Titan Express

University of Detroit Mercy IGVC Self-Drive 2019 Design Report

Submitted May 15, 2019

Captain: Yuyi Li liyu5@udmercy.edu
Team Members: Viken Yeranosian yeranovr@udmercy.edu, John Belanger belangjd1@udmercy.edu, Ratheesh Ravindran ravindra@udmercy.edu, Samar Bayan bayansa@udmercy.edu, Mohamad Ali Mokhadder mokhadmt@udmercy.edu, and Melvin P Manuel manuelmp@udmercy.edu

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 credit in a senior design course.

Michael Santora ______________ 5/15/2019

Utayba Mohammad ______________ 5/15/2019

Introduction	**3**
Team organization	**3**
Design Process	**3**
Innovations	**3**
Mechanical Design	**4**
Electronic and Power Design	**4**
Software Strategy and Mapping Techniques	**8**
Description of failure modes, failure points and resolutions	**10**
Simulations employed	**10**
Performance Testing to Date	**11**
Safety Systems	**12**
Initial Performance Assessments	**13**

Introduction

In this document is the development of a Polaris Gem e2 vehicle called Titan Express, shown in Figure 1, from a human operable vehicle to a self-drive vehicle for the Intelligent ground vehicle. The Titan Express is a street legal 2 passenger Polaris Gem e2 is a 48V electric vehicle with a 6.7 HP motor capable of top speed of 25 mph[1]. In the following sections the team and organization, the mechanical design, the electrical and power design, the software, failure modes, simulation and performance are discussed.

Figure *1*: Polaris Gem e2 (front-side view)

Team organization

Advisors: Utayba Mohammad and Michael Santora
Leader(s): Yuyi Li (Lane Detection) and John Belanger (Navigation/Hardware)
Team Members: Viken Yeranosian (Localization/Hardware), Ratheesh Ravindran (Sign Detection), Melvin P Manual (radar), Samar Bayan (Goal Creation)

Design Process

1. Make initial plan of tasks to reach project goals
2. Research gaps in knowledge
3. Brainstorm possible approaches
4. Develop initial design
5. Analyze results of initial design
6. Reiterate designs and testing
7. Quantify final design.

Innovations

The innovations on Titan Express are the combination of a high capacity computing system of a computer and Nvidia PX2, the implementation of a fisheye camera for use with a neural network to perform lane detection, produce a 3D point cloud from the lane detection for use in the cost map, the use of neural networks to detect signs and pedestrians, construction and implementation of localization, and the conversion of the Polaris Gem e2 to a drive-by-wire autonomous vehicle.

Mechanical Design

The Titan Express is a converted Polaris Gem e2 FMVSS-500. The vehicle is shown in Figure 1. To mount the needed sensor hardware an extruded aluminum frame is placed on top of the vehicle. Attached to the frame is the GPS antennas, LiDAR, and cameras. The extrusion profile allows for easy modification.

Figure 2: Polaris Gem e2 with frame and sensors

Electronic and Power Design

Power: The Polaris is a commercially available electric vehicle. The Polaris' electric motors are directly controlled with a 6.5 HP Sevcon 450A controller, which is powered by 48 volts produced by a series of 6V batteries [1]. The 48 V battery packs are 6kW and capable of giving around a 30-mile range. The charger is capable of 1kW. The autonomous hardware is powered from 48V to 12V DC-DC converters. All of the 12 V power is fused and contained in the cabin of the vehicle. There is also a Pure Sine Wave 2500W 120VAC converter to power the laptop running the Robot Operating System (ROS).

Vehicle: The Polaris is converted to drive-by-wire for throttle, braking, and steering. There are three microcontrollers that interface into ROS for the control of the vehicle. The first is the vehicle control unit (VCU), the second is for encoder capture, and the third is for communication with the remote control. The first microcontroller is an Arduino Mega2560, which is the vehicle control unit (VCU). The VCU controls the throttle, steering and brake of the vehicle. Additionally, the VCU monitors the emergency stops on the vehicle to communicate to ROS when those have been pressed. For safety, the VCU only manipulates control signals when the vehicle is put into computer control. If the vehicle is not in computer control the original factory connections are made. The control for the throttle is shown in Figure 3. The analog control from the Arduino is done using a 10-bit digital-to-analog converter, LTC2465. The autonomous steering for the Polaris is done by manipulating the torque sensors in the Electronic Power Assisted Steering EPAS) System, as shown in Figure 4. This is done in the first Arduino. To allow for the computer control to change to steering effort a dual channel absolute encoder is attached to the steering column of the vehicle. The dual channel absolute encoder is two potentiometers that are opposite each other and give inverted readings of each other. The dual encoders are implemented to add a safety check that steering position is being read correctly and is a safety feature. The position of the steering column is controlled by a proportional controller programmed into the Arduino.

The braking system is also controlled with the first Arduino. The braking system is split between the front and rear brakes for user control or computer control, as seen in Figure 5. The brakes are split to allow the user to have the most control, safety, over the braking system. The front brakes are left for the user control. The rear brakes are either actuated by the emergency brake or a Hydrastar System. The Hydrastar system (Figure 6) is connected to the VCU with a H-Bridge module and is what controls the autonomous braking. When an emergency stop is activated the VCU incrementally increase over 0.5 seconds the Hydrastar brake until 100% brake signal is achieved.

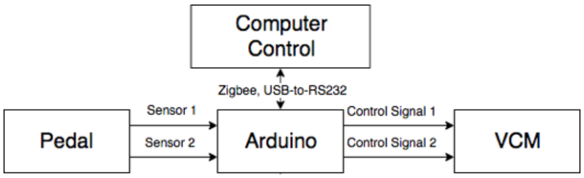

Figure 3: Original wiring diagram of the throttle module [2]

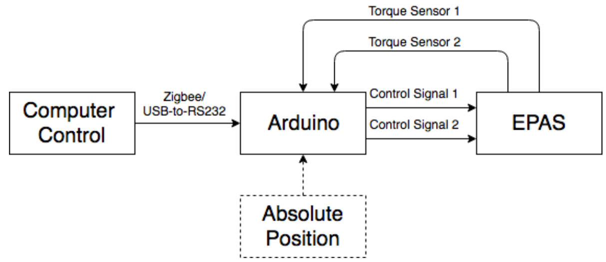

Figure *4*: Wiring diagram of the steer-by-wire system [2]

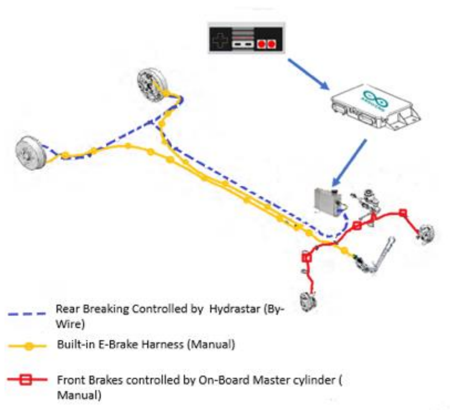

Figure 5: Wiring and hydraulic line diagram for the brake-by-wire system [2]

Figure 6: Hydrastar module [2]

The second microcontroller is programmed to acquire two Encoder Products company (776-B-S-1024-Q-OC-C-P/6-A-N-N) that are installed on the 2 drive shafts of the Titan-Express. The two incremental encoders are being captured by the third ATMega2560 and sent serially to ROS to be published as a topic. To supplement the SPAN solution from the Novatel the sensors can also be utilized to generate odometry data.

The remote-control system is created with two Arduino Uno that have Zigbee wireless communication. There is a user node that transmits the desired mode of the vehicle and vehicle node that receives commands from the user node, communicates it to ROS as a published node. The remote control is shown in Figure 7 and the vehicle remote node and VCU are shown in Figure 8.

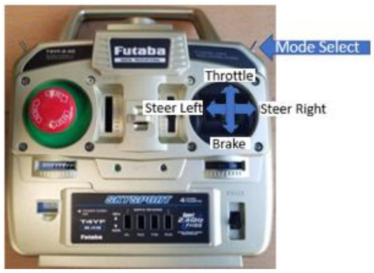

Figure 7: Remote controller potentiometer stick controls [2]

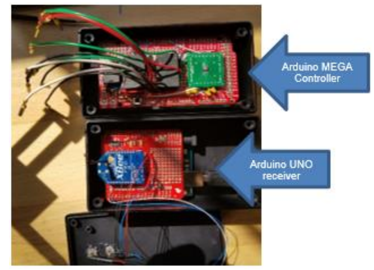

Figure 8: Remote control receiver module (bottom) and vehicle controller unit (top) [2]

Sensors: The Titian-Express is equipped with a suite of sensors for perception, VLP-16 Velodyne LiDAR, 6 – ARS430 RADAR, 4 – SVC210 Continental® cameras, PwrPak7D-E1 Novatel™ and a Sparton AHRS-8 Inertial Measurement Unit.

The VLP-16 Velodyne LiDAR [3] is capable of scanning 360º horizontally by 30º vertically with 16 vertical scanning planes at a maximum range of 30 m. These LiDARs are very efficient at capturing obstacle data accurately for the purpose of developing maps and navigating unknown environments. A VLP-16 LiDAR is mounted on the top front end of Titan-Express and is tilted down at a 15º angle to allow for frontal obstacle detection as well as experimenting with lane line identification. This LiDAR is configured for a 180º horizontal and 30º vertical field of view (FOV) scan as the back side of the LiDAR scan will be aimed at the sky. The full FOV of the Velodyne LiDAR as mounted on Titan-Express is illustrated in Figure 9.

The ARS430 Radar [4] provides different fields of views based on the detection distance and objects of interest. The main usage of the radar is to provide long range detection/classification mechanism that can reach up to 80m. The ARS430 is a 77 GHz radar sensor with digital beam-forming scanning antenna which offers two independent scans for far and short range. The RDI radar system provides a Radar Detection Image (RDI) over ethernet. The Radar Detection Image data steam provided by the sensor that includes all collected radar data above a defined noise threshold with Azimuth/Elvtion, distance and relative velocity attributes attached. On the Titan-Express six radars are used, which are mounted to provide a comprehensive converge for the vehicle front and back area. The six ARS430 RADAR's FOV is shown in Figure 10, and both long-range and short-range detections are highlighted for every RADAR. The maximum pedestrian detection range is 80 m for this setup.

There are four SVC210 Continental® cameras [5] mounted on the front hood, back bumper, and side mirrors of Titan-Express. The SVC210 camera has 1280x1080 resolution and a super fisheye optics with 190º horizontal and 118º vertical FOV. Given the current mountings, the SVC210 cameras provide full coverage for the vehicle surrounding as illustrated in Figure 11. The four cameras have various functionality. The first camera is pointing in the forward direction, FOV shown in Figure 12, and is used for sign detection. There is a second camera in the rear is for detecting lines for parking, and FOV shown in Figure 13. The third and fourth cameras are for vertical and horizontal lane detection in the forward direction. The third and fourth cameras' FOV, driver and passenger sides of vehicle, is shown in Figure 24, both are rotated to see the lane lines on each side of the vehicle.

The PwrPak7D-E1 Novatel™ is a combined GPS and IMU solution that provides filtered position based on both sensor readings. This approach guarantees uninterpreted global localization even when intermittent GPS outages are experienced. The pwrPack 7D-E1 [6] is further equipped with dual antenna system and can provide 40 cm accuracy with TerraStar-L subscription or DGPS, 4 cm accuracy with TerraStar-C subscription, and 1 cm +1ppm with RTK. The GNSS Measurements is captured at a rate of

20 Hz. This level of accurate measurement and fast update rate enables Titan-Express to perform mapping tasks with high accuracy. The antennas are mounted inline down the center of the vehicle.

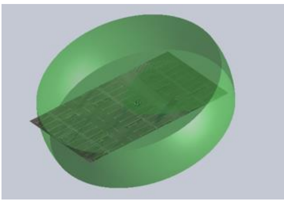

Figure 9: Velodyne VLP-16 Field of View as mounted on Titan-Express

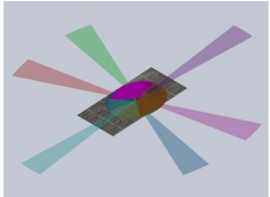

Figure 10: Coverage Area for the Six ARS430 Continental ® RADARs in Titan-Express

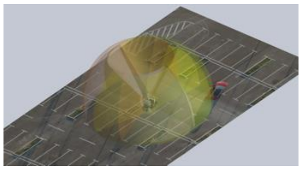

Figure 11: The Combined Field of View for the Four SVC210 Camera in Titan-Express

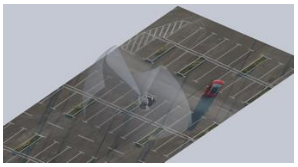

Figure 12: Front SVC210 Camera in Titan-Express for Sign Detection

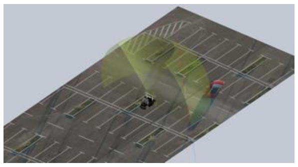

Figure 13: Rear SVC210 Camera in Titan-Express for Line Detection

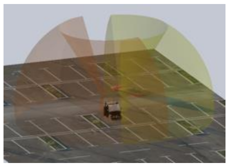

Figure 14: Driver and Passenger Side SVC210 Cameras in Titan-Express for Line Detection

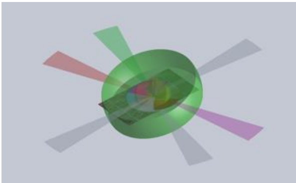

Figure 15: The Cumulative Sensing FOV of Titan-Express

Computation: With the wide array of sensors that is utilized in Titan-Express, there needs to be sufficient computational power to support collecting and processing all the sensor data. Three computing systems are used in Titan Express: NVIDIA Drive PX2, general processing computer, and a Trimble Kenai tablet.

The Nvidia Drive PX2 has two Tegra X2 SoCs and two Pascal generation GPUs. This unit has immense processing power and leans itself to applications in deep learning and image processing. Furthermore, the Drive PX2 comes with 12 GMSL ready interfaces, and hence, can capture data from 12 cameras at a Gbit rate. It also has CAN, LIN, FlexRay, and automotive Ethernet interfaces and all the standard PC communication interfaces.

The Robot Operating System (ROS) will be installed on all computational platforms. All sensor data will be captured and processed by the Nvidia PX2 and converted to ROS compatible messages. A Trimble Kenai tablet is in place for human machine interfacing. All these computing units will be connected to a private network inside the vehicle via wireless or wired interfaces and will be synchronized when started. The overall configuration of the vehicle sensor and computation systems is illustrated in Figure 16.

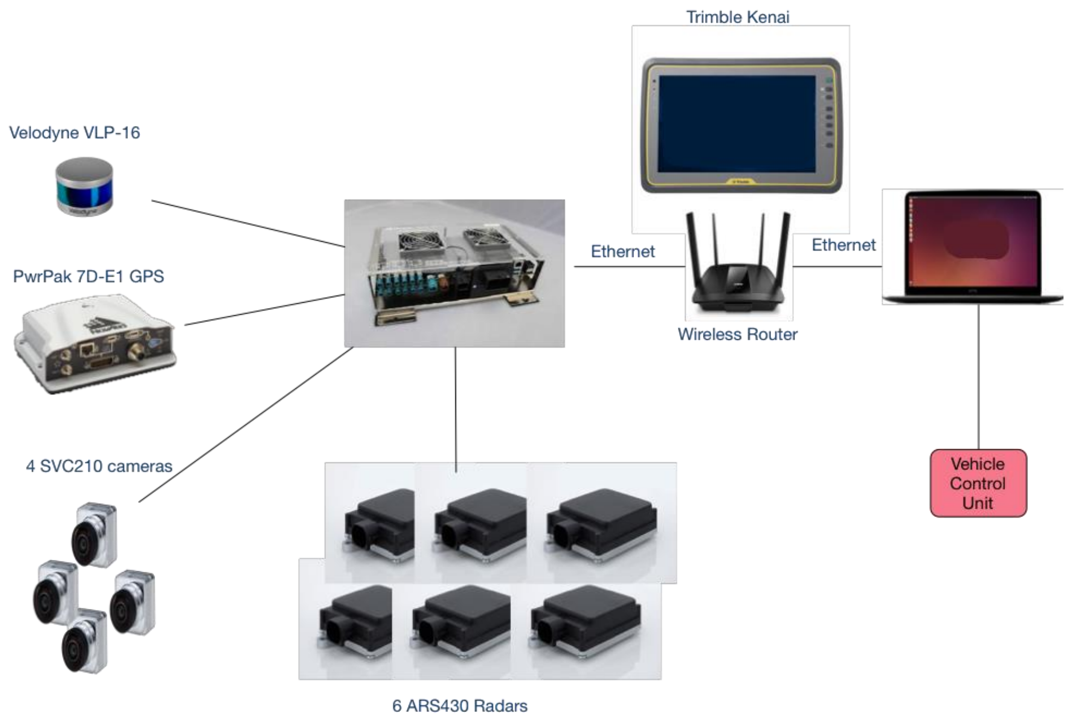

Figure 15: The Overall Sensor & Computation Resources Integration

Software Strategy and Mapping Techniques

The systems that will be implement on the Titan Express are obstacle detection/image processing, localization, map generation, and goal selection and path planning. All of these subsystems will be combined in a way that allows for the vehicle behaviors outlined in the IGVC rules.

Development Environment: For the Titan Express all the algorithms are developed for the Robot Operating System (ROS), a distributed computing system for robots. The subsystems of the vehicle are all employing ROS for integration of the system.

Obstacle Detection Image Processing: The image processing is done in the Nvidia PX2 system. The Nvidia PX2 is configured to use LaneNet and a deep neural network (DNN) for sign detection. LaneNet is a proprietary neural network developed by Nvidia for lane line detection. The PX2 was developed in 2016 and is capable of 8 FP32 TFLOPS.

The LaneNet algorithm performs well for detecting lanes under many conditions: Solid, dashed, and curved lanes. The lane lines can be detected in sunny days, overcast days, and at night. Sample detection images are shown in Figures 16 through 18.

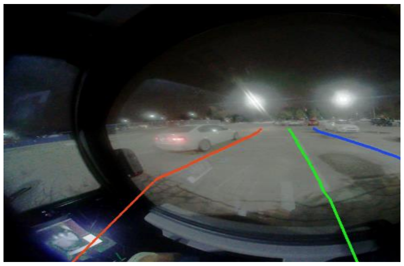

Figure 16: Sample of Lane Detection at Night

Figure 17: Sample of Lane Detection on road

Figure 18: Sample of Lane Detection with Curved Lanes

The main disadvantage of LaneNet is that it cannot detect lanes running horizontal across the image, which is useful for parallel parking and turning at intersections. Therefore, separate systems for detecting horizontal lanes are needed, or a separate camera mounted perpendicular to the vehicle's front detection.

For the lane detection SVC210 Continental® cameras, fisheye cameras, are used as they provide wide field-of-view. Scaramuzza's Ocam fisheye model [7] is used to calculate the orientation of rays emanating from the camera. The intersection of the emanating rays with the ground plane is calculated to produce the location of detected lane lines. After the lane lines are extracted a 3D point cloud of the lines is created and available as a topic in ROS.
For detection of signs a Deep Neural Network is trained to perceive the signs. The neural network is a Convolution Neural Network (CNN) and is trained with the German Traffic Sign Detection Benchmark. Additionally, for texted based signs are detected using sign classification and then Optical Character Recognition (OCR). When the traffic sign detection is complete the classified sign will be published to a topic in ROS.

Localization: Titan Express determines its location using Kalman-filtered GPS and IMU data. The GPS and IMU is a PwrPak7D-E1 Novatel™ receiver coupled with two VEXXIS GNSS-502 Dual Band Antennas. This setup is SPAN-configured, and allows localization within 4 - 40 cm. Two IMUs are used in the Kalman-filtered location - a KVH CG-5100 for angular velocity and acceleration measurements, and an AHRS-8 Sparton for roll and magnetic heading.

Map Generation: The map for decision making is created from an internal package. The package takes the LiDAR data and the 3D point cloud from the PX2. The result is a cost map where the vehicle can and can't go.

Goal Selection and Path Generation: For navigation the Titan Express will use build in path planning algorithms, either A* or Maverick using the generated map. There will be a behavior switch for deciding the action of the Titan Express depending on the signals from external stimuli (signs, emergency stops, etc) and the cost map. In addition to the A* or Maverick there will be an algorithm for goal creation when lane following, where a goal is calculated from the lane line detection data.

Description of failure modes, failure points and resolutions

Numerous failure modes could occur: Batteries could run out of power, weather conditions could disrupt image processing, external components such as cameras or the LiDAR could be knocked off, perhaps by overhanging branches, fast moving obstacles could move into the field of view of the vehicle before it has time to react and stop. The resolution to many of these failure modes is to make sure the vehicle is always fully charged, that care is taken for when weather conditions might be ideal, to watch for

The VCU is a prototype which could be a failure point in the system, but if there is some failure the fail-safe mode of the VCU is to return the vehicle back to its factory connections.

Simulations employed

A Gazebo model for the robot and a test world is built for simulation. The resulting simulation environment is illustrated in Figures 19 through 24.

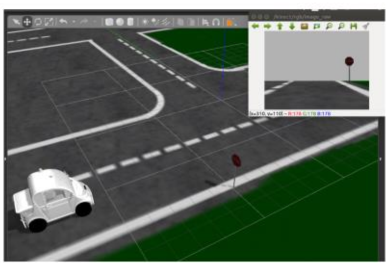

Figure 19: Gazebo Polaris Model in Created Gazebo World Model

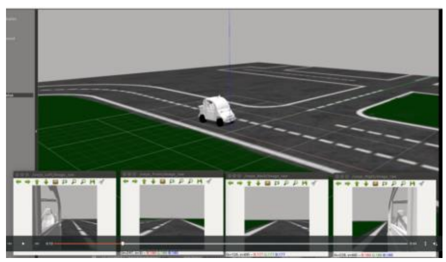

Figure 20: Gazebo with 4 Basic Camera Model

Figure 21: Simulation Joystick Operation

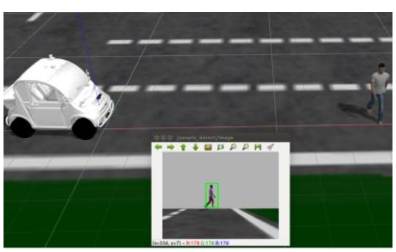

Figure 22: Gazebo Pedestrian Detection

Figure 23: Gazebo Stop Sign Detection

Figure 24: Blurred Sign Detection

Performance Testing to Date

The Titan Express is capable of autonomously navigating between GPS waypoints using the localization package, both on hardware and in simulation. The additional behaviors are currently being implemented. The generation of the cost map for the LaneNet data and the LiDAR has been completed. The sign detection and pedestrian detection has been simulated in Gazebo and working in simulation.

The speed of LaneNet was tested. In the implemented ROS node the time required to run one iteration of LaneNet was measured. The average time over a minute was 121ms per iteration. This result is sufficient for the competition, where the vehicle moves at a maximum of 5 mph. At 5 mph, in one LaneNet iteration, the vehicle moves 0.2 m, a negligible amount.

The accuracy of lane line obstacle detection was tested. The output locations of the 2 lane lines of a parking spot was estimated using the method, as illustrated in Figure 25. The estimated width of the parking spot was compared to the measured width. The average error for 10 trials on different vehicle positions within a lane was 9%, with an error of 0.20m. For example, in the below picture, the estimated width was 2.9 m while the measured width was 2.6 m.

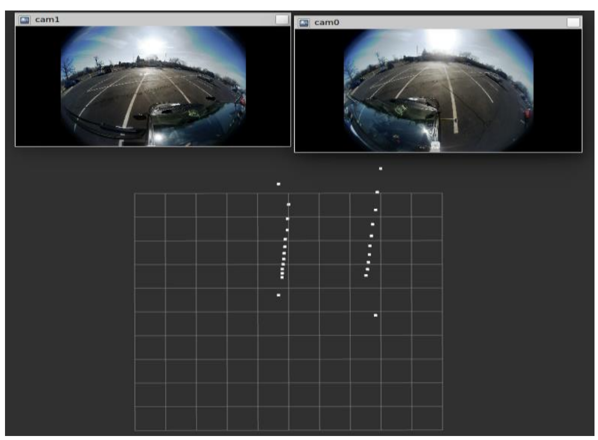

Figure 25: Lane Detection to 3D Point Cloud

A python code has been developed for acquiring data from the ARS Radar. The Continental ARS radar has been interfaced with the computer by using a media-converter from Technicia, and the data from the radar has been acquired. From the acquired data various parameters like header ID, center frequency, distance of the obstacles, its azimuth angle, relative velocity have been parsed out and displayed in the terminal. The radar interfacing and terminal output of the parsed data is shown in in Figure 26.

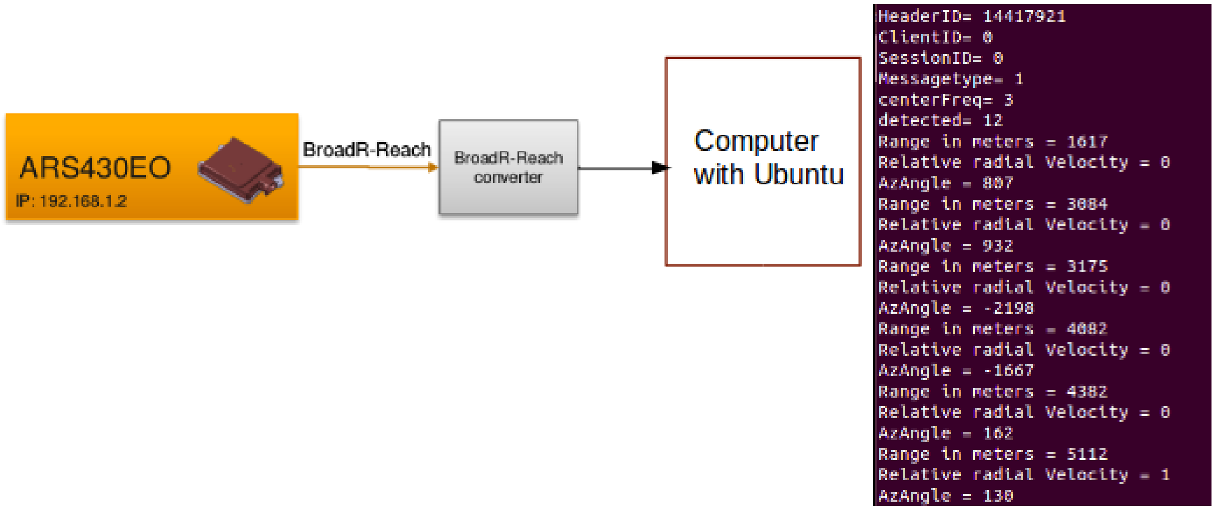

Figure 26. Radar Interface and data from ARS Radar

Safety Systems

There are several safety systems implement on the Titan Express. There is an emergency stop that is activated wirelessly (Figure 7) and a hardwired emergency stop on the dash of the vehicle (Figure 27). In addition, there are Radar that can detect pedestrians at 80 meters, and this can be implemented in safely stopping. The automated steering system uses redundant sensors for error checking, which can signal for a return to user condition of emergency stop. If required the user can also override the autonomous system by having control over the back brakes or by physically counteracting the steering control, which is not ideal. As a last fail-safe the user can turn the power switches off to the autonomous system, seen in Figure 25 as the orange switch covers, to return control to the user.

Figure 27: Dash Emergency Stop and Power Switches

Initial Performance Assessments

Functional: drive-by-wire, brake-by-wire, lane line obstacle detection, GPS waypoint navigation, cost map creation, power system.

References

1. "Model Specifications." Global Electric Motorcars – Passener & Utility Vehicls, gem.polaris.com/en-us/e2/specs/. Accessed Feb. 2019.
2. Viken Yeranosian, Nayan Patel, Joseph Vega, Utayba Mohammad, Mark Paulik, Michael Santora, "A Drive-by-Wire Conversion of an FMVSS-500 Vehicle for Autonomous Driving Applications" SAE WCX 2019.
3. www.velodynelidar.com. (2018). Specs: Velodyne VLP-16. [online] Available at: https://www.velodynelidar.com/vlp-16.html [Accessed 7 Dec. 2018]
4. www.continental-automotive.com. (2018). Specs: ARS430 RADAR. [online] Available at: https://www.continental-automotive.com/en-gl/Trucks-Buses/Vehicle-Chassis-Body/Advanced-Driver-Assistance-Systems/Product-Portfolio/Advanced-Radar-Sensor-—-ARS430. [Accessed 7 Dec. 2018]
5. www.continental-automotive.com. (2018). Specs: SVC210 Camera. [online] Available at: https://www.continental-automotive.com/en-gl/2-Wheeler/Safe-Mobility/Sensors/Surround-View-Camera-—-SVC210. [Accessed 7 Dec. 2018]
6. www.novatel.com. (2018). Specs: PwrPack 7D-E1SPAN solution. [online] Available at: https://www.novatel.com/assets/Documents/Papers/PwrPak7D-E1-PS.pdf. [Accessed 7 Dec. 2018]
7. Scaramuzza, D. Omnidirectional Vision: from Calibration to Robot Motion Estimation, ETH Zurich. Thesis no. 17635. Thesis advisor: Prof. Roland Siegwart. Committee members: Prof. Patrick Rives (INRIA Sophia Antipolis), Prof. Luc Van Gool (ETH Zurich). Chair: Prof. Lino Guzzella (ETH Zurich), Zurich, February 22, 2008.