# Small Scale Intelligent Vehicle

**Sponsor:** Dr. Charles Birdsong
**Advisor:** Dr. Peter Schuster

Sarah De Rosier | Dominic Riccoboni | Paul Rothhammer-Ruiz

**In partnership with:** maxon motor driven by precision

## Problem Statement

The Small Scale Intelligent Vehicle (SSIV) is being developed as a lab platform for a course on intelligent vehicles that will be taught at Cal Poly by Dr. Charles Birdsong. The course will be offered to students of all engineering backgrounds, and will cover topics such as:

- Vehicle Dynamics and Control
- Sensors used in intelligent vehicles
- Data filtering
- Advanced Driver Assistance Systems (ADAS)

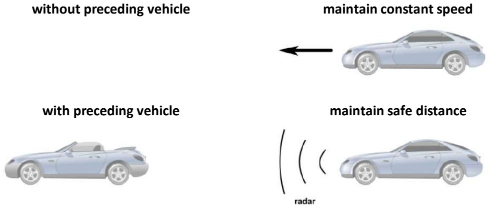
This year's project inherited almost all of the hardware and some of the firmware from µLaren, a Cal Poly Mechanical Engineering senior project team that began this project last year.

To further their work and prepare the SSIV for the course, we integrated sensors, modified and improved firmware, and designed ADAS control systems.

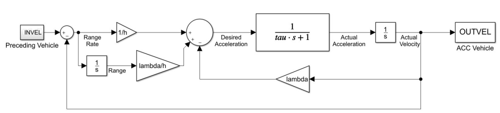
## Adaptive Cruise Controller

### Objectives

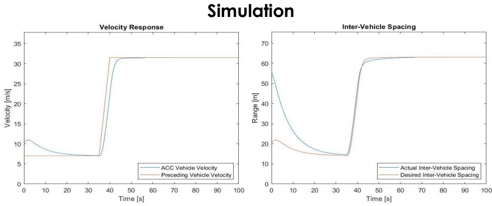
- Maintain constant headway to a preceding vehicle




### Design




### Simulation




- Car tracks the preceding vehicle velocity while maintaining the desired inter-vehicle spacing

## Future Work

1. Configure motor controllers for torque mode, encoder feedback, and other commands
2. Firmware and Simulink communication improvements
3. Produce tighter tolerances in mechanical system
4. Continue refining control systems
5. Implement adaptive cruise controller
6. Improve user interface

**Acknowledgements:** Charlie Refvem
Dr. John Ridgely

Mechanical Engineering CAL POLY

**Motors**

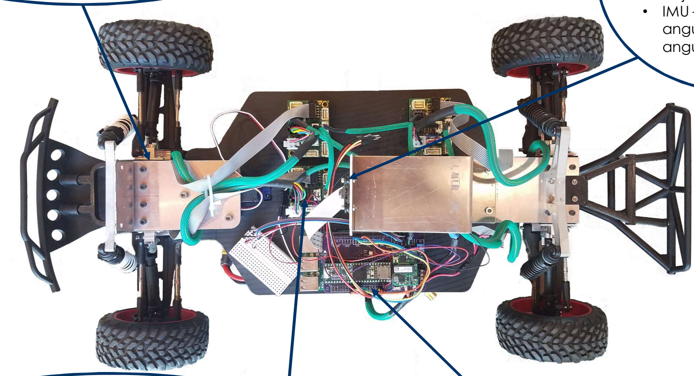
- 1 per wheel
- Maxon 70W Brushless DC motors

**Motor Work**

- Using performance parameter calculations of motors and vehicle dynamics, confirmed that motors selected by last year's team are appropriate

**Sensors**

- Camera – computer vision
- GPS – location
- LiDAR and ultrasonic – object detection
- IMU – accelerations, angular velocities, and angular positions



**Motor Controllers**

- 1 per motor
- Provided by Maxon

**Motor Controllers Work**

- Made progress using CAN to configure the controllers to run in multiple different modes—for use in different control algorithms

**Firmware**

- Raspberry Pi 3B
- Teensy 3.6 Microcontroller Board

**Firmware Work**

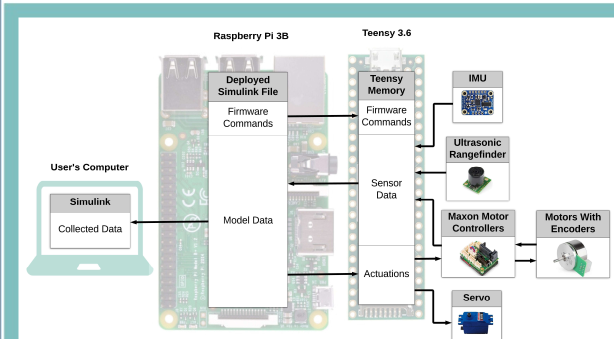
- Code to communicate commands, sensor data, and actuations between Teensy and Simulink on Raspberry Pi

Sensors give the car feedback from its environment including obstacles and lane lines. They also provide feedback on the car's motion and position.

The car approaches an obstacle on the lane line.

The camera runs a computer vision algorithm to detect the lane lines.

The LiDAR detects the obstacle on the roadway.

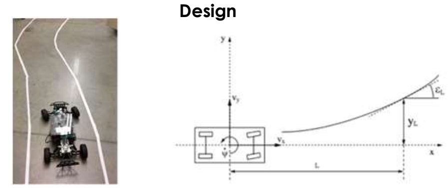



## Lane Keeping

### Objectives

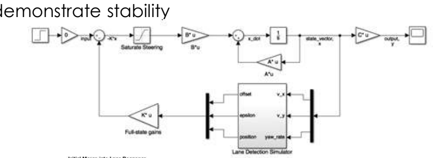
- Detect location of lane lines
- Control the vehicle's position within the lane line

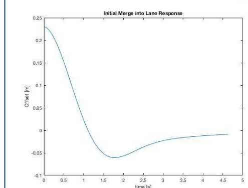
### Design



### Simulation

- Full state feedback controller was tested in a Simulation to demonstrate stability



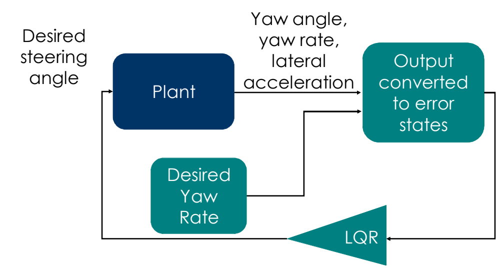

Controller responds well to an initial offset from the centerline

## Yaw Control

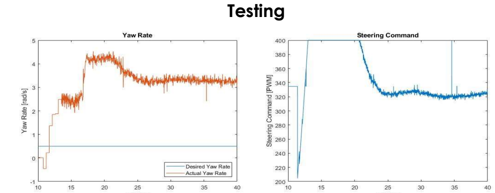
### Objectives

- Track desired yaw rate by controlling steering angle

### Design




### Testing




- Controller responded to a desired yaw rate
- Difficulties include: constructing state feedback from IMU data, incorrect yaw rate measurement, and controller gain tuning