2016 IGVC Design Report

Submitted: May 13, 2016

I certify that the design and engineering of the vehicle by the current student team has been significant and equivalent to what might be awarded credit in a senior design course.

Dr. Chris J.B. Macnab (Faculty Advisor)

Autonomous Robot Club Team

Member	Title	Email
Dr. Chris Macnab	Faculty Advisor	cmacnab@ucalgary.ca
Samuel Doctolero	Captain	sam.doctolero@gmail.com
Naveed Kawsar	Admin	naveedkawsar@outlook.com
Adam Medley	Admin	adam_p_medley@hotmail.com
Matt Angus	Software Lead	mattangus02@gmail.com
Collin Hill	Hardware Lead	cdhill@ucalgary.ca

Table of Contents

1. Introduction

TAURUS is a four-wheeled autonomous ground vehicle that utilizes a LIDAR, GPS, camera, and other sensors for information gathering and navigation. TAURUS has been designed and developed to navigate autonomously through an obstacle course while staying within a set path and seeking GPS waypoints in order to compete in the Intelligent Ground Vehicle Competition (IGVC). TAURUS has been designed and constructed by the Autonomous Robot Club members at the University of Calgary (UofC).

2. Design Process and Team Organization

In order to design and plan for the IGVC we first defined the requirements necessary for TAURUS to compete and succeed in the competition. Based on these specific requirements, several solutions were proposed and analyzed for any potential problems each solution may have. Requirements and proposed solutions considered everything from drive system configuration chassis design, sensors, and computation technologies. Each solution was then evaluated based on criteria that focused on cost, practicality, and ease of implementation. Design and construction then commenced in order to implement the solution that was selected. Each implementation was then tested in order to validate the success of the solution. Improvements and refinements to the solution and implementation were then made in order to address any new problems that unearthed during the development of each solution. This process was repeated several times until we were satisfied with the results. We estimate that approximately 500 man-hours have gone into the vehicle for the 2016 competition.

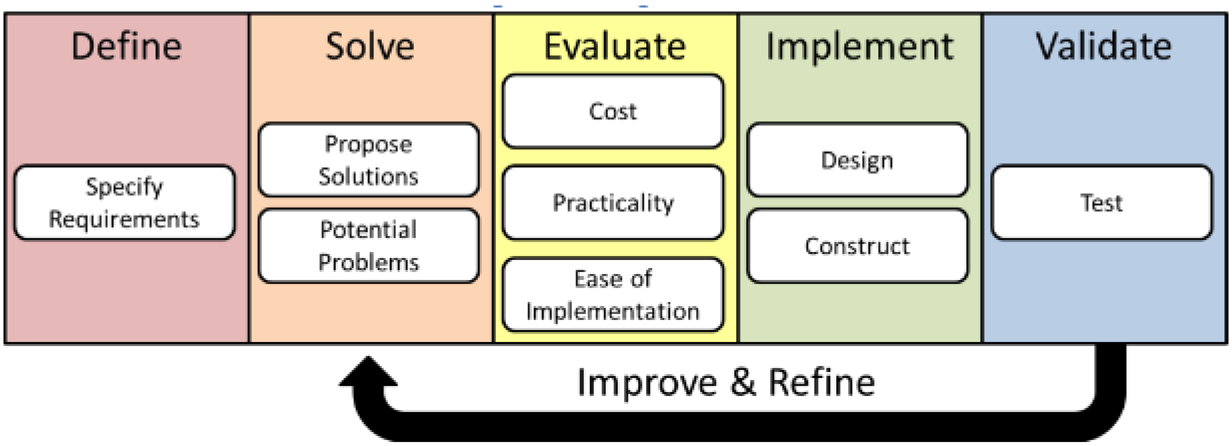


Figure 1: Design process

Team Organization

Team Member	Role	Academic Department	Class
Samuel Doctolero	**President**	**Engineering, Electrical**	**2016**
Naveed Kawsar	**Administrative Lead**	**Engineering, Electrical**	**2017**
Matt Angus	**Software Lead**	**Computer Science**	**2016**
Colin Hill	**Hardware Lead**	**Engineering, Mechanical**	**2017**

The design team working on TAURUS throughout this past year was comprised entirely of undergraduate students. We decided to take advantage of the fact that we currently have a rather small team working towards the 2016 IGVC by implementing a synchronous type of team organizational structure which normally requires a small team size. This type of team structure allows for a high level of interaction, offers the freedom to explore new ideas, and encourages active participation from every team member. In this type of structure, the manager acts more like a facilitator than an autocrat.

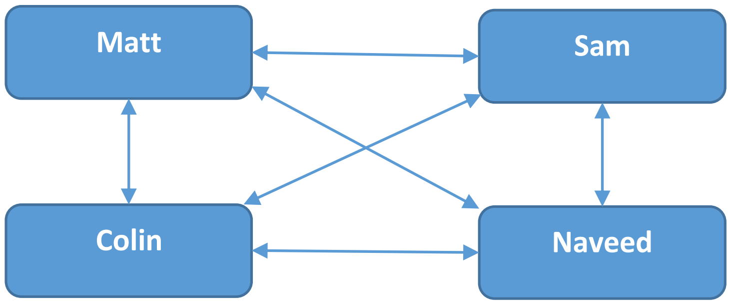


Figure 2: Synchronous Team Structure

Decision-Making

In accordance to our synchronous team structure, decision-making responsibilities are shared within the group and it is encouraged to be open to new ideas and explore them. However, when the best decision was not clear to our group, the president made the final decisions based on best judgement and practicality of the option with regards to the completion of TAURUS and for competing in the IGVC.

Software Development

Most of the software for TAURUS has been developed in MATLAB. Our team decided to code using MATLAB because of the ease of implementation and rapid prototyping ability. However, we understand that MATLAB is not the ideal language for robotics in the IGVC and we would like to translate all code from MATLAB to a more effective language in the future such as C++. We did, however, translate a portion of our camera algorithm (lane detection) to C++ due to OpenCV's superior speed and ease of use in terms of image processing.

As discussed in our design process earlier, once a software solution was selected it was then broken down into smaller components. These small components were clearly defined and they were specific such that integration of these components was as seamless as possible. By breaking down software solutions into smaller coding tasks we were able to delegate different pieces of code to different members of the team for more effective software development. Similarly, large sections of code and their requirements were clearly defined prior to actual coding to more effectively facilitate later integration.

3. Electronic Design

The electronic design consists of sensors, computers, actuators as well as two wireless controllers. The sensors are used to collect data regarding obstacle location, free space, lanes or paths, obstacle colours, vehicle location, wheel rotation, and vehicle orientation. Information is then sent to the computer for interpretation and it is then used in navigation algorithms. The “motor controller” controls the motors according to the navigation system’s decisions. A wireless gamepad is used for manual control of the vehicle through the laptop and a wireless controller/switch is used for activating the emergency stop, which is completely independent of any computational systems onboard. The emergency stop will still work despite any possible hang in the software, systems crash, or power failure.

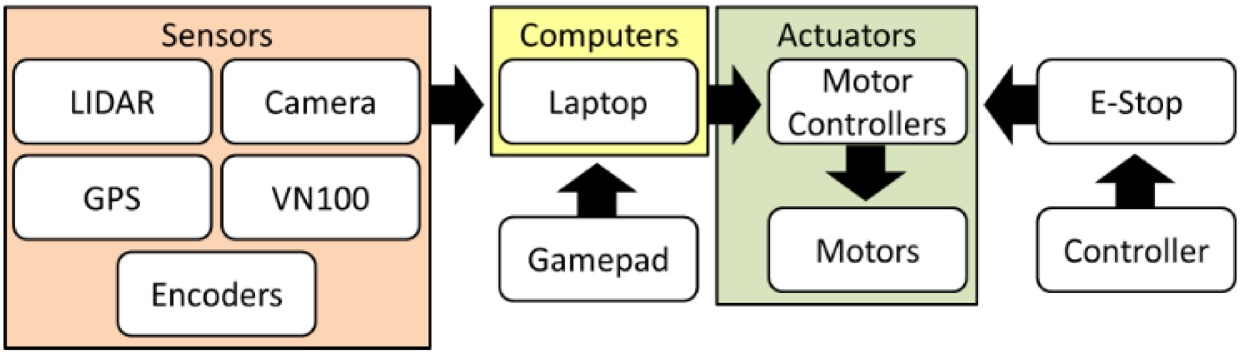


Figure 3: Electronic Design

Electronics

Sensors

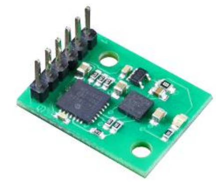
- **GPS** – A hemisphere GPS is used to provide location data for current spatial location. Data from the GPS is used in the navigation system and it is also used for path planning.

- **Camera** – A video camera/webcam is used to continuously stream a video of the surrounding information for data regarding lanes/paths as well as obstacle locations and color. This data is used in the navigation system for mapping and path planning.

- **LIDAR –** A Hokuyo laser scanner is used to provide information regarding location and distance of obstacles in the surrounding area relative to the vehicle. The measurements from this sensor are used for mapping and path planning.

- **Encoders –** Encoders are used with each motor in order to accurately provide data regarding the speed of each wheel. This information is necessary in order to limit the speed that the motors should run at.

- **Compass –** A tilt compensated compass module (CMPS11) is used to provide data regarding the pitch, yaw, and roll of the vehicle. This information is used to feed heading data to the navigation system.



Actuators

- **Motors –** Two NPC motors are used to propel the vehicle along its planned path. These motors receive commands directly from the motor controller.

- **Motor Controller –** A RoboteQ motor controller utilizes a built in PID control system to control the motors as well as provide data such as battery levels back to the computer. The motor controller is controlled directly by the computer on the vehicle.

Computer and Microcontrollers

- **Laptop –** An onboard laptop does all the computation required for the vehicle. This includes receiving and interpreting data from all sensors, planning a path using the navigation system, and sending commands to the motors through the motor controller in order to follow and track the planned path.

- **Arduino –** Three Arduino Uno microcontrollers are used to interface with smaller electronic items (CMPS11 and XBee Modules). The Arduinos act as the medium between the computer and the sensors or electronic modules.

Other

- **XBee Series 1 –** A communication module that can transmit and receive data to and from other neighboring XBee modules. Two XBee modules are used for the wireless emergency stop, one as the transmitter and the other as a receiver.

- **Gamepad –** A Logitech gamepad is used to provide user input to the navigational systems onboard the vehicle for manual user control.

- **Batteries –** Two 12 volt batteries provide power for the entire system.

- **Power Converters –** Several power converters change the voltage level of the input power to the required voltage levels (5 volt and 12 volt).

Electrical System

The previous version of TAURUS initially had two sources of power: 24 volt (two 12 volts in series) and 12 volt banks. The 24 volt supply's main purpose was to power the motors and the other was to supply power to the electronic systems. However, since the 24 volt supply can provide power for both the electronics and the motors we decided to remove the 12 volt battery banks and replace them with power converters. The converters act as a power level shifter and power isolators in order to provide power to all the electronics at required power levels and protect them from current or power surges.

It is also worth mentioning that the emergency stop is powered by itself and it is separate from the rest of the system. Also, the laptop runs on its own power since the run will only last approximately 15 minutes and therefore the laptop's own batteries will suffice.

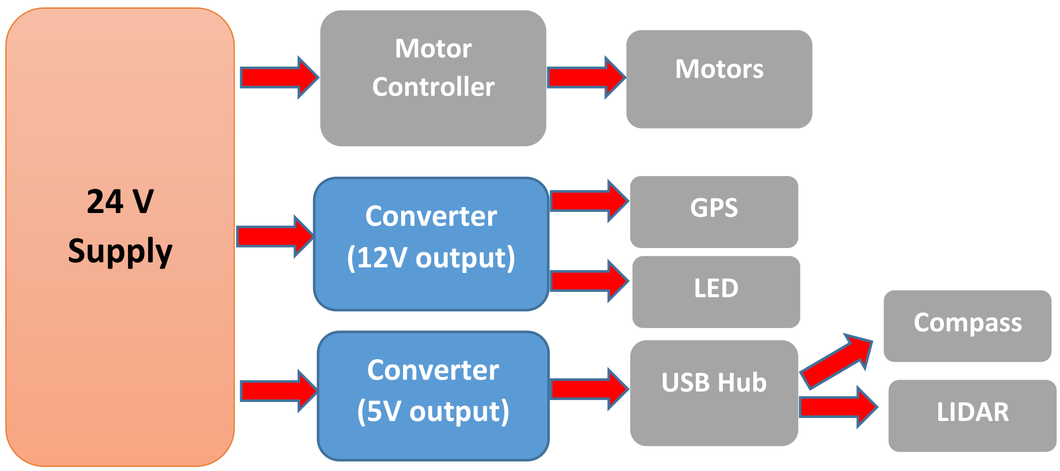


Figure 4: Electrical System Power

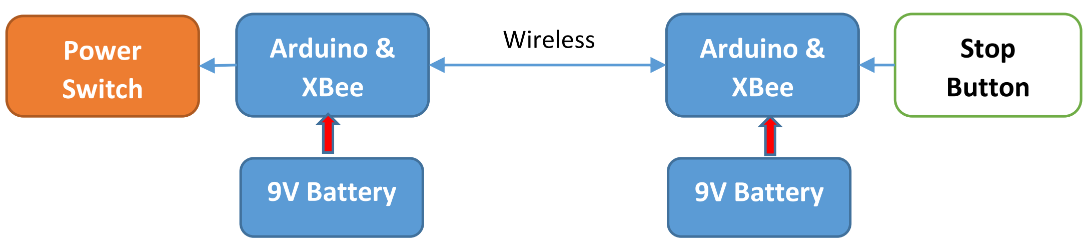


Figure 5: Emergency stop

4. Software Strategy

All of our code was designed and produced with future integration in mind such that the integration of software systems was as seamless as possible. While most of the code was written in MATLAB, pieces of the software were also written in C/C++. MATLAB allows MEX files to be created from C such that we are able to call subroutines written in C/C++ directly from MATLAB. Compatibility was taken into heavy consideration while procuring and designing hardware. Almost all hardware is able to communicate over USB (some components required RS-232 to USB converter) which simplified integration of hardware system.

Fuzzy Logic

At the heart of our software is a fuzzy logic system which decides which direction the vehicle should orient itself. All sensor data is massaged and manipulated before being passed to the fuzzy logic system. We have three functions that affect where TAURUS will navigate to: Near-Far, Target, and Free-Space. Each of these is based off of various combinations of bell and Gaussian curves. The Near-Far function determines if an object is near or far away from Taurus. The Target function determines where the next waypoint is with respect to TAURUS' heading. The Free-Space function determines where free space is with respect to TAURUS' heading. These functions are combined to calculate the rate at which Taurus needs to turn in order to avoid obstacles and also still make progress towards the next waypoint.

Camera Calibration

The camera takes pictures at an angle to the ground; therefore all images need to be transformed to represent accurate distances in laser (real world) space. We used a checkerboard to aid in determining this transformation.

Lane Detection

As mentioned before, our lane detection algorithm is written in C++ since we needed OpenCV's functions and excellent image processing speed. The camera takes an image of the ground at an angle which is then passed to the C++ function to find where the white lines are. Taurus uses a predetermined filter to filter out grass, in CIE L*a*b* colour space. The image is then converted to a black and white image where white pixels indicate where the white lines are and everything else is black. This image is then transformed using a projective transformation (from the calibration) in order to convert it from image space into laser space and have a bird's eye view of what is ahead of the vehicle.

Obstacle Detection

The obstacle detection system includes data input from the LIDAR which produces a 270∘ sweep of the local area to provide data on obstacle distance versus the angle relative to the vehicle's front-center. The obstacle detection system generates a set of distance and angle data which is then given to the navigation system.

Lane and Obstacle Avoidance

Both lane and obstacle detection systems produce distance and angle data sets. These two sets are then meshed together, they take the minimum distance from each angle, and it is then given to the navigation system for decision making. Edges of objects are extended to account for the width of TAURUS. This extension process will close holes in dashed lines to accommodate for errors in the data. The navigation system then finds an angle at which the distance is far and the space between the obstacles is large enough for the vehicle to fit through, this is where the fuzzy logic makes the decision.

Waypoint Navigation

The waypoint navigation system relies on data input regarding current location and the location of the next waypoint. This information comes from the GPS. The direction to the next waypoint is given a high weighting in the consideration of path planning and the vehicle will ultimately end up at the next waypoint. Once the current location and next waypoint are within a specified threshold, the next waypoint is located and the process is repeated.

Path Following

Our algorithm only incorporates local data and does not store any historical data due to the increase in complexity. As a result, we don't have a path following method but rather our fuzzy logic system will direct the vehicle towards the best possible available option.

5. Materials

Cost Estimate and Breakdown

Component	Quantity	Unit Cost	Cost
Hokuyo URG-04LX-UG01 Scanning Laser Range Finder	1	$1,334.47	$1,334.47
Razor Dune Buggy Frame & Shock Absorber	1	$493.49	$493.49
Hemisphere Crescent A100 Smart Antenna GPS	1	$1,680.00	$1,680.00
Wheels & Mounting Hardware	2	$70.35	$140.70
NPC T64 Motor & Gearbox	2	$345.09	$690.19
RoboteQ AX2850 Motorcontroller	1	$704.14	$704.14
Solarbotics GM2/3/8/9 Wheel Watcher Kit Encoder	2	$107.44	$214.88
HP Laptop	1	$503.99	$503.99
OCZ Vertex 2e 60GB 2.5" SATA II Solid State Drive	1	$136.49	$136.49
VectorNav VN-100 Attitude and Heading Reference System	1	$892.96	$892.96
Logitech Gamepad	1	$0.00	$0.00
Misc. Electronics, Materials, & Hardware	1	$1,050.39	$1,050.39
Camera	1	$0.00	$0.00
Futaba Radio Control System	1	$78.75	$78.75
Optima YellowTop D34/78 12V Motor Batteries	2	$272.99	$545.98
12V Accessory Batteries	3	$47.25	$141.75
Caster Wheels	2	$0.00	$0.00
		TOTAL COST	**$8,608.17**

Note: some of the items (i.e. CMPS11, Arduinos, and small electronic components) were provided by the department.

6. Safety, Reliability and Durability

Safety, Reliability, and Durability have been considered throughout the design and development of TAURUS.

Safety

- **Emergency Stop –** A wireless emergency stop is installed on TAURUS that is independent of any computation such that if the software were to hang, the wireless emergency stop would remain operational.
- **Kill Switch –** A large, visible, and accessible kill switch is located on the rear of the vehicle to allow for a quick shut-down of the power-train system to bring the vehicle to a complete stop.
- **Continuous Scanning –** While following the planned path, the LIDAR continuously scans the surrounding area for unexpected objects. If an unexpected object is identified, TAURUS will swerve away from the object.

Reliability

- **Isolated Power Systems –** The power system on the vehicle (electronics, motors, and e-stop) are electrically isolated from each other to improve reliability of the systems. This makes the emergency stop especially reliable as it is not affected by any other system error or failure.

Durability

- **Frame –** A frame was selected that would support the expected weight of the vehicle.
- **Wheels –** The wheels are not filled with air so they will not go flat or blow out. Also, they are rated for heavier and more torque-intensive applications, which is more than sufficient for this application.
- **Shell –** A shell provides a degree of water resistance to the vehicle such that it will be able to operate successfully through precipitation.

7. Expected Performance

- **Speed –** At the rated operation of the motors (230 rpm) and current tire diameter (10 inches), TAURUS' maximum speed is approximately 11 km/h (just under 7 mph).
- **Ramp Climbing Ability –** TAURUS is capable of easily climbing slopes of over 15 degrees.
- **Reaction Times –** The current reaction time of TAURUS is 200 milliseconds. However, safety laser scans are run continuously in order to prevent any possible collisions.
- **Battery Life –** The battery life of the robot lasts for well over 30 minutes of continuous use.
- **Obstacle Detection Distance –** The LIDAR is able to detect obstacles within a 3 meter radius with 95% accuracy.
- **Navigation Waypoint Arrival Accuracy –** The GPS provides an accuracy of 50 centimetres, 95% of the time.
- **Complex Obstacles**
  - **Switchbacks –** The vehicle will take whatever option is available to it even if there is no option that is in the desired heading of the vehicle.
  - **Center Islands –** The path finding algorithm for the vehicle tries to maintain an equal distance between any objects. Additionally, one of the masks increases the tendency of the vehicle to stick with one option. The vehicle will pick one side of the center island to go through and will stay roughly in the middle of that path.
  - **Dead Ends –** If there is no option to move forward even though the GPS heading is ahead, the vehicle will take whatever option is available to it which includes doing a 180 degree turn and heading backwards to attempt another option.

8. Conclusion

TAURUS is an autonomous intelligent ground vehicle capable of obstacle detection and avoidance as well as path following while seeking out GPS waypoints. Despite our in-depth design and planning process, we have faced a large number of unexpected challenges that were all highly varied. The experience in working towards the IGVC has been invaluable for all of the members of our team. The Autonomous Robot Club is very excited to have the opportunity to compete in this this year's Intelligent Ground Vehicle Competition.

Acknowledgements

The Autonomous Robot Club would like to take this opportunity to thank the people and organizations that have supported us throughout this experience. We would like to first thank Dr. Chris Macnab for his enthusiasm and support for our project as well as for being our faculty advisor. Thanks to the University of Calgary and the Schulich School of Engineering for their generous funding and work space provisions. Thanks to Chris Simon for his invaluable support and guidance. Finally, thanks to our past members for their hard work and contributions. This project would not have been possible without all of your support.