Truck Loading Using an Autonomous End Loader

Final Project Report

By:

Kevin Hurley and Ryan Leman

Advisor:

Dr. D.R. Schertz

April 24, 2008

/

Table of Contents:

/

Abstract:

A remote control vehicle was modified to be driven by a microcontroller autonomously. The project utilizes numerous sensors, including infrared LEDs and an infrared transistor, an ultrasonic range finder, a digital compass, and rotary encoders. The vehicle uses these sensors to locate a bin filled with material, navigate to the bin and load the bucket, and then locate and navigate to a truck and dump the material into the truck. This process repeats until the truck is deemed full.

Truck Loading Using an Autonomous End Loader

Introduction:

In applications which require the repetitive transfer of material, such as dirt moving or operations in a rock quarry, it can quickly become prohibitively expensive to employ a skilled laborer to operate an end loader. With the constant advances in computing power, it has become increasingly likely that the automation of this process will become commonplace. To this end, the autonomous truck loading project consisted of developing a small scale autonomous end loader.

The end loader locates and navigates to a load at which point the end loader scoops the load. As the process continues, the end loader locates and navigates to a truck and empties the bucket into the truck. This process is repeated until it is deemed that the truck is filled. A variety of sensors are employed to determine end loader position and motor drive requirements to achieve the desired destination. A Silicon Labs development board is used to interface with the sensors and provide overall system control. It is required that the entire operation be carried out by a low cost fully functional autonomous system which quickly and accurately performs the entire process.

System Description:

The system consists of an abundance of low-cost sensors. To provide vehicle positioning information, a digital compass and an ultrasonic range finder are used. Further enhancing the positioning information, an infrared transistor is utilized to locate infrared beacons located at the truck and load. Velocity and distance calculations are made possible with the use of infrared reflective wheel sensors detecting track rotation. In addition, limit sensors for the bucket arms and tilt are employed to determine bucket positioning. A Silicon Labs development board is used for overall system control and sensor interfacing. The sensors feed information about vehicle, truck, load, and bucket positions to the Silicon Labs development board, which processes the information and controls the motors as needed via pulse width modulated signals. The Silicon Labs development board is mounted on the end loader along with a power source. Figure 1 shows the overall system block diagram.

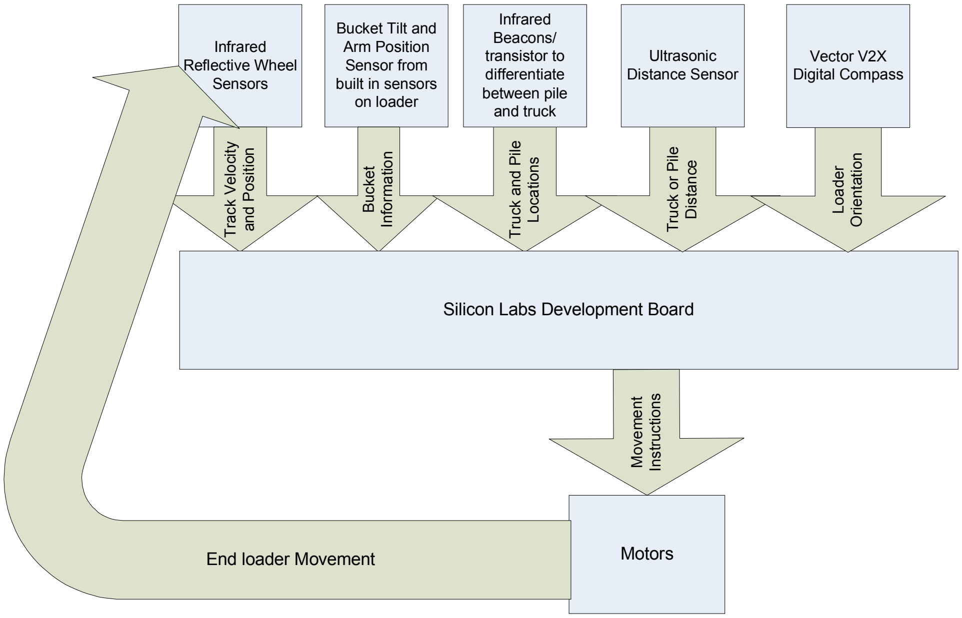


Figure 1: Overall System Block Diagram

Subsystems:

As Figure 1 illustrates, there are seven main subsystems. The subsystems are as follows: the velocity/distance sensors, the bucket tilt and arm position sensors, the truck and load sensors, the distance sensor, the digital compass, the development board, and the motors.

1. Velocity/Distance Sensors: Infrared Reflective Wheel Sensors

Infrared reflective wheel sensors are used as a form of velocity and position determination. They are attached to the side of the end loader, directly inside the tracks. The infrared reflective wheel sensors allow for more complicated navigation routines. By recording the distance travelled by each track, the wheel sensors enable the vehicle to make accurate positioning adjustments and ensure that the tracks are moving at the same velocity. The sensors generate a square wave by sensing the transition between black and white on a pinwheel on the inside of the track wheel.

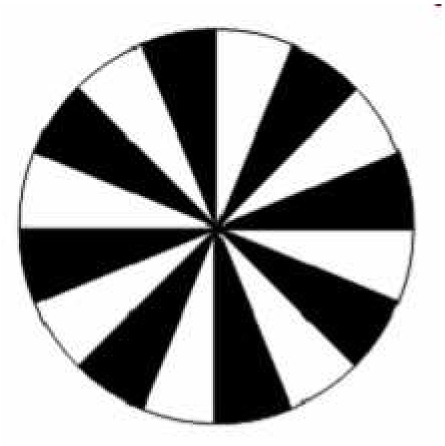


Figure 2: Pinwheel Used for Infrared Reflective Sensors

2. Bucket Tilt and Arm Position Sensors

It is crucial to be able to determine the bucket tilt and arm position during operation in order to scoop and dump the load. The Bobcat vehicle has built in limit switches to determine the endpoints for tilt and arm position. These switches are used to ensure that the arms are lowered to the correct position and the bucket is raised to the appropriate position, enabling the vehicle to scoop the load. Later, the switches are again used to ensure proper positioning of the arms and bucket to dump the load.

3. Truck and Load Sensors: Infrared Beacons

The infrared beacons are required to determine the locations of the truck and the load. A beacon is attached near the load and near the truck. One beacon operates at 8 Hz while the other operates using DC. This allows for differentiation between the load and truck. The signal, when received by the photo-transistor located on the vehicle, allows the vehicle to determine if the object being faced is the load or the truck.

4. Distance Sensor: Ultrasonic Sensor

It is required that the vehicle be able to accurately determine its distance from the object toward which it is navigating: either the truck or the load. In approaching the truck or load, it is vital to accurately gauge the distance in order to stop the end loader before hitting the truck or backstop behind the load. The ultrasonic sensor, when used in conjunction with the digital compass, allows the vehicle to initially be positioned off center and trigonometry used to navigate the vehicle to the correct position to approach the load or truck.

5. End Loader Orientation: Digital Compass

In order to use navigation routines more complicated than driving straight towards the load or truck, a digital compass is employed. The compass allows the end loader to orient itself to approach the load or truck from the correct angle.

6. Development Board

A C8051F340 Silicon Labs development board is used to provide vehicle control. The board uses the inputs from the aforementioned sensors to provide motor control and to determine the appropriate operation to be undertaken. The development board navigates the end loader to the load or truck and then operates the bucket as needed.

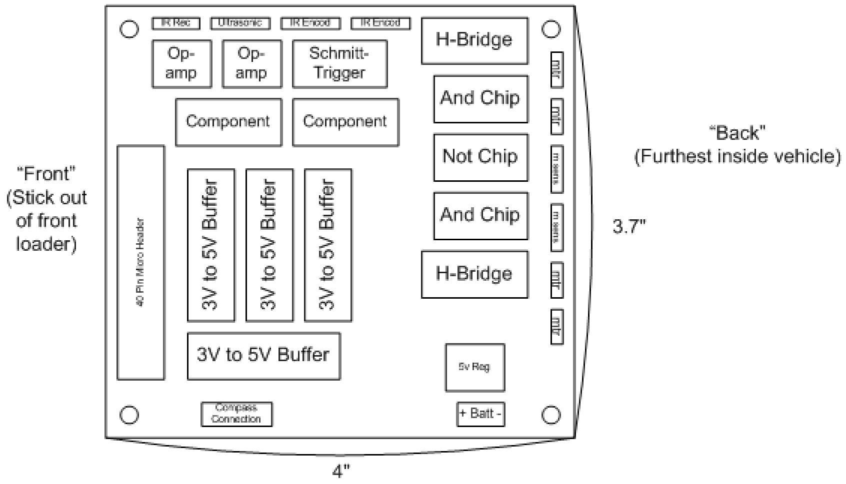
7. Motors

The motors provide end loader movement for navigating to the load and truck. A separate motor controls the right and left tracks in order to provide turning. Both motors operate in forward or reverse independently, giving a smaller turning radius as well as the ability to drive in either the forward or reverse directions. In addition, two separate motors operating the bucket arms and tilt provide for full control during scooping and unloading.

Circuitry

The operation of the sensors and interfacing of the sensors to the development board necessitated that a circuit board be created. A connector for the sensors, one for motor control and another for the compass and push buttons, fed into the circuit board from the components located on the vehicle. To provide bi-directional communication between the microcontroller and the circuitry, a 40-pin header was employed. To prevent false readings from the infrared reflective wheel sensors and the infrared transistor, a Schmitt trigger was used. The infrared transistor also required a series of op-amps to provide current to voltage conversion and additional noise filtering. Logic gates were employed to drive all of the motors as well as to prevent the bucket arms and tilt from being driven past their limits. Half h-bridges were utilized to convert the low current signals from the logic gates to high current signals to be provided to the motors. The five volts required to operate most of the component chips were supplied by a voltage regulator on the circuit board. As the microcontroller functioned at 3.3 volts and the component chips operated closer to five volts, bi-directional voltage translators were also utilized.

Circuit Diagrams:



2 Boards:
Board 1: 4 " X 3.7 "
Board 2: 2.5" X 2.5"
Mounting holes shown in corners, not to scale

Compass Circuit Board

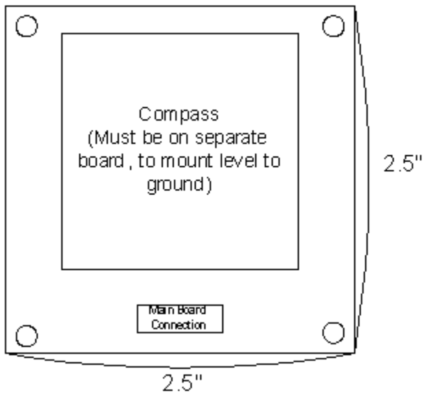


Figure 3: Circuit Board Layout

12	GND	GND	13	12	GND	GND	13
11	GND	P2.7	14	11	GND	NC	14
10	R_Motor Dir	P2.6	15	10	NC	P3.5	15
9	L_Motor Dir	P1.6	16	9	Lo_Bkt Sense	P3.4	16
8	R_Motor PWM	P1.7	17	8	Hi_Bkt Sense	P3.3	17
7	L_Motor PWM	P2.3	18	7	Lo_Arm Sense	P3.2	18
6	Bucket Raise	P2.4	19	6	Hi_Arm Sense	P0.0	19
5	Bucket Lower	NC	20	5	Encoder 2	P0.1	20
4	NC	P3.7	21	4	Encoder 1	P3.1	21
3	Arm Raise	GND	22	3	IR Beacon	GND	22
2	GND	3.3V	23	2	5V	3.3V	23
1	5V	3.3V	24	1	5V	3.3V	24

12	GND	GND	13	12	GND	GND	13
11	GND		14	11	GND	P3.6	14
10			15	10	Arm Lower		15
9			16	9			16
8			17	8			17
7		P1.0	18	7			18
6	Push Button	P0.4	19	6			19
5	Compass OUT	P0.2	20	5		P1.0	20
4	Compass SCLK	P2.1	21	4	Compass PC	P3.0	21
3	US out	GND	22	3	US Trig	GND	22
2	5V	3.3V	23	2	GND	3.3V	23
1	5V	3.3V	24	1	5V	3.3V	24

Figure 4: Voltage Translator Pinouts

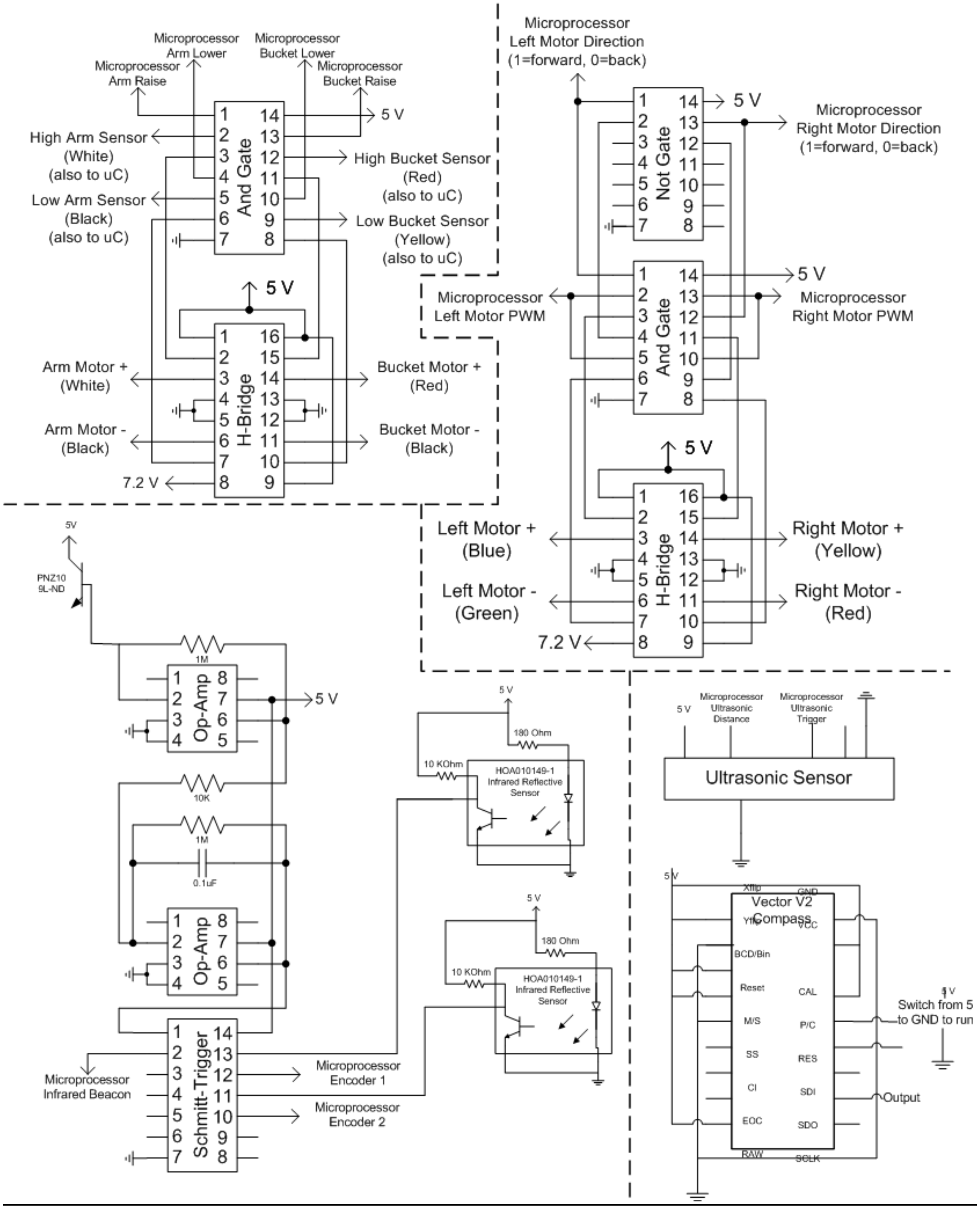


Figure 5: Overall Hardware Schematic

Software:

Software development was required to allow the microcontroller to interpret the sensor readings and provide overall system control. While there were several peripheral functions, only the main functions will be detailed. The main navigation routine is dictated within the newNav function. This function is passed the beacon to approach as an argument. To operate the motors, the setPWM function is called. The setPWM function is passed two arguments: the motor at which to apply the PWM and the PWM (in percent) to be applied. In order to find the beacons, the findFaceBeacon function is employed. The findFaceBeacon function is passed the beacon that the vehicle should face as an argument. Once the navigation routine has moved the vehicle close to the load, the driveForScoop function is called to have the vehicle scoop the load. At this point, the backAway function is called to have the vehicle drive away from the bin holding the load. The process is similarly continued for dumping into the truck.

During the previously mentioned operations it is necessary to use the ultrasonic range finder to determine distances. This is done by calling the USdistance function. In order to adjust the bucket to the necessary positions during operation, the adjustBucket function is called. To operate the compass, the compass function is called, which communicates between the compass and microcontroller using a serial peripheral interface. The final significant portion of code is the control for the infrared reflective wheel sensors. The counts are maintained in interrupts, but the Straight function is called to ensure that the vehicle drives in a straight line by adjusting track PWM.

Navigation Function: newNav

Prior to the newNav function, the main loop initially waits for a button press to indicate that the vehicle is facing the direction from which to approach the load. This is next done for the direction to approach the truck. At both points compass readings are taken and saved. After this point, the newNav function is called to provide navigation for the vehicle. To provide for a more robust navigation routine than simply driving between the truck and load only if the vehicle is immediately positioned directly between the truck and load, readings from the ultrasonic range finder and the compass are combined with trigonometry within the newNav function to develop the path to navigate.

Within the newNav function, the findFaceBeacon function is called to orient the vehicle so that it faces the beacon of interest. At this point, the USdistance function and the compass function are called in order to determine the distance from the object of interest and the angle difference between the beacon direction and the direction to approach the object of interest. Based on these readings, sine and cosine are employed to determine the x and y-coordinate distances that must be driven. The vehicle then drives the distance which is parallel to the wall behind the object of interest in order to center the vehicle.

During this process, ultrasonic readings are constantly taken to prevent running into unexpected objects.

Once the vehicle is centered in front of the object of interest, findFaceBeacon is once again called to ensure that the beacon is correctly approached. The vehicle next drives towards the beacon until the ultrasonic reads a specified distance. At this point, driveForScoop or driveForDump is called to finish driving towards the object of interest and to scoop the load or dump the load. If the infrared transistor is no longer receiving readings from the infrared beacon at any point during the drive towards the beacon, newNav calls findFaceBeacon once again to ensure a correct approach to the object of interest.

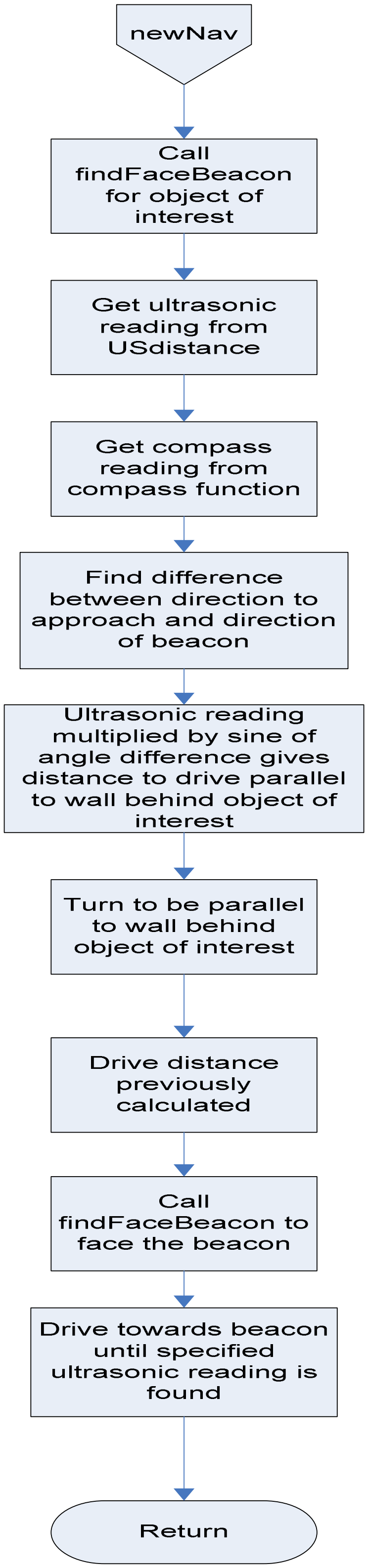


Figure 6: newNav

Motor PWM Control: setPWM

SetPWM is called whenever any motor needs to be turned on or off. A pulse width modulated signal is used for the track motors to give variable speed control to the motors. The function is passed two arguments: the duty cycle (in percent) and the motor to operate. The function can operate the left or right track forward or reverse at the specified PWM, move the arm up or down, or move the bucket tilt up or down. The PWM value is altered by changing the compare value for PCAs modules 3 and 4 as calculated within the setPWM function. The function also checks for bad duty cycle values greater than 100% and runs both bucket functions at full speed. In order to turn off any motors, a 0 value is passed for the PWM argument.

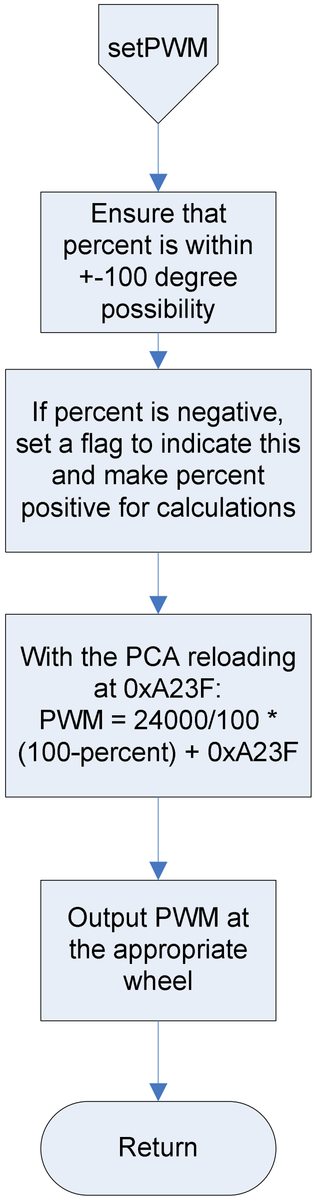


Figure 7: setPWM

Beacon Locating: findFaceBeacon

FindFaceBeacon is used to find the target and to keep in line with the target during movement. It uses setPWM to spin the vehicle until either the flashing load beacon or solid truck beacon is recognized. The beacon is identified using PCA module 2. This function is called whenever a new target is desired or the beacon is no longer being seen by the infrared transistor.

PCA module 2 is used to identify the beacon. The first time a beacon is seen, a flag is set. Each overflow of the PCA (1mS) is counted. Another beacon transition seen in a certain amount of time recognizes the flashing beacon. The overflow interrupt checks if the overflow count gets too high and the beacon pin is still low. This indicates that the solid beacon is being recognized.

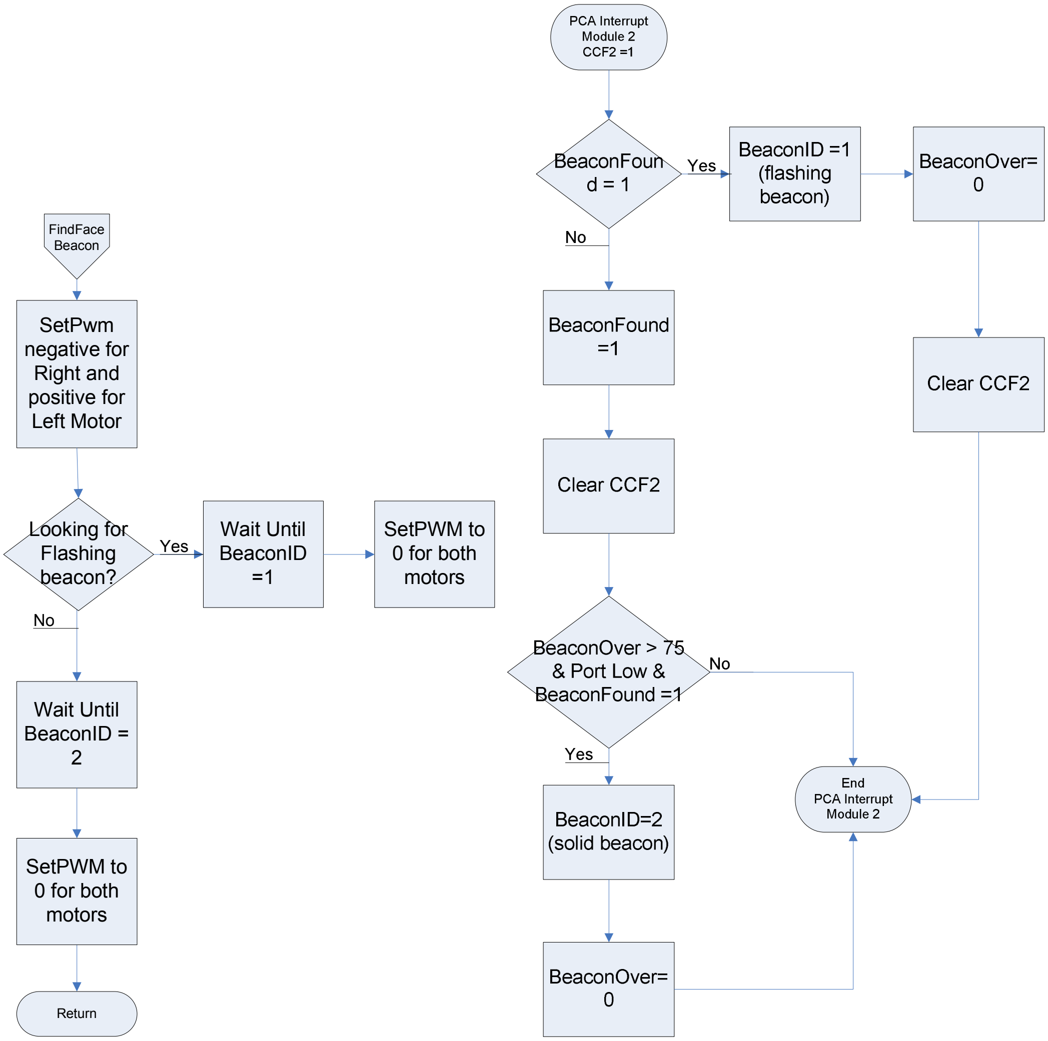


Figure 8: FindFaceBeacon, BeaconID (PCA Module 2)

Final Approach of the Load: driveForScoop

Once the navigation routine has moved the vehicle close to the load, the driveForScoop function is called to finish the approach of the load. This function ensures that the bucket and tilt are at their appropriate levels for scooping and then proceeds to drive to the load for a scoop. Once the vehicle has scooped the load, this function is exited. Much like the newNav function, the driveForScoop function also calls findFaceBeacon if the infrared transistor loses sight of the beacon at any point. This ensures a good approach is achieved. The driveForDump function is utilized on the dump after the navigation from the newNav function and operates in the same manner so it will not be further detailed.

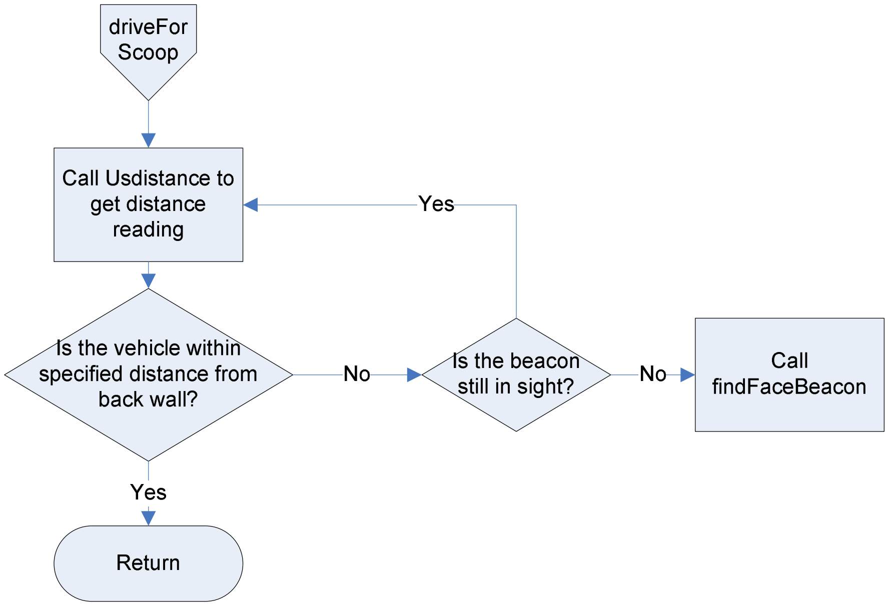


Figure 9: driveForScoop

Backing Away From Truck or Load: backAway

BackAway is a simple function used after a scoop or dump to back away from the load or truck. This allows the vehicle to get clear of all obstacles before rotating to find the next target. This function calls setPWM to set both tracks to reverse and uses the ultrasonic readings to determine if the vehicle has backed away far enough.

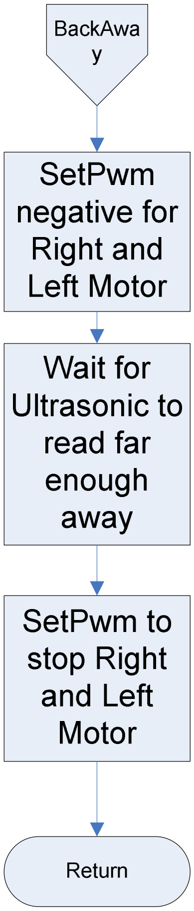


Figure 10: backAway

Distance Readings: USdistance

The USdistance function uses the ultrasonic sensor to read the distance from the target. A 15uS trigger pulse is sent to the ultrasonic ranging device which then returns a pulse with the width indicating the distance from an object. The USdistance function utilizes the PCA to time the reflected pulse, tracking the PCA overflows and capturing the initial and final counts to calculate the pulse time. The time of the pulse is then divided by 1392 to get the distance in centimeters.

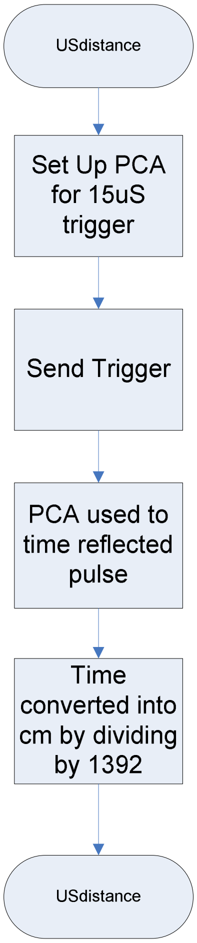


Figure 11: USdistance

Bucket Manipulation: adjustBucket

AdjustBucket is a simple function that does one of four operations depending on the value passed as an argument. The function lowers the arm and adjusts the tilt for scooping, raises the arm and tilt to the limit to approach the truck, lowers the tilt to the limit and raises it back to dump, and raises the bucket and tilt off the ground for transport. This function calls the setPWM function to start and stop the bucket and tilt motors.

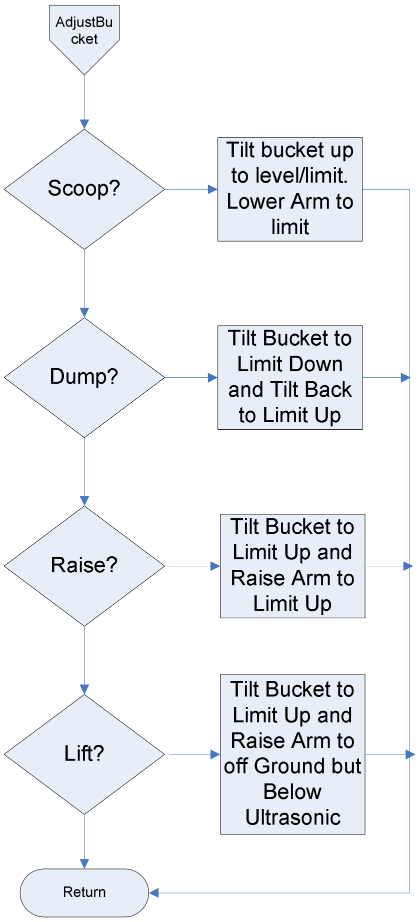


Figure 12: adjustBucket

Orientation Information: Compass

The compass function uses the serial peripheral interface to communicate with the compass. The function sends out a request for a compass reading then waits until the buffer is filled. At this point the buffer is emptied and the function awaits the next 8 bits of the compass reading. The next 8 bits are then shifted in to get the entire compass reading.

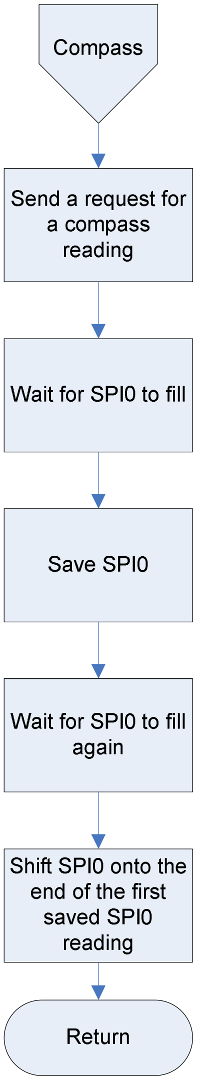


Figure 13: Compass

Infrared Reflective Sensors: Straight and External Interrupts

The Straight function is used when the vehicle is going straight to make sure that both tracks move at the same rate. It uses the counts from both tracks' infrared reflective sensors and calls setPWM to adjust the left track as needed to ensure equal operation. The encoders each trigger an external interrupt that increments a counter which is cleared when the left track is adjusted.

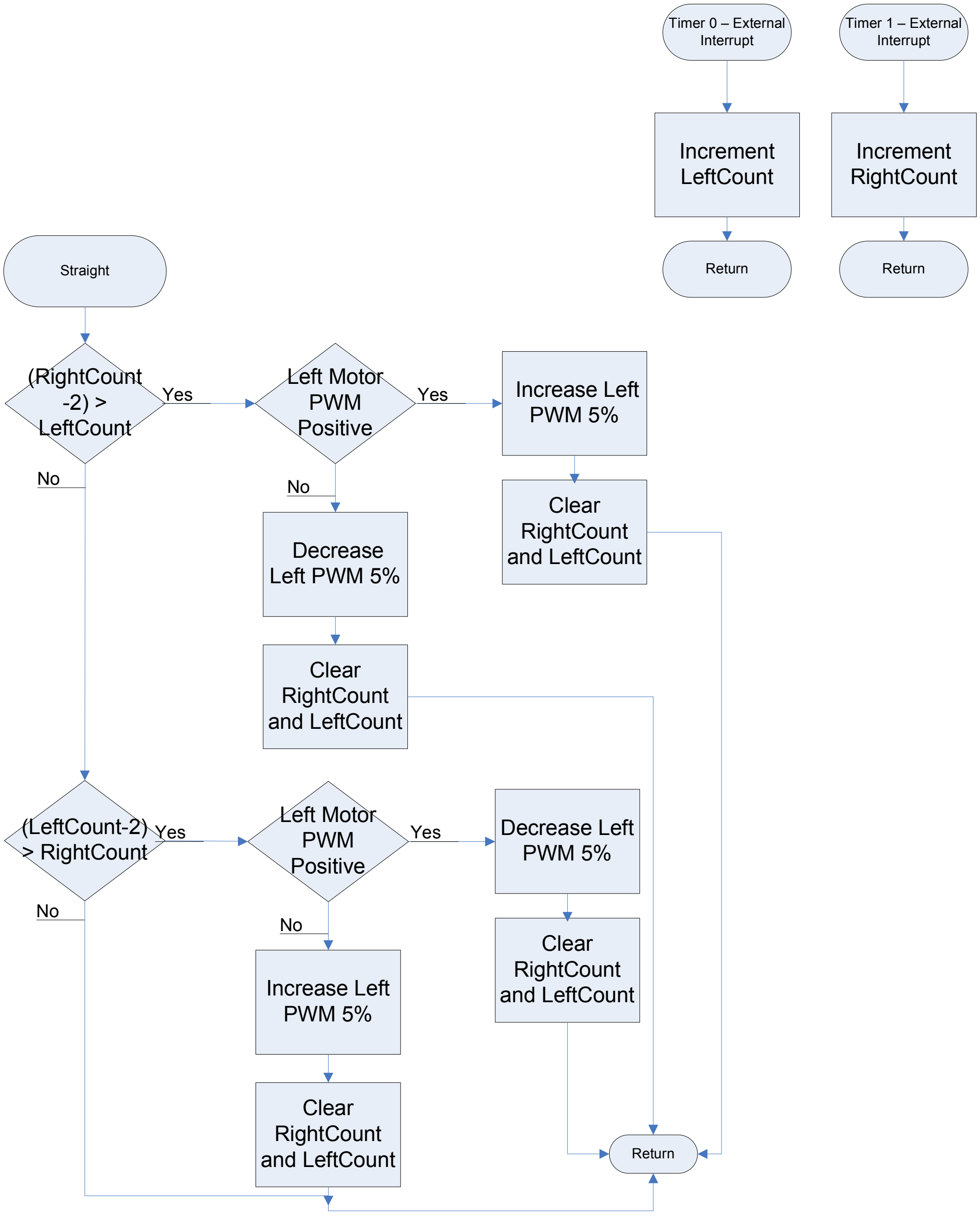


Figure 14: Straight and External Interrupts

Analysis of Results:

For the one and a half semester project, a majority of the time was spent working on the hardware. The first half semester was spent investigating which sensors would work best for this application. The most difficult sensors to integrate were the infrared beacons and transistor. Several high power LEDs were investigated to find the most effective one with the largest range of operation. A current to voltage converter with gain is used to increase the range.

Another additional time consuming task that took longer than expected was the circuit board design. This included wire wrapping and soldering to make all the necessary connections as indicated in Figures 3, 4, and 5.

The development of the software went as expected, but debugging had its difficulties. Some of the issues encountered were due to the SiLabs IDE. Watching the variables in real time was difficult due to the limited capabilities of the watch list within the SiLabs IDE. The most problematic issue encountered involved poor connections with some sockets. This resulted in code working sporadically. After significant troubleshooting to determine poor connections were the problem, new sockets were placed into the old sockets so that the chips could be better seated on the board. Rather than severing connections and replacing the old sockets, it was decided that by inserting new sockets into the old sockets, secure chip seating could be achieved. This solved a large portion of the software issues.

One of the biggest challenges in developing the software was developing code that aligned the vehicle correctly with either the load or the truck. It was desired that the vehicle have a more advanced and robust navigation routine than a straight line routine; however, development of the navigation routine required significant time investment in programming and debugging. The final solution became the newNav function. When the beacons are correctly seen, this operates effectively the majority of the time. While there are minor improvements that could be made with additional time, the vehicle operation is relatively efficient and the code quite robust for handling unexpected and non-ideal situations.

Future Expansion

In future projects, the end loader navigation could more accurately be completed via a camera and image processing. This would allow the vehicle to better determine the locations of the load and truck, as well as removing the human intervention which was required to instruct the vehicle of the direction to approach the load and truck. In addition, by utilizing image processing, this project could be placed onto a full scale vehicle. While this would require significant safety precautions, it would model a more realistic situation.

The project would also benefit from cooperative autonomous agents. An additional vehicle to push the load into a pile would enhance operation. Automating the truck to drive to a predefined location once loaded would also improve the project. By incorporating multiple vehicles, the project would more accurately model the entire process.

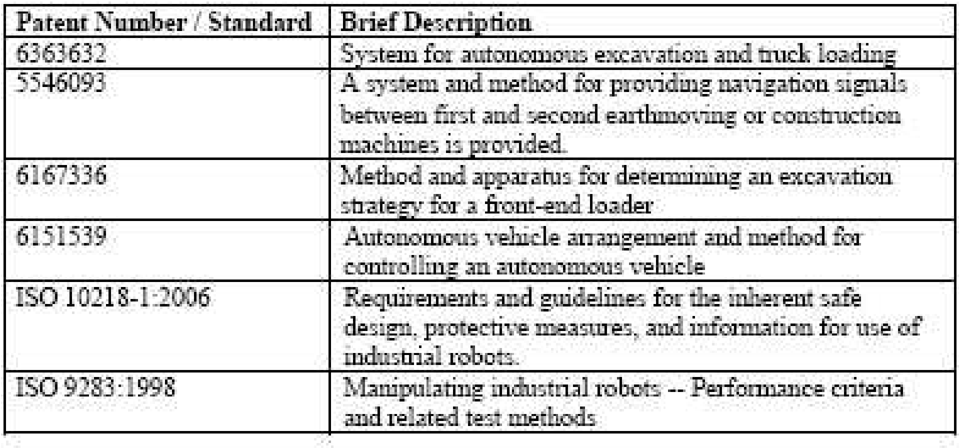
Another improvement on the project would be to develop a web-based control system. Combined with the cooperative vehicles, the user could track the quantity of material moved over a period of time, as well as providing a means to alter vehicle operations in real time.

References, Patents, and Standards:

References: Project success also made possible by contributions from advisor Dr. D.R. Schertz and Mr. Nick Schmidt.

Patents:

The search for patents and standards revealed little directly related to the project. Patent 6363632 is very similar to this application; however, patent 6363632 is for a larger vehicle – an application that the vehicle developed for this project would hopefully be able to develop into with additional modifications. The system in patent 6363632 utilizes a laser or radar scanning system whereas this system will likely use infrared beacons to determine positioning, so the systems do not significantly overlap.



Patent Number / Standard	Brief Description
6363632	System for autonomous excavation and truck loading
5546093	A system and method for providing navigation signals between first and second earthmoving or construction machines is provided.
6167336	Method and apparatus for determining an excavation strategy for a front-end loader
6151539	Autonomous vehicle arrangement and method for controlling an autonomous vehicle
ISO 10218-1:2006	Requirements and guidelines for the inherent safe design, protective measures, and information for use of industrial robots.
ISO 9283:1998	Manipulating industrial robots -- Performance criteria and related test methods