Development of a Series Parallel Energy Management Strategy for Charge Sustaining PHEV Operation

Peter Christopher Manning

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
In
Mechanical Engineering

Chair: Douglas J. Nelson
Alexander Leonessa
William T. Baumann

May 29, 2014
Blacksburg, VA

Keywords: hybrid electric vehicle, plug-in, EV, PHEV, fuel consumption, model-based design, powertrain selection, control system validation, simulation, software-in-the-loop, hardware-in-the-loop

Development of a Series Parallel Energy Management Strategy for Charge Sustaining PHEV Operation

Peter Christopher Manning

ABSTRACT

The Hybrid Electric Vehicle Team of Virginia Tech (HEVT) is participating in the 2012-2014 EcoCAR 2: Plugging in to the Future Advanced Vehicle Technology Competition series organized by Argonne National Lab (ANL), and sponsored by General Motors Corporation (GM) and the U.S. Department of Energy (DOE). The goals of the competition are to reduce well-to-wheel (WTW) petroleum energy consumption (PEU), WTW greenhouse gas (GHG) and criteria emissions while maintaining vehicle performance, consumer acceptability and safety. Following the EcoCAR 2 Vehicle Development Process (VDP) of designing, building, and refining an advanced technology vehicle over the course of the three year competition using a 2013 Chevrolet Malibu donated by GM as a base vehicle, the selected powertrain is a Series-Parallel Plug-In Hybrid Electric Vehicle (PHEV) with P2 (between engine and transmission) and P4 (rear axle) motors, a lithium-ion battery pack, an internal combustion engine, and an automatic transmission.

Development of a charge sustaining control strategy for this vehicle involves coordination of controls for each of the main powertrain components through a distributed control strategy. This distributed control strategy includes component controllers for each individual component and a single supervisory controller responsible for interpreting driver demand and determining component commands to meet the driver demand safely and efficiently. For example, the algorithm accounts for a variety of system operating points and will penalize or reward certain operating points for other conditions. These conditions include but are not limited to rewards for discharging the battery when the state of charge (SOC) is above the target value or penalties for operating points with excessive emissions. Development of diagnostics and remedial actions is an important part of controlling the powertrain safely. In order to validate the control strategy prior to in-vehicle operation, simulations are run against a plant model of the vehicle systems. This plant model can be run in both controller Software- and controller Hardware-In-the-Loop (SIL and HIL) simulations.

This paper details the development of the controls for diagnostics, major selection algorithms, and execution of commands and its integration into the Series-Parallel PHEV through the supervisory controller. This paper also covers the plant model development and testing of the control algorithms using controller SIL and HIL methods. This paper details reasons for any changes to the control system, and describes improvements or tradeoffs that had to be made to the control system architecture for the vehicle to run reliably and meet its target specifications. Test results illustrate how changes to the plant model and control code properly affect operation of the control system in the actual vehicle. The VT Malibu is operational and projected to perform well at the final competition.

ACKNOWLEDGEMENTS

First I would like to thank the Hybrid Electric Vehicle Team at Virginia Tech for their research, modeling, work, and support without which I could not have accomplished this study. Eli White has been my “partner in crime” for the past two years and has made the management of the team a joy instead of a burden. Thanks to Jon King for teaching me most of what I know and sparking my interest in pursuing a graduate degree and leading HEVT.

I would also like to thank the U.S. Department of Energy, General Motors, Argonne National Labs, and all of the other sponsors who make Advanced Vehicle Technology Competitions like EcoCAR 2 a possibility. Participation in this program and the GRA funding provided through the EcoCAR 2 competition has shaped my education and graduate studies and will continue to mold my career as I start a job with dSPACE after graduation.

I would like to thank all of HEVT’s local sponsors for their time and continued support. I have learned a ton working with you. Specifically, I would like to thank Stephen Funk, Ethan Filip, and Phil Foster from Kollmorgen for their help in designing, testing, and running the custom P2 generator motor in the VT Malibu.

Next I would like to thank Dr. Doug Nelson, my faculty advisor and the advisor of HEVT, for his vast knowledge of hybrid and automotive systems and his mentorship over the last three years.

Lastly, thank you to my family and friends who have always helped me along my way, no matter what challenge I decided to take on.

All photos by author, 2014.

TABLE OF CONTENTS

LIST OF MULTIMEDIA OBJECTS

LIST OF TABLES

List of Figures

List of Terms & Abbreviations

505	The first 505 seconds of the UDDS drive cycle
ANL	Argonne National Labs
ASM	Automotive Simulation Model
AVTC	Advanced Vehicle Technology Competition
BCM	Battery Control Module or Body Control Module
BEV	Battery Electric Vehicle
CAN	Controller Area Network
CD	Charge Depleting
CS	Charge Sustaining
DFMEA	Design Failure Mode Effects Analysis
DOE	Department of Energy
DTC	Diagnostic Trouble Code
ECM	Engine Control Module
ECU	Electronic Control Unit
EREV	Extended Range Electric Vehicle
ESS	Energy Storage System
ETE	Emissions Testing Event
EV	Electric Vehicle
GHG	Greenhouse Gases
GM	General Motors
GVWR	Gross Vehicle Weight Rating
HEVT	Hybrid Electric Vehicle Team at Virginia Tech
HIL	Hardware-In-The-Loop
HV	High Voltage
HVSC	Hybrid Vehicle Supervisory Controller
HWFET	Highway Fuel Economy Test
ICD	Interface Control Document
MAB	dSPACE MicroAutoBox II (used as the HVSC)
MBD	Model Based Design
P2	Position 2 Generator Motor (located between engine and transmission)
PEU	Petroleum Energy Usage
PHEV	Plug-In Hybrid Electric Vehicle
RCP	Rapid Control Prototyping
RTM	Rear Traction Motor
SIL	Software-In-The-Loop
SOC	State of Charge
TCM	Transmission Control Module
TRC	Transportation Research Center
UDDS	Urban Dynamometer Drive Schedule
US06	Supplemental federal test procedure drive cycle
VDP	Vehicle Development Process
VTS	Vehicle Technical Specifications
WTW	Well to wheel

1) INTRODUCTION

Rapid Control Prototyping (RCP) is a relatively new technique, being introduced to the US automotive market in the last 20 years. An automotive control engineer usually has a host of tasks during the course of development of a vehicle, in both the physical integration and controls domains. RCP helps to decrease the complexity of these tasks. It is a technique used to build, test and validate control strategies. In larger control systems, it provides a way of introducing new functions in complex control strategies. Once the control strategies are validated against a plant model of the system to be controlled, the control design can then be prototyped for testing under real-time conditions. This thesis will detail the use of RCP in the development of a control strategy for a Series Parallel Plug-in Hybrid Electric Vehicle (PHEV).

1.1) EcoCAR 2

EcoCAR 2 is a three-year competition organized by Argonne National Laboratory (ANL) and headline sponsored by General Motors (GM) and the U.S. Department of Energy (DOE) and is the latest installation in a long line of Advanced Vehicle Technology Competitions (AVTC) hosted by the DOE. Virginia Tech's Hybrid Electric Vehicle Team (HEVT) is one of 15 North American universities competing in EcoCAR 2. In this competition the participating teams have the challenge of transforming a conventional gasoline vehicle into a hybrid to reduce greenhouse gas (GHG) emissions and petroleum energy use (PEU), all while maintaining safety, performance, and consumer acceptability. EcoCAR 2 competition non-year-specific rules [EcoCAR 2] drive some of the design constraints, while others are driven by HEVT's own unique team goals. For instance, competition rules state that the vehicle must have a total range of at least 200 miles without incurring penalties. Likewise there are criteria for performance and efficiency categories which influence the design of HEVT's vehicle, including the control systems. The wide array of dynamic tests ensures that teams must appropriately balance performance and handling with vehicle emissions and energy consumption. From these requirements HEVT defined design targets to meet or exceed Vehicle Technical Specifications (VTS) and succeed in the competition. Table 1-1 summarizes VTS requirements and the design targets provided by the competition [EcoCAR 2].

Table 1-1: EcoCAR 2 Design Targets

Specification	Competition Design Target	Competition Minimum Requirement
Acceleration: 0–60 mph	9.5 sec	11.5 sec
Acceleration: 50–70 mph	8.0 sec	10 sec
Braking: 60–0 mph	143.4 ft. (43.7 m)	180 ft. (54.8 m)
Highway Gradeability @ 20 min	3.5% @ 60 mph	3.5% @ 60 mph
Starting Time	<2 sec	<15 sec
Vehicle Range	> 322 km [200 mi]	> 322 km [200 mi]

1.2) VEHICLE DEVELOPMENT PROCESS

To effectively design a hybrid vehicle that can meet competition requirements, it is necessary to develop a control system that effectively communicates and processes signals from components in the vehicle. Delaying control system design in the automotive industry can result in costly hardware iteration in later stages of the overall design process. However, placing a focus on modeling and simulation for control system development can reduce costs and reduce development time. To fully realize the benefits of math-based design, the developed models should be as flexible and reusable as possible.

HEVT followed the EcoCAR 2 Vehicle Development Process (VDP) to complete the redesign of the 2013 Chevrolet Malibu. This VDP closely mimics GM's Global VDP, which is a detailed process that allows for truly decoupled development of subsystems and international work sharing.

The EcoCAR 2 VDP concentrates on advanced technology powertrain development; it uses milestones and key planning deliverables from the GVDP and adds some unique deliverables as well. It helps to establish a plan for the research, development, analysis, and validation of the EcoCAR 2 vehicle design. The EcoCAR 2 VDP assigns time and resources to the pre-hardware and data-acquisition stages of subsystem development that the teams use initially for architecture selection and then continue to use as they progress through the entire competition. The pre-hardware stage of EcoCAR 2 vehicle development sets up processes for hardware selection and the instrumentation of hardware once it has been received.

The EcoCAR 2 VDP is a basic plan showing how subsystem teams interact throughout the many stages of the competition to ensure that the analysis tools are used to complement the hardware phases, not just to gain initial design approval. It puts in place a long-term, top-level plan for the three years of the competition to help ensure that teams do not rush to design conclusions so they can simply get to work on the physical hardware. Figure 1-1 is a graphical representation of the 3-year EcoCAR 2 VDP.

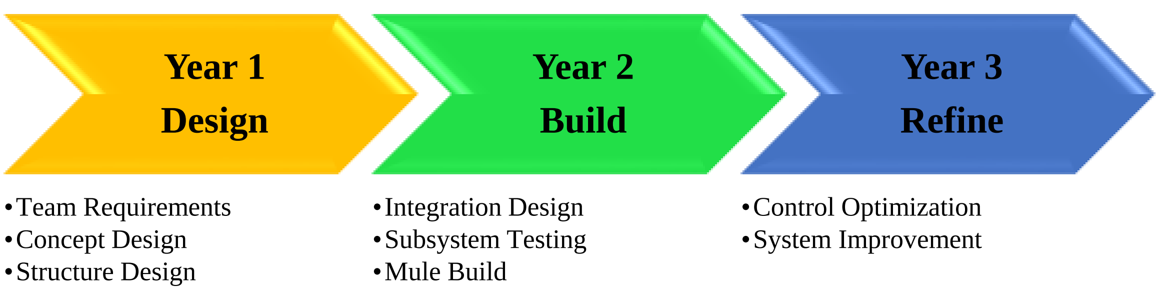

Figure 1-1: EcoCAR 2 VDP

1.3) <u>V-CYCLE</u>

In addition to the EcoCAR 2 VDP, a control system development process, known as the V-cycle. This is a process which defines the steps necessary to successfully design, develop, and validate a control system. Figure 1-2 is a graphic representation of this V-cycle. Note that the left side of the V consists of defining requirements, designing subsystems, and implementation steps. The right side of the diagram includes all of the testing and validation steps.

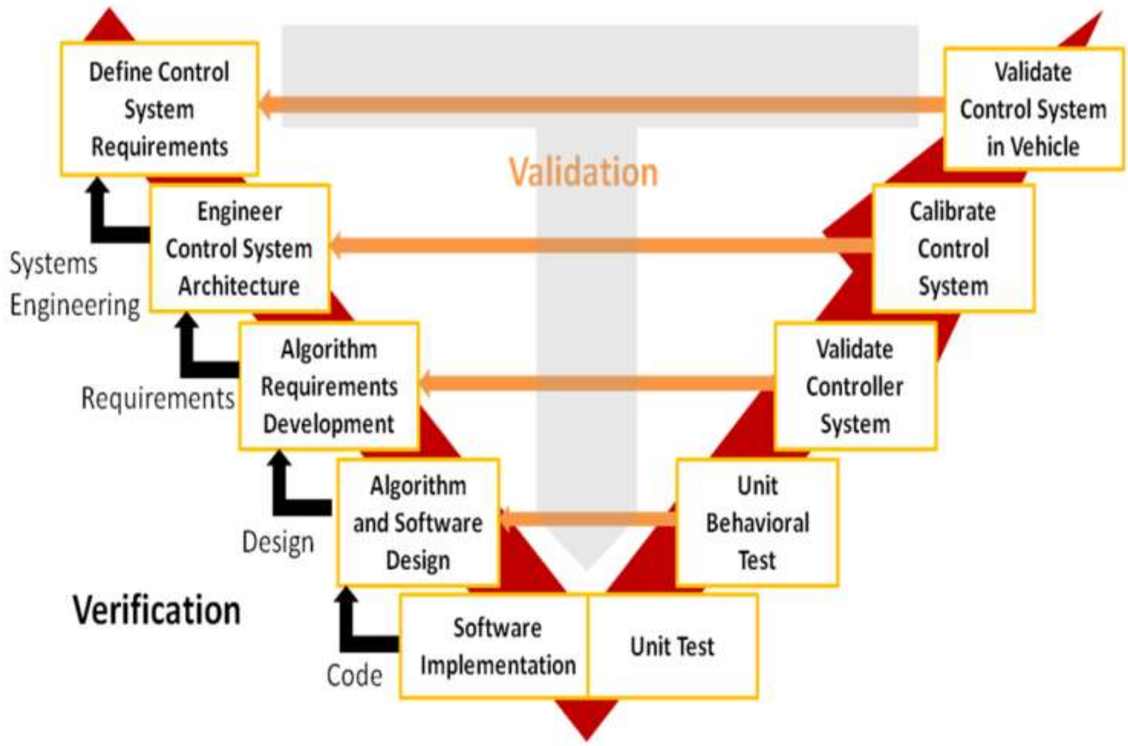

Figure 1-2: V-Cycle

Possibly of more note than the primary steps of the V-cycle are the secondary process flows. These processes are designated by the black and orange arrows in Figure 1-2. The black arrows represent verification steps, or comparison of a step to the defining requirements of the step before. For example, if an algorithm is developed in the *Algorithm and Software Design* phase, then prior to proceeding to *Software Implementation*, the algorithm must be checked against the requirements developed in *Algorithm Requirements Development*.

The other key processes are validation steps, denoted by the orange arrows going from right to left. When the algorithm developed in the *Algorithm and Software Design* phase fails the *Unit Behavioral Test*, the algorithm must be redesigned and all following tasks recompleted.

This V-cycle is a key procedure in the development of a control system which meets the goals of the EcoCAR 2 competition. It is also an industry accepted process and is proven and refined.

1.4) OBJECTIVES

This work aims to document the work done to develop a control system and the control algorithms for an EcoCAR 2 vehicle and all of the steps associated with the EcoCAR 2 VDP and the V-cycle. A discussion of the control system requirements will begin with the system limitations and any assumptions that drive the control system development. These requirements are developed by combining competition requirements, team goals, component limitations, and required interfaces. Control system requirements are developed for each component, for the control algorithms, and vehicle as a whole.

The defining control laws and algorithms which are used in the control system are discussed in detail. Control code was developed to run on a supervisory controller and manage all of the powertrain components. The control algorithms cover system diagnostics, operational mode selection, and final component command execution. Additionally there is an I/O subsystem responsible for signal conditioning and CAN data processing. These control algorithms must be tested and validated by a vehicle model. This model is developed to run on multiple simulation platforms. This model is validated by comparing to actual component test data and vehicle operation. The control algorithms are also validated through component and vehicle testing.

The VT Malibu was subjected to chassis dynamometer testing at the Emissions Testing Event organized by the competition. There, multiple control strategies and operating conditions were run in a closed controlled environment. The results of this testing will be analyzed and discussed in the body of this work. Specifically, the effect of regenerative braking is analyzed in detail. Additionally, the difference between hot and cold starts on an engine is shown in detail with considerations for emissions and drivability.

Lastly, a dialog of the optimization activities is included and final vehicle operation results are discussed. The calibration parameters are presented and discussed and an optimization strategy is shown for the regenerative braking algorithm. The ultimate goal is to have a functioning control system running on the VT Malibu that can meet competition requirements and complete all competition events safely, effectively, and efficiently. Table 1-2 details the goals and objectives of this thesis.

Table 1-2: Thesis Goals & Objectives

1. Develop the requirements for the VT Malibu control system and simulation platforms
2. Test control algorithms and vehicle components to verify functionality and operability and to determine limitations
3. Analyze data from the Emissions Testing Event and discuss the advantages and disadvantages of different control modes

4. Develop and integrate a functioning control system for the VT Malibu which meets competition requirements

2) LITERATURE REVIEW

The automotive industry is moving towards a higher and higher level of software controls and methods of testing these embedded controls through simulation. This shift has been thoroughly documented through the research of others.

2.1) KING: MODEL-BASED DESIGN… (2013)

In the 2013 based research, King lays the groundwork for the research presented in this thesis. The paper begins with a discussion of control system architecture development and a comparison of the pros and cons of a distributed control architecture. Controllers can be located closer to the sensors and actuators they interface with, but there is now a larger focus on reliable communication.

King also introduces a structure of I/O, diagnostics, selection, and execution subsystems to be used in the code for the Hybrid Vehicle Supervisory Controller. The I/O subsystem is reserved for “all signal conditioning and CAN messaging. All data passed between the I/O module and other modules is scaled in engineering units.” The diagnostics module analyzes states of components and input signals and detects faults. Determination of the vehicle state and mode happens in the selection subsystem. Lastly, the execution module takes the desired mode and determines physical commands to send to other components.

The implementation of an online optimization strategy is also discussed after being weighed against an offline optimization strategy. The online strategy implements a “cost” based fuel consumption equation for charge sustaining modes. This equation can be run for many different operating points. The innovative part of this equation comes in the way that King accounts for charging or discharging the battery in terms of equivalent fuel consumption. For charging the battery, there is an equivalent amount of fuel that had to be burned to provide the energy going into the battery. When discharging the battery, the energy coming out of the battery saves an equivalent amount of fuel. These values are both built into King’s optimization strategy along with instantaneous fuel being burned and other miscellaneous factors including emissions factors and state of charge factors, among others.

Lastly, King presents simulation methods and the results from those tests on the developed strategy. Both SIL and HIL techniques are explored and both are used to improve the initially developed algorithm.

As mentioned previously, this research laid the groundwork for the research in this thesis. King’s control algorithm structure is used to develop the control code which runs on the supervisory controller in the VT Malibu. King’s initial algorithms were also developed based on estimations for many controllers and their corresponding interfaces. To be able to use algorithms in an actual vehicle, a large amount of work needs to be completed to get the code vehicle ready. This preparation involves determining actual control interfaces for all of the powertrain components and taking into account the many non-powertrain components which must be controlled and monitored in a drivable vehicle.

2.2) MANNING: DEVELOPMENT OF A PHEV… (2013)

Manning begins the paper by explaining how verification and validation of control algorithms is a rigorous and time consuming process. With the help of rapid control prototyping techniques, designers and developers have cost effective ways of validating controls under a quicker time frame. These techniques involve developments of plant models that replicate the systems that a control algorithm will interface with. These developments help to reduce costs associated with construction of prototypes. In standard design cycles, iterations are needed on prototypes in order to finalize systems. These iterations could result in code changes, new interfacing, and reconstruction, among other issues. The time and resources required to complete these repeated prototypes are far beyond desired.

Manning claims that with the help of simulated interfaces, many of these issues can be recognized prior to physical integration. Like King, Manning primarily uses both SIL and HIL platforms for testing. In this paper, SIL generally focuses on the development of control algorithms and HIL generally helps to finalize the interfaces between the developed controls and the systems that are being controlled.

This paper aims to develop a plant model that will be used in both SIL and HIL testing and to highlight differences between the two testing methods. Modeling the physical plant includes an effort to accurately replicate all signals that are included in the actual vehicle. This paper shows how SIL testing should be used as a feasible option to validate controller algorithms. SIL and HIL testing both end up producing similar results, however, differences can be noticed. Differences in solver characteristics as well as the introduction of realistic IO in HIL tests can be the causes of some of these differences. Issues of latency can also affect the accuracy of simulation results as opposed to final in-vehicle integration. Efforts should be made when modeling systems to represent latency issues. Development of this model and testing strategy is necessary to accurately validate and verify a control strategy prior to in-vehicle integration.

Again, like King, Manning's research lays a large framework for the work presented in this thesis. Many of the component plant models discussed in Manning's paper serve as the starting point for the vehicle model used during simulations to validate the VT Malibu control algorithms. The results of both SIL and HIL testing in Manning's paper, from a general view, are taken into account when performing tests. The discussion of differences between SIL and HIL testing and how those are exhibited and manifested as issues help to improve the reliability of the testing to a large effect.

2.3) HANSELMANN: HARDWARE-IN-THE-LOOP SIMULATION… (1993)

Hanselmann's paper is considered one of the first references related to HIL simulations. The research presented lays the groundwork for the last 20 years of HIL work in the embedded controls industry. At the time that this paper was written, HIL simulations were very new and this paper describes the hardware requirements for early HIL systems. The notion that "the simulation reads ECU output signals which would normally go to actuators and in turn outputs signals which make the ECU 'think' it is controlling a real system" only begin to scratch the surface of HIL testing capabilities today.

Three examples are provided in the paper:

- Complete ABS System – simple model, full ECU, real electromechanical and hydraulic systems
- ABS ECU System – full ECU, modeled electromechanical and hydraulic systems
- Electronic Clutch – full ECU, real proportional valve, modeled engine, car body, clutch, gearbox, driveshaft

Hanselmann also details that "an efficient [Hardware-In-the-Loop system], just a fast processor is not enough." Sufficient input/output hardware and a host computer are also required. The I/O required is fully dependent on the application of use as the unit under test will always have different requirements to fully test it.

The use of a host computer to control the simulation and monitor state space variables is critical as well according to Hanselmann. The host computer allows the user to alter the simulation and to get real time feedback on information not necessarily accessible in a prototype test.

While the development of a HIL system is not a focus of this thesis, many of the ideas provided in Hanselmann's paper help to verify that the HIL system in use is sufficient for the intended uses. This topic of simulation platform requirements is discussed in more detail in Hardware-In-the-Loop (pg. 33).

2.4) WALSH: IMPACT OF SUPERVISORY CONTROL… (2011)

Walsh begins by defining key parameters for a charge sustaining supervisory (torque split) control strategy as well as an engine and catalyst warm-up strategy for a Split Parallel Architecture Extended-Range Electric Vehicle (SPA E-REV). This is accomplished through empirically and experimentally measuring vehicle tailpipe emissions and energy consumption for two distinct control strategies. The results of the experimental testing and analysis define how the vehicle reduces fuel consumption, petroleum energy use and greenhouse gas emissions while maintaining low tailpipe emissions. For a SPA E-REV operating in charge sustaining mode with the engine providing net propulsive energy, Walsh shows that simply operating the engine in regions of highest efficiency does not equate to the most efficient operation of the vehicle as a system and can have adverse effects on tailpipe emissions. Engine and catalyst warm-up during the transition from all-electric charge depleting to engine-dominant charge sustaining modes is experimentally analyzed to evaluate tailpipe emissions. The results presented are meant to define key parameters for a high-level torque-split strategy and to provide an understanding of the tradeoffs between low energy consumption and low tailpipe emissions.

Walsh's research demonstrates some of the tradeoffs that have to be considered when developing a hybrid control strategy. Since EcoCAR 2 places a high priority on both energy consumption and emissions, this research is very applicable when developing the VT Malibu control strategy, however, the related work is not presented in this thesis.

2.5) Summary of Literature Review

The literature review provides background and an introduction of published papers that relate to control systems and testing through –in-the-loop systems. These published papers provide the base work for much of the research presented in the following thesis. While King's and Manning's papers present the development of a control system for Series-Parallel PHEV, little to no research deals with the implementation of control systems these into physical vehicles discussed here.

King provides many examples of results from testing initial control algorithms which are used in the development of the VT Malibu control algorithm and Manning provides testing methods to build upon King's work.

Hanselmann provides a peek into the current HIL testing capabilities by describing the necessary features of any HIL system. A processor capable of handling the plant model, sufficient I/O to simulate the interface of the unit under test, and a host computer for monitoring real time variables are all required elements of a functional HIL system.

Lastly, Walsh explored the tradeoffs of fuel consumption versus emissions, a balance of two of the primary goals of the EcoCAR 2 competition. This work could be useful in further development of the control system for the VT Malibu.

3) VEHICLE OVERVIEW

Prior to development of a control system, a vehicle powertrain was determined and components were selected to meet the requirements of the powertrain and competition.

3.1) STOCK VEHICLE MODELING

In addition to meeting EcoCAR 2 requirements (Table 1-1), HEVT also established its own team goals. Table 3-1 summarizes the additional design goals set by the team. The first goal is to achieve a major reduction in the petroleum energy consumption of the vehicle. This can be achieved by a combination of grid electricity and E85 fuel as both have particularly low well to wheel (WTW) petroleum energy use values. Consumer features are purposefully included in the design as consumer acceptability is represented in a significant portion of competition points. One such consumer feature is a pure Electric Vehicle (EV) mode. The presence of a fully capable EV mode coupled with a significant EV range also greatly displaces WTW petroleum energy and reduces WTW GHG and criteria emissions.

Table 3-1: HEVT Team Goals

Goal	Description
Petroleum Energy Consumption	Reduce petroleum consumption by > 80 %
All-Electric Range	> 56 km (35 mi) range as a pure all-electric vehicle
Passenger Capacity	Retain stock 5 passenger capacity

During Year 1, HEVT used an Autonomie model of the 2013 Chevy Malibu, which was provided to the team by the competition organizers and General Motors. The model provided to the team has many parameters that can be adjusted depending on the desired test. Vehicle mass can be adjusted and a graphical model of the vehicle can be viewed. Drive cycles can be selected from a predefined list and added into the simulation block. Also, user defined drive cycles can be created and added to the list through MATLAB manipulation. A parametric study of different factors can be set up by choosing a type of parametric study and selecting a variable to iterate from, such as engine size. Results including fuel economy and fuel consumption information is output along with input and output energy for the various mechanical components throughout the vehicle and data can be visualized through plots.

For the provided Autonomie model of the 2013 Chevy Malibu, there are 3 inputs defined and 12 components examined. Inputs to the model, not including powertrain components, include the driver, environment and the vehicle propulsion controller. The driver and environment can be adjusted depending on the model and initialization, while the vehicle propulsion controller is created by Autonomie. Figure 3-1 shows a visual representation of the GM provided Autonomie model for the stock 2013 Chevy Malibu powertrain. Table 3-2 also shows the parameters used to simulate each component from the stock vehicle.

Table 3-2: Properties of Model Components in the Provided Autonomie Model

Component	Parameter	Value	Units
12V Battery	Initial SOC	0.7	NA
	Voltage	12	V
Electrical Accessory	Power Loss	200	W
Generator	Gear Efficiency btw Eng. And Gen	1	NA
	Coefficient of Regeneration	1	NA
	Peak Ratio	1	NA
	Fixed Ratio	1	NA
Torque Coupling	Torque Coupling Ratio	0.8	NA
	Speed Threshold	10	rad/sec
Starter	Fixed Ratio btw. Starter and Eng.	10	NA
	Torque req. to start Eng.	30	N-m
Engine	Operating Temp.	90	C
	Maximum Power	133.6	kW
	Response Time	0.3	sec
Mechanical Accessory	Maximum Power	0	W
	Rotational Speed	52.35	rad/sec
Clutch/Torque Converter	Threshold Lock	40	rad/sec
Gearbox	Shift Time	0.5	sec
Final Drive	Ratio	2.89	NA
	Speed Threshold	10	rad/sec
Wheels	Rolling Coefficient 1	0.009	NA
	Rolling Coefficient 2	.00012	NA
	Max Braking Torque	2000	N-m
Chassis	Drag Coefficient	0.295	NA
	Frontal Area	2.295	m^2
	Frontal Weight Ratio	0.64	NA

3.1.1) PARAMETRIC STUDY

Autonomie, a powertrain simulation software suite developed by ANL was used to determine the effects of overall mass and engine scaling on fuel economy. The three masses tested were the base mass of the vehicle (1700 kg), the VT Malibu target mass (2050 kg), and the gross vehicle weight rating (GVWR), 2250 kg. The largest mass was included in tests to show what the fuel economy capabilities of the Malibu are in its heaviest possible state. Also, three engine power values were chosen for scaling. Autonomie initially contained four of the five desired test cycles, with the exception of the US06 City cycle, which was created by isolating the city sections of the US06 cycle.

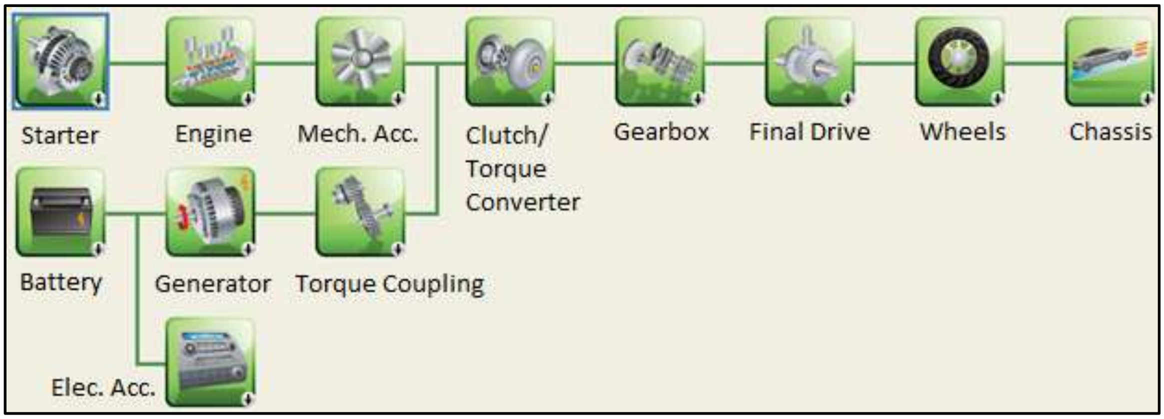

Figure 3-1: Stock 2013 Chevy Malibu Model Components In Autonomie

These parametric studies were necessary to make an educated decision on the final powertrain architecture selection. An understanding of the relationships between different design changes and criteria is critical to make decisions to meet competition goals.

3.1.1.1) ENGINE TORQUE SCALING

The first tests were conducted to test the effect of reducing the power in the engine as well as varying the overall mass of the vehicle. The first value of 134 kW seen in Table 3-3 is the base power of the stock engine. The final value of 104 kW is the power rating for the smallest engine that is available through the competition. Table 3-3 shows that both the power and the overall weight of the vehicle play a significant role in the acceleration time. As the mass of the vehicle is increased and the power is decreased, the acceleration time suffers dramatically.

Table 3-3: 0-60 mph Acceleration Results While Varying Mass & Engine Power

Test Mass	Engine Size		
	134 kW	**104 kW**	**95 kW**
1700 kg	10.8 s	13.9 s	15.3 s
2050 kg	12.7 s	16.6 s	18.3 s
2250 kg	13.9 s	18.1 s	20.1 s

3.1.1.2) DRIVE CYCLE SIMULATION

The model Chevy Malibu was run through 5 different drive cycles to obtain fuel economy results that match today's standards: 505, Urban Dynamometer Drive Schedule (UDDS), Highway Fuel Economy Test (HWFET), US06 City, and US06 Highway. The UDDS represents city driving conditions for light duty vehicles and the 505 test contains the cold start first portion of this cycle. The HWFET represents highway driving conditions for light duty vehicles and has an average speed of 48.3 mph. The US06 cycles are more aggressive and are split into a city and a highway section. With a maximum speed of 80 mph and more aggressive accelerations, the drive schedule is more indicative of contemporary driving behaviors. Figure 3-2 shows both the gearbox ratio and fuel rate for the US06 City cycle as predicted by the Autonomie model. As seen in Figure 3-3, the stock Chevy Malibu closely matched the drive trace of even the rigorous US06 City drive cycle. The vehicle only missed the drive trace by more than 2 mph 4.66% of the time and mostly in the rigorous initial and final accelerations.

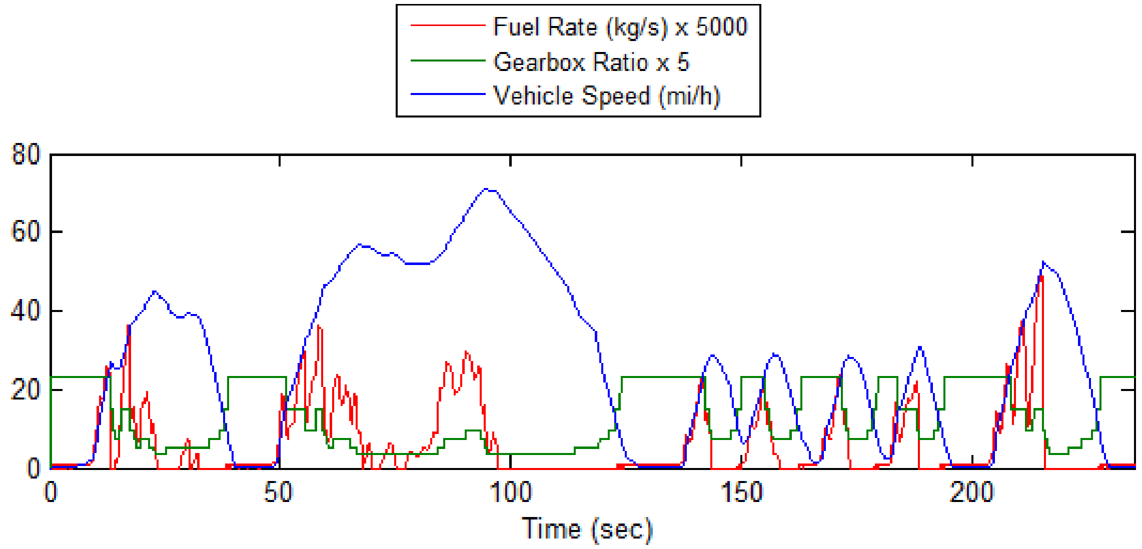

Figure 3-2: Stock Vehicle Characteristics for US06 Drive Cycle (Autonomie)

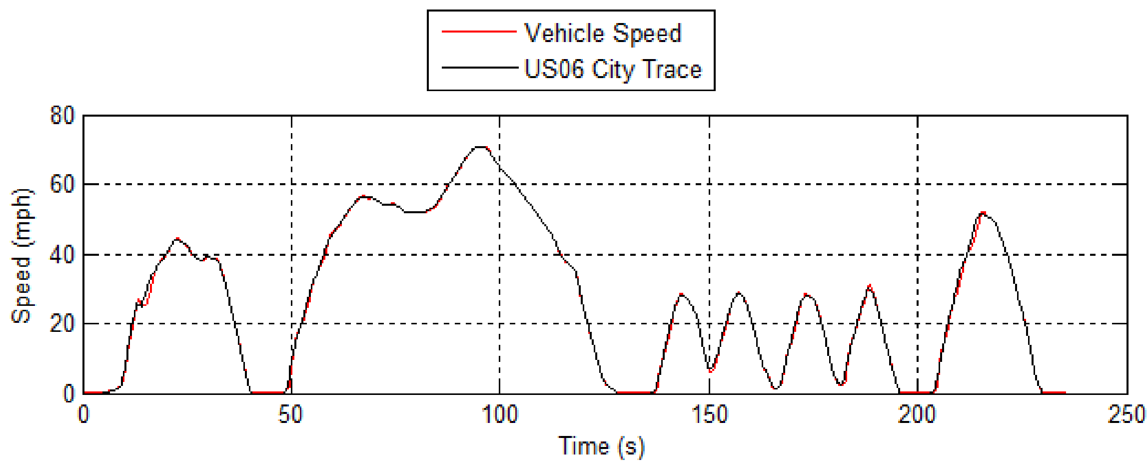

Figure 3-3: Stock Vehicle Velocity vs US06 City Drive Schedule (Autonomie)

3.1.1.3) FUEL ECONOMY SIMULATION

Using the Autonomie model, engine power was scaled for three different vehicle masses as described earlier. The fuel economy results were evaluated for 2 different drive schedules; the 4 cycle test (a weighted average of 4 drive cycles) and the CAFE test (55% UDDS and 45% HWFET). The 4 cycle drive schedule simulates more realistic driving habits and outputs more realistic fuel economy estimations. Any increase in mass directly decreases the fuel economy of the vehicle. This is because the power required to overcome the inertial force increases causing fuel economy to decrease. Although increased mass has adverse effects on fuel economy, this can be counteracted by reducing the engine size. As shown in Table 3-4, fuel economy increased in almost every cycle when using a 104 kW engine as opposed to the base 134 kW engine. However, with a decrease in engine size and increase in mass, the vehicle performance suffers greatly.

Table 3-4: Fuel Economy Results for Varying Test Masses & Engine Scaling

Drive Schedule	Mass (kg)	Engine Power		
		134	104	95
4 Cycle	1700	31.5 mpg	31.8 mpg	31.9 mpg
	2050	28.0 mpg	28.1 mpg	28.2 mpg
	2250	26.1 mpg	26.5 mpg	26.5 mpg
CAFE	1700	40.6 mpg	42.2 mpg	42.6 mpg
	2050	37.2 mpg	37.4 mpg	37.6 mpg
	2250	34.3 mpg	35.2 mpg	35.5 mpg

3.1.1.4) *GRADEABILITY*

To model the base vehicle gradeability at a constant speed, the gradeability simulation in Autonomie was used. The constant linear speed is set to 60 mph, and the gross vehicle weight of 2250 kg is used. Also modeled are the gradeability of the stock vehicle mass and the target EcoCAR 2 mass. Being a sedan, the vehicle wouldn't be expected to be towing frequently, so having no added mass is a valid assumption. The results of the simulation, displayed in Table 3-5, show the maximum grade the stock Malibu can climb, along with values from the engine being scaled down. Based on the relative gradeability results, 134 kW is the base engine power and can climb a grade of 10.1% at a speed of 60 mph with a predicted fuel economy of 7.1 mpg. This grade is much greater than any that would be encountered for long stretches on the highway. As the engine is scaled down to a lower power, the maximum grade the vehicle can climb decreases while the fuel economy increases. Even at the lowest power, the gradeability remains at over 5% which is the typical maximum highway grade. This trend is what can be expected by decreasing the size of the engine.

Table 3-5: Autonomie Model Results of the Gradeability Simulations

Test Mass	Engine Power		
	134 kW	104 kW	95 kW
1700 kg	12.4 %	8.9 %	7.7 %
2050 kg	10.6 %	7.2 %	6.6 %
2250 kg	10.1 %	7.5 %	6.0 %

3.1.1.5) *ENERGY LOSSES*

Using the Autonomie model, drivetrain components losses were calculated. This information is helpful in determining which components of a given architecture are most detrimental to the overall efficiency of the vehicle and where improvements can be made. As shown in Figure 3-4, the largest losses are incurred by the engine. Other significant losses include the torque converter and gearbox. Energy storage, torque coupling and generator losses are relatively negligible. With efficiency defined as energy output at the wheels divided by fuel energy into the engine, the stock vehicle is about 24% efficient in the city and about 28% efficient on the highway.

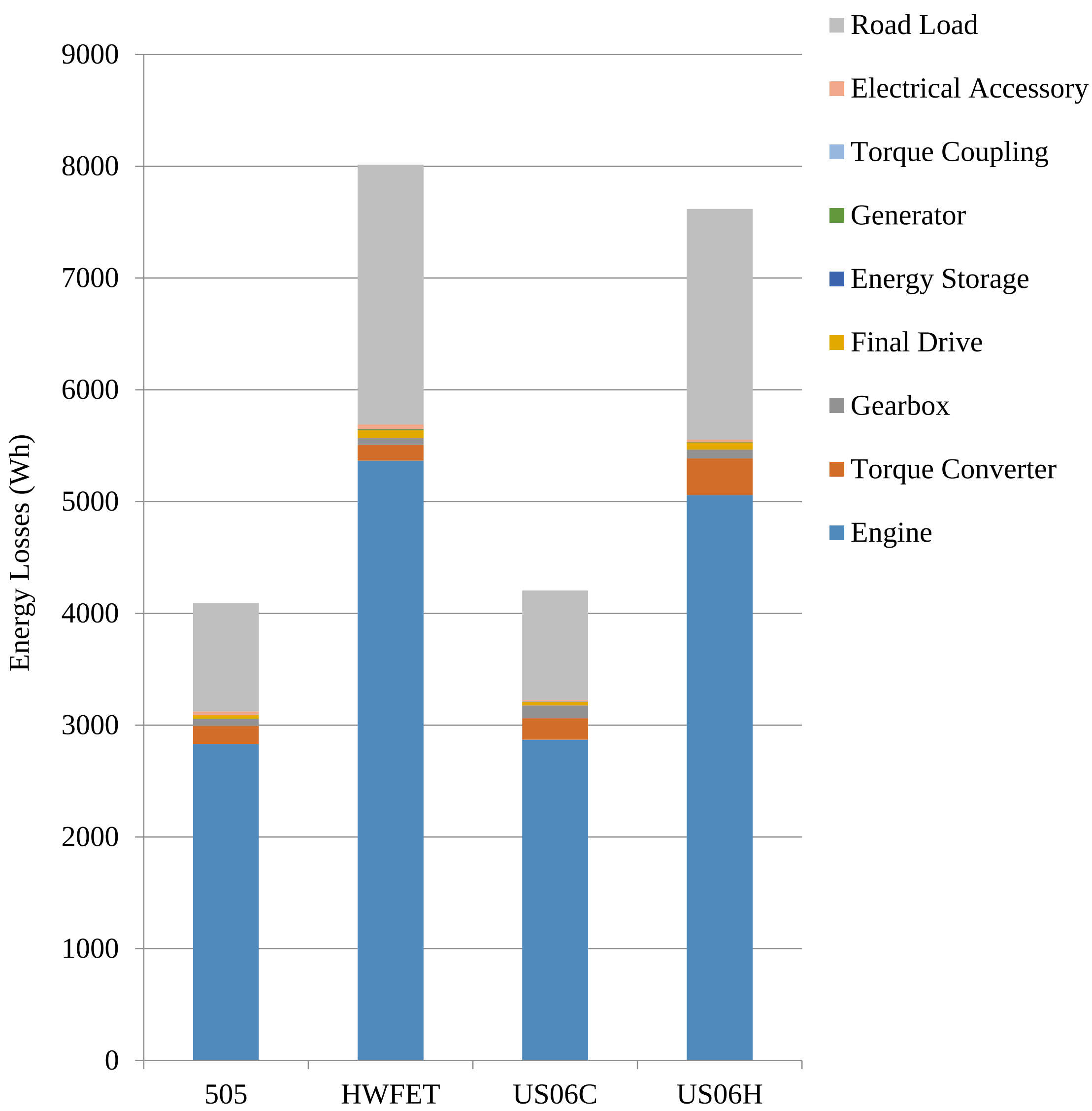

Figure 3-4: Stock Vehicle Distribution of Energy Losses for Tested Cycles

3.2) POWERTRAIN SELECTION

This section discusses the Autonomie simulation and analysis of the two architectures researched: a Battery Electric Vehicle (BEV) and the Series Extended Range Electric Vehicle (EREV). Using Autonomie, each architecture is analyzed and compared in terms of fuel economy, performance characteristics, and PEU and GHG emissions evaluated on a WTW basis. The following sections detail the results from this analysis.

3.2.1) BATTERY ELECTRIC VEHICLE

The first powertrain architecture examined is a BEV. Creating a vehicle that uses only grid electricity would completely eliminate tailpipe emissions, and significantly reduce vehicle energy consumption. This architecture would be appealing to consumers because they would save money on operation costs due to the low cost of grid electricity compared to petroleum based fuels. A high voltage (HV) battery pack of 95 cells is used with a UQM PowerPhase 145 kW motor. Each cell has a capacity of 240 Ah and is capable of producing a continuous current of 240 A and a maximum current of 480 A. This current is sufficient to supply power from the pack without adding cells in parallel. Cells were therefore connected in series to achieve the desired voltage for the vehicle. A single cell has nominal voltage of 3.7 V. 95 cells in series gives a nominal pack voltage of 350 V. This design will allow the vehicle to have full performance and have a 4-cycle range of about 200 miles. Figure 3-5 shows a representation of a BEV architecture.

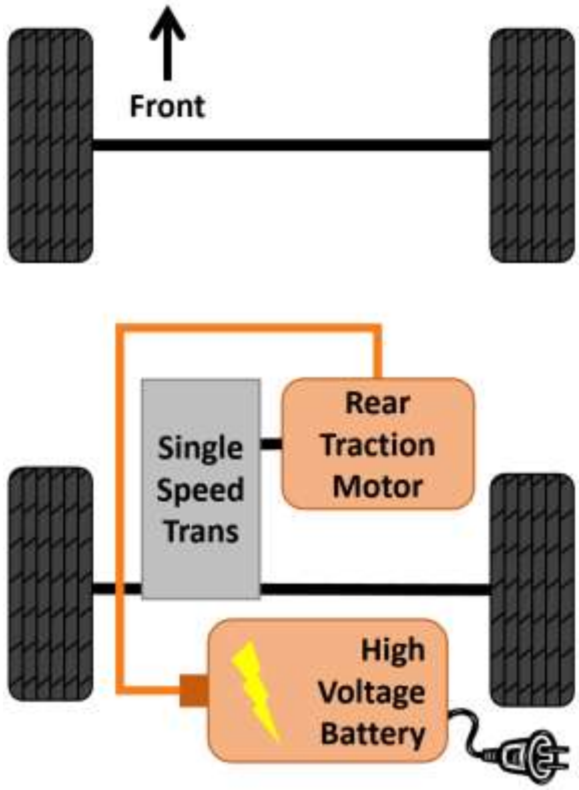

Figure 3-5: Battery Electric Vehicle Architecture

The overall vehicle test weight is 2250 kg, the gross vehicle weight rating, since the mass of the battery pack is significantly large. Table 3-6 summarizes the vehicle component parameters used in the model. This vehicle model is run through a 4-cycle fuel economy test including the UDDS/505, US06 City and Highway, and the HWFET. Energy consumption in AC Wh/mi can be seen for each drive cycle in Table 3-7 below. Vehicle range is estimated using DC Wh/mi consumption results. At GVWR with an 84.4 kWh battery pack, the BEV can barely meet the 200 mile range requirement based on the 4-cycle weighting. Additionally, there is no mass capacity left in the vehicle for additional passengers or any cargo, so this vehicle does not meet the team goals for consumer acceptability.

Table 3-6: BEV Component Description

Vehicle Properties	
Test mass	2250 kg
Energy Storage System	
Energy Capacity	84.4 kWh
Usable Energy Capacity	76.0 kWh
Total Mass	580 kg
Drive Motor – UQM 145 kW	
Peak Power	145 kW

Peak Torque	400 Nm
Continuous Power	85 kW
Gear Reduction	7.17:1
Top Speed	85 mph

After energy consumption and range testing are complete, Autonomie is used to run performance tests. The BEV meets performance expectations with a 0-60 mph time of 9 seconds and a 50-70 mph time of 4.5 seconds. The BEV is able to climb a greater than a 6% grade at 60 mph for 20 minutes. Although the BEV exceeds the performance requirements of the EcoCAR 2 competition, it does not perform well in other areas. Ultimately, a BEV powertrain architecture is a poor choice because of long charging time, difficulty meeting range requirements, and a lack of passenger capacity.

Table 3-7: BEV Performance and Fuel Economy Results

Performance	0-60 mph acceleration	9.0 s
	50-70 mph acceleration	4.5 s
	Gradeability at 60 mph	14.4 %
Fuel Economy	CAFE Energy Consumption	350 AC Wh/mi
	CAFE EV Range	250 mi
	CAFE WTW PEU	11.9 g/mi
	CAFE WTW GHG Emissions	227 g/mi
	4-Cycle Energy Consumption	432 AC Wh/mi
	4-Cycle EV Range	202
	4-Cycle WTW PEU	14.7 g/mi
	4-Cycle WTW GHG Emissions	280 g/mi

3.2.2) SERIES EREV

Despite the benefits of a BEV for WTW PEU, there are also drawbacks. The high mass of the battery pack limits passenger capacity and increases energy consumption. Even with the large battery mass there is difficulty meeting the 200 mile range requirement. Therefore, a series EREV utilizing E85 is considered next. As shown by Figure 3-6, a series EREV has an electric traction motor, a battery pack, an internal combustion engine and a generator attached to the engine. The addition of fuel, engine and generator allow the battery pack to be significantly downsized while still meeting the range requirement. The high energy density of liquid fuel means most of the required range can be met with a small mass of fuel compared to batteries. The vehicle will run in an all-electric or Charge Depleting (CD) mode until the batteries reach a lower state of charge (SOC). At this point, the vehicle will enter Charge Sustaining (CS) mode and the engine will start and power the generator which converts the fuel power into electricity. The electricity from the generator then powers the electric traction motor using energy from the battery as a buffer.

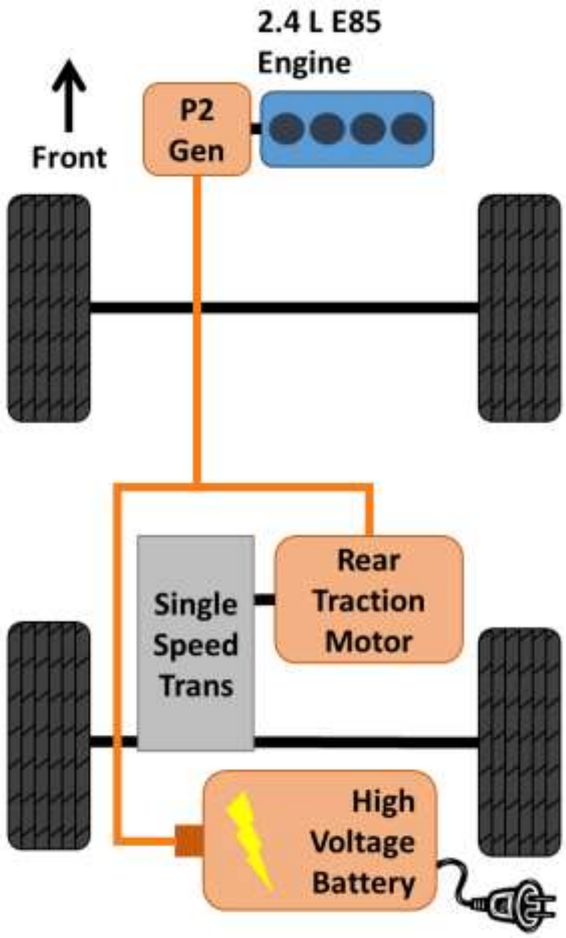

Figure 3-6: Series Extended Range Electric Vehicle Architecture

For a series EREV, a test mass of 2050 kg is representative of the vehicle and two passengers. The weight is much lower than the BEV because of the reduced battery mass. With the addition of the engine, less energy capacity is needed in the batteries to meet the range requirement. Therefore, the battery is downsized to a battery pack with 105 cells in series and 3 in parallel. With the reduction in mass from the BEV, the traction motor is downsized from a UQM 145 kW to a UQM 125 kW while still maintaining performance comparable to the BEV. The model also uses a UQM 75 kW as the motor generator. The overall gear ratio in the series EREV model is 7.17. The drive ratio is chosen to achieve a top speed of 85 mph. The components used in the Autonomie model are shown by Table 3-8.

Table 3-8: Series EREV Component Description

Vehicle Properties	
Test Mass	2050 kg
Energy Storage System	
Energy Capacity	18.8 kWh

Usable Energy Capacity	17.0 kWh
Drive Motor – UQM 125 kW	
Peak Power	125 kW
Peak Torque	300 Nm
Continuous Power	45 kW
Gear Reduction	7.17:1
Top Speed	85 mph
Generator – UQM 75 kW	
Peak Power	75 kW
Continuous Power	41 kW
Peak Torque	240 Nm
Engine – GM LE9	
Peak Power	131 kW
Displacement	2.4 L
Peak Torque	230 Nm

Since the series EREV has a CD mode that uses only electricity and a CS mode that uses E85, both fuel consumption and emissions need to be weighted to calculate overall energy consumption for the vehicle. Using the all-electric range, a utility factor is determined. The utility factor is an estimation of the percentage of time a fleet of certain plug-in vehicles will be operating in EV mode. The utility factor is determined from the vehicle electric range using 2005 National Highway Transportation Survey (NHTS) data on real world driving habits of consumers. Equation 1 shows the use of the utility factor for a series EREV.

$$EC_{total} = EC_{CD} * UF + EC_{CS} * (1 - UF) \qquad \text{Equation 1}$$

where EC_{total} is the total energy consumption, EC_{CD} is the electrical energy consumption in charge depleting mode, UF is the utility factor and EC_{CS} is fuel energy consumption in charge sustaining mode.

For comparison, a B20 series EREV was also analyzed. For both series EREVs, the CD mode was held constant. As shown by Table 3-9, the B20 series EREV consumes less energy than the E85 series EREV. This makes sense, as compression ignited diesel engines are typically more efficient than spark ignited gasoline engines. However, the B20 series EREV consumes nearly three times more WTW petroleum energy than the E85 series EREV.

Table 3-9: Series EREV Performance & Fuel Economy Results

		E85	B20
Performance	0-60 mph acceleration	10.4 sec	
	50-70 mph acceleration	5.7 sec	

	Gradeability at 60 mph	6.58%	
CD Mode	CAFE Energy Consumption	314 AC Wh/mi	
	CAFE EV Range	63 mi	
	Utility Factor	0.758	
	4-Cycle Energy Consumption	424 AC Wh/mi	
	4-Cycle EV Range	49 mi	
	Utility Factor	0.688	
CS Mode	CAFE Fuel Consumption	857 Wh/mi	822 Wh/mi
		39.3 mpgge	41.0 mpgge
	4-Cycle Fuel Consumption	1267 Wh/mi	1085 Wh/mi
		26.6 mpgge	31.1 mpgge
Utility Factor Weighted	CAFE Total Energy Consumption	445 Wh/mi	437 Wh/mi
		74.1 mpgge	75.5 mpgge
	CAFE WTW PEU	74 Wh/mi	179 Wh/mi
	CAFE WTW GHG	208 g/mi	212 g/mi
	4-Cycle Total Energy Consumption	687 Wh/mi	630 Wh/mi
		48.0 mpgge	52.4 mpgge
	4-Cycle WTW PEU	135 Wh/mi	300 Wh/mi
	4-Cycle WTW GHG	292 g/mi	287 g/mi

The advantage of a series hybrid is the ability to decouple the engine from vehicle speed, which allows the engine to run at or near peak efficiency all of the time. Being a series hybrid, all energy comes from the fuel tank and must be converted in a lengthy series path before reaching the wheels. Under city driving conditions, conventional vehicles operate at lower loads where internal combustion engines (ICE) are inefficient. A series hybrid can greatly boost vehicle efficiency for city driving because the engine is decoupled from the wheels and is not constrained to operate at part load. During highway driving, however, conventional vehicles operate much closer to the optimum efficiency ranges of ICE's, hence highway driving is relatively efficient for an engine that directly drives the wheels. A series hybrid on the highway will operate at very near peak efficiency, but will pay efficiency losses at every step of its series path. Thus, the efficiency advantages of a series hybrid are nullified for highway driving compared to a parallel hybrid that can put engine torque directly to the wheels. Another shortcoming of the series hybrid architecture is its inability to use all of its power sources for acceleration. Only the drive motor is mechanically connected to the wheels and the engine cannot directly provide traction. For these two reasons, a parallel path to ground for the engine and generator motor is desirable.

3.3) SERIES PARALLEL PHEV

The third and chosen architecture modeled in Autonomie was a Series-Parallel Plug-In Hybrid Electric Vehicle (PHEV) [Manning, 2012]. This configuration is one that aims to balance all of the targets described in Table 1-1 while also meeting the team specific goals outlined in Table 3-1. To displace PEU and emissions, the two fuels chosen are E85 ethanol and grid electricity. At the vehicle level, grid electricity offers an advantage over other fuels due to incurring no emissions or PEU. In spite of this, an Energy Storage System (ESS) needs to be sized to store energy for continued use which compromises stock cargo capacity. Nevertheless, this is a compromise that the team is making for the benefits of using grid electricity. One of these benefits is the ability to have a fully functional EV system during CD operation. Per the HEVT team goals in Table 3-1, the VT Malibu will be able to operate solely on electricity through the use of the ESS and a Rear Traction Motor (RTM) to deliver torque to the rear axle. This operation will allow for standard performance and also consumer appeal as EV's are very smooth and quiet when operating.

The addition of the 2.4L E85 engine also becomes a benefit for several reasons. Primarily, it extends the range of the vehicle by allowing a charge sustaining operating mode. A CD mode is limited to a range which is designed to optimize the utility of the battery pack based on American daily driving habits. This limited range allows for a conservatively sized ESS which eases packaging, in turn allowing the team to maintain many consumer features. The competition does however have a minimum range requirement which must be met to avoid penalties. The addition of this 2.4L E85 engine offers a solution to this with the combination of electric range and fuel range.

Aside from adding an extended range to the vehicle, the addition of an engine allows for several different operating modes which optimize efficiency in the VT Malibu. The series mode of the PHEV is well suited for optimizing the efficiency of a CS mode during city-like driving situations with many starts and stops and transient speeds. While control of an engine-generator pair can be challenging, there are known solutions. One drawback of series mode is the lossy energy conversions at constant high power conditions similar to highway driving conditions. Another is performance: the vehicle pays the mass penalty for carrying an engine and P2 generator motor (located between the engine and transmission), but those components cannot directly assist the traction motor for acceleration performance or gradeability requirements. The generator must be sized for continuous power requirements for top speed or grade. To alleviate these issues of power requirements for performance and gradeability, the parallel mode is implemented into the VT Malibu. With this mode, an additional path is provided for engine torque to reach the wheels through the use of a 6-speed automatic transmission. Power output is now not solely limited by the RTM or the generator motor, but is limited by the RTM and the engine. This combination allows for great performance but also efficiency during highway driving, considering less powertrain losses are incurred using a direct engine torque path to the wheels. Figure 3-7 shows a powertrain diagram of the Series-Parallel architecture and Table 3-10 includes the specifications for the major powertrain components.

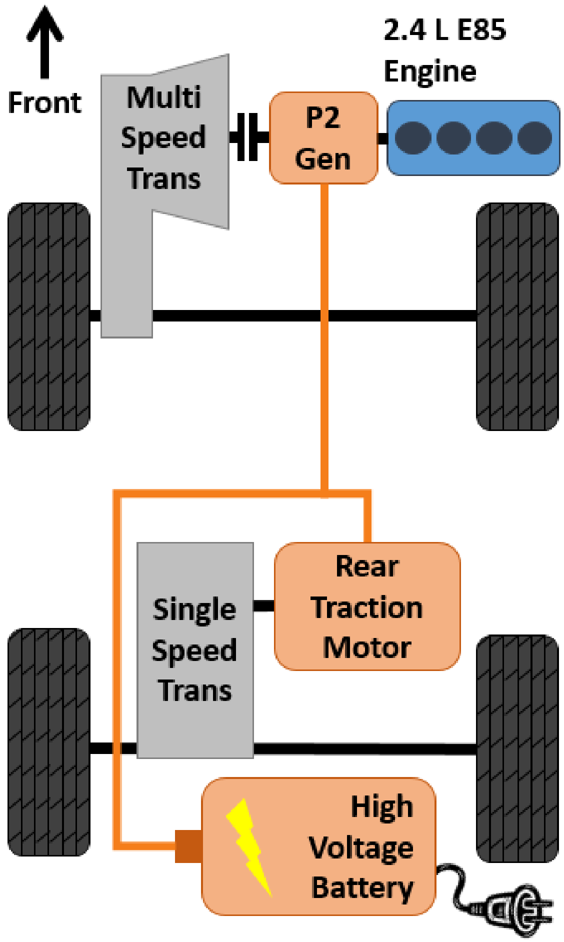

Figure 3-7: Series Parallel Plug-In Hybrid Electric Vehicle Powertrain Architecture.

Table 3-10: Series-Parallel PHEV Component Description

Vehicle Properties	
Vehicle Test Mass	2050 kg
Energy Storage System – A123	
Energy capacity	18.8 kWh
Usable energy capacity	15.0 kWh
Total mass	240 kg
Rear Traction Motor – UQM	
Peak power	125 kW
Continuous torque	45 kW
Peak torque	300 Nm
Gear reduction	7.17:1
Top speed	85 mph
Engine – GM	
Peak power	131 kW
Peak torque	230 Nm

Displacement	2.4 L
Transmission	Auto 6-speed
P2 Generator Motor – Kollmorgen	
Peak DC power	28 kW
Continuous DC power	20 kW
Peak torque	122 Nm

3.3.1) SELECTION MATRIX

Prior to choosing the Series Parallel PHEV, a powertrain design matrix was created, as seen in Table 3-11. The criteria selected to judge each powertrain architecture are split into performance characteristics (weighted 60%) and risk considerations (weighted 40%). The categories and weightings within the performance section are based upon the scored dynamic and static engineering events from vehicle years of the EcoCAR 2 competition. The risk section accounts for packaging and controls challenges as well as team experience and component availability. Engineering value judges the value of the architecture as an engineering endeavor. A BEV, for example, is a relatively simple architecture and is not a novel concept and therefore scores poorly in engineering value. The raw scores used in the matrix come from the Autonomie and CAD modeling performed. Raw scores are normalized and then weighted to find the weighted score. With these scores, the final score of each vehicle was obtained, with the Series Parallel PHEV architecture scoring the highest.

Table 3-11: Vehicle Architecture Selection Matrix

			BEV		**E85 Series EREV**		**B20 Series EREV**		**E85 Series-Parallel PHEV**	
	Objectives	Total Weight	Raw Score	Weighted Score	Raw Score	Weighted Score	Raw Score	Weighted Score	Raw Score	Weighted Score
Performance Considerations: 60%	0-60 Acceleration	2%	9	0.63	10.4	0.28	10.4	0.28	7.5	1.00
	50-70 Acceleration	2%	4.5	1.29	5.7	1.01	5.7	1.01	3.8	1.45
	Drive Quality	2%	1	1.88	0.9	1.69	0.9	1.69	0.8	1.50
	Vehicle Mass	4%	2250	0.00	2050	1.10	2000	1.38	2100	0.83
	Energy Use	8%	432	7.53	685	0.42	552	4.16	606	2.64
	WTW Petroleum Energy Use	9%	14.7	8.49	133.0	5.15	307.0	0.23	105.0	5.94
	WTW GHG Emissions	9%	280.1	3.55	292.0	1.43	284.0	2.85	265.0	6.23
	WTW Criteria Emissions	9%	14.1	7.23	61.0	1.66	23.3	6.14	45.6	3.49
	Consumer Acceptability	10%	0.4	3.94	0.8	7.88	0.7	6.89	1.0	9.84
	EV Range	6%	202	4.55	49	1.10	49	1.10	49	1.10
Risk	Engineering value	6%	0.0	0.00	0.5	3.00	0.5	3.00	1.0	6.00
	Packaging complexity/risk	8%	0.8	1.60	0.2	6.40	0.2	6.40	0.6	3.20
	Controls complexity/risk	8%	0.1	7.20	0.3	5.60	0.5	4.00	0.5	4.00

	Team experience/ facilities	8%	0.7	5.60	0.9	7.20	0.2	1.60	0.9	7.20
	Component availability/ support	10%	0.0	0.00	0.9	9.00	0.7	7.00	0.9	9.00
			Perf:	39.1	Perf:	21.7	Perf:	25.7	Perf:	34.0
			Risk:	14.4	Risk:	31.2	Risk:	22.0	Risk:	29.4
			Total:	53.5	Total:	52.9	Total:	47.7	Total:	63.4

4) CONTROL SYSTEM REQUIREMENTS

In keeping with the V-cycle, Figure 1-2, developing requirements is important and necessary to advance a control system. The process of requirement development is one of the first steps in the control system development process.

4.1) COMPONENT CONTROL REQUIREMENTS

In the development process of a safe and functional control system, a critical step is defining operational requirements for each one of the vehicle components. These requirements are more specifically focused according to the type of operation a component will execute (e.g. diagnostics, communication, starting, operation, etc.). From there, individual requirements are generated for a determined component within a designated category. Table 4-1 provides an example of the developed functionality requirements for the high voltage battery. Other subsystems and components for which this process is followed are detailed in Table 4-2.

Table 4-1: Functionality Requirements for High Voltage Battery

HV Battery	2.1	Communication	2.1.1	The battery control module sends 5 CAN messages including information on battery state, limits, and faults
			2.1.2	The battery control module receives A123_Command from the HVSC
			2.1.3	The battery control module requires an HVIL signal which is HI during normal operation and LO when an E-stop is pressed or certain HV connections are open
			2.1.4	The battery pack includes a second CAN channel which is internal to the pack and can be accessed for extra diagnostics and programming
	2.2	Diagnostics	2.2.1	The battery temperature must remain below 60 deg C
			2.2.2	The battery voltage must remain between 263 V and 378 V
			2.2.3	The battery SOC must not go below 8%
			2.2.4	The battery current must not exceed 250 A
			2.2.5	The HV isolation must remain above 50 ohms/volt
	2.3	Start Up	2.3.1	The HVSC must send the Veh_Wake digital signal to wake up the A123 BCM
			2.3.2	The HVSC must send the A123_Command message
			2.3.3	The HVSC must set the BCM_enable signal to true in the A123_Command message
			2.3.4	The HVSC must set the BCM_mainc_close signal to true in the A123_Command message
			2.3.5	The HVSC must set the BCM_leakage_ena signal to true in the A123_Command message
	2.4	Chargin	2.4.1	The HVSC must measure the proximity sensor signal from the charge port to determine whether the charge cord is plugged in, not plugged in, or plugged in with trigger depressed

			2.4.2	The HVSC must send the ChrgA_Wake digital signal to wake the A123 BCM to initialize charging
			2.4.3	The A123 BCM must send the BRUSA_Command message to enable charging
			2.4.4	The HVSC must disable charging at a max SOC (ex. 98.5%) during normal conditions
			2.4.5	The HVSC must allow the A123 BCM to charge balance when the AllowChargeBalance parameter is set to true by the operator
	2.5	Shutdown	2.5.1	The HVSC must set the BCM_mainc_close signal to false in the A123_Command message
			2.5.2	The HVSC must set the BCM_enable signal to false in the A123_Command message
			2.5.3	The HVSC must stop sending the A123_Command message
			2.5.4	The HVSC must stop sending the Veh_Wake digital signal to wake up the A123 BCM

Table 4-2: Components with Developed Requirements

Components with Developed Requirements
HV Battery (Table 4-1)
RTM
P2 Generator Motor
Engine
Transmission
DC-DC Converter
HV Air Conditioning Compressor
Vehicle & Driver Controllers
Safety Systems (i.e. Emergency Disconnect System, Torque Security, etc.)

Many of the requirements developed for the vehicle components are directly taken from component documentation. For example the HV battery was provided with an Interface Control Document (ICD) which details the communication and interface requirements to operate the components.

These documents, however, were not always comprehensive or readily available. An example of this is the GM provided engine. While there were some specifics provided in GM service manuals and operation documents, many of the specifics were left to the teams to determine. This testing and troubleshooting can be conducted on a test stand and is detailed in a later section (Engine Test Stand, pg. 62).

In a production environment, such as a high capacity automotive manufacturer, the developer has complete control over the interfaces between components. Many of the components used in the VT Malibu, though, are produced and manufactured for a variety of applications. Thus, some of the interfaces may be dependent on application. For example, the P2 inverter has the ability to be controlled either by CAN or by a combination of analog signals to simulate pedal and key inputs. To meet the desired level of control for the VT Malibu, it makes much more sense to control the P2 via CAN. If it were used as a primary traction drive and was directly tied to driver demand, then the analog signals would make the most sense. For reasons like this, control interfaces, and mechanical and electrical interfaces for that matter, can be greatly simplified if components are designed with a specific application in mind.

4.2) SIMULATION PLATFORM & PLANT MODEL REQUIREMENTS

To complete steps 6 through 9 of the V-cycle (Figure 1-2), a plant model must be developed. Sufficient plant models are necessary to accurately test any control algorithms that have been developed throughout the process (i.e. the right side of the V-cycle). A plant is the system, process, or device the user intend to control. In this context, the plant describes the VT Malibu and the powertrain components used. Without an operating and accurate model, validation activities are unable to be completed and sometimes must be skipped. Also important to note is that different simulation platforms and plant models are required during different steps of the EcoCAR 2 vehicle development process. These differences are discussed in detail in the following sections.

4.2.1) POWERTRAIN ARCHITECTURE SELECTION

At the early stages of powertrain architecture selection, many different powertrain architectures and designs must be analyzed and compared without fully knowing all of the integration details, as previously discussed (Powertrain Selection, pg. 16). For example, control algorithms cannot accurately be developed without knowing the full architecture and it is unrealistic to develop control algorithms for every considered powertrain design. In addition, it is often necessary to use or develop very general component plant models to allow for flexibility in final component scaling and selection. Typically components are chosen, designed, and built after the powertrain architecture so the final specifications are often unknown at the time of powertrain architecture selection. Lastly, the simulation platform must be able to accurately test all of the vehicle powertrain configurations on identical tests so results can easily be compared. These requirements are summarized in Table 4-3.

Table 4-3: Simulation Platform & Model Requirements for Architecture Selection

Requirement
Large variety of powertrain configurations
General plant models since final component specifications are unknown
Ability to simulate operation with little/no control algorithms
Ability to run standard tests/compare results between powertrain variations

To meet the requirements listed in Table 4-3, HEVT selected the software suite Autonomie because it allows for variation in many parameters and performance tests for a wide variety of powertrain architectures. Autonomie is designed to be used as a single tool throughout the different phases of Model Based Design (MBD) of the VDP. While Autonomie includes capabilities for four key elements in the development process: modeling a plant (from first principles or system identification), synthesizing and analyzing a controller for the plant, simulating the plant and controller together, and programming/deploying the controller [UChicago], HEVT uses Autonomie primarily for the plant model portion of system identification.

4.2.1.1) MODEL LIMITATIONS AND SENSITIVITIES

The Autonomie modeling software takes into account many vehicle parameters to produce an accurate model. However, this model cannot account for all factors that affect vehicle performance and fuel economy. Fuel economy and performance results are sensitive to changes in engine power and vehicle mass as shown in the previous sections. As engine power and vehicle mass decrease, the overall fuel efficiency of the vehicle increases. The fuel efficiency of the model is much more sensitive to changes in mass than to changes in engine size, within limits. Another important factor to consider that is not addressed in Autonomie is that some engines can run with a variety of fuels. This is the case for Flex Fuel engines that can run on varying mixes of gasoline and ethanol from 0-85% ethanol. The most commonly available mixtures of these fuels are E10 and E85 which are 10% and 85% ethanol by volume, respectively. The effect of different blends of gasoline and ethanol has a great impact on fuel economy because the fuels have distinctly different energy density properties. These variations in fuel composition also affect the GHG emissions and WTW petroleum use. The Autonomie model only approximates cold start conditions and does not model criteria exhaust emissions from the vehicle.

Another factor that Autonomie attempts to model is the driving habits of different human operators. Autonomie offers five different driver models based on different needs (for example normal driver and race driver). These estimations cannot accurately predict the driving characteristics of the average driver. While not every external impact on fuel economy can be accounted for, Autonomie has a feature in the model setup that allows it to account for certain environmental factors. These factors include the density and humidity of the air as well as the ambient temperature and pressure. The ambient temperature and pressure conditions can be easily modified from the model setup. Some factors that are not modeled are weather, such as snow or rain, and road conditions as they vary too much to model accurately. In the provided Chevy Malibu model, mechanical and electrical accessory loads were assumed to be 0 W and 200 W respectively. These values do not account for use of air conditioning in summer which would substantially increase the electrical load. These factors would also have a substantial effect on fuel economy.

4.2.2) CONTROL ALGORITHM TESTING

The majority of the EcoCAR 2 competition is focused around the development of a control algorithm to manage the components in the vehicle. Rapid Control Prototyping provides a great process and platform for accomplishing this development and works well with the previously discussed V-cycle (Figure 1-2). Broadly, RCP has the following stages (Figure 4-1):

- Development of system plant model
- Development of the control strategy
- Testing of the control strategy against the plant model
- Using the validated control algorithm for real-time testing

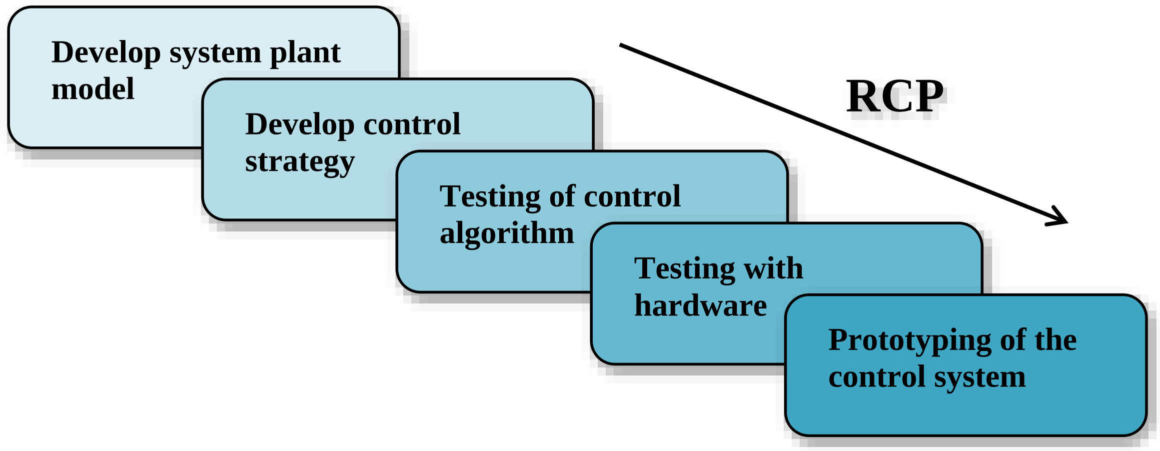

Figure 4-1: Rapid Control Prototyping Process

RCP uses high performance hardware and software. This method serves as both a way to acquire data from the model and a way to control it. By following the RCP process, multiple iterations of building the physical controller are avoided. This saves time and money that would have been spent on changing the interfaces of the plant to interact with the control system every time a small change is made in either of them. The new prototype of the control system is tested using the same hardware as the one used to simulate and validate the plant with the control strategy. The existing control software in the plant stays the same. RCP helps achieve the transition from a pre-processing stage like desktop simulation to real-time simulation automatically. Manually coding the system thus is avoided and significant time is saved along with many errors. RCP avoids this due to the automation of transition. This provides more time for iterations due to changes in control system or plant. There are associated costs with high performance hardware and software. However, looking over the larger picture, the cost and time saving accomplished due to the use of the RCP process is much more than the cost of purchasing the hardware or software.

Depending on the stage of RCP there are various tools available commercially. Plant modelling can be done using software like LMS Imagine- AMESim, Mechanical Simulation Corporation’s CarSim, Dassault Systemes’ Dymola, or Mathworks Simulink, among others. The control strategy can be developed and implemented using tools like Simulink or LabView. Simulink uses more of a block diagram approach and is the software of choice for HEVT during EcoCAR 2. The hardware used can also be varied. The usual method of handling this is to interface a computer (having the plant model and the control strategy) with the control system (Engine Control Unit or Transmission Control Unit), and the data acquisition and data feeding hardware. Commonly used prototype controllers in the automotive industry include dSPACE, National Instruments, and ETAS products.

Software-In-the-Loop (SIL) and Hardware-In-the-Loop (HIL) are two simulation platforms available for verifying and validating control code. SIL and HIL are run against the plant model. Prototyping, or in-vehicle testing in the case of HEVT, is validation in an operational physical system. The aims of a development process involving HIL and SIL simulations are as follows [Pfau]:

- Optimum controller functionality
- Good function validation, ensured by great test depth
- Fast response to technical modifications by means of model-based function development and parameterization
- Maximum added value by systematically automating time-consuming development tasks

SIL and HIL can refer to many different types of so-called 'loops.' One common occurrence is engine-in-the-loop where an engine control algorithm is validated on a physical engine. Since the control system being developed by HEVT is the Hybrid Vehicle Supervisory Controller, or HVSC, controller SIL and controller HIL will be mentioned from this point on to avoid confusion. Block diagrams depicting the hardware and communication of SIL and HIL are shown in Figure 4-2 and Figure 4-3, respectively. Requirements for both of these platforms are described in the following sections.

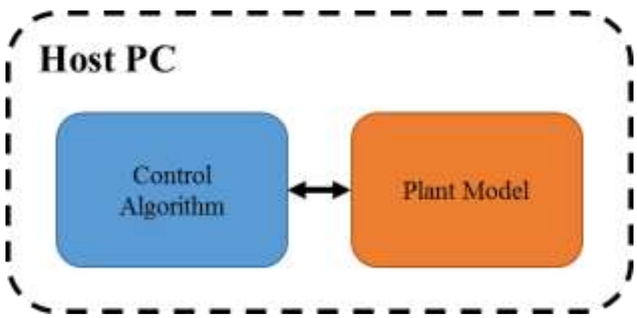

Figure 4-2: SIL Functional Diagram

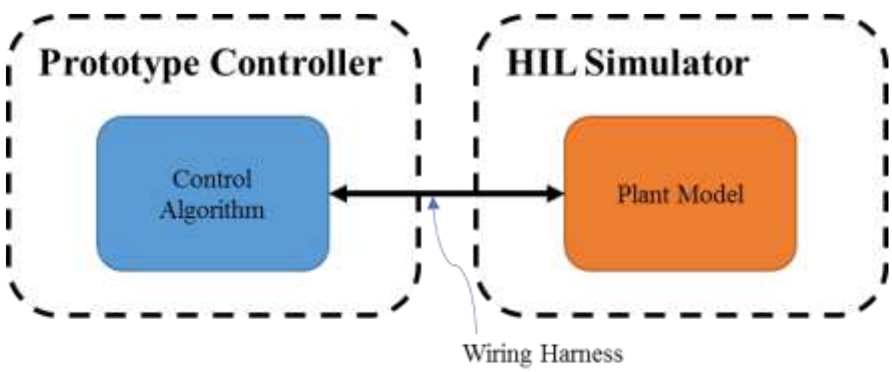

Figure 4-3: HIL Functional Diagram

4.2.2.1) *SOFTWARE-IN-THE-LOOP*

Controller SIL is generally one of the first forms of validation used on control systems during the RCP process that HEVT is implementing. In SIL, the control algorithms and plant model are combined and an offline simulation is performed (Figure 4-2). The generated code is compiled with a host compiler and executed on the host PC [Beine]. Software in the loop can commonly be computed in a simulation time much quicker than it would take if run in real time. This can result in huge cuts in development time by bringing errors out quicker than traditional testing. In addition, the algorithms are the target of SIL testing because the interface between software and plant is greatly simplified.

Since the final powertrain has already been selected, only one powertrain architecture needs to be analyzed. During these simulations, it is necessary to frequently change the control algorithms and compare the results between different algorithms. Additionally, it is necessary to be able to change the component models as new developments arise or differences are noticed between the simulation environment and the physical components. As in the powertrain architecture selection simulation environment, the simulation platform must be able to run repeatable and identical tests so results can easily be compared between control algorithms. Lastly, one of the huge benefits of SIL testing is the ability to run simulations faster than real time, so the platform and model must be able to meet this requirement. These requirements are summarized in Table 4-4.

Table 4-4: Simulation Platform & Plant Model Requirements for Controller SIL

Requirement
Sufficiently model a single powertrain configuration
Detailed plant models match component specifications
Ability to merge with control algorithms on host PC
Run standard tests/compare results between control algorithm variations
Simulation speeds faster than real time

HEVT is using the traditional MATLAB and Simulink environment to accomplish SIL testing. Simulink allows simulations to be run at varying speeds and using a wide variety of solvers. The discrete step size can also be modified which can aid in improving the simulation speeds. Simulink also provides a convenient way to merge with control algorithms which are also produced in Simulink. Model references and library blocks provide interfaces between different Simulink files. Proper organization of the control code algorithms allows certain sections to be used within a simulation. MATLAB also offers a suitable means for post processing the data from simulations. Access to internal state variables of both the model and algorithms is very easy through Simulink. Additionally, Autonomie is based in MATLAB and Simulink and it provides a good baseline for importing models as a starting point.

4.2.2.2) *HARDWARE-IN-THE-LOOP*

Hardware-in-the-loop, on the other hand, turns the focus towards physical interfaces. One of the considerations for determining the required HIL system is adequate input/output counts [Wahlstrom]. This is because controller HIL requires the developed software to be compiled and loaded onto the physical supervisory controller, in the case of HEVT, a dSPACE MicroAutoBox II (MAB). The HIL simulator system must then replicate the physical I/O that the vehicle might provide when interfacing with the control code. Thus, HIL serves to test I/O functionality as its primary goal. Figure 4-3 shows a standard controller HIL setup. The addition of physical signals introduces possible latency issues. The developed software must be accepting of these differences to be effective. Lastly, when transitioning to actual hardware, simulations must now be conducted in real time.

Like SIL, only one powertrain architecture need be analyzed but it is necessary to frequently change the control algorithms by loading different versions onto the prototype controller and compare the results between different algorithms. Again, the simulation platform must be able to run repeatable and identical tests so results can easily be compared between control algorithms. Since the control algorithms are being run on the prototype controller and physical interfaces are included in HIL testing, the simulation must be run in real time. This also helps to validate that the dynamics of the control algorithms are working properly. These requirements are summarized in Table 4-5.

Table 4-5: Simulation Platform & Plant Model Requirements for Controller HIL

Requirement
Sufficiently model a single powertrain configuration
Detailed plant models match component specifications
Sufficient I/O to interface with prototype controller
Run standard tests/compare results between control algorithm variations
Real time processing interface
Access to state variables and logging features

HEVT is using a dSPACE Simulator Mid-Size with a DS1006 processor board. This simulator has ample input and output capabilities to meet the interface requirements for sensor measurement and actuator control for the HVSC under test. This simulator also has fault insertion capabilities that physically alter the signals that are output from the plant model to the controller under test. This capability can help to develop software with robust diagnostics to recognize faults before in-vehicle implementation. dSPACE also provides a Simulink block set to easily configure the I/O of the simulator to interface with an existing Simulink model which makes it very convenient to use in conjunction with the selected SIL platform. Again, the use of Simulink library blocks enables quick switching between SIL and HIL testing but still using the same plant model. Lastly, additional dSPACE software provides a convenient customizable interface to the application running on the simulator with direct access to the CAN bus system, internal state variables, and failure insertion unit. Some of the simulator specifications are included in Table 4-6.

Table 4-6: dSPACE Simulator Mid-Size Specifications Summary

Parameter	Value
Form Factor	19” desktop rack
Signal Conditioning	Yes
Load Simulation	Available
Failure Insertion	Available
Power Supply	16 A maximum
I/O	CAN, Analog, Digital, PWM

4.2.3) CONTROL OPTIMIZATION

Referring back to the V-diagram in Figure 1-2, control optimization and calibration is one of the final steps in the control system development process. Control optimization is the process of refining the parameters used in the control algorithms to find the "best available" values of some objective function (i.e. performance or emissions function) given a defined set of constraints. Since the control algorithms have already been validated by this point in the process, it is often sufficient to calibrate the control algorithms by testing on the physical system, in this case the VT Malibu. However, sometimes other conditions prohibit this from happening. Some conditions that could prevent optimization include lack of access to resources and/or the physical system. On the VT Malibu, this situation could be the case if the vehicle is being serviced or is un-drivable. Another condition would be the inability to create the necessary test conditions or recreate the same test conditions for different parameter sets. Some parameterization my come into effect only in emergency or safety critical situations. Also, when testing out different parameter sets, it is nearly impossible for the driver to recreate the exact same test conditions (vehicle speed, battery state of charge (SOC), time delays, etc.) as the initial test. Thus, it is often more efficient & effective to perform control optimization in a simulation environment provided that the simulation plant sufficiently models the physical plant for the tests being provided.

To effectively perform control optimization, however, there are some additional requirements for the simulation environment. It is no longer sufficient to model the components based on specifications. The plant model must match the physical plant in all aspects related to the test or else the control system won’t be optimized for the system it is intended to control. Additionally, optimization often requires simulations running for many different parameter sets or configurations and it is convenient to be able to automate this process. These requirements are summarized in Table 4-7.

Table 4-7: Simulation Platform & Plant Model Requirements for Control Optimization

Requirement
Detailed plant models match physical plant
Run standard tests/compare results between control algorithm variations
Automated testing preferred

Since both the chosen SIL and HIL platforms of MATLAB Simulink and the dSPACE Simulator Mid-Size meet all of these requirements, it is most efficient to use these platforms for control optimization as well. The one requirement that causes the most issues for the team is the fidelity of the models. However, efforts have been focused on model improvement during Year 3 for this purpose. Since the team only has one prototype controller, it makes more sense to use SIL for the control optimization as it removes the need to separate the controller and vehicle and also has the ability to produce results much quicker.

5) CONTROL CODE DEVELOPMENT

Once requirements are completed, the known interfaces can be combined into an algorithm which will monitor, manage, and command all of the vehicle components in a passive control strategy. This passive control strategy only uses a few inputs from the driver including key state, gear shift selector, accelerator pedal, and brake pedal, much like in a conventional vehicle. Any additional user controls can be distracting to the driver. Based on the driver inputs, the control algorithm must then determine the component operating conditions to ensure safety of the vehicle and driver, meet driver demand, and then do so efficiently.

5.1) CODE STRUCTURE

Throughout the EcoCAR 2 competition, a rigorous design process is used to develop a control algorithm composed of the subsystems presented in Table 5-1. The iterative development process starts with the definition of the control system requirements (Component Control Requirements, pg. 26) and moves through different stages of validation and refinement until successful in-vehicle implementation of the control system is achieved.

Table 5-1: Control Algorithm Subsystems

Subsystem	Function
Vehicle I/O	gateway for all communication signals between the control strategy and physical signals on the vehicle
Diagnostics	determine if faults have occurred and what action to take to mitigation faults and evaluate component limits
Selection	determine most efficient and effective operating point for the vehicle
Execution	determine appropriate signals to send to the individual component controllers to achieve desired vehicle operation

The control algorithm is in constant communication with the vehicle in a closed loop process. The structure of the control platform provides the team with an effective way to identify faulted components and then take action to continue safe vehicle operation and implement fault mitigation. Additionally, this control strategy has the goal of making the vehicle operate in the manner that most efficiently meets driver demand. Following, the structure of the code is explained in detail. Figure 5-1 shows a block diagram of the control algorithm and shows the grouping of the different subsystems.

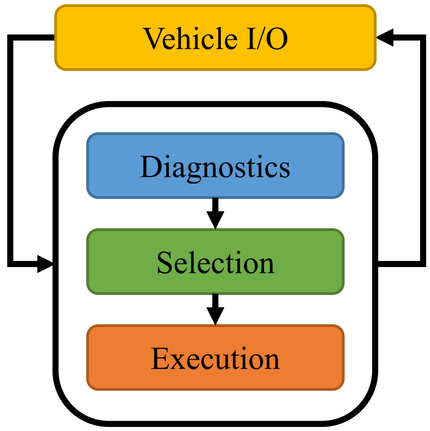

Figure 5-1: Control Algorithm Block Diagram

The inputs and outputs are used by all three of the other controller subsystems and this subsystem is fully dependent on the controller hardware being selected. The physical input and output channels and ports are different for individual controllers and this subsystem would allow for the same algorithm to be used in different platforms. The other three subsystems then execute in order and are described in further detail in the following sections.

5.2) DIAGNOSTICS

The first portion of the control algorithm to execute is the diagnostic strategy. The algorithm receives driver inputs from the car, then determines and evaluates the status of components against their operating limits. This strategy also defines powertrain capabilities that the following sections of the algorithm use as constraints and is therefore very important for the safe operation of the vehicle. The diagnosing process is performed for all major components in the vehicle with the purpose of preventing or limiting operation when faults of any nature (e.g., temperature fault, communication fault, etc.) occur. If no faults occur in any of the components in the diagnostics block, then components are "allowed" to operate and corresponding signals are sent to the selection and execution portions of the algorithm to determine the operation of the vehicle and the individual components. However, when faults are detected, the faulted component is limited or not allowed to operate, and the selection and execution blocks work to determine a different way to safely operate the vehicle and meet driver demand. The issues checked by the diagnostic strategy include safety critical systems as well as general component operational problems. Table 5-2 includes a list of all of the diagnostic checks conducted by the control algorithm as well as the specific conditions and the corresponding remedial actions. If the required condition is not met, then the remedial action is enforced.

Table 5-2: Diagnostic Checks & Remedial Actions

Diagnostic	Required Condition	Remedial Action
APP 1 Voltage In Range	$0.46\ V \leq APP\ 1 \leq 4.7\ V$	Calculated APP = 0
APP 2 Voltage In Range	$0.32\ V \leq APP\ 2 \leq 2.6\ V$	

APP1 & APP 2 Agree	$1.9 \leq \frac{APP\ 1}{APP\ 2} \leq 2.1$	
ECM APP Verification	$0.9 \leq \frac{APP\ to\ ECM}{APP\ reported\ by\ ECM} \leq 1.1$	Disable engine
RTM CAN	*Expected CAN messages are received at expected periodic rates* *Faults trip when a message has not been received in 3x its expected periodic rate*	Disable respective component
Engine CAN		
Transmission CAN		
Body Control Module CAN		
P2 CAN		
Battery CAN		
DC-DC CAN		
P2 Speed	$P2\ Speed < 5000\ RPM$	P2 Torque Command = 0, Engine Torque Command = 0
RTM Speed	$RTM\ Speed < 8000\ RPM$	RTM Torque Command = 0
Battery Contactors	$Contactor\ Status = Contactor\ Command$	Disable HV battery
Ground Fault Detection	$\frac{Isolation\ Resistance}{Battery\ Voltage} > 500\ \frac{\Omega}{V}$	Illuminate indicator
Battery SOC	$SOC > 10\%$	Disable HV battery
P2 Temperature	$P2\ Core\ Temp < 100°C$	Disable P2
Engine Temperature	$Engine\ Coolant\ Temp < 110°C$	Disable engine
Battery Temperature	$Max\ Cell\ Temp < 60°C$	Disable HV battery
DC-DC Temperature	$Heat\ Plate\ Thermister < 75°C$	Disable DC-DC
Power Electronics Coolant Temperature	$Coolant\ Temp\ Sensor < 45°C$	Turn on radiator fans, Limit motor torques
	$Coolant\ Temp\ Sensor < 65°C$	Disable power electronics
RTM Temperature	$OverRotorTemp = false$ $OverStatorTemp = false$ $OverInverterTemp = false$	Disable RTM
Antilock Brakes	*Antilock brakes are inactive*	Disable regen braking

Most of these diagnostics are derived directly from the Component Control Requirements (pg. 26). For example, the ground fault detection diagnostic comes directly from requirement 2.2.5 in Table 4-1. Others are additional checks to make sure that commands are being followed, such as the battery contactor diagnostic.

While the developed control algorithm cannot compete with the level of diagnostics on production vehicle, many of the ideas are derived from production systems. In a modern production car, the controllers run checks on almost every input signal and try to detect whether those signals are valid. There are inspections to determine whether those signals are shorted to ground, 5V (the supply voltage of many sensors), or 12V (the primary power level in most vehicles) or considered an open circuit (indicative of a broken wire). The VT algorithm aims to do this as well. While the majority of the commands are received or sent via the Controller Area Network (CAN) in the VT Malibu, there are still some analog and digital signals used by the HVSC. The APP diagnostics in the VT algorithm can detect a short to ground or 5V because the expected voltages leave buffers on both ends of the spectrum. Figure 5-2 shows the percentage as a function of the voltage for both the APP1 and APP2 signals. Values outside of the expected range (detailed in Table 5-2) result in invalid calculated values. The actual accelerator pedal is comprised of two linked potentiometers. Both potentiometers include an offset resistance and are designed to have a ratio of 2:1.

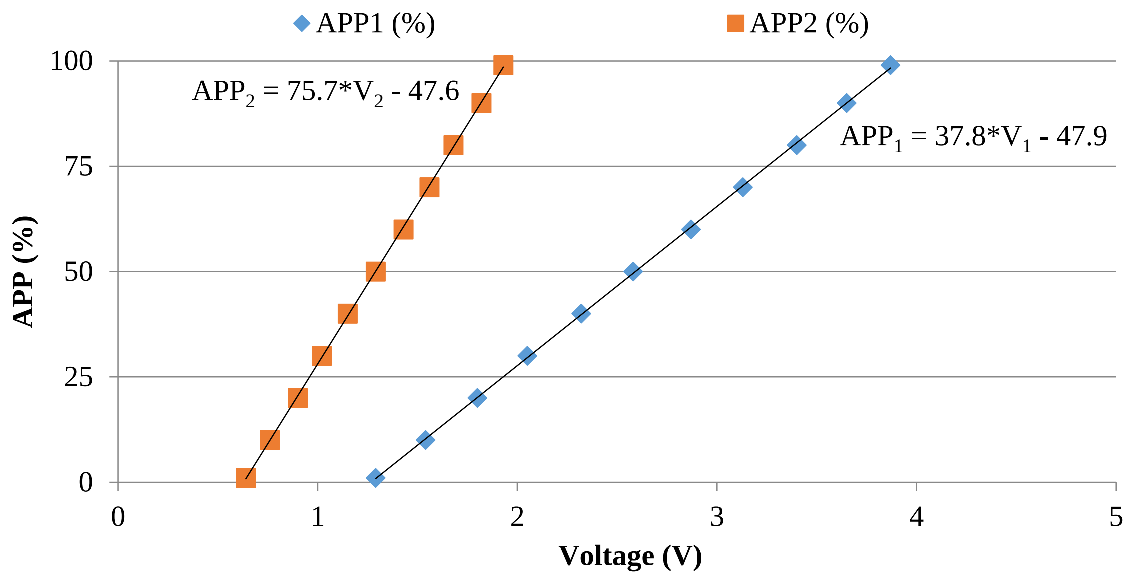

Figure 5-2: Accelerator Pedal Percentage Calculator

In addition to performing diagnostics, viewing the faults is crucial to being able to troubleshoot the vehicles operation. Without a way to determine what faults were triggered and what caused those, the developer is very limited in efforts to correct those conditions and make the vehicle more robust. Information for all of the diagnostic checks is stored in an array in the VT algorithm, including the time the fault was last triggered, the time the fault was last recovered, whether a recovery/reset is allowed (depending on the severity of the fault), and additional fault dependent data to help in determining the root cause.

5.3) SELECTION

After the diagnostics subsystem completes, the next section of the control software is the selection strategy. This subsystem is where the majority of the control algorithm is run. The selection strategy developed has two states which describe the high level operation of the vehicle: charge depleting, and charge sustaining. The selection algorithm determines if a CD or a CS control strategy should be used, in addition to the CS mode of operation (Series mode or Parallel mode) that best meets driver demand. The driver demand is based on the status of components from the previous subsystem and the driver inputs. This information is used to determine the control strategy and mode of operation. For instance, in both CD and CS Series modes of operation, the RTM is the only source of torque to the wheels, so this section calculates the driver requested torque from the vehicle speed and accelerator pedal position. Additionally, parameters of operation for other modes of operation are determined in the selection stage (e.g., P2 generator motor torque for CS mode, RTM regenerative braking torque). This stage is perhaps the most important in the code because it determines high level commands for operating the vehicle powertrain components.

A special addition to the selection portion of the control strategy during Year 3 was the parameterization of regenerative braking by the RTM as shown in Figure 5-3. The developer is able to freely define variables such as starting regenerative braking speed, full regenerative braking speed, maximum regenerative braking torque, and maximum power output. By changing one or more of these parameters, the regenerative braking characteristics can be controlled for any given vehicle speed.

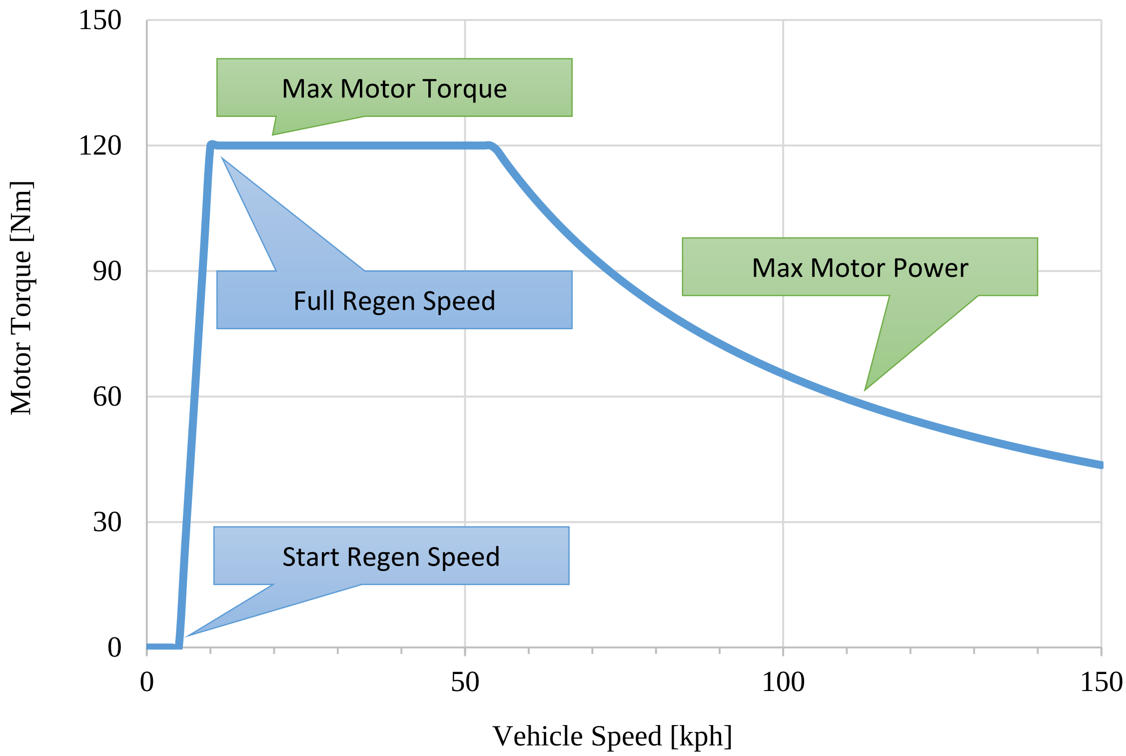

Figure 5-3: Regenerative Braking Parameterization

Transitions between CD and CS states are mainly determined based on battery SOC. The CD state is the primary state of operation and will consist of all-electric driving through the RTM and rear axle. The vehicle will operate in the CD state until a transition battery SOC has been reached. This transition SOC is chosen after careful evaluation of the battery pack provided by A123 to ensure the vehicle is able to meet minimum performance requirements all the way down to the transition point and during CS mode. It is also driven by the confidence in the ability of the vehicle to adequately sustain battery charge during CS mode. The chosen transition SOC is 30%. Once the transition SOC is hit, the control strategy enters the CS state. During CS operation, all net energy comes from fuel through the engine and the battery pack will be used as an energy storage buffer. The battery SOC is maintained at the transition SOC and within a small band of allowable SOC values (+/-3%). In the CS state, the control strategy makes use of the RTM, engine and P2 generator motor to meet the driver demand while minimizing fuel consumption.

In addition to the primary operating modes, the HVSC also includes an operating mode for meeting high driver demands. When a high power is demanded by the driver for a sustained period of time, the controller recognizes that the powertrain components in CD mode are not capable of sustaining operation at certain power levels. One example of this situation is driven by the gradeability requirement for the competition. The vehicle is required to cruise at a speed of 60 mph up a grade of 3.5% for 20 minutes. This requires a DC bus power demand of ~45 kW. At this rate, the battery would transition to CS mode in 17 minutes. In CS mode, however, the generator and engine are only able to output 20 kW continuously and is not able to sustain charge. Therefore, to combat situations like this, a state which recognizes high power demand is triggered when the power demand exceeds a time averaged limit. During the high power demand mode, the engine and generator are turned on, but the SOC is still expected to deplete because the generator cannot meet the power demand of the vehicle.

While the state describes the high level operation of the vehicle, modes are needed to describe the coupling of torque sources. Mode 1 is defined as the transmission is in neutral. This means that the only way to propel the vehicle is through the RTM. Mode 2 dictates that the transmission is in gear, allowing for the engine and P2 generator motor to propel the vehicle through the front axle in parallel with propulsion on the rear axle from the RTM. In each of the two modes the engine can be on or off. Table 5-3 summarizes the four fundamental ways the powertrain can operate, combining the modes with the engine state.

Table 5-3: Combinations of Modes & Engine States

	Engine Off	**Engine On**
Mode 1 (Trans in neutral)	Electric Driving (M1Eoff)	Series Hybrid Operation (M1Eon)
Mode 2 (Trans in gear)	P2 Performance Assist (M2Eoff)	Parallel Hybrid Operation (M2Eon)

Mode 1 Engine Off (M1Eoff) is the primary operation mode during CD mode with only the RTM providing power. Mode 2 Engine Off (M2Eoff) can be used for performance assist in CD mode with both motors providing power. However, high losses are incurred by spinning the engine unfueled since the engine is directly coupled to the P2 generator motor, so this mode is only be used for performance considerations and could be a point of future development for the vehicle. For this reason M2Eoff is not considered for the control strategy. CS operation could make use of any of the four operating modes depending on the current conditions. Mode 1 Engine On (M1Eon) is series hybrid operation with the engine driving the P2 generator motor to supply electrical power to the rear powertrain and manage battery SOC. Mode 2 Engine On (M2Eon) allows parallel operation in which the engine-generator assembly gives torque to the front axle through the multi-speed transmission. Note that in this mode, the RTM simultaneously propels the rear axle, and under certain situations the P2 generator can provide energy to the battery pack or the RTM.

To test vehicle modes of operation, HEVT developed and implemented a front powertrain test stand to gather data and signals from each powertrain element and study the behavior of the components operating together. From this experimental setup, HEVT found that the E85 engine was operating properly when coupled with the P2 generator motor. However, the transmission was never engaged due to an oil leak from the transmission into the P2 generator motor when testing was performed. This test was performed by mounting the P2 generator motor to the transmission on a test stand. The P2 engages both the transmission input shaft and the oil pump. The P2 is then commanded and run at 1000 rpm. After 6 seconds, oil can be seen leaking profusely out of the air gap and bolt holes.

Thus from a mechanical perspective the inability to stop the oil leak from the multi-speed transmission into the P2 generator motor prevented HEVT from being able to implement parallel mode of operation. Even though parallel mode is not operational in the vehicle, initial control algorithms have still been developed and are presented in this paper.

The four modes of operation introduce several degrees of freedom (DOF) in choosing how to meet the driver demand. The outputs of the system are the axle rotational speed, N_o, and the axle torque, T_o. N_o is fixed by the current vehicle speed and T_o is determined from the accelerator pedal position and vehicle speed to infer the driver requested axle torque. As previously mentioned, the only way to meet the driver demand in CD mode is using the RTM. In CS mode however, the control strategy can chose to operate in M1Eoff or series mode (M1Eon). Furthermore, there are many operating points within series mode that the control strategy can choose. The choice, then, is developing an online optimization strategy for minimizing fuel consumption and emissions while meeting driver demand. The strategy works by searching the entire range of possible operating points in each mode and determining which one yields the best efficiency at the current operating condition. In addition to best efficiency, the control algorithm takes into account factors such as drivability, frequency of engine starts and stops, and battery state of charge, among others. These factors are discussed in more detail in later sections.

The first step in development of the strategy is to identify the DOF in each mode. In M1Eoff (EV mode) there are zero DOF as there is only one way to meet the driver demand. M1Eon (series mode), however, introduces two DOF into the system. While there is still only one way to meet the driver demand using the RTM, the genset (engine and P2 generator motor system) can be operated at any torque and any speed within the component limits. This 2 DOF operating range can be reduced to a 1 DOF range by recognizing that many combinations of genset torque and speed can produce the same output electrical power, but there is only one combination which will maximize efficiency for a given power level. Even though parallel mode is not possible for operation, the DOF's and powertrain mapping will be presented in the following subsection. This mode has a discrete speed degree of freedom in that there are 6 gears to choose from that each constrain the engine and P2 generator motor speed. Therefore each gear can be looked at separately. Each gear is a 2 DOF with different combinations of engine torque (T_e), RTM torque (T_M) and P2 generator torque (T_G) that could be used to achieve the driver demand while charging or discharging the battery.

5.3.1) STATIC POWERTRAIN MAPPING

With the degrees of freedom for each mode identified, a method is then developed to compare operating points with each other across different modes in order to choose the best operating point. The strategy HEVT has developed calculates instantaneous powertrain efficiency for a given operating point. This powertrain efficiency is defined as tractive power out divided by fuel power in. Additionally, an important consideration for calculating powertrain efficiency for a given operating point is how to reflect extra electrical power being generated and stored in the battery or power being discharged from the battery to meet the power demand of the motors [King]. In both efficiency equations that follow, the formulas take the ratio of tractive power out to input fuel power. Equation 2 introduces a method for accounting for power going into the battery:

$$\eta_{PT,chg} = \frac{T_o N_o + P_{Batt}\eta_{Chg}\eta_{Dischg}\eta_{Tr}}{P_{fuel}} \qquad \text{Equation 2}$$

The "$T_o N_o$" term in the numerator represents the tractive power out. The second term in the numerator accounts for the extra battery power by calculating the theoretical future tractive power out that this battery energy will eventually go on to make. This is calculated by multiplying the battery power with the charging efficiency that will be observed at the current time, then multiplying by the average drive cycle discharge efficiency and the average drive cycle electric powertrain efficiency. These average drive cycle efficiencies are used because it is difficult to track at what power level and therefore what efficiency the stored energy will eventually be used. For the case of energy being discharged from the battery Equation 3 is used:

$$\eta_{PT,dch} = \frac{T_o N_o}{P_{fuel} + \frac{P_{Batt}}{\eta_{Dischg}\eta_{Chg}\eta_{genset}}} \qquad \text{Equation 3}$$

For the discharge case, power being discharged from the battery is accounted for by assuming that battery energy must have been originally stored in the battery from burning fuel. The battery power at the terminals is therefore divided by the battery discharge efficiency and then by a drive cycle average battery charging efficiency and a drive cycle average marginal genset efficiency. These four drive cycle average efficiencies become tunable parameters that affect the behavior of the strategy by changing how much charging and discharging the battery is favored for increasing engine efficiency.

However, for this strategy to be useful in a dynamic control strategy, additional factors must be considered. This strategy, for instance, would have no direct control over battery state of charge. On drive cycles with low load, such as UDDS, the strategy would favor charging the battery often to increase the efficiency causing battery energy to be accumulated and never used. On higher loading drive cycles, battery discharge would be favored and excess battery energy would be used without being replenished. To account for SOC control, an SOC cost penalty is introduced. This cost would be subtracted from the numerator of the efficiency equations to arbitrarily lower the theoretical efficiency of an operating point that is undesirable from an SOC standpoint. As an example, if the SOC is at the CS target, then the cost factor would be zero. Alternatively, if the SOC is above the target SOC, for instance, the cost factor would be increased for battery charging and made negative for battery discharging. This cost factor would then be multiplied by the battery power for a given operating point to calculate the penalty for that operating point. Hence, discharging efficiency would be artificially inflated and favored by the strategy.

One current consideration other than powertrain efficiency and battery state of charge includes the frequency of engine starts and stops. The algorithm currently limits these frequencies by requiring the engine to be on or off for minimum time values. This part of the algorithm prevents the controller from continuously switching the engine on or off based on the powertrain efficiency equations described above. Limiting engine starts can also cut back on emissions and fuel consumption, both of which are much higher during an engine start than during steady operation.

Future work will continue to refine this strategy, mainly for CD and CS series mode, by accounting for more dynamic costs. Some of these costs will include diagnostics considerations driven by the limitations reported by the diagnostic section of the control software. For instance, cost penalties will be assigned for high RTM power if the RTM temperature is getting too high. Cost can also be assigned for operating the battery pack at current levels that are too high relative to the present battery current limits.

5.4) EXECUTION

Following the diagnostics and selection of an operating strategy, the control algorithm must send specific command signals to individual components in the vehicle to comply with the mode of operation selected. In the third stage of the code, transitions between modes of operation occur and the logic of operation for each major component is also performed. The selection portion performs high level control by selecting a vehicle operating strategy, whereas the execution portion performs low level control by commanding each component on how to operate to achieve the vehicle performance that will best meet driver demand while also handling transients between modes. For example, the selection algorithm would simply decide whether or not to close the battery contactors. The execution algorithm, on the other hand would handle the interpretation of that state into signals that the battery pack is expecting. Figure 5-4 shows the process for closing contactors and shows how it is dependent on the command from the selection algorithm and other controller inputs and outputs. Table 5-4 details all of the required output signals for the primary powertrain components that are generated in the Execution subsystem.

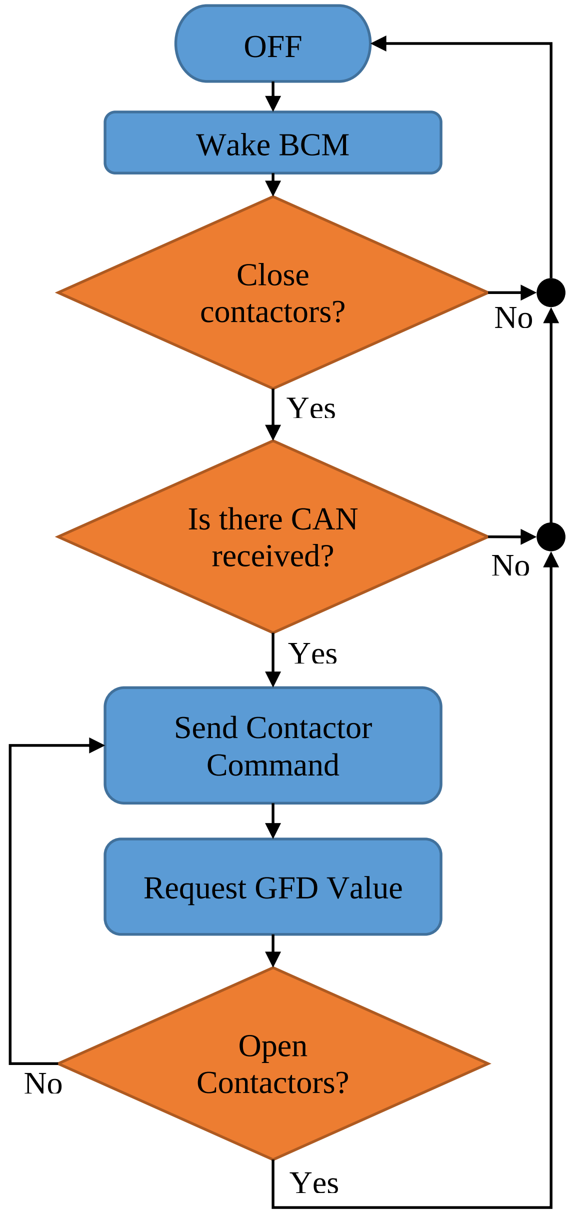

Figure 5-4: Abridged Battery Contactor Execution Process

Table 5-4: Required HVSC Output Signals

Battery Control Module – A123		
Battery Normal Wake		Digital
Battery Charge Wake		Digital
A123 Command	BCM Enable	CAN
	BCM Main Contactor Close	CAN
	Emergency Power Off	CAN
	Request GFD	CAN
Rear Traction Motor Inverter – UQM		
RTM Enable		Digital

RTM Command	Enable	CAN
	Mode	CAN
	Direction	CAN
	Command	CAN
	Limit 1	CAN
	Limit 2	CAN
Engine Control Module – GM		
APP_1 to ECM		Analog
APP_2 to ECM		Analog
12V Starter Enable		Digital
ECM Ignition		Digital
PPEI Platform General Status	System Power Mode	CAN
P2 Generator Motor Inverter – Rinehart		
P2 Wake		Digital
P2 Enable		Digital
P2 Command	Direction	CAN
	Enable	CAN
	Speed Command	CAN
	Torque Command	CAN
	Inverter Discharge	CAN
Transmission Control Module – GM		
IMS C		Digital
IMS P		Digital
ECM P/N		Digital

An important factor that the control algorithm takes into account is drivability. For example, starting an engine while driving can be startling for the driver, for this reason the VT algorithm takes factors of this nature into account in different ways. One factor is the use of motors as an immediate source of torque. These transient factors are accounted for in the mode transition sections of the execution algorithm. These sections often cause the most issues for drivability due to the dynamics of the components. Dynamics are even considered when the vehicle is not switching between modes. As an example, when the vehicle is in series mode, or M1Eon, the most efficient operating point for the engine generator set is not always adjacent to the operating point previously chosen by the algorithm and requires some traditional controls in order to reach the intended operating point. In the situation where the selection algorithm commands an increase in engine speed, engine torque, generator torque, and generator output power, the transition code would first supply the commanded output power at the current engine speed in order to meet the demand. Next, the code implements a PI (proportional integral) controller on the engine torque in order to match the operating speed to the commanded speed. Future considerations for drivability include tuning parameters for smoother transitions and matching the reactions to the accelerator pedal while in CS mode to the reactions in CD mode.

6) VEHICLE MODEL DEVELOPMENT

The fundamental goal of designing a physical plant model is to validate the control system that is being designed. In the VT Malibu, a distributed control system is implemented. In this system, all components, including stock control modules are monitored and commanded by the HVSC. The HVSC also commands operations from all of the components.

Once controller requirements are developed, a plant model is used in the verification of each of these requirements. As a baseline in Year 1, development of a series parallel vehicle model started from a model that was donated to all teams by dSPACE, one of the competition sponsors. Developed in MATLAB Simulink, this model includes detailed and comprehensive subsystems developed for components in a conventional vehicle, including an engine and driveline. Also included are calculations for vehicle dynamics and a driver model.

6.1) COMPONENT MODELS

In order to transition from the provided conventional vehicle model to a series parallel configuration, subsystems are added for all hybrid components that HEVT is integrating. This includes a motor, generator, and high voltage battery. Figure 6-1 shows a block diagram of the vehicle model from a high level. In the developed model, all component subsystems interface with other component subsystems through realistic signals, including torques, speeds or electrical energy. For example, the battery model outputs a battery voltage value that other components can reference. For the front powertrain system, both the engine and P2 generator motor calculate output torque values from fuel flow and electrical energy, respectively. The driveline then uses the torque and force values from components as well as calculating resistances and then calculating the speeds of the drivetrain. These values are then fed back to the corresponding components.

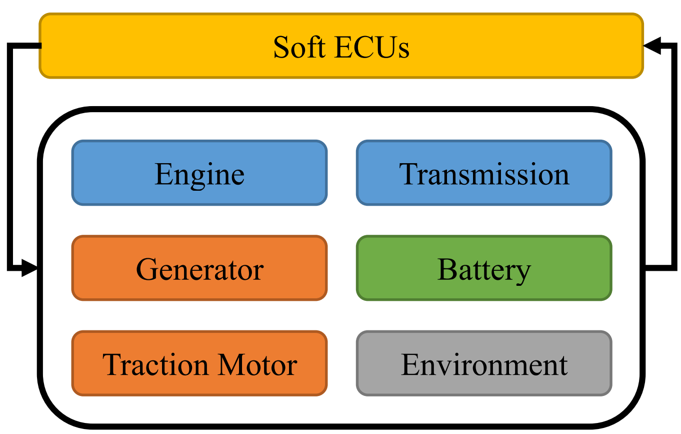

Figure 6-1: Vehicle Plant Model Organization

6.1.1) ENVIRONMENT & VEHICLE PROPERTIES

The environment subsystem contains algorithms to model drive cycles and a driver. This driver would control the typical driver interfaces (accelerator and brake pedals, gear shifter, key, etc.) to meet input driving conditions.

The vehicle dynamics properties include the values needed to calculate the driving resistances acting on the vehicle while driving. These properties are used in Equation 4, Equation 5, Equation 6, Equation 7, and Equation 8 to evaluate the energy requirements of a vehicle at a given speed, V, and acceleration:

$$F_{rolling} = mgC_{rr} \quad \text{Equation 4}$$

$$F_{aero} = \frac{1}{2}\rho C_d A_f V^2 \quad \text{Equation 5}$$

$$F_{grade} = mg \sin \Theta \quad \text{Equation 6}$$

$$F_{inertial} = m_i a_x \quad \text{Equation 7}$$

$$F_{tr} = F_{rolling} + F_{aero} + F_{grade} + F_{inertial} \quad \text{Equation 8}$$

Here F_{tr} is the tractive force required at a specific time to overcome the rolling resistance, aerodynamic drag, effects of grade, and vehicle inertia due to acceleration. In addition, g is the gravitational constant, ρ is the density of air, F_{grade} is the force opposing a vehicle driving on a grade, and $F_{inertial}$ is the force required to accelerate the vehicle. Figure 6-2 details the relationship of these forces.

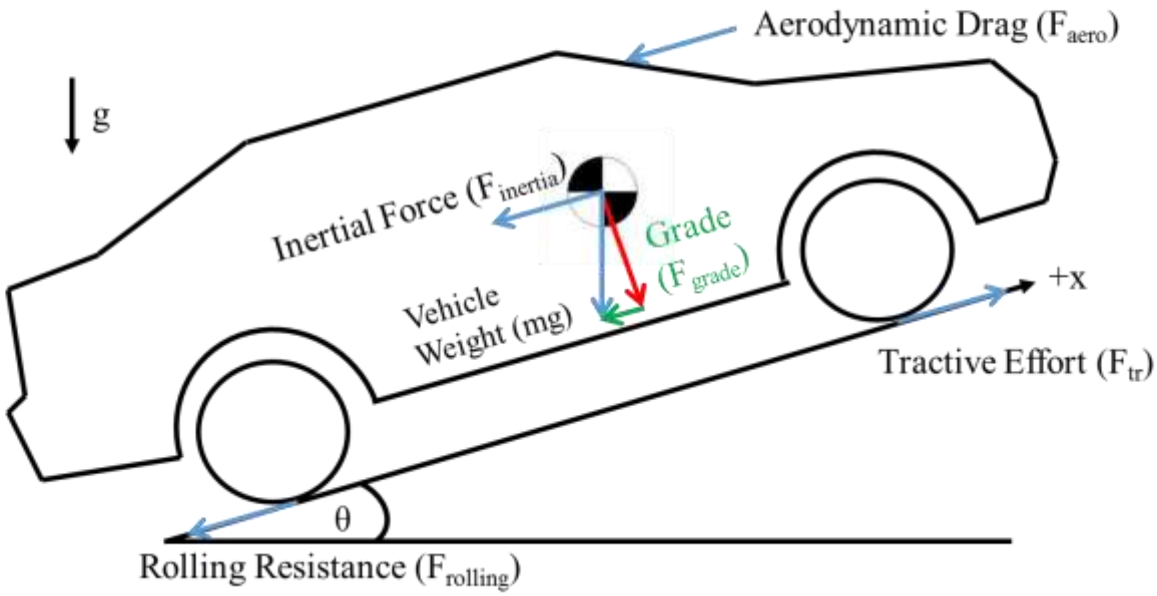

Figure 6-2: Tractive Forces Acting On a Vehicle in Motion

Both the environment and driveline models are included in the stock vehicle model donated by dSPACE at the beginning of Year 1.

6.1.2) Engine

The engine model is also included in the donated stock vehicle model but had to be parameterized to match the engine that HEVT selected to use in the VT Malibu. This model has a high level of detail including subsystems for fuel system, air path, combustion processes, cooler, and exhaust system. This subsystem is hardly changed for the hybrid plant model.

6.1.3) Drivetrain

Like the engine model, the drivetrain model is included in the donated stock vehicle model and needs to be parameterized. This subsystem includes algorithms for modeling the crankshaft, transmission, and differential. Since the torque converter is removed in the VT Malibu, it is also removed in the plant model. The torque inputs for the torque converter are then rerouted to the transmission.

6.1.4) Energy Storage System

Competition organizers along with A123 and dSPACE provided models for the high voltage battery that is used in the vehicle. This Automotive Simulation Model (ASM) saved the team a lot of work.

It includes a model of the three contactor energy distribution module which connects the battery to the HV bus in the vehicle. It also models the battery pack as a multicell pack where output voltage is a function of the input current. This model also includes a thermal approximation of the internal battery temperature. This is a useful approximation because the battery pack overheating is a very critical condition in the vehicle.

6.1.5) REAR TRACTION MOTOR

The motor model is developed using Autonomie. Autonomie has the ability to parameterize models for the motor using the small amount of information at the team's disposal. The team exports models for the UQM 125 kW motor and all of the unnecessary signals are removed and the simplified blocks are inserted into the stock vehicle model and the signal paths are determined and wired accordingly. Then, parameters for the RTM are included in the initialization files and the model is built in Simulink.

6.1.6) P2 GENERATOR MOTOR

The process for modeling the P2 motor began simple. The same RTM block is used, but loaded with parameters scaled to match the preliminary design data for the custom P2 motor based on Year 1 designs. However, the Year 1 model was immature and insufficiently modeled. Once the generator design was completed in Year 2, the model was parameterized to match these final design specifications. Figure 6-3 details the process used for the team's P2 generator model throughout the competition.

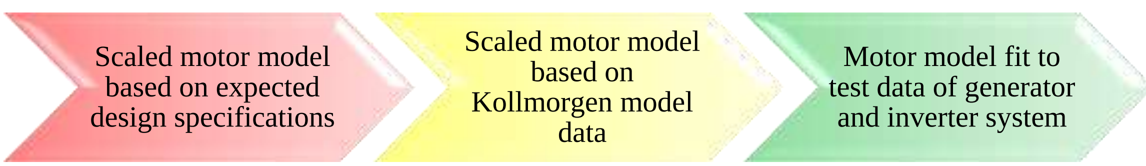

Figure 6-3: P2 Generator Model Development Flow Chart

6.2) SOFT ECUS

The last major subsystem included in the provided model is for soft Electronic Control Units (ECUs). Soft ECUs are a modeled representation of real ECUs that are included in the prototyped and final products. The purpose of these systems within a plant model, especially in a distributed control system, is to provide an interface between the controller under development and the plant models. In a production vehicle, almost all components are governed by specific controllers: engine control modules, transmission control modules, motor drives, etc. These controllers are the signal interface for the components and in order to validate a controller, that signal interface must be properly replicated. All of the subsystems provided by dSPACE are able to be parameterized in order to match the components which HEVT is actually integrating into the vehicle.

6.2.1) ENGINE & TRANSMISSION

The VT Malibu features a 2.4 L FlexFuel engine. This engine requires precise control to achieve high performance and efficiency with low emissions. The engine is outfitted with sensors such as crank angle and speed sensors. There are also oxygen sensors in the exhaust and an air-flow sensor in the intake among others. With this information, fuel delivery and ignition timing must be controlled to achieve desired engine performance. For this type of low level control, the stock GM ECM is used. The main control input to the ECM is desired torque while the main output are actual torque and engine speed. Other important outputs of the ECM are information such as engine coolant temperature. A soft ECM was provided by dSPACE and is already implemented into the model, however, this soft ECM is a generic version provided with the dSPACE software package and does not replicate all of the actual I/O that the GM ECM has, specifically the CAN messages. Since this I/O is one of the primary reasons for building soft ECUs into the vehicle model, the functionality of the soft ECM needs to be expanded. For this reason, the generic dSPACE soft ECM is augmented by adding realistic control interfaces gathered from bench testing (pg. 62). Bench testing reveals that the GM ECM transmits 24 CAN messages during regular operation and 2 CAN messages when diagnostic trouble codes are set. The plant model is updated to populate as many of the signals in these messages as possible.

Similarly, the Transmission Control Module (TCM) inside the automatic transmission is responsible for controlling shift solenoids and transmission fluid pressure. The shift solenoids inside the transmission allow for the fluid flow throughout the transmission, as well as directly control the clutches inside the transmission. This is important in controlling what gear the transmission is operating in. The control input to the TCM is the desired gear for the transmission while an output is the actual current gear. A soft TCM was provided by dSPACE and is already implemented into the model. However, like the soft ECM, the functionality of this is expanded to more accurately replicate the I/O of the actual TCM. Again, CAN messages that are both transmitted and received by the TCM are added to the plant model.

6.2.2) Battery Control Module

The A123 HV battery pack has three contactors that must be controlled in order to enable and disable the HV system. The pack is also outfitted with temperature sensors and voltage sensors throughout each module to monitor the conditions of individual cells. The A123 Battery Control Module (BCM) is responsible for controlling the battery contactors, and interfacing with all the sensors. The control input to the BCM is the command to enable the HV system while important information transmitted by the BCM includes SOC, cell temperatures and voltages, and allowable current limits. The BCM also monitors and transmits ground isolation levels. The soft A123 BCM was provided by A123 and dSPACE and did not require significant modification. The model includes all of the CAN signals the ESS sends and receives and all of the necessary parameters. Coordinating I/O between the ESS model and the controller has been greatly simplified by the use of this ASM.

6.2.3) RTM & P2 Inverter

To operate the traction motor, the 3-phase current that it is supplied must be controlled to generate the requested torque. This requires a HV motor inverter which serves as the control interface to the motor. The inverter monitors temperatures in the motor as well as the position of the rotor to enable precise control. The main input to the motor inverter is requested torque and direction while the output is 3-phase current that translates to actual motor torque and speed. The inverter also reports temperatures of parts in the motor as well as temperatures of the power electronics in the inverter. Just like the previous soft ECUs, the proper communication signals (i.e. digital, CAN, etc.) are implemented to achieve the previously mentioned functions in the plant model.

Once the Rinehart PM100DX was chosen in Year 2, the P2 soft ECU was implemented. This is a component that requires very precise controls in the final vehicle architecture and a high fidelity model. Since the P2 generator motor is custom built, it is difficult for the team to gather every piece of data necessary to accurately model the P2. It is easy to start with the same baseline as the RTM soft ECU and simply modify the parameters and corresponding CAN messages to match the requirements for the P2 inverter.

6.3) MODEL VALIDATION

As mentioned previously, it is very important to validate the model with test data to ensure that it correctly portrays the vehicle throughout the development cycle. An accurate model allows for faster implementation of control code changes and will ultimately increase the overall quality of the control strategy implemented. To test component models the team followed the procedure steps listed below. Priority of these changes is based on Design Failure Mode Effects Analysis (DFMEA) for different elements of the control algorithms. Control algorithms related to safety critical systems are of utmost importance and model elements related to testing these algorithms are the first to be validated.

1. Collect data from test bench or vehicle testing. Take careful note of initial operating conditions such as initial battery SOC and cold or warm engine start and code revision.
2. Log vehicle parameters throughout the data collection. Important vehicle parameters to log are vehicle speed, component torques, battery SOC, and vehicle operating mode charge depleting mode or charge sustaining mode.
3. Use the logged vehicle speed to create a matching simulation drive cycle. Input other initial conditions such as SOC and operating temperatures.
4. Log vehicle parameters throughout the simulation. Important vehicle parameters to log are vehicle speed, component torques, battery SOC, and vehicle operating mode charge depleting mode or charge sustaining mode.
5. Compare vehicle drive log and simulation log for differences.
6. Conduct root cause analysis to determine cause for simulation differences.
7. Implement model changes and repeat steps 5-7 until the model is meets accuracy requirements.

6.3.1) <u>COMPONENT MODELS</u>

Figure 6-4 shows data from the vehicle model and actual drive data from a dyno test at the Emissions Testing Event (ETE). The plot shows vehicle speed for both cases and motor torque collected from the simulation and the vehicle after the test. The plot shows that the two torque curves are in very close agreement with each other. This shows that the models reaction to control commands is very similar to the vehicles. It also shows that the torques required to meet the same driver demand are very similar so the resistances to driving (i.e. aerodynamic drag, rolling resistance, e.g.) are modeled appropriately. Further examination of the results of this test showed that the battery SOC from the simulation matched the in-vehicle data within 5% for the entirety of test, proving that the motor model also accurately simulates the motor losses and electrical energy required to produce the corresponding torques.

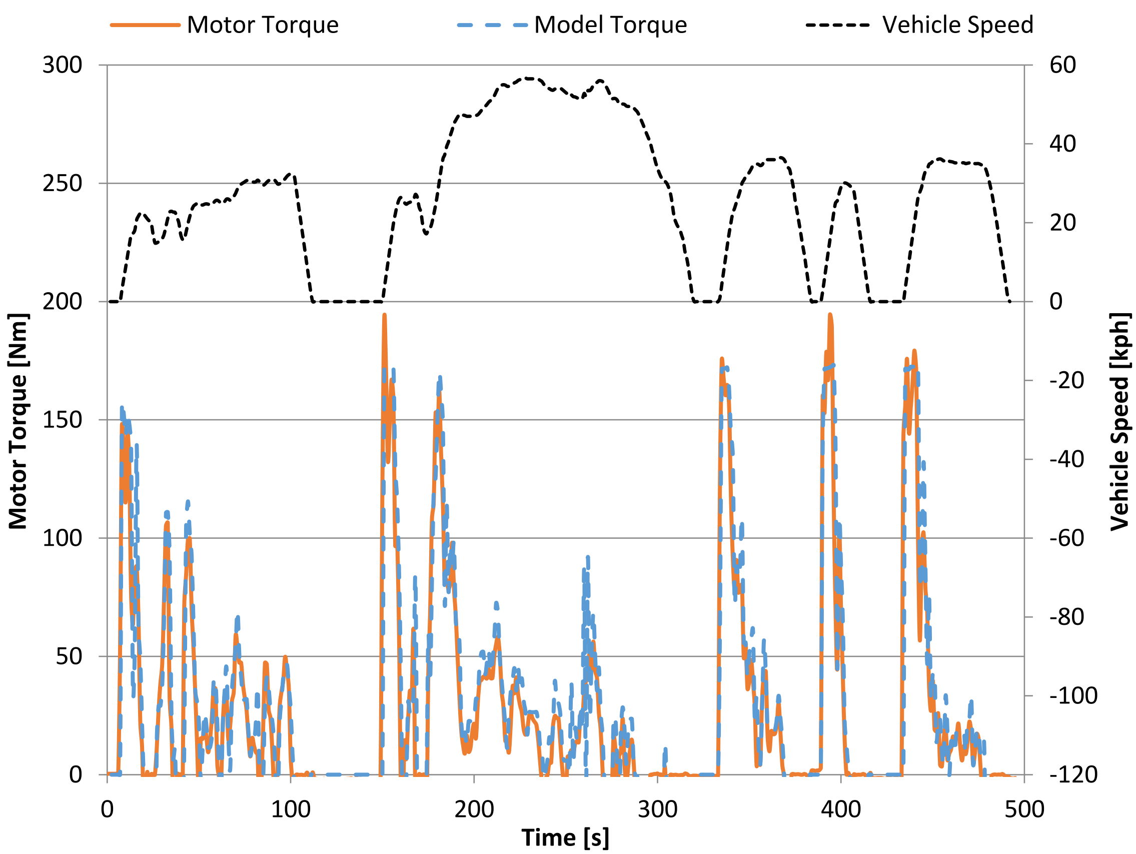

Figure 6-4: RTM Torque Validation Data (SIL)

Tests like this have been completed for many of the main powertrain components. The focus has been on those models which have been created and modified models as opposed to competition supplied ASMs. For example, access to the motor dyno at Kollmorgen for testing of the P2 generator motor provided data that showed the initial design specifications had to be slightly revised. Since these tests were only run with the P2 and P2 inverter, simulation results were only analyzed for these two components as well. The data was able to help improve the P2 model significantly.

Having a detailed model of the P2 generator and inverter is critical to the powertrain control and the transition between operating modes in the actual vehicle. Without a sufficient model, the controller could falsely pass tests and then have detrimental effects in the vehicle. If the inertia or physical delays are not properly accounted for, transient controls could cause catastrophic damage of irreplaceable components. Figure 6-3 details the process used for the team's P2 generator model throughout the competition.

6.3.2) SOFT ECUS

Having accurate soft ECU models is also critical to accurately test the control algorithms. In the distributed control system that HEVT is implementing in the vehicle, the HVSC only communicates with the individual component controllers (i.e. ECM, TCM, motor inverter) and not the actual components (i.e. engine sensors, motor, battery). Thus it is critical to validate the operation of these soft ECUs to ensure that the communication algorithms are sufficiently tested.

Some of the first operational tests that are executed on any new version of the control code validate the safety critical startup procedures of the vehicle. In a conventional vehicle, a startup is easily noticed when the engine turns on. However, in a hybrid vehicle such as the VT Malibu, a startup is defined as the energizing of the high voltage system by the action of contactors closing. Figure 6-5 and Figure 6-6 demonstrate the proper startup procedure programmed into the control code. This procedure includes 3 main conditions:

- Gear shift selector is in Park
- Brake is pressed
- Key is Cranked and released to On

These figures are tests that are run in order to verify that the controller will not command the contactors to close unless all 3 startup conditions are met. In both figures, red traces designate improper startup conditions while green traces denote those states which meet the conditions for startup. Figure 6-5 shows an improper startup procedure and shows that the contactor command is never sent. Figure 6-6 shows the proper startup procedure and shows that the contactor command is sent until the key is turned to the off position.

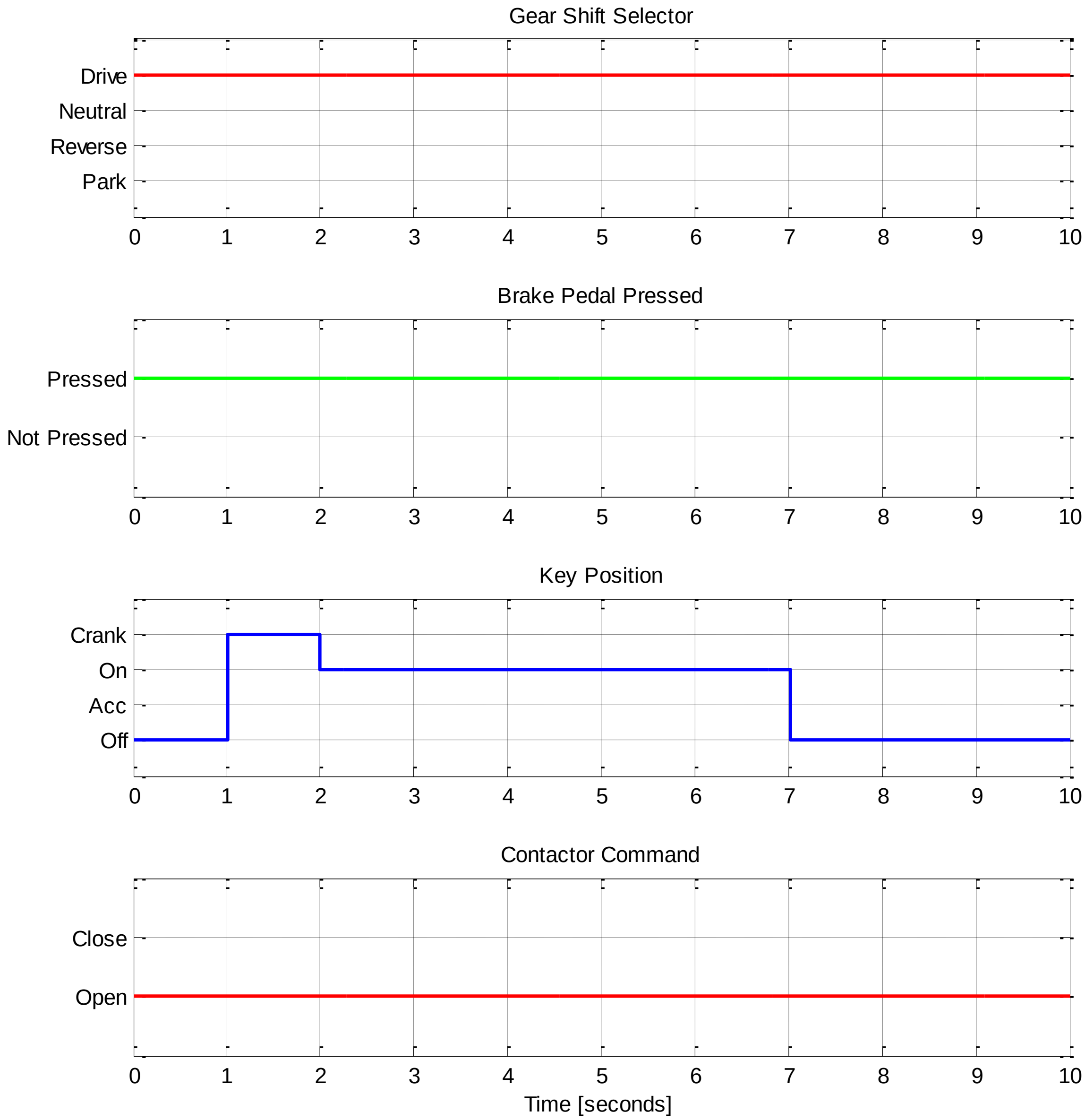

Figure 6-5: Incorrect Startup Procedure (Gear Shift Selector Not In Park) (SIL)

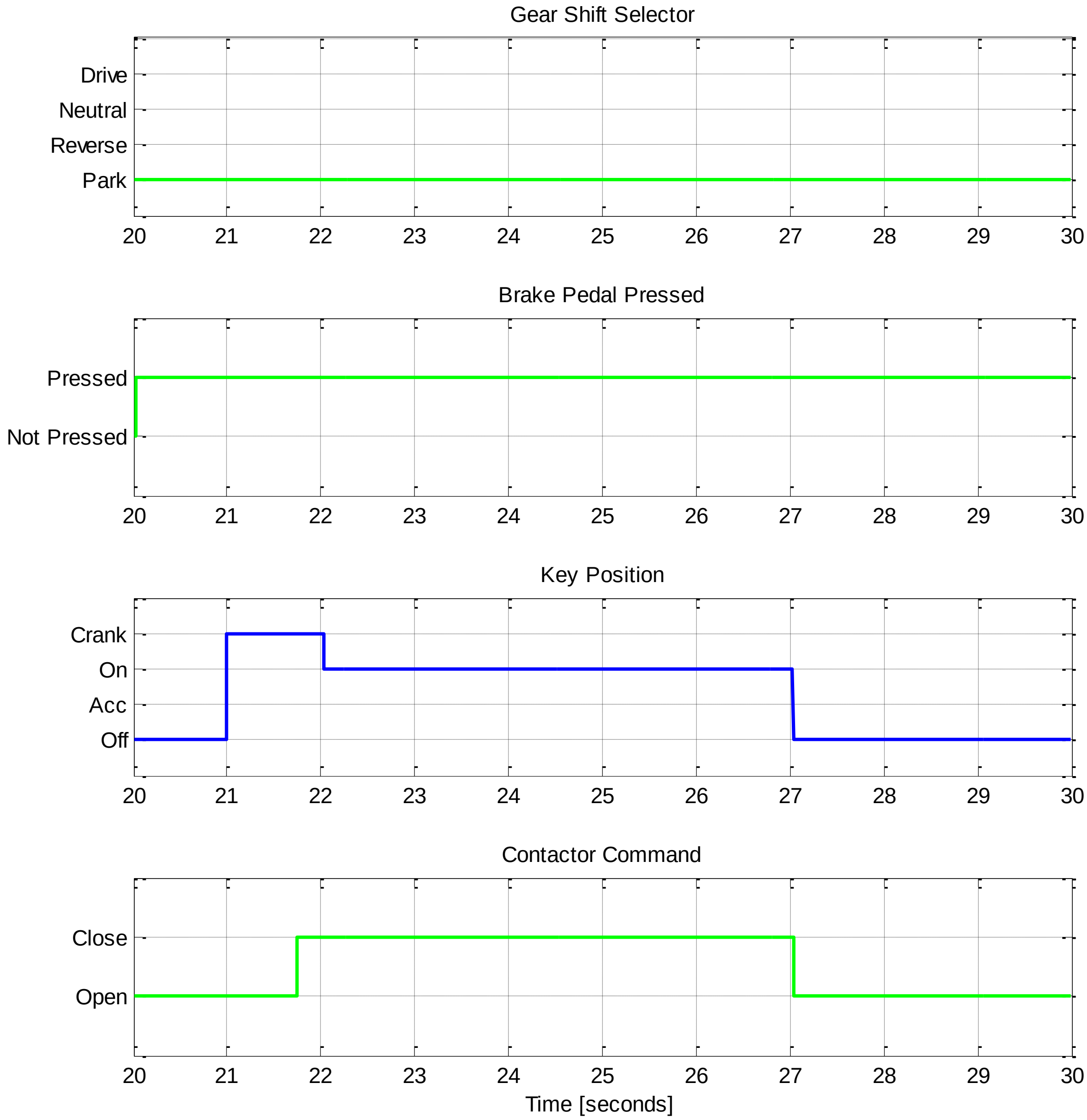

Figure 6-6: Correct Startup Procedure (SIL)

While it is important to validate the procedures which the soft ECUs must simulate, it is also necessary to make sure that the soft ECU models are transmitting all of the correct diagnostic information. Further tests of soft ECUs check certain diagnostic functions which affect the developed control algorithms.

7) SIMULATIONS & TESTING

After development of requirements, construction of a control algorithm, and production of sufficient vehicle model, a control system developer can proceed to testing. Testing, however, is such a broad term. Many different levels of tests were run on the components, portions of the algorithm, and the full control algorithm. Those tests are detailed in the following sections.

7.1) <u>TEST STAND DATA</u>

To effectively control the components in the VT Malibu, the exact interfaces for each component must be known. This information can be difficult to determine for custom components or components that are being used outside of their original design space. Components such as the high voltage battery pack and rear traction motor were both designed for use in custom hybrid applications, were provided with detailed documentation on the control interfaces, and were relatively simple to control in bench environments and then in the vehicle.

7.1.1) <u>P2 GENERATOR MOTOR DYNO TESTING</u>

The P2 generator motor, however, is a custom permanent magnet synchronous motor developed with the help of Kollmorgen of Radford, VA. The first step is to tune the Rinehart firmware to work with the generator. This tuning was done with the help of Rinehart and Kollmorgen engineers. Kollmorgen was able to provide motor specifications to Rinehart to parameterize motor settings. Since Rinehart was not able to validate these parameters with a physical motor, that job was the responsibility of HEVT. Through the use of a motor dynamometer at Kollmorgen's prototype laboratory, the Rinehart PM100DX was tested with the custom Kollmorgen P2 generator motor and verify system operation. The following tasks were completed while bench testing the P2 generator and inverter:

- Inverter firmware update procedure
- Inverter parameter calibration
- Inverter communication protocol (CAN)
- Resolver calibration
- Generator control
    - Torque mode
    - Speed mode
- Generator thermal limitations

These tests were completed by running the P2 generator motor with a dyno motor setup as shown in Figure 7-1. The inverter was also connected to a high voltage power supply and a coolant system with a flow rate of 10 liters per minute and 55 °C inlet temperature. Figure 7-1 shows the P2 generator motor mounted to the table and connected at the shaft to the existing dyno setup and sensor. On the lower shelf of the table is the Rinehart inverter with current sensors on each phase.

Figure 7-1: P2 Generator Motor Test Setup

Through the testing process, the initial thermal designs were realized to be inaccurate. Multiple tests were run to reach a steady state temperature. Tests run to the initial maximum temperature resulted in an expansion of the ecap material used to enclose the stator. This caused the air gap between the rotor and ecap material to shrink and caused the ecap to rub. This issue can be easily cleaned out in the test lab environment but would have serious side effects if the dust was allowed to remain in the motor during operation and would be impossible to clean while the generator is installed in the vehicle. New temperature limits were then derived from the tests and validated with new tests. Figure 7-2 shows the results of one of these tests. The test conditions are as follows for Figure 7-2:

- 21.0 kW mechanical power
  - 2600 RPM
  - 77 Nm Torque
- 19.2 kW DC electrical power generated
- ~92% system efficiency (including inverter)
- 10 L/min of coolant @ 55°C inlet temp

The data in Figure 7-2 is collected from the CAN bus and transmitted by the P2 inverter. The inverter measures this value from a positive temperature coefficient temperature sensor located on the interior surface of the stator core. During testing, there were additional temperature sensors used to measure temperatures at various other locations in the generator, however, those are not available for use in the vehicle application. While this temperature is not allowed to exceed 100°C, the hottest temperature in the motor reaches approximately 165°C.

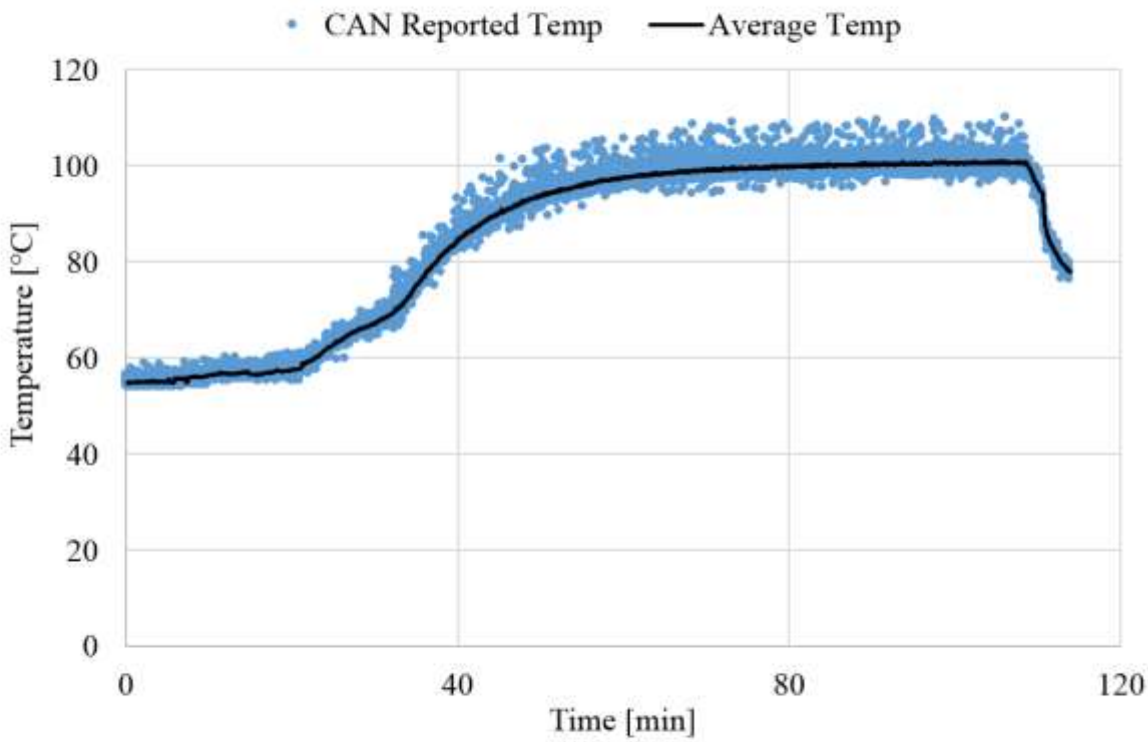

Figure 7-2: P2 Generator Motor Thermal Test Data

The data and procedures collected during the testing are important to final vehicle integration and P2 control limits.

7.1.2) *ENGINE TEST STAND*

In addition to running tests on the P2 generator motor, it was important to determine the proper control interface of the GM supplied engine. This interface is important because the engine and its corresponding engine control module (ECM) are being used in a vastly different environment in the VT Malibu. For example, the engine is not intended to run during charge depleting operation of the vehicle and thus the proper control interface to start the engine on command must be determined.

The secondary goal of testing the engine outside of the vehicle is to develop and validate a full wiring harness. The supplied engine and ECM were not directly compatible and an engine harness had to be significantly modified to mate with the engine, ECM, and other vehicle systems.

Figure 7-3 shows the setup of the engine test stand including the following elements:

- 12 Volt Battery & High Power Charger
- Fuel Tank & Pump
- Fuse, Relay, & Switch Board
- GM 2.4L E85 Engine

- GM Engine Control Module
- Radiator, Fan, & Coolant Reservoir
- P2 Generator Motor

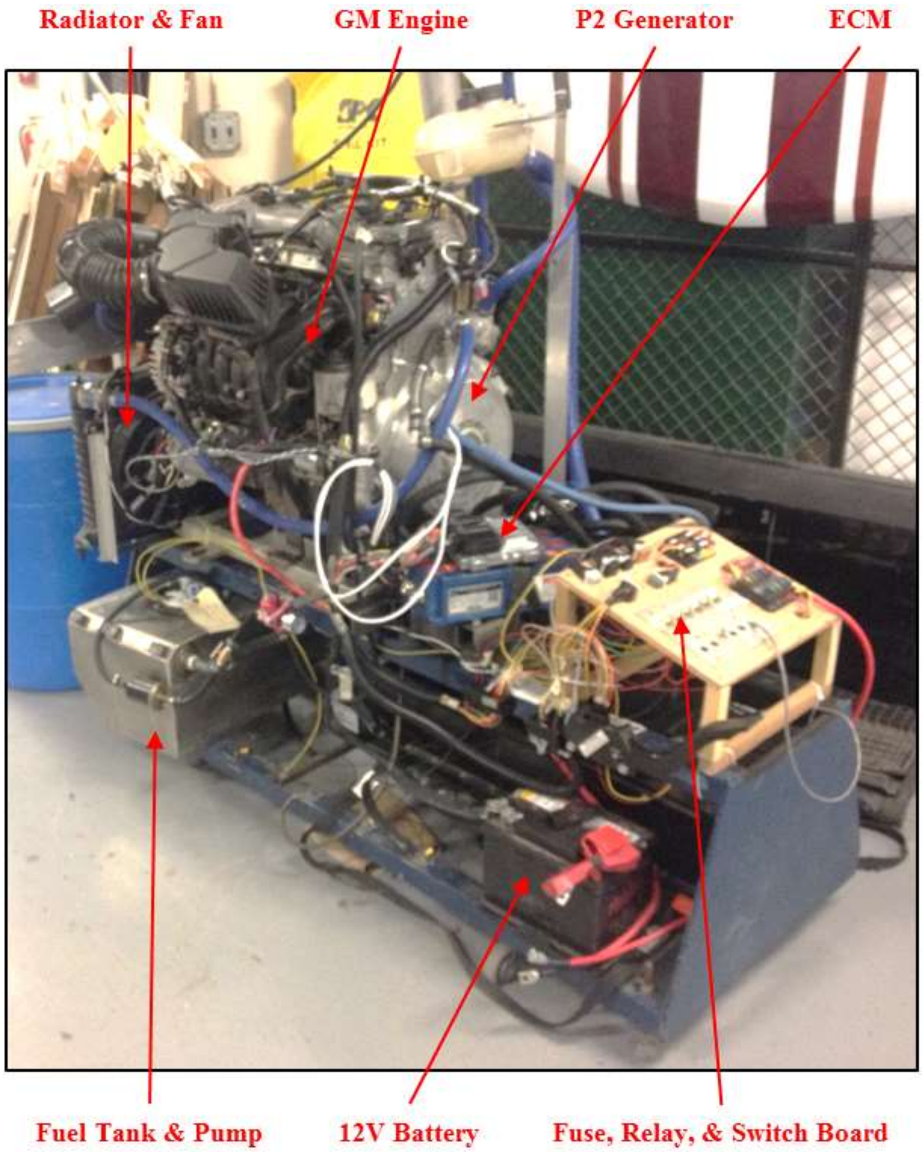

Figure 7-3: Engine Test Stand Setup

This test stand allows the minimum requirements to run the engine to be determined. There are four required CAN signals to operate the engine and they are detailed in Table 7-1. The digital Run/Crank and Accessory signals sent by the Body Control Module are also key to operating the engine. The functions of these signals are detailed in Table 7-2.

Table 7-1: CAN signals required to run engine

Message Name	Hex ID	Signal Name
PPEI_Trans_General_Status_1	0x0F9	Transmission Output Speed
PTEI_Trans_Torque_Request_HS	0x199	Transmission Alive Rolling Count
ETEI_Transmission_General_Status	0x19D	Transmission Alive Rolling Count
PPEI_Platform_General_Status	0x1F1	System Power Mode

Table 7-2: Digital signals required to run engine

Signal Name	Purpose
Accessory	Wake ECM, enable CAN communication
Run/Crank	Enable Powertrain Relay which powers engine actuators

Once installing the engine in the vehicle, the engine would not start. The primary conditions of spark, air, and fuel were all tested and it was determined that the ECM was not firing the fuel injectors even though there was sufficient fuel pressure at the fuel rail.

One change that had been implemented between engine testing on the test stand and in the vehicle was control of the 12V starter with the Hybrid Vehicle Supervisory Controller (HVSC) instead of the GM ECM. This change was done so that the ability to start the engine with the P2 generator motor could be retained and the 12V starter would not need to run.

However, the GM ECM performs many diagnostic and security checks. One such check can determine if the circuitry to the starter relay is faulty. Since this control has been removed from the ECM, it no longer sees a resistance consistent with an automotive micro relay. The addition of a new circuit to replicate this resistance is sufficient to start the engine again.

7.2) SIMULATIONS

Unfortunately, valid SIL and HIL simulation results were not able to be generated to validate the entire control strategy. While some component models were able to be generated and validated (Model Validation , pg. 53), the entire vehicle model was never completed and all modes of operation were not able to be tested. Some SIL data has been included in other sections for EV tests to show functionality of some portions of the algorithms.

One primary reason for this is the inability to follow the development cycle for the model. As detailed in (Vehicle Model Development, pg. 48), dSPACE provided a model of the base donated Malibu to all of the teams. Initial tests were run on this conventional vehicle model to get familiar with the model structure and operation. However, a strategic approach to turn this model into the VT Series-Parallel vehicle was not employed and a full vehicle model was never completed. A strategic approach would have been as follows:

1. Develop model subsystems (i.e. RTM subsystem)

2. Test model subsystems
3. Insert subsystems into vehicle model
4. Run system tests

This approach would have allowed the model developer to troubleshoot issues much quicker. However, without following this approach, issues were very difficult to track down. Often, issues would be traced throughout various subsystems in the model but lead back to the initial location.

Additionally, SIL testing is typically done prior to HIL testing as it performs checks on the algorithm and does not require a prototype controller. The EcoCAR 2 deliverables however put the focus for Year 1 of the competition on HIL testing. Year 2 focused on SIL testing. This order caused some model development issues as well. The VT model is designed so that a subsystem can be replaced which either includes the control algorithms for SIL or the hardware interface blocks (i.e. analog digital converter or CAN blocks) for HIL. Integration of the control algorithms proved to be much more difficult when implemented after the hardware interface blocks.

Furthermore, lack of specifications and data caused delays in the modeling process. The primary issue here is the P2 generator motor. The design was not finalized until late in Year 2 and was not fully tested until midway through Year 3. The lack of information and timeliness caused issues in the development of a sufficient model. Moreover, while GM gives EcoCAR 2 teams access to a lot of proprietary information, there is still a significant amount of information that is required to develop a sufficient model. As mentioned in Soft ECUs (pg. 51), the dSPACE provided models were able to model the components but not the communication interface. GM did not provide enough data to develop the logic or reactions to certain commands. This all had to be determined through bench or vehicle testing and then incorporated into the model.

Lastly, the focus of the work here is to get a running vehicle and time was devoted first and foremost to getting components working on a bench and then in the vehicle. Modeling was a second priority throughout this competition, though it could have been a much more useful tool when utilized correctly.

8) VEHICLE INTEGRATION

Once component interfaces are known and the control algorithms have been tested, vehicle integration is the next step. The following section details the design of the vehicle control systems.

8.1) SYSTEM DESIGN

System requirements are developed early in the process. A sample of these system requirements have already been provided in Table 4-1. These requirements are typically driven by the target goals and by available components. After requirements are determined, the rest of the system needs to be designed. Some of the choices required to complete the system design are as follows:

- Communication network design
- Controller hardware design
- Sensor and actuator design
- Ways to meet additional customer requirements.

8.1.1) HYBRID VEHICLE SUPERVISORY CONTROLLER

To meet all of the control system requirements, a dSPACE MicroAutoBox II was chosen from the competition supplied options. The MicroAutoBox (MAB) can start up autonomously after power-up, with ECU-like boot-up times. A PC or notebook can be easily connected to download an application or analyze data. Application programs are stored in nonvolatile memory. The MicroAutoBox contains signal conditioning for automotive signal levels and an integrated programmable data recorder for long-term data acquisition. Figure 8-1 and Table 8-1 show more details of the chosen HVSC.

Figure 8-1: dSPACE MicroAutoBox II Used as HVSC, https://www.dspace.com, used under fair use, 2014.

Table 8-1: MicroAutoBox II 1505/1507 Specifications

Parameter	Value
Analog Input	16 12-bit channels 0…5 V
Analog Output	8 12-bit channels 5 mA max
Boot time	340 ms (for 3 MB application)
CAN Interface	4 channels
Digital I/O	Up to 32 digital inputs Up to 26 digital outputs
Power Consumption	25 W max @ 12 V
PWM I/O	4 PWM inputs 4 PWM outputs

Note the ample amount of I/O that the MicroAutoBox is capable of, as it is typically used as a prototyping controller in industry. The ability to have 4 CAN channels is also key to interface with all of the CAN networks that exist or have to be added to the vehicle. The quick boot time is also an enticing feature so that a driver does not have to wait for the controller to be ready before starting and driving the vehicle.

8.1.1.1) *CONTROLLER CALIBRATION*

One desirable feature of many production controllers is the ability to calibrate quickly and easily. Production engine controllers can be used across many different engine and vehicle platforms by simply changing the parameters which define different algorithms. These parameters can be changed via a scan tool without regenerating the full control code.

The MAB has the ability to store parameters in non-volatile flash memory and reload those parameters upon controller start-up each time. This feature is especially useful for tuning algorithms such as regen braking (Figure 5-3) where the parameters need to be tuned and saved.

8.1.1.2) *DIAGNOSTICS*

An additional feature that is common on most production vehicles is an accessible diagnostic infrastructure for on-board diagnostics. This refers to a diagnostic data management structure that can provide the following functionality:

- Links a diagnostic algorithm to a diagnostic trouble code (DTC)
- Stores the DTC in non-volatile memory
- Stores freeze frame data (snap shot of key parameters when a fault is confirmed)
- Enables communication of fault information to an off-board scan tool
- Enables DTCs to be cleared by an off-board scan tool or by the diagnostic system itself

Along with the MAB, dSPACE provides both a full software suite for working with the MAB as well as a Simulink block set for implementation in the control algorithm. Included in the software suite is ControlDesk Next Generation, which unites functionalities that used to be covered by several specialized tools. It provides access to simulation platforms and connected bus systems, and can perform measurement, calibration and diagnostics on ECUs. Its flexible, modular structure provides high scalability to meet the requirements of specific application cases. The Simulink block set allows diagnostic information to be stored and ControlDesk functions as the scan tool with access to the diagnostic information.

8.1.2) Sensor & Actuator Design

In the VT Malibu control system, most sensors and actuators are incorporated in the individual subsystems and handled by the individual component controllers (i.e. crankshaft position sensor and fuel injectors controlled by the ECM) which is typical of a distributed control system. However, there are some measurements that the HVSC must make because they are customized or added sensors. Table 8-2 details the sensors developed/used in the VT Malibu control system.

Table 8-2: Sensors Added To VT Malibu

Sensor	Type
Accelerator Pedal	Double Potentiometer
Charge Plug Proximity Sensor	Analog Input
Coolant Temperature Sensor	Negative Temperature Coefficient
Disable Regen Braking Switch	Single Pole Single Throw Switch
Force Charge Sustaining Switch	Single Pole Single Throw Switch
Force Engine On Switch	Single Pole Single Throw Switch

In addition to sensors, actuators need to be integrated into the control system. As mentioned previously, many of the actuators are handled by individual component controllers, though the HVSC must actuate a few simple components. Table 8-3 details the actuators developed/used in the VT Malibu control system.

Table 8-3: Actuators Added To VT Malibu

Actuator	Type
Charging LED	Digital Output
ECM Measured Accelerator Pedal	Analog Output
Ground Fault Indicator LED	Digital Output
Vehicle Ready LED	Digital Output
Wake Signals	Digital Output

8.2) Network Design

The HVSC must also communicate with each of the ECUs in order to indicate the low level controls that will run each component. Furthermore, the MAB is a critical piece of equipment for the team because it acts as gateway for different CAN networks in the vehicle. Nearly all of the components in the vehicle are controlled via CAN; therefore, use of the CAN networks already present in the Malibu is the obvious choice for network designs. There are four primary CAN busses present in the stock Chevy Malibu and their purposes can be seen in Table 8-4. One additional CAN bus is added to the VT Malibu to accommodate the CAN controlled relay box and coolant pump which run on a different baud rate than all of the other controllers in the vehicle. Another bus is added for the P2 inverter. The P2 inverter requires a separate bus because it causes issues when on the Powertrain Expansion bus (P2 CAN, pg. 70).

Table 8-4: VT Malibu CAN networks

CAN Bus	Stock Use	VT Use
HSGMLAN	Primary powertrain components	Stock powertrain components
Powertrain Expansion	Additional/hybrid components	Added VT components
Chassis Expansion	Safety critical components	Same as stock
Low Speed GMLAN	Body components	Same as stock
Midspeed Bus	N/A	Relay box and coolant pump control
P2 CAN	N/A	Control of Rinehart inverter and P2 generator motor

Figure 8-2 displays the entire CAN network layout in the VT Malibu. HEVT assigned all ECUs to a determined CAN bus, although some ECUs are shared by two busses. Note that the Chassis Expansion bus does not include any HEVT components and is left in the stock configuration. Having four other separate busses gives HEVT the ability to physically separate components to have greater control over transmitted signals with the HVSC. One example of this is having the TCM on a different bus than the ECM, which allows the transmission to be in neutral while the engine gives torque to the P2 generator motor as in CS Series mode. Typically, in stock applications, the ECM and TCM communicate heavily and operate in harmony with each other. To enforce the desired control in the VT Malibu, it is necessary to have them on separate busses and to produce messages to each component to satisfy the communication requirements. Additionally, had all of the units been on the same CAN bus, the CAN bus would have been overloaded and messages would have been lost frequently. A CAN bus with a load of greater than 60% is often considered overloaded.

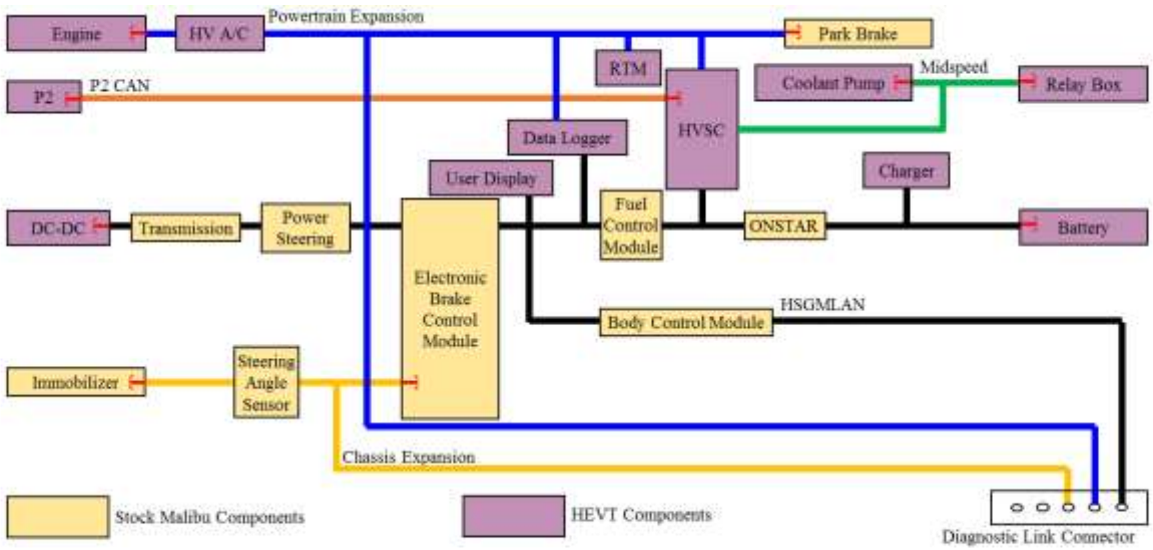

Figure 8-2: VT Malibu CAN network layout

8.3) P2 CAN

For the most part, the physical CAN wiring has proven to be robust and capable of meeting specifications. However, during the competition sponsored ETE, the team discovered that there are significant CAN issues that appear on the Powertrain Expansion (PE) CAN bus when the P2 inverter is connected. The CAN error frames that were detected by the control code are summarized in Table 8-5.

Table 8-5: PE CAN Bus Error Frames During P2 Inverter Operation

P2 Inverter On	**--**		x	x
P2 Inverter Enabled	**--**			x
Busload	**%**	38%	65%	68%
Error Frames	**fr**	10	0	6883
Average Error Frames	**fr/s**	~0	0	19

When the inverter is not enabled, there are minimal error frames that appear on the CAN bus regardless of whether the P2 inverter is on or off. However, error frames appear once the inverter is enabled. This case is shown in the last column in Table 8-5. A more serious issue was then identified regarding the inability to reliably control the motor speed through the CAN interface due to error frames. There were cases that occurred when the P2 generator motor spun while a zero speed command was sent through the bus.

Errors alone are not necessarily a cause for concern. Some errors happen naturally such as stuff bit errors. However, the errors on the PE bus caused by enabling the P2 inverter are causing the CAN transceiver on the HVSC to fault to a bus off state less than 1 second after enabling the inverter. This causes control issues between the HVSC and inverter.

These issues are solved by moving the P2 inverter to its own CAN bus (Figure 8-2). This is not an ideal situation because one of the main benefits of CAN is the multi-master broadcast aspects where each node is able to send and receive messages. With the P2 on a separate CAN bus for two way communication with the HVSC, there are no major communication issues.

9) VEHICLE TESTING & OPTIMIZATION

In keeping with the V-cycle (Figure 1-2), the final stage of control system development is the testing and optimization of the system and algorithms. This stage can be completed through a variety of means. HEVT uses a chassis dynamometer for repeatable testing under different strategies and on-road testing for finalizing control algorithms under real world conditions.

9.1) TRC DYNO TESTING

The competition organizers were able to secure a week of vehicle testing for all teams at Transportation Research Center (TRC) dyno and emissions facilities. While at the dyno facilities, HEVT gathered drive cycle, emissions, thermal, and CAN noise data at ETE. The team made modifications to improve the efficiency and performance of the vehicle using the conclusions drawn from the drive cycle data. The information from the emissions data is used to make improvements in charge sustaining operation modes. The data and plans are discussed in more detail below.

9.1.1) ENERGY CONSUMPTION LESSONS

9.1.1.1) REGENERATIVE BRAKING

The vehicle underwent chassis dynamometer testing to determine the effect of adding regenerative braking (regen braking) on electric energy consumption. Competition organizers provided a drive trace of the on-road test that will be run during the Emissions & Energy Consumption event at final competition. This drive trace includes three primary portions listed in Table 9-1.

Table 9-1: Competition Provided Drive Schedules

Cycle	Description
ToTrack	"public" road driving from the garage/pits to the circle track, adhering to posted speed limits
CityHighway	controlled, closed-course, repeatable drive schedule based on signage on the circle track
FromTrack	"public" road driving from the circle track back to the pits, adhering to posted speed limits

At competition, the cycles will be run in the following order for a total of 104 miles:

- ToTrack x1
- CityHighway x3
- 20-minute key-off break
- CityHighway x4
- FromTrack x1

While testing on the dyno, the vehicle ran one cycle each with and one without regen braking active of the CityHighway and ToTrack drive schedules. Figure 9-1 shows the energy consumption from the ESS over time for the CityHighway cycle.

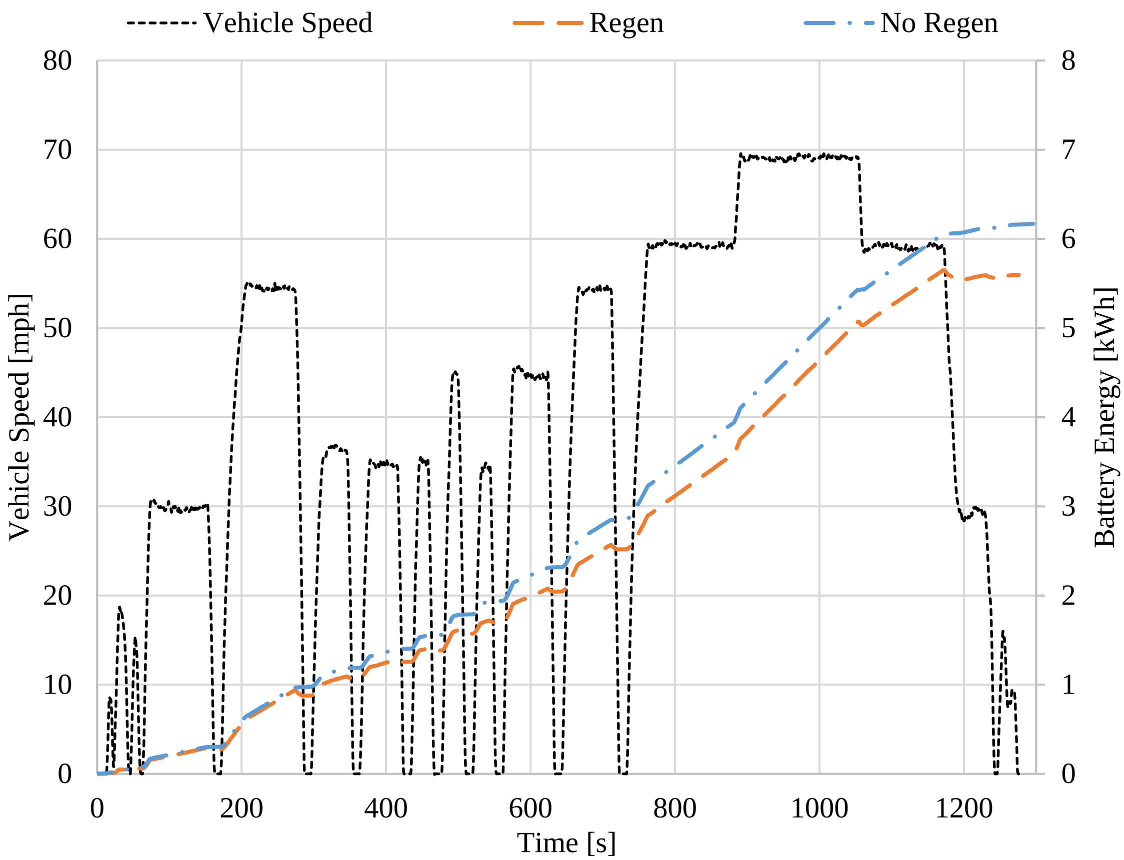

Figure 9-1: Energy Consumed From ESS vs. Time for the CityHighway Drive Cycle

Note that the vehicle speed over time is overlaid on the graph for reference. As shown on the plot, a 9.1% decrease in energy consumption was achieved when regen braking was active. Figure 9-2 shows the energy depleted from the ESS over time for the ToTrack cycle without grade emulation. From the ToTrack cycle, a 19.9% decrease in energy consumption was achieved when regen braking was active. This larger decrease in energy consumption compared to the CityHighway cycle is due to the less aggressive drive schedule and shorter distance of the ToTrack. The expected results at competition will improve for the ToTrack portion since there is a large downhill portion that was not simulated on the dyno. The results from both drive cycles show that regen braking is an important component in reducing energy consumption.

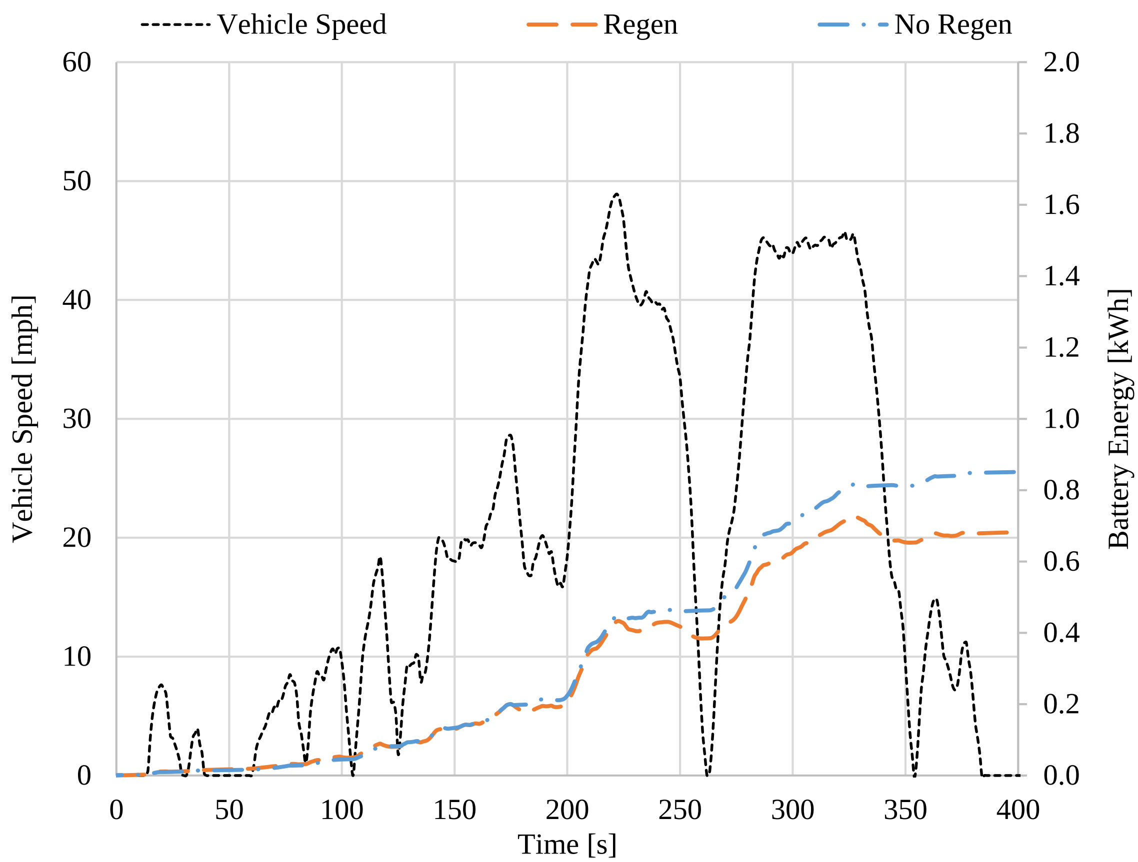

Figure 9-2: Energy Consumed From ESS vs. Time for the ToTrack Drive Cycle

9.1.1.2) ***ENGINE OPERATION***

At the time that HEVT had access to the TRC facilities, the charge sustaining mode was not fully operational. Therefore, to make use of the emissions measurement equipment at TRC, data was also collected during engine idling while on the chassis dynamometer. This testing was done to observe the differences between a cold start and a warm start. The engine idled for approximately 300 seconds for both tests. Plots of the engine speed and engine temperature over time for both tests are shown in Figure 9-4 and Figure 9-4.

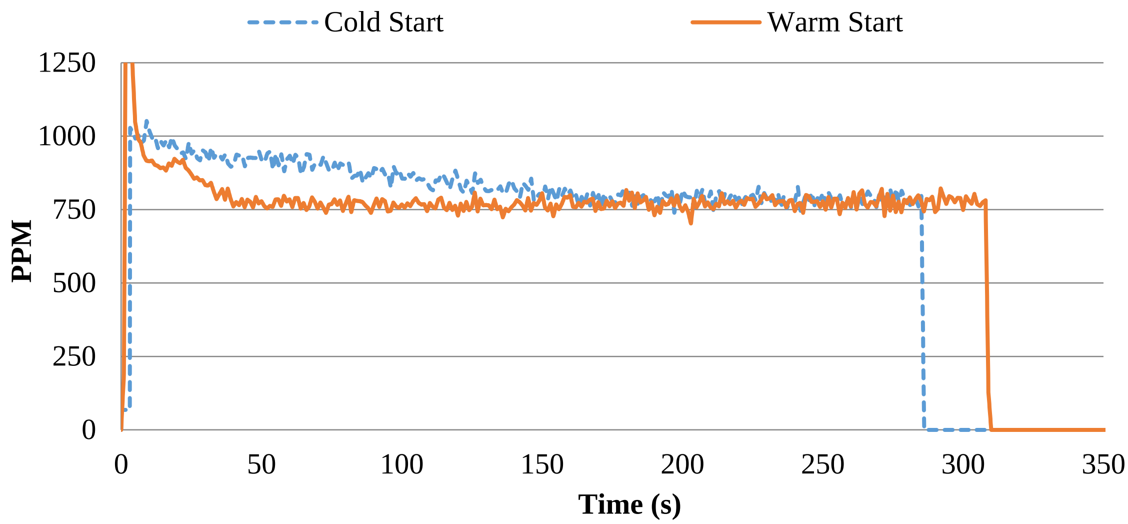

Figure 9-3: Idling Engine Speed vs. Time

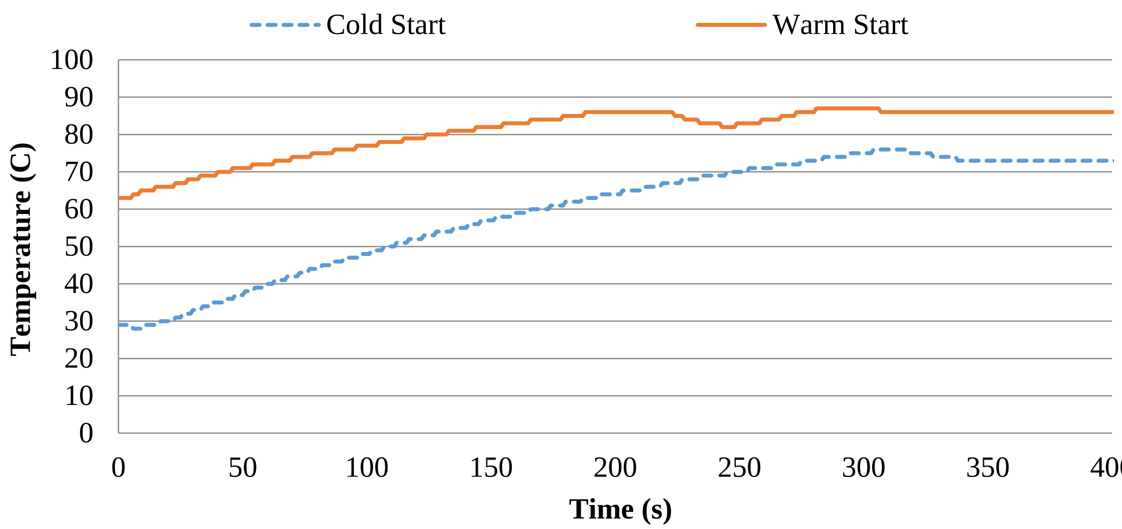

Figure 9-4: Idling Engine Temperature vs. Time

Figure 9-3 shows that the idling speed starts at a higher level (1000 RPM compared to 850 RPM) and Figure 9-4 shows that the engine takes a longer time to reach the steady state speed for a cold engine start than for the warm engine start. The second plot is used as reference to show that there was a difference of more than 30 degrees between the cold and warm engine starts.

9.1.2) EMISSIONS LESSONS

Exhaust THC, CO, CO2, and NOx emissions were collected from the tailpipe during the cold and warm starts and are plotted verses time below in Figure 9-5. The data reveals that there are significant reductions in all four emission types for the warm start compared to the cold start. This knowledge can be used to help refine the control strategy to minimize cold starts to improve the emissions and energy consumption of the vehicle.

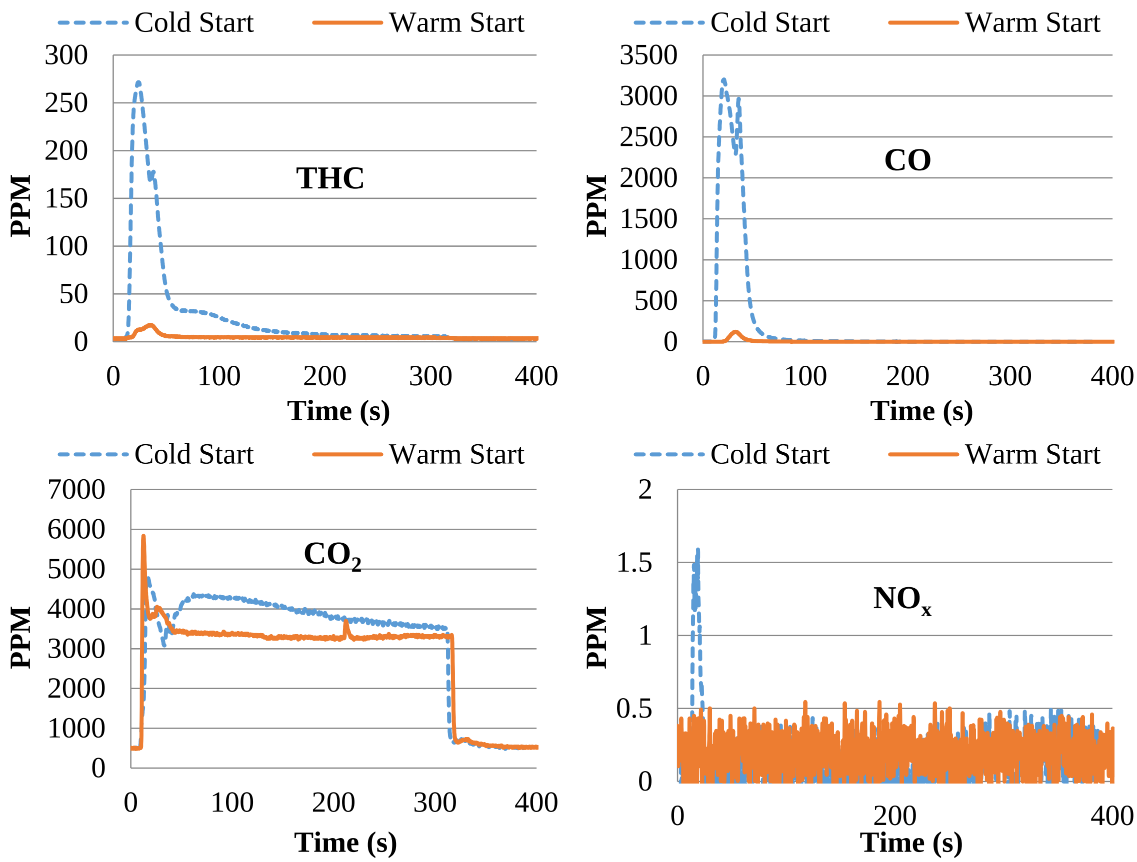

Figure 9-5: Exhaust Concentrations vs Time for THC, CO, CO2, NO_x

9.2) OPTIMIZATION STRATEGIES

Table 9-2 lists the key control parameters that have a leading effect on the vehicle functionality in the three key areas of emissions, performance and energy use. These parameters are grouped by control code subsystem as described previously.

Table 9-2: Key Control Parameters

Subsystem	Key Control Parameter	Vehicle Performance
Diagnostics	Max Engine Coolant Temperature	Emissions
	RTM Start Limiting Temperature	Performance
	RTM Max Temperature	Performance
	RTM Max Torque @ Temperature Limit	Performance
	P2 Max Motor Temperature	Energy Use
	P2 Max Speed	Energy Use
	P2 Speed Rate Limit	Energy Use
	Battery Max Temperature	Energy Use

Subsystem	Key Control Parameter	Vehicle Performance
	Power Electronics Loop Max Inlet Coolant Temperature	Performance
	Regen Braking Max SOC	Energy Use
Selection	Enable Charge Balancing	Energy Use
	Max Charging SOC	Energy Use
	Transition SOC	Energy Use
	SOC Window	Energy Use
	Driver Requested Torque (2-D lookup f(APP & Vehicle Speed))	Performance
	Regen Torque Limit	Energy Use
	Regen Power Limit	Energy Use
	Regen Start Ramp Speed	Energy Use
	Regen Full Torque Speed	Energy Use
	Charge Sustaining – Engine/Generator Speed	Emissions
	Charge Sustaining – Series Mode Output Electrical Power	Energy Use
Execution	Ki – Series Mode Output Elec Power → Engine Torque	Performance
	Kp – Series Mode Output Elec Power → Engine Torque	Performance
	Ki – Engine Torque → APP to ECM	Performance
	Kp – Engine Torque → APP to ECM	Performance
	RTM Torque Down Ramp Rate	Performance
	RTM Torque Up Ramp Rate	Performance

Parameters such as the transition SOC and SOC band are key to the Charge Sustaining strategy whereas the RTM torque rates significantly affect the drivability and performance of the vehicle.

Table 9-3 shows a design of experiment to determine the optimal combination of maximum regen torque and power. This experiment looks at the resulting regen fraction and efficiency of motor mechanical energy to battery internal energy. The test is performed for a range of parameters on a deceleration test. As the table shows, there are a few combinations of parameters which meet the target result and these results can be further refined by looking at other results such as drivability scores.

Table 9-3: Regen Optimization Evaluation (SIL)

Algorithm	Regen Torque Calculation (Req. 1.4.2)				
Test	Braking 60-0 mph in 150 m (approx.. 11 s) (Test 3.3.4)				
Target Result	Regen Fraction, f > 70% Regen Efficiency, e >= 84%				
Regen Torque Limit [Nm]	**Regen Power Limit[kW]**				
	30	**40**	**50**	**60**	**70**

60	f = 48% e = 81%	f = 49% e = 81%	f = 49% e = 81%	f = 49% e = 81%	f = 49% e = 81%
80	f = 56% e = 82%	f = 58% e = 83%	f = 59% e = 83%	f = 59% e = 83%	f = 59% e = 83%
100	f = 61% e = 83%	f = 64% e = 84%	f = 66% e = 84%	f = 66% e = 84%	f = 66% e = 84%
120	f = 65% e = 83%	f = 69% e = 84%	f = 71% e = 84%	f = 73% e = 84%	f = 73% e = 84%
140	f = 68% e = 83%	f = 75% e = 84%	f = 78% e = 84%	f = 79% e = 84%	f = 79% e = 84%

In Table 9-3, the data used to determine the optimal parameters was generated using a model of the vehicle. Both the regen fraction and regen efficiency are values calculated using the following equations.

$$f = \frac{E_{tr}^{-}}{E_{mot,mech}^{-}} \quad \text{Equation 9}$$

$$e = \frac{E_{mot,mech}^{-}}{E_{batt}^{-}} \quad \text{Equation 10}$$

In Equation 9 and Equation 10, E_{tr}^{-} is the negative, or braking, tractive energy, $E_{mot,mech}^{-}$ is the mechanical motor energy during braking, and E_{batt}^{-} is the internal battery energy during braking. All of these values are integrated over the drive cycle during braking events so there is no propulsion power included in this analysis.

While this analysis leaves multiple parameter combinations that meet the initial goals, another metric to take into account when choosing final parameters is the drivability. While the Regen Start Ramp Speed and Regen Full Torque Speed parameters (Figure 5-3 & Table 9-3) are expected to have the greatest effect on drivability, both Regen Torque Limit and Power Limit do affect the feel of the brake pedal and can change the overall drivability of the vehicle. Further exploration of this is left for future work.

10) CONCLUSIONS

Altogether, the development of a Series-Parallel PHEV control system begins with the development of system requirements. These requirements are developed based on overall goals for the vehicle both to meet competition and team requirements. Requirements are also based off of component specifications and interface standards. These requirements are then used to develop the control system and control algorithm.

The VT Malibu control system consists of a distributed control system where a Hybrid Vehicle Supervisory Controller is responsible for interpreting driver demand and commanding powertrain components, each with their own individual controllers, to safely and efficiently meet driver demand. The algorithm running on the HVSC is developed to meet these requirements and the requirements of the competition.

The control algorithm running on the HVSC consists of four primary subsystems: I/O, diagnostics, selection, and execution. Each of these subsystems has a primary purpose and can be developed independently of each other. This structure is highly adapted from the work of King (pg. 6). The selection subsystem contains the algorithm for managing the energy flow throughout the vehicle and has been tested to ensure safe and reliable operation.

These algorithms are tested against a detailed plant model which included all primary powertrain components. Some models were donated, had to be modified, or had to be created from scratch. These models are then validated against component or vehicle test data to ensure that simulations were accurate and beneficial.

Integration of the system into the vehicle has to be carefully planned to meet control system requirements. A CAN bus network is developed with two primary CAN busses for powertrain components, one for safety critical systems, a low speed bus for accessory components, and a mid-speed bus for VT added components.

Vehicle testing was completed at the Emissions Testing Event and was able to provide valuable information to either validate existing control requirements, or show that new requirements were needed. The addition of regenerative braking is shown to reduce energy consumption by almost 10% on a rigorous drive schedule that simulates a combination of city and highway driving conditions. Other tests and simulations are run to help with optimizing the algorithm parameters to meet energy consumption or emissions goals.

Referring back to Table 1-2, the first objective is to develop the requirements for the VT Malibu control system and simulation platforms. This objective is met and detailed in Control System Requirements (pg. 26). Control system requirements are developed for each component in the vehicle and for the vehicle as a whole. These requirements are used heavily in the following steps of the V-cycle for system design, testing, and validation. Simulation platform requirements are detailed for multiple situations: powertrain selection, control algorithm development, and control optimization. Ultimately, Autonomie is selected as the platform for powertrain selection because of the simple user interface and ability to model a wide variety of powertrains. SIL and HIL testing platforms are chosen for both control algorithm development and optimization because of the ability to generate detailed models in Simulink and run that model on dSPACE hardware for HIL testing.

The second objective is only partially met, as a complete vehicle model was never completed for use in SIL and HIL testing. While both platforms are used at various times during the competition to test stock vehicle models and charge depleting models, the full power management strategy was never fully tested due to modeling complications. The control algorithms and vehicle components were tested, however, through other means. Bench testing is very useful in determining exact control interfaces and limitations of components as detailed for the P2 generator motor and the GM engine. In vehicle testing proves that the control algorithm is functional even though it is not subjected to simulation testing.

Third, the data from ETE is analyzed and discussed in detail in TRC Dyno Testing (pg. 72). This data shows that the overall electric energy consumption is decreased heavily with the addition of regen braking. A 10% drop is noticed for the VT Malibu over the course of the competition required CityHighway drive schedule which is used at the final competition. The drop in energy consumption is even more prevalent if braking occurs more frequently, as in the ToTrack cycle. Furthermore, engine operation is explored by looking at data from both cold and hot starts. Hot starts prove to be beneficial across the board in THC, CO, CO2, NO_x as well as the amount of time it takes to reach the target idle speed. This information is useful in tuning the control strategy to minimize emissions during charge sustaining operation.

Lastly, all of the work in this thesis leads towards a functioning control strategy which is implemented into the VT Malibu for the competition. Overall, the topics detailed in this thesis build upon the framework detailed in the referenced literature. The developed control system has been implemented and tested in the VT Malibu and should be able to meet competition targets.

10.1) FUTURE WORK

The work presented in this thesis leaves plenty of room for further exploration. The processes and ideas in this paper could be expanded to additional hybrid powertrains, for example, in the upcoming EcoCAR 3 competition where the team will receive a different platform and likely choose a different powertrain.

Parallel mode was only initially developed and tested in early simulations, but dropped later in the competition because of some issues that were preventing the team from having a vehicle operational in any mode. Further exploration of the implementation of a parallel mode into the control algorithms would be possible if the steps are taken to get a system mechanically ready. This task would take a great deal of work in determining a precise interface with the GM multispeed transmission and the shifting strategy would take a great deal of work to get functional and, furthermore, optimized.

It would also be interesting to judge how this student developed system compares to a control system from a production vehicle. For example, the developed diagnostics of the VT Malibu pales in comparison to the diagnostics present on a production car. While an effort is made to mimic a production diagnostic infrastructure, the resources from companies producing mass market vehicles is much more complex. This is just one area where comparisons could be made to production systems.

REFERENCES

Beine, M., Otterbach, R., and Jungmann, M., "Development of Safety-Critical Software Using Automatic Code Generation," SAE Technical Paper 2004-01-0708, 2004, doi:10.4271/2004-01-0708.

EcoCAR 2 Non Year Specific Rules, Rev N 01/13/2014, Available from Argonne National Labs.

Hanselmann, H., "Hardware-in-the-Loop Simulation as a Standard Approach for Development, Customization, and Production Test," SAE Technical Paper 930207, 1993, doi:10.4271/930207.

King, J., Nelson, D., "Model-Based Design of a Plug-In Hybrid Electric Vehicle Control Strategy," SAE Technical Paper 2013-01-1753, 2013, doi:10.4271/2013-01-1753.

Manning, P., White, E., Alley, R., King, J. et al., "Vehicle System Design Process for a Series-Parallel Plug-in Hybrid Electric Vehicle," SAE Int. J. Alt. Power. 1(2):503-524, 2012, doi:10.4271/2012-01-1774.

Manning, P., White, E., Nelson, D., and Khare, A., "Development of a Plug-In Hybrid Electric Vehicle Control Strategy Employing Software-In-the-Loop Techniques," SAE Technical Paper 2013-01-0160, 2013, doi:10.4271/2013-01-0160.

Pfau, J., “Strategic Use of HIL and SIL,” dSPACE NEWS: Mar. 2006.

UChicago Argonne, LLC. "WELCOME TO AUTONOMIE." Autonomie. http://www.autonomie.net/index.html (accessed April 29, 2014).

Wahlstrom, M., Falcone, F., and Nelson, D., “Integrating Hardware-In-The-Loop Into University Automotive Engineering Programs,” ASEE Technical Paper AC 2012-1409, 2010.

Walsh, P., Nelson, D., "Impact of Supervisory Control on Criteria Tailpipe Emissions for an Extended-Range Electric Vehicle," SAE Technical Paper 2012-01-1193, 2012, doi:10.4271/2012-01-1193.