Charging Hardware & Software

STUDENTS: Maria Lourdes Felix, Noah Monson, Noah Beckford, Xiaoning Jing, Tugrul Efesoy

E-Truck Challenge

- The E-Truck RSO at the UW is a collaboration with PACCAR Inc. to convert a Class 7 Peterbilt 337 diesel truck into a full electric vehicle
- Currently in the 3rd out of its 4-year schedule
- Our team is focused on the DC charging system hardware & software integration

Figure 1: **PACCAR E-Truck**

Charging System

- The CCU uses PLC & PP communication from the EVSE to relay critical charging information to the VCU via CAN
- Understanding these signals & their integration is the focus of this project, providing functional documentation for future teams

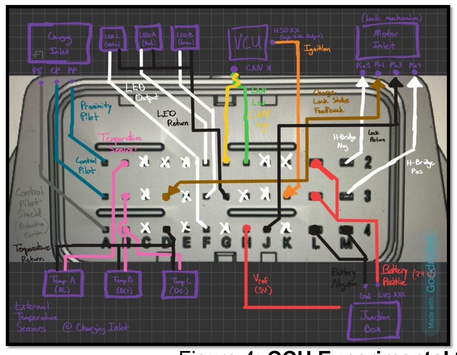


Figure 4: **CCU Experimental Setup**
The left figure is our wiring connection schematic
The right figure shows the CCU & Harness we created

Hardware Integration

- Hardware-In-Loop testing is a standard procedure to simulate signals before component integration
- The Charging Inlet is the port where the EVSE delivers DC Fast Charging
- We simulated the Inlet's handshake signals
  (i.e. inlet's temp & motor lock)

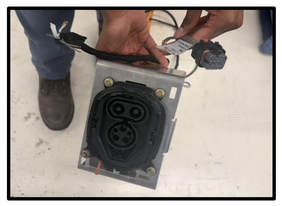


Figure 7: **Charging Inlet**

System Diagram w/ Glossary

EVSE – Electric Vehicle Supply Equipment: Delivers power to the vehicle
CCU – Charge Control Unit: Onboard component that interfaces w/ EV Chargers
VCU – Vehicle Control Unit: Central 'Brain', manages & relays vehicle data
CAN – Controller Area Network: Standard in-vehicle communication protocol

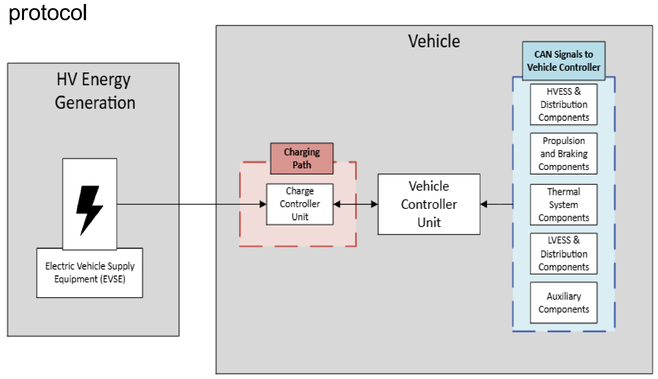


Figure 2: **Block Diagram**
High Level Block Diagram of our charging system & dependencies

International charging standards

- ISO 15118 – Defines EV-to-charger communication, covering session initiation, power delivery, & optional V2G
- SAE J1772 – Defines the standard charge inlet, connector, & signal for communicating charging limits
- SAE J2847 – Specifies DC fast charging message requirements between the vehicle & charger
- SAE J1939 – CAN-based messaging standard for heavy-duty vehicles; used here between the CCU & vehicle controller

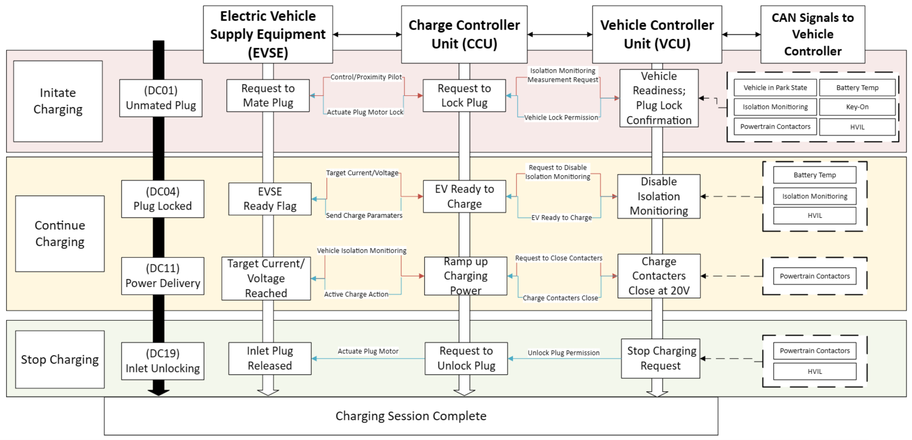
Roadblocks

- Every tool & software used in this project was new to our team, adding to the learning curve
- Verifying the CCU with CANalyzer revealed errors we were unable to resolve, stalling progress
- Design is sequential, so we needed CCU data gained from Hardware-in-Loop to update the Simulink model

Figure 8: **Vector VN1630a**

Figure 3 (right): **Signal Flow Diagram**
Shows approval processes for each part of the signal flow

Software Integration



Simulation Logic in MATLAB Simulink

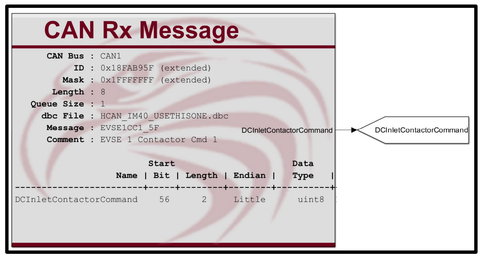
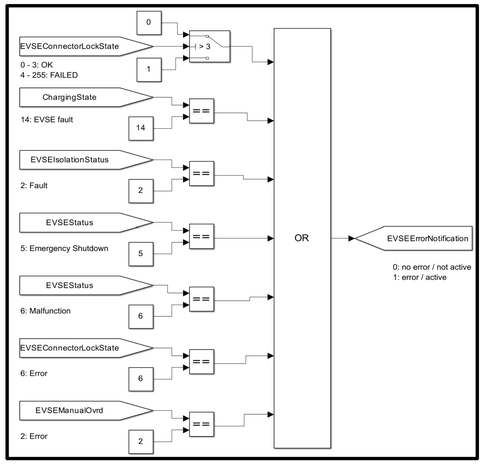


Figure 4 (above): **Signal Input**
Shows received message from CAN bus

Figure 5 (right): **Signal Logic**
Shows logic behind EVSE error notification signal value assignment using other received signals



Future Work and Acknowledgments

- Add advanced error-handling to our MATLAB Simulink Logic
- Complete test bench setup for full hardware-in-loop testing
- Integrate the charging system into the truck for real-world validation

We would like to give special thanks to our mentors Shweta Hardas & John Reece. We would also like the thank the RSO leads Ben Eisenhart & Yusuf Dineh.

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISERS: SHWETA HARDAS, JOHN REECE

SPONSOR: PACCAR Inc