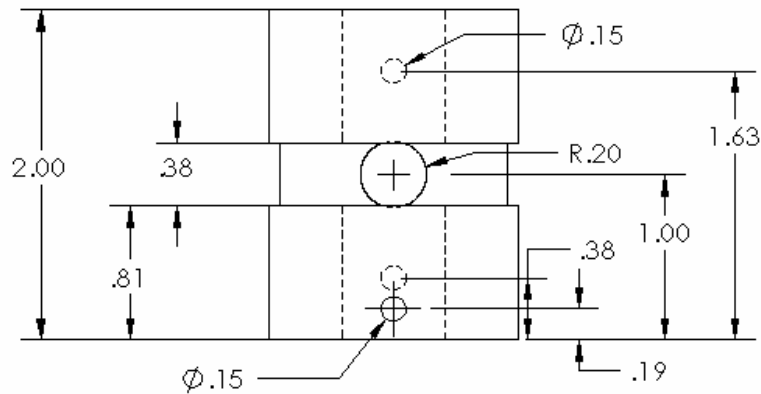
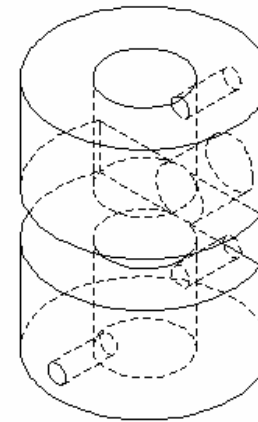
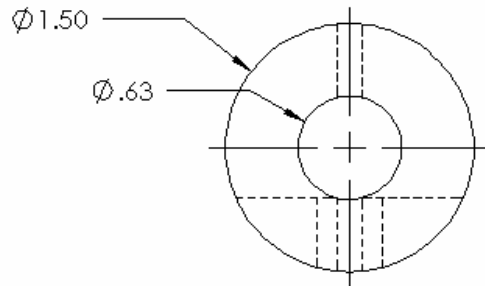
↑

4

3

2

1



PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WORCESTER POLYTECHNIC INSTITUTE. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WORCESTER POLYTECHNIC INSTITUTE IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:
 DIMENSIONS ARE IN INCHES
 TOLERANCES:
 FRACTIONAL $\pm 1/32$
 ANGULAR ± 3
 TWO PLACE DECIMAL ± 0.01
 THREE PLACE DECIMAL ± 0.005

MATERIAL ALUMINUM

FINISH

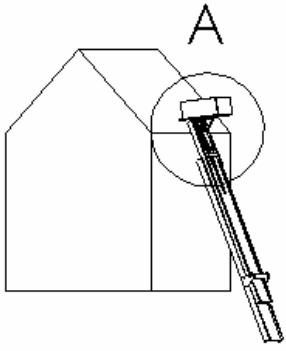
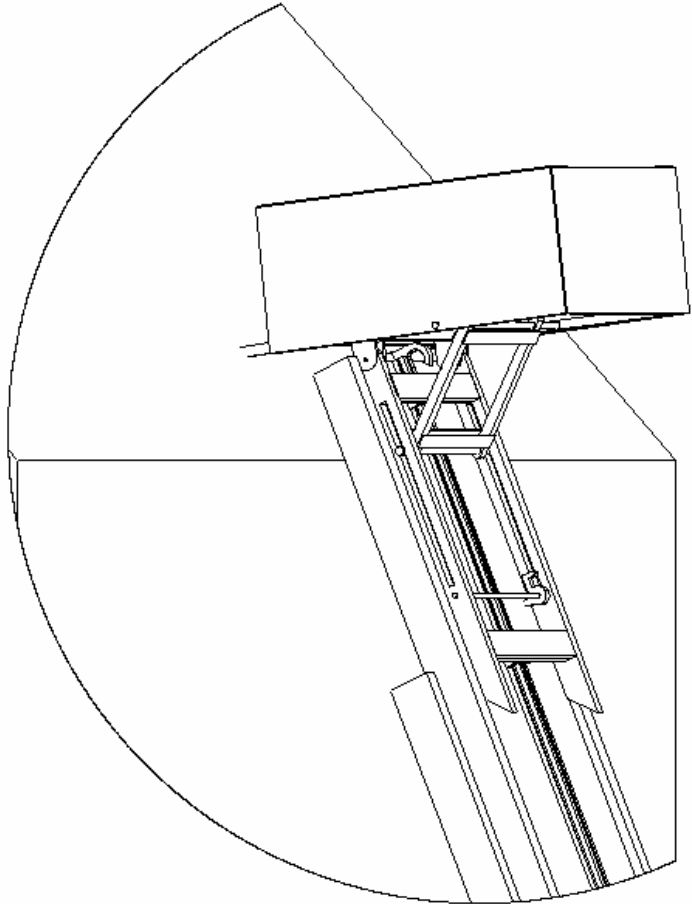
DO NOT SCALE DRAWING

	NAME	DATE
DRAWN	SFF	4/25/07
CHECKED		

COMMENTS:

WORCESTER POLYTECHNIC INSTITUTE		
TITLE: JOINT INNER HOUSING		
SIZE A	DWG. NO. JJ05	REV 1
SCALE: 1:1 WEIGHT: 0.24 LBS SHEET 1 OF 1		

5 4 3 2 1

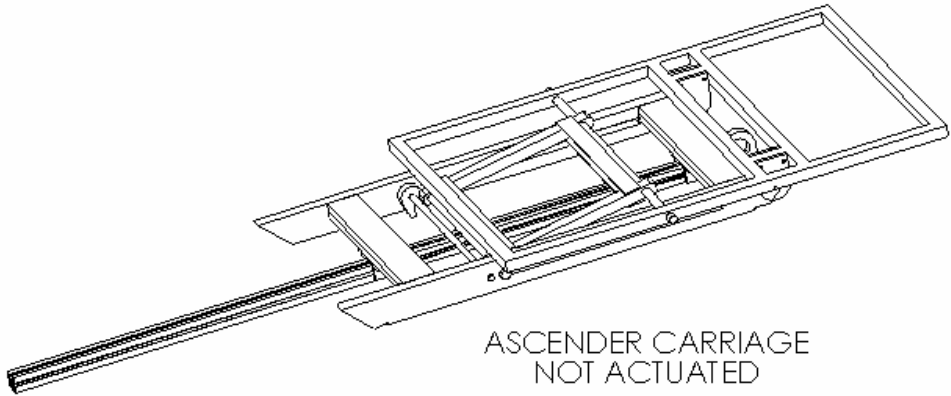


DETAIL A
SCALE 1:12

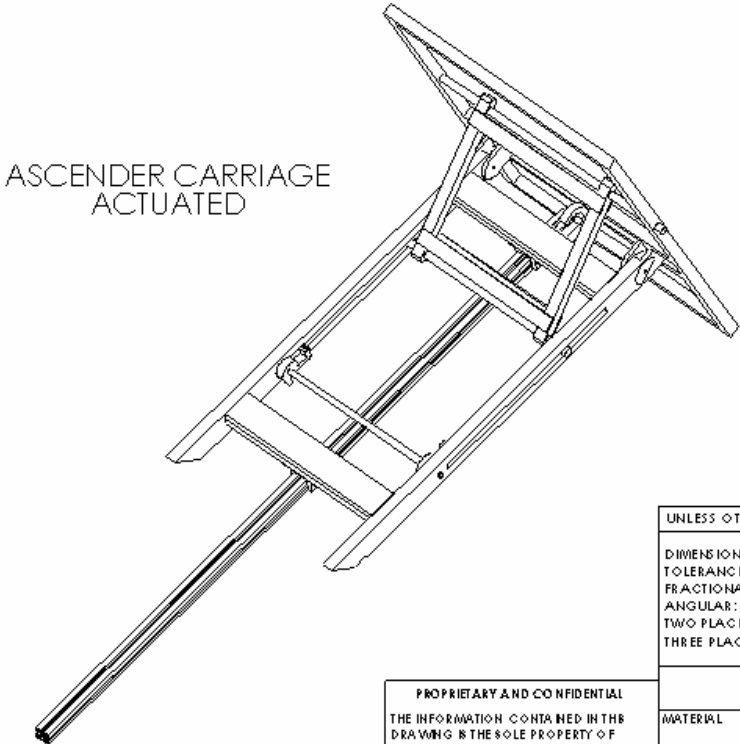
PROPRIETARY AND CONFIDENTIAL
 THE INFORMATION CONTAINED IN THIS
 DRAWING IS THE SOLE PROPERTY OF
 WPITA AND ST PALL TRAVELERS. ANY
 REPRODUCTION IN PART OR AS A WHOLE
 WITHOUT THE WRITTEN PERMISSION OF
 WPITA AND ST PALL TRAVELERS IS
 PROHIBITED.

UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: ASCENDER MODEL WITH ROOF
TOLERANCES:		CHECKED			
FRACTIONAL $\pm 1/32$					
ANGULAR ± 3					
TWO PLACE DECIMAL ± 0.01					SIZE DWG. NO. REV A ASCENDERMODEL 1
THREE PLACE DECIMAL ± 0.005					
		COMMENTS:		SCALE: 1:100 WEIGHT: 1 SHEET 1 OF 1	
MATERIAL		SEE CD FOR ASCENDER PART FILES			
FINISH					
D= NOT SCALE DRAWING					

5 4 3 2 1



ASCENDER CARRIAGE
NOT ACTUATED



ASCENDER CARRIAGE
ACTUATED

UNLESS OTHERWISE SPECIFIED:		NAME	DATE	WORCESTER POLYTECHNIC INSTITUTE	
DIMENSIONS ARE IN INCHES		DRAWN	NCM	4/24/07	TITLE: ASCENDER CARRIAGE
TOLERANCES:		CHECKED			
FRACTIONAL: 1/32					
ANGULAR: ±3					
TWO PLACE DECIMAL ±0.01					SIZE DWG. NO. REV A ASCENDERCARRIAGE 1
THREE PLACE DECIMAL ±0.005					
MATERIAL		COMMENTS:			SCALE: 1:10 WEIGHT: 1 SHEET 1 OF 1
FINISH					
DO NOT SCALE DRAWING					

PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS
DRAWING IS THE SOLE PROPERTY OF
WPIA HD ST PAUL TRAVELERS. ANY
REPRODUCTION IN PART OR AS A WHOLE
WITHOUT THE WRITTEN PERMISSION OF
WPIA HD ST PAUL TRAVELERS IS
PROHIBITED.

5 4 3 2 1

Appendix B- Robot Code

```
// runRobot.c : implementation file
#include "API.h" //this is the API to access all the Vex's sensors/motors/etc.
#include "math.h"

//Global Variables
unsigned char servoNeutral = 127;

//*****
//variables for Wheel feedback

float error;
float correction;

int target[4];
int output[4];

int oldCount1[4];
int oldCount2[4];
int newCount[4];
int numClicks;

float expectedClicksPerSec[4];
float expectedTimeBetweenClicks;
unsigned long oldClickTime1[4];
unsigned long oldClickTime2[4];
unsigned long newClickTime[4];
long timeInterval;
unsigned long stuckWaiting[4];

long delay[4];

float clicksPerSec[4];
float clicksPerSec1[4];
float clicksPerSec2[4];
//*****
//variables for geometry
float halfLength = 9 ;
float halfWidth = 9 ;
//*****
//variables to equate motorspeeds
float rateOfTurn = .05;
float P = 50 ;
float P2 = 30 ;

//*****
int panTiltDelay = 10 ; //this is a delay - higher rate is slower pan
int panTiltDelayCount = 0;
//*****
```

```

//variables for Joint feedback
float potNeutral = 512;
float potAngle;
float halfTangent;
float rightToLeftRatio;
float slowSide ;
float fastSide ;
//*****
//PID "Goal Values"
float targetAngle;
char I = 0;
int D ;
unsigned char targetAngPWM;
int targetFL; //Target speed for the front left motor
int targetBL;
int targetFR;
int targetBR;
unsigned char targetPan = 127 ;
unsigned char targetTilt = 127 ;
//*****
//PID "Feedback Values"
unsigned int potReading;
//*****
unsigned char inputFB; //forward and reverse joystick input
unsigned char inputLR;
unsigned char inputPan;
unsigned char inputTilt;
float joystickInRadians ;
unsigned char inputOverRide;
//*****

//*****

void runRobot(void);
void init(void);
void getInputs(void);
void getFeedback(void);

void updateDisplay(void);

void updateTargets(void);
void setTargetsForTurning(void);
void setTargetsForDriving(void);
void setTargetsForPanTilt(void);

void calcSpeed(int i);
void adjustForError(int i);

void drivePanTilt(void);
void driveMotors(void);
void lockOut(void);

```

```

void runRobot(void)
{
    int i;

    init();

    while ( 1 )
    {

        getInputs();

        updateTargets();

        getFeedback();

        target[0] = targetFL;
        target[1] = targetFR;
        target[2] = targetBL;
        target[3] = targetBR;

        calcSpeed(0);
        calcSpeed(3);
        calcSpeed(1);
        calcSpeed(2);

        for( i = 2; i <=5 ; i++)
        {
            if (GetAnalogInput(i) < 85 && inputOverRide != 255)
            {
                lockOut();
                PrintToScreen ( "%d\n" , (int)i ) ;
                PrintToScreen ( "%d\n" , (int)GetAnalogInput(i) ) ;           // this is leftover from debugging
                Wait(50);
            }
        }

        driveMotors();
        drivePanTilt();
    }
}

//*****
//          initialization sequence
void init(void)
{
    int i;

    getFeedback();
    targetAngle = potAngle ;           // this prevents the joint from spazzing on startup
    D = potReading ;
    panTiltDelayCount = 0;
}

```

```

for( i = 0 ; i <= 3 ; i++)
{
    PresetEncoder(i+1,0);
    StartEncoder(i+1);

    PresetTimer(i+1,0);
    StartTimer(i+1);
}
}

//*****
//          update inputs
void getInputs(void)
{
    inputLR = GetRxInput ( 0 , 1 ) ; //joystick axis 1 controls Left/Right
    inputFB = GetRxInput ( 0 , 2 ) ; //joystick axis 4 controls Front/Back
    inputTilt = GetRxInput ( 0 , 3 ) ;
    inputPan = GetRxInput ( 0 , 4 ) ;
    inputOverRide = GetRxInput ( 0 , 5 );
}

//*****
//          get feedback
void getFeedback(void)
{

    int countDiff1;
    int countDiff2;

    long timeDiff1;
    long timeDiff2;

    char i;

//*****
//joint feedback
//*****
    potReading = GetAnalogInput ( 1 ) ;
    potAngle = ((float)potReading - potNeutral) /508 ;
    if((potAngle < .03) && (potAngle > -.03)) //i'm trying to avoid a divide
        halfTangent = .0001;           //by zero later on. .03 radians is about 1.7degrees
    else
        halfTangent = tan(potAngle/2);
}
/*
Explanation of the number "508" above:
I took readings off the pot while I turned the knob.
It would seem that about pi/2 radians of rotation corresponds
to about 400 "bits" on the analog input range. hence,
pi/2 radians = 400 bits / 508
}
*/

```

```

//*****
//wheel feedback
//*****
for ( i = 0 ; i <= 3 ; i++ )
{

    newCount[i] = GetEncoder(i+1);           //this section of code prevents the encoder
    newClickTime[i] = GetTimer(i+1);        //counters from overflowing

    if (newCount[i] > 32000)
    {
        countDiff1 = newCount[i] - oldCount1[i];
        countDiff2 = oldCount2[i] - oldCount1[i];

        oldCount2[i] = 0;
        oldCount1[i] = countDiff2 ;
        newCount[i] = countDiff1 + countDiff2;

        PresetEncoder(i+1,newCount[i]);
    }

    if (newClickTime[i] > 1000000000)       //this prevents the timers from overflowing
    {
        timeDiff1 = newClickTime[i] - oldClickTime1[i];
        timeDiff2 = oldClickTime1[i] - oldClickTime2[i];

        oldClickTime2[i] = 0;
        oldClickTime1[i] = timeDiff2 ;
        newClickTime[i] = timeDiff1 + timeDiff2;

        stuckWaiting[i] = newClickTime[i];

        PresetTimer(i+1,newClickTime[i]);
    }
}

void updateDisplay(void)                   //this section of code is for debugging
{
    PrintToScreen ( "%d\n " , (int)slowSide ) ;
    PrintToScreen ( "%d\n " , (int)fastSide ) ;
    // PrintToScreen ( "%d\n " , (int)(potAngle*57.) ) ;
    PrintToScreen ( "%d\n " , (int)(rightToLeftRatio*1000) ) ;
    PrintToScreen ( "%d\n " , (int)targetFL ) ;
    PrintToScreen ( "%d\n " , (int)targetFR ) ;
    PrintToScreen ( "%d\n " , (int)targetBL ) ;
    PrintToScreen ( "%d\n " , (int)targetBR ) ;
    PrintToScreen ( "\n" ) ;
    // Wait ( 100 ) ;
}

//*****
//      set targets based on input
void updateTargets(void)

```

```

{
// in this control scheme, the LR joystick position should correspond
// directly to the position of the joint motor

    joystickInRadians = (float)((servoNeutral - inputLR))/ 175 ;

    rightToLeftRatio = (halfWidth + halfWidth*halfTangent)/(halfWidth - halfWidth*halfTangent);

/* the above is a confusing magic conversion, so that a full throttle
joystick position should correspond to about .785 radians, or a full,
45 degree turn*/

    setTargetsForTurning();

//during development, I found that while driving, with the wheels already spinning, the joint motor could
// use brute force to turn the robot. So I set targets for turning, which would normally cause the wheels to
// spin in opposing directions, but then I override those targets to set the wheels to the normal driving speed.

//a consequence of this is that an explicit signal to stop the robot isn't clearly written into the code
//it is buried in an if statement in the setTargetsForTurning() subroutine

    if ((inputFB - servoNeutral) < -15 || (inputFB - servoNeutral)> 15)
        setTargetsForDriving();

    setTargetsForPanTilt();

    if (panTiltDelayCount < panTiltDelay) // I did this to slow the pan tilt down
    {
        if ((inputPan - servoNeutral) < -15 || (inputPan - servoNeutral)> 15) //set deadbands

            if ((inputTilt - servoNeutral) < -15 || (inputTilt - servoNeutral)> 15)

                panTiltDelayCount++;
    }
    else
        panTiltDelayCount = 0;

}
//*****
//          set DRIVING TARGETS
void setTargetsForDriving(void)
{
    //float rightToLeftRatio ;
    float maxPower = 50. ;
    float powerScaleToJoystick ;

    if (rightToLeftRatio > 1 || rightToLeftRatio < -1)
        powerScaleToJoystick = maxPower / rightToLeftRatio ; //dynamically scales the throttle range
    else
        powerScaleToJoystick = maxPower * rightToLeftRatio ; //dynamically scales the throttle range
}

```

```

slowSide = ((float)(inputFB - servoNeutral) / 128) * powerScaleToJoystick ;

// the code below should keep it out of the deadband
// it assumes that the maxPower setting will keep it from maxing out

if (slowSide > -15 && slowSide < 0)
    slowSide = -15 ;
if (slowSide >= 0 && slowSide < 15)
    slowSide = 15 ;

if (rightToLeftRatio > 1 || rightToLeftRatio < -1)
{
    fastSide = slowSide * rightToLeftRatio ;
    targetFR = (int)(slowSide);
    targetBR = (int)(slowSide);
    targetFL = (int)(fastSide);
    targetBL = (int)(fastSide);
}
else
{
    fastSide = slowSide / rightToLeftRatio ;
    targetFR = (int)(fastSide);
    targetBR = (int)(fastSide);
    targetFL = (int)(slowSide);
    targetBL = (int)(slowSide);
}

}

//*****
//          set TURNING targets
void setTargetsForTurning(void)
{
    targetAngle = joystickInRadians ;

    if((potAngle - targetAngle) > -.037 && (potAngle - targetAngle) < .037)
    {
        l = 0;
        slowSide = 0;
    }
    else
    {
        if ( ( D - (int)potReading) > -5 && ( D - (int)potReading) < 5
            && (-20 < l < 20) )
        {
            if (potAngle < targetAngle)
            {
                slowSide = -12;
                if (l > -12)
                    l = -12 ;
            }
            else
                l--;
        }
    }
}

```

```

        if (potAngle > targetAngle)
        {
            slowSide = 12;
            if (l < 12)
                l = 12;
            else
                l++;
        }
    }
}

D = potReading ;
targetAngPWM = (unsigned char)(servoNeutral + l);

if (rightToLeftRatio > 1 || rightToLeftRatio < -1)
{
    fastSide = slowSide * rightToLeftRatio ;
    targetFR = (int)(slowSide);
    targetBR = (int)(-slowSide);
    targetFL = (int)(-fastSide);
    targetBL = (int)(fastSide);
}
else
{
    fastSide = slowSide / rightToLeftRatio ;
    targetFR = (int)(slowSide);
    targetBR = (int)(-slowSide);
    targetFL = (int)(-fastSide);
    targetBL = (int)(fastSide);
}
}

//*****
//          set the PAN/TILT
void setTargetsForPanTilt(void)
{
    int turnOffset;
    int scaledPan;
    float scale;

    if (panTiltDelayCount < panTiltDelay)
    {
        if ((inputTilt - servoNeutral < -15) && targetTilt > 1)
            targetTilt--;
        if ((inputTilt - servoNeutral > 15) && targetTilt < 255)
            targetTilt++;

        panTiltDelayCount++;
    }
    else
        panTiltDelayCount = 0;
}

```



```

turnOffset = (servoNeutral - inputLR) / 1.5;

if ((inputPan - servoNeutral) > 0)
    scale = (float)(127 - turnOffset) / 127.;
if ((inputPan - servoNeutral) <=0)
    scale = (float)(turnOffset - (-127))/127;

scaledPan = (inputPan - servoNeutral) * scale ;
targetPan = (unsigned char)(scaledPan + turnOffset + servoNeutral);

}
//*****
//          calculate Current Speed

void calcSpeed(int i)
{
    expectedClicksPerSec[i] = (float)(target[i])* .85;
    if (target[i] < 0)
        expectedClicksPerSec[i] = -1 * expectedClicksPerSec[i];

    expectedTimeBetweenClicks = (1200 / expectedClicksPerSec[i]) ;

    if (expectedTimeBetweenClicks > 120)
        expectedTimeBetweenClicks = 120;

    // the above should give you the time between clicks...
    // there is a little extra leeway to account for rounding error
    // and stuff like that

    if(newCount[i] > oldCount1[i])
    {
        timeInterval = newClickTime[i] - oldClickTime2[i] ;
        numClicks = newCount[i] - oldCount2[i];

        oldClickTime2[i] = oldClickTime1[i];
        oldClickTime1[i] = newClickTime[i] ;

        stuckWaiting[i] = newClickTime[i] ;      //a click has occurred,

        oldCount2[i] = oldCount1[i] ;          //so reset everything
        oldCount1[i] = newCount[i] ;

        clicksPerSec[i] = 1000 / (float)(timeInterval / numClicks);

        adjustForError(i);
    }

    if ((newClickTime[i] - oldClickTime1[i]) > expectedTimeBetweenClicks)
    {
        clicksPerSec[i] = 0;
    }
}

```

```

if ((newClickTime[i] - stuckWaiting[i]) > expectedTimeBetweenClicks)
{
    stuckWaiting[i] = newClickTime[i];
    delay[i] = 2;
    adjustForError(i);
}
}

//*****
//          adjust output for error
void adjustForError(int i)
{
    int Pwheel;
    int speedBehind;
    float percentError;
    char comp;

    delay[i]++;

    if(delay[i] < 2)
        return;
    else
        delay[i] = 0;

    if( i == 0)                //figures out which wheel is on the same side, (left and right sides)
        comp = 2;
    if( i == 1)
        comp = 3;
    if( i == 2)
        comp = 0;
    if( i == 3)
        comp = 1;

    error = clicksPerSec[i] - expectedClicksPerSec[i] ;
    percentError = error / expectedClicksPerSec[i] ;
    speedBehind = (int)(clicksPerSec[i] - clicksPerSec[comp]) ;

    if (percentError > .1)
        Pwheel = (-percentError * 2) - 1;
    else if (percentError < -.1)
        Pwheel = (-percentError * 2) + 1;
    else
        Pwheel = 0;

    if (percentError < -.1 &&
        speedBehind < -12 )
        Pwheel = Pwheel + 10;

    if (target[i] > 0)
        correction = Pwheel ;
    else if (target[i] < 0)
        correction = -Pwheel ;
}

```

```

output[i] = output[i] + correction;

//*****
//dead zone
if (target[i] == 0 )
    output[i] = 0 ;

if (target[i] > 0 && output[i] < 12)
    output[i] = 12;
if (target[i] < 0 && output[i] > -12)
    output[i] = -12;

//*****
//dont max out
if (output[i] > 127)
    output[i] = 127 ;
if (output[i] < -127)
    output[i] = -127;

}

void driveMotors(void)
{

    SetPWM ( 1 , (unsigned char)(output[0] + servoNeutral ) ) ;
    SetPWM ( 2 , (unsigned char)(output[1] + servoNeutral ) ) ;
    SetPWM ( 3 , (unsigned char)(output[2] + servoNeutral ) ) ;
    SetPWM ( 4 , (unsigned char)(output[3] + servoNeutral ) ) ;
    SetPWM ( 5 , targetAngPWM ) ;
}

void drivePanTilt(void)
{
    SetPWM ( 6 , targetTilt ) ;
    SetPWM ( 7 , targetPan ) ;
}

void lockOut(void)
{

    output[0] = 0;
    output[1] = 0;
    output[2] = 0;
    output[3] = 0;
    targetAngPWM = 127;
}

```

Appendix C- Weight Breakdown

Bill of Materials			
Item	Quantity	Weight (lbs)	Total (lbs)
Wheel Assembly	4	1.3	5.2
Window Drive Motors	4	1.2	4.8
Batteries	3	1.3	3.9
Misc Wires and Hardware	1	2.9	2.9
Bottom Panel	2	0.7	1.4
Side Panel	4	0.33	1.32
Victor Speed Controllers	5	0.25	1.25
Joint Drive Motor	1	1	1
X10 Wireless Camera	1	1	1
Vex Power Pack	1	0.71	0.71
Bracket	2	0.23	0.46
Vex Controller	1	0.28	0.28
Pan Tilt Assembly	1	0.25	0.25
Camera Mast	1	0.24	0.24
Joint Inner Housing	1	0.24	0.24
Joint Vertical Shaft	1	0.22	0.22
Joint Outer Housing	1	0.16	0.16
Joint Side Block	1	0.14	0.14
Joint Block	2	0.07	0.14
Joint Capstan	1	0.07	0.07
Joint Toggle	1	0.06	0.06
Joint Bottom Block	1	0.05	0.05
Joint Bottom Block	1	0.05	0.05
Joint Spacer	1	0.05	0.05
Joint Top Plate	1	0.01	0.01
		Total	25.9

Table 1: Weight Breakdown

Appendix D- Bill of Materials

Bill of Materials			
Item	Quantity	Cost	Total
Vex Controller	1	\$300.00	\$300.00
Window Drive Motors	4	\$41.75	\$167.00
Victor Speed Controllers	5	\$116.00	\$580.00
Pan Tilt Assembly	1	\$65.00	\$65.00
Joint Drive Motor	1	\$42.00	\$42.00
Wheel Material	1	\$13.00	\$13.00
X10 Wireless Camera	1	\$100.00	\$100.00
Aluminum	1	\$100.00	\$100.00
Lexan	1	\$100.00	\$100.00
Misc Wires and Hardware	1	\$100.00	\$100.00
Batteries	3	\$60.00	\$180.00
Laptop	1	\$400.00	\$400.00
Ladder	1	\$230.00	\$230.00
8020 Rail	1	\$50.00	\$50.00
8020 Bearings	2	\$30.00	\$60.00
		Total	\$2,487.00

Table 2: Total Cost