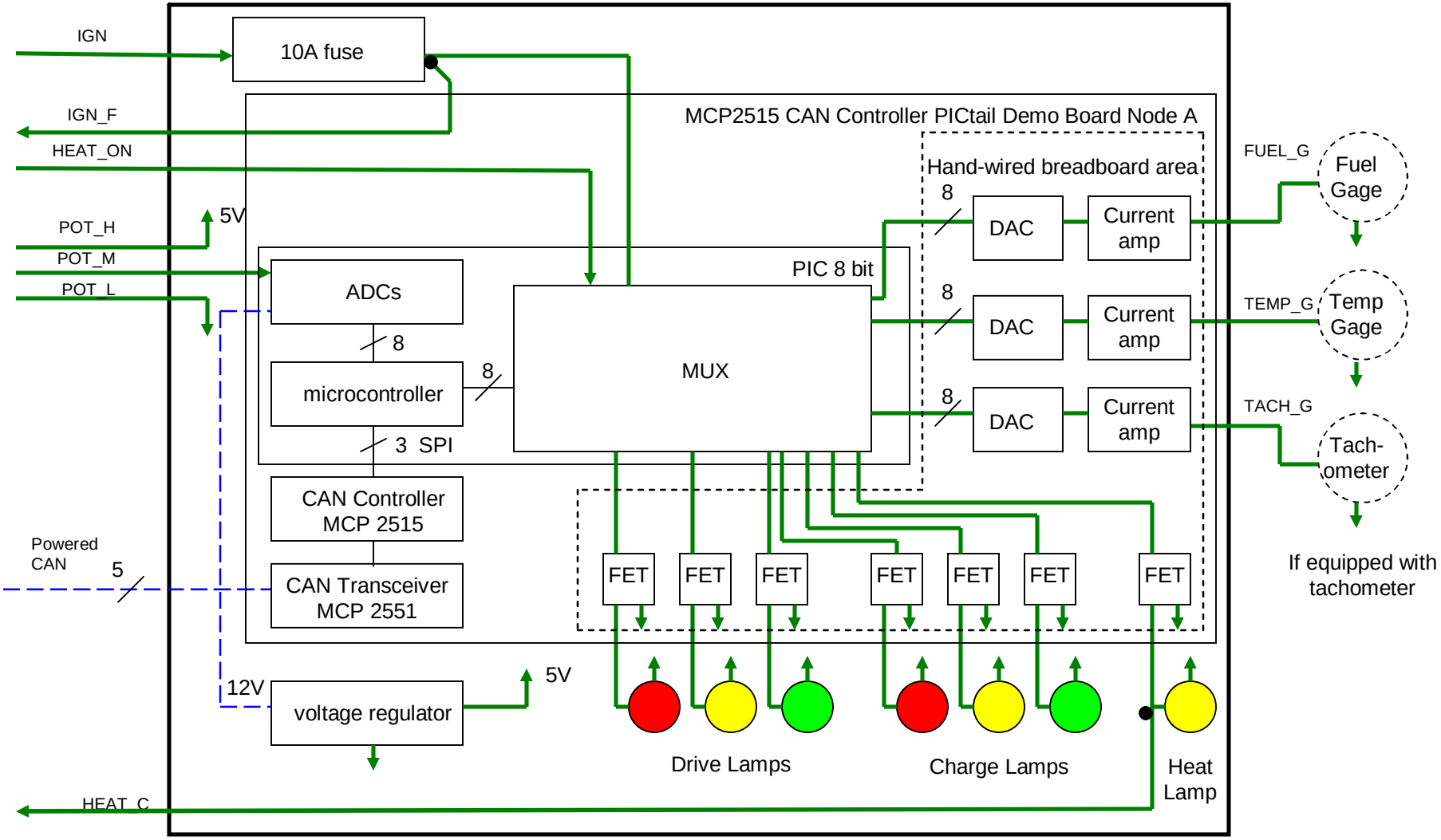
OSEV CONSOLE

Block Diagram



OSEV CONSOLE

Features

Read

From CANBus:

From Battery Management System: status, pack current, pack voltage, cell voltages, cell temperatures

From Motor Controller: status, motor temperature, motor controller temperature, motor RPM

From Charger: status

From Analog to Digital Converters:

Accelerator potbox voltage, DCDC voltage

From I/Os:

Ignition, heat switch

Interim Calculations

State of Charge from cumulative pack current, cell voltages and cell temperatures (unless accurate state of charge provided via CANBus by BMS)

Torque request based on accelerator potbox voltage, cell voltages, cell temperatures, motor temperature and motor controller temperature: cut back gracefully as cell voltage drops and temperatures rise

OSEV CONSOLE

Features

Final Calculations

Fuel gage voltage from, state of charge, fuel gage gain and fuel gage offset

Temperature gage voltage from cell temperatures, motor temperature, motor control temperature, temperature gage gain and temperature gage offset

Tachometer voltage from motor RPM, RPM gage gain and RPM gage offset

Drive and charge lamp switch drive voltages from ignition, motor controller status, charger status, state of charge and DCDC voltage

Heat lamp switch drive voltage from ignition and heat switch

Write

To CANBus:

To Motor Controller: torque request
To Charger: nothing (assumes battery manager sends voltage and current set points)

To Digital to Analog Converters:

Fuel gage voltage, temperature gage voltage and tachometer gage voltage

To I/Os:

Drive, Charge and Heat lamp switch control voltages

OSEV CONSOLE

Breadboard Development Tasks

1. Create detailed schematic, parts list and collection of component data sheets an application notes.
2. Create the detailed software design document.
3. Obtain MCP2515 CAN Controller PICtail Demo Board, PIC microcontroller programming development hardware and software. Estimated cost <$300.
4. Obtain DACs, FETs and amplifier components. Estimated cost <$50.
5. Solder components on to demo board breadboard area and attach them per schematic with wires.
6. Obtain 12V power supply, 13.5V power supply and voltmeter.
7. Write the following microcontroller code in C:
    1. Lamp driver: test with program that turns them on and off automatically; test output with voltmeter
    2. Gage driver: test with program that turns them up and down automatically; test output with voltmeter
    3. Digital input driver: test with program that runs the lamp and gage driver tests based on digital inputs of ignition and heat switch; test output with voltmeter
    4. Digital output driver: test with program that runs the gages based on ADC inputs ; test output with voltmeter
    5. CAN input driver: test with program that drives fuel gage based on state of charge reported by Valence BMS; requires testing on a car with a Valence BMS
    6. CAN output driver: test with a program that drives an Azure Dynamics motor controller in CANBus mode by potbox input; requires testing on a car with an Azure Dynamics motor controller
    7. Implement and test all of the features on the bench, then in vehicles
8. Document design in specification and application note forms as appropriate.
9. Obtain design and performance feedback from EV community and accommodate changes in breadboard and documentation.

OSEV CONSOLE

Future Potential Features

Pulse error code over status lamps when ignition key pulsed 3 times so no computer is needed to troubleshoot to component level

Store data for troubleshooting and development uses

Reduce drive power rate based on input dial or economy mode switch, that turns off temporarily when accelerator maximized

Reduce charger rate based on input dial located on console, under hood or behind fuel door

Acquire and store additional data such as distance, GPS position and ambient temperature

Develop software that can be installed on a PC that extracts the data easily via USB to CANBus converter

Develop software that can be installed on a PC that displays the CANBus data real time in graphical and tabular format for troubleshooting and development uses

Develop software that searches through the data, calculates and displays key EV parameter trend data and status over time