Lawrence Technological University

H_2Bot II

IGVC 2007 Autonomous Vehicle

Team Members

Marcus Randolph, Brace Stout, Gary Givental

Faculty Advisor Statement(*)

I, Dr. CJ Chung of the Department of Math and Computer Science at Lawrence Technological University, certify that the design and development on H_2Bot II has been significant and each team member has earned credit hours for their work.

Dr. CJ Chung (chung@ltu.edu, 248-204-3504) Date

(*) Co-advisors: Dr. Lisa Anneberg, Dr. Peter Csaszar, Dr. Robert Fletcher, Prof. Maurice Tedder, and Dr. King Man Yee

1. Introduction

For the 2007 Intelligent Ground Vehicle Competition (IGVC), Lawrence Technological University (LTU) presents H_2Bot II, the product of a collaborative effort among a diverse and multidisciplinary group of LTU engineering and computer science students. H_2Bot II hardware is based on the 2006 H_2Bot and the software system was completely re-written for the 2007 IGVC. This paper describes the various aspects of H_2Bot design, design trade-off considerations, and improvements over LTU's past IGVC entries.

2. Innovations

H_2Bot II is one of the first fuel cell powered robot vehicle in the world. H_2Bot II has improved hardware, control and sensor software, and simulation software for the 2007 IGVC competition. The following sections present these changes in detail.

2.1 Hardware Updates

The H_2Bot II has received an upgraded vision system, utilizing a 3 CCD camcorder for image capture. The generated frames exhibit superior color separation as compared to the 2006 IGVC camera. The resulting increase in image quality results in improved detection and isolation of scene components for the 2007 H_2Bot II. A protractor was introduced to accurately set the angle of the camera.
The location of the GPS antenna has been moved from the main body of the vehicle to the mast. This location offers improved reception of GPS signal over the 2006 H_2Bot.
This location offers improved immunity from magnetic fields generated from the hydrogen fuel tank and computer components of the H_2Bot II.

2.2 Control and Sensor Software Updates

Control and sensor modules for the H_2Bot II have been improved for 2007, primarily addressing issues that came to light during the 2006 IGVC competition. These improvements include:

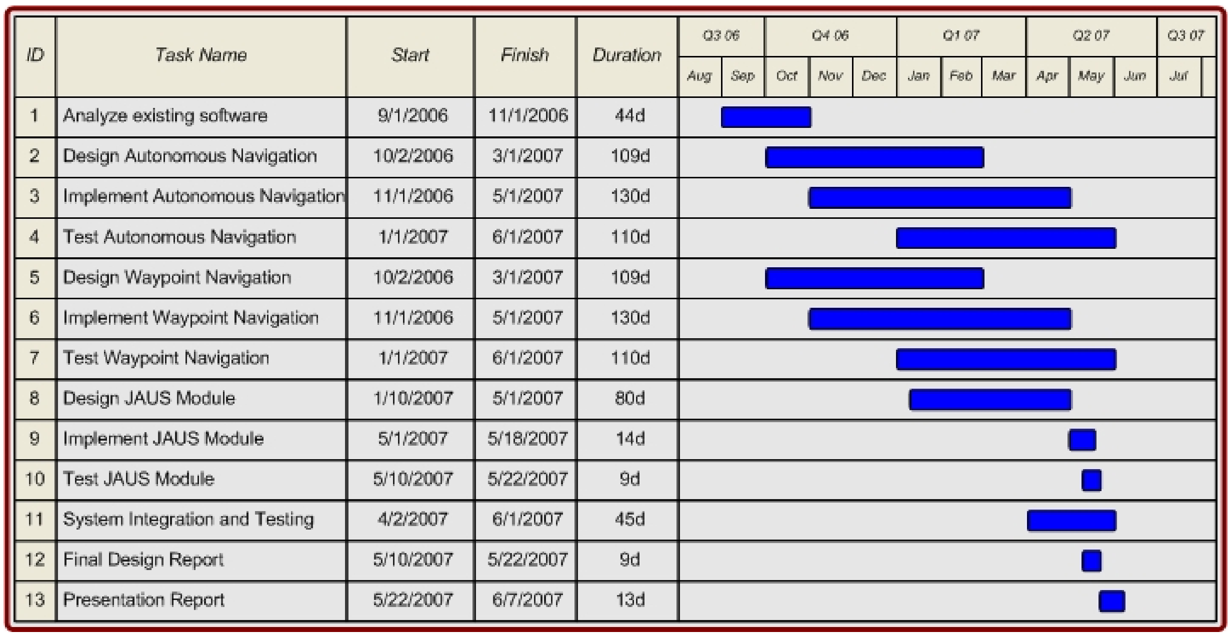
- Motor Controller Interface – The motor controller interface was rewritten to reduce coupling and improve real-time performance through the introduction of threads and associated synchronization mechanisms.
- Laser Measurement System Interface – The laser measurement system interface was rewritten to be more tolerant to communications errors.
- The 2007 H_2Bot II attempts to recover from communications errors by re-synchronizing with the protocol, rather than initiating a time-consuming process of resetting the hardware, which was the 2006 H_2Bot behavior.
- Image Processing - H_2Bot II has improved image processing filters which can use either RGB or HSB color space. H_2Bot II has two distinct vision modes which allow for a more robust system that is easily adaptable.

- Control Algorithms – H_2Bot II has entirely new control algorithms for Autonomous, Navigation, and JAUS challenges. These algorithms focus on immediate, achievable goals, whereas the 2006 algorithms were overly broad in scope and assumed an accuracy of sensors readings that were unrealistic.

3. Design Process

3.1 Project Planning Process

The development for this year's IGVC effectively started in September 2006. **Figure 1** below describes the project plan followed by the team. Another important part of the project has been the L2Bot robot platform developed by Lawrence Tech University that allowed team members to test many different software algorithms for autonomous navigation at home without having access to the actual H_2Bot II platform. The L2Bot is a very simple low-cost robot platform that just allows a laptop with a webcam to control the robot wheels.



ID	Task Name	Start	Finish	Duration
1	Analyze existing software	9/1/2006	11/1/2006	44d
2	Design Autonomous Navigation	10/2/2006	3/1/2007	109d
3	Implement Autonomous Navigation	11/1/2006	5/1/2007	130d
4	Test Autonomous Navigation	1/1/2007	6/1/2007	110d
5	Design Waypoint Navigation	10/2/2006	3/1/2007	109d
6	Implement Waypoint Navigation	11/1/2006	5/1/2007	130d
7	Test Waypoint Navigation	1/1/2007	6/1/2007	110d
8	Design JAUS Module	1/10/2007	5/1/2007	80d
9	Implement JAUS Module	5/1/2007	5/18/2007	14d
10	Test JAUS Module	5/10/2007	5/22/2007	9d
11	System Integration and Testing	4/2/2007	6/1/2007	45d
12	Final Design Report	5/10/2007	5/22/2007	9d
13	Presentation Report	5/22/2007	6/7/2007	13d

Figure 1. Project Tasks

3.2 Development Process

The team used agile development methodologies for redesigning the software for the 2007 IGVC competition. LTU used Set-based concurrent development to investigate several options for software modules and algorithms and found the best solution. Using set-based engineering favors, while carrying multiple design options in the early stages of development, increased the chances of constructing a completely successful system. The team also used the L2Bot robot platform to test various software approaches quickly. Keeping options open allowed for many feedback cycles and iterations to obtain a near-optimal design. **Figure 2** illustrates how initial options gradually narrowed over time, but were not necessarily frozen at the same point in time.

New for 2007 was the development of a high fidelity autonomous vehicle simulator, configured to simulate H_2Bot II sensor, control, and response characteristics. This allowed us to develop and test control algorithms independent of hardware availability.

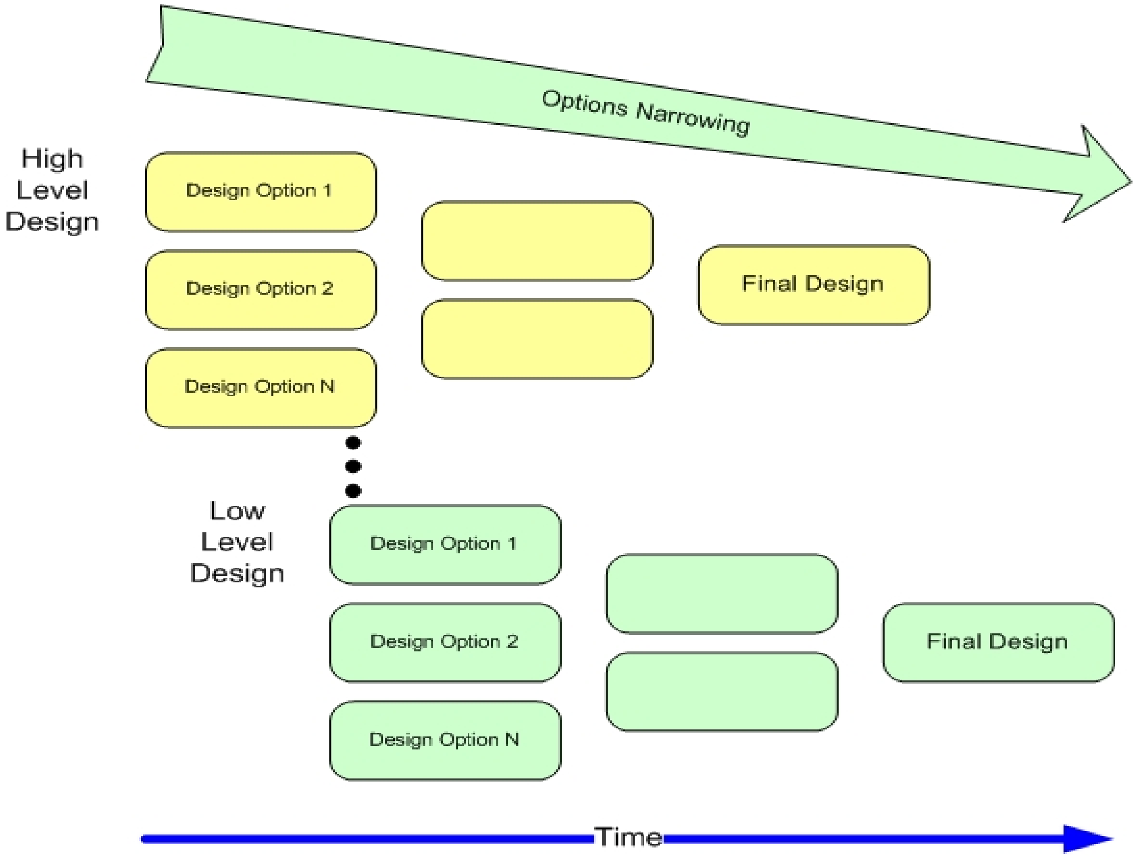


Figure 2. Set Based Concurrent Development Model

3.3 Testing Process

3.3.1 Hardware Testing

A 200 m circular test track was setup on Lawrence Technological University's campus that mimicked possible tracks that H_2Bot II would encounter at the IGVC competition. This allows team members to collect real world data and find cases where software failures occur.

3.3.2 Software Testing

The team performed extensive unit testing, systems testing and integration testing of the code base. The robot was regularly taken for short test runs to make sure the software modules were working properly.

3.3.3 System Integration

There have been no significant changes to the hardware architecture from the 2006 H_2Bot. As such there were no significant systems integration issues for the 2007 H_2Bot II.

3.5 Collaborative Team Organization

This year the team for H_2Bot II was quite small as we mostly focused on software improvements. The team members are:

- Marcus Randolph, MSCS – Team Captain
- Brace Stout, MSCS
- Gary Givental, MSCS

3.6 Vehicle Conceptual Design

For the 2007 IGVC competition, LTU is re-using the H_2Bot platform from the previous year. The H_2Bot was designed to exceed the requirements for structural support, rigidity, reliable component mounting, and ample torque. It employs a modular component design allowing for easy adjustable mounting and accessibility to sensors. Other notable hardware features include the ability of camera mast to break down for transport and the adjustable LMS mount. The software for the H_2Bot has been redesigned to improve on last years design and address the problems encountered in previous competitions.

4. Hardware Design

4.1 Robot Structure

H_2Bot II is a three wheel vehicle with front differential drive steering and a rear caster wheel (as in the 2006 design). This design allows for zero-turn radius maneuverability and simple motion control. The frame shown in **Figure 3** is enclosed in aluminum sheet metal. A removable body covers the power source and is latched in place (not pictured). This body supports the required payload, as well as providing ample ventilation and weather proofing. The camera and LMS ports are also enclosed to protect them from rain.

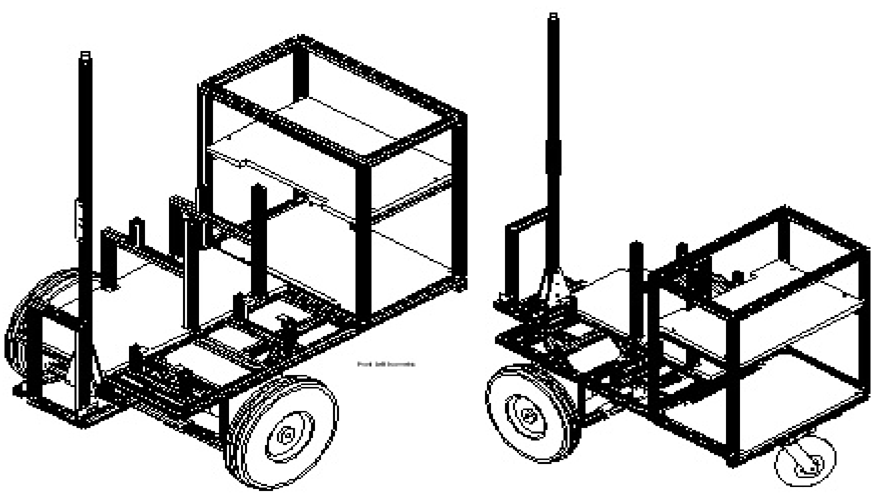


Figure 3. CAD Models of Vehicle Structure

4.1.1 Drive Train

There have been no significant upgrades to the drive train from the 2006 design. The drive train provides 389 Lb-In per front drive wheel and has a typical power consumption of about 250 W.

4.1.2 Motor Control

Motor control is facilitated by a Roboteq AX3500 60A/channel controller with a serial port Java interface. The AX3500 provides velocity feedback measurements through optical encoders mounted on the motor shafts. The E-Stop is wired to the controller main drive power lines via a mechanical relay which can be triggered by the top-mounted kill switch or a wireless control system.

4.1.3 Sensors

In keeping with the theme of modularity, all H_2Bot II sensors (and controls) utilize RS-232 or USB 2.0 standard interfaces. **Table 1** summarizes the sensors used in H_2Bot II.

Sensor	**Function**
Optical Encoders	Motor shaft position feedback for servo control
NovaTel ProPack-LB DGPS unit	Global positioning
Digital Compass/Inclinometer	Heading, roll, and pitch information
High Dynamic Range DV Camera	Capture field image stream
Sick Laser Measurement Sensor	Provides a 180 degree polar ranging array

Table 1. Vehicle Sensor Summary

4.1.4 Vehicle Alert System (JAUS Horn and Lights Hardware)

H_2Bot II includes a module that can switch relatively large electrical loads (using relays) controlled with commands sent over USB interface. This module is utilized by the implementation of the vehicle alert system, which is exercised during JAUS checkout. Power is routed to the horn and lights when activated by JAUS software.

4.1.5 E-stop

H_2Bot II is equipped with both a manual and a wireless (RF) remote emergency stop (E-Stop) capability. The KE100-BFR-CL Remote Keyless Entry System manufactured by Bulldog Security is used as the wireless E-Stop component. The trunk-release function is integrated into the H_2Bot II's E-stop system. Activating the E-Stop removes power from the drive train and engages both front wheel failsafe electromagnetic brakes.

4.2 Electrical System

4.2.1 Power Source

The H_2Bot II has interchangeable power systems, each of which is capable of supplying all of the robot's electrical needs (**Figure 4)**.

The primary module uses a Ballard 1.2kW Nexa Fuel Cell Module as the power source (**Figure 5A**). The fuel cell system houses all components required for operation. This includes the Hydrogen Fuel Cell, Hydrogen tank, Morningstar charge/power controller, leak detector, fuel line, 24V battery pack, and 24/12V DC/DC converter for 12V systems. The fuel cell uses 99.999% pure hydrogen gas, and can safely operate both indoors and outdoors as it produces only water vapor and heat as byproducts of electrical generation. A single tank of hydrogen supplies sufficient fuel for the fuel cell to operate at 100% capacity (1200 watts) for approximately 45 minutes, however, H_2Bot II normally requires only a fraction of that power. The power output of the fuel cell is actively adjusted with onboard load sensing.

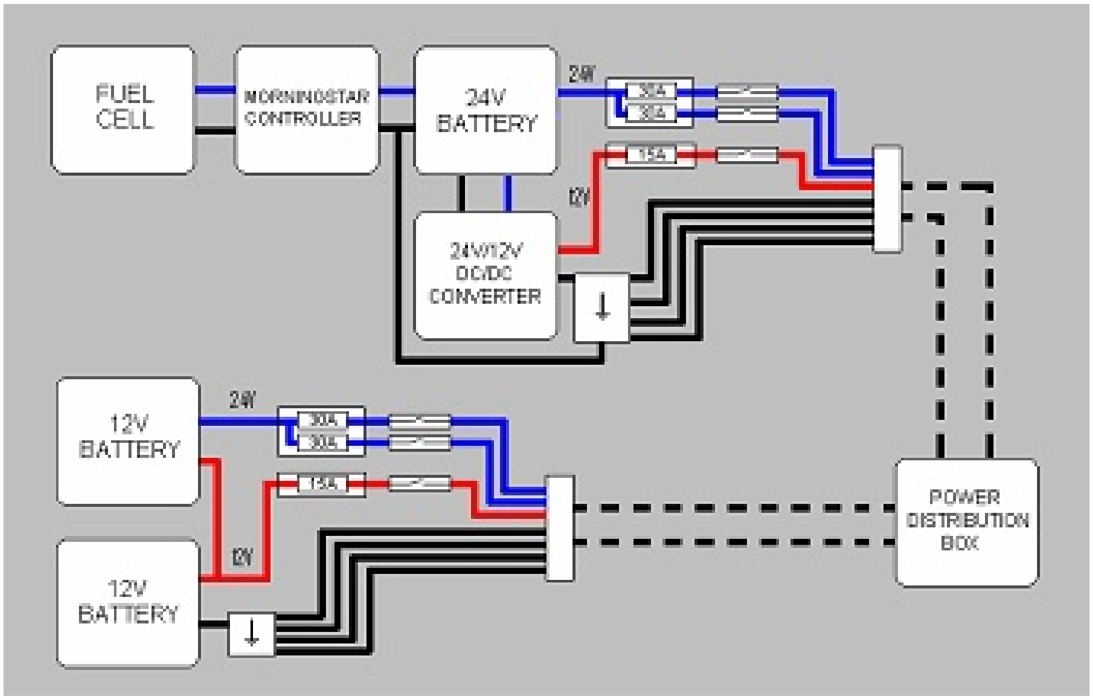


Figure 4. Power Module Schematic

The secondary module consists of two 40Ah AGM Batteries. An onboard charger with 110VAC interface is employed to restore battery power as required.

The installed power module feeds directly to a power distribution box (as shown in **Figure 5B**), which distributes power to the vehicle. Switching power modules is a straightforward process, and can be done in a matter of minutes. The modular design also allows additional power sources for H_2Bot II to be incorporated with a minimum of effort.

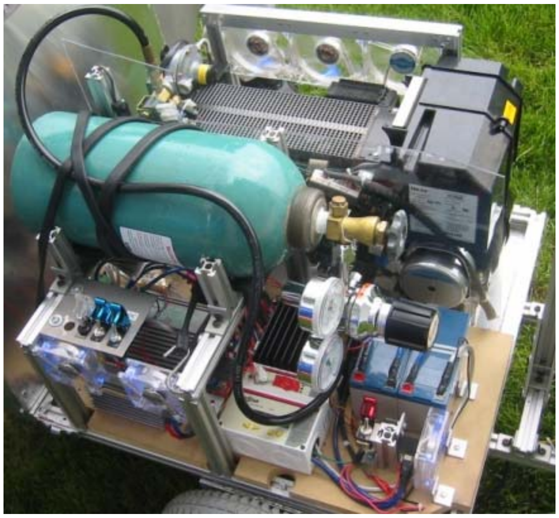


Figure 5A. Fuel Cell

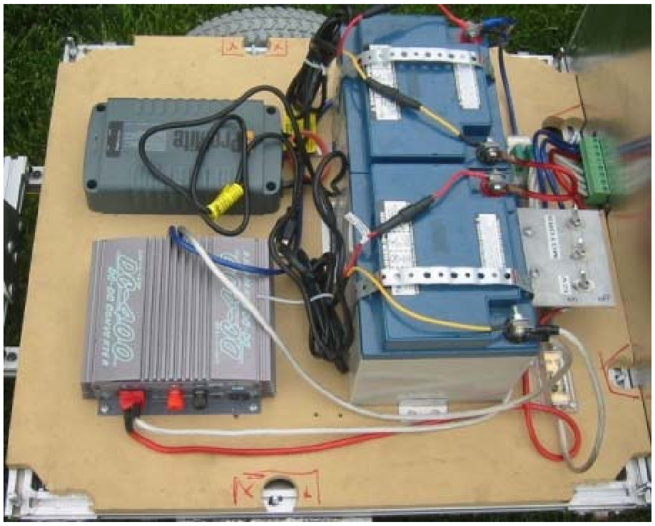


Figure 5B. Battery Module

4.2.2 Power Distribution

A box-mounted printed-circuit board (PCB) distributes and switches power to each of the electrical components. Molex® and Phoenix Contact® connectors provide a quick-connect interface for switching between power modules. A separate wiring harness routes power to the motor controller and the power distribution PCB. The PCB provides connections for the E-stop wireless device and a DC-DC converter to supply regulated 24V power required for sensitive electronics. Toggle switches mounted to the box allow the power to each device to be switched individually. The power and communications control system schematic for H_2Bot II vehicle is shown in **Figure 6**.

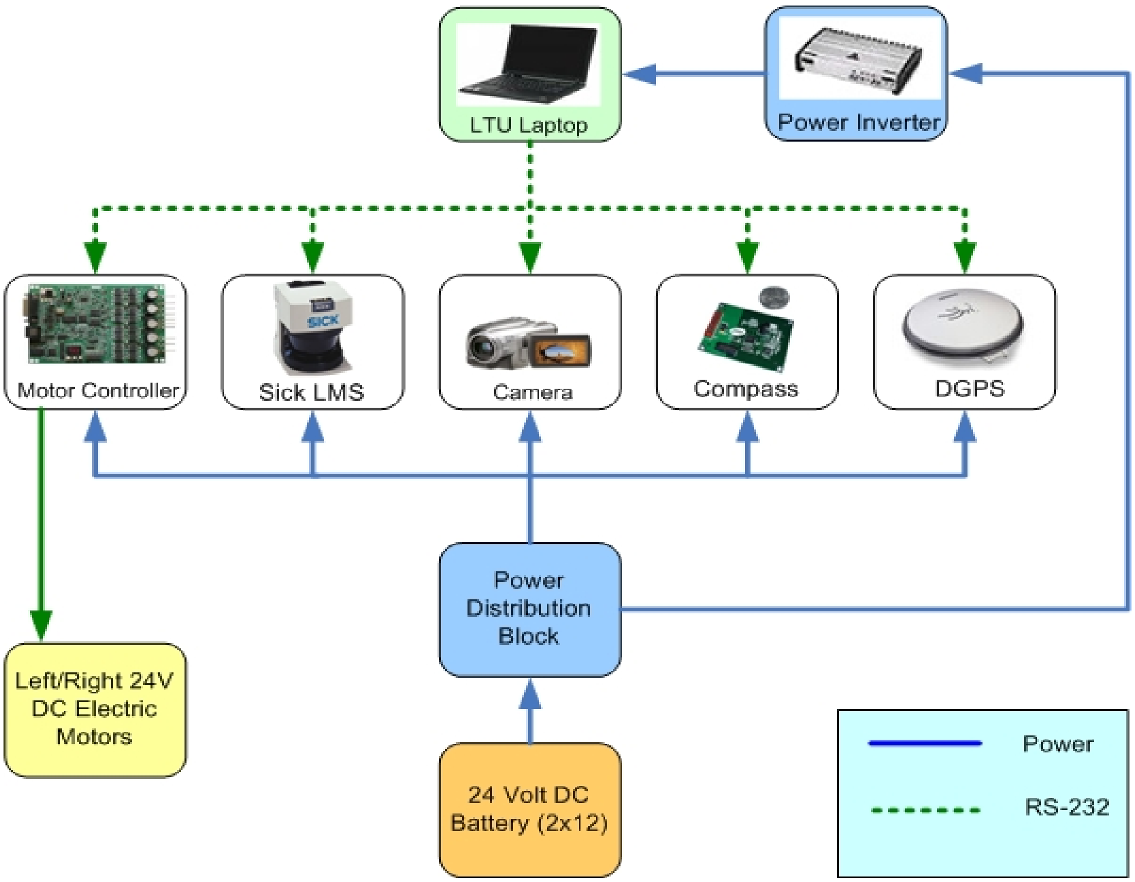


Figure 6. Power and Communications Control System Schematic

5. Software Design

5.1 Software Strategy

The H_2Bot II software expands on the hardware interface layer developed in 2006, continuing the use of the freely available Sun Java programming language. Java provides a flexible, portable, and scalable approach to multi-threaded software development, all of which are important as the team continues to build on previous work. **Figure 7** shows the software architecture for autonomous challenge. Software architecture for navigation challenge is shown in **Figure 8**.

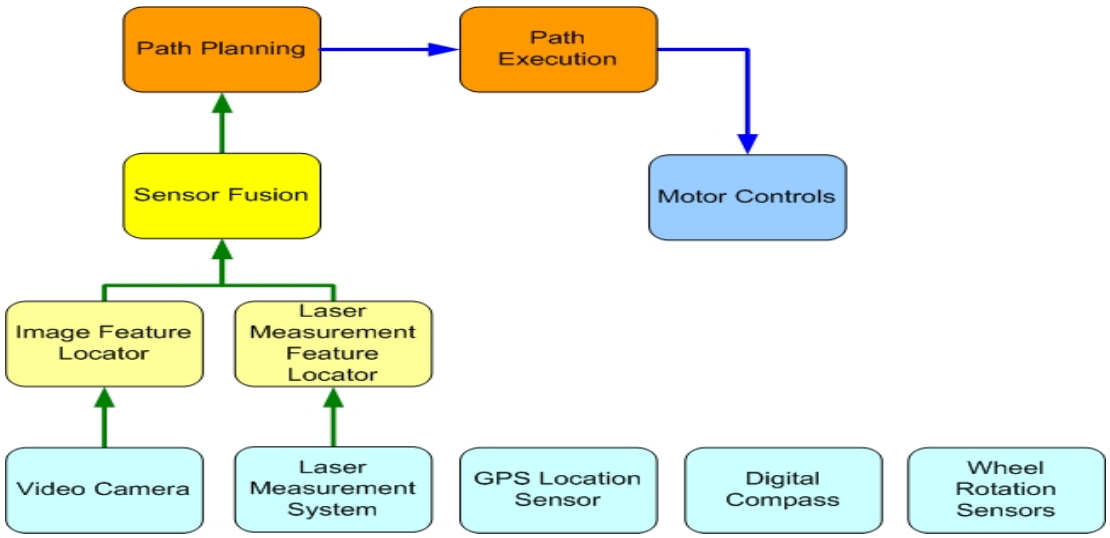


Figure 7. Autonomous Challenge High Level Design

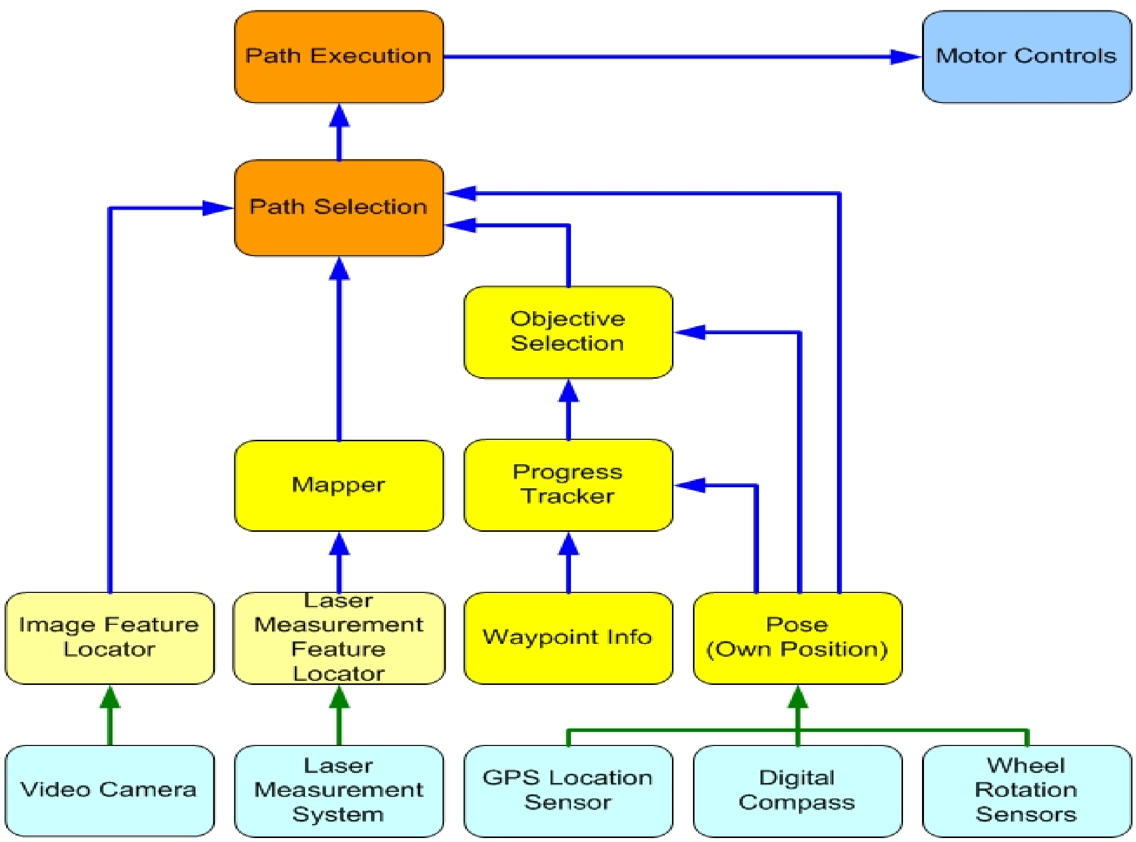


Figure 8. Navigation Challenge High Level Design

5.2 Sensor Fusion

Sensor Fusion is achieved by merging the data obtained from the Ladar with the image processing from the video camera. **Figure 9** below shows an example of obstacle detection with both the Ladar and the camera images overlaid.

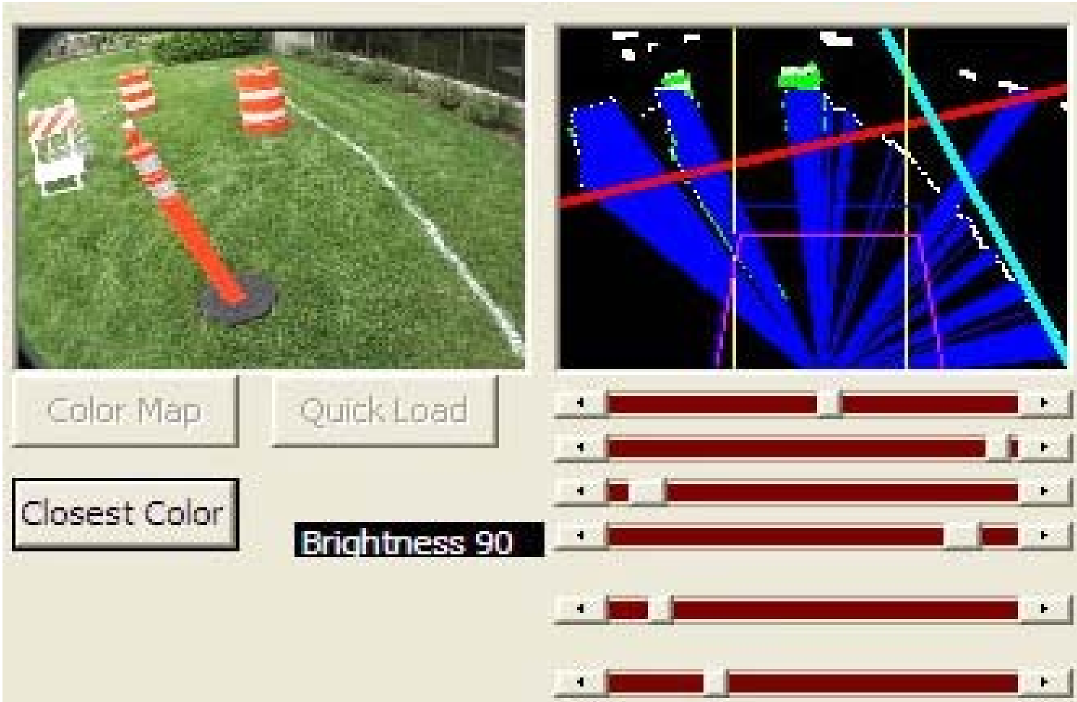


Figure 9. Sensor Fusion

5.3 Simulator Design

New to 2007, H_2Bot II control algorithms were developed with the aid of a high-fidelity autonomous vehicle simulation, configured to match the physical and response characteristics of H_2Bot II. While the simulator was developed with the 2007 IGVC competition in mind, its modular structure and consistent framework ensure that its usefulness will extend well beyond 2007. Written in Java™, it leverages the Java3D API for image generation and laser measurement simulations, shown in **Figure 10**.

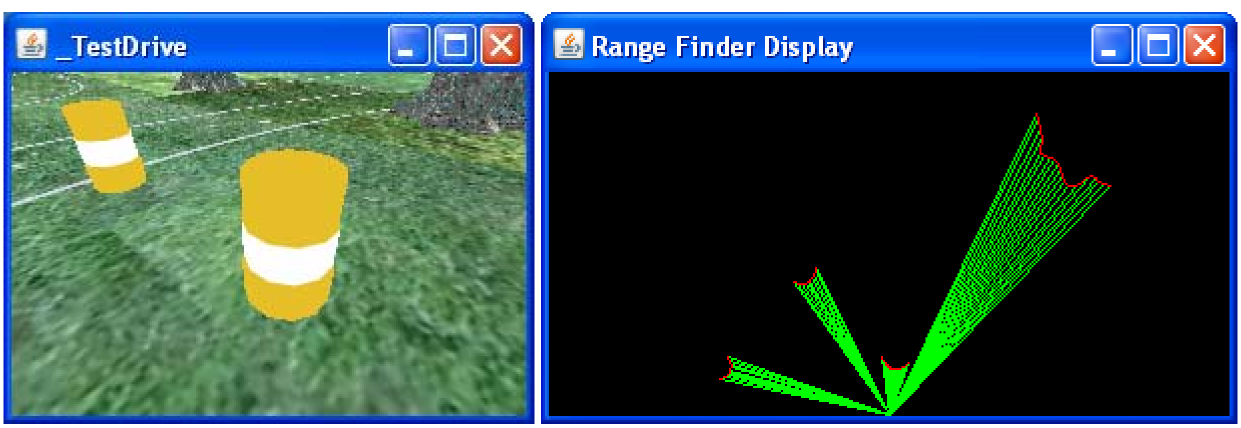


Figure 10. High Fidelity Simulation Using Java3D

The simulated H_2Bot II exports the same control and sensor interfaces as the physical vehicle, thereby avoiding the possibility of errors being introduced when control algorithms are migrated between models. The simulated environment uses standard 3D model files (wave-front) for representing the terrain and obstacles. The open-source “Blender” tool was used for constructing the necessary 3D models.

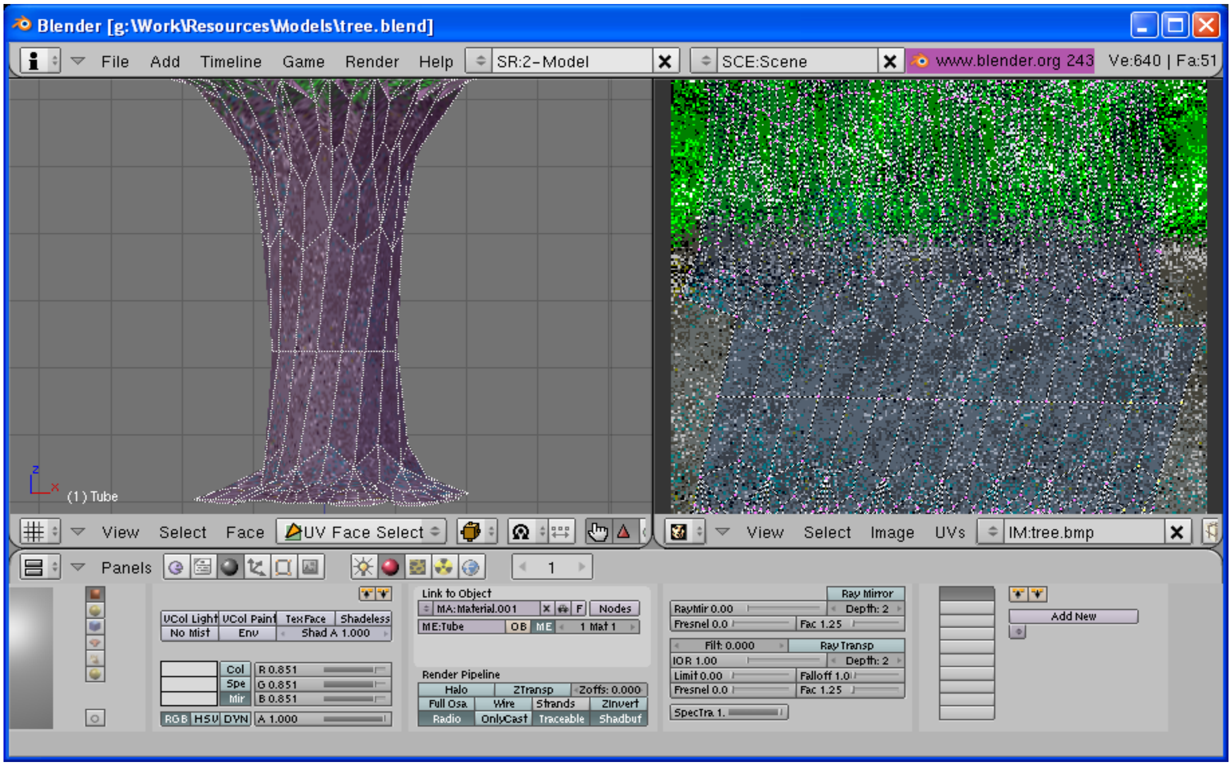


Figure 11. Modeling Obstacles Using Blender

5.4 Autonomous Challenge Details

5.4.1 Image Processing

H_2Bot II Image Processing Algorithm has been greatly improved from its 2006 predecessor. H_2Bot II has two main parts to its image processing algorithm, Image Filtering and Data Extraction. Currently it can filter an image with two different modes and with either RGB or HSB color models. H_2Bot II software platform can be configured to select one or both image filtering modes. The image filter mode will be selected based competition day test results.

5.4.2 Filter Mode – Closest Color

The first filter mode for H_2Bot II's image processing process is using "Closest Color". In this mode we are able to manually label foreground (white), background (black), and obstacle colors (orange) by taking samples from the actual color image. Once enough color samples are collected, we're able to process a color image and label all the colors of interest. This allows for the quick mapping and segregation of colors that is not easily done by applying a simple threshold algorithm. **Figure 12** demonstrates this algorithm in action.

5.4.3 Filter Mode – Threshold Color:

The second filter mode for H_2Bot II is a more traditional threshold algorithm. In this mode a threshold is applied followed by a contrast filter and quick noise sweeping filter to remove particles. Thresholds define the "minimum" levels for desired colors. **Figure 13** demonstrates this in action.

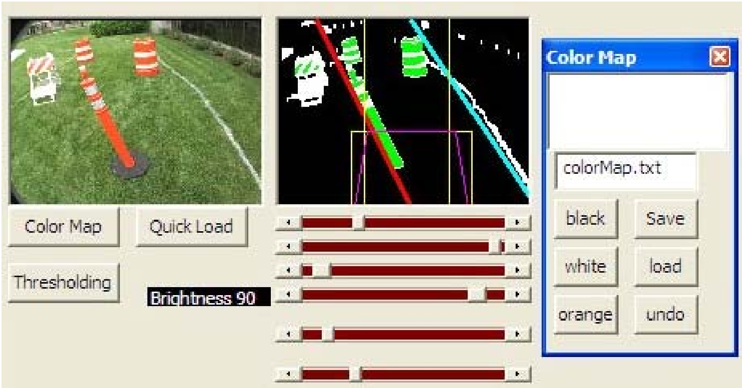


Figure 12. Nearest Neighbor Color

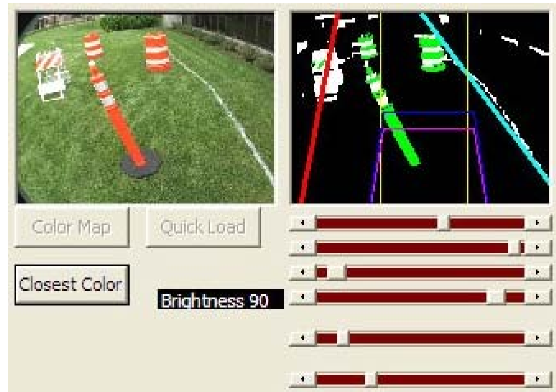


Figure 13. Threshold Color

5.4.4 Lane Following

Lane following is accomplished by first detecting where a possible line could be on the right and the left side of the input image. A bisecting line is then drawn between the two detected lines to give a possible heading.

Line/dash line detection is accomplished by taking all the white pixels, in the processed black and white image, and calculating the slope and intercept to each other white pixel. Using a large matrix of counters to track the length of all lines in the screen, the longest line is found and returned. This algorithm treats a dashed line as one solid line. If no line is present, the result will be empty. Large amount of noise can interfere with the line finding algorithm; this can be compensated by applying adjustable thresholds to color detection.

5.4.5 Obstacle Avoidance

Obstacles are detected using the Ladar, camera, or the combination of the two. The Ladar can detect obstacles within 8 meters and the camera obstacle detection distance varies with camera tilt angle. All obstacles including potholes are mapped to a 2D grid. The path planner finds the best route through the obstacles using a Numerical Potential field as described by Matt Greytak from MIT (http://web.mit.edu/mgreytak/OldFiles/Public/NPFwebsite/).

5.5 Navigation Challenge Details

The navigation challenge uses the GPS and laser range finder sensors in conjunction with dead-reckoning information available from the motor controller (derived from optical encoders on the motor shafts). Images from the camera are also used to locate the circles marked on the ground at the navigation challenge waypoints.

5.5.1 Path Planning

Known characteristics of the course are entered in the form of waypoints and barriers to the navigation challenge software. Synthetic waypoints are optionally added manually to aid in guiding the vehicle through known hazards. An initial desired path is provided that takes into account the initial direction of the vehicle. During the challenge, the set of unvisited waypoints is continually reevaluated (at low priority) to determine the shortest path required to visit the remaining waypoints (subject to known barriers) and return to the starting position. The resulting path is used to determine the next objective.

5.5.2 Path Execution

During the challenge, path execution is concerned with reaching the next objective, while also adding information regarding detected barriers to the set of known barriers. GPS and dead reckoning are used to determine when an objective is within range, at which time image processing is used to assist in guiding the vehicle to the center of the marked circle. The waypoint is removed from the set of unvisited waypoints, and the next objective is retrieved from path planning. This continues until all waypoints have been visited. The original starting position is then the objective. Again, image processing is used to assist in guiding the vehicle to the center of the starting box.

5.6 JAUS Challenge Details

For this year's competition, the JAUS handling code has been re-written by new team members. The new JAUS module handles both Level 1 and Level 2 JAUS challenges. **Figure 14** shows the high level design for the JAUS challenge architecture.

5.6.1 Process for Learning JAUS

To learn about JAUS the team employed resources available on-line and the JAUS reference architecture specification provided by Jauswg.org. The team was also provided the JAUS Compliance Tool Suite (JCTS) that allowed us to send JAUS messages and get an idea of how they were composed. The tool was essential in helping us test our JAUS module.

5.6.2 JAUS Integration into the Design

To integrate JAUS into the H_2Bot II a module was created that handles listening for JAUS messages send via the UDP protocol. The received messages are checked for validity and if the JAUS ID matches that of the vehicle the commands are parsed and passed back to the main H_2Bot II controller modules. Only the commands described in 2007 IGVC rules are supported at this time. To support Level 2 of the JAUS challenge, the JAUS module can also fetch information from the waypoint navigation module and send out a properly formed JAUS message containing the waypoint information. The design is modular and extendible allowing for ease of future additions.

5.6.3 Challenges Encountered

The main challenge has been in understanding how JAUS messages are composed and implementing code to parse out the needed information. The JCTS tool provided to us was helpful, but also incurred a learning curve to use properly. Integration with the rest of the H_2Bot II systems was also a non-trivial task.

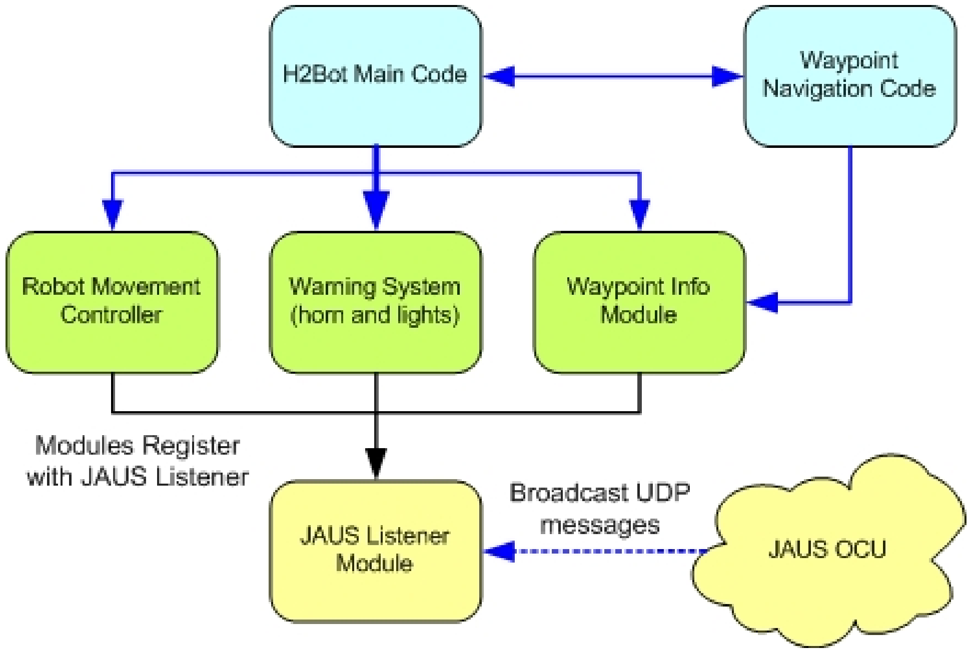


Figure 14. JAUS high level design

6. Performance Analysis and Estimate

6.1 Safety

H_2Bot II safety features include manual and remote E-Stop, failsafe brakes, and hydrogen leak detection shutdown.

6.2 Robot Performance

Vehicle performance was estimated based on the design parameters. These same attributes were measured after integration as shown in **Table 2**.

Attribute	Design Prediction
Maximum Speed	4.2 mph
Climbing Ability	2.5 inch curb
Nominal Power Consumption	500 watts
Battery Operating Time	4 hours
Hydrogen Operating Time	2 days
Brake holding capacity	>30% grade
Waypoint accuracy (with Omnistar)	.1 to .8 meters

Table 2. Performance Analysis

6.3 Reliability

The H_2Bot II utilizes dual modular power sources with increased capacity and features a thoroughly tested drive train design that performed well in the 2006 competition. The robot uses a rigid aluminum frame and extensive weather-proofing. Extensive testing has been done for the hardware and the software components. During the software overhaul, regression and system testing was done on each module to ensure proper functionality and integration with the hardware components.

6.4 Vehicle Cost Summary

Table 3 summarizes the total material cost for the H_2Bot II vehicle. Most components were reused from the 2006 H_2Bot vehicle.

Component	Total Cost	Team Cost
(1) MPC Laptop	$2,215	$0
(1) Sick LMS 291-S05 LMS	$7,000	$0
(1) Nexa 1.2 KW Fuel Cell	$7,000	$7,000
(1) NovaTel ProPak-LB DGPS & GPS-600-LB Antenna	$2,700	$0
(2) Brush DC planetary gear motors with e-brake	$2,000	$2,000
** (1) Panasonic PV-GS500 digital camcorder	$800	$800
(2) Hydrogen tanks + fuel	$900	$900
(1) PixelLink PL-A544 Video Converter box	$500	$0
Electrical Hardware	$500	$500
(1) PNI TCM2-20 digital compass/inclinometer	$400	$0
(1) Roboteq AX3500 Dual Channel Motor Controller	$395	$395
(2) Main Battery 12 Volt 40 Ah AGM	$150	$150
Chassis Materials	$300	$0
Miscellaneous Hardware (nuts, bolts, etc…)	$200	$200
(2) Hollow Shaft Optical Encoder Kit	$122	$122
(2) 14" Tire & Wheel Assembly	$58	$58
Total	$25,240	$12,125

** New for 2007

Table 3. Vehicle Cost Summary

7. Conclusion

H_2Bot II continues the LTU tradition of innovation and continuous improvement. Designing the vehicle from scratch by the incorporation of the pioneering fuel cell power source, modular electrical system, new drive system, and intelligent software, H_2Bot II will make an interesting and competitive entry.

8. References

[Ward 1994] Ward, A.C., Sobek, II, D.K., "Toyota and Set-Based Concurrent Engineering," 1994 ASME Design Theory and Methodology Conference Proceedings, Minneapolis, MN

[Ballard 2002] An Introduction to Fuel Cells and Related Technologies, Ballard Power Systems, Inc. 2002

[**http://www.jauswg.org**] The Joint Architecture for Unmanned Systems. “Reference Architecture Specification. Volume 2,” 2004.