EMBRY-RIDDLE Aeronautical University

Mechanical Engineering

ignite CREATE • RESEARCH • CHANGE

Instrumented Research Vehicle for Quantifying Real-World Emissions

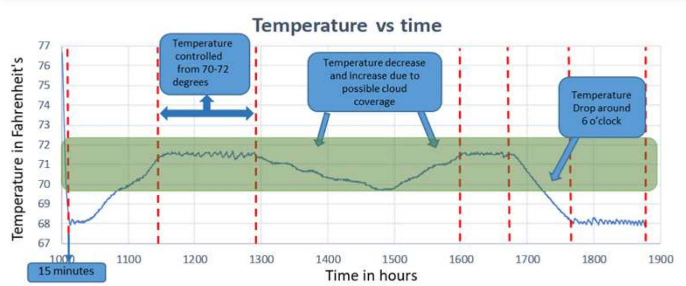
Jonathan Prine
prinej1@my.erau.edu
Arnaldo Carrasquillo
carrasa2@my.erau.edu
Angel Song Liu
songliua@my.erau.edu

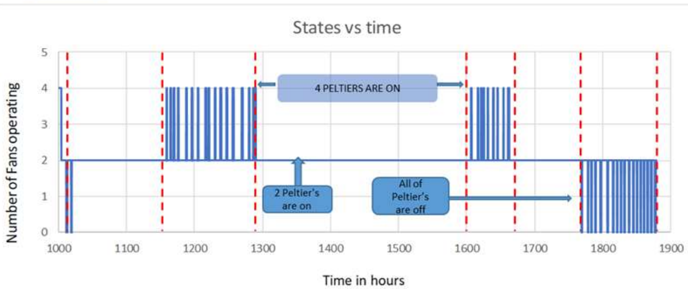
Computing enclosure

The Computing enclosure houses a variety of microcontrollers such as Raspberry Pi 3, Arduino Mega, GPS Module, Internet Router, and a Emission Analyzer. Cooling the microcontroller are solid-state Peltier coolers that control the internal temperature with a bang-bang controller. Each device has the capability of removing up to 50 Watts of heat, totaling 200 Watts for the enclosure. Applying Thermal Ohms law, it was calculated that in direct sunlight the enclosure would need 4 Peltier Devices to cool the system efficiently. The Peltier were equipped with 8 fans directly on then to promote convective cooling. An additional level protection was a carbon fiber heat shield being fabricated to lessen the heat entering the enclosure. Internal to the enclosure Acoustic Polyurethane foam was added to secure components while providing and extra layer of insulation from heat and vibration from the engine. The Peltier's are controlled by an Arduino Mega that contains a simple switch case statements to control the temperature between 50-80 F. It also takes into consideration the saturation temperature at which the enclosure would condensate. This is essential to prevent failure of the components. To combat this desiccant package were added to absorb any water that the Peltier coolers produced. Using a SD shield the Mega can collect the average temperature and the state of the Peltier's, as seen below.

Computing Enclosure Requirements:

1.1 Shall fit within the vehicle bed
1.2 Shall report interior temperature data for the last 7 days
1.3 Shall provide easy access to interior
1.4 Shall ensure no chafing on power and data wires
1.5 Shall securely mount to vehicle (no vertical or horizontal movement while driving)
1.6 Shall fit all sensors, computers and display with vibration isolation
1.7 Shall regulate temperature between 50 F – 80 F
1.8 Shall not drip condensation on the interior (avoid dew point)
1.9 Shall provide NEMA Type 4 protection for dust, UV, and hose-down





Abstract

Embry-Riddle Aeronautical University has developed an Instrumented Research Vehicle (IRV) for collecting real-world emissions with remote monitoring via live streaming telemetry. The long-term research goal is to experimentally quantify emissions over repeatable campus drive cycles with and without automated 'traffic assist' control laws. This paper presents vehicle instrumentation, drive cycle definition, and baseline results. A Diesel, 4-seat campus vehicle has been equipped with a custom weatherproof, outdoor computing enclosure to house sensors and computers. Using Embry Riddle's campus as a driving environment to collect data using an emission analyzer to quantify the emissions being produced. An Enerac M700 Portable Emission Analyzer is installed inside the enclosure along with IMU, GPS, and throttle, brake, and steer angle sensors. The outdoor computing enclosure is temperature regulated using thermo-electric devices and a solar heat shield. The enclosure improves reliability of low-cost prototyping hardware such as Arduino and Raspberry Pi computers. Sensor measurements are collected on-board and streamed at a lower rate via mobile phone network to an Internet-of-Things (IoT) server for real time, web-based monitoring. Live streaming telemetry architecture and software components are described. The web-based browser routinely achieves >10Hz vehicle updates using open-source software and consumer grade mobile devices. Current data output includes geo-tagged emissions correlated with driver throttle, brake, steering, vehicle speed, orientation (yaw, pitch, roll) and location along the driving course. This driving platform will provide valuable sensitivity data to focus subsequent research efforts on emissions and energy reduction.

Software and Instrumentation

The sensors and software record data while driving to an on-board computer and a remote display server for real-time telemetry. Programming languages include Arduino, Python, and MATLAB's Simulink. The primary data logging computer is a Raspberry Pi 3(RPi3), which is a single-board computer that collects data from all sensors and a remote linux-based server for monitoring in real time. Sensors include:

- uBLOX GPS receiver
- 9-DOF Inertial Measurement unit (IMU)
- Enerac M-700 Emission Analyzer
- Steering, throttle, and brake position
- Four wheel encoders for odometry
- Driver's view camera

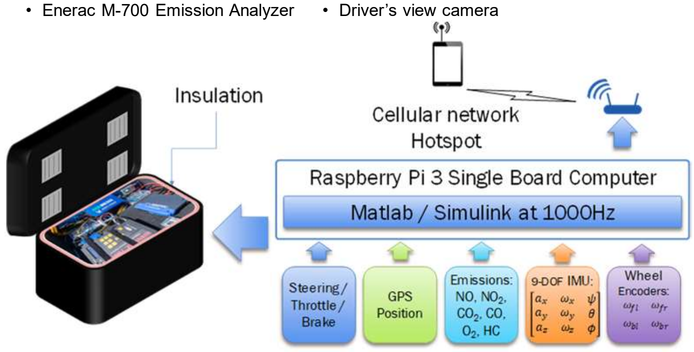


Figure 1: On-Board Vehicle Sensors Diagram

Design requirements for the telemetry and data logging include a high update rate (10Hz - 30Hz), modularity for debugging, local high frequency logging for post-processing, a remote observation on any common platform (Mac/PC/Tablet/iPhone/Android) without having to download or update an application. The application to monitor sensor logging was programmed using *kivy* which is a Python GUI library. A TCP tunnel was coded in Python to provide a connection to the single board microcomputer over a cellular network. Live telemetry is displayed using a Python web-based visualization library called *bokeh*. The telemetry provides emissions, GPS and driver inputs at around 10Hz over a publicly available cellular network.

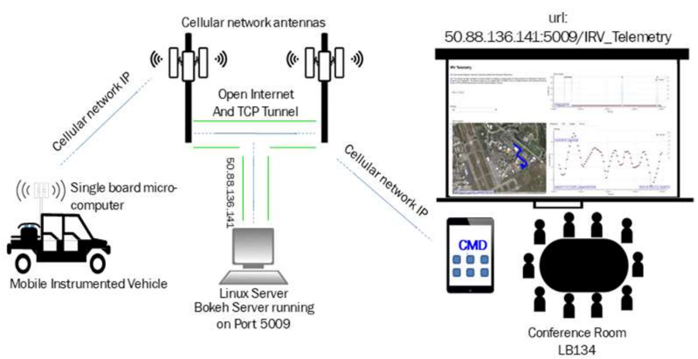


Figure 2: Network communication diagram

EPA United States Environmental Protection Agency