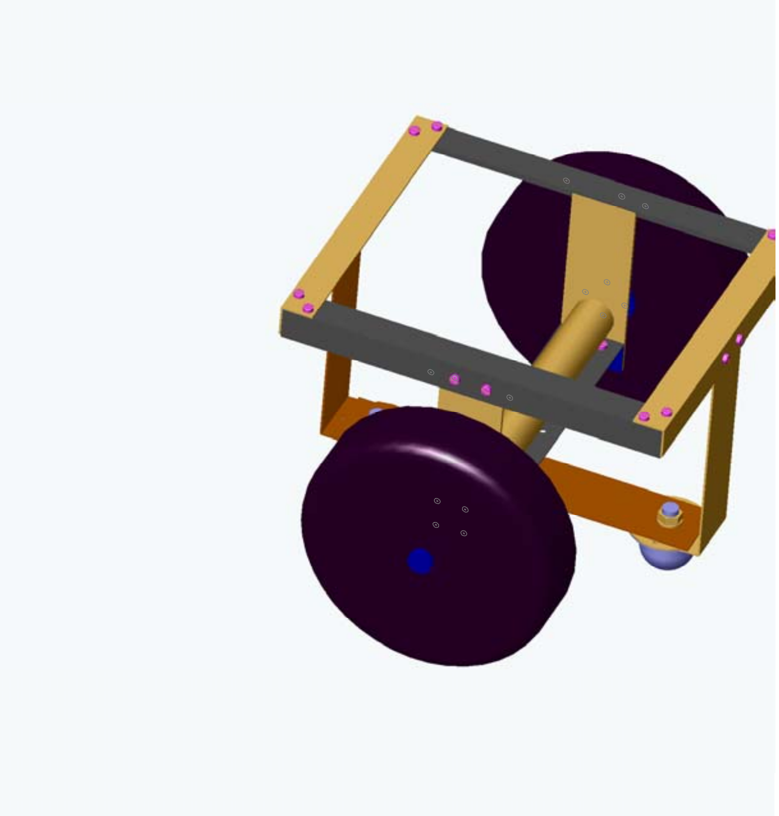
Senior Design Project at the State University of New York at Buffalo School of Engineering

Design of a Mobile Robot



Advisor: Venkat Krovi
Students: Damon Knapp
Rachel Ibaugh
Mike Sleasman

University at Buffalo The State University of New York

Fall 2001

ACKNOWLEDGEMENTS

We, Damon, Rachel and Mike, would like to take this time to thank:

Dr. Venkat Krovi	*Advisor*
Ken	*Machine Shop*
Rajankumar Bhatt	*Graduate Student*
Narayana Sundaram	*Graduate Student*

Thanks for all your help

TABLE OF CONTENTS

ABSTRACT

The study of Robotics is one of the fastest growing areas of engineering today. Unfortunately, most robots are very complex and require a large amount of knowledge and experience. Our task, as a group, was to design and build a mobile robot. For this project, Solid Works (a 3d modeling program) was used to design and model the robot along with Simulink to create a virtual model of the robot. The goal of the project is to provide a versatile platform for future research in the field of robotics and to make use of virtual modeling tools to analyze the robots performance.

PROBLEM

The design and construction of a mobile robot incorporates mechanical and electrical engineering along with some computer programming. This presents quite a challenge for three undergraduate Mechanical Engineers. A number of requirements and restrictions were placed on the robot.

REQUIREMENTS

1. The size must not exceed 12 inches in width and length.
2. The robot must carry a small pc computer that weighs approximately five. Pounds
3. The frame of the robot must be easily adaptable for future projects.
4. Input velocity and rotation must be supplied by a remote control.
5. Frame must be lightweight, strong and relative easy to assemble
6. The project must be completed by the end of the semester

DESIGN AND MODELING

The design of the mobile robot was done is such a way to be lightweight, inexpensive and easy to assemble. Because of its strength, cost and low weight, aluminum was selected to be the structure material. Box beams and angle beams were both researched when selecting the frames shape. Using a cantilever bending moment equation, with a one millimeter maximum deflection, the 1” box beam proved to be much stronger than the 1”x 1” angle beam. Yet the angle beam provided adequate strength at lower cost. Therefore the frame was constructed using 6063 aluminum angle beam, 1”x 1” with a 1/16” thickness. The frame (where the computer will rest) is in the shape of a 10” by 12” rectangle, fastened by a 3/16” nuts and bolts. (frame shown below)

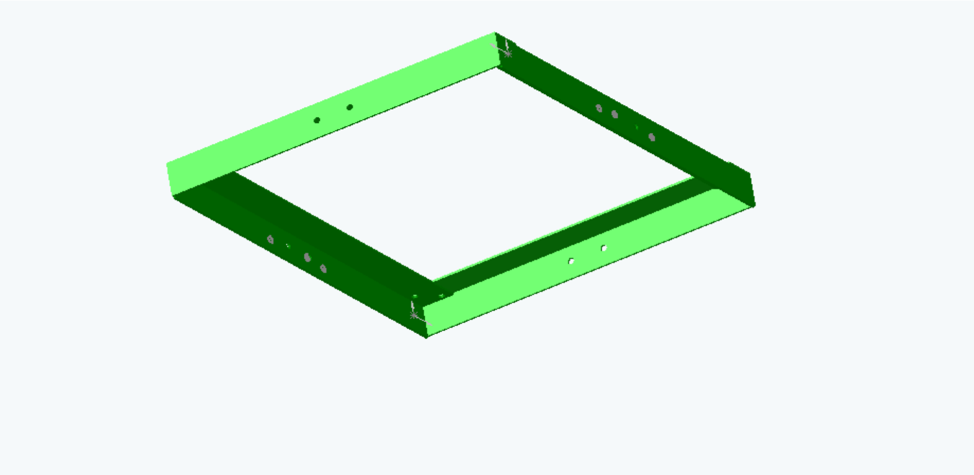


Fig 1.1 Frame

Eight inch by two inch thick Poly-Lok polyurethane wheels with roller bearings were chosen. This wheel style was chosen for traction on the tile floors that the robot will be operating on. Once the wheels arrived, the group members realized that roller bearings are not necessary. Therefore, an axle hub was designed as follows.

Constructed from a 2-inch thick bar stock of aluminum, the axle hubs were turned to 1 ¾" and cut to a length of 2 7/8".

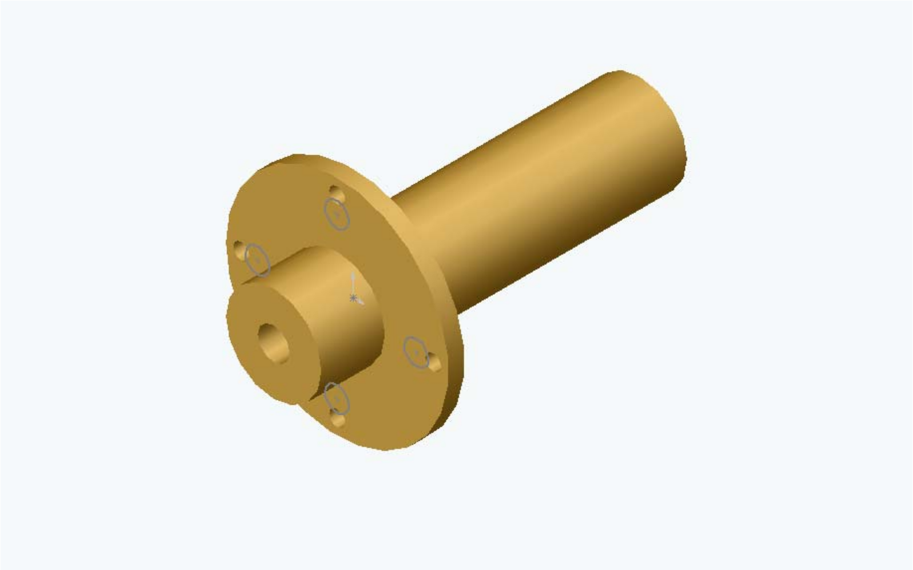


Fig 1.2 Axle Hub

Four 1/8" holes were drilled 90 degrees apart on the flange to allow #4 ¾" screws to fix the wheel to the hub at a radius of 3/4". The motor shaft had to be inserted into the hub through a ¼" diameter hole with a ½" depth. Locking the hub to the motor shaft was done with a #10-32 x 3/13 screw used as a setscrew.

A motor Flange was designed to attach the motors to the frame. Stability of the flange was of great concern. Originally, the Motor Flange was designed to be 1/16" thick. Once machining began it was decided that a 1/8" flange should be used instead. This was done for two reasons; the first being the importance of keeping the motors fixed along the same axis while the robot is in motion. If the motors do not stay on the same axis, response time and accuracy will be lost. The second reason considers any hardware addition to the frame that may occur in future research, which may increase the weight of the structure. Having the 1/8" plates will allow the frame to stay rigid with a weight increase. The flange has a length of 6 5/8” and a width of two inches. Our motor has a flange that will fit comfortably in the lower drill hole with a diameter of ¾”. Four 1/8” holes are set 90 degrees apart along a 1.02” diameter to allow a M3x6 bolt to fasten the motors.

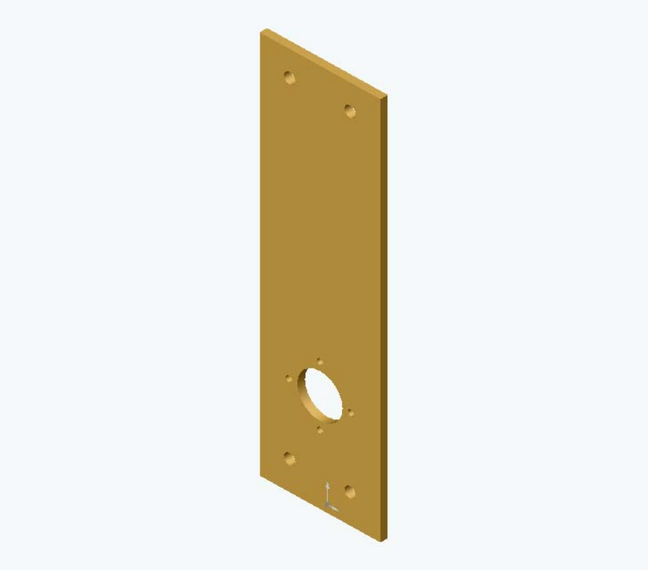


Fig. 1.3 Motor Flange

Two, two inch angle bars were machined to form brackets to support a 9 5/8 cross Plate. The drill holes are 3/16” located ½” from both edges.

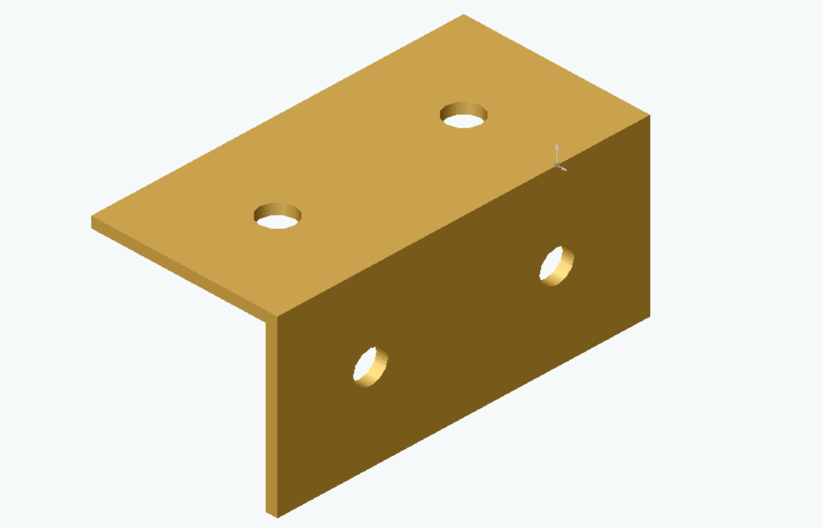


Fig 1.4 Bracket

The purpose of the Cross Plate is to provide stability for the frame and a place for the motors to rest upon. A thickness of 1/8" was chosen again for the same reasons above.

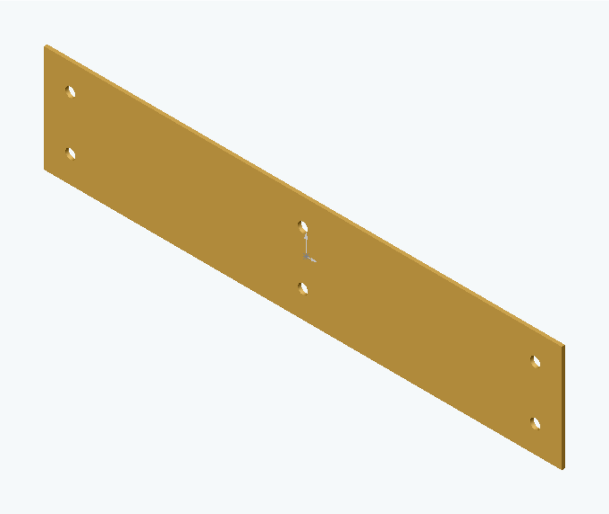


Fig. 1.5 Cross Plate

In order to balance the robot Stud-Mount Ball Transfers, STD, 1" Ball Dia, 3/8-16x11/16 Stud, 200#, were added to the front and rear of the frame.

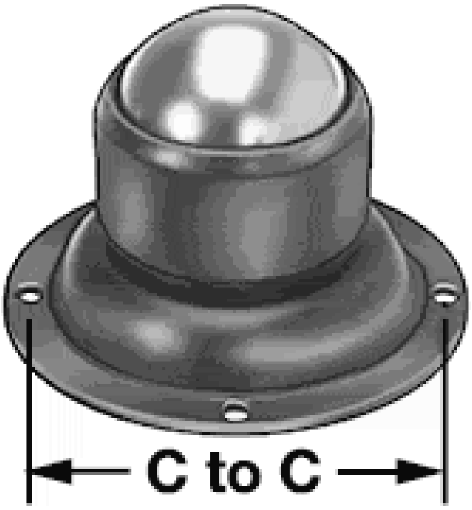


Fig 1.6 Castor

The castors are bolted to a Castor Leg constructed from 1/16” plate that is bent at 90 degrees.

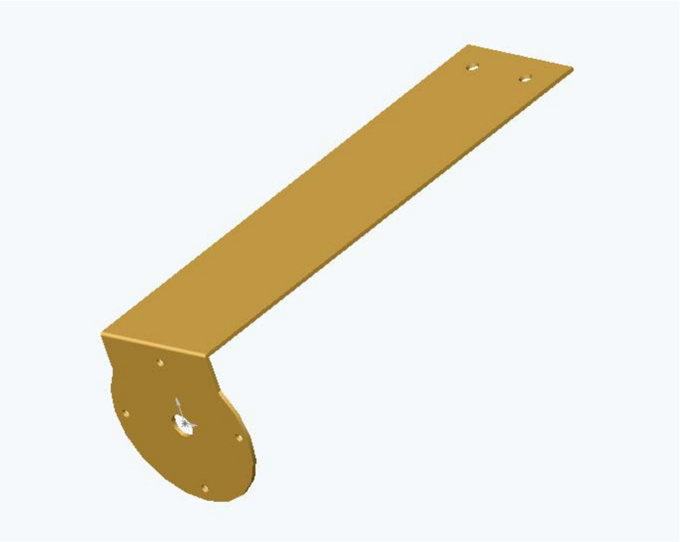


Fig. 1.7 Castor Leg

The front and rear legs are supported by a 11 ¾” plate. Using a 1/16” thickness gives the frame some flexibility, which will be needed to pass through rough terrain.

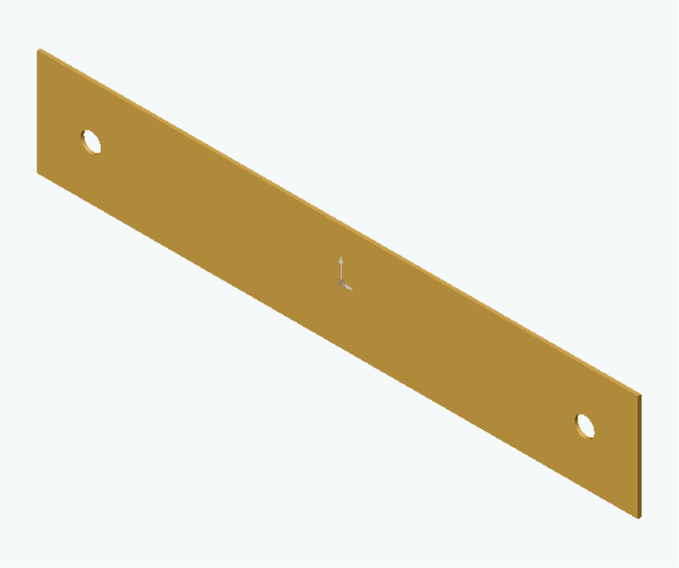


Fig. 1.8 Castor Bracket

The frame stands at a height of 10”. While the length is 12 inches, but the width is slightly larger than 12” constraint originally set. This was discussed with the group advisor and the oversized width was approved. Below is a 3-D model of the completed frame with wheels and motors.

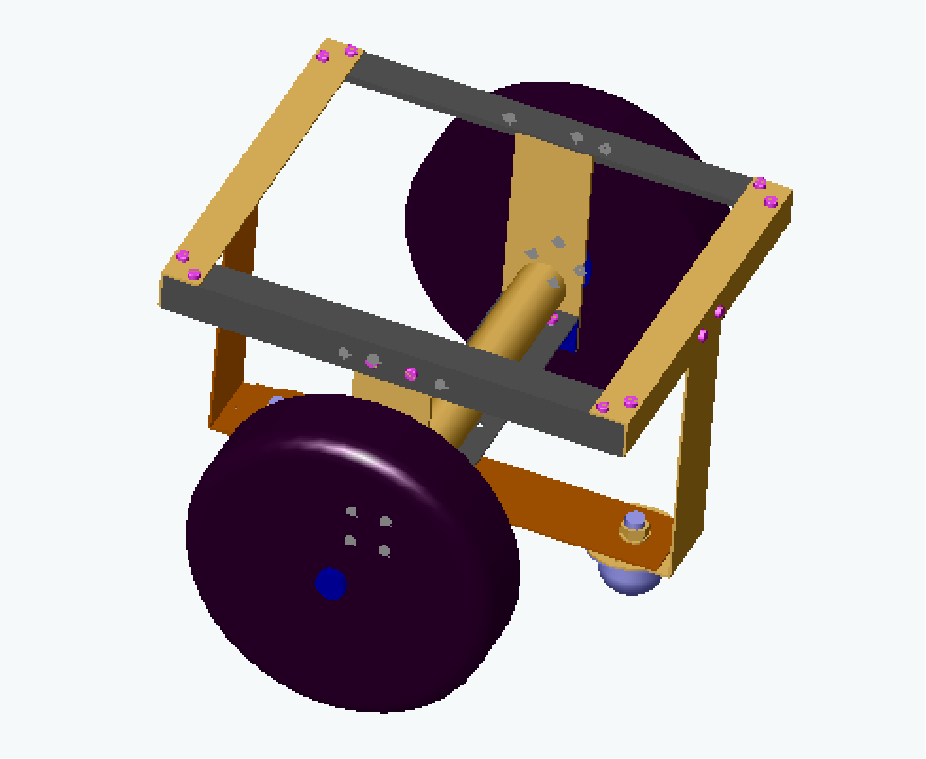


Fig. 1.9 Final Assembly

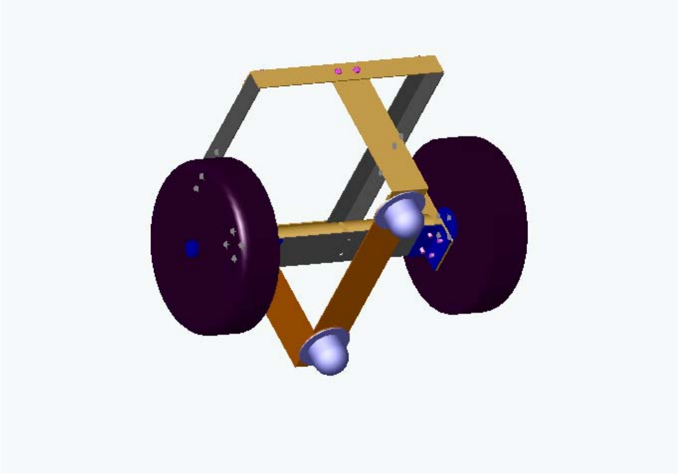


Fig 1.9 Final Assembly (View 2)

Attaching the computer hardware to the frame was easily done by using four Velcro straps. These straps are inexpensive, easy to use and offer enough strength to keep the computer in place.

DESIGN REVIEW

THE PURPOSE OF THIS SECTION IS TO DISCUSS MODIFICATIONS THAT CAN BE IMPLEMENTED FOR FUTURE BUILDS.

The 3/16” bolts are larger than what is needed. Instead the use of 1/8” bolts would provide enough strength along with a small reduction in cost and a more aesthetically pleasing appearance.

The castor legs are currently designed to have some flex when the robot encounters an obstacle. This will allow the robot to pass thru terrain that may be uneven. To improve this design, a spring suspension would be much more beneficial. This will allow the robot to climb inclines and pass thru rugged terrain while the drive wheels maintain contact with the ground.

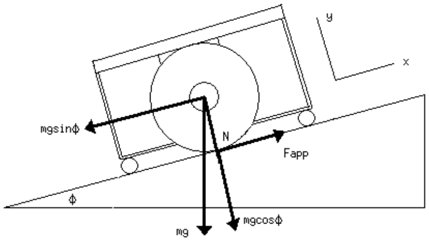
With dimensions on 8” dia. x 2” thick, the current wheels are heavy compared to the overall robot. Thinner lighter wheels will accomplish two things. One being reducing the overall weight of the robot, which will allow the use of smaller less expensive motors. The other is the cost of the wheels will decrease with smaller dimensions.

NOTE ON MACHINING

The machining process, which was originally thought to be quick and easy, was time consuming and more difficult than expected. This is due to the fact the group as a whole was inexperienced in such tasks. The major problem that occurred was drilling the bolt holes with precision. After some defects, the group learned how to correctly lay out the parts and most importantly how to use the correct methods for machining. To decrease the amount of scrap, both holes were filed slightly larger for assembly. Due to the strength of the material and the torque of the bolt, the structure will not lose strength because of the modified holes.

MOTOR CONFIGURATION

Two A-max, 20 watt motors with graphite brushes and a Planetary Gearhead with a 86:1 reduction are used in combination to drive the robot. These motors require a 12 V power supply and a max current of 1840 ma. At maximum amperage, each motor is capable of putting out 6000 RPM with a torque of 44.5 mNm. The motor choice was based on a few simple calculations to determine the required power to drive the robot (over estimated at 15 lbs or 67 N) easily up a thirty degree incline. By adding these parameters, we are sure to calculate sufficient torque in even the worst possible scenarios. The following calculations show how the continuous torque of each motor was determined.



Calculation of Force applied by each motor:

$$P_m = (F_{app})(V_{el})$$

for velocity of 2 m/s and coefficient of friction (μ) of 0.3

$$F_{app} = mg(\mu\cos\phi+\sin\phi)$$

$$51.39N = 67N[(0.3)(0.89)+(0.5)]$$

$$\therefore\ P_m = 10.3\ W$$

$$\text{each motor} = ½\, P_m$$

$$½\, P_m = (\tau)(\omega)$$

$$\omega = (V_{e}l)/(\text{radius of the wheel})$$

$$39.2 \text{ radians/sec} = (2.0 \text{ m/s})(0.051\text{m})$$

$$\therefore \ \tau = 1.3 \text{ Nm}$$

While this amount of torque for each motor should be sufficient, the book Mobile Robots by Jones, Seiger and Flynn recommend over-sizing the motors by a factor of three. The calculated torque then becomes 39 Nm. Since the motors selected still exceed this torque value, there is no doubt that the motors will have sufficient power.

Encoders

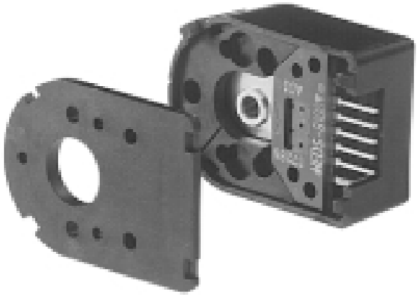


Fig 2.2 HEDL55 Optical Encoder

A digital encoder Hedl55 with line driver is also integrated into the system to allow for very precise control of the robot. These encoders were part of the motor package from Maxon motors, so compatibility was not an issue. These encoders are supplied five volts from the ESC-629 motor driver card, and are capable of 1000 counts per turn, have an operating frequency of 100Hz, and can measure a maximum acceleration of 250,000 radians per second. These features allow extremely accurate control of the motors.

ESC-629 Motor Driver Card

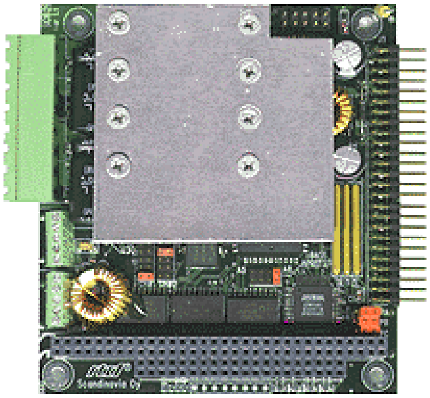


Fig 2.3 ESC-629 Motor Driver Card

The ESC-629 Motor Driver Card was used in conjunction with the PC/104 to control the motors. This card offers two independent channels and motor interfaces. The ESC-629 also supplies the power needed to operate the HEDL55 digital encoders and the power junction for the 12 volts required to run the motors. Additional features are listed below:

- Two full bridges for direct motor connection
- 60 V, 10A onboard MOSFET H-bridges
- -40 to +85°C operating temperature range

Remote Control Interface and BASIC Stamp

Fig 2.4 Sony Universal Remote

Fig 2.5 BASIC Stamp Board of Education Kit

In order for the robot to move anywhere, it must be given velocity and/or rotational input. This input is given by the remote control, which is a common Sony Universal Remote control used for televisions and VCRs. The infrared signal is picked up by an IR sensor connected to a BASIC Stamp 2 microprocessor. The microprocessor was programmed to read the remote's signal and send the information to the PC/104. The ESC-629 Motor controller Card then takes this information and feeds its to the motors and encoders. The following path is a simple representation of this process.

Remote → BASIC Stamp → PC/104 → ESC-629 Motor Driver → Motors & Encoders

The numbers one through nine on the remote represent different motions. The buttons two and eight will give the robot a forward and reverse direction respectively, while four and six will cause the robot to turn left and right respectively. One, three, seven and nine will cause a combination of translation and rotational motion (eg. button one will cause the robot to move forward and turn to the left). Button five is reserved as an emergency stop button, and will cause the robot to halt whatever motion the robot is currently in. Other buttons on the remote may be added on later for additional commands

The remote control is also able to provide the translation and rotational input to the Simulink program in order to simulate the robots movement. This can be done separately to evaluate the motion of the robot or while the robot is moving for testing purposes.

Dynamic Virtual Modeling

An important contribution to the designing and controlling of a mobile robot is the ability to watch a similar virtual robot on the computer screen. This virtual mobile robot was built and controlled to resemble the actual robot. There are two important roles for this computer-modeling portion of the design project.

The virtual mobile robot can test the both capabilities and limitations of the design. In fact, this virtual robot can give the designers and idea of how the mobile robot will move before the actual robot is even built. The virtual mobile robot also provides a visual image. It is useful to have a visual on the screen of the motion that the user is controlling the actual robot to perform, especially if the actual robot were not in the same place as the controller.

Creating the Virtual Mobile Robot:

The process for creating a dynamic virtual image of the mobile robot is quite complex. First, the appropriate image must be modeled. A simple mobile robot was modeled using SolidWorks 2001 and is shown in Figure 2.1. The mobile robot in Figure 1.9 was not used for this virtual simulation for various reasons. Great care must be taken when modeling the virtual robot. The model must be built to scale so the defining equations match the equations that would define the actual robot. Also, the coordinate axis for all parts modeled in the assembly must be the same, and must lay at the exact center of the robot when looking at it's top view, as well as laying the axis of both wheels. This then simplifies the equations that control the virtual robot and negates any position problems when transferring the model to the VRML view screen.

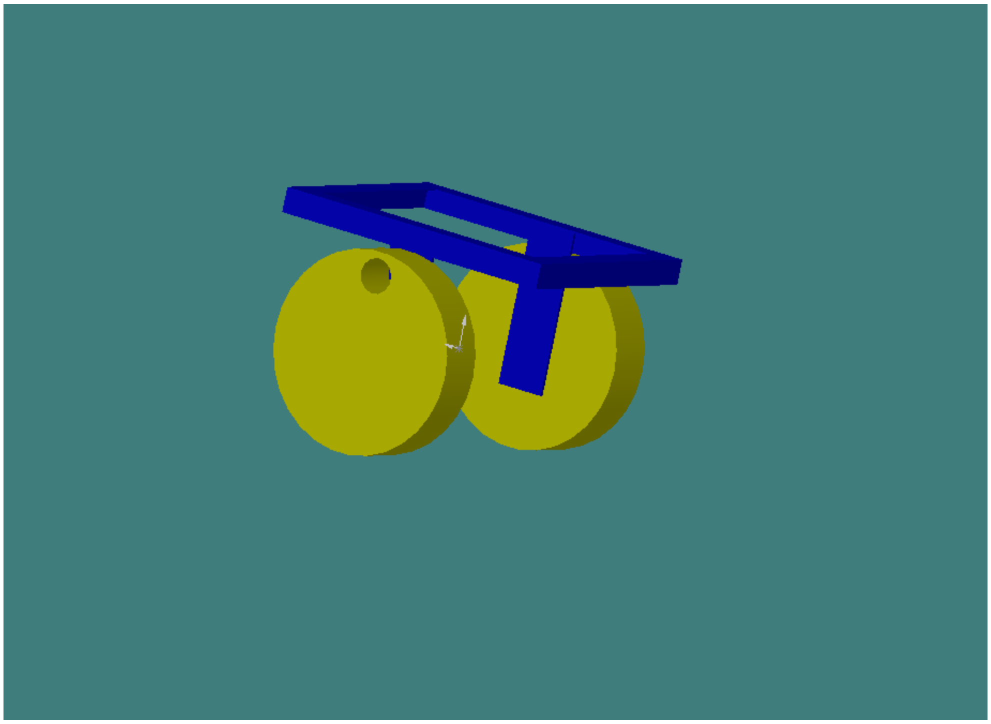


Figure 2.1 Virtual Mobile Robot

It is also important to note that the robot modeled in Figure 2.1 has a few key variations from the robot modeled in Figure 1.9. Only the frame of the robot and the two wheels were modeled for ease of analyzing the motions on small screens. The fear was that the complex mobile robot might hinder the user's ability to view the motion of the wheels and the robot itself. In addition, an offset hole was added to each wheel. This was done so that the motion of the wheels as they roll may be observed as the assembly translates and rotates across the screen.

After the virtual mobile robot was modeled, it was saved as a VRML file, which may then be viewed in a VRML web browser. This web browser displays the motions of the mobile robot, once these motions are programmed using Simulink.

Why Simulink?

There is an easier way to display the motions of the virtual mobile robot assembly. SolidWorks 2001 has built in it a program called Dynamic Designer. In Dynamic Designer, the user has the ability to give velocities and rotations to the parts of the assembly and then watch the simulation run. This program works fine, however the user has no way of knowing whether the motions displayed on the screen are absolutely correct or not. This is because there is no way to see the mathematical calculations governing the movements. Therefore, Dynamic Designer is not an ideal program to use when the both exact motions of an assembly and the reasoning behind those motions are desired. For this reason, a Simulink program was built to govern the motions of the virtual mobile robot. A copy of this program may be found in Appendix B.

Programming the Virtual Mobile Robot:

The red and blue boxes in the Simulink Figure (Appendix B) are the inputs to the program, velocity and angular velocity respectively. These values are obtained from the program that enables the remote control to control the actual mobile robot. These inputs are then run through the blocks that make up the equations defining the motions of the mobile robot. Each block, using C++ programming, contains a function (such as a trigonometric function, or the integral function, etc.). The inputs may then be guided through the appropriate blocks using the connecting lines and arrows as seen in the Figure. The final translation and rotation vectors are then outputted through the VR Sink (Virtual Reality sink), which is in this case, our virtual mobile robot in the VRML browser.

The equations defining the position of the virtual mobile robot are derived from the following figure. Figure 2.2.

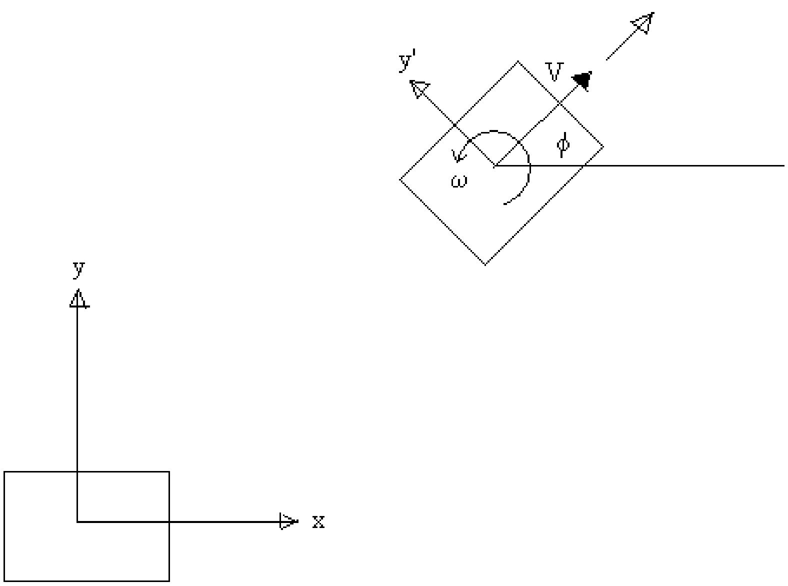


Figure 2.2 Instantaneous Mobile Robot Position

The mobile robot has an initial coordinate system shown at the bottom left of Figure 2.2. It is then moves to a new position. (Recall, the forward velocity V and the angular velocity ω are known values inputted to the Simulink program.) From the above figure, the componential velocities of V, in the x and y directions are the following equations:

$$x,\ dot = V * \cos \phi \qquad \text{Eq. (2)}$$

$$y,\ dot = V * \sin \phi \qquad \text{Eq. (3)}$$

Integrating the above equations in Simulink gives the instantaneous position of the mobile robot. Similarly, integrating the angular velocity, ω, that is inputted to the Simulink program gives the instantaneous angle of the mobile robot with respect to the original coordinate system.

Figure 2.3 begins the derivation of the equations governing the rotation of the virtual mobile robot’s wheels.

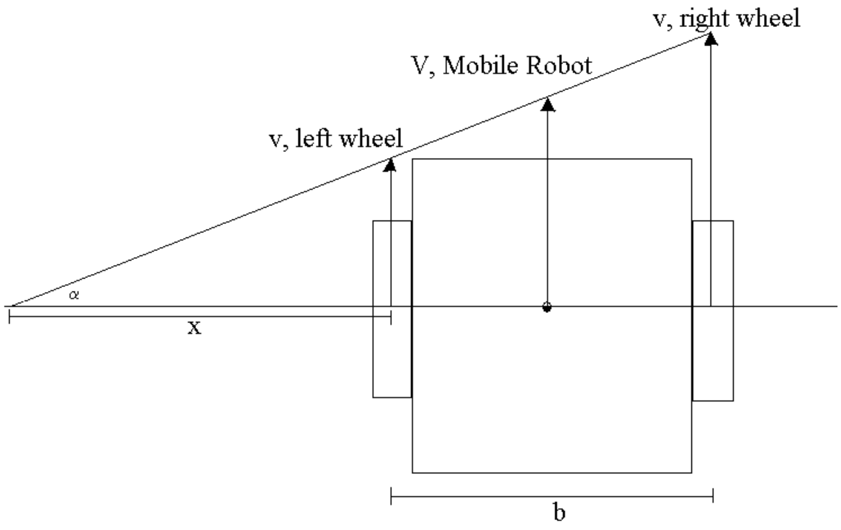


Figure 2.3 Diagram of Mobile Robot

Figure 2.3 is a diagram of the top view of the mobile robot, displaying the velocities of each wheel, and the center of the mobile robot. Similar triangles result in equation (4):

$$(R*\Theta,dot_{(left)}\ ) / x \ = V / (x + b/2) \ = \ (R*\Theta,dot_{(right)}) / (x + b) \qquad \text{Eq. (4)}$$

where: R = the radius of the wheels

$\Theta,dot_{(left)}$ = angular velocity of the left wheel as it rotates (rolls)

$\Theta,dot_{(right)}$ = angular velocity of the right wheel as it rotates (rolls)

Matrix algebra concludes to the following equations, solved for the angular velocities of each wheel as they roll:

$$\Theta,dot_{(left)} = V / R - b/(2*R) * \omega \qquad \text{Eq. (5)}$$

$$\Theta,dot_{(right)} = V / R + b/(2*R) * \omega \qquad \text{Eq. (6)}$$

In the same manor as before, the angular velocities of each wheel are created in Simulink and then integrated for the instantaneous angle of each wheel. This then determines the angle of each wheel, for each instant of time and therefore, the virtual mobile robot moves exactly as the actual robot does.

When applying these derived equations to the virtual mobile robot, through the Simulink program, careful attention must be paid to make sure that the translation and rotational movements are about the right axis. A problem encountered when giving the 3 parts of the assembly translation and rotation movements was the fact that the rotation sf 4 vector that defines each part gets very complex when attempting to give a part rotations about more than axis. This was needed when controlling the wheels. In the VRML image, the wheels must rotate (turn) with the frame about one direction vector, and rotate (roll) a different angular velocity about a second direction vector. This problem was overcome by assigning the two wheels to be children of the frame part in the VRML editor. Then, the wheels translate and rotate along with the frame, but they may be assigned an additional rotation about a different vector also.

With the Simulink program correctly built, the designer uses the remote to send rotational and translation input to the robot, controlling the motion of the actual robot and the virtual model simultaneously.

CONCLUSION

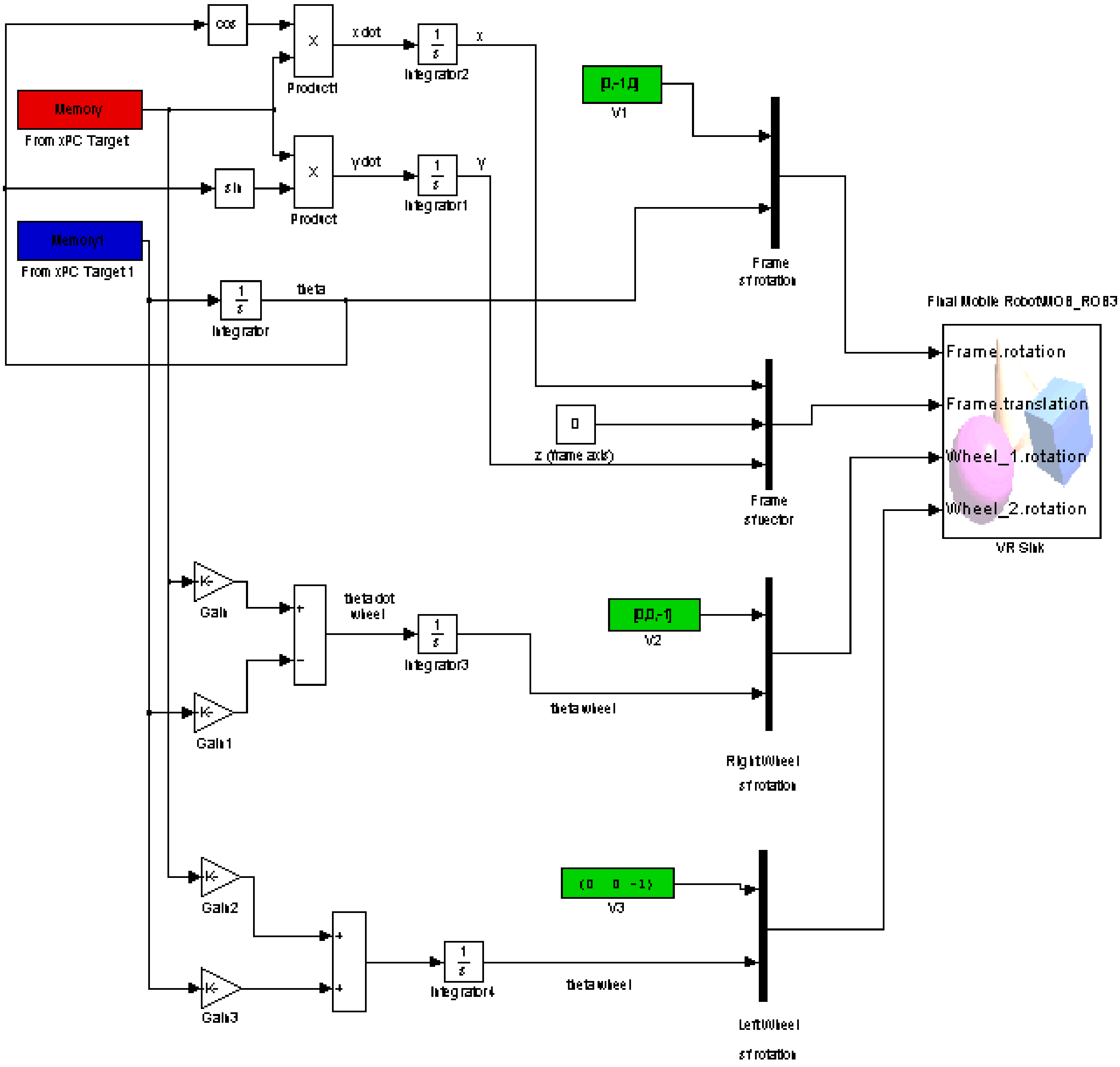
Our group feels the goals of the design are accomplished with the robot's simplicity. With such a basic frame, it would be very easy to add on additional features, such as arms, batteries, and additional sensors, for future robotic projects. The motors chosen are over-powered for their application, but the additional power increases the robot's versatility. The remote control system is able to transmit rotation and rotational input as desired to both the robot and the virtual model. The Simulink program describes the motion of the robot very accurately, allowing the user to see and analyze the movements even when the robot is turned off or out of sight. This system should prove to be a very useful and adaptable platform for future robotic applications.

Appendix A:

Part Description	Quantity	Cost Per Pcs.	Total
Raw Material			
Alloy 6063 Aluminum 90 Deg. Angle 116" Thick, 1x1" legs, 8' length P/N 88805K47	4'	$ 9.40/ 8"	$ 4.70
StudMount Ball Transfer, STD, 1" Ball Dia, 3/8-16x11/16: Stud, 200#	2	$ 7.29	$ 14.58
Ploy-Lok Polyurethane Wheels, 8" Dia., 2" Thick, With Roller Bearings	2	$ 26.75	$ 53.50
Aluminum 2" Dia x 1' long Solid Rod	5 3/4"	$ 32.32/Ft	$ 15.49
#10-32 x3/13 bolts	24		
M3 x 6 Bolts (to attach motors)	8	1.50 Pkg.	$ 3.00
Washers	2	$ 0.30	$ 0.60
Nuts (castors)	2	$ 0.30	$ 0.60
#4 x 3/4" screws for attching hub to wheel	8	$.96/14pcs.	$ 0.55
Motors			
A-Max 32 Graphite Brushes. 20 watt	2		
Planetary Gear Head GP 32	2		
Encoder, HP-REDL	2		
		total	$ 972.23
Computer Hardware			
2-Line Serial LCD Module	1	$ 41.65	$ 41.65
LCD Cable	1	$ 5.00	$ 5.00
9 Volt DC 300ma Wall Mount	1	$ -	$ -
Board of Ed. -Full Kit	1	$ 92.65	$ 92.65
SBC w/366 MHz K^ 128 MB SDRAM	1	$ 894.00	$ 894.00
Versabox Enclosure	1	$ 125.00	$ 125.00
200 watt ATX PS	1	$ 72.00	$ 72.00
1" RJ45 Extension	1	$ 19.00	$ 19.00
Cable Kit for VSBC-6	1	$ 112.00	$ 112.00
1.44MB Floppy	1	$ 49.00	$ 49.00
15GB Hard Drive	1	$ 249.00	$ 249.00
ECS629	1	$ 495.00	$ 495.00
Miscallanious			
22 Gage Wire	1 roll 100'	$ 3.99	$ 3.99
Sony Remote Control	1	$ 6.99	$ 6.99
		Final Cost of Robot	$3,230.53

Appendix B:

Simulink Control of the Virtual Mobile Robot



REFERENCES

MOBILE ROBOTS – Inspiration to Implementation 2nd edition

J.L. Jones, B.A. Seiger, A.M. Flynn