AS Capstone Project Proposal

Winter 2021

ICE Reader

Fredrick Collins

March 15th, 2021

Project Description

I am proposing a plug-and-play solution to fault code reading, real-time feedback, data logging, and analysis for the commercial vehicle via the OBD-II port through ICE (*Internal Combustion Engine)* Reader. Utilizing a Raspberry Pi Zero for computing, ICE Reader will operate by drawing power from any 12V plug and interfacing with the ECU (engine control unit) over micro-USB to query and return a range of useful values. After powering on the device, ICE Reader will draw to an on-board OLED display complete with a joystick and two buttons to load a start menu. From the menu, users can start/stop data logging, check for diagnostic trouble codes, or select a real-time readout of values such as speed, intake pressure, rpm, and more.

This solution doesn't stop when the car does, because if the user wishes to view and manipulate a variety of data series they have logged, they can connect to the Raspberry Pi's own wireless network and *visit the ICE Reader website* to chart, study, view reports, and compare recorded data from their database as they please.

Note: This project will be expanding upon work I had done in my senior year of high school. Previously, I had established a connection between the Raspberry Pi and the ECU, queried intake pressure, and displayed it to the screen in real time - the buck stopped there. All user inputs via menu, queried values, data logging, data manipulating, and charting features will *be new. For reference: 40 lines of python code vs near 1,000 of python, client-side js, server-side js, css, and html.*

Features

- Read and record common ECU fault codes
- Start and stop recording a wide array of engine variables in real time via user interface
    - Display a wide array of engine variables in real time via OLED screen
- Charting capability supporting multiple series on the same x-axis
    - Adjustable display scale, automatic high/low/mean/r^2 labeling
    - Ability to add moving averages to smooth data

Technology

- ❑ Raspberry Pi Zero W
    - ❑ Microcomputer running Raspbian, a linux-based operating system
    - ❑ Will run its own web server and put up a connectable network with hostapd
- ❑ Python 3
    - ❑ Programming language, will be used to write the on-board software for the reader
    - ❑ Several libraries for utility and connectivity will be imported
- ❑ Javascript/HTML/CSS/NodeJS
    - ❑ Suite of web languages that will be used to interact with an SQLite database and provide a user interface and charting application

❏ NodeJS backend serves requested files from the webserver and interprets them

Special Consideration: This device only works on cars made AFTER 1995!