Michigan Tech

Michigan Technological University

"Charlie"

April 28th, 2016

TEAM INFORMATION		
Captain	Phillip Miller	phmiller@mtu.edu
	Brian Terry	bmterry@mtu.edu
	Brian Wilder	bawilder@mtu.edu
	Haden Wasserbaech	hwasserb@mtu.edu
	Trevor Peffley	tbpeffle@mtu.edu
	Noah Symanzick	njsymanz@mtu.edu

I hereby certify, as the faculty advisor, that the design and engineering of the vehicle outlines in this report to be entered in the 2016 Intelligent Ground Vehicle Competition has been significant and equivalent to what might be awarded credit in a senior design course

Glen Archer

Dr. Glen Archer, Faculty Advisor

TABLE OF CONTENTS

CONDUCT OF DESIGN PROCESS, TEAM IDENTIFICATION AND TEAM ORGANIZATION

INTRODUCTION

Michigan Tech's entry into the Intelligent Ground Vehicle competition is designed within the Blue Marble Security Enterprise as part of the Engineering Enterprise program. The design of this vehicle was started in January 2015, meant to replace the aging vehicle previously used in the competition. The team is composed of undergraduate students ranging from sophomore to senior status within the Electrical and Computer Engineering, Mechanical Engineer, and Computer Science departments.

ORGANIZATION

The team is organized on the basis of administrative responsibilities and technical responsibilities. The three main administrative roles are Project Manager, Documentation Chief, and Financial Manager. The Project Manager is tasked with overseeing team organization, planning, and ensuring that the team stays on the critical path. The Documentation Chief ensures that the entire design process is properly documented, manages repositories, and ensures that all paperwork required from the department are completed and filed properly. The financial manager ensures that out budget is properly balanced and seeks out additional funding if necessary.

Technical responsibilities are given based on class standing, background knowledge, project familiarity and interests. Generally, the more senior members are paired with the junior members to ensure that knowledge is passed down appropriately.

Last Name	***First Name***	***Email***	***Administrative Role***	***Technical Role***
Miller	Phillip	phmiller@mtu.edu	Project Manager	Navigation Mapping
Peffley	Trevor	tbpeffle@mtu.edu	Documentation Chief	Image Processing
Wasserbaech	Haden	hwasserb@mtu.edu	Financial Manager	Navigation Mapping
Symanzik	Noah	njsymanz@mtu.edu	N/A	Systems/Power
Wilder	Brian	bawilder@mtu.edu	N/A	Goal Selection Navigation
Terry	Brian	bmterry@mtu.edu	N/A	Image Processing

Table 1.1: Membership Information and Roles

DESIGN ASSUMPTIONS AND DESIGN PROCESS

Project requirements were initially derived based on the IGVC 2015 Rulebook. Additional requirements were gathered based on what was learned from our previous entry, physical constraints imposed on us by our laboratory setting, and monetary constraints. The design history of this vehicle is outlined in the chart below.

Spring 2015	Initial Requirements Capture
	Part Selection
	Chassis Design

	Electronic Design
	Mechanical and Electrical Validation
Fall 2015	Wireless Controller Redesign and Validation
	Sensor Node Development
	Mapping Development
	Pathfinding/Obstacle Avoidance Development
Spring 2016	Pathfinding Validation
	Image Processing Development
	Goal Selection Development and Validation
	Control Algorithm Development
Summer 2016	Vehicles First Competitive Attempt

Table 1.2: Vehicle Development History

INNOVATIONS

This vehicle's design addresses several of the issues seen with our previous design. The previous design had significant mechanical modifications from its conception to its last competition. Because of this, many of the electronics were not easily accessible, making maintenance difficult. To prevent this from happening, the chassis was designed with 80/20 aluminum, making the any future chassis modifications relatively easy if needed. Electronics were placed inside of a pelican case. This case is only partially mounted to the chassis and can be easily removed for bench testing or if chassis maintenance is required. This case also has the added benefit of significantly improving our water-proofing.

TECHNOLOGIES

We have also chosen to use Harbrick's PolySync middleware platform as the basis of our software system. The system handles all back-end communication between nodes, provides critical data logging and system replay, and provides plug-and-play capabilities with various popular sensors out-of-the-box. Given the fact that the software has just come out of beta, this is more than likely the first vehicle at IGVC to be running PolySync.

MECHANICAL DESIGN

OVERVIEW

The chassis was design to comply with IGVC Competition rules as well as our physical constraints in our laboratory. These requirements, as gathered from the IGVC Rules, are outlined in the table below.

Requirement	**Details**
Vehicle Type	Ground Vehicle; Must have direct contact to the ground
Length	Minimum: 3ft Maximum: 7ft
Width	Minimum: 2ft Maximum 2ft 8in (Doorway Width in Laboratory)
Height	Maximum: 6"

Table 3.1: Mechanical Requirements

STRUCTURE AND HOUSING

The structure of the vehicle (Figure 3.1) is composed of 80/20 and aluminum sheeting when appropriate. This material was chosen based on its light-weight properties and modularity. A three-wheeled design was chosen with two 29” bicycle wheels in front and an unpowered caster wheel in the rear. This is based on the previous design, which had no issues with mobility, and was chosen for this vehicle because of its mechanical simplicity.

Most of the electronics are held in a water-proof pelican case. Due to space constrictions, a few electronics reside in their own cases outside of the pelican case.

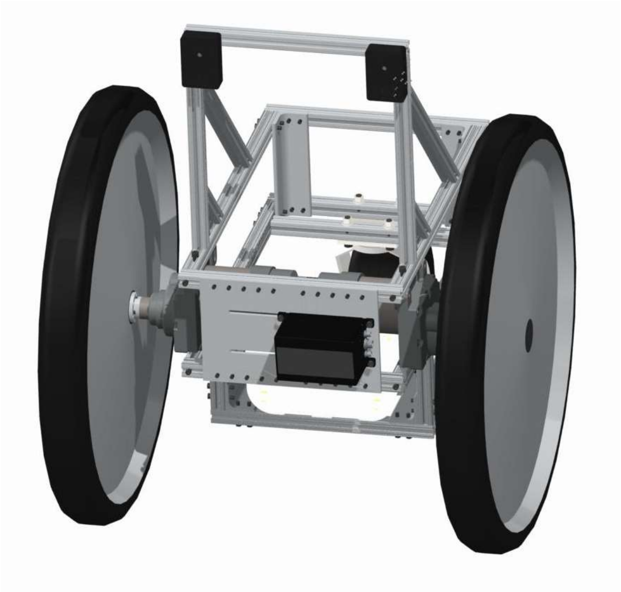

Figure 3.1: Vehicle Structure Model

SUSPENSION

This vehicle currently has no suspension. Our previous experiences have suggested that suspension is not needed.

WEATHER PROOFING

A majority of electronics are placed in a waterproof pelican case. Some smaller electronics are currently outside of the main case in their own cases. A field monitor is also outside of the waterproof case. In the case of rain, this monitor would be wrapped in plastic to prevent damage.

ELECTRONIC AND POWER DESIGN

OVERVIEW

Our electronics system was designed with the chief criterion being ease of implementation and power efficient. We ultimately decided to take advantage of polysyncs powerful communication backend and design the system with a distributive computing model in mind.

POWER DISTRIBUTION

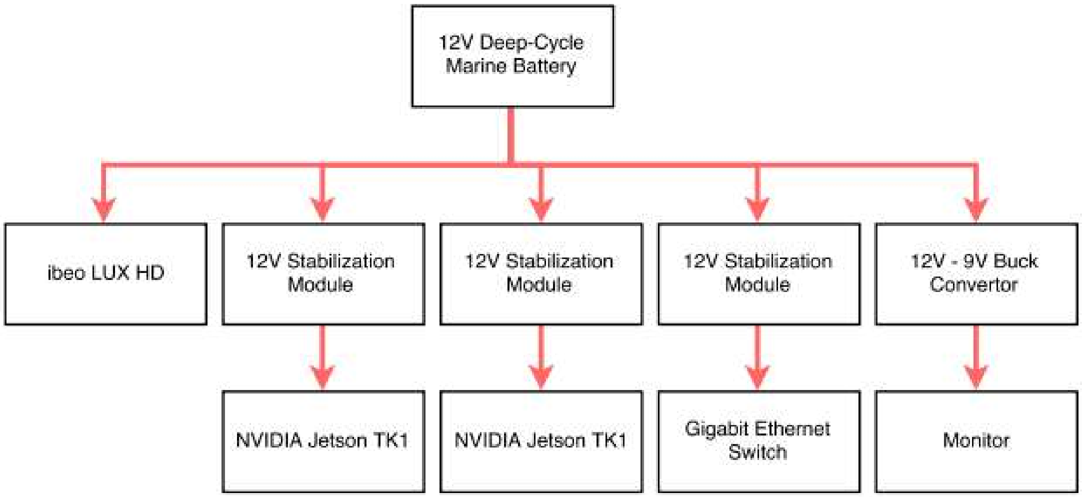

Figure 4.1: Power Distribution

The vehicle is powered by a 12V Deep-Cycle marine battery. More sensitive electronics are protected with stabilization modules. With our current setup, we expect a typical run time of just under 2 hours on a consistent run. However, empirical testing has shown runtimes closer to 6 hours.

Component	Min	Typ Current	Max
IBEO Lux		0.666666667	0.833333
Left Motor		9.6	32
Right Motor		9.6	32
Top Jetston	0.251667	2	5
Bottom Jetson	0.251667	2	5
Monitor	2.1	2.1	2.1
Ethernet Switch	1	1	1
Stack Light	0.5	1	3
Left Raspberry Pi 2	0.22	0.5	1.8
Right Raspberry Pi 2	0.22	0.5	1.8
Left Encoder		0.029	0.033
Right Encoder		0.029	0.033
Phidget Spatial		0.055	0.055
Jetson BuckBoost Converter Loss		0.11	0.556
Jetson BuckBoost Converter Loss		0.11	0.556
Totals	4.543333	29.29966667	85.76633

	Min	Typ	Max
Runtime (Hrs)	12.10565	1.877154461	0.641277

Figure 4.2 Estimated Run Time

ELECTRONICS SUITE

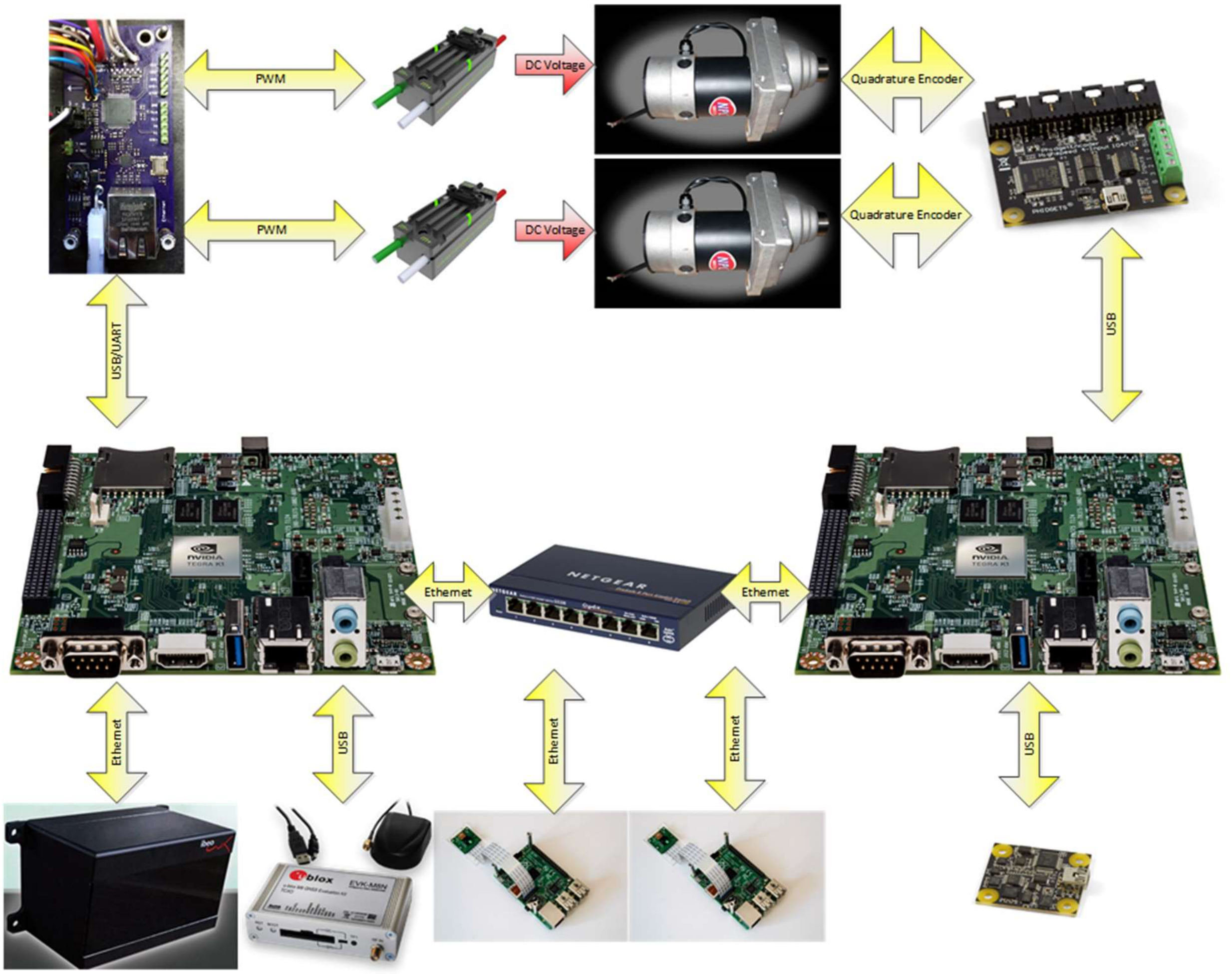

Figure 4.3: Vehicle Electronics Suite

Our electronics suite (figure 2) follows a distributed computing modeling in which multiple nodes exist on a network capable of communicating with each other through a common framework.

NVIDIA JETSON TK1

The Nvidia Jetson TK1 was picked for its powerful and efficient chipset (Tegra K1) and its mobile friendly platform. It features a quad core ARM Cortex A15 CPU as well as a separate single core low power CPU that is great for low power consumption. It also has a 192 core Kepler Mobile GPU with CUDA support and OpenCV specific functions for fast image processing. It runs off of 12 volts which is easily obtained from the onboard battery after going through a voltage stabilization module to hold it at a constant 12 volts. The many peripheral interfaces make it easy to get and share data with other nodes. We currently employ the Ethernet, USB, UART, and I2C interfaces.

IBEO LUX HD

The ibeo LUX HD is a 4 layer, 110° field of view laser range finder. Data from the ibeo can be transferred by both Ethernet and CAN. Our current implementation employs the Ethernet interface but may be expanded to the CAN Bus if necessary. The Ethernet interface routes directly to a Jetson for a dedicated data link. This allows fast data access. It can be powered off of the 12 volt battery directly.

RASPBERRY PI 2 CAMERA’S

Image acquisition is fulfilled by two Raspberry Pi B2 single board computers with the CSI MIPI camera add on boards. The CSI interface allows quick and undelayed access to the 5 MegaPixel sensor unlike most USB cameras. Because the Raspberry Pi is a computer there is a possibility of doing pre-processing before sending the data back to the Jetsons. This will minimize the amount of data that needs to be sent and improve frame rates. The Raspberry Pi’s connect back to the Jetsons through a gigabit Ethernet switch.

EVK-M8N

The ublox EVK-M8N is a GPS evaluation kit based on the ublox M8 chipset. This module has multiple interfacing modes including an onboard USB to UART adapter allow easy connection and integration with linux. The unit can be powered straight from the USB port and therefore eliminates the need for an external power supply. The included active antenna provides a stronger and more accurate GPS fix over the passive ceramic antenna used in many low cost modules.

PHIDGET SPATIAL IMU

The phidget spatial IMU is a low cost 9 DOF IMU that connects over USB. It contains low and high sensitivity 3 axis accelerometers, gyroscopes, and magnetometers. The IMU is easily integrated in linux through the provided driver and sample code.

GIGABIT SWITCH

Due to the vast amount of data that needs to be transferred around, a gigabit Ethernet switch was picked in order to quickly and reliably get data from point A to point B. Either the 5 port (GS105) or 8 port (GS108) may be used depending on the number of devices that need to be connected. This is a key feature for the expandability of the project.

CUSTOM INTERFACER

We needed something to take drive commands from the Jetsons and turn that into motor control commands. There are not too many options on the market that are affordable when it comes to these sorts of interfaces so we decided to design our own. The heart of the interfacer is a PIC32 microcontroller that accepts drive commands from the Jetsons over an Ethernet or UART connection and then give the proper signals to the motor controllers. The most popular motor controller interfaces are CAN bus and PWM so we implemented both in our design. To allow proper control of the motors, this board also pre-charges the flux capacitor at startup ensuring maximum performance. Since this board will be the central point of motor control, we decided that it will handle the manual control as well through a dedicated wireless module. This allows manual control even when the Jetsons are not powered on and the ability to take over control in case the Jetsons issue unsafe drive commands. We also decided to implement the light controller into this board as the light status is directly based off of the manual control status.

VICTOR SP MOTOR CONTROLLER

The motor controllers have a small form factor and have passive cooling while maintaining high performance. It has a wide input voltage (6-16 volts) that is a perfect fit for our 12 volt system. They can handle 60 Amps continuous current draw and are controlled over a PWM interface.

MOTORS

The NPC-T64 motors were chosen based on the teams prior experience and their built in gearbox. The gearbox has mounting points that allow it to be easily integrated into the mechanical design. The motors are rated for 24 volts but provide ample power when run at 12 volts.

DC-DC USB CONVERTERS

Because a battery does not have a constant voltage over its discharge cycle, a DC to DC converter with a buck-boost topology is needed to provide a stable 12 volt supply to the Jetson TK1's. These converters provide 100 watts, or about 8 amps which is more than the Jetsons will pull.

ENCODER INTERFACE

The phidget encoder interface provides an easy means of processing the data from our motor encoders through the means of off-the-shelf hardware and an easy-to-use built-in library.

12V BUCK-BOOST CONVERTER

As the two Jetson boards incorporated in Charlie's design require 12V to run, a 12V Buck-Boost Converter was necessary in order to adjust any input voltage from the battery to 12V. The battery has a range of 9V when it's low on power to 15V when fully charged. The team researched into the cost of buying a ready-made buck-boost converter and found one that was $60. Some members of the team investigated into building a converter that would be cheaper, but the cost of parts and the time needed to build and test a converter was too high. The choice was therefore made to purchase the DCDC-USB, Intelligent DC-DC converter with USB interface on Amazon at a cost of $58.49. Three Converters were purchased, one for each Jetson board and another for the Ethernet switch

SAFTEY DEVICES

BLINKING LIGHT

The blinking light design that was used for Bishop had many good aspects so this design was improved upon for the Charlie.   Three lights, one red, yellow, and green, were used again Charlie. The multiple lights allow the team to communicate more from robot when the robot is at a distance from the team. A new board was made that would allow for easier interface with the lights. Soldering the new board allowed members of the team to gain valuable surface mount experience

WIRELESS EMERGECY STOP

Our wireless emergency stop is built into a modified RC Helicopter controller. Cutting off power to this device will immediately stop the vehicle.

MECHANICAL EMERGENCY STOP

A mechanical emergency stop is mounted facing the rear of the vehicle, as per IGVC rules. Activating this button will cut off power to the motors via a relay.

SOFTWARE STRATEGY AND MAPPING TECHNIQUES

OVERVIEW

Our software systems distributes core functionality across several nodes on several pieces of hardware. Each node is capable of communicating with each other via the PolySync middleware. This multi-node approach allows us to made changes to each core system independently without disrupting development on a different node. This approach also allows us to take advantage of the NVIDIA Jetson TK1s exceptional multi-tasking abilities. This node network is summarized in the figure below.

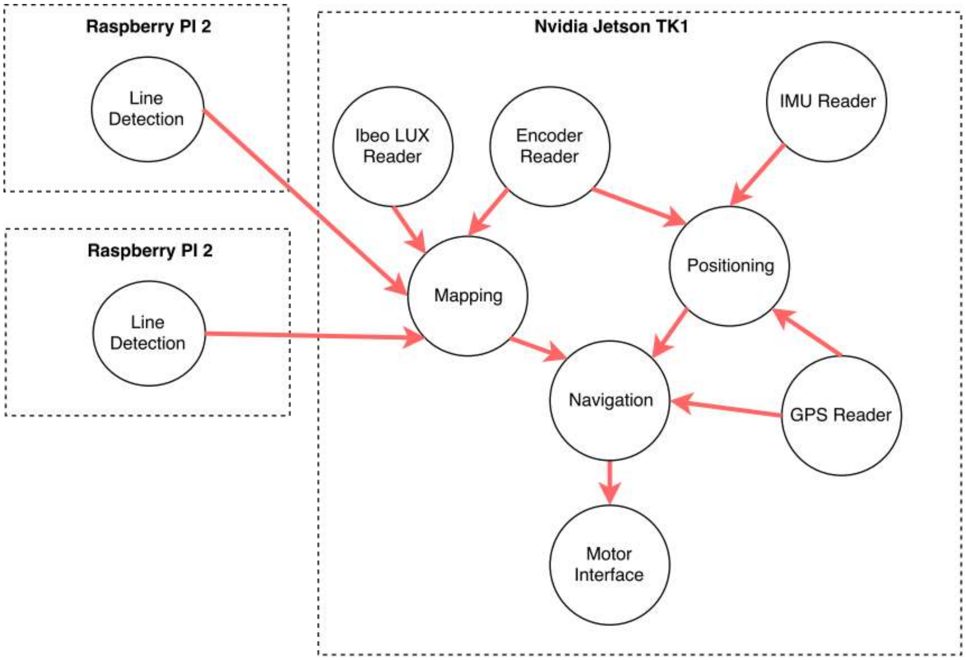

Figure 5.1: Network Design

Given how the higher level nodes such as navigation and mapping rely heavily on the lower-level "reader" nodes, our design employed a bottom to top design approach where the core functionality of each lower-level node was stress-tested and validated before the development of the higher-level nodes that rely on the integrity of the data coming out those nodes.

OBSTACLE DETECTION

The ibeo LUX HD LIDAR provides with a tremendous advantage with its embedded ground filtering and echo detection. As a result, the raw data obtained from our LIDAR has exceptional accuracy and reliability. Our obstacle detection system checks for reoccurring LIDAR points and calculates the probability of an obstruction being at the point as a function of the amount of points measured within a given radius of the point.

$$P_{obj}(x,y) = \frac{Hits \quad ,y)}{Misses(x,y)}$$

Where *Hits* is the number of a times a liDAR point near x and y and *Misses* is the number of times a liDAR point was not found near x and y. This method does a sufficient job of identifying stationary obstacles. The addition of a range around each point results in the detected obstacles having significantly larger size than they actually do. This is done purposely to minimize our risk of collision.

MAP GENERATION

CAMERA INTEGRATION

Our camera software uses a multi-step process to detect white lines and send the corresponding line positions to the mapping node. The image stream from the raspberry pi camera is initially run through a color filter, resulting in an image where only white objects are visible.

The processed image is then run though a Canny edge detector to extract only the edges of white objects. The edges detected are then run through a probabilistic Hough Transform.

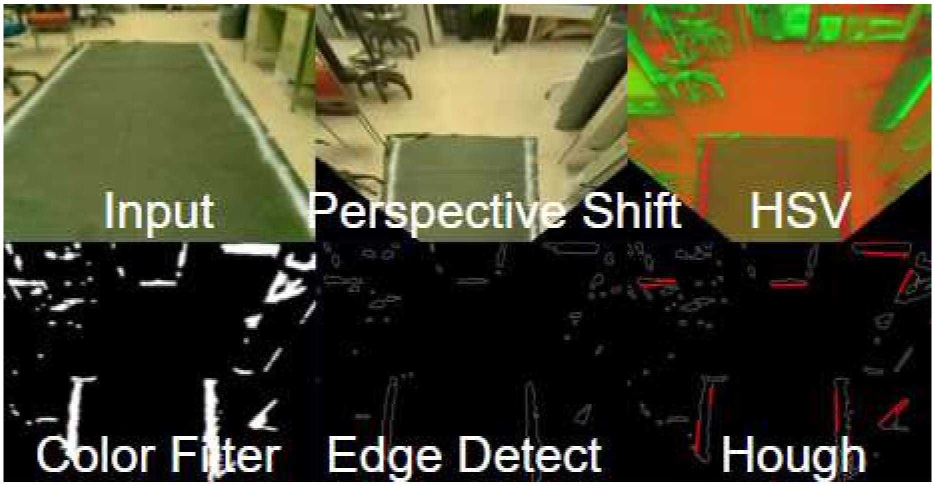

Figure 5.2: Image Processing System

OCCUPANCY GRID

The objects detected from the liDAR data are overlaid onto the lines detected from our cameras. These systems are married together into their final form as an occupancy grid. This occupancy grid is published via PolySync and later used by the path generation node.

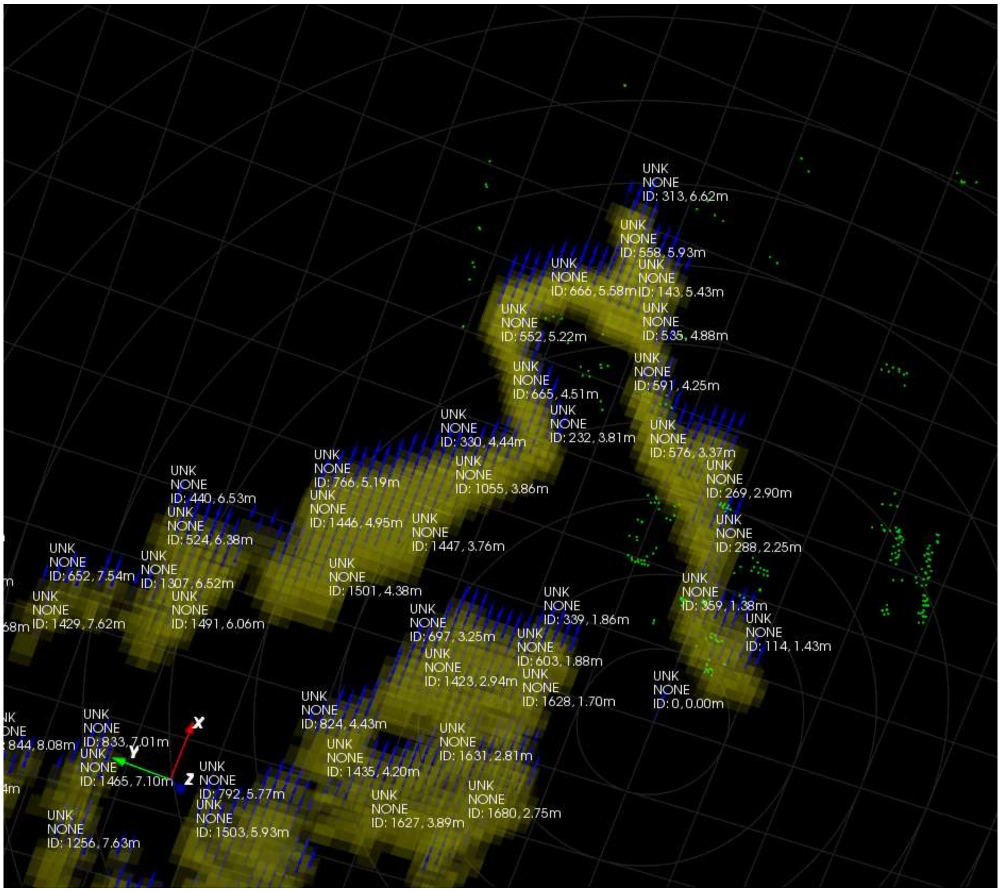

Figure 5.3: Vehicles Occupancy Grid as shown on the PolySync viewer software

GOAL SELECTION AND PATH GENERATION

Goals are selected by running through an ordered list of GPS waypoints stored in a text file. The selected GPS waypoint's location given in degrees longitude and degrees latitude and converted into a Cartesian coordinate using the Haversian formula system with the origin corresponding with the vehicle's initial location.

These points are then inputted into a modified A* algorithm that utilizes the occupancy grid generated from the mapping node to determine the shortest path between the vehicle and the GPS waypoint. The resulting path is stored and the vehicle is commanded to move along the generated path.

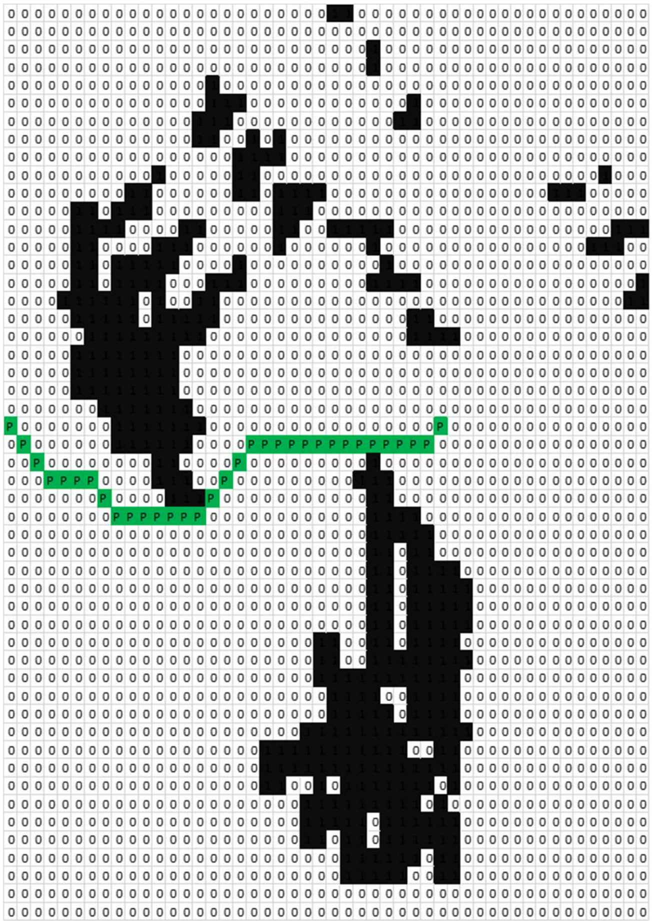

Figure 5.4: A Path Generation Simulation using actual map data*

FAILURE MODES, FAILURE POINTS AND RESOLUTIONS

SOFTWARE FAILURES AND RESOLUTIONS

The most likely source of failure in software is an error in the data that is sent into the mapping node. If the data points do not properly correspond with their real-life positional relationship with the vehicle, then we are prone to collisions with obstacles. The address this concern, we apply probabilistic filters to all data coming of the nodes that feed into the mapping node to help eliminate any extraneous data. Our mapping node also creates a “no-fly” zone around all obstacles and lines. This gives our path-following algorithms a large margin for error and minimizes the effects of both failures with our control algorithms and mapping nodes.

MECHANICAL AND ELECTRICAL FAILURES AND RESOLUTIONS

Our chassis has several redundant supports to minimize the effects of any mechanical failure in the chassis.

All devises in our system are properly connected with fuses to minimize the possibility of an electrical failure from damaging our more sensitive electronics.

Our sensitive electronics are all enclosed within the body of the vehicle with the exception of our LiDAR, which is enclosed between the two front wheels. In the case of a motor lockup or a control failure, all sensitive electronics are properly protected from damage.

FAILURE PREVENTION STRATEGY

While we have planned for more redundant safety features, we lack the human and monetary recourses to implement them at this stage in the vehicle's design. Currently, our chief failure prevention strategy is less about preventing failures, but more minimizing the risk of failure. To do so, we have design each system with exceptionally large margins for error at the expense of vehicle efficiency. Our navigation node will also cease operation if any of the nodes that it relies on fail.

PERFORMANCE TESTING TO DATE

Because we focused on the core systems, our testing focused on competition of a qualifier-like course. This course consisted of a 5 meter long lane indicated by two perpendicular white lines separated by five feet painted on artificial grass and a single barrel.

Additional navigation tests were performed to test the robustness of our pathfinding system by giving the vehicle a waypoint 10 meters in front of it and placing three to five barrels in its path.

INITIAL PERFORMANCE ASSESMENT

Our newly design vehicle is already outperforming the previous design—which struggled to even qualify even after three years of development. We do not expect the vehicle to perform well during the auto-nav portion of the competition as the vehicle is still in its infancy.

Early testing does suggest that a chassis modification might be necessary to either redistribute weight off of the caster wheel or move over to a four-wheeled design. Both of these modifications would aid in the vehicles mobility and ease the burden on future control development.