

Fault Diagnosis and Hardware in the Loop Simulation for the
EcoCAR Project

A Thesis

Presented in Partial Fulfillment of the Requirements for the Degree
Master of Science in the Graduate School of The Ohio State
University

By

John Kruckenberg, B.S.

Graduate Program in Electrical and Computer Engineering

The Ohio State University

2011

Master's Examination Committee:

Dr. Stephen Yurkovich, Advisor

Dr. Giorgio Rizzoni

© Copyright by

John Kruckenberg

2011

Abstract

Technological systems today require increasingly robust and precise electronic controls to deliver improved safety, reliability, and performance while also delivering more numerous and diverse results faster than ever. Additionally, automotive technology is changing rapidly to meet increasingly stringent emissions and fuel economy government regulations while adding new consumer features and capabilities to interface with portable devices such as smart phones and music players. Automotive technology includes advanced vehicle technologies ranging from powerful electric machines to fuel cell systems and battery technologies. In this thesis, modern tools and methods such as hardware in the loop (HIL) simulation, rapid prototyping embedded control systems, and auto code generation are applied to a prototype vehicle design and the results and benefits are discussed.

Hardware in the loop simulation is presented as a powerful tool for control validation because it can be applied to vehicle designs independent of vehicle availability, and the hardware and software tested using the process can be scalable and adaptive to relevant problems throughout the design process. Automotive manufacturers such as General Motors and Ford have been showing increased interest in HIL simulation and its benefits for improving vehicle reliability, safety, and maintenance costs. Controller validation and failure simulation have become increasingly popular uses of HIL

simulation. A much faster design cycle has been a side effect, drawing the attention of other industries.

The work described in this thesis has been applied to the Ohio State EcoCAR vehicle during all three years of the EcoCAR competition. EcoCAR is a student competition among sixteen North American universities in which students design and build advanced powertrain vehicles based on a donated platform to compete on metrics of fuel economy, emissions, performance, and utility. The OSU team demonstrated a refined vehicle using several ambitious designs and components including: (1) the mechanical and control conversion of a compressed natural gas engine to run on E85, (2) a custom dual clutch transmission system allowing both series and parallel (combined) operation similar to the Chevy Volt, and (3) an electrically heated catalyst after-treatment system with secondary air injection for preheating and oxygen storage optimization. Despite the core design challenges, industry practices and tools allowed the team to not only demonstrate a reliable vehicle, but one refined to adhere to modern standards such as electric vehicle charging and industry standard communication networks.

This thesis is dedicated to family and friends. Love and support from my fiancée Leisa Clark, parents Norm and Lois Kruckenberg, and numerous friends and educators made it possible for me to follow my dreams.

Acknowledgments

My research experience at Ohio State would not have been possible without the combined efforts of several great people at the Center for Automotive Research. Support from faculty members, including my advisor Dr. Stephen Yurkovich, Dr. Giorgio Rizzoni, Dr. Shawn Midlam-Mohler, and Dr. Yann Guezennec was invaluable through all my efforts. They provided expert advice on mathematical modeling, control system theory, hybrid electric vehicle designs, batteries, engines, and innovative exhaust aftertreatment systems that prepared me to meet valuable research goals and industry expectations.

Ohio State graduate student project leaders have also taught me a great deal about project management and other fields of engineering. Eric Schacht, Beth Bezaire, Brad Cooley, and Dr. Kerem Koprubasi have done exemplary work on the Ohio State EcoCAR and ChallengeX projects while being exceptional people and teaching me more about the field of engineering than I could ever expect.

Thank you all for your efforts. I could not expect more from my education or the experience I've had over the past six years at Ohio State.

Vita

October 12, 1986	Born - Chillicothe, OH
September 2006	Began involvement in OSU hybrid electric vehicle research projects
November 2009	SAE Panelist on HIL simulation, San Antonio, TX
December 2009	B.S. E.C.E., The Ohio State University
2009-present	Graduate Fellow, The Ohio State University.
August 2010	ASME Conference Presentation, Montreal, QC, Canada

Publications

Research Publications

K. Bayar, B. Bezaire, B. Cooley, **J. Kruckenberg**, E. Schacht, S. Midlam-Mohler, and G. Rizzoni, “Design of an extended-range electric vehicle for the EcoCAR challenge.” *ASME Conference Proceedings*, vol. 2010, no. 44090, pp. 687-700, 2010.

E. J. Schacht, B. Bezaire, B. Cooley, K. Bayar, and **J. Kruckenberg**, “Addressing drivability in an extended range electric vehicle.” Detroit, MI: SAE World Congress and Exhibition, SAE paper no. 2011-01-0911, April 2011.

Fields of Study

Major Field: Electrical and Computer Engineering

Table of Contents

	Page
Abstract	ii
Dedication	iv
Acknowledgments	v
Vita	vi
List of Tables	x
List of Figures	xi
1. Introduction	1
1.1 Rapid Vehicle Development	2
1.2 Fault Diagnosis	3
1.3 Fault-Tolerant Control	3
1.4 Organization of this Thesis	4
2. Hardware and Software	5
2.1 Vehicle Architecture	5
2.2 HIL simulation	9
2.2.1 Useful Applications of HIL for Automotive Testing	11
2.2.2 HIL Methodology	15
2.2.3 Software Execution in Real Time	16
2.2.4 CAN Communication Issues	18
2.2.5 Hardware Fault Simulation and Test Automation	21
2.3 Software Tools	24
2.4 Summary	24

3.	Modeling	26
3.1	Model Fidelity	26
3.2	EcoSIM	27
3.3	Model Implementation for HIL	30
3.4	Model Development Process	31
3.4.1	Battery model	32
3.4.2	Clutch model	36
3.4.3	Summary	41
4.	Fault Diagnosis	42
4.1	Dual clutch transmission system	43
4.1.1	HIL Testing	44
4.1.2	Limit Checks	48
4.2	Converted E85 Engine and Electric Machines	49
4.3	Torque Security	53
4.4	Summary	57
5.	Control Design for Fault Mitigation	58
5.1	Real-Time Control Implementation	59
5.1.1	Controller Hardware Capabilities	59
5.1.2	Model Structure	60
5.2	System Reconfiguration	64
5.3	Power Reduction	71
5.3.1	Series Mode	71
5.3.2	Parallel ECMS	72
5.4	Summary	84
6.	Contributions and Future Work	85
6.1	Thesis Summary and Contributions	85
6.2	Future Work	86
6.3	Conclusions	86
	Appendices	88
A.	List of Symbols and Abbreviations	88

B. DFMEA	90
C. V-Model Design Process	92
Bibliography	94

List of Tables

Table	Page
2.1 Embedded controllers programmed by OSU EcoCAR for vehicle systems	13
2.2 Software tools	24
3.1 HIL model development	31
4.1 Clutch actuator manufacturer data	44
4.2 Clutch actuator motor controller	45
4.3 Dual clutch transmission system limit checks	49
4.4 Comparison of electric machine communication	53
5.1 Powertrain failure mode analysis matrix	66
5.2 Conditions for engine and EM power limiting	71

List of Figures

Figure	Page
1.1 EcoCAR integration time line	2
2.1 Vehicle powertrain components	7
2.2 Vehicle control system	7
2.3 Gear ratios and clutches	8
2.4 Emissions aftertreatment schematic	9
2.5 Comparison of SIL, HIL, and vehicle applications	10
2.6 Components used in HIL testing	12
2.7 EcoCAR vehicle control system diagram	14
2.8 dSPACE MABX turn around time	17
2.9 Real time task overrun effects at mode transition	17
2.10 MotoHawk CPU percent used	18
2.11 Spikes in vehicle speed caused by missed CAN messages	20
2.12 Median filter implementation in Simulink	21
2.13 dSPACE CAN message Rx delta time	22
2.14 Corrected vehicle speed signal	22

2.15	AutomationDesk flow chart	23
2.16	AutomationDesk report	25
3.1	Model fidelity vs computation time	27
3.2	OSU EcoSIM simulator	28
3.3	OSU EcoSIM simulator powertrain	30
3.4	Battery model Thévenin equivalent with constant values	33
3.5	Voltage sagging battery model	33
3.6	Battery parameter dependence on temperature and state of charge . .	34
3.7	Simulated vs modeled battery voltage	35
3.8	High voltage battery schematic	36
3.9	Optimal series resistance	37
3.10	Clutch system Simulink model	38
3.11	Clutch system model validation	40
4.1	(a) Transmission CAD model, (b) Transmission as built	43
4.2	Clutch actuator test rig assembly	46
4.3	Clutch actuator test rig completed and tested on HIL	46
4.4	Clutch actuator measured data in vehicle	47
4.5	Measured current consumption for clutch engagement and disengagement	48
4.6	Clutch engagement time distribution	50
4.7	Clutch engagement and disengagement box plots	51
4.8	Engine rev limiter	52

4.9	Unintended acceleration detection modeled in Simulink	55
4.10	Unintended acceleration detection results from simulation	56
5.1	Fault tolerant control fields of study (from Patton [17])	59
5.2	Top level dSPACE model structure template from dSPACE simulator documentation (EcoCar_HIL_v1p2)	62
5.3	I/O mapping in dSPACE model structure template from dSPACE simulator documentation (EcoCar_HIL_v1p2)	63
5.4	MotoHawk recommended model structure from MotoHawk training supplement	64
5.5	MotoHawk model structure template from MotoHawk default template .mdl file	65
5.6	Fault tree for vehicle powertrain failure	68
5.7	State machine for vehicle operating modes implemented in StateFlow	70
5.8	ECMS simulated actual total torque request vs torque delivered . . .	75
5.9	ECMS total torque with simulated engine failure	76
5.10	Engine temperature input to invoke power limiting operation	78
5.11	Engine power limit percentage for linear power reduction	79
5.12	Engine indicated torque from simulation given oscillating power limit	80
5.13	Engine power limit percentage when over temperature	82
5.14	Engine indicated torque given adjusted power limit hysteresis	83
B.1	DFMEA example: electronic park brake	91
C.1	V-model design process	93

Chapter 1: INTRODUCTION

The motivation of this thesis is to play a part in developing a reliable prototype research vehicle using techniques combined from published research, industry practices, and project needs. The thesis documents the development of fault diagnosis strategies and hardware in the loop simulation efforts over the course of the three year EcoCAR project at The Ohio State University. EcoCAR is a student competition among sixteen North American universities in which students design and build advanced powertrain vehicles based on a donated platform to compete on metrics of fuel economy, emissions, performance, and utility.

The schedule of the EcoCAR competition required rapid integration of components and control systems before the scheduled vehicle operation deadline. A high level schedule of this process is shown in Figure 1.1, where the October 2009 - March 2010 period is highlighted as the main schedule crunch in the process of assembling the OSU EcoCAR vehicle. Off-vehicle simulation efforts and fault diagnosis proved invaluable in order to present a safe and reliable vehicle for competition events that would minimize costly component damage in terms of both time and resources.

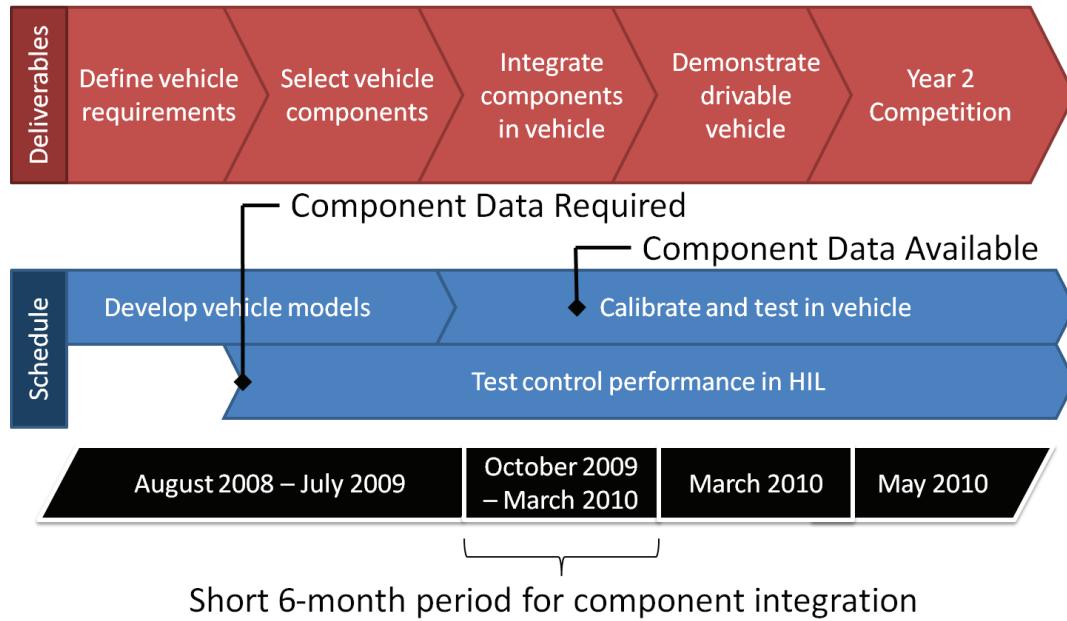


Figure 1.1: EcoCAR integration time line

1.1 Rapid Vehicle Development

Due to rapid growth of technology and the speed of competitive business, there is a huge drive to identify useful vehicle architectures and develop them for testing and marketing. Several techniques are used to develop new vehicle architectures extremely quickly, including modeling and simulation, rapid control development software, and the reuse of existing platforms and systems.

In the process of rapid vehicle development, prototype cost, system availability, and system safety become huge factors determining how quickly and how many tests can be performed. Various simulation processes allow for testing and additional design to proceed before completed systems are available. These include model in the loop

simulation (MIL), software in the loop simulation (SIL), and hardware in the loop (HIL) simulation.

A growing automotive industry challenge is developing a vehicle platform using advanced technology on a tight schedule. In the field of control systems development, this involves rapid control development, fault diagnosis, and fault-tolerant control, among other things. The challenges, approaches, and results associated with these challenges and their application to the EcoCAR challenge vehicle at Ohio State will be discussed.

1.2 Fault Diagnosis

Safety and reliability of vehicle systems rely heavily on strong fault diagnosis and fault-tolerant control strategies, in addition to robust hardware designs. Fault diagnosis is approached in the industry setting through rigorous tasks such as Design Failure Mode Effects Analysis (DFMEA) and Fault Tree Analysis (FTA) to ensure thorough test coverage of high priority failure cases. Fault diagnosis techniques from published research, academic courses, and industry practices were combined in this thesis to provide solutions to important failure modes of the prototype vehicle design.

1.3 Fault-Tolerant Control

Fault tolerant control is a field of control design in which systems are designed to continue the best possible operation in the event of individual or multiple system faults.

Fault tolerant control research currently involves work in three distinct areas [17]: Fault detection and isolation, robust control, and reconfigurable control. All three

of these methods are applied to the EcoCAR vehicle design in order to apply available tools to design a safe control system. Additionally, industry DFMEA and FTA practices from industry are applied to quantify improvements in fault tolerance and failure risk.

1.4 Organization of this Thesis

The rest of this thesis is organized as follows:

Chapter 2 introduces the hardware and software involved in the EcoCAR rapid control development. The architecture of the EcoCAR vehicle, HIL simulation, and applied software tools will be discussed.

Chapter 3 describes different types of models and their applications used in the EcoCAR control development process. This includes a discussion of model fidelity, real-time implementation of models for HIL, and validation of models with vehicle or component testing.

Chapter 4 introduces fault diagnosis concepts and their applications to the EcoCAR control strategy. This thesis covers fault diagnosis of the custom transmission system, the electric machines, and vehicle operation failures.

Chapter 5 discusses applications of fault diagnosis to improving the controls of the vehicle. Fault tolerant control is applied to the supervisory controller's Equivalent Consumption Minimization Strategy (ECMS) to develop a modified strategy to operate robustly and safely in the event of failures.

Chapter 6 summarizes the results of the thesis and gives pointers to future research.

Chapter 2: HARDWARE AND SOFTWARE

Chapter 1 introduced the need to explore vastly different vehicle architectures to face modern challenges. Given this challenge, the Ohio State EcoCAR team designed and constructed a vehicle to meet a set of Vehicle Technical Specifications, including high performance and reliability. This chapter describes the hardware and software used in the vehicle design as well as the control system testing in Hardware in the Loop (HIL) and vehicle testing.

2.1 Vehicle Architecture

The Ohio State EcoCAR is an Extended-Range Electric Vehicle (E-REV) based on a GM crossover vehicle. The major components added to the stock vehicle are shown in Figure 2.1 and consist of:

1. Brusa NLG5 110/240V On-Board Charger (3 kW)
2. A123 Systems LiPO₄ Battery Modules (21.3 kWh)
3. GM EV1 Induction Electric Machine (103 kW)
4. Honda 1.8L CNG Engine converted for dedicated E85 fuel (100 kW)
5. Remy HVH250 Permanent Magnet Electric Machine (82 kW)
6. Custom Transmission with Luk Dual Clutch

These components allow the vehicle to operate in several distinct operating modes, including one defining feature: its parallel (combined) operation mode. This mode allows the onboard engine-generator to directly drive the wheels at highway speeds, allowing for fuel economy improvements up to 20% over an equivalent series hybrid [3]. The complete list of modes includes:

- Charge depleting, all electric: The vehicle can drive about 40 miles at city and highway speeds before turning on the engine for extended range driving.
- Charge sustaining series: The rear electric machine drives the vehicle and the front electric machine and engine operate as a generator to power the vehicle.
- Charge sustaining parallel: The engine and both electric machines are connected to the wheels and provide torque to drive the vehicle
- Charging: The vehicle charges from a 110V or 240V wall outlet and the power-train is disabled (using SAE J1772 charge coupler and communication [9])

Vehicle operation is managed by a network of control modules shown in Figure 2.2. The vehicle platform donated by GM came equipped with the GMLAN and PTE CAN busses, and two more CAN busses were added by necessity of different CAN baud rates and communication standards. The clutch actuators communicate over a 1 Mbps CAN network using a request-response format, while the front electric machine inverter uses a 250 kbps CAN network using the SAE J1939 standard [24]. The dSPACE MicroAutoBox (MABX) programmed by the OSU team serves as the supervisory controller managing the distributed network, which has advantages of centralized authority and local, short-distance communication for actuator control.

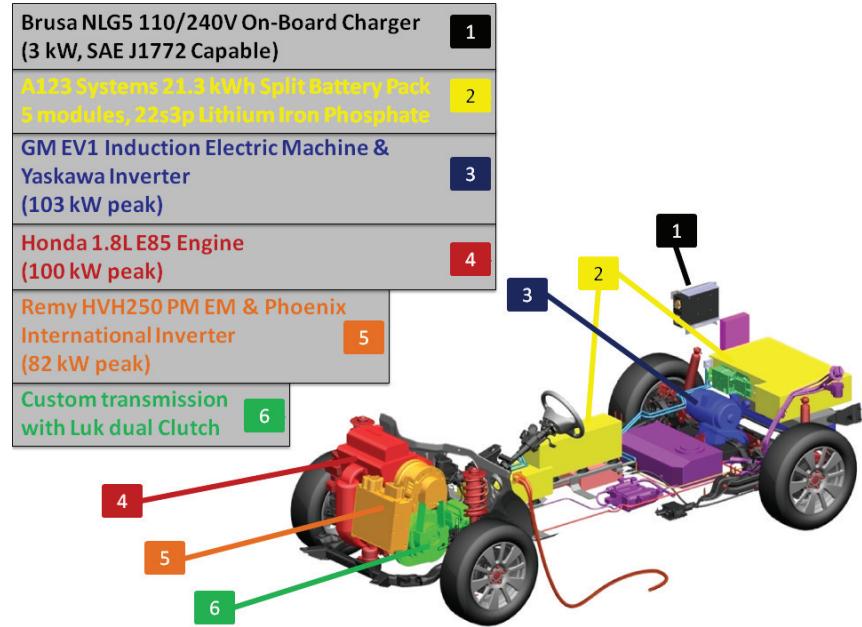


Figure 2.1: Vehicle powertrain components

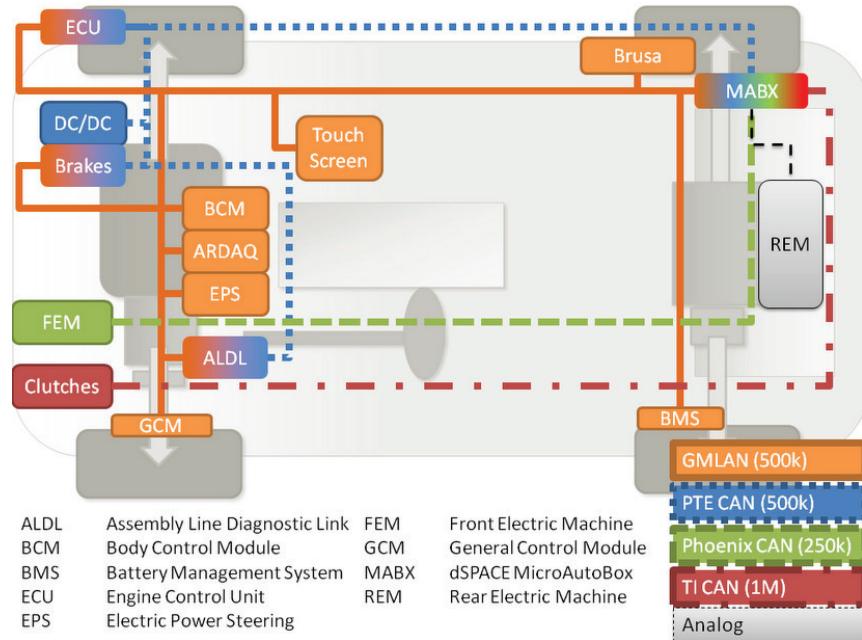


Figure 2.2: Vehicle control system

In the design of the OSU EcoCAR vehicle, some references to the two clutches in the dual clutch transmission system and powertrain shaft speeds are often made, so the diagram in Figure 2.3 may be helpful for some background information. The front electric machine (FEM) and engine are connected by a belt ratio of 1.5:1, and the wheels connect to the FEM through a fixed final drive gear ratio of 2.77:1. Due to this fixed gear ratio, engine power is only delivered directly to the wheels at highway speeds. A pair of electrically actuated dry clutches allow the vehicle to operate the front powertrain in different modes. The engine can be decoupled from the FEM through “Clutch 1.” The FEM can be decoupled from the transmission and wheels through “Clutch 2.”

Gear Ratios and Clutches

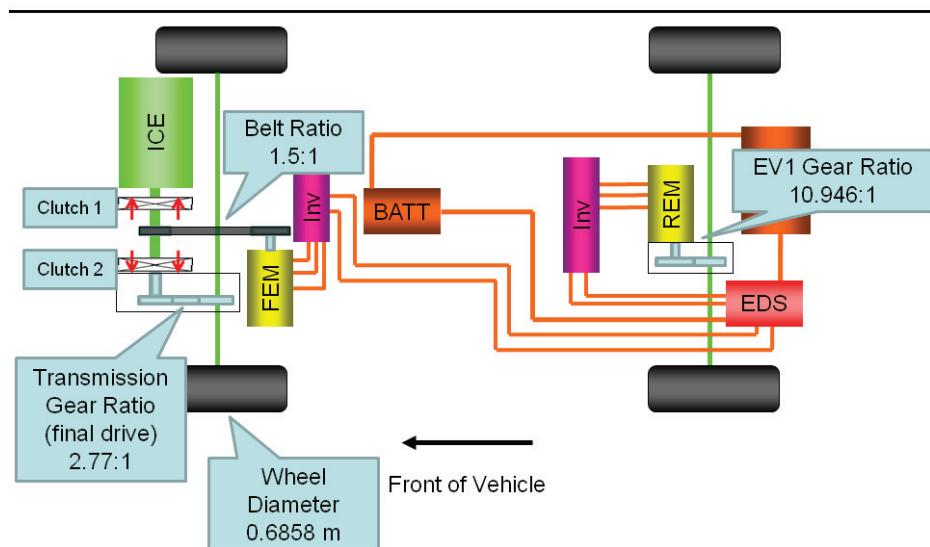


Figure 2.3: Gear ratios and clutches

Additionally, the emissions aftertreatment system is discussed later in Chapter 4 in terms of the fault diagnosis techniques employed due to the importance of emissions in the EcoCAR competition. A schematic of the aftertreatment system is shown in Figure 2.4. This system consists of a 12V electrically heated catalyst, a secondary air injection system, and various sensors used for active feedback control to improve the vehicle's emissions.

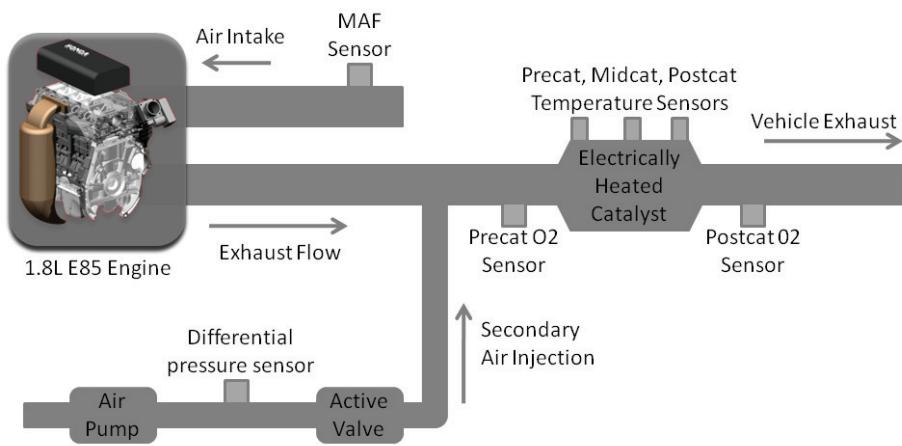


Figure 2.4: Emissions aftertreatment schematic

2.2 HIL simulation

Hardware in the Loop (HIL) simulation is a technique usually used to test a vehicle control system using real hardware and complex plant models together to simulate the feedback control loops [21], although open-loop testing can also be useful [14]. HIL testing is a powerful technique used in the automotive, aerospace, and power

electronics industries to test expensive and safety critical systems before implementation of the control into physical systems. As a visual introduction to HIL simulation in relation to software in the loop (SIL) simulation and real world implementation, Figure 2.5 shows the separation of vehicle controllers, models, and the real vehicle in these three levels of simulation.

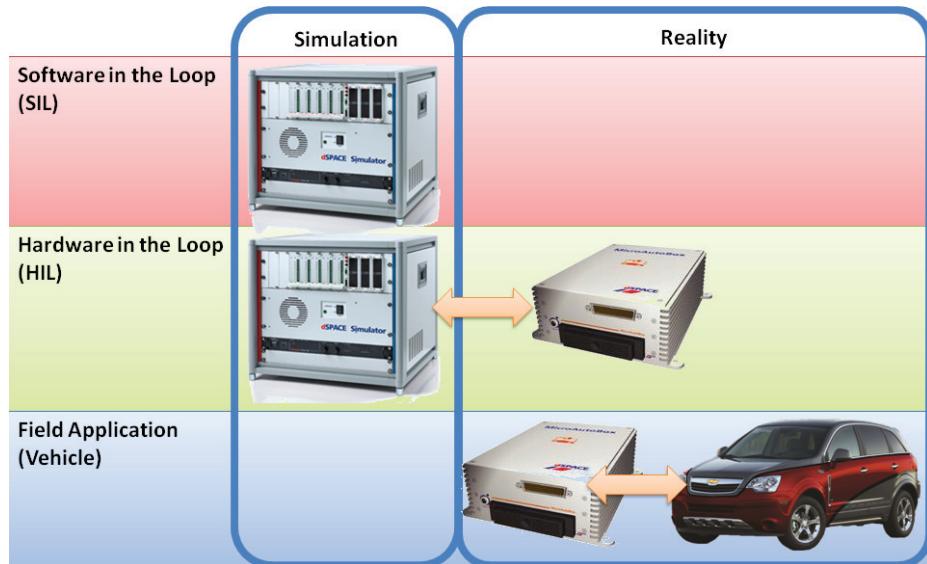


Figure 2.5: Comparison of SIL, HIL, and vehicle applications

In the general sense, HIL simulation is a process that allows for some increased realism in simulation which can be designed to perform useful subsystem testing [6]. The OSU EcoCAR team has implemented test configurations very similar to those used by GM, Ford, and dSPACE [19].

2.2.1 Useful Applications of HIL for Automotive Testing

In the automotive industry, HIL testing is used for design validation and includes investigation of issues such as:

- Systems connected to controllers
- Embedded controller hardware
- Communication networks
 - Lost CAN messages
 - Analog signal jitter
 - Electrical faults
 - Short circuit, Open circuit
 - Over voltage, Under voltage
- Software execution in real time

In the process of testing for these issues, some specialized test algorithms have been designed and demonstrated with some resulting examples that will be shown in the following sections.

The Ohio State EcoCAR project employs a dSPACE midsize HIL unit in order to provide a test bench for software control development and test the I/O functionality of the embedded controllers before testing them in the real plant, which is typically an expensive and safety critical system. The components of the HIL configuration are labeled in Figure 2.6. The core components that allow for HIL testing are the dSPACE Midsize HIL and the HIL interface PC. For simple SIL plant validation, already discussed in Figure 2.5, the midsize HIL acts as an external processor capable of running simulations in real time. The addition of controller hardware such as the dSPACE MABX and Motohawk controllers allows for HIL simulation and controller validation that contributes to many steps of the design process. Hardware such as the electronic throttle body, clutch actuator, and motor controller allow for improved

subsystem control testing in vehicle operating conditions before actually testing on the vehicle. Finally, displays and calibration tools such as laptop interfaces can be used in the same way as they are used on the vehicle for diagnostics, calibration, and data logging to ensure that software is configured properly for efficient use of time on the vehicle.

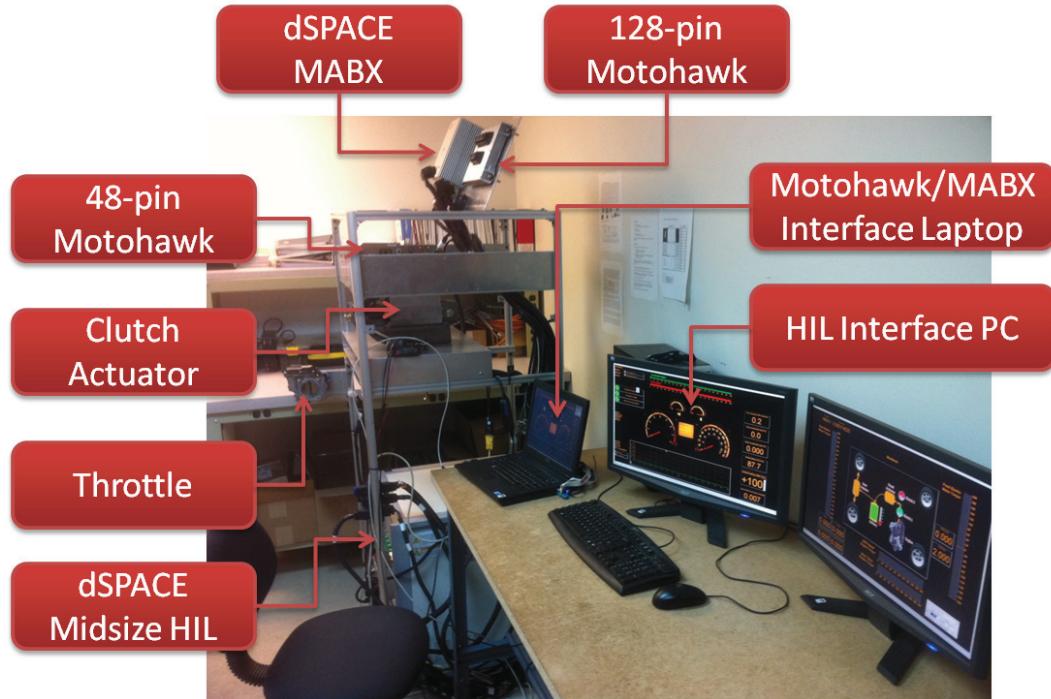


Figure 2.6: Components used in HIL testing

Three embedded controllers are developed for use in the EcoCAR vehicle. The controllers and their applications are listed in Table 2.1. The dSPACE MicroAutoBox is chosen as the supervisory controller due to its superior processing capabilities, but the two Woodward Motohawk controllers are appropriate for engine and cooling

subsystem control due to specialized I/O hardware such as high power driver circuits and crank or camshaft sensor inputs.

In addition to the OSU-programmed controllers, the vehicle control system interfaces with several OEM controllers from the original vehicle (brake controller, body controller, power steering module) and other systems added to the vehicle (battery controller, motor controllers). A diagram showing the modular vehicle architecture is shown in Figure 2.7.

Table 2.1: Embedded controllers programmed by OSU EcoCAR for vehicle systems

			
Supplier	dSPACE	Woodward	Woodward
Controller	MicroAutoBox 1401/1505/1507	ECM-0565-128-L702	ECM-0563-048-0701
Processor	IBM PPC 750FX, 800 MHz	Freescale MPC565, 56 MHz	Freescale MPC563, 40 MHz
RAM	8 MB	548 Kb	32 Kb
Flash	16 MB	1 Mb	512 Kb
Application	Supervisory Controller	Engine Controller	Cooling Systems and Actuators

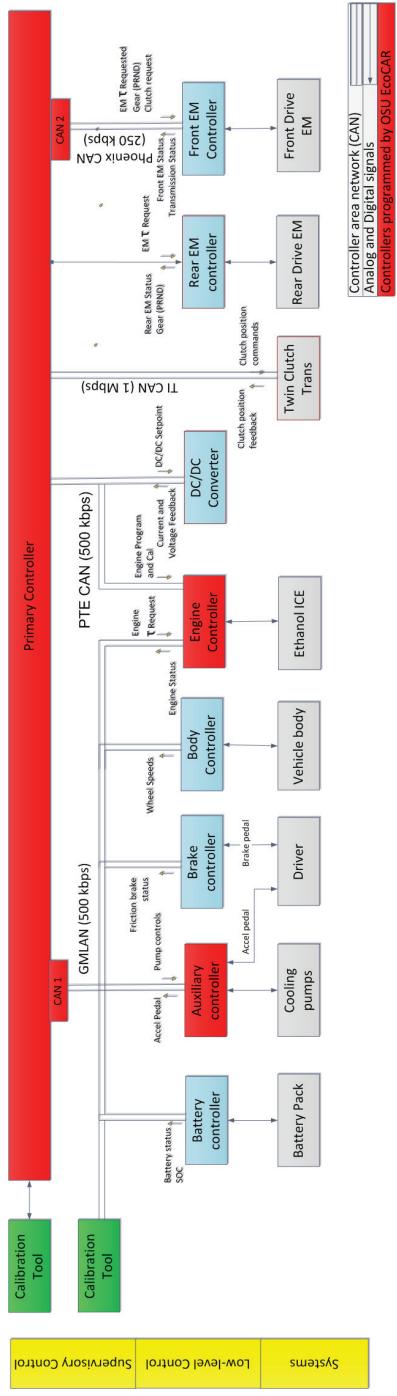


Figure 2.7: EcoCAR vehicle control system diagram

2.2.2 HIL Methodology

Testing and validation are performed at each stage of the V-model design process after the control strategy is designed and implemented (see Appendix C for detail about the OSU EcoCAR design process). Software in the loop (SIL), HIL, and vehicle testing are employed in parallel in order to take advantage of the benefits of each testing method. For example, some re-design of energy management strategies can be reviewed in SIL even when the HIL and vehicle systems are available due to reduced hardware risks and increased simulation speeds. The HIL system is the preferred choice for failure mode testing due to the possibilities of test scripting and hardware fault insertion. Vehicle testing and validation was performed for the driver interface, E-stops, battery, and powertrain to the extent that their vehicle integration is complete.

The dSPACE HIL hardware simulator allows the team to simulate hardware using real analog, digital, PWM, or CAN signals to test all possible inputs to the vehicle controllers. In addition, real hardware such as of the accelerator and brake pedals can be connected to the HIL such that the vehicle controller can be controlled by either real driver inputs or simulated signals distinguished only by a switch in the ControlDesk software.

The control logic was validated and optimized for the most realistic situations possible before in-vehicle tests. Because off-vehicle design and calibration can expedite in-vehicle testing and reliability, thorough exploration of a large range of tests and rigorous SIL and HIL applications are invaluable tools for the design process. In addition to control logic, controller I/O was also validated in HIL simulations, allowing for faster integration in the vehicle and more tests of failure mode cases.

For vehicle validation and testing, recorded data was compared with HIL data in order to validate models and improve model fidelity as needed. An example of this comparison is described in the next section and shown in Figure 2.8.

2.2.3 Software Execution in Real Time

Software process execution time can be tested in HIL simulation since the real controller hardware is used. The dSPACE and Woodward controllers monitor the CPU loading during normal operation, and the respective interface tools can measure and log the CPU usage while running various processes. Figure 2.8 shows a screenshot from dSPACE ControlDesk showing displays and plots of the dSPACE MicroAutoBox controller’s task execution performance. dSPACE displays the software “Sample Time” and the “TurnAroundTime,” the time needed to execute one sample step of the software at a certain time. If the “TurnAroundTime” is approaching or exceeding the “Sample Time,” unexpected behavior may result. From the dSPACE documentation, a likely result is failing to execute certain time-consuming functions to allow other processes to execute properly at the next sample step.

In the first year of the EcoCAR challenge, running control algorithms in real-time became an issue in HIL implementation. Figure 2.9 shows a comparison of SIL and HIL versions of the same model. The large difference between these two models was caused by a task overrun in the MABX controller, resulting in failure of the torque split optimization algorithm to execute in the vehicle’s parallel drive mode.

Similar CPU loading issues are relevant for the vehicle’s two MotoHawk controllers as well. Figure 2.10 shows the Motohawk ECU’s “CPU Percent Used” as measured through real-time calibration with Etas Inca. During high speed operation while using the controller’s CCP (CAN calibration protocol) interface. Special care is taken to ensure that the CPU is not overloaded to prevent undesired engine operation.

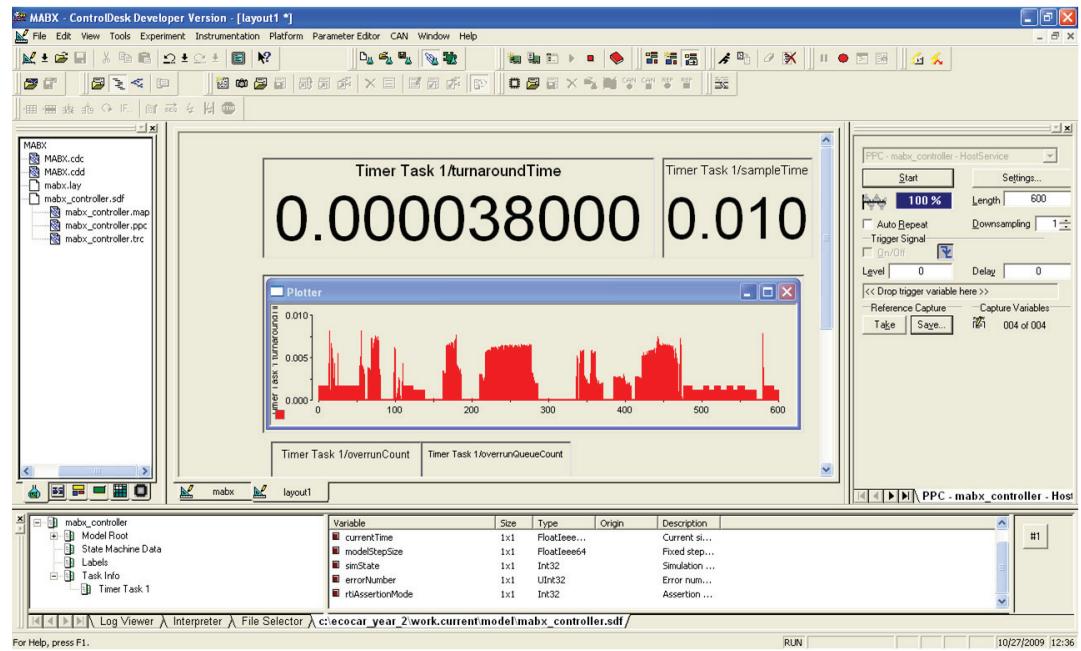


Figure 2.8: dSPACE MABX turn around time

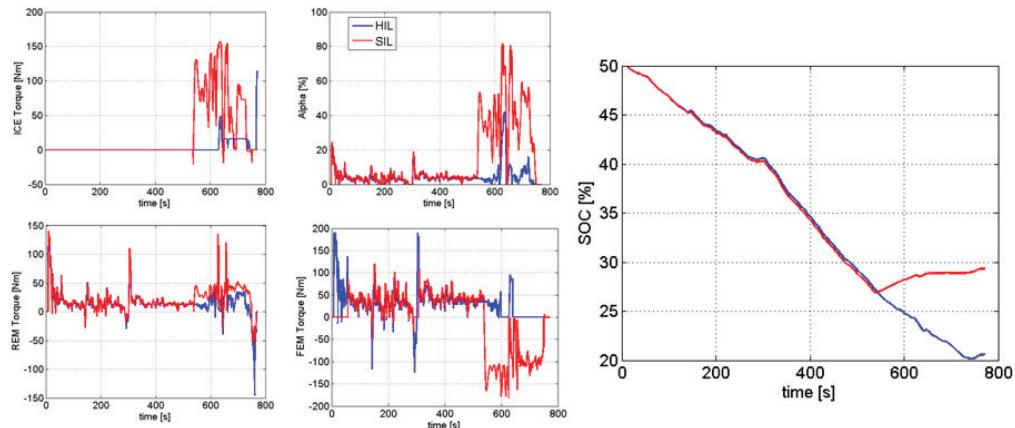


Figure 2.9: Real time task overrun effects at mode transition

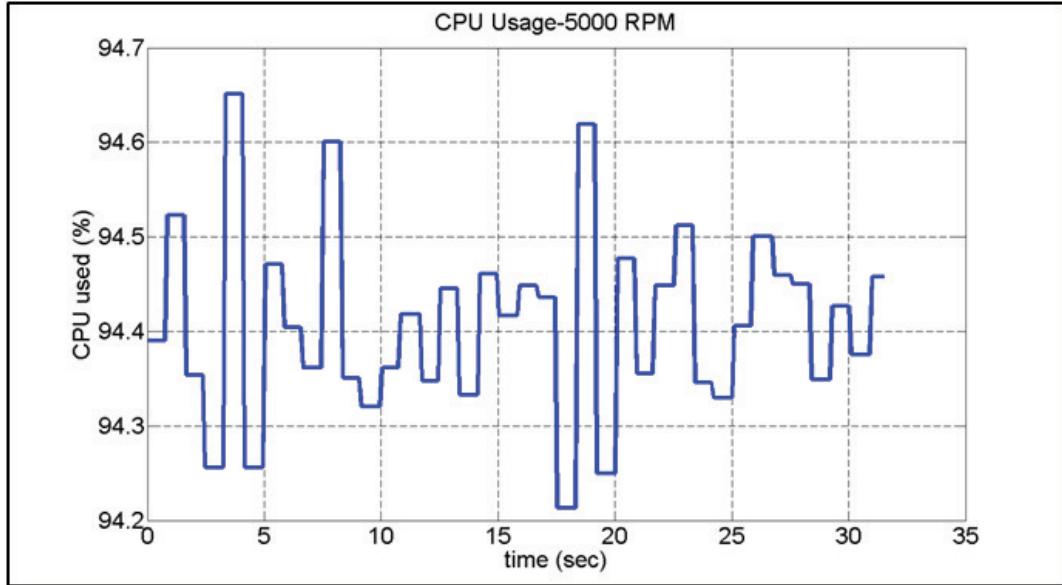


Figure 2.10: MotoHawk CPU percent used

2.2.4 CAN Communication Issues

A large number of communication issues can be designed and validated using HIL simulation techniques. One specific issue discovered in vehicle data logs and resolved with the HIL system involved missing CAN messages during heavy usage. Missing CAN messages result in some inconsistent delay issues, but a specific issue also resulted in the vehicle experiencing some torque delivery issues.

Figure 2.11 (top) shows a vehicle speed measurement derived by taking the derivative of wheel encoder pulse counts. Periodic speed impulses appear in the speed measurement when the CAN message for the wheel encoder pulse counts is missed or delayed due to heavy messaging traffic. This happens because the real-time controller uses the last available message data. When a message is missed, the controller assumes the wheel speed has suddenly changed to zero, followed by an increase in

speed when messages are received again. Even with linear filtering, this result is highly undesirable because:

- The wheel speed impulse looks like clutch slip because the wheel speed differs from the front motor speed. This results in the controller reducing front axle torque, as shown in Figure 2.11.
- The change in measured speed results in different maximum torque requests for the powertrain components. This affects the total capability for torque in the vehicle, and can affect the mapping of the accelerator pedal to actual torque output.
- The change in measured speed affects the most efficient torque split for the ECMS strategy. Different torque distributions may be chosen due to an erroneous speed.
- Power steering and brake boosting feel is adjusted based on vehicle speed reported on this CAN message.

In practice, these errors are all small. However, they remain a high priority since they affect torque delivery in the vehicle. Various algorithms for using the dSPACE block's data indicating new messages and message receive times were attempted, but a small window median filter was more reliable for consistent operation. The median filter method was chosen for its repeatability and effectiveness at completely removing out-of-range measurements despite the short delay added by the filter. Considerations for this filter design include:

- Out of range noise is filtered out if its length is less than half the window size.
- A delay of half the window size may be applied before the effect of the input appears in the output.

- The sorting algorithm used in finding the median may become computationally intensive for a large number of median filters or large window sizes.

Figure 2.12 shows the implementation of a median filter to improve wheel speed measurements.

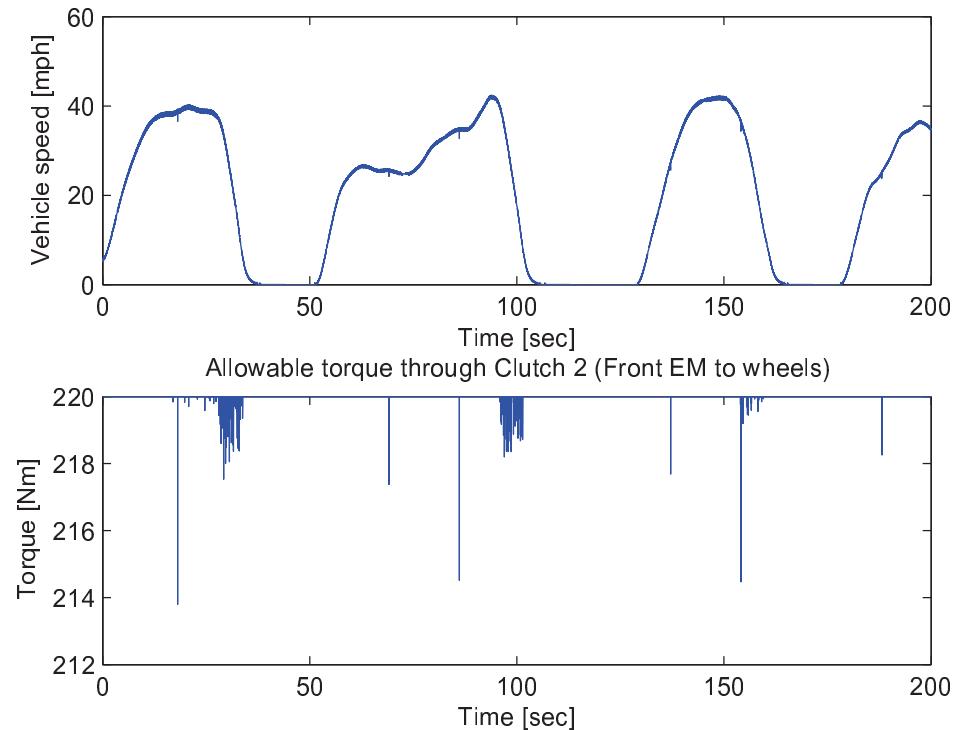


Figure 2.11: Spikes in vehicle speed caused by missed CAN messages

Figure 2.13 shows the erroneous vehicle speed measurement as modeled on the HIL system in order to test control algorithms to mitigate the effects of missed messages.

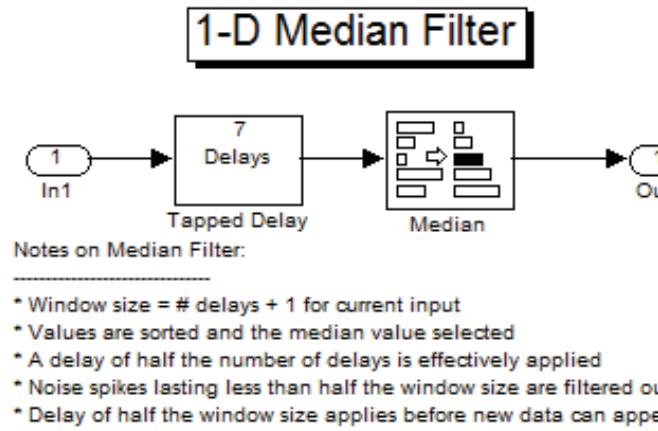


Figure 2.12: Median filter implementation in Simulink

Figure 2.14 shows the result of an algorithm to reduce these effects. The median filter with a small window size of eight elements was chosen for the best performance and acceptable delay effects.

2.2.5 Hardware Fault Simulation and Test Automation

Hardware failure and test automation are very powerful tools for HIL systems. Implementation with real hardware allows for testing of hardware I/O and timing, under-voltage responses, and signal detection algorithms for shorts to ground or supply voltages. Test automation provided by the dSPACE AutomationDesk software, shown in Figure 2.15, allows for a flowchart style design environment for implementing tests that can be performed on successive versions of software to ensure that important design criteria are met. The results of this automated testing can be automatically documented in reports, as shown in Figure 2.16.

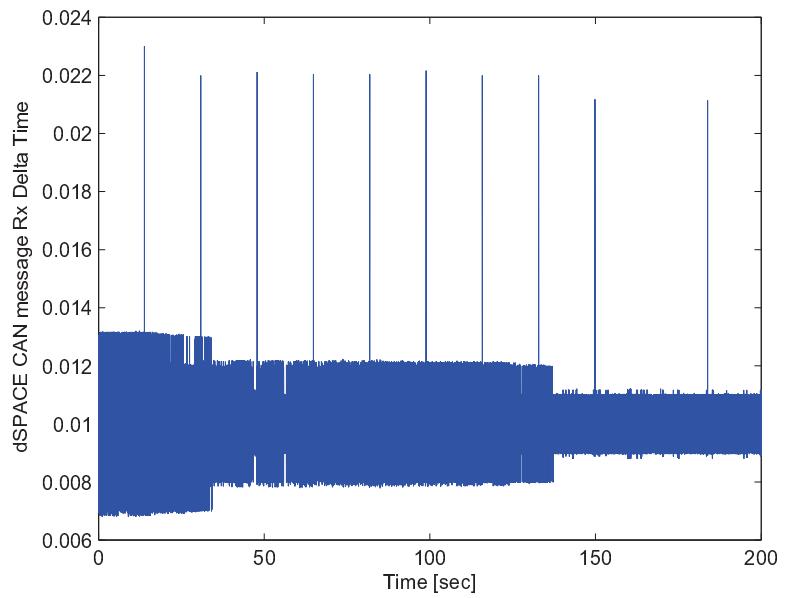


Figure 2.13: dSPACE CAN message Rx delta time

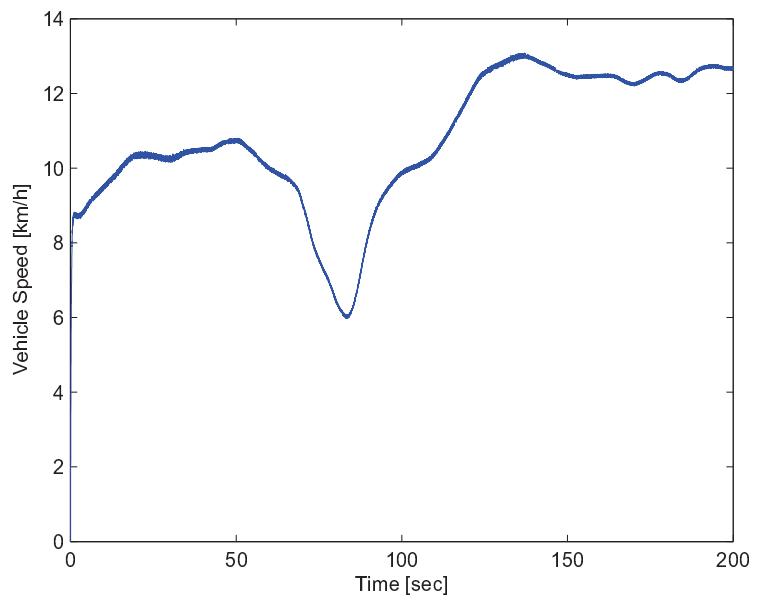


Figure 2.14: Corrected vehicle speed signal

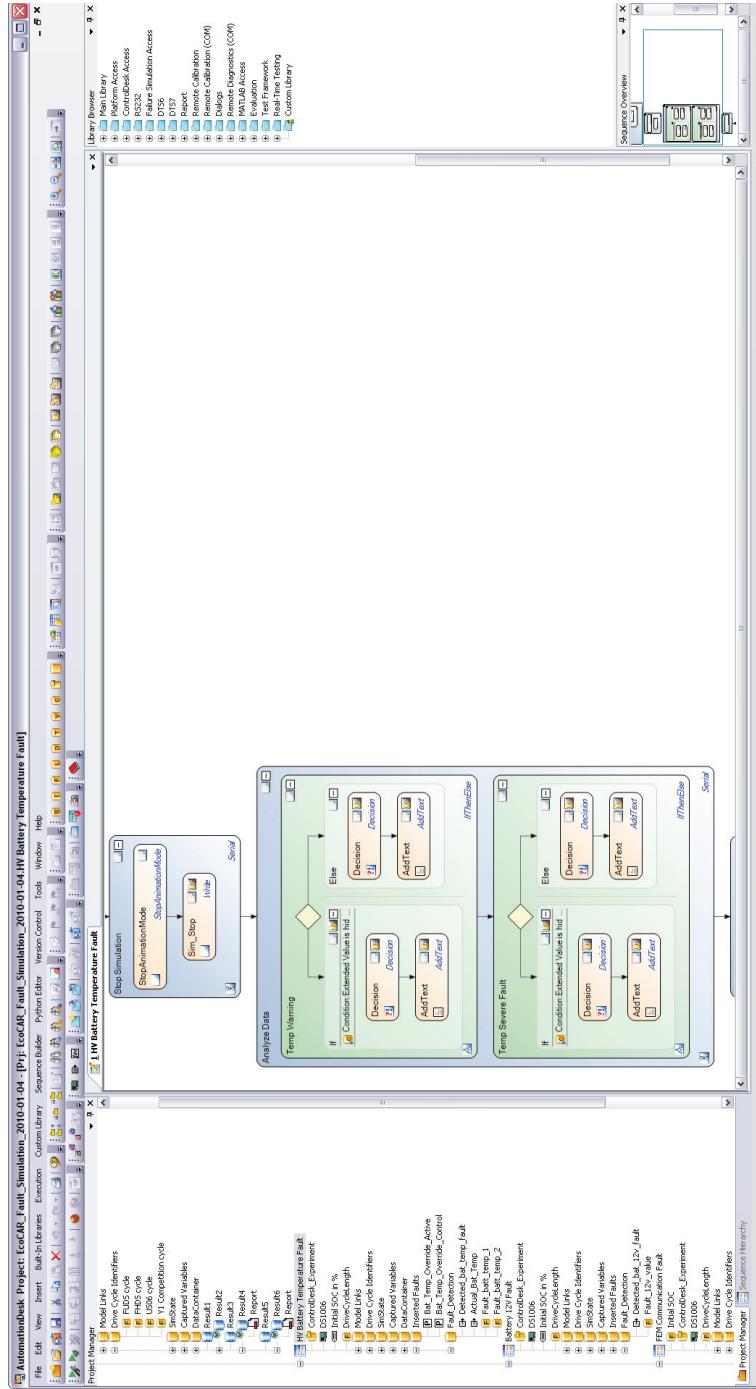


Figure 2.15: AutomationDesk flow chart

2.3 Software Tools

A number of software tools make HIL simulation and controls development possible for the selection of hardware used on the project. Because several software suites exist for similar purposes, the list in Table 2.2 is provided for context of the tools used in this particular application.

Table 2.2: Software tools

Developer	Title	Description
dSPACE	AutomationDesk	HIL Test Automation
dSPACE	ControlDesk	Real-Time Calibration and Display
ETAS	INCA	Real-time Calibration and Display
Woodward	MotoTune	Real-Time Calibration and Display
KVaser	CanKing	CAN network PC Interface
Mathworks	Matlab/Simulink	Modeling, Simulation, and Code Generation
Microsoft	Visual Studio	PC Displays and Interfaces
Tigris	Tortoise SVN	Version control software
Vector	CANape	CAN network PC Interface
Vector	CANoe	CAN network interface and diagnostics

2.4 Summary

This chapter has provided an overview of hardware and software used in the EcoCAR project and HIL simulation as a foundation for the rest of the development of this thesis. An overview of the EcoCAR vehicle design, HIL system design, and some demonstrations of HIL system capabilities have been shown. In Chapter 3, the EcoCAR modeling capabilities will be demonstrated, including the types of simulations that can be performed in HIL.

HV Battery Temperature Fault (Sequence)			
Start time:	2010/01/04 19-28-31	Stop time:	2010/01/04 19-29-30
Execution duration:	58.437 sec.	Name:	HV Battery Temperature Fault
Library link:		Creation date:	2009/11/17, 10-54-34
Modification date:	2010/1/4, 19-29-30	Author:	EcoCAR
Hierarchy:	EcoCAR_Fault_Simulation_2010-01-04.. HV Battery Temperature Fault		
Result state:	Passed		

Battery high temperature warning detected.

Battery high temperature severe fault detected.

Capture Result

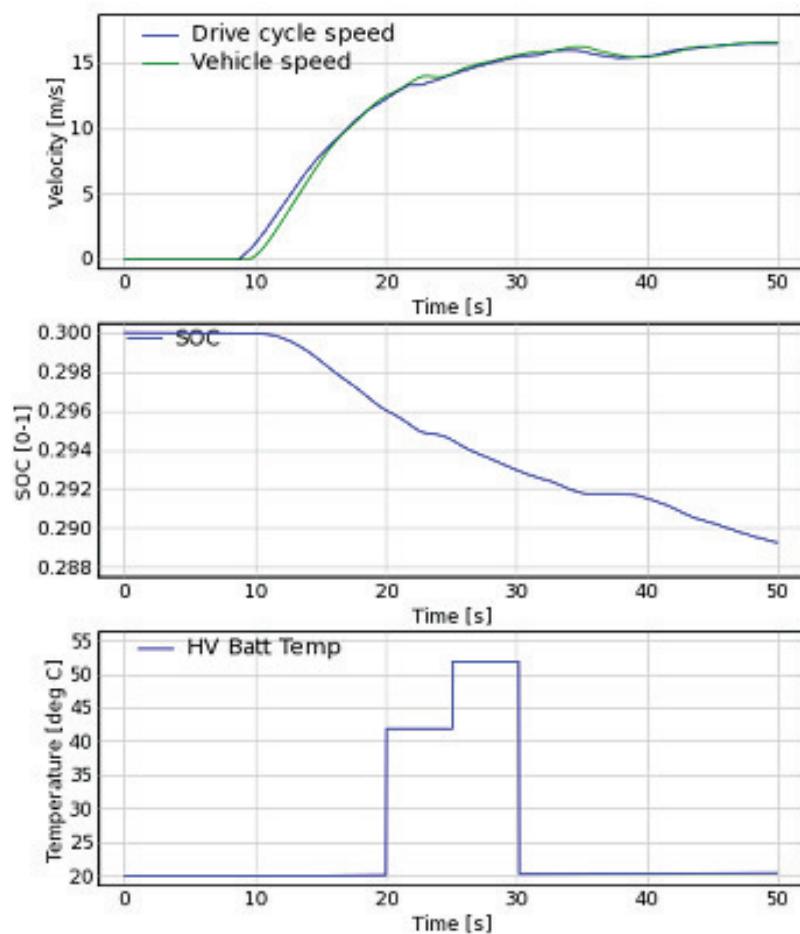


Figure 2.16: AutomationDesk report

Chapter 3: MODELING

Mathematical models for different development purposes require varying levels of accuracy for reproducing the behavior and states of real world systems. The EcoCAR modeling efforts progressed from simplified models to perform range and performance calculations up to higher fidelity mean value engine models and electrical simulations as the development process progressed. Hardware in the loop simulation requires certain limitations and additional efforts to perform tests using a variety of models. Hardware and feasibility constraints involved in HIL simulation were introduced in Chapter 2, and these play a role in the hardware and model development used in simulation.

3.1 Model Fidelity

It is always very important to keep in mind the purpose of a model being developed or applied. The least complex model that will accomplish the design goal is desirable because it requires less simulation time, development time, and test data. Since the main functions of the EcoCAR models involve longitudinal dynamic simulation of torque split functionality, energy analysis, component interactions, and fault diagnosis, the most-used models for control include the minimal high frequency dynamics and effects required for the embedded controllers to perform their functions. Higher fidelity models are used offline for more accurate simulation, including high voltage

system analysis using SimPowerSystems tools from the MathWorks and high fidelity engine modeling packages such as GT Power for engine development. A diagram of model computation time vs. fidelity is shown in Figure 3.1. As a reference point, mean value and black box models are usually very capable of running in real-time simulation, but more complex plant models are usually left for offline simulation.

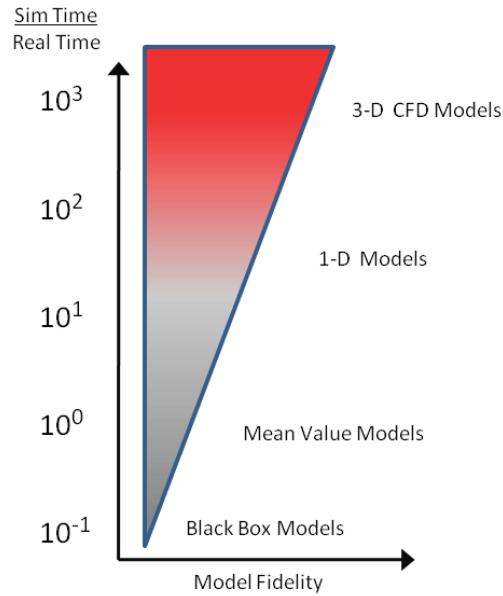


Figure 3.1: Model fidelity vs computation time

3.2 EcoSIM

The Ohio State team developed a torque feed forward, speed feedback longitudinal dynamics simulator for the design and analysis of the EcoCAR vehicle, developed in Matlab Simulink. The simulator is a Model in the Loop (MIL) simulator usually run on a PC, but can also be tested in SIL by running on the dSPACE Simulator.

The EcoSIM simulator is the starting point for HIL simulations as well, allowing for comparisons and validation to ensure that the general operation of the simulator remains the same when running on hardware.

This simulator is updated throughout the vehicle design process, keeping the same structure such that the control software is exactly the same as that which is loaded on vehicle controllers, and plant models are exactly the same as the HIL system models.

A screenshot of EcoSIM is shown in Figure 3.2. The simulator is designed such that the supervisory control algorithm usually loaded on the dSPACE MicroAutoBox (MABX) is the same as the EcoSIM’s “MABX Primary Controller” subsystem.

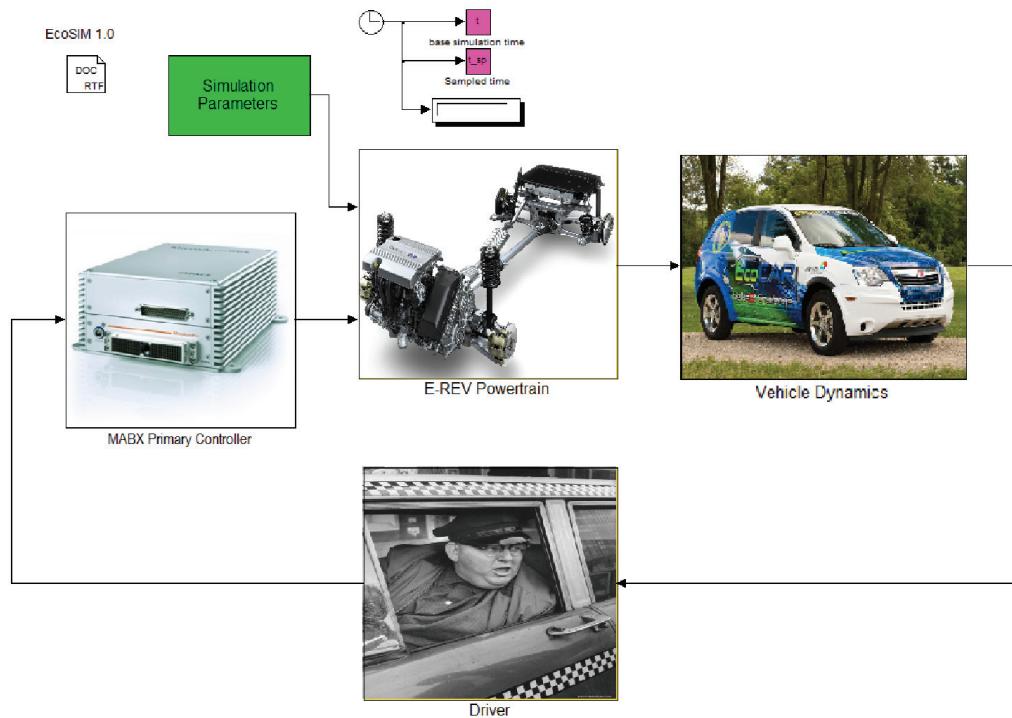


Figure 3.2: OSU EcoSIM simulator

The component plant models inside the “E-REV Powertrain” subsystem include the vehicle’s torque production and energy storage components, as shown in Figure 3.3. Later in this chapter, some development of the Clutch and High-Voltage Battery models will be discussed in more detail.

The powertrain models for the engine and electric machines consist of lumped inertial dynamics in addition to energy consumption lookup tables for each torque-speed operating point, similar to several previous OSU projects [12] [10] [16]. This type of model allows for longitudinal dynamics models to easily simulate energy consumption based on torque required at the wheels in the form of a quasi-static simulator [11].

Torque and speed of the engine and electric machines can be modeled using a first order approximations [8] including torque production, given as

$$\dot{T}_i(t) = \frac{T_{i,req}(t - \delta) - T_i(t - \delta)}{\tau_{i,m}} \quad (3.1)$$

where $i \in \{FEM, REM, ICE\}$, δ is the simulation time step, and $\tau_{i,m}$ is the first order time constant of the i^{th} powertrain component in operating mode m .

Lumped inertia models of the powertrain components, shafts, and associated components which also discretely change based on the vehicle operating mode are modeled using

$$\dot{\omega}_i(t) = \frac{T_{i,req}(t - \delta) - T_i(t)}{J_{i,m}} \quad (3.2)$$

with mode and component-dependent rotating inertia $J_{i,m}$. Once the modeled torque and speed are calculated, two dimensional lookup tables are used to estimate electrical and fuel energy consumption at each torque and speed operating point.

Aerodynamic drag, rolling resistance, and grade resistance are considered separately in the “Vehicle Dynamics” subsystem, which are shown in more detail in Chapter 4. The “Driver” subsystem includes a PID control and the accelerator, brake, and

shift lever driver interface inputs to the controller. The simulator uses standard drive cycle traces such as the US HWFET, UDDS, and US06 cycles [5], as well as custom competition drive cycles and drive traces from recorded vehicle data.

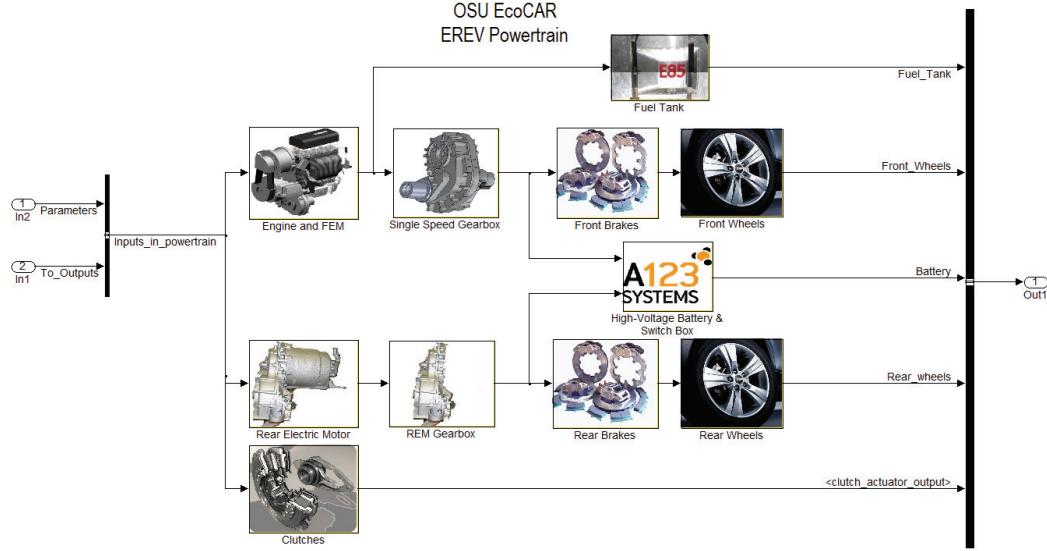


Figure 3.3: OSU EcoSIM simulator powertrain

3.3 Model Implementation for HIL

Notable results of HIL testing for the EcoCAR team include testing component interactions and hardware capabilities. Reliable interaction among components, including various controllers, is vital for proper fault diagnosis and robust vehicle safety and reliability. Hardware limitations such as signal conversion resolution, delay, CAN message bandwidth, and real-time processing capabilities are also verified in HIL simulation. The process for changing models and HIL system hardware is outlined in Table 3.1, where the changes to the controller, plant, and models are described in

comparison to the vehicle development time line through three years, Y1-Y3. Initial simulations in Y1 began using a single vehicle controller, the dSPACE MicroAutoBox, and the dSPACE midsize HIL simulator with the ds1006 processor board and ds2202 I/O board hardware. After initial SIL tests, CAN, analog, and digital hardware was added to perform HIL testing. Later significant developments adding additional model fidelity as control algorithms become more detailed. Major developments involved engine control systems [4] such as the electronic throttle, injectors, crank and camshaft signals using the dSPACE ds2210 I/O board, load cards, and Motohawk ECU.

Table 3.1: HIL model development

	Controller	Plant	Models
Y1	HIL ds1006 / ds2202	HIL ds1006 / ds2202	Energy analysis models: powertrain and battery information based on scaled data from known hardware
Y1	MicroAutoBox	MicroAutoBox	Initial proportional torque splits and fast mode transitions
Y1	MicroAutoBox	HIL ds1006 / ds2202	Closed loop control using CAN, analog, and digital I/O
Y1	MicroAutoBox Engine ECU	Added HIL ds2210, throttle, injectors, pedals	Mean value engine model based on dSPACE ASM, Equivalent Consumption Minimization Strategy (ECMS), updated component information from manufacturer data
Y2	MicroAutoBox Engine ECU	Added displays, calibration and diagnostic tools	Vehicle startup process CAN messages, displays, shutdown, clutch actuation models, fault detection and test automation
Y3	MicroAutoBox Engine ECU Cooling GCM	Added clutch actuator and motor controller	Updated models to use all 4 vehicle CAN networks, testing SAE J1939 CAN messaging for front electric machine and response/request 1 Mbps CAN network for clutch drivers

3.4 Model Development Process

The OSU EcoCAR project's model development process involved predictive modeling which began before real components and systems were available for testing

and validation. Simplified models based on data scaled from similar components allowed for testing of simple component interaction and high level energy analysis of the vehicle in the early development process, but calibration of energy minimization strategies, emissions control, and vehicle noise, vibration, and handling cannot yet be considered.

To perform model validation, individual model subsystems were separated and studied using recorded data from the vehicle. Ultimately, validation of the entire vehicle model to ensure correct performance is the goal, but each subsystem must also be functioning properly.

3.4.1 Battery model

The purpose of battery modeling changes as the vehicle design detail increases. Initial battery models focus on the energy storage capacity of the battery system. Later models focus on the voltage swing during transients, thermal effects, fault detection, and communication.

During the EcoCAR proposal process [2] and initial modeling for component selection, a very simple battery model is used, as shown in Figure 3.4. The simplified model is a Thévenin equivalent circuit model, the advantage of which being that assumptions about cell chemistry, packaging considerations, or manufacturer data that are not typically required at early stages of development. The model in Simulink is detailed enough to represent battery losses and capacity limits needed for component sizing and selection.

Once battery components are selected, a more advanced model can be used to perform additional design steps. The voltage sagging model [23] shown in Figure 3.5 represents complex internal battery impedance and electrolyte transfer dynamics, and the Randle circuit values R_1 , R_2 , C , and the open circuit voltage V_{oc} vary based

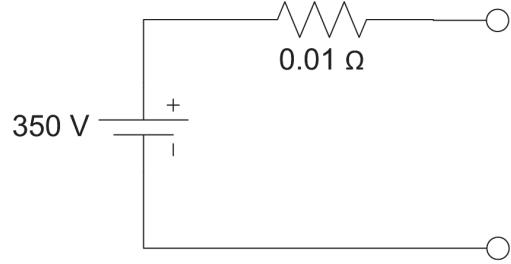


Figure 3.4: Battery model Thévenin equivalent with constant values

on temperature, state of charge, and whether the cells are charging or discharging. These parameters are modeled using two dimensional lookup table data that was fit to battery manufacturer specifications. The Simulink model of this parameter dependence is shown in Figure 3.6.

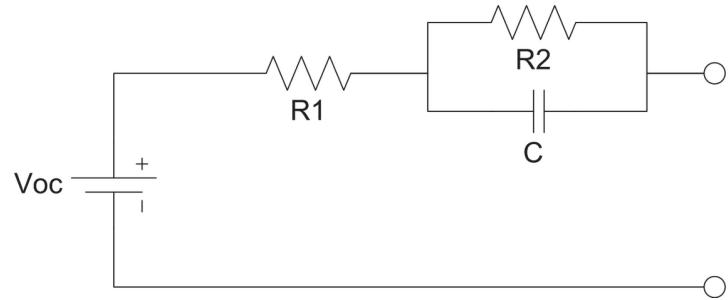


Figure 3.5: Voltage sagging battery model

Accurate modeling of the battery's terminal voltage is useful to ensure proper wire sizing and fusing, and appropriate component voltage limits. To validate that modeled battery voltage represents the actual battery as implemented in the vehicle, the battery model outputs are compared to recorded data from a vehicle drive cycle. Recorded battery current and initial SOC data from the vehicle are used to ensure

BATTERY PARAMETER ESTIMATION

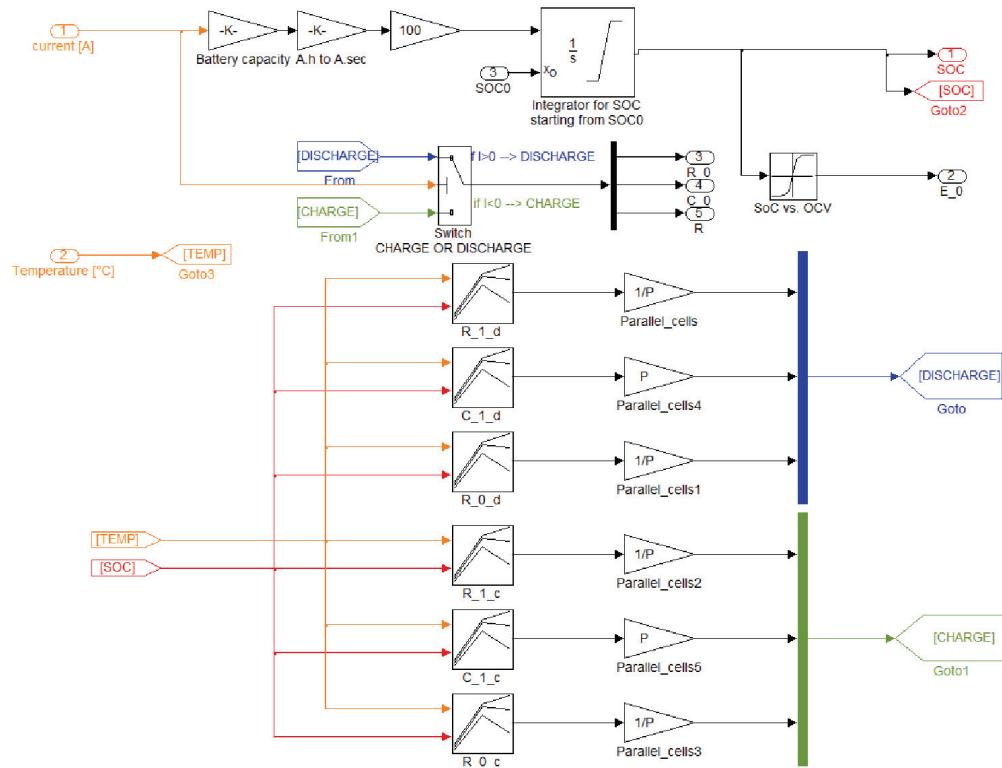


Figure 3.6: Battery parameter dependence on temperature and state of charge

that the same load conditions are replicated in the model and the output voltage of the model is compared to measured battery voltage. Initial comparisons using only manufacturer data showed very similar performance, as shown in Figure 3.7. By inspection of the figure, the modeled voltage swings too high when charging and too low when discharging. This indicates an incorrect real part of the battery internal resistance, corresponding to incorrect values for wire resistance between modules in the split battery pack configuration. As shown in the high voltage battery schematic in Figure 3.8. The series battery resistance includes 10 meters of 50 mm² cable as well as contactors, fuses, service disconnects, and internal pack connections in the vehicle’s high voltage electrical system. The vehicle’s split battery pack design was chosen because the packaging was minimally invasive into the passenger and cargo areas and provided a good front to rear weight distribution in the vehicle, despite increased complexity for cable and cooling wiring.

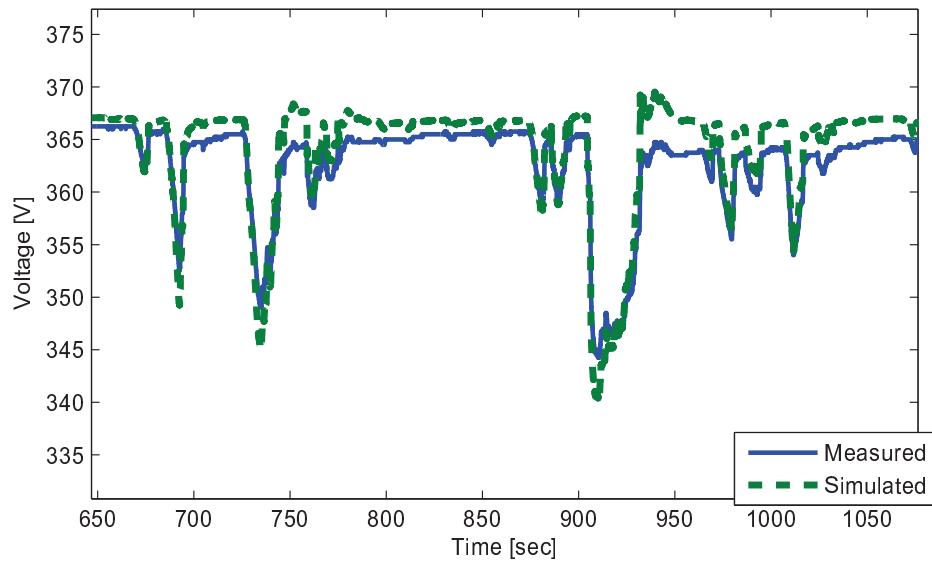


Figure 3.7: Simulated vs modeled battery voltage

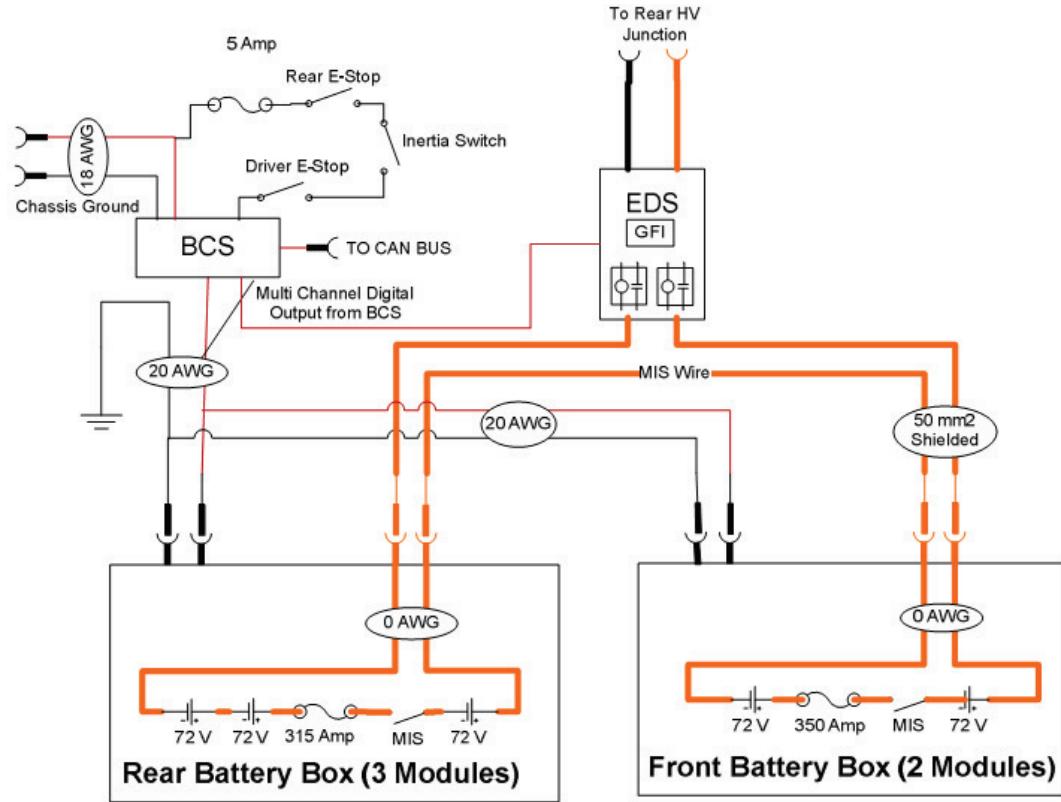


Figure 3.8: High voltage battery schematic

Figure 3.9 shows the RMS error minimization used to improve the model accuracy and validate its effectiveness. By changing only the series resistance, the RMS error between simulated and measured vehicle data can be minimized to 4 volts, which is acceptable for many simulation efforts. The optimal series resistance value was found to be 0.22 ohms.

3.4.2 Clutch model

The dual clutch system consists of the Luk clutch package, linear actuators and lever arms containing DC motors and hardware error checking, DC motor controllers

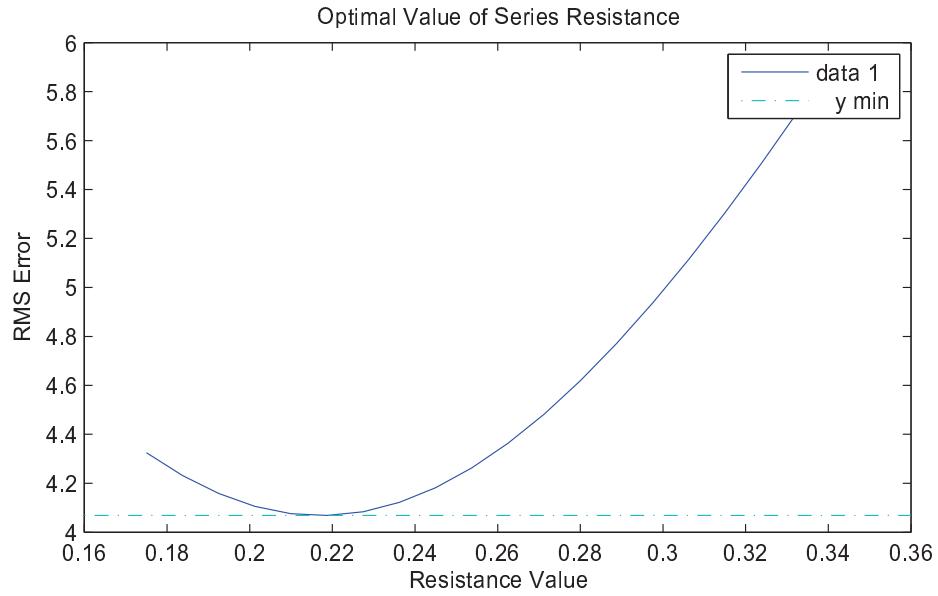


Figure 3.9: Optimal series resistance

operating feedback control modes from quadrature encoder signals, and 1 Mbps CAN bus communication which were discussed earlier and shown in Figure 4.1. The control algorithm to change between engaged and disengaged clutch states simply applies $\pm 12V$ to the linear actuator to actuate the clutch as quickly as the hardware allows while checking for fault thresholds from unexpected behavior.

The purpose of a clutch model in the EcoSIM simulator is to represent typical actuation time and reasonable values for position, speed, and current consumption by the clutch system for the controller to operate properly. Two approaches were taken in order to quickly validate control algorithms effectively. First, a Simulink model is created, as shown in Figure 3.10. Second, a spare linear actuator and motor controller were installed on the HIL system for testing and model comparison.

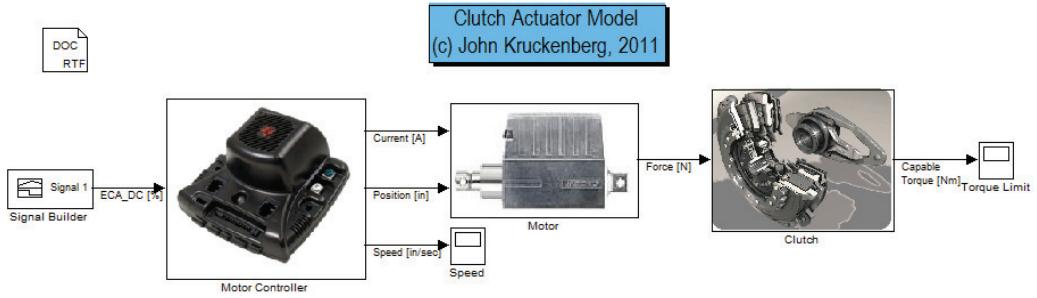


Figure 3.10: Clutch system Simulink model

The Simulink model of the linear actuator in the clutch actuator system is derived from a DC motor state variable representation. Manufacturer data was used to characterize the motor controller and clutch system using lookup table data. The linear system model for a rotating DC motor uses the current $i(t)$ and angular velocity $\omega(t)$, and angular position $\theta(t)$ as states of the system, defining the state variable $x(t)$ as

$$x(t) = \begin{bmatrix} i \\ \omega \\ \theta \end{bmatrix} \quad (3.3)$$

The linear system, disregarding nonlinear effects such as friction and state variable limits, accounts for the electrical circuit dynamics and mechanical inertial dynamics using

$$\dot{x}(t) = \begin{bmatrix} -R/L & -K_b/L & 0 \\ K_m/J & -K_f/J & 0 \\ 0 & 1 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1/L \\ 0 \\ 0 \end{bmatrix} V(t). \quad (3.4)$$

For the purposes of the EcoCAR actuator model, the electrical resistance R , electromotive force coefficient K_b , armature constant K_m , and mechanical system damping ratio b are small and neglected because their effects are much faster than the linear actuator's performance.

The output is then selected as

$$y(t) = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} x(t). \quad (3.5)$$

with constants k_1 , k_2 , and k_3 selected as the unit conversions and gear ratios such that the output contains linear actuator force F in pounds (force), actuation distance d in inches, and actuation speed v in inches/second.

$$y(t) = \begin{bmatrix} F \\ d \\ v \end{bmatrix} \quad (3.6)$$

Adjustments are made in the Simulink model to limit the actuator's range of motion. Model actuation performance is tuned using vehicle data to ensure that engagement and disengagement times are representative of the real system to help with fault diagnosis development discussed in Chapter 4.

This clutch model was validated by comparing to recorded vehicle data. Figure 3.11 shows clutch actuation when moving the gear shifter from drive to park for one minute before shifting back to drive. Since Clutch 2 is engaged (at 1.7 inches) in Drive and Reverse gears, the clutch moves twice when shifting to and from Park. The model was calibrated to tune delay and actuation speeds in both directions to represent the same clutch positions in appropriate operating modes.

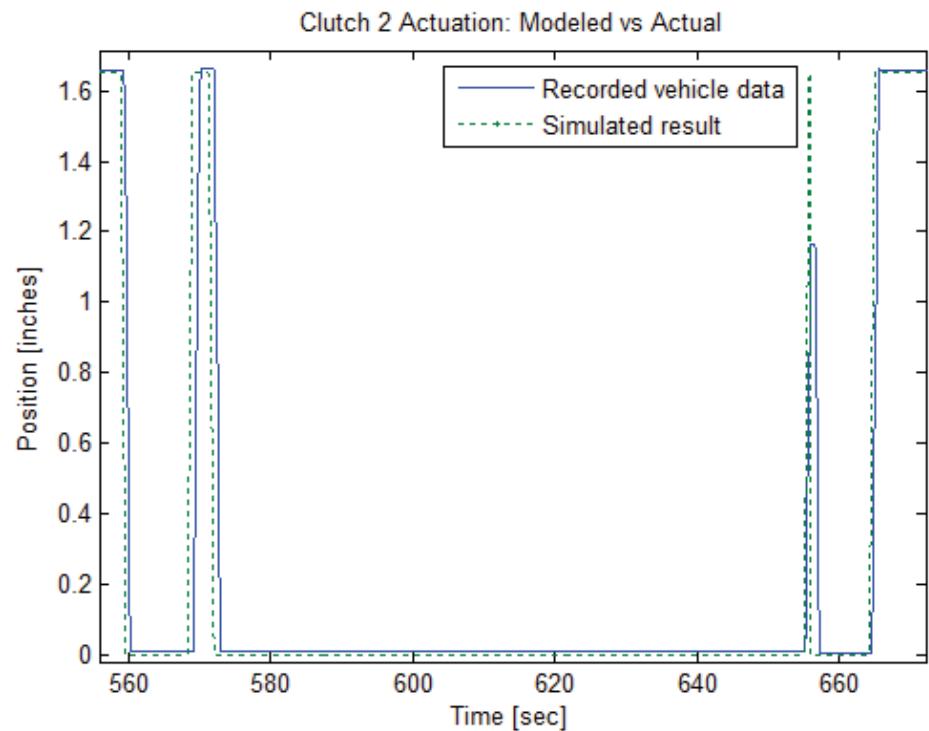


Figure 3.11: Clutch system model validation

3.4.3 Summary

This chapter has described the process of model development as the purposes for modeling and available data change through the development process. A description of changes to the HIL system over time has been shown, as well as specific examples of model development for battery and clutch systems in the vehicle. The high voltage battery model development was discussed from simplified to more complex dynamic models which were validated by comparison to the completed vehicle. Clutch system models were discussed along with their relationship with real hardware used on the HIL system and in the vehicle. In the next chapter, these modeling results will be extended for use in fault diagnosis applications in order to understand and detect vehicle system failures for improved safety and reliability.

Chapter 4: FAULT DIAGNOSIS

Fault diagnosis is a critical concept for the safety, functionality, reliability, and maintainability of any vehicle system. Not only are diagnostics such as the OBD-II standard required for production vehicles, but they make the operation of production and even prototype vehicles safer and more reliable. The Failure Mode Effects Analysis (FMEA) process (see Appendix B) is used to determine which faults are the most important to diagnose according to the following criteria:

- *Severity* of Failure
- Likelihood of *Occurrence*
- Likelihood of *Detection*

Subsystems designed, built, and programmed by the OSU team tend to score as much higher priorities by these criteria, so the analysis of the twin clutch transmission, converted E85 engine, and vehicle torque security discussed in this chapter are also highly relevant to the vehicle's fault diagnosis strategy.

Many fault diagnosis strategies can be tested in Hardware in the Loop (HIL) simulation using plant models (see Chapter 3) and controller hardware features (See Chapter 2).

4.1 Dual clutch transmission system

The OSU EcoCAR team designed a dual clutch single speed transmission to allow the vehicle to operate in distinct series (city) and parallel (highway) modes, similar to the operation of the planetary gearset in the 2011 Chevy Volt. The transmission system components are shown in Figure 4.1.

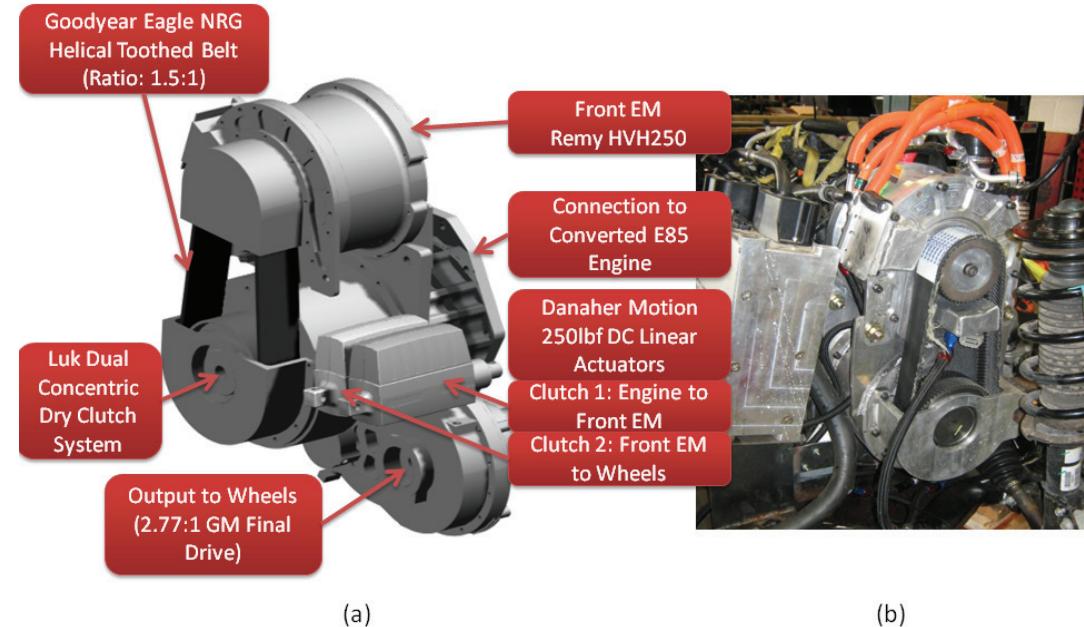


Figure 4.1: (a) Transmission CAD model, (b) Transmission as built

Based on the team's FMEA, important faults of this transmission system include clutch slip, improper clutch actuation, and exceeding limits on actuation time, distance, and current consumption. When working properly, the transmission system

provides multiple options for mitigating other vehicle powertrain failures due to multiple torque paths. However, in the event of a transmission failure, the range extending capability of the vehicle as well as overall functionality may be limited.

4.1.1 HIL Testing

Hardware in the loop testing was used to test clutch actuator control and performance without risk of unsafe or damaging effects on the vehicle. A test rig was developed to emulate the actual movement in the vehicle. The clutch actuator component selected for the EcoCAR vehicle to match the Luk clutch requirements are shown in Table 4.1. This electromechanical actuator is able to provide enough force to allow maximal torque to be transmitted through the clutch system, and the Danaher Motion system also includes internal stall protection and quadrature position and speed measurement. The vehicle employs two of these actuators to actuate the two clutches in the Luk dual clutch system using lever arms. HIL testing also results in characterizing the nonlinear limit circuitry of the actuators.

Table 4.1: Clutch actuator manufacturer data

Manufacturer	Danaher Motion
Description	Electromechanical Linear Actuator
Part Number	PR12-02-2A65-04DCS
Feedback measurement option	Quadrature AB encoder
Actuation force	250 lb _F

The CAN-controlled motor driver was also tested on the HIL system, and its specifications are shown in Table 4.2. Two of these actuators communicate with the

dSPACE MABX primary controller in the vehicle on a 1 Mbps CAN network (refer back to Figure 2.2).

Table 4.2: Clutch actuator motor controller

Manufacturer	Texas Instruments
Description	Motor Controller
Part Number	MDL-BDC24
Operating Voltage	12/24VDC
Peak Continuous Current	40A
I/O Capabilities	PWM, Quadrature encoders, 1 Mbps CAN
Software Features	Onboard fault diagnosis, feedback control

The clutch actuator and motor controller subsystem was mounted in a test rig, shown in Figure 4.2, to immitate the range of motion experienced in the vehicle and to inspect possible failures in operation. This rig was connected to the HIL equipment rack, shown in Figure 4.3, in order to allow some component testing with limited coordination with the HIL plant model. The actuator was fitted with a collar to provide hard stops mimicking the operating position range in the vehicle. HIL testing along with the controller and plant models allows for testing the actuator at the rate it will be used in a typical drive cycle in vehicle operation.

Test data from this system in the vehicle indicates some problems with the performance which can be detected and resolved using HIL testing. Figure 4.4 shows clutch state requests as (0) disengaged to 0 inches and (1) disengaged to 1.7 inches. The state requests and actual positions do not always match and sometimes chatter, which is a problem with the controller requesting clutch transitions more quickly than they can actually occur.



Figure 4.2: Clutch actuator test rig assembly



Figure 4.3: Clutch actuator test rig completed and tested on HIL

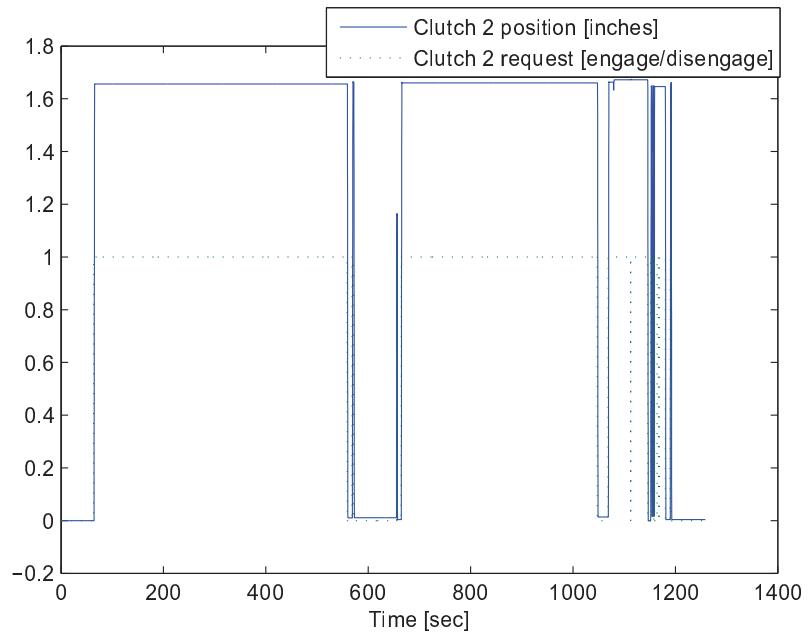


Figure 4.4: Clutch actuator measured data in vehicle

The clutch actuator's range of normal current consumption when commanded to extend or retract was determined experimentally on the HIL system for a baseline value. Figure 4.5 shows the expected current profiles for extending and retracting the linear actuator. Fourteen curves are shown for repeated actuator movement through the range of motion experienced in the vehicle, simulated by adjustable hard stops. The current profiles show that the current consumption is very repeatable and distinct for engaging and disengaging actions. This information is used to set preliminary limits for fault detection, discussed in the next section.

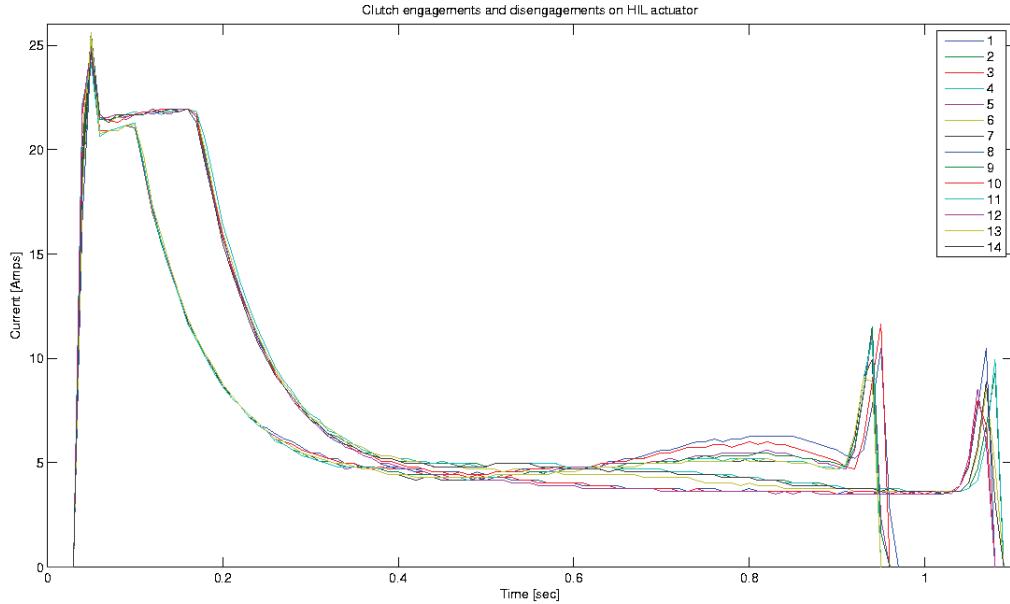


Figure 4.5: Measured current consumption for clutch engagement and disengagement

4.1.2 Limit Checks

Based on initial component testing and HIL implementation, some initial values for limit checks were chosen based on intuition and assuming a low penalty for slightly slower detection. Limit checks involved in the dual clutch transmission system include the quantities listed in Table 4.3.

Limit thresholds for the detection of faults are often initially set based on intuition, but there are benefits to minimize fault detection time and false positives while maximizing detection accuracy. Statistical methods are helpful to characterize typical performance and set limits for unexpected values. The distribution shape for clutch engagement time is shown in Figure 4.6. This distribution plot is shown for some background information on measured clutch actuation during the EcoCAR dynamometer testing at the EPA National Vehicle and Fuel Emissions Laboratory

Table 4.3: Dual clutch transmission system limit checks

Description	Value	Measurement criteria
CAN message Δt	1 sec	From dSPACE CAN blocks
Clutch Slip Percent	20%	CAN axle speed messages
Max Position (Extended)	2 in	CAN linear actuator position messages
Min Position (Extended)	1.3 in	CAN linear actuator position messages
Max Position (Retracted)	0.1 in	CAN linear actuator position messages
Engagement Speed	0.15 in/sec	CAN linear actuator speed messages
Time to Extend	2.1 sec	CAN linear actuator position messages
Time to Retract	1.65 sec	CAN linear actuator position messages
Clutch Actuator Current	30A	CAN linear actuator current

in March 2011. Due to hard stops on the clutch actuator movement, long actuation times are not possible, thus skewing the distribution. A Weibull distribution is a representative fit to account for this skew. Additionally, box plots for time to engage and retract are shown in Figure 4.7. The engagement and disengagement box plots show that engagement typically takes 0.1 seconds longer, and show the maximum and minimum recorded time to complete actuation. These maximum and minimum values recorded under normal operation can be used to set fault limits. Since the penalty for false detection is high, the limits are set to 50% above the maximum recorded values, which was shown in Table 4.3.

4.2 Converted E85 Engine and Electric Machines

The EcoCAR vehicle's two electric machines and engine are highly critical to the safety and reliability of the EcoCAR vehicle. Since the vehicle's engine control was completely developed by the OSU team, fault diagnosis was required for a large number of functions. Rev limiting, oil pressure limits, thermal power limiting, and

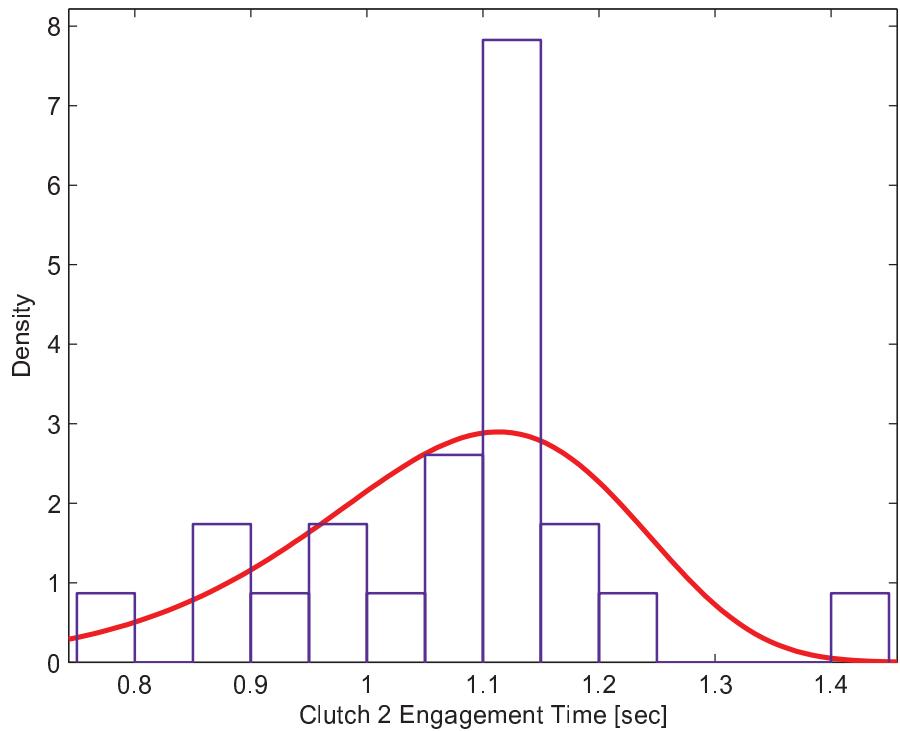


Figure 4.6: Clutch engagement time distribution

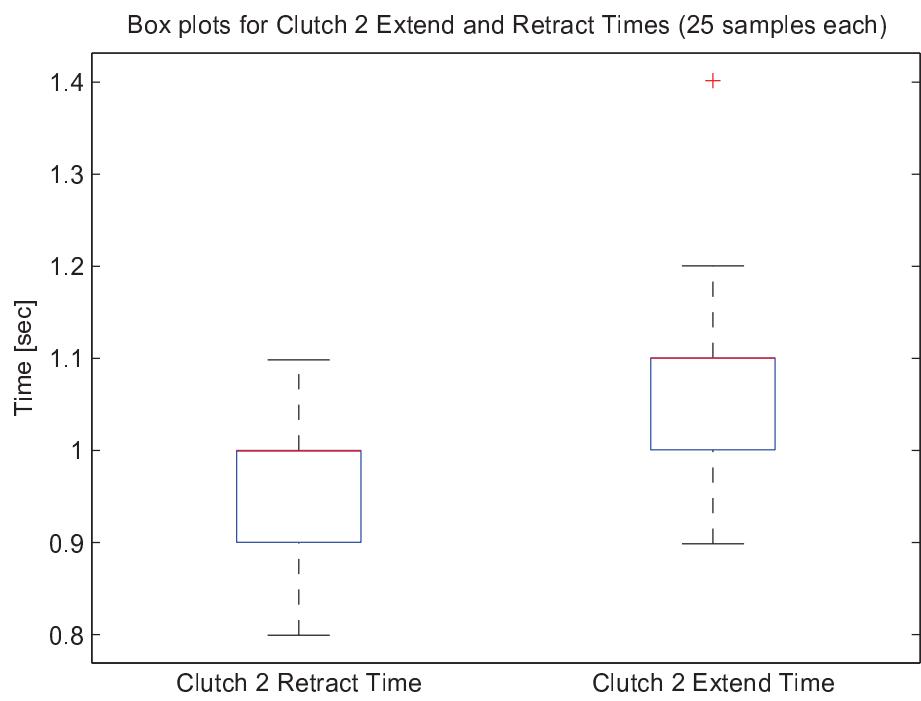


Figure 4.7: Clutch engagement and disengagement box plots

exhaust temperatures are monitored to ensure safe operation of the prototype engine conversion. The rev limiter, for example, was developed offline in simulation and HIL testing before ever running the development engines. Figure 4.8 shows the rev limiter as tested in the engine dynamometer, set to 3000 RPM rather than the 5000 RPM that is more dangerous for operation. The strategy is to simply cut fuel when the engine speed exceeds its threshold. Fault library development from Woodward and previous OSU developments [18] aided this process immensely.

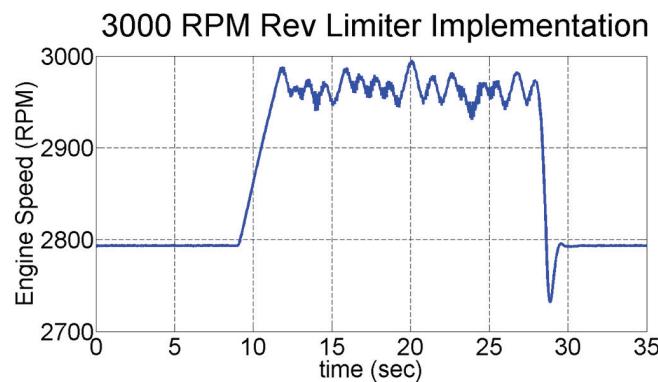


Figure 4.8: Engine rev limiter

The vehicle's two electric machines require unique fault detection, as shown in Table 4.4. Internal fault detection does occur for both electric machines to protect the components. The front electric machine reports internally detected faults through CAN communication to the primary controller, but the rear electric machine is configured to be controlled by analog signals, and limited fault detection is available for feedback. Therefore, additional estimation strategies are needed to ensure proper operation. Torque estimation will be discussed in the next section as an example which applies to all powertrain components.

Table 4.4: Comparison of electric machine communication

Description	EV1 Gen 1 Induction Machine	HVH-250 PM Machine
Manufacturer	General Motors	Remy
Rated Power	103 kW	82 kW
Temp Feedback	Analog RTD	CAN
Torque Feedback	None	CAN
Speed Feedback	CAN (Wheel encoders)	CAN (Quadrature resolver)

4.3 Torque Security

The detection of unintended accelerations relies on an understanding of fundamental vehicle dynamics and forces acting on a vehicle while operating. Estimates of vehicle conditions such as grade and mass are needed to accurately perform this sort of estimation, but this has been proven effective by Winstead [26]. Powertrain component torque must also be well known. Some components, such as the front electric machine and engine, may be able to provide estimates of applied torque. However, other components do not provide this capability or the value is not available due to proprietary messaging. To improve the vehicle-level detection of unintended vehicle accelerations, a model reference fault diagnosis strategy has been employed in simulation to set bounds for acceptable vehicle acceleration given driver pedal inputs and vehicle speed. A reference model applies vehicle dynamics equations and uncertainty analysis on mass, grade, wind speed, and torque uncertainty to create dynamic fault limits. The aerodynamic drag force is modeled as

$$F_{aero} = \frac{1}{2}\rho C_d A V^2 \quad (4.1)$$

where ρ is the air density and V is the vehicle velocity. The drag coefficient C_d and vehicle frontal area A are given by GM as constants used for the EcoCAR vehicle platform. Rolling resistance is modeled as

$$F_{rolling} = mgC_r \cos \alpha, \quad (4.2)$$

where the dimensionless rolling resistance coefficient C_r is initialized as 0.013 according to some vehicle dynamics texts [7], and tuned to match measured vehicle performance. Finally the grade resistance is simply calculated using the mass, acceleration due to gravity, and grade angle α as

$$F_{grade} = mg \sin \alpha. \quad (4.3)$$

The vehicle's tractive force is calculated as the sum of the torques of each individual component and the appropriate gear ratios, so

$$F_{tractive} = \frac{T_{FEM} \cdot g_{FEM} + T_{REM} \cdot g_{REM} + T_{ICE} \cdot g_{ICE} + T_{braking}}{r_{tire}}. \quad (4.4)$$

Using these equations, the force balance on the vehicle is performed in order to create a simple model of expected acceleration for a given driver input. The implementation in Simulink is shown in Figure 4.9.

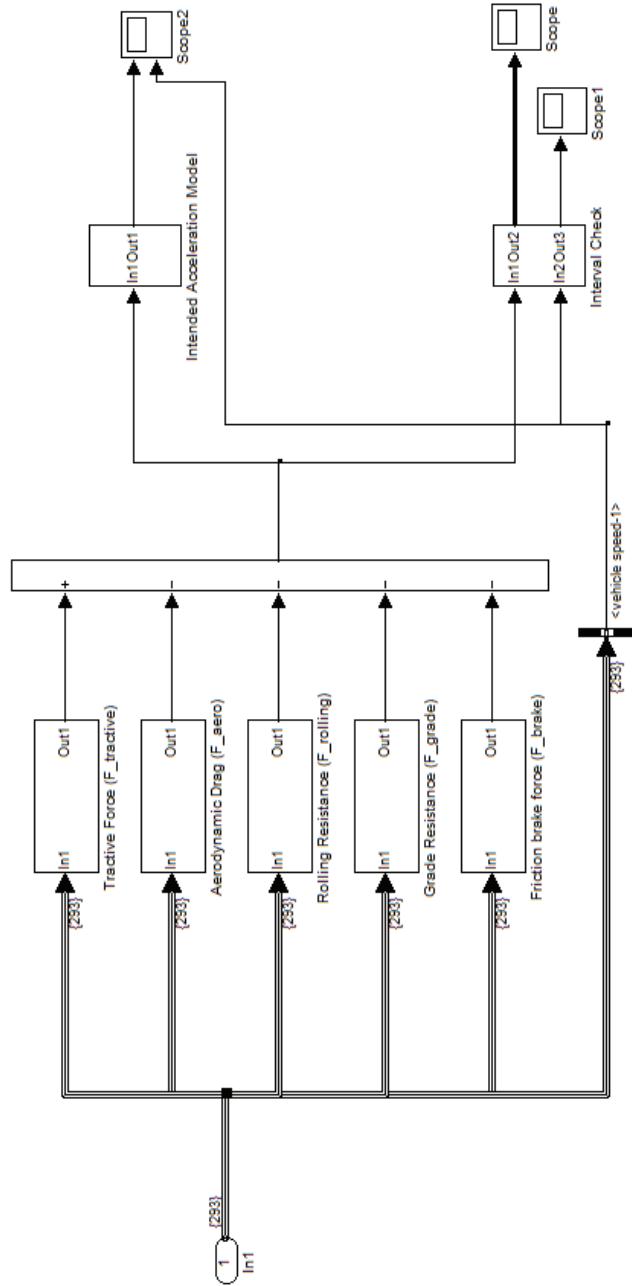


Figure 4.9: Unintended acceleration detection modeled in Simulink

The results of this model can be shown in different levels of implementation. Figure 4.10 shows the effectiveness of this model in SIL simulation and the ability to place bounds on allowable accelerations for comparison. In simulation, the actual vehicle acceleration can easily be bounded by expected accelerations accounting for uncertainty.

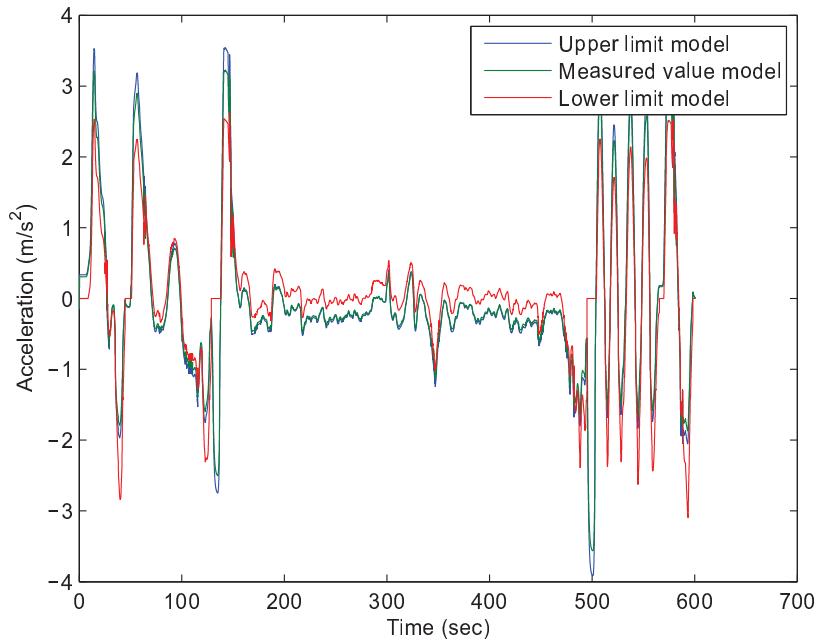


Figure 4.10: Unintended acceleration detection results from simulation

This detection algorithm works very well in simulation because the vehicle forces are calculated in a similar way. However, torque uncertainty in braking force and rear electric machine torque are not well known and require more testing time to make this algorithm practical on the vehicle.

4.4 Summary

This chapter has provided an overview of the control design process for fault diagnosis. Examples of fault diagnosis methods for the dual clutch transmission system, motors and engine, and vehicle level torque security were demonstrated using constant and model reference limit checks. Statistical methods were used to justify limit values using simulation and recorded vehicle data.

Chapter 5 expands on the fault diagnosis concepts described in this chapter to improve vehicle control algorithms using knowledge of failures and improper operation. Concepts from FTA and DFMEA will be discussed again as motivation for selecting high priority failures that can be improved by control system modifications.

Chapter 5: CONTROL DESIGN FOR FAULT MITIGATION

Based on the control system's fault diagnosis strategy discussed in Chapter 4, the vehicle control strategy can be adapted to provide the maximum functionality with minimal risk with respect to safety and component damage. In some cases, such as a high voltage battery failure, continued operation of the vehicle powertrain may not be possible, but braking and 12V accessories may continue to operate, for example. This chapter will provide an overview of control development for the EcoCAR vehicle with an emphasis on the effects of Hardware in the Loop development and fault diagnosis. The field of fault tolerant control systems involves overlapping studies of fault detection and isolation (FDI), robust control, and reconfigurable control, as shown in Figure 5.1.

This process includes failure analysis of the vehicle powertrain, failure mode effects analysis, and system configuration for fail-safe operating modes. A brief overview of real-time control implementation will be followed by methods for making supervisory powertrain controls more fault-tolerant. Finally, the combination of fault-tolerant adaptive strategies with the Equivalent Consumption Minimization Strategy (ECMS) will be presented.

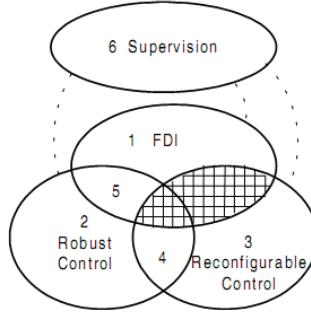


Figure 5.1: Fault tolerant control fields of study (from Patton [17])

5.1 Real-Time Control Implementation

The real-time control implementation process ensures that software developed in offline simulations can perform properly on actual hardware, including executing software within a sample step time and operating within the limits of I/O hardware. Since this transition from offline simulation to real-time hardware implementation is repeated for many software revisions, some practices of maintaining certain model structures can be very helpful to separate control software from software to manage I/O hardware. Real-time hardware capabilities and recommended model structures will both be discussed within this section.

5.1.1 Controller Hardware Capabilities

Chapter 3 provided detail on real-time implementation issues as they relate to Hardware in the Loop development and testing. One lesson learned in the development of the EcoCAR control strategy was the value of testing task execution time. Through discussions with engineers at dSPACE and The Mathworks, we were able to improve the efficiency of complex algorithms while simultaneously adding utility

to the software. Both dSPACE and The Mathworks have tools to measure task execution time. dSPACE uses their ControlDesk real-time interface to the hardware device to measure the time for each sample step, but does not separate the execution time of each subsystem. Execution time can be shown to vary due to vehicle operating modes, however, so a combination of mode diagnostics and dSPACE execution time measurements can be quite helpful. The Mathworks provides a tool called the Simulink Profiler which shows the execution performance of each task and subsystem, but only represents the execution time for the host PC rather than the actual hardware. The Profiler measurements may be most useful for comparison of relative complexity among versions of software and various operating modes. Using these tools, we discovered some methods for eliminating Simulink embedded Matlab functions to greatly improve the performance and modular nature of the Equivalent Consumption Minimization Strategy for a maintainable and calibratable control strategy.

5.1.2 Model Structure

Maintaining useful model structures can make the transition of control software between offline and real-time implementations easier. The main concern is in keeping the same signal links in Simulink for different simulations. Additional work is still needed when new signals are added, but the most complete model layouts are backward compatible. Perhaps the most important features of this model structure are:

- Maintaining clear organization as control strategy develops
- Separating control software from I/O interface

- Separating I/O interface descriptions to allow porting control software to different hardware, such as the HIL system rather than its original controller

Examples of the top level Simulink model templates as well as the I/O interface descriptions provided for the dSPACE and Motohawk controllers are shown in the following figures. Figure 5.2 shows the top-level template from the dSPACE documentation, showing the basic concept of separating I/O mapping from the control algorithm. More detail on the I/O mapping recommendations is shown in Figure 5.3, including mapping from physical units to ADC counts, frequencies, or other communication medium units.

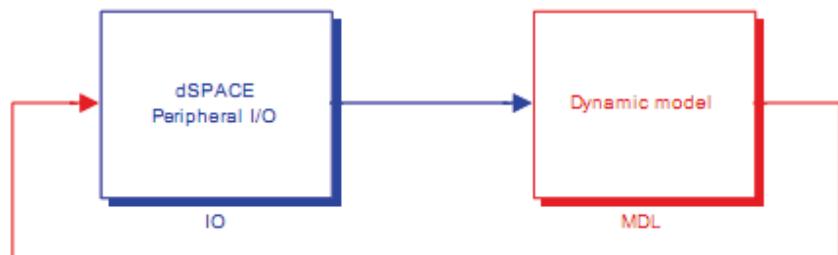
As shown in Figure 5.4, this practice is also recommended for the Woodward Motohawk controllers with the slight variation of separating sensors and actuators. A similar process for signal scaling and mapping is recommended in Woodward templates and documentation, with “Read from HW,” “Convert to Engineering Units,” and “Create a Bus of Sensors” segments annotated within the same subsystem rather than separating them as in the dSPACE blocks. A screenshot of this method is shown in Figure 5.5.

RTI Data

HIL Simulator V1.0

Template

copyright 2004 by
dSPACE GmbH



dSPACE Version
06-Oct-2005

Figure 5.2: Top level dSPACE model structure template from dSPACE simulator documentation (EcoCar_HIL_v1p2)

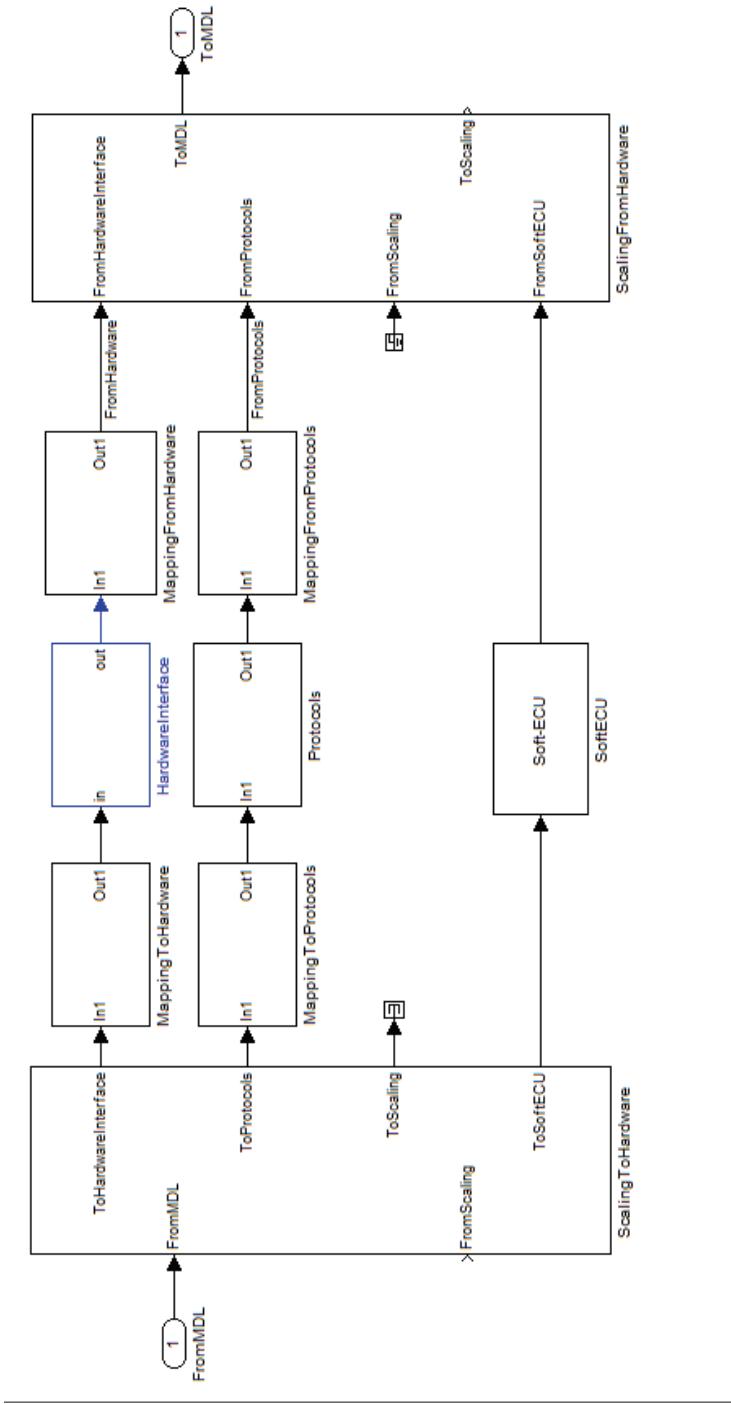


Figure 5.3: I/O mapping in dSPACE model structure template from dSPACE simulator documentation (EcoCar_HIL_v1p2)

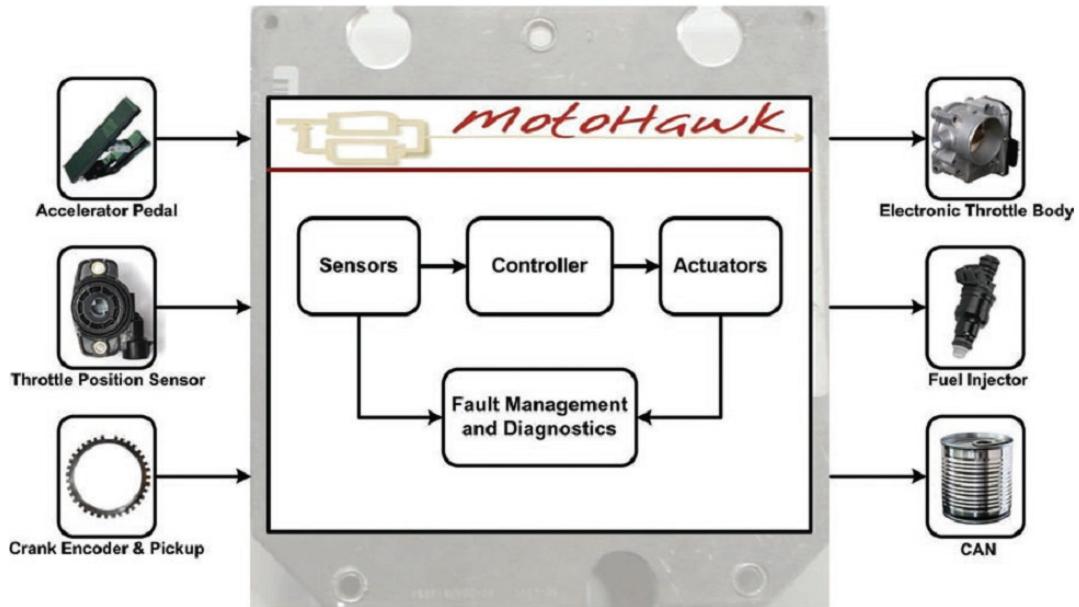


Figure 5.4: MotoHawk recommended model structure from MotoHawk training supplement

5.2 System Reconfiguration

Due to the availability of multiple traction drives in the EcoCAR vehicle, the control strategy of the vehicle can be modified to reconfigure the system operation in the event of a partial powertrain failure. The failure matrix for up to two severe powertrain faults in the vehicle is shown in Table 5.1. These system reconfiguration “limp home modes” are implemented in the supervisory controller’s state machine to ensure safe operation with the most utility available for the vehicle. Some interesting points that are apparent from this matrix are:

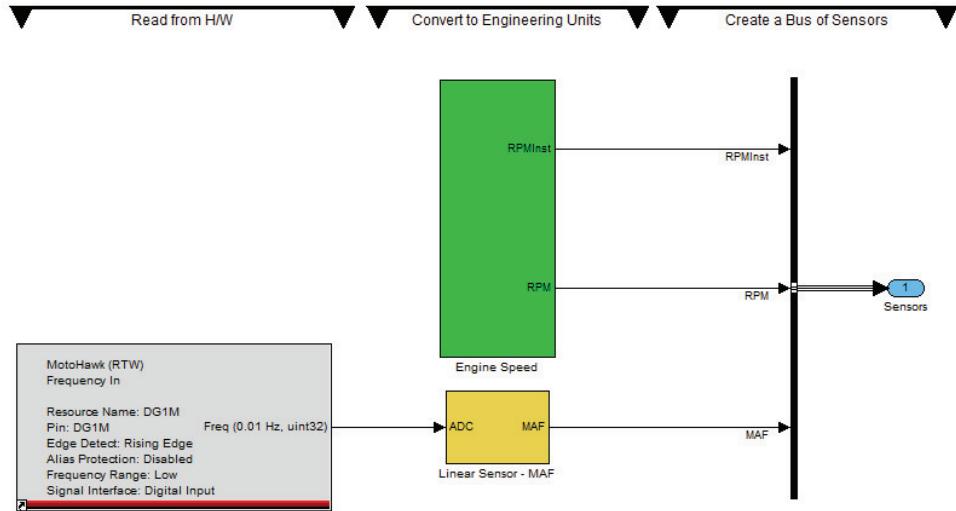
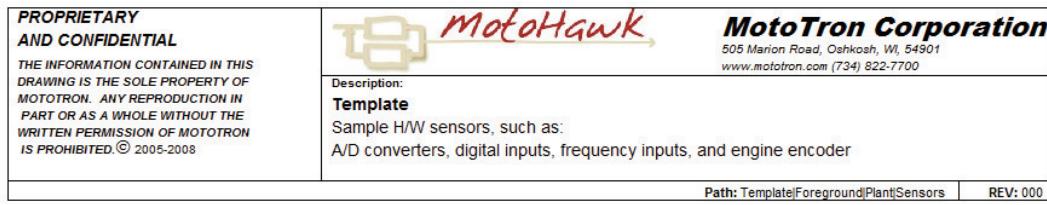


Figure 5.5: MotoHawk model structure template from MotoHawk default template .mdl file

- These failure modes are quite conservative to avoid unsafe or destructive operation.
 - Clutch failures are assumed to be a worst case scenario. If a clutch fails in an engaged position, there should be no assumption that operation in that state will be reliable. The position of a failed clutch could change intermittently or be undetermined depending on the nature of the fault.
 - Charge sustaining series and parallel modes require all powertrain components to operate properly due to the design of the EcoCAR vehicle.
- The HV battery is the only single component that can cause the entire powertrain to fail.
- Most failures result in a charge depleting rear wheel drive operation. By investigating the likelihood of failure, this information can be helpful to determine failures which are most relevant as discussed later in the DFMEA section.

Table 5.1: Powertrain failure mode analysis matrix

Best available limp home mode for 2-system failure cases						
	Clutch 1	Clutch 2	ICE	FEM	REM	HV Battery
Clutch 1	CDRWD	NONE	CDRWD	CDRWD	CDFWD	NONE
Clutch 2	NONE	CDRWD	CDRWD	CDRWD	NONE	NONE
ICE	CDRWD	CDRWD	CD4WD	CDRWD	CDFWD	NONE
FEM	CDRWD	CDRWD	CDRWD	CDRWD	NONE	NONE
REM	CDFWD	NONE	CDFWD	NONE	CDFWD	NONE
HV Battery	NONE	NONE	NONE	NONE	NONE	NONE

Modes:

NONE	No operating modes possible
CD4WD	Charge Depleting 4 Wheel Drive
CDFWD	Charge Depleting Front Wheel Drive
CDRWD	Charge Depleting Rear Wheel Drive

A similar method for investigating paths to failure of the powertrain is Fault Tree Analysis (FTA), shown in Figure 5.6. FTA is a deductive, top down approach to considering possible failures and also includes methods for quantifying risk of failure due to individual components. The process of identifying “minimal cutsets” from the fault tree involves boolean algebra to find components likely to lead to system failure. In the design process, this information can be used to identify where more reliable or redundant components are needed. In failure analysis, this calculation can also be used to identify high risk failures, or ones that are more likely to happen as determined by DFMEA risk priority numbers (see Chapter 4). Probabilities of component failure from failure rate databases such as the Department of Defense MIL-HDBK-217 [25], the Offshore REliability DAta (OREDA) publication [1], or reliability engineering texts such as [22]. Details of DFMEA applications are shown in Appendix B.

In addition to powertrain failures, additional subsystem failure modes are also considered. Since emissions results are a large part of the EcoCAR competition, failure of the emissions control system are a high priority.

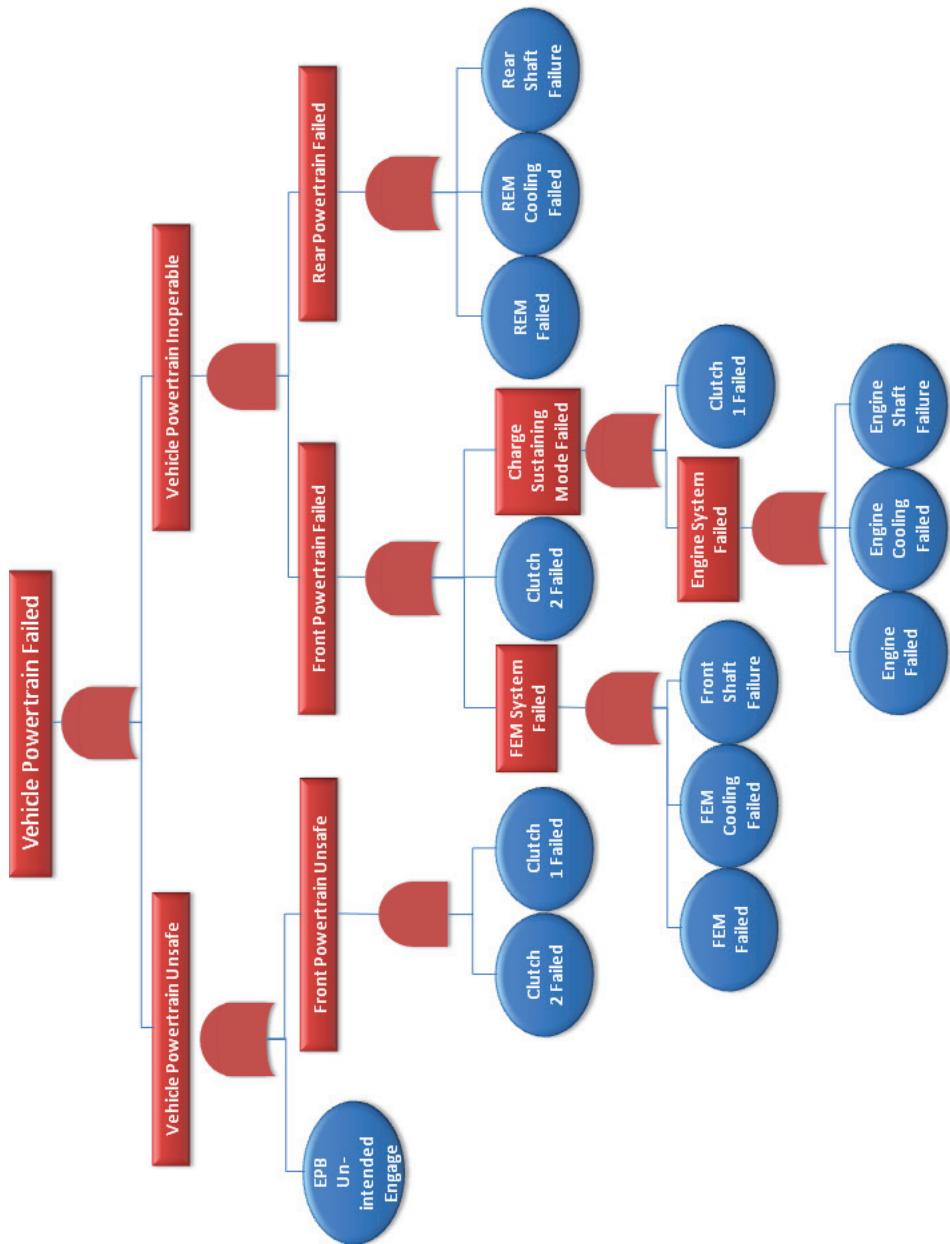


Figure 5.6: Fault tree for vehicle powertrain failure

By becoming aware of situations where limp-home modes are necessary, the state machine that controls the vehicle's operating modes can be adjusted. For EcoCAR, this is implemented in a Simulink StateFlow chart as shown in Figure 5.7. This centralized mode selection has proven to be very helpful to the EcoCAR team compared to distributed software functions performing transition tasks as used in previous competitions.

This section has demonstrated a vehicle control design that adjusts discrete operating modes based on severe powertrain failures. In the next section, an additional application of fault tolerant control will be shown in the power limiting strategy for failures resulting in reduced, but not eliminated, capabilities.

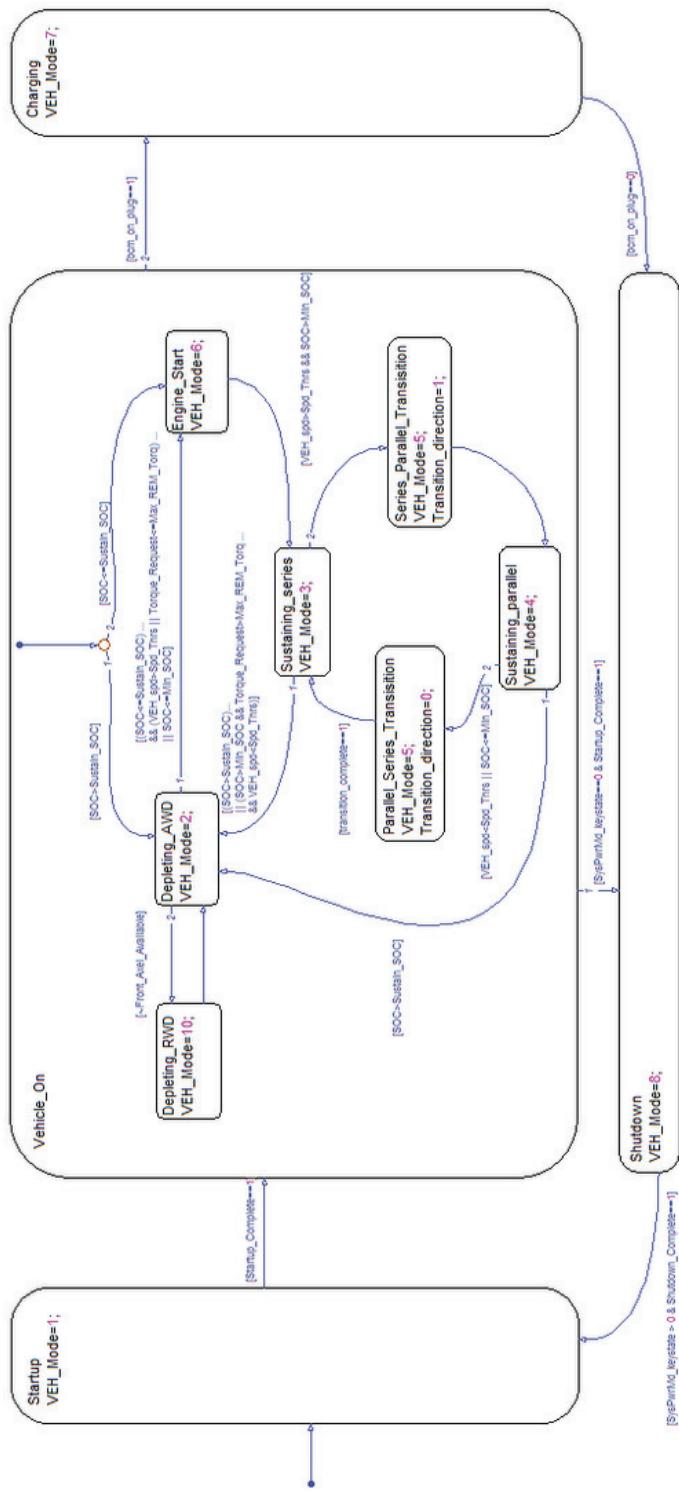


Figure 5.7: State machine for vehicle operating modes implemented in StateFlow

5.3 Power Reduction

In some failure states, operation of the vehicle and powertrain components with reduced power may be possible and desirable. Power limiting failure modes include temperature limits, battery charge and discharge current limits, and clutch slip limits. The power limiting conditions that apply to each powertrain component are shown in Table 5.2.

Table 5.2: Conditions for engine and EM power limiting

Powertrain Component	Thermal Limits	Battery Current	Clutch Slip
Engine	X		X
Front Electric Machine	X	X	X
Rear Electric Machine	X	X	

Thermal power limiting allows for power reduction depending on the severity of an over-temperature event. Battery current charge and discharge limits are rated characteristics of the vehicle's battery pack, and power reduction must occur to prevent emergency shutdown of the battery pack by the manufacturer's Battery Control Module. The clutch torque limits for the Luk clutches used for the EcoCAR vehicle are rated at 250 Nm, but depend on the actuation force provided by electromechanical linear actuators. If this limit is exceeded and the clutch slips, power must be immediately limited to return to static friction operation of the clutch.

5.3.1 Series Mode

The EcoCAR series mode involves driving the vehicle with the rear electric machine while the engine and front electric machine generate electricity. In this mode,

power limiting of the FEM or engine will likely result in an inability to maintain the battery SOC, and power limiting of the REM will result in reduced torque available to the driver at the wheels.

5.3.2 Parallel ECMS

The EcoCAR parallel operation mode couples both EM's and the engine to the wheels, so a variety of torque splits are possible for improved drivability [20], fault tolerance, and efficiency.

The OSU vehicle, as well as previous hybrid vehicle projects, uses the Equivalent Consumption Minimization Strategy (ECMS) as an energy management strategy [15]. This strategy effectively derives a cost function equating electrical energy use to E85 fuel use for the vehicle. An equivalence factor, $s(t)$, scales the cost of electrical energy. If $s(t)$ is small, electrical energy has no cost compared to the cost of fuel. This is generally the case for charge depleting vehicles while stored electrical energy is available. If $s(t)$ is large, the cost of fuel energy may be lower than electrical energy, so use of the internal combustion engine is preferred. This is the case when the vehicle's electrical energy storage system is depleted and more engine use is needed to sustain the state of charge of the battery. The change in $s(t)$ for different drive cycles and operating modes results in an Adaptive Equivalent Consumption Minimization Strategy, or A-ECMS [13].

Typical ECMS Implementation

Define $J_{ECMS}(t)$ as the cost of energy use in the vehicle. The total “fuel” cost is the sum of the liquid fuel $\dot{m}_{ICE,f}(t)$ and the product of the equivalence factor $s(t)$ and the electrical energy consumed by each electric machine, $\dot{m}_{i,eqf}(t)$ for each $i \in \{FEM, REM\}$. This cost is given by

$$J_{ECMS}(t) = \dot{m}_{ICE,f}(t) + s(t)\Sigma\dot{m}_{i,eqf}(t). \quad (5.1)$$

The ECMS control output to the vehicle modular controllers, the engine controller and electric machine inverters, is given by

$$U_{ECMS}(t) = argmin_{u \in U} \{J_{ECMS}(t)\} \quad (5.2)$$

where the control input U minimizes the cost function J . The concept is to minimize the cost function using discrete values of powertrain energy use while adapting the equivalent cost of electricity in terms of fuel.

The ECMS solution is subject to several constraints in the minimization process. To implement these in real-time operation, the EcoCAR team's method for doing so is to apply a high cost to torque splits which do not meet the criteria. The control constraints are given as

$$SOC_{BATT,min} < SOC_{BATT}(t) < SOC_{BATT,max} \quad (5.3)$$

$$P_{BATT,min} < P_{BATT}(t) < P_{BATT,max} \quad (5.4)$$

$$\begin{cases} T_{ICE,min(fric)} < T_{ICE,req}(t) < T_{ICE,max}(t) \\ T_{REM,min}(t) < T_{REM,req}(t) < T_{REM,max}(t) \\ T_{FEM,min}(t) < T_{FEM,req}(t) < T_{FEM,max}(t). \end{cases} \quad (5.5)$$

The battery state of charge $SOC_{BATT}(t)$ is sustained at a low value near 20 percent to provide maximal functionality for the depleted battery during charge sustaining operation. The power limits of the battery P_{BATT} also cannot be exceeded to prevent cell damage or emergency shutdown of the battery by the battery controller. The

torque constraints $T_i(t)$ require that the torque limits of each powertrain component are not exceeded, which include mode-dependent constraints such as clutch limits.

Effects of Failures for ECMS

Failed or otherwise unavailable components have particularly undesirable effects because the total torque request may not be proportional to the torque request of each component. The unrecognized failure of one component can result in torque realizations not matching driver requests. This is demonstrated using the EcoSIM simulator in the following figures. Figure 5.8 shows a typical comparison of vehicle velocity to drive cycle velocity under normal operation. However, if a component fails and the operation of the torque split strategy does not change, some undesired results may arise. Shown in Figure 5.9, an engine failure is undetected and the ECMS control continues to divide torque requests among the two electric machines and the engine. Because torque requests are split based on efficiency and not proportionally, actual vehicle speed does not correspond to desired speed.

Changes for Fault Tolerance

To operate properly in the event of detected powertrain faults, ECMS must be modified to allow components to be disabled or power limited. Different levels of faults are possible, including complete failure of a component such as a motor, or linear degradation of the total power output based on the degree of a fault, such as an over-temperature issue.

In either case, the change is implemented in ECMS by changing the search space allowed to find the optimal torque split. The implementation of this change described next.

Considerations when implementing fault-tolerant ECMS are:

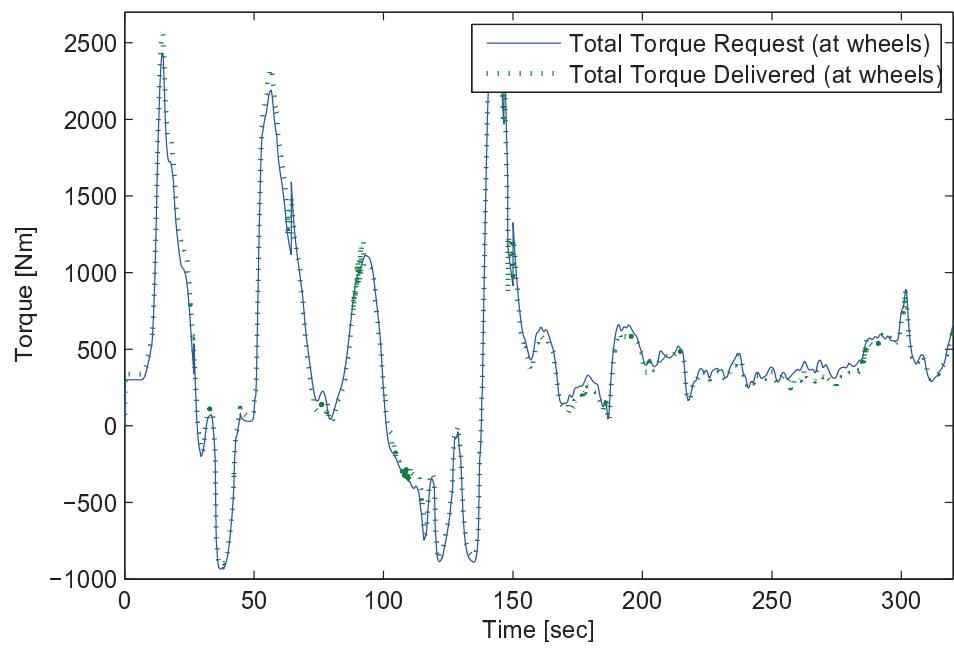


Figure 5.8: ECMS simulated actual total torque request vs torque delivered

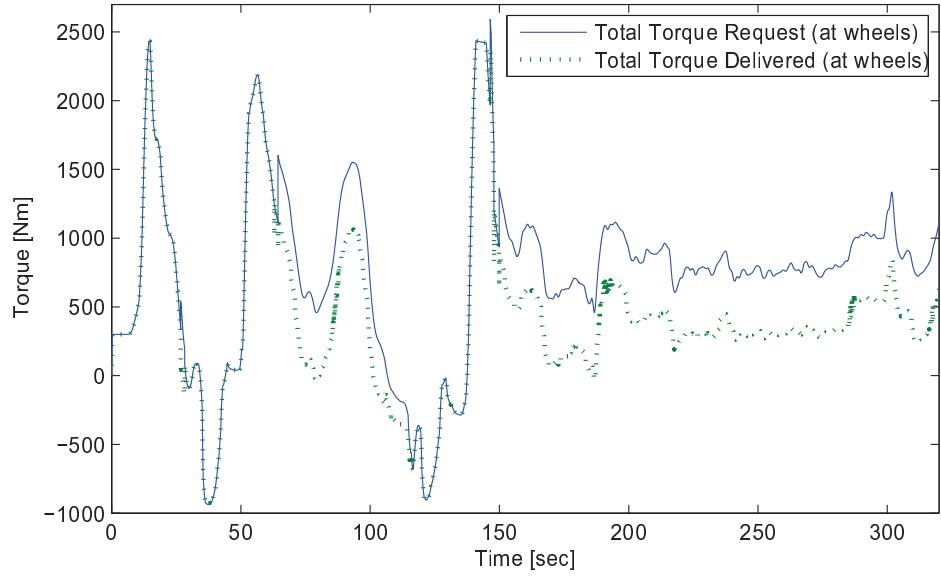


Figure 5.9: ECMS total torque with simulated engine failure

- Driver startle: Unexpected torque delivery
- Smoothness of control: Reduction of oscillation and discontinuities in torque delivery
- Consumer acceptability: Noise, vibration, harshness, and responsiveness

The bulk of the work in implementing a fault-tolerant ECMS involves identifying which failures result in torque limiting or component disabling, but after this detection and determining the method for resolution, the ECMS calculation must be modified. This is done for each of the powertrain components, essentially by modifying the maximum and minimum torque that are considered for the optimization calculation. The original A-ECMS equation (Equation 5.1) is modified to implement a power

limiting factor $\alpha \in [0, 1]$ to account for this power reduction for each powertrain component, so

$$J_{ECMS}(t) = \alpha_{ICE} \dot{m}_{ICE,f}(t) + s(t) \sum \alpha_i \dot{m}_{i,eqf}(t). \quad (5.6)$$

The effectiveness of this control variation can be shown in simulation using various thermal input traces to trigger power limiting failure states. The EcoSIM simulator introduced in Chapter 3 is used to demonstrate torque requests when following a drive cycle. The general result is that the driver torque request is now successfully met, but the total available power and the ability to sustain state of charge of the high voltage battery may sometimes be limited by unavailable components. Due to these limitations, driver notification is an integral part of any system reconfiguration routine that changes the capabilities of the vehicle.

The engine temperature fed into the EcoSIM simulation is shown in Figure 5.10. This temperature signal is selected from recorded signals in the vehicle because it is particularly noisy, and is shown to enter the power limiting region of the engine as noted by dashed lines on the plot. Power reduction begins when the engine temperature exceeds the 100 deg C threshold to start power limiting, and the engine shuts off at 110 deg C. The "Reset power limit" threshold refers to a strategy for holding the peak temperature value measured while above the threshold to avoid oscillations in power limiting factors.

The initial strategy for engine power limiting tested in simulation simply reduced engine power linearly in the range specified in Figure 5.10. This resulted in a power limiting percentage shown in Figure 5.11. The resulting engine torque is shown in Figure 5.12. Note the oscillations that occur at the end of the FHDS cycle. This undesired oscillation is caused by the changing power limit based on temperature sensor data.

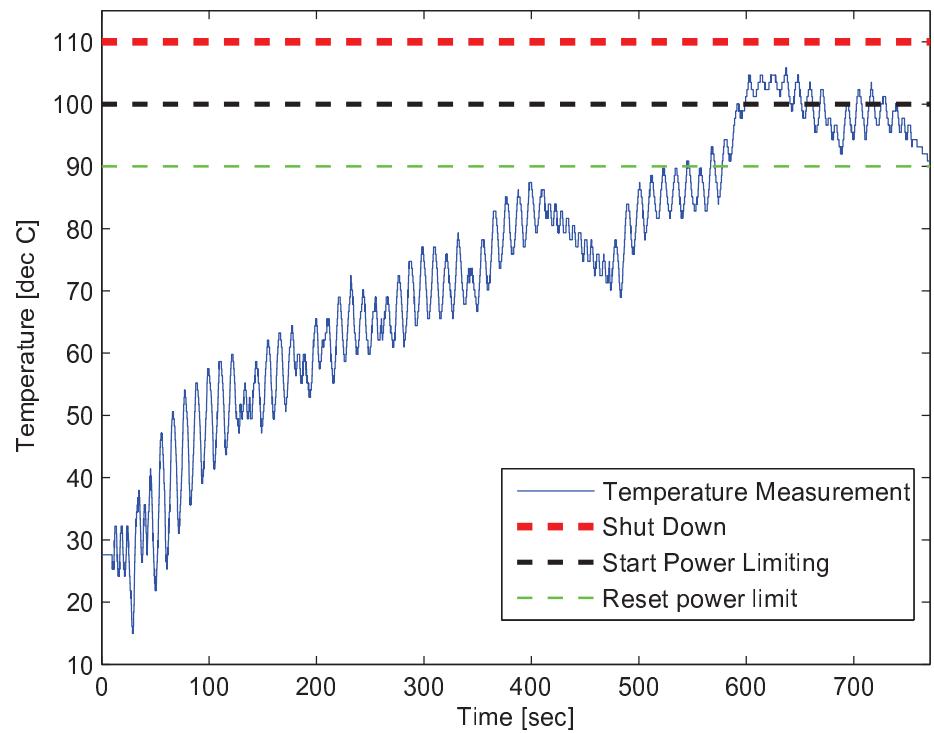


Figure 5.10: Engine temperature input to invoke power limiting operation

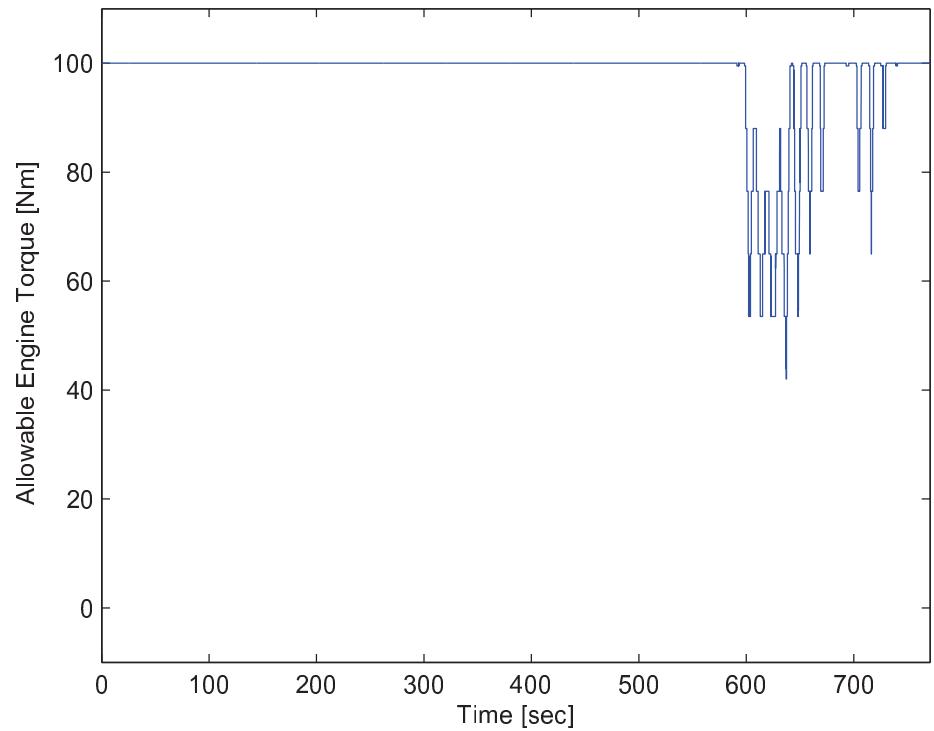


Figure 5.11: Engine power limit percentage for linear power reduction

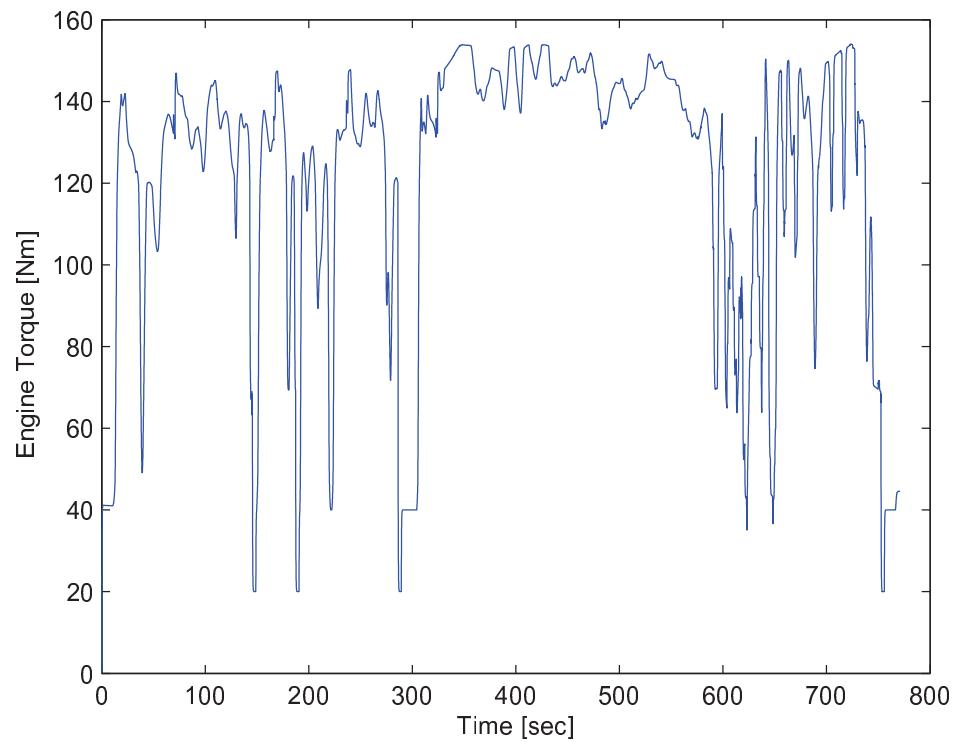


Figure 5.12: Engine indicated torque from simulation given oscillating power limit

The oscillating engine torque shown in Figures 5.11 and 5.12 is undesirable for emissions, efficiency, and drivability of the vehicle, so another method for power reduction is needed. A common method for removing this noise is to design a filter, but this adds delay to the measurement while lower frequency oscillations and variations may remain, which are still undesirable. Therefore, the chosen method for setting power limit thresholds was to record the maximum temperature that has occurred since the limiting began. To prevent chattering at the “Start Power limiting” threshold, a deadband was implemented by adding a “Reset power limit” threshold at a lower temperature, as referenced in Figure 5.10. The resulting power limit percentage is shown in Figure 5.13, and the engine torque resulting from this implementation is shown in Figure 5.14.

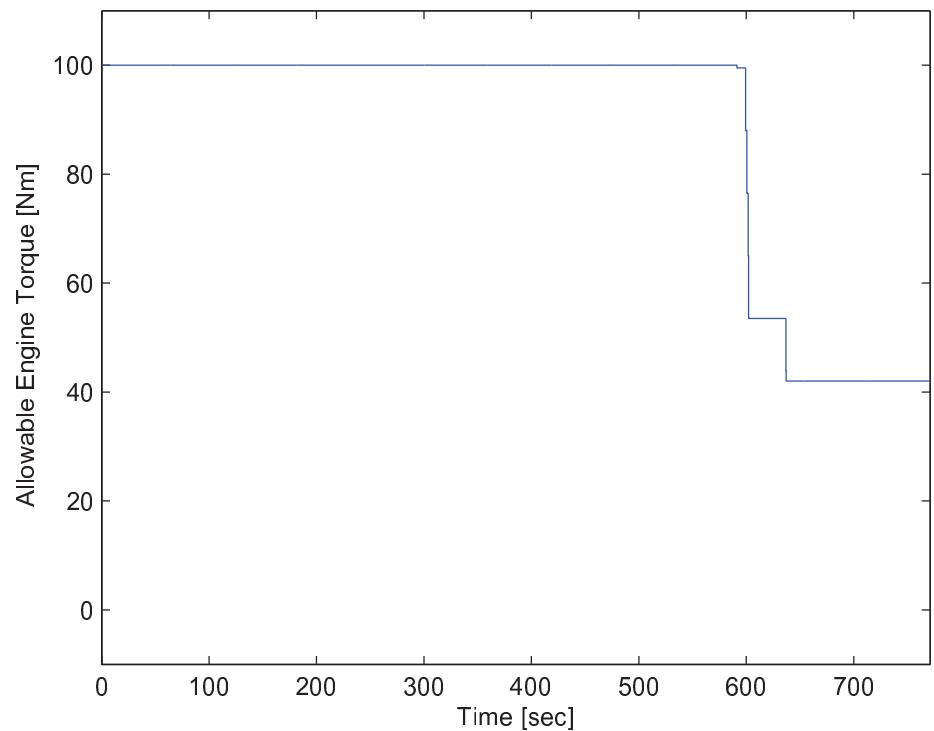


Figure 5.13: Engine power limit percentage when over temperature

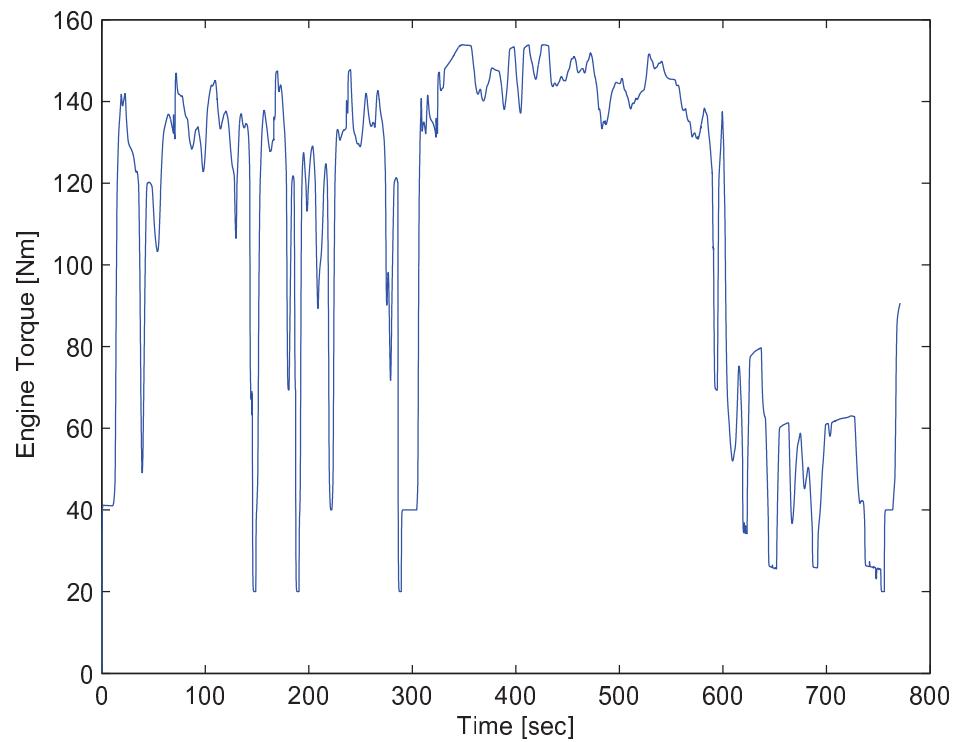


Figure 5.14: Engine indicated torque given adjusted power limit hysteresis

5.4 Summary

This chapter has shown the effectiveness of applying fault diagnosis information to improve the overall vehicle control design. Two general cases of fault tolerant control examples included (1) system reconfiguration of discrete operating modes and (2) power reduction operation of the A-ECMS control. Simulation and formal analysis techniques have been used to demonstrate the EcoCAR vehicle's control design as well as its effectiveness.

The next chapter continues with a summary of the entire control design process involving real-time simulation, fault diagnosis, and control design. Contributions of this project and future work will be discussed along with conclusions.

Chapter 6: CONTRIBUTIONS AND FUTURE WORK

Chapter 5 demonstrated how a variety of tools and methods can be applied to the design of a safe and robust control system for a prototype vehicle in a rapid development environment. This work was performed as part of the OSU EcoCAR project, fall 2008 through spring 2011. Similar research will continue in the EcoCAR 2 project from fall 2011 through spring 2014. Related work will also be applied to similar projects at the OSU Center for Automotive Research using tools and methods developed through this process.

6.1 Thesis Summary and Contributions

This thesis has demonstrated the progression of a control design process from modeling (Chapter 3) to fault diagnosis (Chapter 4) and their effects on a complete vehicle control system (Chapter 5).

Model fidelity, modeling methods, and validation were introduced as a framework for off-vehicle development, which is critical to developing prototype systems on a fast schedule. Models can be used in various stages of control loop development, including model in the loop, software in the loop, and hardware in the loop simulations in order to develop and test control algorithms quickly before complete systems become available for testing, or in parallel with other development work.

Fault diagnosis can begin at the modeling stage of control development. This is particularly helpful in developing limit checks and confirming component interaction so that it is developed by the time hardware is implemented. Fault diagnosis related to the OSU EcoCAR project’s custom dual clutch transmission system and vehicle level torque security events were explored in detail due to the high priority of their detection as determined by the team’s DFMEA (Appendix B).

Vehicle system models and fault diagnosis strategies are used together to develop fault tolerant control systems for improved vehicle safety, reliability, and performance. Fault tolerant control methods falling under three major categories were explored: (1) fault detection and isolation, (2) robust control, and (3) reconfigurable control. Reconfigurable control is especially prominent in the EcoCAR vehicle prototype due to capabilities of providing limited operating modes through the vehicle’s custom dual clutch transmission and aftertreatment systems.

6.2 Future Work

Future developments in fault diagnosis and hardware in the loop simulation will continue at the Ohio State Center for Automotive Research in the areas of automated controller testing in HIL simulation, torque security for innovative powertrain architectures, and fault detection algorithms using model predictive control and estimation. These new developments will be able to use existing models and software developments as a framework for future projects.

6.3 Conclusions

Fault diagnosis and HIL simulation methods play a large role in accelerating advanced technology development cycles. With improved reliability and performance,

control systems can be designed that are proven to be ready for production years faster than previously possible.

Due to space constraints, confidentiality agreements, and time limitations, only a selection of relevant work on the EcoCAR project has been presented in this thesis. However, the completeness of the project is evident in the success of the vehicle in competition and around campus as a working vehicle prototype. The research presented herein has contributed greatly to the success of the current project as well as laying a firm foundation for future work in EcoCAR 2 and future innovations.

Appendix A: LIST OF SYMBOLS AND ABBREVIATIONS

α	Power limiting factor for reduced power mode ($\alpha \in [0, 1]$)
ADC	Analog to Digital Converter
A-ECMS	Adaptive Equivalent Consumption Minimization Strategy
BATT	High Voltage Battery
CAN	Controller Area Network
CCP	CAN Calibration Protocol
CNG	Compressed Natural Gas
DFMEA	Design Failure Mode Effects Analysis
E85	85% Ethanol, 15% Gasoline Fuel
ECU	Engine Control Unit
EM	Electric Machine
eqf	Equivalence Factor
E-REV	Extended Range Electric Vehicle
EV	Electric Vehicle
f	fuel
FDI	Fault Detection and Isolation
FEM	Front electric machine
FMEA	Failure Mode Effects Analysis
FTA	Fault Tree Analysis

GM	General Motors Corp.
GMLAN	GM CAN network protocol
HIL	Hardware in the Loop Simulation
ICE	Internal combustion engine
i	Powertrain component index, $i \in \{FEM, REM\}$
I/O	Input/Output
$J_{ECMS}(t)$	ECMS Cost Function
$\dot{m}_{i,eqf}(t)$	Mass flow rate of electrical equivalent energy
$\dot{m}_{ICE,f}(t)$	Mass flow rate of ICE fuel
MABX	dSPACE MicroAutoBox
Mbps	Megabits per second, i.e. a CAN network baud rate
OEM	Original Equipment Manufacturer
PWM	Pulse width moduleation
P	Power
REM	Rear electric machine
RMS	Root mean square
$s(t)$	ECMS Equivalence Factor
SIL	Software in the Loop Simulation
SOC	Estimated Battery State of Charge
t	Time
τ	First order system time constant
T	Torque
$U_{ECMS}(t)$	ECMS Control Input of Torque Requests
VTS	Vehicle Technical Specifications
ω	Angular velocity

Appendix B: DFMEA

The fast development cycle of the EcoCAR vehicle poses challenges for developing fault detection strategies, since it is not practical to detect and react to every conceivable failure the vehicle could encounter. To make sure the most important failures are considered and remedied, a technique known as Design Failure Mode Effects Analysis (DFMEA) is employed. This technique is a deductive approach, involving the consideration of low level failures and their effects on higher level vehicle operation.

The DFMEA technique serves as a strategy for ranking the importance of failures as well as the effectiveness of specified remedial action. This is done through the use of Risk Priority Numbers (RPN's). Using qualitative factors of Severity, Occurrence, and Detection, this RPN is calculated as the product of all three as shown in Figure B.1. These qualitative factors can be tuned quantitatively by pursuing more detailed failure analysis techniques, especially to predict the likelihood of occurrence.

Failure Mode and Effects Analysis Worksheet

General System Function		System Function		Detailed Function	
#	Item and Function/ Requirements	Potential Failure Mode	Potential Effect(s) of Failure	Severity	Occurrence
1	EPB provides emergency braking in the event of primary brake failure	EPB does not engage when requested	Vehicle fails to stop	X	EPB switch failure, controller error
2	EPB secures the vehicle on a 20% grade when stopped	EPB engages without request	Driver starts	X	Controller error, Controller error, unexpected vehicle weight
4	When stopped, Parking Brake button toggles the EPB	EPB does not engage when requested	Vehicle rolls unexpectedly	X	EPB switch failure, controller error
5	When driving (>0 mph), Parking Brake button engages with gradually increasing torque if held down	Parking brake engages fully while driving	Driver starts/ loss of control	X	Controller error
6	Actual parking brake state displays on dash using stock Parking Brake button	Parking brake state not displayed to driver	EPB state not known to driver	X	Digital output failure

#	Item and Function/ Requirements	Potential Failure Mode	Potential Effect(s) of Failure	Severity	Occurrence	Current state	Enhanced state
				RPN	Recommended Action		
				Severity	Detec tion	Detec tion	
1	EPB provides emergency braking in the event of primary brake failure	EPB does not engage when requested	Vehicle fails to stop	X	EPB switch failure, controller error	6	2 24 Notify driver
2	EPB secures the vehicle on a 20% grade when stopped	EPB engages without request	Driver starts	X	Controller error, Controller error, unexpected vehicle weight	8	2 5 Detect unexpected braking, notify driver
4	When stopped, Parking Brake button toggles the EPB	EPB does not engage when requested	Vehicle rolls unexpectedly	X	EPB switch failure, controller error	4	2 7 Increase linear actuator current, warn driver
5	When driving (>0 mph), Parking Brake button engages with gradually increasing torque if held down	Parking brake engages fully while driving	Driver starts/ loss of control	X	Controller error	9	2 6 Detect unexpected braking, notify driver
6	Actual parking brake state displays on dash using stock Parking Brake button	Parking brake state not displayed to driver	EPB state not known to driver	X	Digital output failure	4	2 7 Warn driver if applied torque is resisted by EPB

Occurrence	Severity	Detection
None	1 - None	1 - None
Low	2 - Minor	2 - Minor
Medium	3 - Moderate	3 - Moderate
High	4 - High	4 - High
Very High	5 - Very High	5 - Very High
Extremely High	6 - Extremely High	6 - Extremely High
Unintended	7 - Unintended	7 - Unintended
Unknown	8 - Unknown	8 - Unknown
N/A	9 - N/A	9 - N/A
Not Applicable	10 - Not Applicable	10 - Not Applicable

Figure B.1: DFMEA example: electronic park brake

Appendix C: V-MODEL DESIGN PROCESS

The OSU EcoCAR team performed their vehicle design according to the V-Model, diagrammed in Figure C.1. This model describes a method for progressing from system requirements to implementation and testing. The V shape is intended to demonstrate a horizontal connection between design steps and test validation steps. When designing an algorithm, a test plan should be written to ensure that this algorithm meets design requirements. To validate the implemented software, the test plan should be followed and traced to the original requirements.

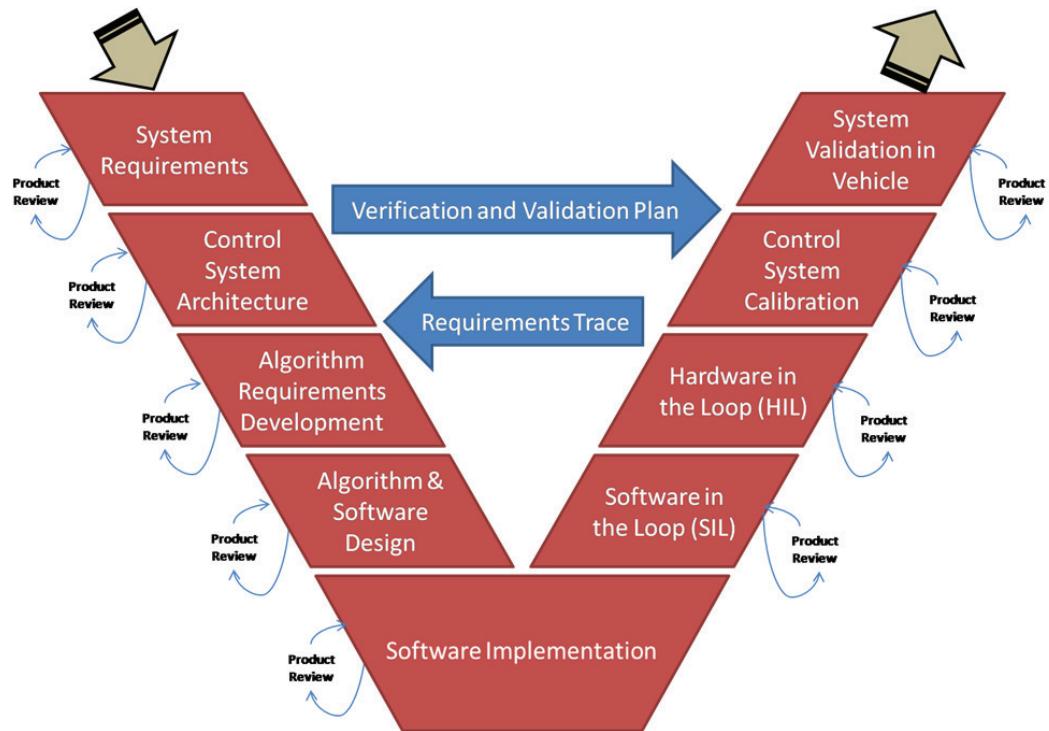


Figure C.1: V-model design process

Bibliography

- [1] *Offshore Reliability Data Handbook*, 4th ed. OREDA Participants, 2002.
- [2] Argonne National Labs, “EcoCAR 2 request for proposals,” Online distribution, Nov. 2010.
- [3] K. Bayar, B. Bezaire, B. Cooley, J. Kruckenberg, E. Schacht, S. Midlam-Mohler, and G. Rizzoni, “Design of an extended-range electric vehicle for the EcoCAR challenge,” *ASME Conference Proceedings*, vol. 2010, no. 44090, pp. 687–700, 2010.
- [4] R. B. Cooley, “Engine selection, modeling, and control development for an extended range electric vehicle,” Master’s thesis, The Ohio State University, 2010.
- [5] Delphi, “Worldwide emissions standards: Passenger cars and light duty vehicles,” Distributed booklets and online distribution, 2011, <http://delphi.com/pdf/emissions/Delphi-Passenger-Car-Light-Duty-Truck-Emissions-Brochure-2010-2011.pdf>.
- [6] A. Dhaliwal, S. Nagaraj, and S. Jogi, “Hardware-in-the-loop testing for hybrid vehicles,” *Evaluation Engineering*, pp. 30–34, Nov. 2009.
- [7] M. Ehsani, Y. Gao, and A. Emadi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory, and Design*, 2nd ed. CRC Press, Sept. 2009.
- [8] Y. Hori, Y. Toyoda, and Y. Tsuruoka, “Traction control of electric vehicle based on the estimation of road surface condition,” *IEEE Transactions on Industry Applications*, vol. 34, no. 5, pp. 1131–1138, Sept. 2002.
- [9] Hybrid Committee, “Electric vehicle and plug in hybrid electric vehicle conductive charge coupler,” *Society of Automotive Engineers Technical Standards*, 2010, J1772.

- [10] K. Koprubasi, “Modeling and control of a hybrid-electric vehicle for drivability and fuel economy improvements,” Ph.D. dissertation, The Ohio State University, 2008.
- [11] M. Liukkonen, A. Hentunen, and J. Suomela, “Validation of quasi-static series hybrid electric vehicle simulation model.” Lille: IEEE Vehicle Power and Propulsion Conference, Sept. 2010, pp. 1–6.
- [12] J. M. Morbitzer, “High-level modeling, supervisory control strategy development, and validation for a proposed power-split hybrid-electric vehicle design,” Master’s thesis, The Ohio State University, 2005.
- [13] C. Musardo, G. Rizzoni, Y. Guezenec, and B. Staccia, “A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management,” *European Journal of Control*, no. 11, pp. 4–5, 2005, fundamental Issues in Control, Special Issue.
- [14] S. Nabi, M. Balike, J. Allen, and K. Rzemien, “An overview of hardware-in-the-loop testing systems at Visteon,” *SAE Testing and Instrumentation*, no. 2004-01-1240, 2004.
- [15] G. Paganelli, M. Tateno, A. Brahma, G. Rizzoni, and Y. Guezenec, “Control development for a hybrid-electric sport-utility vehicle: Strategy, implementation, and field test results.” Arlington, VA: American Control Conference, June 2001.
- [16] G. Paganelli, T. M. Guerra, S. Delprat, J.-J. Santin, M. Delhom, and E. Combes, “Simulation and assessment of power control strategies for a parallel hybrid car,” *Journal of Automobile Engineering*, vol. 214, no. 7, pp. 705–717, 2000.
- [17] R. J. Patton, “Fault-tolerant control systems: The 1997 situation,” in *IFAC SAFEPROCESS 1997*, R. Patton and J. Chen, Eds., IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes. Kingston Upon Hull, UK, 1997, pp. 1033–1054.
- [18] A. Radwan, G. Rizzoni, and A. Soliman, “Fault library generation using model-based approach,” The Ohio State University, Tech. Rep. 739386-6, June 2001.
- [19] D. Ramaswamy, R. McGee, and S. Sivashankar, “A case study in hardware-in-the-loop testing: Development of an ECU for a hybrid electric vehicle,” Software/Hardware Systems, Systems Engineering, Advanced Electronics Packaging, and Electromagnetic Compatibility (EMC). SAE Technical Paper Series, March 2004.
- [20] E. J. Schacht, B. Bezaire, B. Cooley, K. Bayar, and J. Kruckenberg, “Addressing drivability in an extended range electric vehicle.” Detroit, MI: SAE World Congress & Exhibition, April 2011, no. 2011-01-0911.

- [21] H. Schuette and M. Ploeger, “Hardware-in-the-loop testing of engine control units - A technical survey,” *SAE Technical Papers*, no. 2007-01-0500, 2007.
- [22] D. J. Smith, *Reliability, Maintainability, and Risk*, 6th ed. Newnes, March 2001.
- [23] R. Somogye, “An aging model of Ni-MH batteries for use in hybrid-electric vehicles,” Master’s thesis, The Ohio State University, 2004.
- [24] Truck Bus Control And Communications Network Committee, “Recommended practice for a serial control and communications vehicle network,” *Society of Automotive Engineers Technical Standards*, 2010, J1939.
- [25] US Department of Defense, “Reliability prediction of electronic equipment,” Tech. Rep., Feb. 1995, MIL-HDBK-217F.
- [26] V. Winstead and I. Kolmanovsky, “Estimation of road grade and vehicle mass via model predictive control,” *Proceedings of 2005 IEEE Conference on Control Applications, 2005*, pp. 1588 – 1593, 2005.