Big Autonomous Truck
Final Report

Joseph Gaita
8/03/2005

Instructors: A.A. Arroyo & Eric M. Schwartz

University of Florida
Department of Electrical and Computer Engineering
REU Summer 2005

Table Of Contents

Abstract

Working on a construction site can be one of the most dangerous jobs for any individual. There are countless factors that contribute to accidents and injures, including human error, negligence or just plain carelessness. Creating an autonomous dump truck that is able to efficiently navigate a construction site while avoiding objects in search of its destination will help alleviate the hazards of construction work.

Executive Summary

Big Autonomous Truck is a fully autonomous dump truck robot that will search its environment for a hole that it will attempt to fill. The entire design process of the robot was both a challenge and learning experience. The body of the robot was designed to resemble a dump truck and was equipped with large off-road tires to allow the robot to navigate a rough terrain. DC motors driven by a dual h-bridge were used for actuation and a servo was used to incline the bed. Numerous ultrasonic sensors were integrated to provide obstacle avoidance no matter what the lighting conditions in the room. I attached bump sensors around the sides and back of the robot incase the ultrasonic sensors failed.

For hole detection, a forward facing ultrasonic sensor was mounted on top of the grill in front of the robot. Once the hole was detected, the idea was for B.A.T. to back up, turn around, position itself over the hole, then incline the bed. In order to line up with the hole, three CDS cells were mounted underneath the back of the robot. The environment in which B.A.T. will navigate is black and there is a white border around the hole. The CDS cells detect this border, and are the robot will realign itself with the hole if it missed. After completing the robot, I would have added a speaker that beeped when the robot backed up.

Introduction

Big Autonomous Truck, or B.A.T. for short, will be a fully autonomous truck programmed to navigate a construction site in search of a destination while avoiding anything in its path. B.A.T. will be equipped with off-road tires to allow it to easily navigate the rugged terrain displayed by some construction sites. If successful, B.A.T. has the potential to prevent countless injuries sustained by today's construction workers and save builders time and money.

Integrated System

The integrated system includes the following parts:

- Three sonar range finders all mounted in the front of the robot. Two will be forward facing and used for obstacle avoidance. The third will be mounted underneath the frame and be aimed downward at a 45 degree angle. This sonar detects the border of the hole to be filled on the construction site.
- Three CDS cells aid the range finders in detecting the hole. Three mounted in the back to center B.A.T. on the hole and on mounted in the front to confirm a hole is detected. All the CDS cells point down. The hole is framed out by a white border, while the rest of the construction site is black. When the downward facing range finder thinks it sees a hole, the CDS cell tells the microcontroller if it is a false detection, or it really is a hole.
- Two IR range finders mounted on either side of B.A.T. to help with obstacle avoidance. The sensors are mainly used when B.A.T. is turning around after detecting the hole.

- 20 x 4 Hitachi compatible interface LCD screen with a backlight allows B.A.T. to provide feedback to the outside world.
- All sensors are controlled with a Mavric IIB microcontroller board, equipped with an ATMEGA 128 processor running on a 16MHz clock.

Mobile Platform

I choose to design B.A.T. to resemble a dump truck as close as possible. The forward facing range finders are mounted where headlights usually go. B.A.T. is also equipped with a stylish grill, allowing ample airflow through the inside of the robot, keeping the electronics well below maximum operating temperature.

A prototype of B.A.T. was fist built using empty cereal boxes and masking tape. Once I was satisfied with the results, the platforms was built in AutoCAD and modeled in 3D. I choose to spend extra time in the design of the platform to ensure there would be no surprises when final platform was constructed. Figure 1 below is an AutoCAD rendering of the platform design.

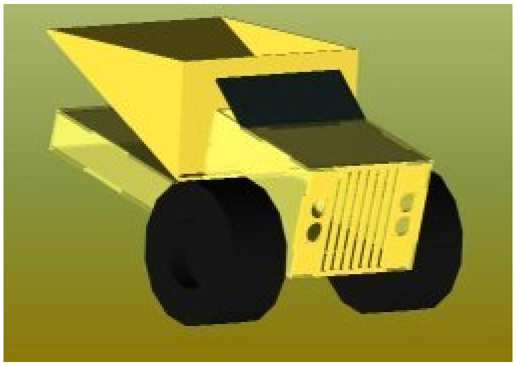

Figure 1: AutoCAD rendering

I choose to use two large front wheels for traversing a rough terrain and a castor in the rear (not shown in the rendering). After looking at several toy dump trucks and experimenting with other possible designs, the design described above proved to be the best option for mobility and durability.

Actuation

I used two DC motors driven by a dual h-bridge. The motor driver board used a TPIC0701 chip made by TI. This chip allows the motors to be speed controlled by a PWM output and direction controlled by a direction pin - 1 for forward, 0 for reverse. The first PWM channel of the Atme was used to control the motors. The initial functions that controlled motor speed produced a very jerky motion when B.A.T. was turning. A smoothing function which changed the speed of each motor in three steps was implemented and resulted in a much smoother turn. Also, when B.A.T. was moving at full speed, it often would collide with an object after detecting it before it had a chance to stop and attempt to avoid it. To remedy this, a half speed function was written and B.A.T. would only move at half speed when it was within 8”-12” of any detected obstacle. After adding this function, B.A.T. could nearly anything in its path. When B.A.T. was fully assembled, the added weight of the bed slowed down the robot significantly. This eliminated the need for the half speed function.

I used a Hitec HS-422 Standard Deluxe Servos to incline the bed of the robot. I attached a 3” servo arm to the servo and reinforced it with a flat piece of 1/8” aluminum. The servo is controlled by the second PWM channel and inclines the bed when it is set to fill forward. To minimize the overall weight of B.A.T., ping-pong balls were placed in the bed of the truck.

Sensors

- **Sharp GP2D12 IR Rangers:** These sensors were mounted on either side of the robot and were used to prevent the robot to bump into anything when turning around. These sensors output an analog value that decreases with distance as seen in figure 2.

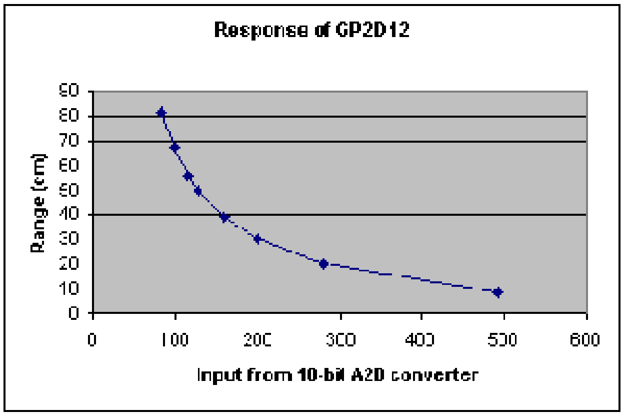

Figure 2: IR sensor output

The sensors were read each iteration through the main loop of the program when the robot was turning around. The function took three values from each sensor and returns the average of the three to prevent an erroneous reading. In addition, the sensors took 10 readings at startup to calibrate the robot to its surroundings with no walls nearby. This is to compensate for the sensitivity of IR sensors to different lighting conditions.

Figure 3: Sharp GP2D12 IR Rnger

- **Daventech SRF04 Ultrasonic Rangefinders:** Two of these sensors were mounted facing forward and were used for obstacle avoidance and one was used for hole detection. They have a range from 3" to 10' and are very affordable, making them perfect for this application. They operate at 40 KHz and are not sensitive to changes in lighting conditions. To avoid the sensors interfering with each other, only one is turned at a time. To operate the sensors, the trigger pin is pulsed high for 10us. The transmitter emits a 8 cycle sonic burst and the echo pin goes high for 100us to 18ms, depending on the distance of the nearest object. The time this pulse is high is measured in a loop to determine how close the robot is to an obstacle. I also included a timeout in the loop incase the sensor does not respond correctly to the input pulse. These sensors were purchased from acroname.com.

Figure 4: SRF04 Ultrasonic Rangefinder

- **Bump Sensors:** These were the cheapest and easiest sensors that were used on the robot. Because there were plenty of open digital I/O pins available on the Atmel, each of the 4 sensors was attached to a separate external interrupt pin.

Each switch closes when pressed so the pins were pulled up to 5V via a 10k resistor and. When the switch closes, the voltage at the pin drops to 0V, causing an interrupt. Alternatively, a voltage divider network could have been used on an A-D pin to determine which sensor was hit. When a sensor was hit, the robot would stop and get a reading from all its sensors. If one of the two rear bump switches was hit, the robot would move forward and continue to look for the hole. If either the left or right switches were hit, the robot would turn out of the way and continue in its previous direction.

- **CDS Cells:** CDS cells are variable resistors that vary with the amount of light that hits them. They were used to detect the white border around the hole in the environment. There were three CDS cells mounted underneath the rear of the robot. The sensors were each connected to an A/D input and were only measured when the robot had found the hole and turned itself around. The digital output of all three CDS cells when over the white border was about 240 and 100 over black so 200 was chosen as the cutoff between black and white. If one of the outside sensors were not over the hole, the robot would go forward, turn slightly, and reverse back to the hole.

Behaviors

There were two basic behaviors for the robot: searching for the hole and aligning with the hole. On start up, B.A.T. will take readings from all its sensors for calibration then it will begin to search for the hole. While roaming around the construction site, B.A.T. performs obstacle avoidance while continuously reading the downward facing ultrasonic sensor in search of the hole. Once it thinks it has found the hole, it will begin lining itself up with the hole.

To line up with the hole, B.A.T. reverses a short distance, turns around then backs up to the hole. Once it has finished backing up, it takes readings from the CDS cells to ensure that it is over the hole. If it is over the hole, the bed inclines and the ping pong balls roll off the bed into the hole. If one or more of the CDS cells are not over the hole, the robot attempts to realign itself. If only one of the three is over the hole, B.A.T. moves forward, turns 30°, then backs up. If, after two tries, B.A.T. is not over the hole, it assumes a false detection and continues to search for a hole. If 2 of the 3 CDS cells are initially over the hole, it only attempts to realign itself twice before inclining the bed. I chose to do this because when it did find the hole, it usually corrected itself within two attempts and if it took more, it usually would get stuck trying to align itself. After inclining the bed, the robot would move forward a few inches and spin clockwise in celebration.

A big challenge was ensuring the robot turned 180° around consistently. Sometimes the wheels would slip and it would only make it around 90°. In these cases, a false detection is the best I could hope for.

Experimental Layout and Results

Although all the sensors worked independently of each other, shortly after integrating all sensors in one program, the power board shorted out because a lose screw rolled under it causing a short. A 4’ by 8’ course was built out of a single sheet of plywood and a 2’ by 2’ hole was cut near one end of the course. The course was painted black and a 2” white border was painted around the opening of the hole. Foam core was used to make 6” black walls around the course. When aimed at the hole, the robot had a 95% success rate of finding it. There were a few occasions where the robot would falsely detect the walls as the hole and back into them.

Conclusion

I learned a great deal about microcontrollers and how to integrate several sensors to interact and provide feedback to each other. I feel that I learned more in this REU program than I have learned in any one class I have taken so far. Although I did not have a successful demo, B.A.T. did have several successful test runs. If I were to do any future work, I would add some type of beacon to the hole, and have the robot search for the signal so it would do more than blindly search for the hole.

Documentation

MAVRIC-IIB Manual:
http://www.bdmicro.com/images/mavric-iib.pdf

Atmega128 Complete Datasheet:
http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf

SRF04 Datasheet:
http://www.acroname.com/robotics/parts/R93-SRF04p.pdf

GP2D12 Datasheet:
http://www.acroname.com/robotics/parts/SharpGP2D12-15.pdf