Old Dominion University

Batten College of Engineering & Technology

Self-Drive Car "Monarch I"

Date Submitted: May 15, 2021

Team Captain:

Alexander Fix(afix001@odu.edu)

Team Members:

Dana Pelland(dpell001@odu.edu)
Justin Heisterkamp(jeis005@odu.edu)
Christopher Vautier(cvaut002@odu.edu)
Warren Shields(wshie003@odu.edu)

Trevor Novisk(tnovi002@odu.edu)
Peter Espinoza(@odu.edu)
Chris Adolphi(@odu.edu)

We, the students of Old Dominion University, aspire to be honest and forthright in our academic endeavors. Therefore, we will practice honesty and integrity and be guided by the tenets of the Monarch Creed. We will meet the challenges to be beyond reproach in our actions and our worlds. We will conduct ourselves in a manner that commands the dignity and respect that we also give to others.

In addition, and as the faculty advisor, I hereby certify that the design and engineering of the vehicle (original or changes) by the current student team has been significant and equivalent to what might be awarded in a senior design course.

Faculty Adviser: Dr. Lee A. Belfore (lbefore@odu.edu)

1 Conduct of design process, team identification, and team organization

1.1 Introduction

This paper represents the conceptual design of our vehicle, ODU Monarch I, and its components. ODU Monarch I was built using the GLS&T and Army contract awarded to Old Dominion University (ODU). The objective of this contract was to implement the ARMY owned and developed Robotics Technology Kernel (RTK) 2019 software in a self -drive vehicle. ODU Monarch I is a Polaris GEM e2 electric car bought from AutonomouStuff was originally outfitted with a drive-by-wire system, a Velodyne VLP-16 LiDAR, a Lucid Triton camera system , and the AutonomouStuff Spectra computing platform. Added to what was already installed was the ZED 2 3D stereo camera, the ZED-9FP High U-blox high precision GNSS module and the XSENS MTi 670 IMU. Ubuntu 16.04 and Robotic Operating System (ROS) Kinetic was installed on the computing platform as the primary operating system to which RTK was added.

The main objective of the Monarch I is to properly and safely navigate the Intelligent Ground Vehicle Competition (IGVC) self-drive challenge course simulating traffic situations seen by everyday drivers. To do this, Monarch I will be configured to navigate within the road lanes using lane detection software, recognize and obey road signs using a neural network, and negotiate obstacles in the vehicle’s path by using object detection and avoidance software that utilizes the onboard camera systems and the Velodyne VLP-16 LiDAR.

1.2 Organization

This project was an undertaking by an interdisciplinary team consisting of undergraduate students completing their Electrical and Computer Engineering Senior Design I and II as well as graduate students working on their Master’s thesis or Master’s projects. These team members specialize in the several areas of Electrical and Computer Engineering. The diversity of the knowledge of the team allowed for both software and hardware development in several areas, that when fully integrated will produce a working self-drive vehicle.

The competition team was divided into different areas to develop the different components and then came together to integrate the components into the GEM e2 electric car.

Name	**Major**	**Primary**	**Secondary**
Alexander Fix	M of ECE[1]	Team Captain	Imaging Software Support

[1] Master’s of Electrical and Computer Engineering

Dana Pelland	M of ECE	Mapping Software Support	Path Planning Software Support
Justin Heisterkamp	ECE[2]	System Sensor Hardware/Software Support	Power Distribution and Computer Hardware Integration
Christopher Vautier	EE[3]	Safety Lead	Power System &Hardware Configuration
Warren Shields	CpE[4]	Software Support	Data Analysis
Trevor Novisk	EE	Hardware Mechanical Systems	Power Systems
Peter Espinoza	M of ECE	Lane Detection	Software Support
Chris Adolphi	CpE	Neural Network	Software Support

Table 1: Tasks and Responsibilities

1.3 Design assumptions and design process

Our assumption in the design is to have an autonomous vehicle that will qualify and compete in the Self-Drive Challenge. By utilizing the Rational Unified Process we were able to integrate hardware in software to perform lane detection, object detection, and image processing. Safety lights and emergency stops will be affixed on the vehicle to satisfy safety requirements.

The design expectations and processes are in accordance with the 2021 IGVC Rules[5]. Since this will be the first attempt at qualifying and competing in the Self-Drive Challenge, we are focused on meeting all the requirements to qualify and compete.

1.3.1 Safety Lights

The ODU Monarch I is equipped with red brake lights, yellow reverse lights and a LED safety light on the roof of the vehicle. The safety light will come on when the vehicle is in autonomous.

1.3.2 Lane Detection and Following

Software for the vehicle to determine the traffic lane in front of it. As the vehicle travels along the software determines where it is in the traffic lane and provides inputs to vehicle to make adjustment to maintain its position within the lane. If the vehicle crosses beyond the boundary of the lane the software provides inputs to correct the vehicle's path and bring it back to within the boundary.

---

[2] Electrical Engineering and Computer Engineering (Dual Major)
[3] Electrical Engineering
[4] Computer Engineering
[5] https://www.igvc.org/2021rules.pdf

1.3.3 Obstacle Detection and Avoidance

The ODU Monarch I utilizes the Velodyne VLP-16 LiDAR and the onboard cameras to detects obstacles in the traffic lane. The data from these systems into the costmap and navigation stack to maneuver the vehicle to avoid the obstacle detected.

1.3.4 Road Sign Detection

The ODU Monarch I utilizes a combination of the Velodyne VLP-16 LiDAR and the onboard cameras to detect the road sign. The information is then fed into a Faster R-Convolution Neural Network (CNN) to determine what type of road sign it is. The information from the sign determinate will be integrated with the cost map to determine how the vehicle responds.

1.3.5 Waypoint Navigation

The vehicle must be able to detect GPS waypoints. Once the waypoint is processed and provided to the costmap, it is expected that the costmap will create a planned path to navigate the vehicle around the obstacle to the waypoint.

2 Effective innovations in vehicle design

2.1 Innovative concepts from other vehicles designed into our vehicle

2.2 Innovative technology applied to our vehicle

2.2.1 Expand USB capability

The Spectra computing platform currently has four USB ports. Due sensors requiring USB connections there exists a need to expand the number of USB ports on the computing platform. Furthermore, some peripherals, such as the ZED 2 requires an independent high-speed channel. To expand the USB capability a FebSmart PCIe 3.0 X4 USB 3.1 Gen 2 expansion card will be installed in the computing platform. This new card will expand the number of USB ports by four.

2.2.2 Increasing computing data processing capability

The hard drive currently in the computing platform is 500 gigabytes. To increase the current computing capability a second solid state device will be installed in the computer. A second 500 gigabyte hard drive (Samsung 870 EVO) will be installed to support a dual boot system that can optionally be configured as an encrypted drive. An additional one terabyte SSD (Crucial BX500) will be installed to gather run time telemetry.

2.2.3 Distributed computing to improve data processing performance

In order to facilitate the data processing required to operate the Monarch I, using an NVIDIA Jetson TX2 to offload the data processing for a StereoLabs ZED 2 stereo camera. The StereoLabs provided drivers and ROS nodes will run on the Jetson.

2.2.4 Increasing power capability

Due to some of the sensors requiring clean DC power there is a need for AC power. To achieve this a DC to AC inverter is being installed. The sensor's power adapter is plugged int the inverter to provide the clean DC power needed.

3 Description of mechanical design

3.1 Overview

ODU Monarch I is a Polarias GEM e2 electric car. The car was outfitted with several sensors that will be used for functions such as obstacle detection and avoidance, lane following, and waypoint navigation. These sensors include a Velodyne VLP-16 LiDAR, a Lucid Triton Model 2.3 MP camera, a ZED 2 3D stereo camera, a ZED-F9P high u-blox high precision GNSS Module, and a XSENS MTi 670 IMU. Detection algorithms were developed to perform lane following, obstacle detection and avoidance, and road sign recognition.

3.2 Sensors and Sensor Placement

The Velodyne VLP-16 LiDAR [1] is the primary sensor for object detection. The Velodyne VLP-16 LiDAR is mounted on the rack located on the roof of ODU Monarch I. This placement provides the full 30 degrees of vertical field of view in front of the vehicle but only the positive 15 degrees to the rear. The curve of the roof blocks the negative 15 degrees.

The Lucid Triton camera [2] is a color camera with 2.3-megapixel resolution. The camera is mounted on the ODU Monarch I's roof rack approximately 30cm to the left of the Velodyne VLP-16 LiDAR.

The ZED 2 3D stereo camera [3] is manufactured from StereoLabs. The camera is mounted on the dashboard of the ODU Monarch I. This placement provides a wide field of view directly in front of the vehicle.

The u-blox ZED-F9P is a high precision GNSS module [4] that provides positioning support. The antenna for this is mounted to the rack located on the roof of the vehicle.

The XSENS MTi 670 IMU [5] contains inertial sensors. The unit is mounted at a central point inside the vehicle.

3.3 Description of drive-by-wire kit

The drive-by-wire capability for the ODU Monarch I was installed by AutnomouStuff. The installed module supports steering, throttle and braking controlled by a Logitech game controller. The drive-by-wire functionality is implemented into the vehicle utilizing the ROS nodes.

3.4 Suspension

The suspension for the vehicle is the stock suspension system installed on the vehicle when built at the factory. It uses an automotive style suspension [6] consisting of MacPherson struts with 5.6 inches travel in the front and independent trailing arms with 6 inches of travel in the rear.

3.5 Weather proofing

To weatherproof the vehicle the doors have been purchased to installed on the sides. The doors will close the open portion of the cab making it weatherproof. With exception of the sensors mounted on the roof. The remaining sensors, displays and computing platform are internal to the cab.

4 Description of electronic and power design

4.1 Overview

The Polaris GEM e2 electric car utilizes an ACM battery to run the vehicle. To distribute the power from the battery the vehicle uses power distribution system managed by a multiplexed vehicle electrical center. The onboard computing platform is the AutonomouStuff Spectra installed by AutonomouStuff. The operating system installed on the Spectra is Ubuntu 16.04 with ROS Kinetic. For the GLS&T and Army contract, RTK 2019 software is being used.

Power distribution system (capacity, max, run time, recharge rate, additional innovative concepts)

The ODU Monarch I uses a distance AGM battery to power the vehicle. The range on a full charge is approximately 125 miles and requires approximately 10 hours to recharge when the battery is fully depleted.

Monarch I’s Power Distribution System is managed by a multiplexed vehicle electrical center (mVEC) unit that distributes 12V, 150A fuse protected, into eight 12V, 20A fuse protected, outputs and four 12V unprotected lines. A 12V to 24V Converter provides the power for the onboard Spectra computer. Some of the smaller equipment components can be powered from USB connections on the Spectra computer. However, some components may require utilizing an open 12V slot on the mVEC PDS. Currently, the onboard display, the Velodyne VLP-16 Lidar, and the Triton Camera use a slot and there are nine open 12V sources.

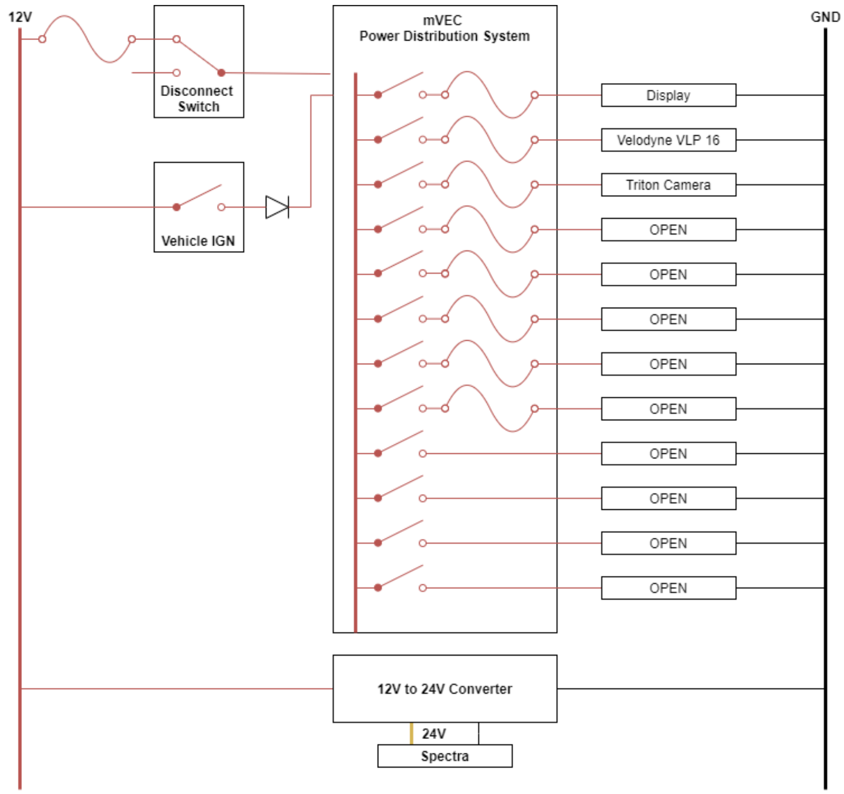

Figure 1: Monarch I Power Distribution

4.2 Electronics suite description including CPU and sensors system integration/feedback concepts

The CPU for the vehicle is the AutonomouStuff Spectra. The Spectra contains 32GB of RAM and includes two gigabit Ethernet ports, four USB 3.0 ports, two COM ports, two DVI ports and three PXIe slots. Also included is a RTX-2080 GPU and a four port gigabit server card. To assist with the ability to operate the vehicle outdoors both computing platform and the GPU are equipped with thermal designs to keep them cool.

The Velodyne VLP-16 LiDAR [1] can sense object up to a distance of 100 meters in any direction around the sensor and is precise to within 3 cm. The sensor can also detect vertically at ± 15 degrees horizontally from the sensor. It has a rotational scan rate between 5 and 20 Hz. Its low operating power requirements of only 9-18 Volts makes it an ideal sensor for the Polaris GEM e2 electric car.

The Lucid Triton camera [2] is a color camera with 2.3-megapixel resolution. The camera resolution is 1920x1200 pixels that is processed at frame rate of 52 frames per second. The camera utilizes a f/1.8 to f/11 lens with a 33-300 focal length.

The ZED 2 3D stereo camera [3] is manufactured from StereoLabs. It has a 110 degree horizontal field of view and a 70 degree vertical field of view. These fields of view extend out in front of the camera to a range of 20 meters.

The u-blox ZED-F9P is a high precision GNSS module [4] provides positioning support. It utilizes real time kinematics and multiband GNSS. It supports multiple types of global positioning including GPS, Galileo and BeiDou constellations.

The XSENS MTi 670 IMU [5] contains inertial sensors that include an accelerometer, a magnetometer, & barometer and is capable of processing measurements at a rate of 400Hz. The information processed is integrated with the GNSS to provide positioning.

4.3 Safety devices and their integration into your system

The safety devices used on this vehicle include several mechanical emergency stops that when pushed will immediately stop the vehicle. These emergency stops are located in several locations in and around the vehicle as specified in the IGVC 2021 rules. There will be one button in the cab of the vehicle, one on the rear of the vehicle and one on either side of the vehicle. These buttons are in series with the emergency stop system installed by AutonomouStuff, so that activation of any them will stop the vehicle. Additionally, there is a wireless emergency stop, Kar-Tech model 2A466 emergency stop installed that is functionally up to 100 feet in distance from the remote transmitter.

The other safety devices used on the vehicle include the use of lights. There are red brake LED lights and yellow LED lights for when the vehicle is in reverse installed on the rear of the vehicle. There is a LED Light for when the vehicle is in autonomous mode mounted to the rack located on the top of the vehicle.

5 Description of software strategy and mapping techniques

5.1 Overview

The software strategy for the ODU Monarch I was to use ROS as the main robotic operating system. The ROS software is designed to process various inputs as nodes that subscribe and publish the necessary messages to safely maneuver the vehicle. Python OpenCV was used to process video images needed for lane following, object detection, and road sign recognition. MatLab was used to build and run the Faster-R-CNN used for road sign recognition. RVIZ was used to visualize the mapping created from the different sensors and the proposed path based on the occupancy maps generated. The combination of the different programs allows the vehicle to operate and maneuver.

5.2 Obstacle detection and avoidance

The vehicle is expected to maneuver around the self-drive course, that is designed to simulate real world roads. In order to do this the vehicle is required to be able to detect the lane of travel in which the vehicle must stay. As it travels in this lane, it is required to stay within the lane and when an obstacle is detected within that lane and determine how to maneuver around that obstacle.

The lane following software was designed to use image data provided by the ZED 2 camera and provided to a lane following algorithm generated with OpenCV. Because the vehicle would need to navigate curved paths as well straight paths the warped perspective transformation along with a window search algorithm was used. This process takes an image from the ZED 2 camera processes it with OpenCV and then using the two functions getPerspectiveTransform and warpPerspective determines the points to identify the edges of the lane and applies windows to them with the window search algorithm. The data from the algorithm is then unwarped using the same OpenCV functions used to warp the image. The final process is to overlay the data over the input image to create the final output image defining the lane. As the vehicle travels this lane, the lane will indicate green as long as it stays within it. If it begins to travel outside or the lane it will turn a red indication and based on the correction data sent to the vehicle's navigation stack the vehicle will be maneuver back into the lane.

Additionally, it is required to recognize traffic signs and based on that sign determine how the vehicle must react. To do this, the vehicle will utilize image data from the ZED 2 3D camera and the Lucid Triton color camera systems as well as point cloud data from the Velodyne VLP-16 LiDAR to detect any obstacles within the established lane as well as any road signs. When a road sign is detected the information is provided to the Faster-R-CNN where a decision is made to direct the vehicle to move based on the sign. In the case of the stop sign, vehicle will stop.

5.3 Software strategy and path planning

The software strategy used was to integrate the data from the sensors in the form of point clouds and image processing and supply them as inputs to the costmap software. The costmap generates maps based on the obstacles detected by the sensors. These inputs are displayed on the costmaps based on the configuration settings within the yaml files. Based on these configurations occupancy maps are generated. When in autonomous mode, a destination is chosen based on a provided waypoint. Using the waypoint information in conjunction with the occupancy map information, the path is planned.

5.4 Map generation

Sensor data is provided to both the local and global costmaps. Using the data from the various sensors an occupancy map is generated. The occupancy map consists of various colors based on parameters in the yaml file. Those parameters determine how far out around each obstacle to set a perimeter. As the vehicle moves about the course more data is collected and provided to the costmaps to create a more complete picture of the course. The maps can be saved to the map server for later usage.

5.5 Goal selection and path generation

When a waypoint provided to the vehicle as a destination, the data from the costmaps is provided to the path planner. Based on the information from the occupancy maps a path is planned to move the vehicle from its current location to its desired destination. The occupancy map determines how close the vehicle can get to the obstacle before it must be maneuvered to avoid the obstacle. This information is fed to the move base which tells the vehicle how to move.

5.6 Additional creative concepts

Additional creative concepts that are a desire to have implemented into the ODU Monarch I is the use of ZED 2 camera system to detect pedestrians in the vehicle’s path. Once detected the information will be provided to the Faster-R-CNN. The CNN would then provide an input to the navigation stack to tell the vehicle to stop until the pedestrian has moved out of the vehicle’s path.

5.7 Vehicle safety design concepts

The emergency stop internal to the cab of the vehicle is installed on the dash and when pushed will immediately stop the vehicle. To meet the requirements of the competition additional emergency stops are being installed in series with the installed emergency stop. This will allow the vehicle to stop immediately when any of the emergency stops are pushed. In addition to the manual emergency stops, a wireless emergency stop is installed that is activated with a remote transmitter.

Safety light have been installed on the vehicle in accordance with the competition rules. The installed lights include red LED lights for when the vehicle is braking, yellow LED lights for when the vehicle is operating in reverse, and a LED light on the top of vehicle to indicate the mode the vehicle is operating in.

6 Simulations employed

During the development of the software several simulations were used to develop and test software prior to implantation on the vehicle. Some of those simulations include the use of Gazebo, RVIZ, and road test videos provided by Udacity's Self-Driving Car Engineer Nanodegree – Advance Lane Finding project.

6.1 Simulations in virtual environment

Several types of simulations were used to test software. At the beginning of the development for costmaps a simple simulated vehicle with a differential drive was created in gazebo and its sensors displayed in RVIZ. These simulations allowed for initial costmap generation and yaml file configuration. This same simulated vehicle was later used to develop the CNN for detection of stop signs. They utilize the point cloud data provided by the Velodyne VLP-16 LiDAR to detect objects.

Later simulations involved the use of simulated Polaris GEM e2 vehicle. This vehicle has Ackerman steering implemented instead of the differential drive the previous vehicle had. This new vehicle was moved using a differential drive based teleop program but the turning did not work properly. Several Ackermann steering controllers were tried but none were able maneuver the vehicle.

The lane detection software was tested using the road test videos provided by Udacity's Self-Driving Car Engineer Nanodegree – Advance Lane Finding project. This is because the software has not been implanted into the vehicle yet so the software had to be tested on the simulation instead. Results of this simulation testing show promising results for the software when it is fully implemented into the vehicle.

7 Performance Testing to Date

7.1 Component testing, system and subsystem testing, etc

As of this date the installed sensors have been individually tested and tested as a whole by manually driving the vehicle and driving the vehicle using the drive-by-wire functionality. The data from these sensors have been captured and stored in rosbags.

8 Initial Performance Assessments

8.1 How our vehicle is performing to date

Our vehicle at this time, the vehicle has been manual driven and driven using the drive-by-wire. During these driving events data has been captured using the installed GPS, IMU, and Velodyne VLP-16 LiDAR. This data has been saved to rosbags for later use. The data in the rosbags have been used to begin creating costmaps. At the time of the writing of this report costmap generation is approximately 85 percent complete.

9 Mandatory Unit Tests up to date

Due to continued implementation of software and hardware, unit tests have not been conducted at the time of writing this report. Status of these tests will be provided at the presentation of the report.

References

1. Velodyne Lidar, *Velodyne Lidar Puck*, data sheet, September 2019.

2. Lucid Vision Labs, *Triton 2.3 MP Model, specifications*, thinklucid.com/product/triton-23-mpimx392,accessed 03/02/2020.

3. StereoLabs, *ZED technical documentation*, www.stereolabs.com/docs, accessed 03/02/2020.

4. XSENS, *MTi 600-series Data Sheet*, September 2019.

5. u-blox, *ZED-F9P High Precision GNSS Module*, data sheet, July 2019.

6. GEM e2, *GEM e2*, data sheet, May 2021.