Robotics Challenge: Design and Construction of an Autonomous Robot

Colby McKibbin
cl.mckibbin@colostate-pueblo.edu

Austin Meminger
a.meminger@colostate-pueblo.edu

Kenny Morrison
kl.morrison@colostate-pueblo.edu

Eric Robinson
es.robinson@colostate-pueblo.edu

Shae Anderson
s.anderson@colostate-pueblo.edu

Colorado State University - Pueblo
Dr. Huseyin Sarper
huseyin.sarper@colostate-pueblo.edu
April 6th, 2012

Abstract

The project for this team comes from this year's 6th Annual Robotics Challenge. This robotics challenge is different from others because every robot competitor will have to be autonomous. The future of robotics is a robot that can do a job on its own and needs no input from its human builders, except programming.

The mission of this team is to design and build an autonomous rover that can operate in difficult terrain and be able to navigate around obstacles in its path while following a radio signal. The rover will be equipped to receive a signal from the transmitting beacon and decipher the direction of the beacon. The rover will have to follow this signal and reach the stationary transmitting beacon without any system or mechanical failures. It will accomplish this by utilizing a skid-steering system and sensors, which will allow the rover to find its way to the beacon located 50 meters away. The path the rover takes to reach the beacon will be completely dependent upon the inputs the rover receives from its sensor arrays. The rover will be equipped with IR and touch sensors to navigate around impeding obstacles. The IR sensors will be used to detect larger objects that are further away and will allow the rover to make course corrections in advance. The touch sensors will be used to detect smaller, closer objects that the IR sensors do not detect. If a touch sensor is activated the robot will stop and back Up and make the necessary course corrections before moving forward.

Four dc electric motors will be used to power the rover. There will be one motor at each wheel. The mechanical design of the robot was focused on being able to navigate difficult sandy terrain. The rover's mechanical and electrical systems will be integrated using the Arduino platform. The Arduino will receive the inputs from the sensors and then give directions to the motors based on the way in which it has been programmed. When this year's robot competition is over there will be a firm base for future autonomous robot competitions. Overall, with or without success, this project will be a great engineering design experience into the field of robotics.

1. Introduction

The project for this team comes from this year's 6th Annual Robotics Challenge. This robotics challenge is sponsored by Adams State College and NASA's Colorado Space Grant Consortium. This year's robotics challenge is different from others because every robot competitor will have to be autonomous. The future of robotics is a robot that can do a job on its own and needs no input from its human builders, except programming.

2. Problem Statement

The mission of this team is to design and build an autonomous rover that can operate in difficult terrain and be able to navigate around obstacles in its path while following a radio signal. The robot will possibly face up to four courses of increasing difficulty. The first course has no obstacles so the only problem for the robot is to find the radio signal and move to it. Further courses will have obstacles such as varied terrain levels and physical objects between the start point and the radio beacon. The location of the courses will be in Alamosa, Colorado on the sand dunes. This means that sand will be a constant problem that will need to be solved through design.

3. Problem Approach

The rover will be equipped to receive a signal from the transmitting beacon and decipher the direction of the beacon. The rover will have to follow this signal and reach the stationary transmitting beacon without any system or mechanical failures. It will accomplish this by utilizing a skid-steering system and sensors, which will allow the rover to find its way to the beacon located 50 meters away. The path the rover takes to reach the beacon will be completely dependent upon the inputs the rover receives from its sensor arrays. The rover will be equipped with IR and touch sensors to navigate around impeding obstacles. The IR sensors will be used to detect larger objects that are further away and will allow the rover to make course corrections in advance. The touch sensors will be used to detect smaller, closer objects that the IR sensors do not detect. If a touch sensor is activated the robot will stop and back up and make the necessary course corrections before moving forward. Four dc electric motors will be used to power the rover. There will be one motor at each wheel. The mechanical design of the robot was focused on being able to navigate difficult sandy terrain. The rover's mechanical and electrical systems will be integrated using the Arduino platform. The Arduino will receive the inputs from the sensors and then give directions to the motors based on the way in which it has been programmed.

3.1. Design

The basis for the rover design is determined by the need to for the rover to be able to navigate a variety of difficult terrains. It would need to be able to move through loose sand, over small rocks, and be able to turn around objects that it cannot go over. An initial rough design sketch (shown Figure 1.) was drawn after discussion of what the rover should incorporate to be successful in moving in varied terrain. The design would incorporate a flat mounting platform and housing to accommodate the electronics. The platform would ride on four extended struts, each with independent suspension and one dc motor.

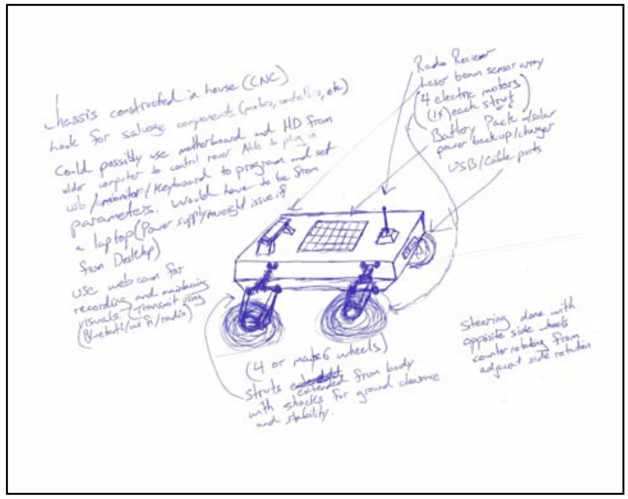

Figure 1. Initial Rough Design Sketch

The motor, (Figure 2.) selected to power the rover, needed to be light (as there would be one at each strut) and powerful enough to climb over smaller obstacles.

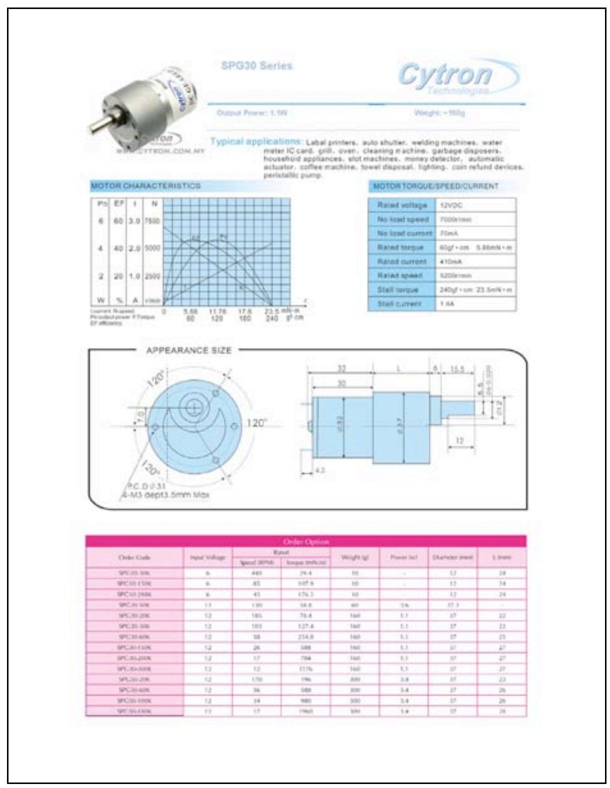

Figure 2. Motor Specifications

The struts would be long and angled to provide adequate ground clearance and have enough movement in the suspension to be able to ride over rocks without the rover tipping over. The motors are mounted at the end of each strut arm and wheels are directly mounted to the motor drive shaft. A gearing system with one motor driving four wheels was considered, but this design was favored because there would be minimal problems from sand causing problems with the drive system. The "four motor" design also eliminated the need for a complicated steering system. With the "four motor" design the rover is able to utilize skid steering.

After the rough concept was drawn, work began of designing the concept in Solid Works. Several variations of the rover concept were created. The first design was a couple simple renderings, which were helpful in visualizing the rover. (Figure 3. and 4.)

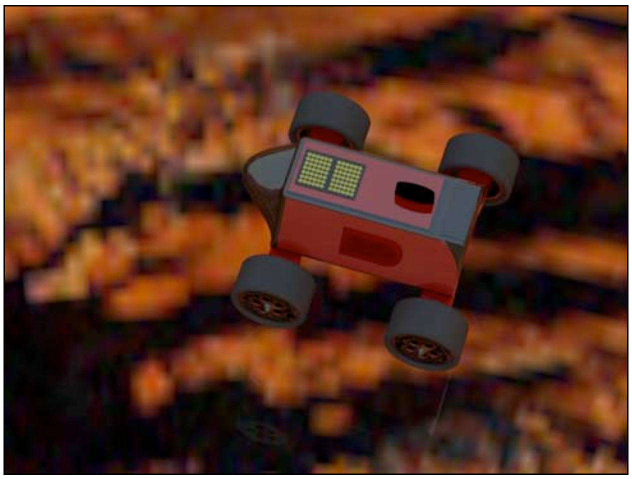

Figure 3. SolidWorks Design Renderings

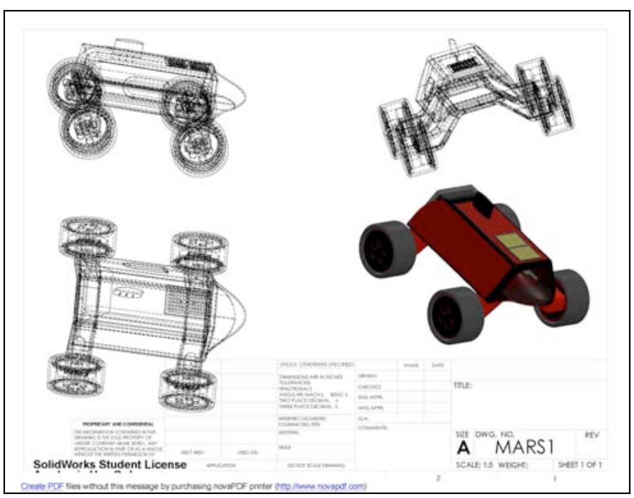

Figure 4. SolidWorks Design Renderings

These early drawings in Solid Works proved to be very helpful in moving forward with the design. The next step was to incorporate a working suspension into the Solid Works model. The first idea was to create an internal suspension in the strut mounts. (Figure 5.) Each strut has two mounts. Only one is shown in this image.

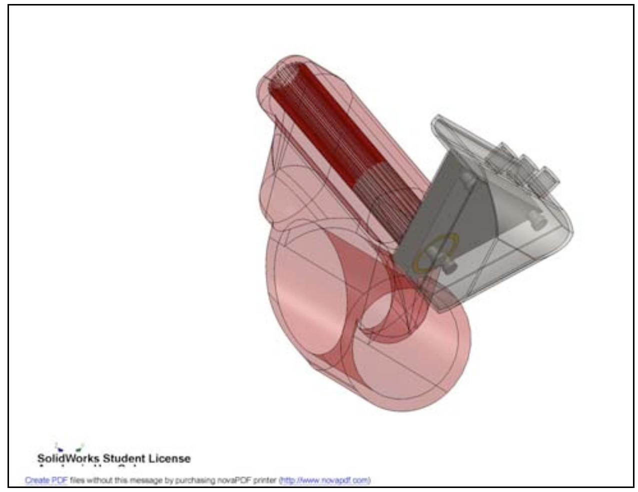

Figure 5. Internal Suspension

The idea was to use two tension springs opposite of each other to create equilibrium on the strut to keep it level. The strut would move when a force greater than tension in the springs acted on the strut. Although this design would work, it was decided to use alternate design with the springs on the outside to keep the weight of the rover down. (Figure 6.)

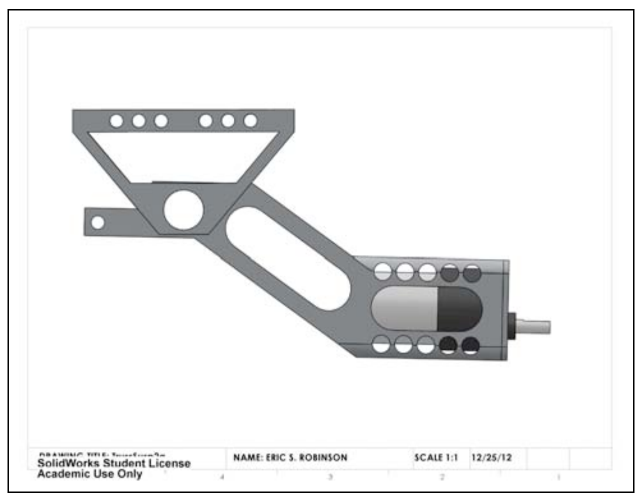

Figure 6. Alternative Design

Once determined the strut design was feasible a subassembly was created in Solid Works. (Figure 7.) This subassembly was used to determine the weight of rover using a project evaluation program in SolidWorks. The estimated weight for the rover subassembly is 1.3Kg. (DC motors included)

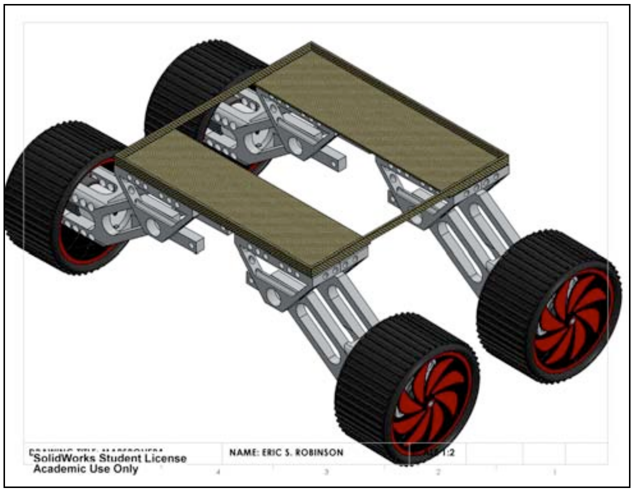

Figure 7. Subassembly

Now that the design was finalized and determined to be feasible to build, work began on building a rough prototype (Figure 8. And 9.) until fabrication can begin on the SolidWorks design.

Figure 8. Prototype

Figure 9. Prototype

The prototype body seen in Figure 8. and 9. has been made out of scrap metal and uses torsion springs on the pivot points along the strut arm giving the prototype rover a crude but working suspension. Until the final design can be completed (Figure 10. and 11.) this prototype platform will allow for thorough testing of the electronic components and programming.

Figure 10. Final Design

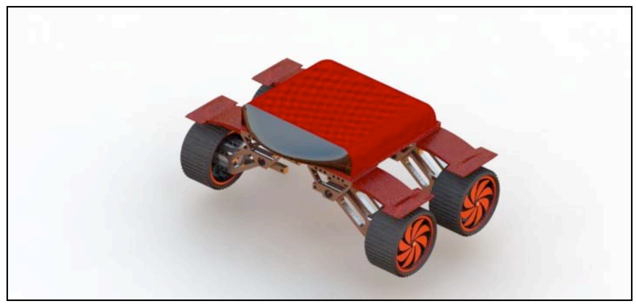

Figure 11. Final Design

3.2. Programming

The basis for the programming is the code provided at the Robotics Challenge Workshop held on November 21st, 2011. Our rover was not built at this point, so the priority was constructing the robot so until mid-March, the Arduino board was dissembled and set aside. When the time came to reassemble it, the code no longer worked. Each part of the code was tested in order to figure out where the error was. After determining that it was not the motors or IR sensor, the final thing to test was the compass. Any code relating to the compass would not work with the setup. The first assumed possibility was a faulty compass, however upon further investigation it was discovered that it was simply placed upside down on the circuit board. After this correction, the Robotics Challenge Workshop code worked again, enabling the robot to go west and avoid obstacles. Although it was frustrating to be back at square one, this hiccup did provide the opportunity to gain further understanding of the current code.

The next objective was to wire and program the receiver. The fritzing diagram provided at the second Robot Challenge Workshop made the wiring easy; and the basis for the coding was the code provided at the second Robot Challenge Workshop. The receiver the team purchased only functions as a receiver (it does not have a TX/RX pin), so the code provided had to be adjusted. Fortunately, the only place where this mattered was at the beginning, where it set the value of that pin to 0. This simply puts it into receiving mode. Ours only works as a receiver so this line of code and the line of code that created a variable for the TX/RX pin were deleted. This slightly modified code was put at the beginning of the void loop to get a heading value, which is subtracted appropriately from the compass heading value to make the robot treat the beacon heading direction as west and go that direction. At this point, there is no way to text this code because no beacon simulator is available.

The IR sensor currently is not very efficient due to its narrow view. In order to fix this an additional sensor is needed, so the next procedure is to add a touch sensor. This will require simple wiring such as that of the IR sensor. The programming will be similarly easy as it will simply be adding an "or" command where the code checks the IR sensor value and having it check the touch sensor's value as well. Additional IR sensors may also be added to improve its ability to avoid obstacles, which will be placed appropriately and require coding similar to that required when adding the touch sensor.

4. Conclusion

In the end, this team was unable to compete in this year's robotics challenge but hopes to use the work completed for future competitions. The robot still needs to be tested extensively before taking part in any competition. To do this a practice beacon will need to be constructed in order to simulate the competition format. Overall, this project has been a great engineering design experience into the field of robotics.