

9-2011

Development and Implementation of a Fault Mitigating Control System for a Biodiesel Plug-In Hybrid Electric Vehicle for the EcoCar: The NeXt Challenge Competition

Sean Christopher Carter
Embry-Riddle Aeronautical University - Daytona Beach

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Mechanical Engineering Commons](#)

Scholarly Commons Citation

Carter, Sean Christopher, "Development and Implementation of a Fault Mitigating Control System for a Biodiesel Plug-In Hybrid Electric Vehicle for the EcoCar: The NeXt Challenge Competition" (2011). *Dissertations and Theses*. 35.
<https://commons.erau.edu/edt/35>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in Dissertations and Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

Development and Implementation of a Fault Mitigating Control System for a Biodiesel
Plug-In Hybrid Electric Vehicle for the EcoCar: The NeXt Challenge Competition

By

Sean Christopher Carter

A Thesis Submitted to the College of Engineering Department of Mechanical
Engineering
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Mechanical Engineering

Embry-Riddle Aeronautical University
Daytona Beach, Florida
September 2011

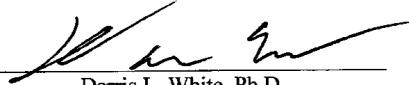
Development and Implementation of a Fault Mitigating Control System for a Biodiesel Plug-In Hybrid Electric Vehicle for the EcoCAR: The NeXt Challenge Competition

by

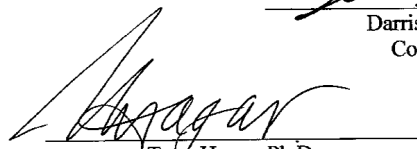
Sean Christopher Carter

This thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. Darris L. White, Associate Professor, Daytona Beach Campus, and Thesis Committee Members Dr. Tony Hagar, Associate Professor, Daytona Beach Campus, and Dr. Marc D. Compere, Assistant Professor, Daytona Beach Campus, and has been approved by the Thesis Committee. It was submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

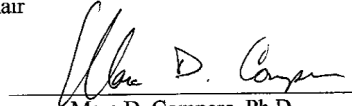
Thesis Review Committee:



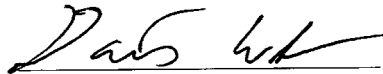
Darris L. White, Ph.D.
Committee Chair



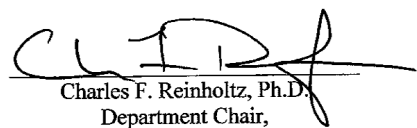
Tony Hagar, Ph.D.
Committee Member



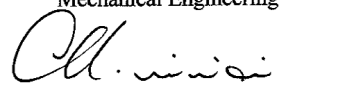
Marc D. Compere, Ph.D.
Committee Member



Darris L. White, Ph.D.
Graduate Program Chair,
Mechanical Engineering



Charles F. Reinholtz, Ph.D.
Department Chair,
Mechanical Engineering



Maj Mirmirani, Ph.D.
Dean, College of Engineering



Robert Oxley, Ph.D.
Associate Vice President of Academics

9-19-2011

Date

Acknowledgements

I want to thank the team for all of their work. The vehicle would not even be running or come to fruition if it were not for them. I specifically want to thank Zachary Karstetter, Brian Harries, and Brandon Smith for all of their efforts. They kept the team together through all of the rough times and helped make all of those late nights in the lab all the more fun! I also want to thank Dr. Liu for giving me the opportunity to work with his Systems Engineering class. The systems engineering students deserve praise for all of their hard work on procuring and developing all of the documentation that kept the project moving. I personally want to thank my friends and family that have supported me throughout the years of EcoCar. If it were not for them, I would not have pursued the perfection I sought to achieve with the EcoEagles vehicle. I also want to thank Dr. Hagar for the motivation to keep going when times were tough. Dr. Compere was also a great help in giving fresh ideas that were never considered. Last, but not least, I want to thank Dr. White. He is the one who pushed me to become the person I am now and gave me the opportunity to gain valuable experience for a future job with GM.

Abstract

Researcher: Sean Christopher Carter

Title: Development and Implementation of a Fault Mitigating Control System for a Biodiesel Plug-In Hybrid Electric Vehicle for the EcoCar: The NeXt Challenge Competition

Institution: Embry-Riddle Aeronautical University

Degree: Master of Science in Mechanical Engineering

Year: 2011

The automotive industry is continuously developing, and with it hybrid vehicle technology is a growing field of interest. The design of the electric vehicle is a pressing matter and grows in complexity with new powertrain components such as power inverters and transmission systems that use electric motors. As a control system develops, the architecture always comes back to systems engineering documentation to find safety protocols, solutions to problems through fault testing, and validating and verifying the control architecture throughout the whole process. Testing and evaluation plans are required more than ever and are constantly being updated and implemented in today's automotive production standards. The paper discusses the development and implementation of the control system through the use of systems engineering of a hybrid vehicle as part of a competition called EcoCar: The NeXt Challenge.

Table of Contents

	Page
Thesis Review Committee.....	ii
Acknowledgements.....	iii
Abstract.....	iv
List of Figures.....	vii
List of Acronyms.....	ix
Chapter	
I. Introduction.....	1
Significance of Study.....	1
Statement of the Problem.....	2
Purpose.....	4
II. Review of the Relevant Literature.....	6
Systems Engineering.....	6
Design Failure Modes and Effects Analysis.....	7
Fault Tree Analysis.....	9
Validation & Verification.....	11
History of Advanced Vehicle Technology Competitions.....	13
Summary.....	15
III. Control System Development and Implementation.....	16
1) Vehicle Control System Overview.....	16
1. Control Systems Development using Systems Engineering.	20
2. Gateway – Host Code Development.....	23
3. Gateway – Field Programmable Gate Array Development..	31

2) Vehicle Control System Implementation.....	39
1. Database and Communication Development.....	39
IV. Results.....	45
EcoCar Control System Performance.....	45
Systems Engineering Results.....	49
V. Discussion, Conclusions, and Recommendations.....	53
Discussion.....	53
Conclusions.....	55
Recommendations.....	55
References.....	57
Appendices.....	60
A. ASME 2011 5 th International Conference on Energy Sustainability Paper.....	60
B. ASME 2011 International Mechanical Engineering Congress & Exposition Paper.....	66

List of Figures

Figure	Page
1. EcoCar Timeline and Deliverables.....	3
2. Initial Vehicle Technical Specifications.....	5
3. Example of DFMEA Documentation (Function, Failure Mode, Effect, Severity).....	7
4. Example of DFMEA Documentation (RPN, Occurrence, Detection, etc.).....	8
5. Example of DFMEA Documentation (Actions and Responsibility).....	9
6. Minor Fault for Fault Tree Analysis.....	10
7. Major Fault for Fault Tree Analysis.....	10
8. Validation and Verification process.....	12
9. General Overview of Control Systems.....	17
10. The Battery Pack, DDE, and BRUSA Charger.....	18
11. 1.3L Turbo Diesel GM Engine.....	19
12. The sbRIO – 9642 (left) and sbRIO – 9602 (right).....	20
13. Front Panel of the Gateway Host Code.....	24
14. Gateway Host Code FPGA Initialization.....	25
15. Gateway Host Code Message ID List Configuration.....	25
16. Gateway Host Code Memory Write Loop.....	27
17. Gateway Host Code FPGA Check.....	28
18. Gateway Host Code Driver Panel Management Initialization.....	28
19. Gateway Host Code Driver Panel Management Message Bundling.....	29
20. Gateway Host Code Driver Panel Management Periodic Message Handling....	30
21. Gateway Host Code Driver Panel Port Control.....	30

22. Gateway FPGA Front Panel Interface.....	32
23. Gateway FPGA CAN, Boolean, and Port Initialization.....	33
24. Gateway FPGA VI Memory and First-In / First-Out (FIFO) Configuration....	34
25. Gateway FPGA Write to Memory Loop.....	35
26. Gateway FPGA CAN Module Communication Restart.....	35
27. Gateway FPGA CAN Read Loop.....	36
28. Gateway FPGA Memory Checker Loop (simple).....	37
29. Gateway FPGA Memory Checker Loop (complex).....	37
30. Gateway FPGA CAN Write Loop.....	38
31. Gateway FPGA Driver Panel Notification Port Control Loop.....	38
32. Supervisory Control Unit FPGA Communication Development.....	40
33. Supervisory Control Unit FPGA Subsystem Control Loop.....	41
34. National Instruments Measurement and Automation Explorer.....	42
35. Vehicle Engine Bay.....	46
36. Vehicle Exhaust System (DPF on left).....	47
37. IDEA Computer with Student Designed Bezel.....	48
38. EcoEagles Production Vehicle Readiness.....	50
39. Vehicle Technical Specifications (VTS).....	52

List of Acronyms

ANL	Argonne National Labs
CAN	Controller Area Network
DFMEA	Design Failure Modes and Effects Analysis
DIS	Daytona International Speedway
DOE	Department of Energy
DPF	Diesel Particulate Filter
DSP	Digital Signal Processing
ECM	Engine Control Module
EPA	Environmental Protection Agency
FEA	Finite Element Analysis
FIFO	First In / First Out
FMECA	Failure Modes, Effects, and Criticality Analysis
FPGA	Field Programmable Gate Array
FTA	Fault-Tree Analysis
GM	General Motors
GMLAN	General Motors Local Area Network
GW	Gateway
HIL	Hardware-in-the-Loop
IDEA	Intelligent Driver Efficiency Assistant
NI	National Instruments
PTEB	Powertrain Extended Bus

RPM	Rotations Per Minute
RPN	Risk Priority Number
sbRIO	Single Board Reconfigurable Input / Output
SCU	Supervisory Control Unit
SIL	Software-in-the-Loop
TPM	Technical Performance Measures
V&V	Validation and Verification
VCS	Vehicle Control System
VTS	Vehicle Technical Specifications

Chapter I

Introduction

Engineered systems have a functional purpose in response to an identified need and have the ability to achieve some stated operational objective [1]. They are brought into being and operate over a life cycle. These systems begin with a need, and continue until phasing out is required or if the product needs to be disposed. Engineered systems are often composed of subsystems, or development groups that interact with each other. These are the basics of any engineered system and are integrated into many developmental processes in industry.

Significance of Study

System evaluation is the assessment and examination of a system or system element [1]. With system evaluations and assessments, these tools help determine whether or not the system itself is on track and meeting the end goal desired. The evaluations derived from the system are continuous through the product's life cycle and only stop once the product no longer exists. With newly developed technological advancements, there arise new procedures and protocols that have to be developed and evaluated to ensure the safety of use by customers and co-workers alike. The automotive industry is such an example. General Motors (GM), Ford, and Chrysler are some of the many automotive industries that provide luxury cars for the middle class world and are investing more heavily in electrical technology.

Systems engineering has an important role in developing the newest hybrid technology. The process and principles used from systems engineering allow the

automotive industry to grow and develop new technologies quickly, efficiently, and safely. Through the use of design fault mitigation and effects analysis (DFMEA), proper planning can be done to assist in quickly developing a vehicle. The use of fault tree analysis (FTA) allows a vehicle to develop proper safety ratings. Validation and verification (V&V) enables the vehicle to develop efficiently and helps ensure that requirements are met. The EcoCar: The NeXt Competition is an example of the uses of these practices.

Statement of the Problem

Systems engineering plays a vital role in the automotive industry and can be seen in the EcoCar competition. The areas that are focused on for each year are shown in Figure 1 on the next page. This shows the deliverables that were expected of the students from the competition organizers. From a systems engineering viewpoint, these are the milestones of the product over the next three years. Importance is stressed in certain areas of systems engineering to make sure that the vehicles operate correctly and safely for each team. The aspects of systems engineering that are important to the competition are the validation and verification of the results obtained through the design process, the fault-tree insertion into the different aspects of the project to ensure safe operation and safety of the driver, and design failure modes and effects analysis for continuous change and observation of the high risk priority items. These are the problems faced by every team through the entire three years of the competition so that each team can develop safe vehicle architectures.

During year two of development, the systems engineering process was held back when certain problems starting occurring during vehicle development for the EcoEagles. The control system was underdeveloped and was causing problems when trying to properly validate and verify the subsystems. The fault tree analysis was not helpful and the DFMEA documentation needed updating. This was not a fault of the EcoEagles or any sponsor, but merely a lack of full understanding of how the vehicle architecture properly worked. The transmission and engine were two subsystems that were never meant to be together and the EcoEagles had to discover a means to incorporate the technology.

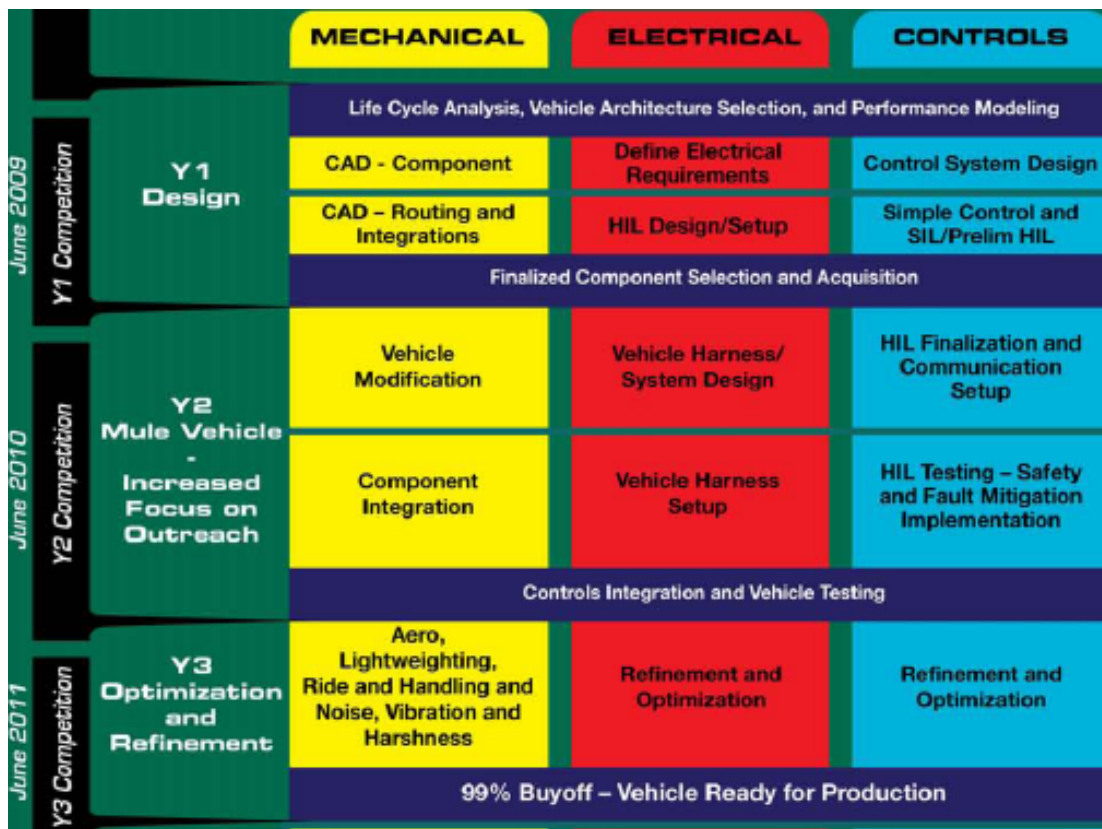


Figure 1: EcoCar Timeline and Deliverables [2]

Purpose

Control system development is complex, especially with the newer hybrid technologies being produced by the automotive companies in today's industries. Systems engineering is a beneficial process to help develop and implement such a complex system into a vehicle and have the vehicle operate correctly and safely. The EcoCar competition required a complex control system and had a lot of preliminary planning and documentation developed to help support a secure architecture.

Some of the DFMEA, FTA, and V&V in place towards the end of year two helped in understanding the problems the EcoEagles faced, the team from Embry-Riddle Aeronautical University. There was not enough information on the transmission and engine to properly develop the control architecture. Through the efforts of GM, the systems engineering students, and the EcoEagles the vehicle documentation could be properly updated. From the end of year two and the beginning of year three, the systems engineering principles became vital to the EcoEagles success. The intention of this paper is to go into detail about the EcoEagles control system development and implementation through the use of systems engineering tools. The goal is to also discuss the fault mitigation incorporated into the control system and the results from the competition on the success of the systems engineering practices. Figure 2 shows the higher-level requirements that each team was required to improve or meet according to the vehicle technical specifications for the competition.

Specification	Competition Requirements	
EcoCar	Production Vue	Competition Requirement or Target
Accel 0-60	10.6s	≤ 14 s
Accel 50-70	5.7	≤ 10 s
Towing Capacity	680 kg (1500 lbs)	≥ 680 kg @ 3.5% grade, 20 min at 72 kph (45 mph)
Cargo Capacity	0.83 m ³	Height: 457 mm (18 in) Depth 686 mm (27 in) Width 762 mm (30 in)
Passenger Capacity	5	≥ 4
Braking 60-0	38 - 43 m (123 - 140 ft)	< 51.8 m (170 ft)
Mass	1758 kg (3875 lb)	2268 kg (5000 lb)
Starting Time	≤ 2 s	≤ 15 s
Fuel Consumption CAFÉ unadjusted, Combined, Team: UF Weighted	8.3 L / 100 km (28.3 mpgge)	7.4 L / 100 km (32 mpgge)
Charge Depleting Fuel Consumption	N/A	N/A
Charge Sustaining Fuel Consumption	N/A	N/A
Charge Depleting Range	N/A	N/A
Petroleum Use	0.85 kWhr / km	0.77 kWhr / km
Emissions	Tier II Bin 5	Tier II Bin 5
WTW GHG Emissions	250 g / km	224 g / km

Figure 2: Initial Vehicle Technical Specifications

Chapter II

Review of Relevant Literature

Systems Engineering

Hall [1962] asserts that the first attempt to teach systems engineering as we know it today came in 1950 at MIT by Mr. Gilman, Director of Systems Engineering at Bell [3]. Since the discipline's inception, the mission of systems engineering has been to "engineer the system" to meet acquirer/user needs within budget and on schedule [4]. Hall [1962] defined systems engineering as a function with five phases: (1) system studies or program planning; (2) exploratory planning, which includes problem definition, selecting objectives, systems synthesis, systems analysis, selecting the best system, and communicating the results; (3) development planning, which repeats phase 2 in more detail; (4) studies during development, which includes the development of parts of the system and the integration and testing of these parts; and (5) current engineering, which is what takes place while the system is operational and being refined [3]. These steps are similar to the project definition stages, or earlier stages of what is defined as a systems life cycle according to Systems Engineering Standard ISO/IES 15288 [5]. Importantly, it is imperative to integrate program needs, cost, performance, schedule, and risk with the acquisition strategy to obtain the intended program solution [6]. Engineers, especially automotive engineers of future complex systems, face an emerging challenge of how to address problems associated with integration of multiple complex systems [7].

Design Failure Modes and Effects Analysis (DFMEA)

DFMEA, alternatively FMECA [1], is a systematic team driven approach that identifies potential failure modes in a system, product, or manufacturing / assembly operation caused by design or manufacturing / assembly process deficiencies [8]. The overall goal is to find potential failures within the system being designed and to determine the effect, the severity of the failure, how often the failure occurs, how to prevent or manage the failure, and who is responsible for that failure's analysis. The information is then organized, and put into a spreadsheet, shown in figure 3.

Line No.	Item / Function of the Part	Potential Failure Mode (Loss of Function or value to customer)	Potential Effect(s) of Failure	SEV	Class
1	Accelerator Pedal Sensor	Incorrect Accelerator Signal	Erratic Vehicle Performance	8	
2	Brake Pressure Sensor	Incorrect Brake Signal	Erratic Vehicle Performance	7	
3	Transmission Sensors	Incorrect Mode Readings / Incorrect Torque Readings / Incorrect Speed Readings	Erratic Vehicle Performance / Loss of Operation	6	

Figure 3: Example of DFMEA Documentation (Function, Failure Mode, Effect, Severity)

Each item or function discussed should be examined for any potential failure mode that could potentially occur during vehicle operation or even when the boards are simply starting up. Potential effects from the failure also had to be discussed along with the severity of the problem. The severity level of each failure is assigned a rating from one to ten, one being the least severe and ten being the most severe. Depending on product development, or if other problems discovered are more of an issue, the severity rating could change.

Discussion of failures that could commonly happen is a great way to discover and document as many potential failures as possible. These causes are later used in fault

mitigation and testing. This also leads to the discussion of the rate of occurrence, which is a rough estimate of how often the problem may occur on the product. This number is assigned a rating from one to ten, one being least likely to happen and ten being most likely, and could also change based on production progression. Preventative measures to help make the system tolerant of faults and detection to help mitigate any fault that would occur are ways to verify and validate that the failure can be managed safely, and an example of the documentation is shown in figure 4. The detection rating, another important factor for faults, is assigned a rating from one to ten, one being most likely, and ten being least likely to be detected.

The most important column that will constantly change is the risk priority number (RPN). The RPN is a numerical way of determining which fault is most important. The higher the occurrence, severity, and detection rating, the higher the RPN will increase as well. The main goals are to try and reduce the RPN by trying to affect the occurrence of the fault, detect the problems more efficiently before the fault occurs, and by trying to reduce the severity of the problem. All these anomalous situations are collected on a table, and for each fault scenario the RPN is evaluated and recommended actions are suggested to improve the situation [9].

Potential Cause(s) / Mechanism(s) of Failure	O C C	Current Design Controls (Design actions planned or completed to prevent or reduce occurrence of failure, provide details and Best Practices used) Prevention	Current Design Controls (Analytical or physical validation method planned or completed) Detection	D E T	R P N
Poor Electrical Connection / Mechanical Failure of Connection	3	Best Practices for Electrical Routing / Reuse of Proven Designs	Test Electrical Connections and Routing	4	96
Poor Electrical Connection	1	Best Practices for Electrical Routing	Test Electrical Connections and Routing	2	14
Poor Electrical Connection	3	Best Practices for Electrical Routing	Test Electrical Connections and Routing	3	54

Figure 4: Example of DFMEA Documentation (RPN, Occurrence, Detection, etc.)

Ultimately, the actions area of the DFMEA documentation, shown in figure 5, is determined and then modified later as the failure is tested and validated once the project reaches that stage in the development process. Responsibility is truly shared throughout the project, but a group or subgroup is in charge of making sure that the failure is properly detected or prevented. The group that is responsible is normally determined through discussion and what makes common sense. The continuous updating of the DFMEA documentation is responsible for a living document that keeps track of the fault mitigation progress on product development.

Recommended Action(s)	Responsibility
HIL Fault Insertion Testing / Display Warning, shut down vehicle if accelerator signal is lost.	Electrical/HIL/Controls Team
HIL Fault Insertion Testing / Display Warning / reduce available power if brake signal is lost.	Electrical/HIL/Controls Team
Display Warning / Monitor engine, motors speeds	Electrical/Controls Team

Figure 5: Example of DFMEA Documentation (Actions and Responsibility)

Fault Tree Analysis (FTA)

Fault tree analysis (FTA) is a top down, deductive failure analysis in which an undesired state of a system is analyzed using Boolean logic to combine a series of lower-level events [10]. The process involves introducing failures into a system to yield results. The actual faults can be inserted into the system to determine reliability, but more often than not the faults being tested are possible causes and not actual. The false occurrences introduced into the system allow detection of improper function and the ability to

properly take action without risking safety or damage. With FTA, the process can be used to evaluate design alternatives and to establish performance-based design on the faults instigated [11]. The faults put into the system can range from minor to critical and obtain results of equal criticality. By introducing minor faults into a system, it may lead to the discovery of a major fault that could occur. Allowing major faults into the system also improve the ability of detection by noticing minor faults that potentially occur as a result. A lot of the fault trees created stem from the DFMEA documentation. FTA also helps by finding other potential causes for the other causes that were discovered, enabling the DFMEA to expand and consider more possible failures. Some of these failures are shown in figure 6 and 7.

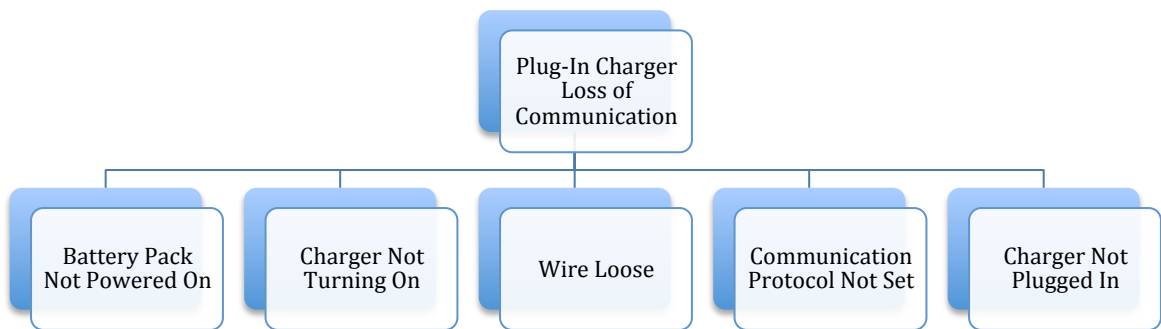


Figure 6: Minor Fault for Fault Tree Analysis

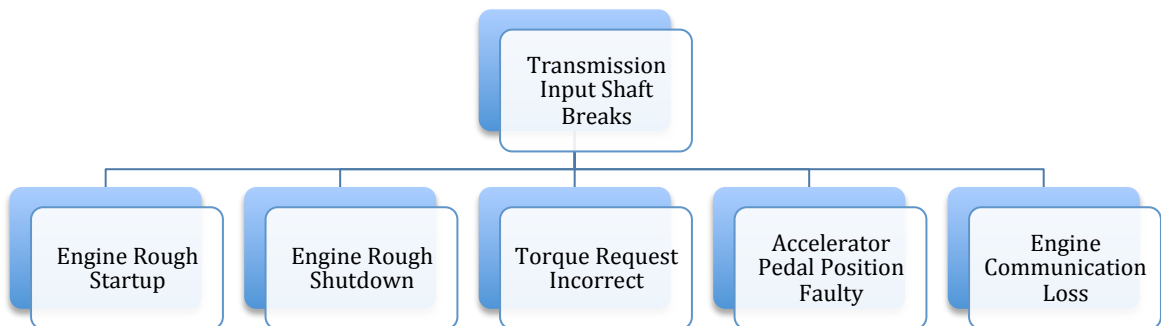


Figure 7: Major Fault for Fault Tree Analysis

All of those causes would then be tested and inserted into each respective system to see how well the safety protocols perform and creating a fault mitigating system. These tests could potentially lower the RPN and help improve the overall safety of the product development process. FTA further helps to overcome some of the limitations such as computational time, expertise necessary for fault tree analysis and repeatability of the analysis [12]. This system engineering development tool plays a crucial role in ensuring the safety of the product and the consumer.

Validation and Verification (V&V)

Validation is the process of making sure the system fulfills its intended purpose [13] or that the right product is developed. Verification is making sure the system meets specifications [13] or that the product is built correctly. V&V is an indispensable step when developing a product. V&V is a continuing process of looking to the original design criterion and determining that the design process and product meet the requirements stated (verification) and meets the customer's needs (validation). For each step of the development phase, the project goes through and makes sure that the newest addition to the product meets the requirements stated. Even if requirements and model validation result in a design that should meet the ultimate need, the steps of verification and system validation are required to prove the as-built system in fact does meet those requirements and satisfies the ultimate need [14].

The V&V process is incorporated into every aspect of the development process. From start to finish, the product is analyzed and critiqued as subsystems and subassemblies are introduced. As the development process of the life cycle of the project

progresses, the validation and verification process stays important throughout product development. System software testing must include stress testing and fault injection in a suitable simulation environment to determine the limits of capability and search for hidden flaws [14]. The cycle begins with integration, testing, and verification, and then the process goes into system verification and validation and finally ending on operations and maintenance. Figure 8 represents the basic idea of validation and verification and the involvement with the life cycle process.

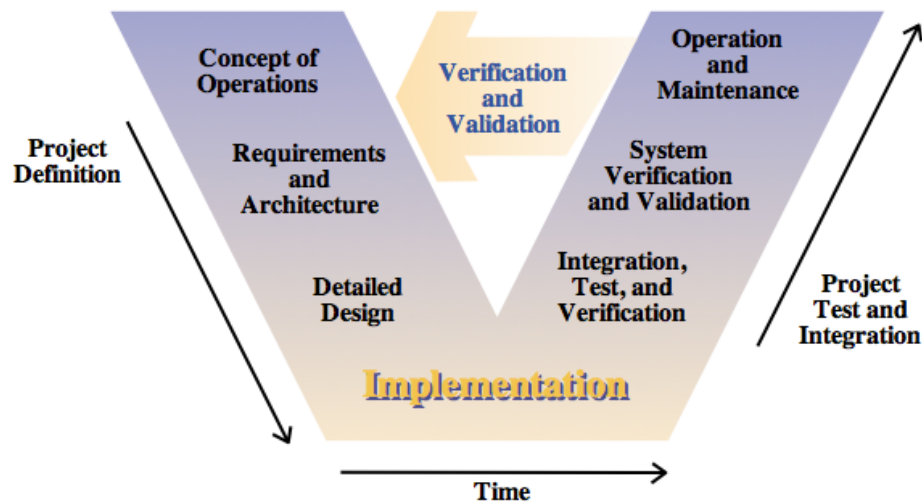


Figure 8: Validation and Verification process [15]

The figure expresses the complete product life cycle in the form of what is known as the “V” model. The left side of the “V” is the design aspect of the product development, moving from the top to the bottom. The right side of the “V” is the integration of the systems moving from the bottom and towards the top. The figure shows that products are designed in a hierarchy from the top down to the smallest of subsystems and then integrated and tested from these subsystems until the overall system is eventually tested and completed. The arrows going from the right to the left express the validation and verification of the project as the systems are integrated.

Without the DFMEA documentation and the FTA, V&V has no starting place. The DFMEA, FTA, and V&V areas of systems engineering need each other in order to properly develop a product. The three topics together provide necessary documents that allow the product development to continuously be improved. Many governing organizations, such as the United States Department of Energy (DoE) Advanced Vehicle Technology Competition (AVTC) and Environmental Protection Agency (EPA), develop, define, and disseminate information, requirements, and testing and evaluation procedures that affect how car manufacturers, like GM, design, build, test, evaluate, manufacture, and monitor their vehicles [16]. The United States DOE AVTC is a great example of incorporating these systems engineering tools into a project that involves the automotive industry.

History of Advanced Vehicle Technology Competitions (AVTC)

The AVTCs have been a part of the DOE and Argonne National Labs (ANL) since 1987 [17]. They have sponsored over 45 AVTCs over the past twenty-four years [17]. These competitions accelerate the development and demonstration of technologies of interest to DOE and the automotive industry while providing the automotive industry with a new generation of engineering leaders with highly desirable experience [17]. The competitions in order from the earliest to the most recent are Methanol, Natural Gas, Ethanol, Propane, Sunrayce, HEV, FutureCar, FutureTruck, Challenge X, EcoCar, and the newest competition EcoCar 2 [18]. Each competition is different in length, but the goals and purpose are the same. Each team participating is required to improve the efficiency of the vehicle and maintain consumer acceptability.

In past competitions, the automotive development played an important role in showing the different improvements that could be made to increase efficiency and reduce overall petroleum use in vehicles. In FutureTruck 2000, a 13% improvement was attained in on-road fuel efficiency (MPGE), and a 26% reduction was attained in greenhouse gas emissions, compared with the stock Chevrolet Suburban [19]. In FutureTruck 2003 the greenhouse gas emissions of eight student vehicles were less than those of the control vehicle, with West Virginia University reducing GHG emissions by an incredible 48% [19]. As these developments progressed, so did the automotive industry and the future AVTCs.

The competition that recently ended is EcoCar: The NeXt Challenge. This AVTC involved the past three years and involved sixteen universities from the United States and Canada. Each team submitted different vehicle architectures for the competition and was expected to develop the vehicle through computer-aided drafting, SIL and HIL development, and safe electrical development and implementation. Throughout the three-year competition, the Virginia Tech team achieved their goals of a fuel-efficient vehicle at 81.9 miles per gallon gasoline equivalent, or 70 percent over the stock vehicle [20]. Overall, the DOE sponsors these competitions with the main goal in mind to train new engineers and make contributions that will help keep the North American automotive industry competitive in the global marketplace, which is increasingly adopting fuel-efficient designs [21].

Summary

AVTCs have helped improve the automotive industry and train future engineers for the workforce. These competitions were a success due to the amount of planning and work done by the DOE, GM, and ANL. Through them, guidelines and requirements were set to help keep all of the teams on track and to provide an example of what the automotive industry does when developing a vehicle. This could not have been done without the basis of proper systems engineering implementation and development.

DFMEA provides a great way to maintain documentation on safety critical systems for the AVTC competition. The documents provide a means for new students to understand the functionality of the subsystems and how to meet the next milestone for the vehicle development process. This leads to FTA and how the process helps update the DFMEA documentation as the teams develop the hybrid vehicle technologies. Designing a vehicle using FTA helps provide a safer environment for the driver and shows how well the teams are prepared and have thought through the designing process. Using V&V throughout the whole process of vehicle development keeps testing a priority and making sure requirements are met. These tools are important and necessary for these vehicles to run correctly and efficiently when it comes to test them at the competition. Systems engineering has played a key role in ensuring efficient, safe, and well maintained products, and the students learn to use these tools to develop a fully functioning vehicle.

Chapter III

Control System Development and Implementation

Vehicle Control System (VCS) Overview

A VCS is a major part of the vehicle development process and a good portion of the DFMEA documentation. For the competition, the VCS is split into two separate parts. There is the stock VCS that GM has developed and there is a student part for each university to develop and integrate into the vehicle. Each team had to integrate a new battery pack and a new powertrain system into the project vehicle. The EcoEagles chose to integrate an A123 Lithium-Ion Iron Phosphate battery system, a GM 1.3L turbo diesel engine, and a GM 2-Mode transmission. These choices were among the few that were given to every team to develop their vehicle architecture.

The EcoEagles VCS needed to communicate with the stock VCS and be able to control each sub system separately. To do this, the VCS is comprised of four Controller Area Network (CAN) busses. These busses are General Motors Local Area Network (GMLAN), Powertrain Extended Bus (PTEB), ERAU High Speed (HS), and ERAU PTEB [22]. The vehicle has GMLAN and PTEB as stock busses on the vehicle and the team had to add the EcoEagles HS and PTEB busses to help isolate controlled components, as shown in figure 9. GMLAN and PTEB are expressed in figure 9 as stock VCS HS CAN and stock VCS PTEB CAN respectively. These isolated control subsystems are the engine control module (ECM) and the battery pack control module (BPCM). Each subsystem controls what it is rightfully named and needs isolation from one another to ensure no cross communication could potentially cause damage. The subsystems transmit messages over the CAN busses that could be received by one

another and potentially cause damage. The two boards also shown in the figure are the supervisory control unit (SCU) and the gateway (GW).

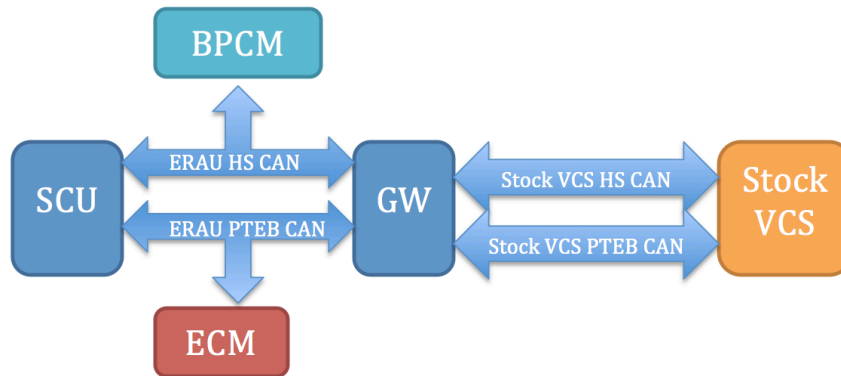


Figure 9: General Overview of Control Systems

Corresponding to figure 9, each subsystem shown has a specific purpose.

- **Supervisory Control Unit (SCU)** – The SCU is in charge of controlling subsystems within the vehicle and the GW is in charge of the isolation and communication management. The SCU’s main goal is to control the transmission, battery, and engine systems parallel to another to ensure that each is properly operating. The SCU also controls subsystems not shown such as the fuel pump, vehicle throttle control, and the urea injection system.
- **Gateway (GW)** – The GW is in charge of the four busses shown in figure 9: ERAU PTEB CAN, ERAU HS CAN, Stock VCS PTEB CAN, and Stock VCS HS CAN. The GW makes sure that none of these systems can interfere with each other and to ensure strong communication between the SCU and each subsystem. The main goal for the GW is to properly isolate each respective subsystem from interfering and potentially causing damage to one another.

- **Battery Pack Control Module (BPCM)** – The battery pack is a Lithium-Ion Iron Phosphate battery pack designed and developed by the company A123. The battery pack had a voltage of 330V and was capable of 12.8 kWhrs of energy. The battery pack contained four modules in series, but also separated with a manual disconnect switch as a safety precaution and requirement for the vehicle shutdown procedure. This battery pack was also connected to a charger produced by the company BRUSA and was capable of automatically controlling the charging process once plugged in. The EcoEagles designed a distribution and disconnect enclosure (DDE) to manage these high voltage systems. A picture of the pack is shown in figure 10.

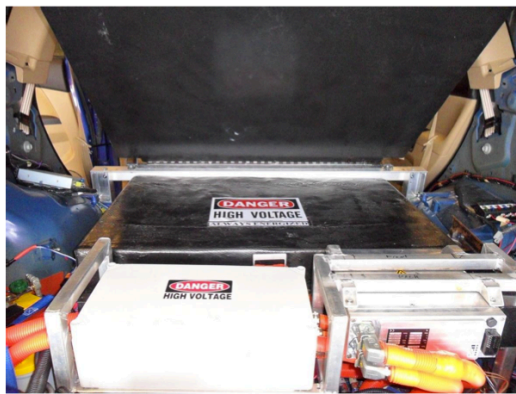


Figure 10: The Battery Pack, DDE, and BRUSA Charger

- **Engine Control Module (ECM)** – The engine is treated a lot like a black hole in space. This part of the EcoEagles control system relies on information that is given to the team, but not so much what is sent to the engine. The engine itself is a 1.3L turbo diesel engine designed and manufactured by GM for the Vauxhall Astra in the European automotive market. The engine is capable of producing 60 kWhrs of power and will be fueled using B20, which is a combination of 80%

regular diesel fuel and 20% biodiesel fuel manufactured on campus. The engine can be seen in figure 11.



Figure 11: 1.3L Turbo Diesel GM Engine

- **SCU and GW Control Boards** – The boards that control the EcoEagles VCS are two boards from National Instruments (NI). The supervisory control unit (SCU) is a single-board reconfigurable input / output 9642 (NI sbRIO – 9642) [23]. The gateway (GW) is a NI sbRIO – 9602 [24]. The difference between the boards is the port configurations, where the sbRIO – 9642 has analog input and output capabilities and the sbRIO – 9602 only has digital input and output. As mentioned in the name, they are both reconfigurable, which allows for rapid prototyping capabilities and faster development for the control architecture. They are both shown in figure 12. Both boards were programmed in LabVIEW 2009 with patch f3 prior to service pack 1 [25]. LabVIEW is a unique way of programming that uses a graphical interface and translates the user’s graphical representation into C-Code the boards can understand. The EcoEagles used LabVIEW throughout the control architecture development process and utilized a lot of the tools that the program had to offer.

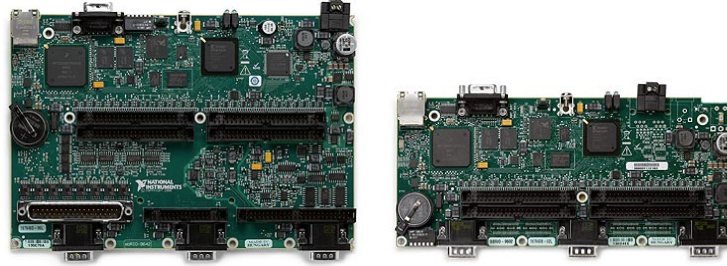


Figure 12: The sbRIO – 9642 (left) and sbRIO – 9602 (right)

- **Vehicle Control System (VCS)** – The purpose of the VCS is exactly what the name implies. This system is the rest of the vehicle and the controllers that GM has created to control each subsystem on the vehicle. The VCS is responsible for controlling and reporting the typical vehicle activity that would happen from everyday driving as selected subsystems are monitored by the SCU.

Gateway development during year two of the competition along with field programmable gate array (FPGA) development was a main topic of concern. As vehicle development progressed, work on the SCU database, FPGA, and communication were main topics of concern for year three of the EcoCar competition. The next sections will go into detail how these boards, the control architecture, and the systems engineering principles came together to develop a more stable control system in little under a year for year three of the EcoCar competition.

Control Systems Development using Systems Engineering

The GW and SCU were developed utilizing the “tools” mentioned as DFMEA, FTA, and V&V. By using these tools, a system could be developed efficiently with little risk and effectively. The process first starts by developing the DFMEA documentation in

a basic form. More detail goes into the DFMEA documentation as the project development progresses. Conceptually, the document contains all of the functions and hardware and the possible failures of each item. The GW and SCU were discussed as having communication and control potential for failure and were given an initial rating for each subsequent possible failure.

One example would be the control over the input shaft of the transmission. The system requirements involved with the input shaft were allocated and identified early on. The team had to make sure that the input shaft was correctly controlled. As the system was investigated further, possible failure modes were identified but not all. These failures would later then be used for FTA. After some of the failures were identified, possible effects and causes were determined. This was discussed to find a good means of identifying the failure when the problem occurred. The failures were then discussed to find out how to detect each one. The means of detecting each failure is key to mitigating or properly tolerating the problem. Each failure is then given a severity, detection, and occurrence rating. As testing and development progresses through the use of V&V, the DFMEA documentation will continuously change and hopefully to reduce the RPN.

For example, the input shaft initially had the occurrence rating set to a critical rating between the numbers of 8-10. This is the high range that the EcoEagles determined as a critical problem. The minor rating would range from 1-3 and the major range would be 4-7. The detection rating and severity rating are similar. Both of these were set to a critical rating from 8-10. This ultimately gave the input shaft failure a relatively higher RPN and was noted as a high priority item in the DFMEA documentation.

As development progress on the vehicle, the failure finally occurred for the input shaft and was broken for the first time. Discussion led to a problem with the control architecture and with this feedback led to the addition of a control architecture change to prevent this fault from occurring again. The fault was now easier to detect and was able to reduce in occurrence because of this change so the ratings for each went down respectively within the DFMEA documentation. The EcoEagles still continued to label this as a high priority item in respect to RPN, but the overall number was reduced through this discovery and testing.

Unfortunately, the failure occurred again and led into another discussion of the reason for the failure. This time the discovery was the engine was not being controlled properly during a procedure required for proper vehicle operation. The failure was discovered after testing through the fault tree analysis designated from the DFMEA. The failure in question was not actually listed within the DFMEA documentation and was further updated with this new possible failure. The control architecture was changed once again to properly mitigate this failure from occurring. This led to the team lowering the RPN number once again and almost reducing the failure from a critical state to a major state, but the failure was still a high priority item and the team would never know if all faults were discovered.

This was a continuous process that occurred all over the vehicle during vehicle development. The team would test for potential failures, or actual failures, and then update the DFMEA documentation if the system was not validating it was built correctly or verifying that it was built to the expectations required of the system. Feedback is a critical step in developing any system and is a crucial part of V&V. Testing and

evaluation continued until each system and subsystem was validated to work as designed and then verified to meet the requirements for each system. DFMEA, FTA, and V&V played this role throughout the EcoEagles vehicle development. Without these development tools, the problems would continue to occur as vehicle development continued and potentially increase project development time. The next few sections will discuss more in detail some of the faults that were occurring with the GW and the SCU coded architectures and what was done to potentially lower the RPN for each system.

Gateway – Host Code Development

The best way to describe the gateway conceptually is a lot like a bouncer at a nightclub. The gateway allows messages to pass through in either direction but only if the message ID is on the message list. Some of the messages are only allowed to pass from the stock VCS to the EcoEagles HS, while others are allowed to pass freely from EcoEagles HS to either stock VCS bus HS or PTEB. The list is regulated by the controlled variables that need to be handled by the SCU. If the engine messages need to be modified before being sent to the vehicle, the GW will make sure that the SCU is the only controller that receives the message prior to being sent to the vehicle controllers. The following figures will show how the host code of the GW works and how the DFMEA, FTA, and V&V tools helped develop the control architecture. Figure 13 shows some of the debugging that was done to ensure that the GW worked and some of the message ID management that was done to make sure that the right IDs were being allowed on each appropriate bus.

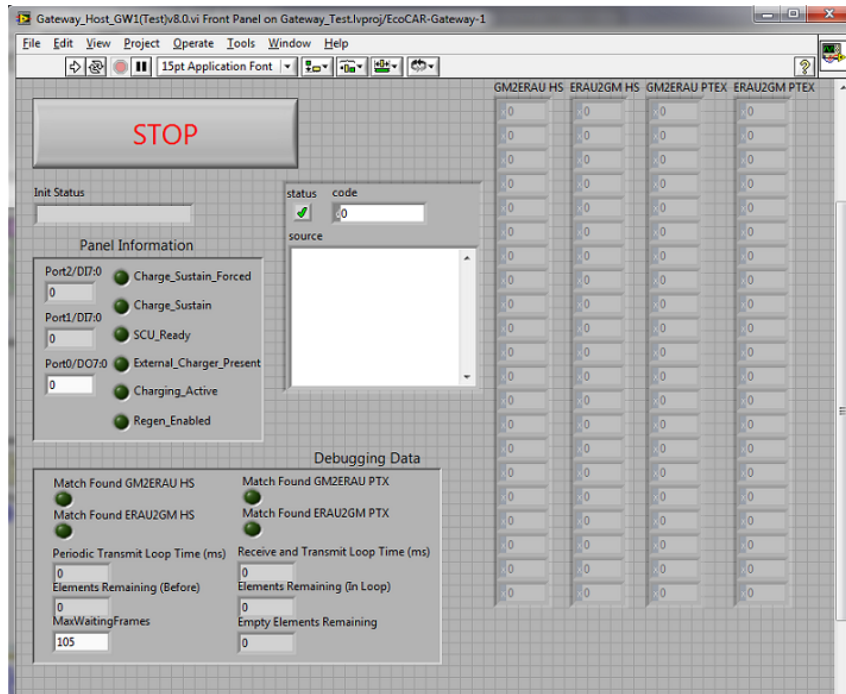


Figure 13: Front Panel of the Gateway Host Code

The debugging window in figure 13 was used to make sure that the GW was allowing messages to pass through and to make sure that the FPGA was operating correctly. This was part of the FTA process when determining communication failures and seeing if the GW was properly mitigating the problem if the failure did occur. The GW was also meant to serve as an information panel to the driver to notify when the vehicle was charging, vehicle is ready, in regenerative braking mode, and when in charge sustain mode. This was originally part of the requirements for the GW but later changed when the IDEA system was developed and will be discussed later.

The FPGA initialization, shown in figure 14, starts running the FPGA code by opening and running the FPGA VI that is targeted. The box located on the upper right hand side of the figure with a picture of glasses and a pencil near the top of the box is the FPGA read / write control function and sends initialized data for the FPGA code [26]. This section is meant to make sure that the FPGA code does not continue on until later

parts of the code allow the FPGA to move onto the next step. This was developed during part of the V&V process when determining proper FPGA communication was established. The control system would often fail due to certain Boolean variables left in the true case and prevent the FPGA code from properly working. After some testing through fault insertion, it was determined this was the best way to prevent this fault from occurring and completely mitigating the problem.

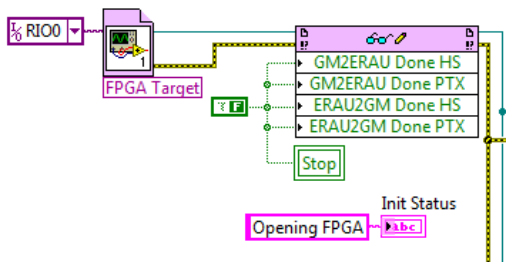


Figure 14: Gateway Host Code FPGA Initialization

The message ID list configuration is the next step of the GW host code shown in figure 15. This part of the code begins by entering a flat sequence structure, which is the grey box that is surrounding the figure. A flat sequence structure is used to ensure that a sub diagram executes before or after another sub diagram [27]. During part of the V&V and FTA testing of the communication to the FPGA, the host code would not run in the order that was necessary. The flat sequence structure was used to force the code to operate in a sequential manor.

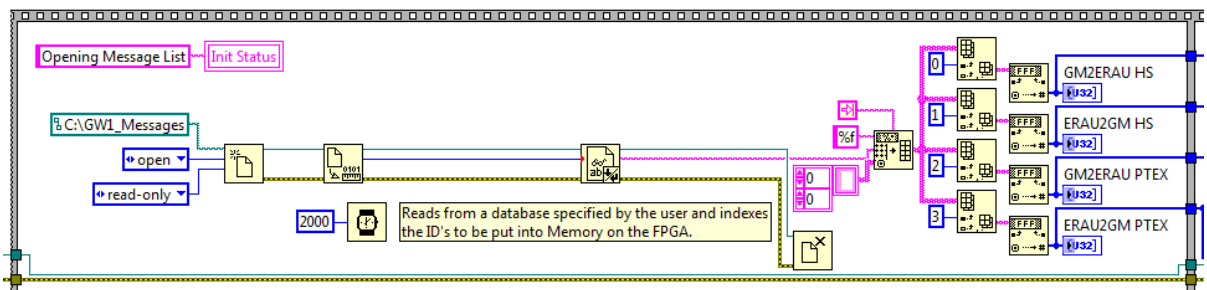


Figure 15: Gateway Host Code Message ID List Configuration

There is a file loaded onto the board's flash memory that can be targeted by the host code, and is targeted as C:\GW1_Messages shown on the left side of the figure. This file contains in a tab-delimited format the message IDs allowed to pass through the GW in either direction for all four busses. This section of code opens the file [28] as a read-only file, reads off the IDs in a string format [29] denoted as the pink lines in figure 15, converts the strings into hexadecimal numbers [30], organizes them into separate arrays [31] denoted by the thick blue lines on the right side of figure 15, and then sends the arrays off into the next stage of the host code. To the knowledge of the team at the time, there was no direct way to read and translate a file. This was the best way to ensure that the sequential order operated correctly and mitigated any communication issues.

The EcoEagles made sure to have LabVIEW treat every message ID that is dealt with on the bus is in a hexadecimal format. This allowed easier recognition of messages relating to documentation given to the team from GM. There was a two second delay integrated into this step using the wait VI [32] to make sure the code had ample time to organize the messages appropriately. Throughout testing a discovery was made that even with the sequential order now applied to the host code, the speed needed to be constrained to ensure that the message list was properly communicated to the FPGA code. The code would often skip over a few messages from the list due to this issue.

The loop shown in figure 16 took the arrays from the previous step in the flat sequence structure and then sent the hexadecimal IDs one by one to the FPGA code using a "for" loop [33]. The benefit of the "for" loop was the ability to send one message ID at a time to the FPGA code instead of one massive array and was developed this way to ensure proper communication requirements. As each one was sent, the host code would

then send a Boolean to the FPGA using the FPGA read / write control function to make sure that the code new it was done with one ID and it was now supposed to move onto the next ID. The “for” loop would only run however many times there were messages for that particular direction. Since there are four busses, there are four “for” loops running and sending arrays to the FPGA. Once the “for” loops were done with the last ID, the code would then move onto the next section of the flat sequence structure. The four “for” loops enabled a more visual way of showing how each separate bus the GW handled. The host code was organized in this way to enable ease of use and understanding to future control students.

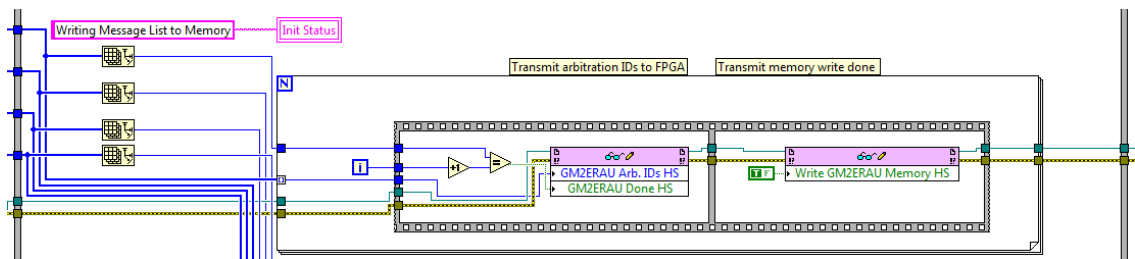


Figure 16: Gateway Host Code Memory Write Loop

This next section is the FPGA check, figure 17. This part of the code runs using a while loop [34] continuously until the FPGA sends the appropriate Boolean. The host code is meant to stay here until the FPGA is done writing all of the IDs to memory and to make sure the code has time to be ready to move onto the next section. A lot of the checking states of the code were implemented to use as debugging tools, as part of FTA, and to make sure that communication was working properly before moving onto the next stage of operation.

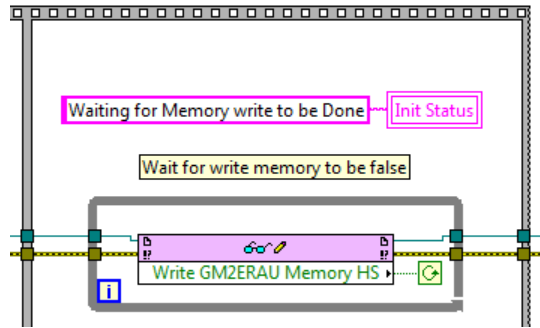


Figure 17: Gateway Host Code FPGA Check

Although the following sections of code were not used for the EcoCar competition, the GW had a portion of code that was able to handle driver display information and some of the messages that the SCU controlled. The figure below, figure 18, shows the initialization stage of the code. This section of the flat sequence structure opened a database file on the board and obtained message information while organizing all of the information into arrays. This was all done using the CAN frame to channel conversion library provided by NI [35].

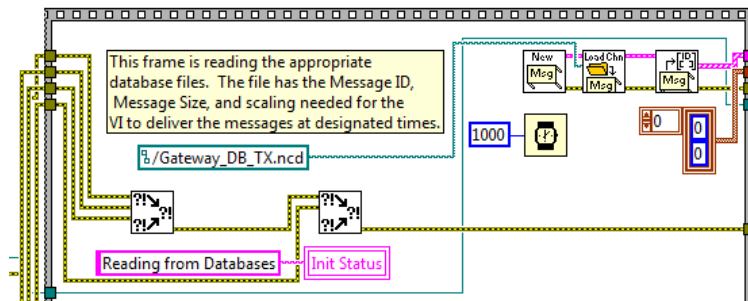


Figure 18: Gateway Host Code Driver Panel Management Initialization

After opening the database and organizing all of the information, the next section of code in the flat sequence structure, shown in figure 19, takes all specific information from the messages using `unbundle by name` [36] and then combines the information and bundles by name [37] into a cluster of information. This can be done for each message

through use of the “for” loop. Once the clusters are created and the array of clusters is organized, the next section of code is utilized.

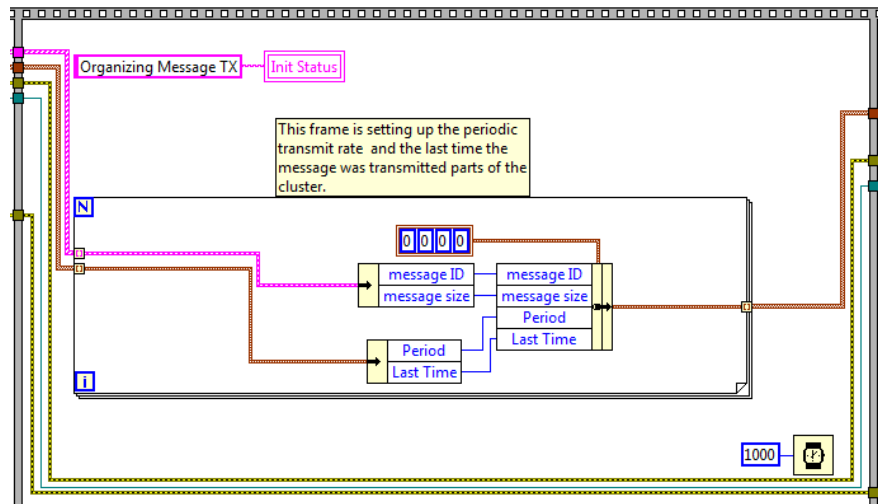


Figure 19: Gateway Hose Code Driver Panel Management Message Bundling

The code enters the next step by assigning a periodic transmit rate to each message, shown in figure 20 on the next page. The GW would be able to handle multiple messages with communication dependability and speed under consideration. It was discovered through FTA and testing that too many messages would potentially slow down the communication rate and lead to lag or potential communication loss. Once all of the messages were set with their respective periodic rates, the code entered a continuous state of running until the stop button was hit or if the board was powered down. This new section also handled the messages that needed to be received or transmitted, shown in figure 21, and also handled the driver panel notification through FPGA port control [38] using the FPGA read / write function.

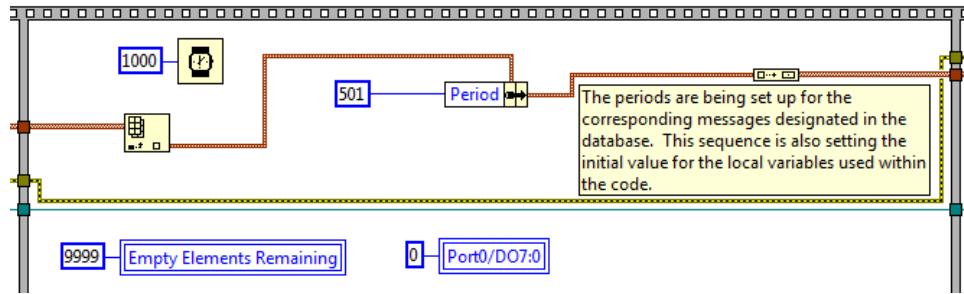


Figure 20: Gateway Host Code Driver Panel Management Periodic Message Handling

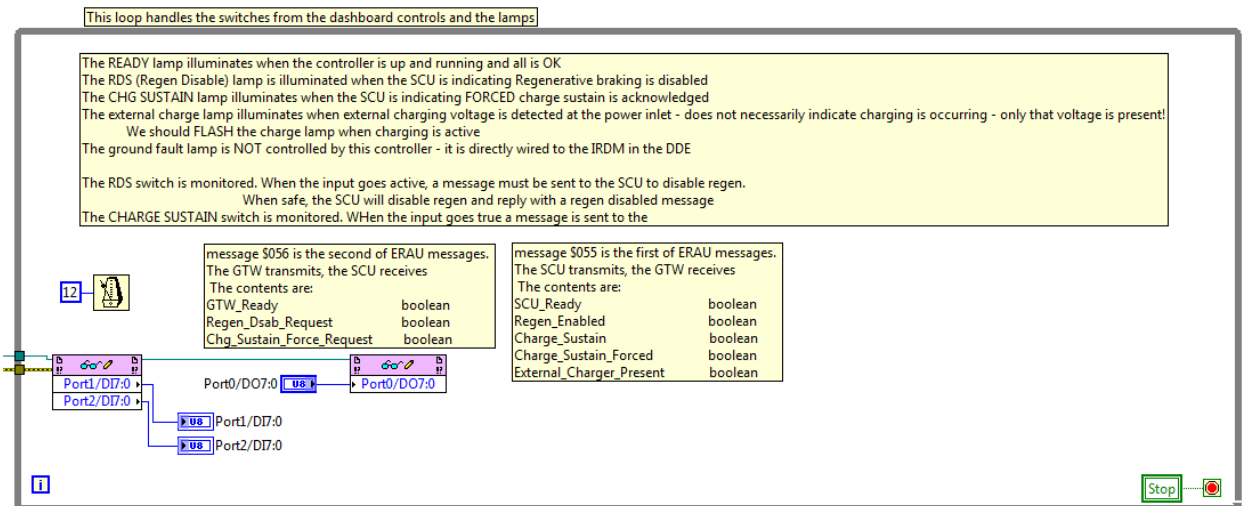


Figure 21: Gateway Host Code Driver Panel Port Control

The port control was never fully developed and tested. This was because of the IDEA control system. The IDEA control system took over the driver notification panel and any message handling that went along with the notifications. Leaving this code in the GW did not slow down the communication but did enable the GW to expand if necessary for vehicle development. Everything else within the GW host code was developed using FTA during the year two competition and again during vehicle development leading to year three competition.

FTA played a major role during the development of the host code of the GW. Initially, the code had a lot of issues with communication between the FPGA and the host

code. The flat sequence structures were discovered to assist in the debugging process. To mitigate the communication errors occurring on the GW, the host code and FPGA were both developed to acknowledge when certain steps were complete. Previously, the code was able run without the acknowledgement and this was causing sections of code to not establish proper communication. The flat sequence structure coupled with while loops solved the issues causing the communications problems. The sequenced acknowledgments, or handshaking, allowed the codes to interact and accomplish the targeted goal without issue. Eventually, the IDs were being set correctly and those messages were transmitting correctly on the respective busses. A majority of the message ID control and communication control is set within the FPGA code on the GW board. The FPGA code embedded is embedded into the GW board and was developed in parallel to the host code.

The Gateway – Field-Programmable Gate Array Code Development

The NI LabVIEW FPGA Module extends LabVIEW graphical development to field-programmable gate arrays (FPGAs) on NI Reconfigurable Input / Output hardware [39]. You can use this custom hardware for unique timing and triggering routines, ultrahigh-speed control, interfacing to digital protocols, digital signal processing (DSP), communications, and many other applications requiring high-speed hardware reliability [39]. When ensuring communication and proper control over all of the subsystems, reliability was crucial, like any other product under development. The EcoEagles developed the FPGA code to manage the board's ports and interfaces using the FPGA I/O node function. The FPGA code was compiled using a compiler integrated into LabVIEW.

What is unique to the FPGA code is its ability to operate within the nanosecond. This is a lot faster than what is necessary, but allows communication to operate smoothly and without much lag or interference. As mentioned, the FPGA interfaces with the hardware side of the board and allows both the FPGA code and host code to control the hardware. The boards have a CAN interface that NI produces that is attachable to the board. The product is the two-port, high speed CAN module for NI compact RI/O, or the NI 9853 [40]. The SCU and GW are equipped with two of the NI 9853s. The FPGA allows the ability to use these and isolate the busses.

The front panel of the FPGA, shown in figure 22, shows some of the Boolean and arrays that were interfaced with by the host code. This panel also shows some of the debugging tools that we linked to the host code to make sure that communication was actually occurring during FTA and V&V development of the GW operation. The “match found” Booleans along with the “total received” indicators were used to check and make sure that communication was working and that the message list was set correctly.

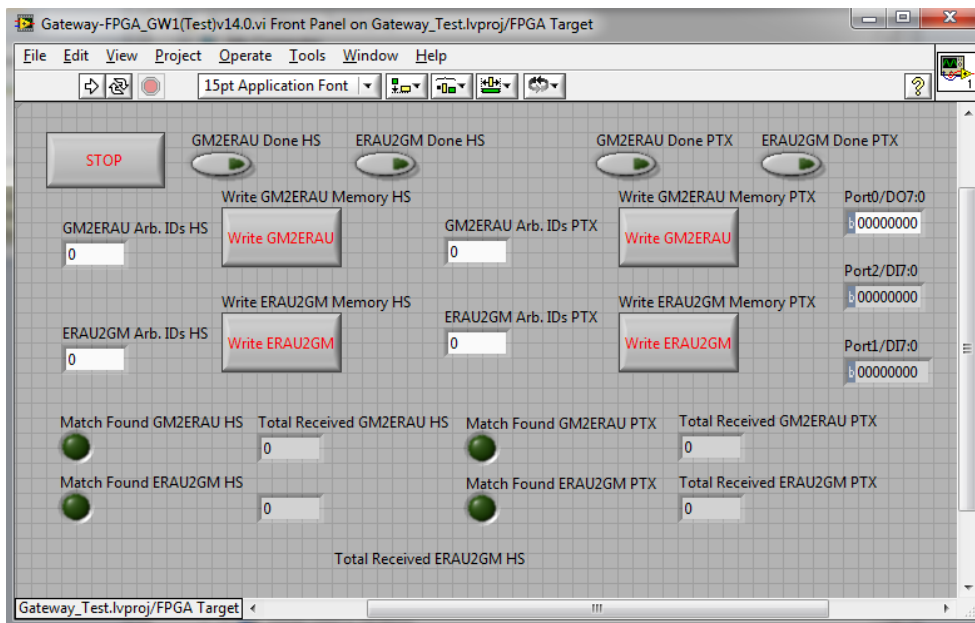


Figure 22: Gateway FPGA Front Panel Interface

The first section of the flat structure sequence in the FPGA code, figure 23, initializes the FPGA. This step stops the CAN modules from communicating using the invoke method function [41], sets all the Boolean variables to false, and enables the digital input / output ports to a certain value using the invoke method function. The FPGA code, along with the host code, was also setup to set the Boolean variables to false to make sure that both codes were properly initialized. The double redundancy was developed to ensure the communication fault would not occur. This section of code is meant to make sure that no CAN modules are still running and to reset all the values prior to going into the next phases of the flat sequence structure. This helps ensure proper communication by making sure all modules are off prior to running. During some of the testing and development, the CAN modules were discovered to still transmit if a failure were to occur and prevent proper reestablishment of control.

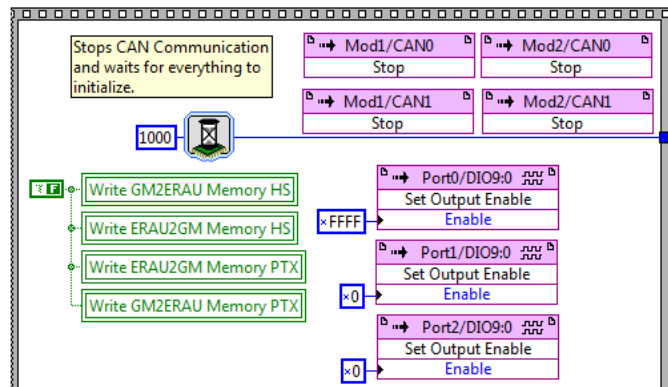


Figure 23: Gateway FPGA CAN, Boolean, and Port Initialization

Before moving on to the next section of code, the VI needs to have on board memory and FIFO allocation. The memory block serves as the list to check and see if the message is allowed to pass through [42]. The FIFO is a method that should be used to transfer data safely from a time-critical VI to a communication VI running at normal priority, which can then be used to transfer the data to the host machine without affecting

the system determinism [43]. In other words, it acts as a buffer between the host code, FPGA code, and CAN modules to make sure that communication is not lagged or being dropped due to one module running faster than the other. These FIFOs were discovered to come in handy in preventing communication failures. The FIFOs and memory blocks within the code are shown in figure 24.

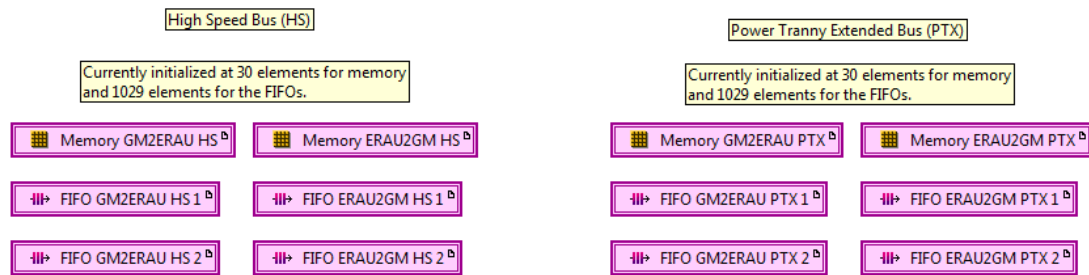


Figure 24: Gateway FPGA VI Memory and First-In / First-Out (FIFO) Configuration

The memory-writing loop shown in figure 25 interacts with the host code of the GW. Once a hexadecimal number is sent to the FPGA code from the host code, the host code would send an acknowledgement Boolean to the FPGA and the code would then write the number to memory and send an acknowledgement in return to tell the host code that the FPGA is ready for the next one. Since the FPGA code operates at a faster pace than the host code, messages would often not be written to memory or the FPGA would think something was wrong and timeout. A lot of FTA testing was done to make sure that the codes interact in this way to confirm proper communication. This repeats itself until it is done with the last message ID and then stops the loop and continues to the next section of the flat sequence structure.

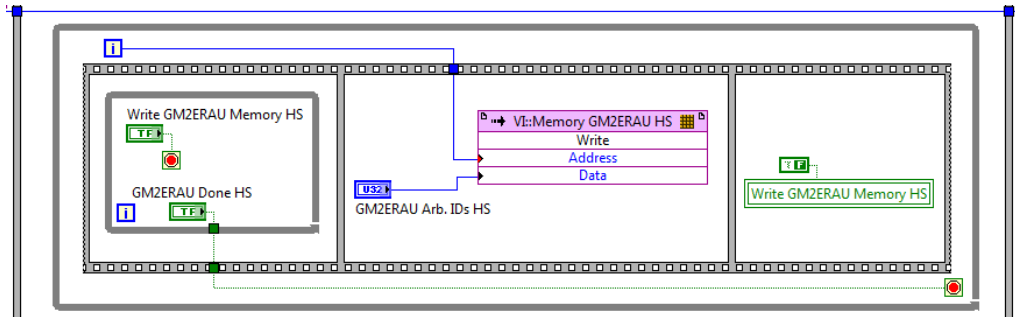


Figure 25: Gateway FPGA Write to Memory Loop

The next section restarts the CAN modules and readies them for communication, shown in figure 26. Once the FPGA has restarted CAN communication, the next section of code takes over and continues to run until the GW is either told to stop or is powered down.

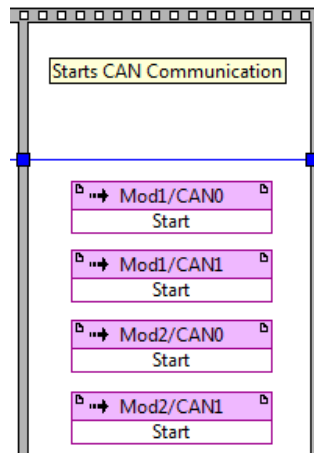


Figure 26: Gateway FPGA CAN Module Communication Restart

The FPGA has twelve loops handling communication and one loop handling the driver notification panel and interface. Four of these loops are CAN read loops, shown in figure 27 on the next page, and it begins by taking the CAN data from the CAN bus and making sure that a message with the ID of x0 is not allowed to flood the bus. This ID in the past has caused the CAN bus to cause loss of communication and lag by taking the entire baud rate. Through some testing and evaluation, instigating the fault into the

system has shown to eliminate the problem. Only message ID x0 was filtered out, any other message ID was allowed into the case structure. A case structure is one or more sub diagrams, or cases, exactly one of which executes when the structure executes [44]. The case structure is what enables the code to act as a filter and only allow the messages we want. The case that does allow the messages to go through has a “for” loop that will run six times and write the data for the message into the FIFO.

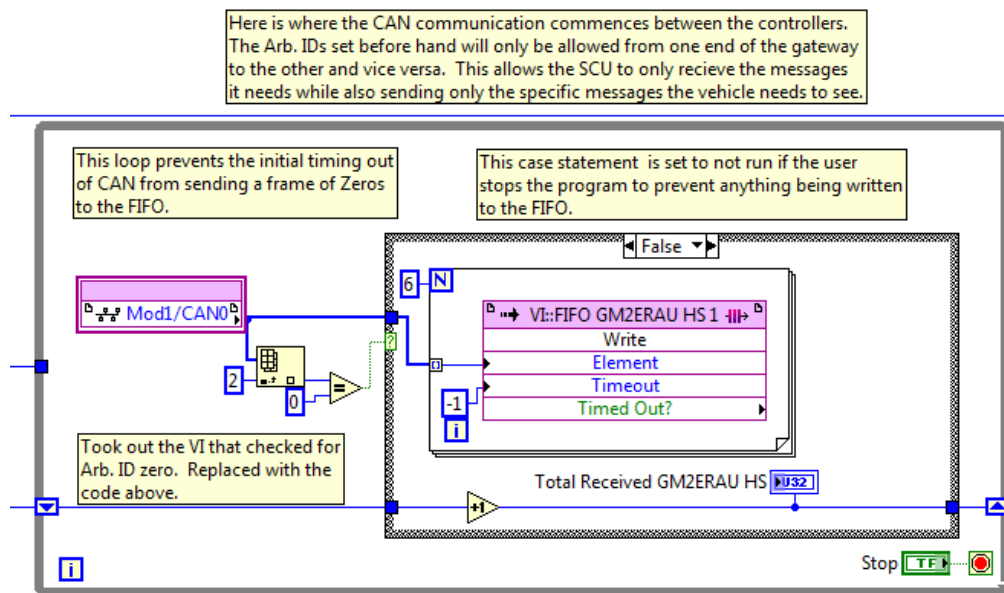


Figure 27: Gateway FPGA CAN Read Loop

Another four loops that are running in parallel to the CAN read loop is the memory checker loop. The memory checker loop blocks any unnecessary messages that are not allowed in the direction the loop was designed for. Shown in figure 28, this loop utilizes the FIFOs and memory blocks internal to the VI. The memory checker loop takes the CAN message data saved in the read loop FIFO and checks for six elements. The elements are checked to make sure that a full message was sent. The six elements include timestamp high, timestamp low, message ID, message size, the first 32-bit data set, and the second 32-bit data set. Once a complete message is received the code will then read

the message from the FIFO and check the message ID with the list that is stored in memory. If memory has that ID stored, then the case structure is set to true and allowed into the next FIFO. The case structure does nothing if the message ID is not stored in memory. When the loop sets the case structure to true, the “match found” Boolean goes true as well and acts as an indicator that communication is working properly for debugging purposes.

There is a second version for this loop and it accommodates for messages that need to cross busses, shown in figure 29. In the second version, there is a case structure that allows only the specific cases to cross over and communicate with the other bus. That case structure is controlled through checking the message IDs coming into the loop and having a specific case for each ID.

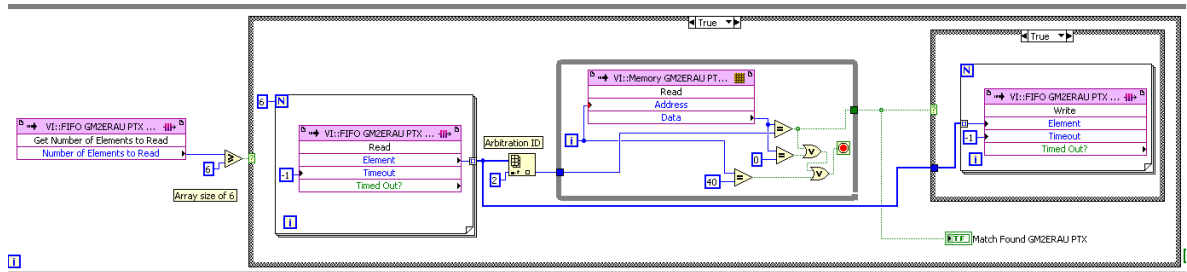


Figure 28: Gateway FPGA Memory Checker Loop (simple)

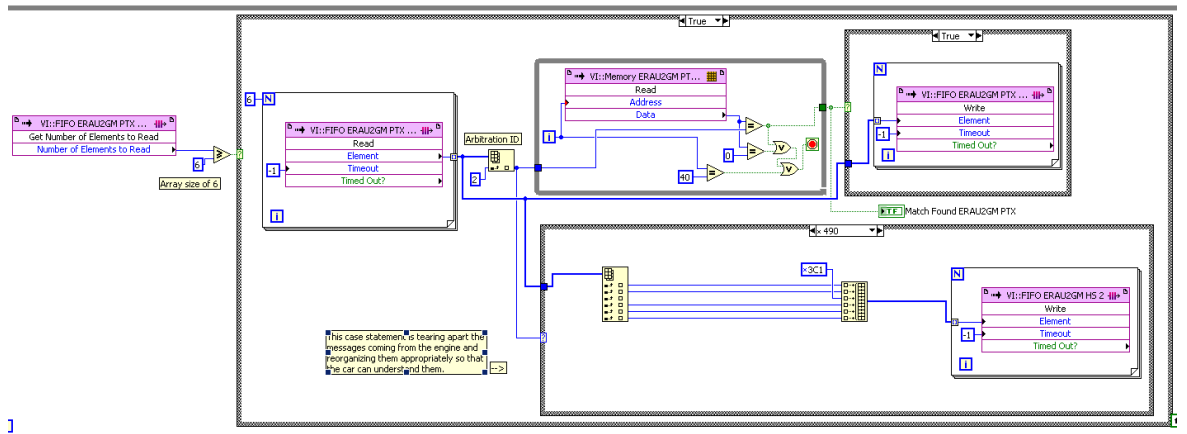


Figure 29: Gateway FPGA Memory Checker Loop (complex)

The next four loops that are running in parallel with the other eight is the CAN write loop. The write loop takes the elements that passed the ID check in the memory checker loop and sends the elements out as an array of six elements over CAN, just as it was received. The CAN write loop also checks for six elements before sending to ensure a complete message. This loop is shown in figure 30.

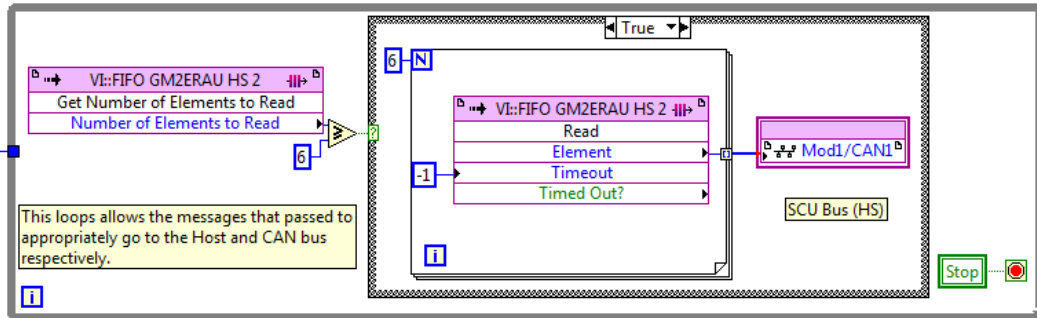


Figure 30: Gateway FPGA CAN Write Loop

The last loop within the FPGA code is the driver panel notification port control loop. This section of the GW control architecture was taken over by the IDEA system. The driver panel notification port control loop, shown in figure 31, was left in the code if the team ever decided to try and utilized the GW for what it was originally designed.

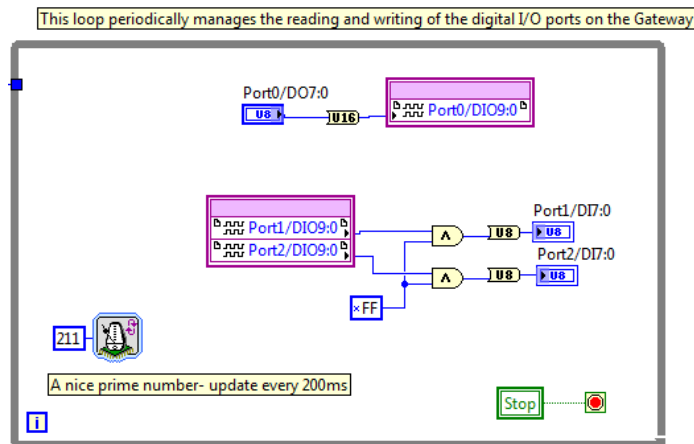


Figure 31: Gateway FPGA Driver Panel Notification Port Control Loop

The FPGA code that was developed has been validated and verified through communication fault mitigation. The FPGA code shown throughout this section is the final result. Communication faults discussed through FTA and DFMEA were used to develop the FPGA code. Systems engineering helped reduce communication lag and finding the issues that caused problems within the GW system. The next thing that needed to be implemented was the communication and databases handled by the SCU.

Vehicle Control System Implementation

A majority of the GW work was done in year two of the competition. The SCU was also being developed but communication with the vehicle was still not working properly. After a thorough amount of validating and verifying the GW was communicating appropriately to the SCU and the vehicle, the SCU needed some refinement. Over the summer, between the end of year two and the beginning of year three of the competition, the SCU FPGA and the databases used for communication were modified to improve vehicle controllability and reliability.

Database and Communication Development

The 2-Mode transmission for full-size, full-utility SUVs integrates two electro-mechanical power-split operating modes with four fixed gear ratios and provides fuel savings from electric assist, regenerative braking and low-speed electric vehicle operation [45]. This transmission is a complex system and steps were taken to properly develop a control strategy. The first steps that were taken were to ensure proper communication before any database editing. The SCU was not properly communicating with the 2-Mode

transmission and was causing improper and sporadic vehicle behavior. This required editing the FPGA code for the SCU. To fully understand the problems encountered with the 2-Mode transmission, GM was gracious and allowed the teams to use their hybrid garage in Milford, MI. They helped each team by donating their time and engineering expertise to solve every problem or question. The engineers at GM helped the EcoEagles by showing the team how to handle protection values and rolling counts that would often be part of important messages being sent over the CAN busses. The work done that alleviated the problem is shown in figure 32 on the next page.

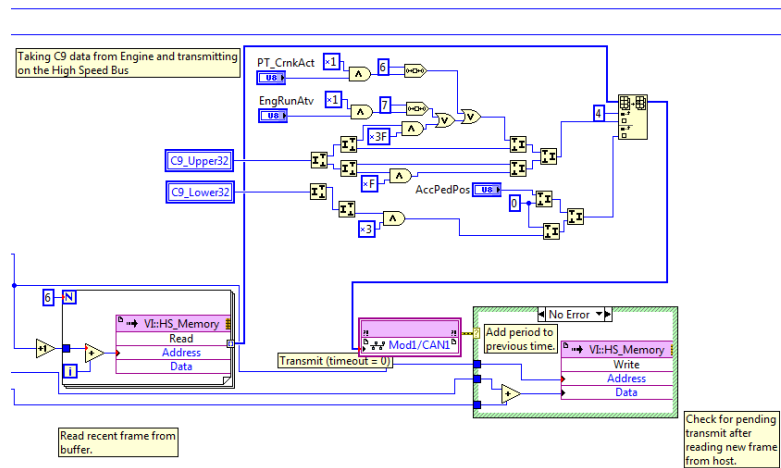


Figure 32: Supervisory Control Unit FPGA Communication Development

This work was a majority of the updating that was needed for the DFMEA documentation, FTA testing, and V&V testing that was currently being done for the vehicle development. Now that the team knew about this problem, the rest of the messages that required these edits were fixed and the communication problems no longer occurred based on this possible failure point. This particular fault was keeping the team from progressing in vehicle development and use of the DFMEA documentation, for it was an unforeseen problem with no real solution at the time.

After the FPGA was capable of handling the messages properly and the controllers on the vehicle responded appropriately, the next step was to establish control of all remaining sub systems. The rest of the subsystems were controlled by providing power through relays or analog voltages. Utilizing the SCU boards capabilities with analog inputs and outputs as well as the digital inputs and outputs did this. To control these subsystems, the SCU FPGA needed to be programmed to use specific ports so the SCU host code could use the hardware. A separate loop was created to run in parallel with the rest of the SCU FPGA code. Keeping this section of code in a separate loop helped organize the code and allowed future students to know which loops were required for analysis if a fault did occur. Part of the code used to do digital and analog control is shown in figure 33.

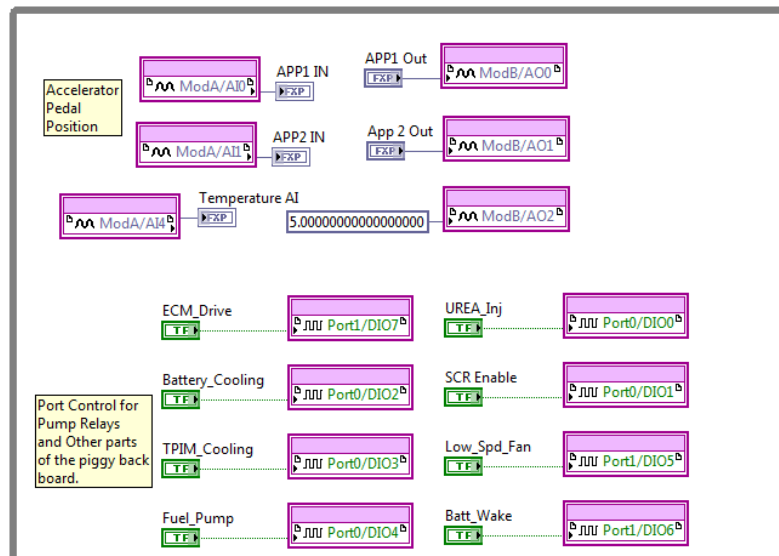


Figure 33: Supervisory Control Unit FPGA Subsystems Control Loop

The next development phase was proper database management. The vehicle was finally able to be communicated with correctly after a lot of updating of the DFMEA documentation and control architecture, so the team started working on making sure the

proper databases were created. Each database was developed with the future in mind. The reasoning was because of the amount of development time required for each database change. When a database was changed, a lot of the variables within the host code had to be reorganized to accommodate the new messages, or lack of messages. There was no better way to make the process more efficient that was known at the time. So to create these databases, the program used was the measurement and automation explorer (MAX) [46], shown in figure 34 on the next page. This program is part of LabVIEW and the NI CAN drivers had to be downloaded and installed in order to allow MAX to create messages that followed along with the CAN communication protocol [47] so the team could develop the databases.

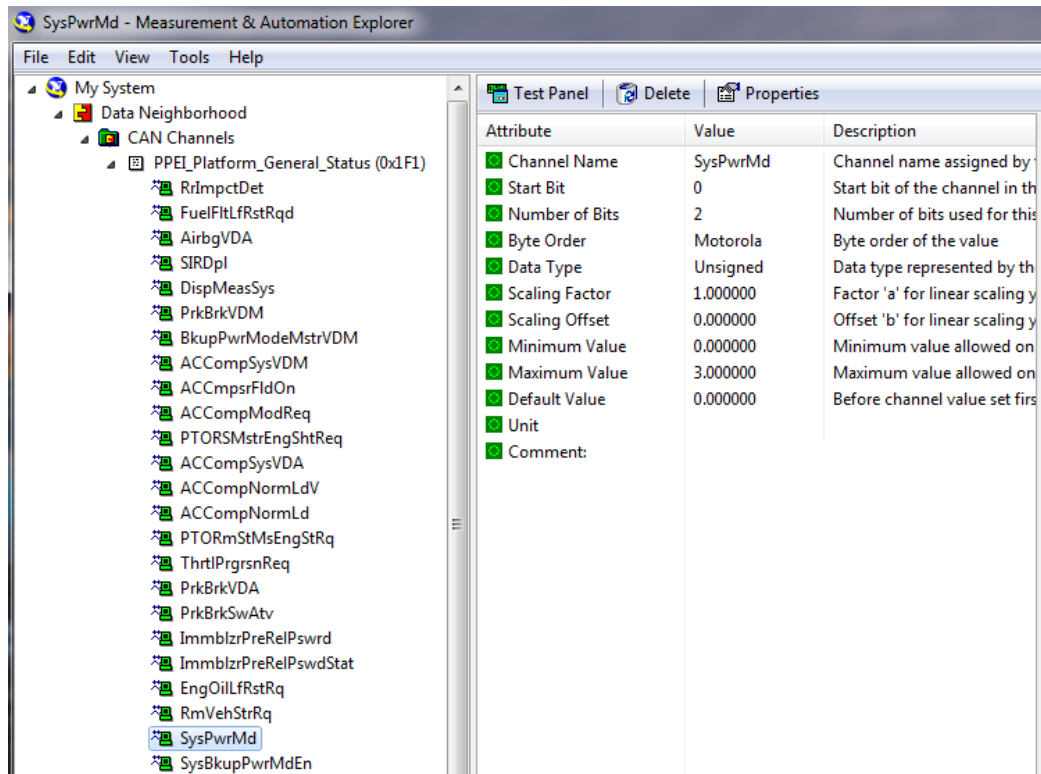


Figure 34: National Instruments Measurement and Automation Explorer

All of the communication and database development, along with the FPGA development of both the GW and SCU had to be validated and verified to work properly

and in accordance with the requirements. This was a continuous process throughout the EcoCar competition. The validation and verification processes played a key role once communication was established and the databases created. Troubleshooting through FTA and checking the DFMEA documentation would occur during vehicle development to verify that problems could not occur, or when problems did arise they were handled quickly and safely through tolerance testing. One such example would be when a relay signal would be intermittent. The team first looked at all of the electrical connections to the relay. The investigations eventually led to the signal wire coming from the board. The discovery was that the voltage would predictably drop every time the wire was moved. The wire was replaced, and the relay was working properly once again.

Another problem the EcoEagles faced during the beginning of year three was the 1.3L turbo diesel engine. The team originally drove the vehicle by faking the engine data to the vehicle. This temporarily allowed the vehicle to operate in mode one during the integration process prior to real engine testing, which was an all-electric driving mode up to speeds of 25 miles per hour. Unfortunately this meant the vehicle could not shift into mode two and reach higher speeds. Without the engine, the transmission could not accommodate the higher speeds due to the main oil pump requiring the engine to operate. The engine controller never communicated over the CAN bus prior to mode one capabilities.

After reading documentation online from Penn. State, the engine controller was configured over CAN and verified sending data on the CAN bus to the SCU. The next step was to take the data the engine was sending and let the vehicle see the specific data needed. This was accomplished through verifying communication of engine CAN

messages a few at a time on the bus required and validating the requirements for each message needed for proper vehicle control. Once that was accomplished, the vehicle was operational now with the engine controller taking over compared to the SCU faking the engine data. The process took two months of testing, but the vehicle was finally operating with the engine and capable to reach highway speeds after fault mitigation testing for proper communication and engine control.

Another important part of the car that needed communication development was the charger and battery pack. A123 Systems designed the battery pack control module to be able to communicate with the BRUSA charger. This was never tested prior to year two in vehicle development. After looking at the A123 and BRUSA charger documentation, all of the wiring required was connected and the control system was ready to be tested. When the charger was plugged into the wall, the CAN line was observed to see if communication was established between the charger and the battery pack. The team discovered that the charger and battery pack work together and the charger could safely manage the battery control system automatically. All of this was done in accordance with DFMEA documentation and the requirements given by A123 Systems.

Chapter IV

Results

EcoCar Control System Performance

Year three of the competition was a year for refinement. The vehicle was in a partially operational state at the beginning of year three and a lot of work was needed, especially in accordance to the requirements that were set in year one. The EcoEagles needed to get the engine controlled, have the IDEA system running, be able to achieve highway speeds, apply aerodynamic modifications, and gain full control of all of the subsystems. Over the course of the year, the team managed to accomplish this and be ready for the year three competition.

Engine control was a vital step in vehicle development. This enabled the team to begin shift strategy development along with power management of the charging capability of the engine. Once the communication and control was validated, work began on the shift strategy. The shift strategy was created using software-in-the-loop (SIL) system and tested using the vehicle as a hardware-in-the-loop (HIL) system. Development quickly progressed and led to the EcoEagles testing the control system on campus and was met with success. The vehicle control architecture was able to start the engine, shift into neutral and mode two without any issues or problems. That day the vehicle was able to achieve speeds of 30 miles per hour and higher. Figure 35 shows the engine bay with the engine on the left and the tractive power inverter module (TPIM) on the right hand side.



Figure 35: Vehicle Engine Bay

There were two issues that arose during vehicle development involving the engine. The transmission input shaft sheared apart and the team had to discover the issue that caused this problem. Checking the DFMEA documentation led to improper communication, incorrect torque request, or incorrect rotations per minute (RPM) setting. After a long investigation, it was finally determined that engine shutdowns had to be smoother to make sure that the input shaft was not fighting the engine during this phase. The engine was taking control of the RPM of the input shaft because during the engine shutdown procedure the controller would think the engine is about to stall and try to inject more fuel into the system to compensate. To solve that problem, the SCU was programmed to fake the engine messages temporarily while the engine control module (ECM) was shut off for a brief second to prevent the ECM from thinking the engine was stalling during the shutdown. Shutdowns resulted to be a lot smoother, but the input shaft was a major concern and the controls team began testing within SIL systems to find ways to prevent an input shaft failure from occurring again.

Eventually there was the second issue with the engine at the year three competition. The turbo on the turbo diesel engine failed due to backpressure on the exhaust system. The diesel particulate filter (DPF) failed to be burned off due to the rotations per minute (RPM) control from the EcoEagles control system, shown on the left side of the picture of the exhaust tubing in figure 36. This potentially led to another failed input shaft along with the broken turbo. To fix this problem the team would have to reprogram the SCU to detect when the engine needs to burn off the DPF and allow the RPM control to set the engine at a higher RPM. The team plans on replacing the turbo for the diesel engine and incorporating the new DPF detection into the code.

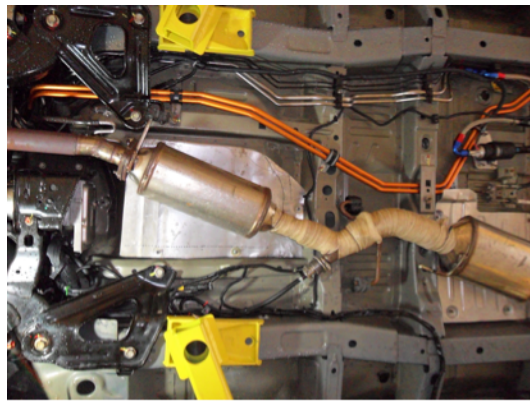


Figure 36: Vehicle Exhaust System (DPF on left)

The IDEA system developed quickly over year three of the competition, which is shown in figure 37. The VCS developed with consumer acceptability in mind. The EcoEagles IDEA team lead worked on developing a panel that look appealing to the eyes, while enabling the team to be able to monitor vehicle status in accordance with competition requirements. The team also decided to integrate the required driver display into the IDEA system. LabVIEW was installed on the IDEA computer so that the device could interface with the SCU and be able to control certain subsystems. Eventually the IDEA system was able to control the driver panel notifications, which included

regenerative braking, charge sustaining, external charge detection, control system readiness, and ground fault detection.



Figure 37: IDEA Computer with Student Designed Bezel

Testing for highway speeds and aerodynamic modifications occurred at the same time. The Daytona International Speedway (DIS) offered to allow us to use the backstretch of the racetrack after some of the team inquired. The team was able to take the vehicle over and commence with basic testing of acceleration, braking, top speed, and some of the aerodynamic modifications. Although acceleration, braking, and the aerodynamic modifications were not fully tested the higher speeds of the vehicle were tested. The EcoEagles managed to acquire a new high speed of 65+ miles per hour. This speed was the highest speed the team has ever achieved from the vehicle.

The subsystems were also a hassle during development of the vehicle during year three. A lot of the time the subsystems would work, and then sporadically they would not. However, charging the 12V battery typically solved this problem. It has not been confirmed yet, but the SCU and GW might need a steadier state of 12V on the voltage bus in comparison to one that fluctuates on a vehicle. The problems that would occur would sometimes be loss of communication, and other times would be proper voltage control over some of the relays but not all. If a 12V charger were on the vehicle and the

team decided to deploy the SCU and GW controller codes, the vehicle would work. This is a current problem that has not quite been fixed but there is an idea on how to fix the issue. The idea is to put a capacitor in parallel with the 12V battery to help the SCU and GW maintain a firmer 12V signal on the low voltage bus. This would help accommodate any large current draw that the boards may need on booting. Currently, the boards work well once they have booted with proper power and without any problems. This solution may help the vehicle run more efficiently and without issue with the subsystems.

Systems Engineering Results

The integration of Systems Engineering into the EcoCar project has been rather difficult, but has helped the team greatly. Through DFMEA, the team was able to find possible solutions or even create new ones based on the experience from the issue. DFMEA played a big role in the trouble shooting process whenever a problem would arise. The EcoEagles would check the documentation to get a good idea of what may cause the problem, and then the team would start by putting that fault into the SIL systems. The team was also able to properly identify how critical a lot of the problems were in relation to the control system and the importance to the competition requirements. With this further understanding, the team was able to lower the RPN on a few items through the use of testing and evaluating the control architecture. Some of the RPNs lowered were for communication issues, wiring dependability with the piggyback board, and the accelerator pedal position signal and sensor for the engine controller.

The fault tree analysis helped the control architecture greatly. By instigating faults into a SIL, and later the vehicle as a HIL, a majority of the problematic faults were

mitigated and tested to ensure safety of the driver. The team made sure that if any critical problem did occur on the vehicle that proper control settings were tolerant and in place to allow any driver to have control of the vehicle to get to safety. Accelerator pedal failures, CAN communication loss, input shaft failure, and loss of relay control are some of the problems that were tested, and actually happened during vehicle development that the team strived to fix and make sure to mitigate or tolerate properly.

The validation and verification process is what ultimately ties everything together. V&V does not exist without proper FTA and DFMEA. Throughout the competition, the team would often look back at the requirements to make sure the project was on task and on time. The team was also making a graph to represent the overall production readiness of the EcoEagles vehicle to keep track of progress, shown in figure 38.

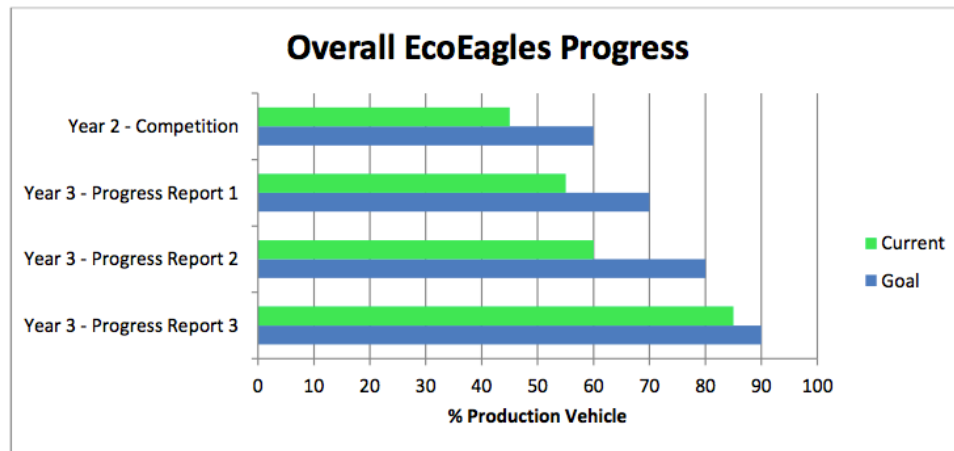


Figure 38: EcoEagles Production Vehicle Readiness

Figure 38 expresses how the EcoEagles progressed over the summer between year two and year three and throughout year three of the competition. Validation of the vehicle control system and mechanical operation were satisfied for the requirements by the end of year three competition.

The results shown in figure 38 are based off vehicle requirements and how the team feels the vehicle compares to the production standards. At the end of year two, shown in the figure, the vehicle was supposed to be at a 60% readiness in accordance with vehicle production standards of the automotive development process. There is no true definition of “60 % readiness” other than what is required of the vehicle for the competition. The teams base the 60% readiness on how well they feel the vehicle is performing and the current stage of development. This also counts toward the other sections of the figure. The part of the graph that best expresses the most improvement of the vehicle development for the EcoEagles is during the time between progress report two and progress report three. The result of implementing better systems engineering practices allowed the team to facilitate faster development through less risk. This gave the team a 25% overall increase of what was felt as the production readiness of the vehicle increasing from 60% to 85%.

The vehicle technical specifications were the requirements that needed to be based on the performance of the vehicle. As the competition progresses in the various stages, each university needs to predict the performance of the vehicle being designed. Systems engineering integration into the project helped keep the development on track and keep the predictions relatively close to the actual performance of the vehicle. Testing, evaluation, validation and verification through FTA and DFMEA helped with keeping the VTS up to date. The VTS can no longer be updated during year three of the competition. This forces the teams to ensure performance measures are met and that vehicle development progresses as set by the individual teams. These same practices are done in

the automotive industry and prepare students for the workforce. The EcoEagles VTS that progressed over the three years of the competition is shown in figure 39.

Specification	Competition Requirements		EcoEagles	EcoEagles Competition
	Production Vue	Competition Requirement or Target		
EcoCar			Projected	Measured
Accel 0-60	10.6s	≤ 14 s	11.8 s	≥20*
Accel 50-70	5.7	≤ 10 s	8.6 s	N/A
Towing Capacity	680 kg (1500 lbs)	≥ 680 kg @ 3.5% grade, 20 min at 72 kph (45 mph)	680 kg	N/A
Cargo Capacity	0.83 m ³	Height: 457 mm (18 in) Depth 686 mm (27 in) Width 762 mm (30 in)	Height: 457 mm (18 in) Depth 686 mm (27 in) Width 762 mm (30 in)	Height: 457 mm (18 in) Depth 686 mm (27 in) Width 762 mm (30 in)
Passenger Capacity	5	≥ 4	5	5
Braking 60-0	38 - 43 m (123 - 140 ft)	< 51.8 m (170 ft)	46 m (151 ft)	38.4 m (126 ft) *
Mass	1758 kg (3875 lb)	2268 kg (5000 lb)	1974 kg (4352 lb)	2043 kg (4504 lb)
Starting Time	≤ 2 s	≤ 15 s	10 s	1 s
Fuel Consumption CAFÉ unadjusted, Combined, Team: UF Weighted	8.3 L / 100 km (28.3 mpgge)	7.4 L / 100 km (32 mpgge)	6.25 L / 100 km (37.7 mpgge) (0.43 UF)	N/A
Charge Depleting Fuel Consumption	N/A	N/A	5.23 L / 100 km (45.26 mpgge)	N/A
Charge Sustaining Fuel Consumption	N/A	N/A	7.4 L / 100 km (32 mpgge)	N/A
Charge Depleting Range	N/A	N/A	38.54 km (23.95 mi)	N/A
Petroleum Use	0.85 kWhr / km	0.77 kWhr / km	0.40 kWhr / km	N/A
Emissions	Tier II Bin 5	Tier II Bin 5	Tier II Bin 4	N/A
WTW GHG Emissions	250 g / km	224 g / km	158 g / km	N/A

* Indicates calculations from CAN messaging data

Figure 39: Vehicle Technical Specifications (VTS)

Chapter V

Discussion, Conclusions, Recommendations

Discussion

The EcoEagles had a few reoccurring problems throughout vehicle development. The utilization of LabVIEW became a few problems when developing the control system. Sometimes the control system would not work properly if certain aspects of the graphical user interface were moved. Another issue that arose dealt with deployment of the code. The version of LabVIEW that the EcoEagles used required some finesse when applying state-charts, which are similar to state-flow in MatLab, to the control architecture. The boards required an older style of formatting to properly store the state-charts in memory and properly deploy. When designing a fault mitigating system, these are just a few variables that you do not expect when determining possible fault causes of a failed control system.

No matter how well planned out a system may be, unexpected occurrences will always arise, but properly tolerated if the system is designed correctly. One way to eliminate these possible faults from occurring during the control system development would be to keep secure version control over any code being created. Another possible solution could have been to update the program to the latest version, since the latest version may have gotten rid of these issues. Updating to the latest version may cause different issues, and the risks would then have to be weighed to discover the best consideration.

Another issue that arose during vehicle development was the 2-Mode Transmission communication with the SCU. In an unfortunate circumstance, the 2-Mode

transmission was no longer being supported for the 2009 Saturn Vue due to vehicle production being canceled. This caused GM to not have the capability to provide the amount of support needed for the teams using this transmission. However, GM was able to assign two engineers to help the teams discover the issues that were occurring. These two engineers were able to give the teams a better understanding of how the transmission operated and give them more confidence on a proper control strategy. The EcoEagles were able to incorporate a diesel engine with the transmission where it was thought not possible.

The EcoCar competition required a lot of planning to properly integrate everything into the vehicle safely and efficiently. The control system took a majority of the time due to the complexity of the 2-Mode transmission. Due to the complexity, the control system held the team back for almost a year. The vehicle was supposed to be in an operational status of 60% production readiness by the end of the year two competition, but the vehicle was unfortunately closer to 45%. Thanks to GM and ANL, the coordinated efforts enabled the 2-Mode teams to fix all of the issues at hand and get the vehicles operational. The EcoEagles had to pick up where the year two competition left off and fully develop and refine all systems on the vehicle by the year three competition deadline. This task was hard and tedious, but the team managed to pull through and get the vehicle to a state of operation of 90% before the final competition.

Conclusions

Systems engineering played a big part by the team utilizing the DFMEA documentation along with the test procedures created to help assist the team during development. Testing procedures help validate and verify system operation along with helping keep the students who worked on the vehicle safe. The FTA that was created helped the team discuss any possible failures that could occur on the vehicle and how to prevent or detect these issues and properly mitigate the problem. These discussions led back to the DFMEA documentation and assisted in keeping it up to date. Through the guidance of the team, the systems engineering class was able to create documentation that helped lead the team to work efficiently and more importantly safely. Systems engineering was influential throughout this project and trained the students to discuss, think, and more importantly cooperate and come together and develop a vehicle. Systems engineering was important for this project, and it trained all of the students to better understand the process and ultimately give them the experience they need to work in industry. Because of this, the control system was successfully implemented and operated safely for the GM drivers that tested the EcoEagles vehicle on the Milford Proving Grounds in Michigan.

Recommendations

A few things are considered for this project to be a complete success. One thing that needs to be completed is the DPF section of the code to properly control engine so as to not break another turbo. Another topic that was not fully developed on the vehicle is the power management of the powertrain systems. Research was done on how to best use

the engine with the transmission, but was not fully implemented into the vehicle due to safety and time constraints. This would increase the vehicle overall efficiency and performance and provide a better drive quality when in operation. Another system that was not finished was the fuel gauge for the diesel tank. The wires for the tank exist but not enough time was available to properly integrate that into the piggyback board and SCU control system. The air conditioning (AC) is another device that has not been tested and implemented into the control architecture. The changes that would need to be made are a database change, electrical wiring, and conversion of the data. The proper message needs to be converted and sent out to properly control the AC and the electrical wires are to receive the AC high pressure reading for that specific message.

The most important recommendation is keeping with the systems engineering principles. One thing that was noticed was the team's development and progression of the EcoCar project. Systems engineering practices were not being used in certain areas of the project and that hindered the team. This was not realized until the end of competition and towards the beginning of the third year, but it is important to note. The reincorporation of systems engineering after most of the information was given to the team allowed the vehicle development to increase to a point where the vehicle went from a 45% state of readiness to 85% in under a year. Making sure that a project keeps systems engineering practices and principles to mind will ultimately save time and money in the long run.

References

- [1] Blanchard, B. S. and W. J. Fabrycky. *Systems Engineering and Analysis*. 3rd Edition. Upper Saddle River: Prentice Hall, 1998.
- [2] EcoCar. "2010 EcoCar Competition Finals Year 2 - Integration and Outreach." US DOE and GM, 2010.
- [3] Buede, Dennis M. *The Engineering Design of Systems*. New Jersey: John Wiley & Sons, Inc., 2009. Print
- [4] Farrell, Curt; Sturges, Jim. "Systems Engineering." *Aerospace America* 15 Dec. 2005. Pg. 35. Print.
- [5] Software & Systems Engineering Standards Committee. "Systems and software engineering – System life cycle processes." ISO/IEC 15288:2008(E). IEEE Std 15288™ – 2008.
- [6] Eiler, John R. "Systems Engineering." *Aerospace America* 15 Dec. 2008. Pg. 41. Print.
- [7] Keating, Charles; Rogers, Ralph; Unal, Resit; Dryer, David. "System of Systems Engineering." *Engineering Management Journal* 15.3 (2003): 36-45. Print.
- [8] Quality One International. Quality-One. 2011. 27 August 2011 <http://www.quality-one.com/services/fmea.php>
- [9] M. Boldrin; A. De Lorenzi; A. Fiorentin; L. Grando; D. Marcuzzi; S. Peruzzo; N. Pomaro; W. Rigato; G. Serianni. "Potential failure mode and effects analysis for the ITER NB injector" *Fusion Engineering and Design* 84.2-6 (2009): 466-469. Print.
- [10] Wikipedia contributors. "Fault tree analysis." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 22 Jul. 2011. Web. 20 Sep. 2011. http://en.wikipedia.org/wiki/Fault_tree_analysis#cite_ref-0
- [11] Dr. William Vesely, SAIC. *Fault Tree Handbook with Aerospace Applications*. Handbook. NASA. Washington: NASA, 2002.
- [12] R. Ferdous; F.I. Khan; B. Veitch; P.R. Amyotte. "Methodology for Computer-Aided Fault Tree Analysis." *Process Safety and Environmental Protection* 85.1 (2007): 70-80. Print.
- [13] Wikipedia contributors. "Verification and validation." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 19 Sep. 2011. Web. 20 Sep. 2011. http://en.wikipedia.org/wiki/Validation_and_verification
- [14] Duren, Riley M. "Validation and Verification of Deep-Space Missions." *Journal of Spacecraft and Rockets* 41.4 (2004): All. Print.
- [15] *Clarus Concept of Operations*. Publication No. FHWA-JPO-05-072, Federal Highway Administration (FHWA), 2005.
- [16] Karner, Donald, et al. "Plug-in Hybrid Electric Vehicle (PHEV) Integrated Test Plan and Evaluation Program." Draft. U.S. Department of Energy FreedomCAR & Vehicle Technologies Program Advanced Vehicle Testing Activity, 2007.

- [17] Rosa, Kristen de la. "Advanced Vehicle Technology Competitions for Students." *Argonne National Laboratory* June 2011. Web. 27 Aug. 2011. <http://www.transportation.anl.gov/competitions/index.html>
- [18] Jehlik, Forrest. "Advanced Vehicle Technology Competition: Challenge-X 2008 DOE Merit Review." 2008. 27 Aug. 2011. http://www1.eere.energy.gov/vehiclesandfuels/pdfs/merit_review_2008/gate/merit08_jehlik.pdf
- [19] Rosa, Kristen de la. "Competition Innovations." *Argonne National Laboratory* Aug. 2009. Web. 27 Aug. 2011. <http://www.transportation.anl.gov/competitions/innovations.html>
- [20] Rosa, Kristen de la. "Virginia Tech Wins EcoCar Competition With an Extended-Range Electric Vehicle, The Ohio State University and the University of Waterloo Finish Second and Third." *Argonne National Laboratory* June 2011. Web. 27 Aug. 2011. http://www.transportation.anl.gov/competitions/EcoCar_2011.html
- [21] Department of Energy. "Department of Energy Announces Advanced Vehicle Technology Competition, EcoCar 2: Plugging into the Future." *Department of Energy EERE News* 13 April 2011. Web. 27 Aug. 2011. http://apps1.eere.energy.gov/news/progress_alerts.cfm/pa_id=514
- [22] Harries, Brian. "Control System Design and Optimization Using LabVIEW for a Plug-In Hybrid Electric Vehicle as part of EcoCar: The NeXt Competition." *ASME 2011 International Mechanical Engineering Congress & Exposition*. Hyatt Regency Denver & Colorado Convention Center. Denver, Colorado. Nov. 2011. Paper.
- [23] sbRIO-9642/9642XT. National Instruments. 27 Aug. 2011. <http://sine.ni.com/nips/cds/view/p/lang/en/nid/205900>
- [24] sbRIO-9602/9602XT. National Instruments. 27 Aug. 2011. <http://sine.ni.com/nips/cds/view/p/lang/en/nid/205892>
- [25] LabVIEW. National Instruments. 27 Aug. 2011. <http://www.ni.com/labview/>
- [26] FPGA Read / Write Control Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371599F-01/lvfpghost/readwrite_control/
- [27] Flat Sequence Structure. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/flat_sequence/
- [28] Open/Create/Replace File. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361D-01/glang/open_create_replace_file/
- [29] Read from Text File Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/read_characters_from_file/
- [30] Hexadecimal String to Number Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/hex_string_to_number/
- [31] Index Array Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/index_array/
- [32] Wait Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/wait_ms/

- [33] For Loop. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/for_loop/
- [34] While Loop. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/while_loop/
- [35] CAN Frame to Channel Conversion Library User Guide. National Instruments. 27 Aug. 2011. <http://digital.ni.com/manuals.nsf/websearch/247EBF8EF4661015862576D600526455>
- [36] Unbundle by Name Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361F-01/glang/unbundle_by_name/
- [37] Bundle by Name Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361F-01/glang/bundle_by_name/
- [38] FPGA I/O Node (FPGA Module). National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371599C-01/lvfpga/fpga_io_node/
- [39] LabVIEW FPGA. 27 Aug. 2011. <http://www.ni.com/fpga/>
- [40] 2-Port, High-Speed CAN Module for NI CompactRIO. 27 Aug. 2011. <http://sine.ni.com/nips/cds/view/p/lang/en/nid/201972>
- [41] Invoke Method Function. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371599E-01/lvfpga/hosts/invoke_method/
- [42] VI-Defined Memory Configuration Node. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371599E-01/lvfpga/fpga_vi_scoped_memory/
- [43] VI-Defined FIFO Configuration Node. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371599E-01/lvfpga/fpga_vi_scoped_fifo/
- [44] Case Structure. National Instruments. 27 Aug. 2011. http://zone.ni.com/reference/en-XX/help/371361G-01/glang/case_structure/
- [45] Grewe, Tim M.; Conlon, Brendan M.; Holmes, Alan G. "Defining the General Motors 2-Mode Hybrid Transmission." *SAE World Congress & Exhibition*. Detroit, MI. April 2007. Paper.
- [46] What is Measurement & Automation Explorer? National Instruments. 27 Aug. 2011. <http://digital.ni.com/public.nsf/allkb/71544521BDE34FFB86256FCF005F4FB6>
- [47] NI-CAN 2.7.3 – Windows 7/XP x86/Vista, Pharlap. National Instruments. 27 Aug. 2011. <http://joule.ni.com/nidu/cds/view/p/id/2646/lang/en>

Appendices

Appendix A

ASME 2011 5th International Conference on Energy Sustainability Paper

ESFuelCell2011-54382

DRAFT

SYSTEMS ENGINEERING DEVELOPMENT ON A STUDENT-DESIGNED B20 PLUG-IN HYBRID ELECTRIC VEHICLE FOR THE ECOCAR COMPETITION

Sean Carter

Graduate Teaching Assistant at Embry-Riddle
Aeronautical University
Daytona Beach, FL, USA

Darris White

Associate Professor at Embry-Riddle Aeronautical
University
Daytona Beach, FL, USA

Ashley Karr

Graduate Assistant at Embry-Riddle Aeronautical
University
Daytona Beach, FL, USA

Dahai Liu

Associate Professor at Embry-Riddle Aeronautical
University
Daytona Beach, FL, USA

ABSTRACT

Systems engineering plays a vital role in project and production development. The complete life cycle of a project incorporates systems engineering throughout. Systems are involved from the beginning, involving the conceptual design and planning processes, to the very end with testing procedures and production finalization. The topic involves subject matter as testing procedures, validating data, verifying the requirements, designing for failure modes, instigating faults into the system, and most importantly developing requirements for the system continuously throughout the life cycle.

The Department of Energy and many other large organizations use systems engineering to plan, process, and produce multiple types of products and projects. A lot of companies also follow this process due to the usefulness and productivity improvement. Engineers that learn this process are able to help streamline their area of expertise.

General Motors along with Argonne National Laboratory and The Department of Energy have sponsored a competition named EcoCar: The NeXt Challenge. This competition requires a lot of planning by both the organizers and the collegiate students involved. A lot of requirements were developed and then given to the universities for the development of a hybrid vehicle architecture that would be successful after three years.

The EcoEagles were required to develop documentation while following the simple guidelines of systems engineering. The paper plans to describe each area of systems engineering

the team used along with what was learned and how the team benefitted from using the process. The paper more importantly plans to go into detail how systems engineering practices are vital to project development, safe products, and quality production standards.

INTRODUCTION

System evaluation is the assessment and examination of a system or system element [1]. With system evaluations and assessments, these tools help determine whether or not the system itself is on track and meeting the end goal desired. The evaluations derived from the system are continuous through the product's life cycle and only stop once the product is no longer existent. With all of the newly developed technological advancements, there arise new procedures and protocols that have to be developed and evaluated to ensure the safety of use by customers and co-workers alike. The automotive industry is such an example. Toyota, GM, Ford, Honda, Nissan, and some of the other automotive industries that provide luxury cars for the middle class world are starting to invest in electrical technology. The design of the electric vehicle is a pressing matter and grows in complexity with new powertrain components such as power inverters and transmission systems that use electric motors. Testing and evaluation plans are required more than ever and are constantly being updated and implemented in today's automotive production standards.

Many governing organizations, such as the US DOE AVTA and EPA, develop, define, and disseminate information,

requirements, and testing and evaluation procedures that affect how car manufacturers, like GM, design, build, test, evaluate, manufacture, and monitor their vehicles [2]. GM also uses these guidelines, along with Argonne National Labs, to help develop requirements and test and evaluation procedures for the EcoCar Challenge competition, shown in Figure 1 [3].

	MECHANICAL	ELECTRICAL	CONTROLS
Y1 Design Jan 2009 - Jun 2009	Life Cycle Analysis, Vehicle Architecture Selection, and Performance Modeling	Define Electrical Requirements	Control System Design
	CAD - Component	HIL Design/Setup	Brake Control and Slip/Steer HIL
	CAD - Routing and Integrations	Fractal Component Selection and Annotation	
Y2 Mute Vehicle + Investigated Focus on Outreach Jul 2009 - Jun 2010	Vehicle Modification	Vehicle Harness/System Design	HIL Finalization and Commissioning Setup
	Component Integration	Vehicle Harness Setup	HIL Tuning - Safety and Fault Mitigation Implementation
	Controls Integration and Vehicle Testing		
Y3 Optimization and Refinement Jul 2010 - Jun 2011	Aero, Lightweighting, Ride and Handling, Noise, Vibration and Harshness	Refinement and Optimization	Refinement and Optimization
	60% Budget - Vehicle Ready for Production		

Figure 1: EcoCar Timeline and Deliverables

From the start of year one, conceptual designing and requirement development took place by selecting vehicle architecture based on models and simulations run on hardware-in-the-loop systems. The second year of competition involved requirements integration through various HIL evaluation events, on-road safety evaluations (ORSE), a lane change challenge, a towing ability event, a vehicle design review (VDR), acceleration tests (0-60, 50-70), braking events, autocross, fuel consumption tests, well-to-wheel greenhouse gas emissions (GHG) tests, petroleum energy use (PEU), tailpipe emissions, AVL drive quality, dynamic consumer acceptability (DCA), static consumer acceptability (SCA), an electrical presentation, a mechanical presentation, a controls presentation, and an outreach program. The final year of competition includes consumer acceptance and refinement and optimization of the vehicle to near-showroom quality.

Ultimately requirement procurement, validation and verification, designed fault mitigation effects analysis (DFMEA), and fault tree analysis (FTA) are the important factors that take place within the competition.

OVERVIEW OF ECOCAR CHALLENGE AND THE ECOEAGLES

Embry-Riddle Aeronautical University (ERAU) is participating in the EcoCar: The NeXt Challenge competition. This is a three year collegiate competition sponsored by the Department of Energy (DOE), General Motors (GM), and Argonne National Labs (ANL). The advanced vehicle technology competition that ANL has organized and run has been successful in developing collegiate minds for the past 20 years.



Figure 2: The competition vehicle donated by GM.

The competition challenges 16 North American universities to reduce the environmental impact of a GM production vehicle by minimizing its fuel consumption and reducing the well to wheel green house gas (WTWGHG) emissions while retaining the vehicle's performance, safety, and consumer acceptability. Part of the competition requirements is to use real world vehicle development strategies and processes that would meet GM's standard practices and safety protocols. All of the sponsors of the competition provide teams with engineering tools, equipment needed to create a realistic vehicle, and project design support to the teams throughout the competition. The ERAU team named EcoEagles has devised a Plug-In Hybrid Electric Vehicle (PHEV) propulsion system.

EcoEagles is accomplishing this by designing a PHEV that can drive 20 miles on all electric by implementing a 1.3L GM diesel engine and an A123 330V Lithium-Ion 12.8kWhr battery pack. The average commuter travels 33 miles [4] between work and home each day. The EcoEagles vehicle could achieve an improved efficiency in total diesel energy use by 40-50% just by not running the engine until the battery is discharged. Utilizing the diesel not only improves efficiency, but also emissions by using biodiesel (B20), which the EcoEagles plans to make.



Figure 3: The ERAU EcoEagles.

The competition is in the phase of year three which requires optimization of all systems on the fleet vehicle

utilizing systems engineering practices. Ultimately for control systems, the drive quality and operation of all the powertrain components must be up to GM's 99% production standards. Driver intended abilities must be met while also allowing safe operation using Design Failure Mode Effects Analysis (DFMEA). Safety protocols have to be in place and ready to engage for any "what if" events.

Systems Engineering is an important factor in overall product development. The paper will go into detail on communication between the LabVIEW programming language, National Instruments (NI) hardware, drive quality, performance improvement, and design milestones. This paper will also describe system engineering methods used to develop and test the vehicle control code including the uses of (DFMEA), Fault Tree Analysis (FTA) and Hardware-in-the-Loop (HIL) testing.

APPLYING SYSTEMS ENGINEERING TO THE ECOCAR

The EcoEagles team developed their own architecture, design, and test and evaluation plan to help meet the competition requirements [3]. The designs, architecture decisions, and testing procedures were all based off of overall requirements presented to the EcoEagles from the sponsors. The documentation was procured through the hard work of graduate students within the Human Factors and Systems Department. Through their studies, the students were learning the different aspects and importance of systems engineering and how it is involved with product and project management.

The important aspects of systems engineering that are important to the competition are the validation and verification of the results obtained through the design process, the fault-tree insertion into the different aspects of the project to ensure safe operation and safety of the driver, and DFMEA for continuous change and observation of the high risk priority items. In year one, DFMEA and FTA were integrated into the project and growing consistently as more knowledge was obtained about the requirements and what problems could occur. Year two involved more refinement of the DFMEA and FTA documentation that was developed while also integrating test plans to ensure safe driver operation. Year three has incorporated a lot more verification and validation to clarify that the EcoEagles team has stayed on track and are meeting the requirements that were set back in year one. The past year has also involved a lot more testing procedure development and validation of the results from these tests.

DFMEA AND ECOCAR

DFMEA is a systematic team driven approach that identifies potential failure modes in a system, product, or manufacturing / assembly operation caused by either design or manufacturing / assembly process deficiencies [5]. This analysis is done to organize the critical safety systems from sub systems that may cause lower risk errors. The design is a process that allows engineers to verify and validate the risk priority of the problems involved with the project and allow

safety measure to be implemented into the system to help lower the overall risk if the problem occurs.

EcoCar students had to go through in the first year of the competition and procure documentation for future reference throughout the competition. The EcoEagles worked in cooperation with the Human Factors and Systems Department to create the documentation required for the competition and for the assurance of vehicle operational safety for the future driver of the vehicle. Figure 4 shows an example of what the DFMEA documentation looks like.

Example DFMEA Worksheet

Function	Failure mode	Effect	S (severity index)	O (occurrence index)	D (detectability index)	RPN (Risk Priority Number)	Control Plan	Other (related parameters, related)	RPN (new)	Responsible person	Response/Action	
FT 1.0	High speed sensor malfunction	Speed sensor not working properly	3	2	5	30	FT 1.0.01 based on FT 1.0.1 to low speed sensor		4	CC	Review and analysis of vehicle control system for high speed sensor and high speed sensor	See the 10/04/02

Figure 4: DFMEA Documentation Layout

With this layout, the systems engineering graduate class were able to provide quality and key information that was vital to the safe operation of the EcoEagles vehicle. Because of the development of this documentation, the EcoEagles were able to apply the information obtained to the project and update where necessary. The EcoEagles managed to lower the risk priority number (RPN) on numerous points listed within the archive of information given and obtained a higher quality of assurance that the vehicle is safer to drive.

FTA IMPLEMENTATION INTO ECOCAR

FTA is a deductive, failure-based approach [6]. The process involves introducing failures into a system to yield results. With FTA, the process can be used to evaluate design alternatives and to establish performance-based design on the faults instigated [6]. The faults put into the system can range from minor to critical and obtain results of equal value.

The FTA documentation was created by the systems engineering students enrolled in the Human Factors Safety degree program. With the new information given to the EcoEagles team, controller code and mechanical failure systems were investigated. Accelerator pedal failure, CAN messaging latency, CAN communication loss, and many more problems were introduced into the system development involved with the EcoEagles vehicle. By creating these faults and testing the EcoEagles new powertrain and control systems, the results were being verified and validated and reducing the RPN of the DFMEA that was set forth from year one. Figure 5 below shows an example of a FTA when figuring out possible failures to the problems faced by the EcoEagles.

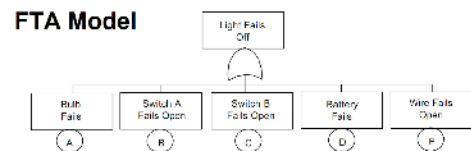


Figure 5: Example of FTA on a light bulb. [7]

Investigating the different faults and being able to produce data from these tests was vital to the success of the EcoEagles. Through FTA, the EcoEagles managed to reduce a lot of the high RPN listed within the DFMEA documentation and verify and validate the requirements procured from the competition requirements along with the team's personal requirements. The EcoEagles through year two were able to use FTA to reduce the overall risk and create a better safety factor involved with the project vehicle.

VALIDATION AND VERIFICATION IMPORTANCE TO ECOCAR

Validation and verification is a continuing process of looking to the original design criterion and checking to make sure that the design process and product meet the requirements stated. For each step of the development phase, the project goes through and makes sure that the newest addition to the product, or in this case the vehicle, meets the requirements stated. The project begins the verification and validation check once the project reaches a point to where all of the parts start to become integrated into the system. As the life cycle of the project increases, the more validation and verification is important. The cycle begins with integration, testing, and verification, and then the process goes into system verification and validation and finally ending on operations and maintenance. Figure 6 represents the basic idea of validation and verification.

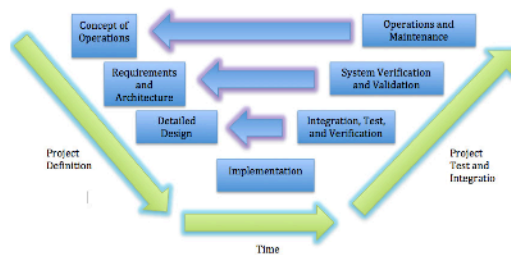


Figure 6: Validation and Verification process.

With the verification and validation process, the EcoEagles were able to develop software and incorporate the mechanical systems into the vehicle and meet the requirements. The code was developed using FTA and DFMEA to ensure safety and requirements set by the competition sponsors. As the development of the vehicle was progressing, the team was checking back with the original requirements to make sure that the current progress was validated through results and verifying that the results are correct and meet the requirements.

DISCUSSION & CONCLUSION

Systems engineering is an important aspect to any production part made or project that may just be starting. DFMEA is important due to the emphasis on RPN and

prioritizing the critical safety systems that need assistance with ensuring safety. The EcoEagles has learned a lot from DFMEA through utilizing the documentation and the requirements provided by the competition sponsors. Without the help of the Human Factors and Systems department, the documentation would not have been developed properly procured and developed under the systems engineering disciplines taught within the department. The lessons learned by DFMEA are few and far between. The document is an ever-growing one, and you can never have too long of a list. The EcoEagles have learned to write down as many as possible to overcome any possible fault that may occur within the project development phases.

FTA has also played an important role to the EcoEagles. With the instigation of the faults into the different systems, the vehicle has now passed safety inspection in the recent year three spring workshop. A lot of the inspection was passed due to FTA that was created in year two and one. The EcoEagles learned a lot from FTA and definitely refined the development of the vehicle through the amount of FTA testing accomplished.

Validation and verification is the most important of all three mentioned throughout this paper. The process ties in the FTA and DFMEA to ensure a working and safe vehicle. With the process, the EcoEagles were able to look back at the requirements and verify that we were meeting them by validating the results from the data retrieved through testing. The testing procedures used were also developed by the Systems Engineering class and were used extensively in the validation and verification phases.

In conclusion, the hard work done by both the senior design class for High Performance Vehicles and the Systems Engineering class from the Human Factors and Systems Department has benefited the EcoCar program at ERAU immensely and has helped tie in Systems Engineering into the engineering academic program. A lot of the lessons learned throughout the competition have been through looking back throughout the requirements, and by checking over all of the documentation that once put the team on the right track in the first place.

ACKNOWLEDGMENTS

A lot of the work for this paper could not have been done if it were not for the hard work of the graduate students within the Systems Engineering class for Human Factors and Systems. Another thank you that is important are the advisors that have helped make this project a success and all of the team members that have made the project a fun experience. Last, but most important of all, are the sponsors of the EcoCar: The NeXt Challenge competition. Due to the hard work done by the people from Argonne National Labs, General Motors, A123, National Instruments, dSPACE, Environmental Protection Agency, and the Department of Energy, this project would not have been a great success.

REFERENCES

- [1] Blanchard, B. S. and W. J. Fabrycky. Systems Engineering and Analysis. 3rd Edition. Upper Saddle River: Prentice Hall, 1998.
- [2] Karner, Donald, et al. "Plug-in Hybrid Electric Vehicle (PHEV) Integrated Test Plan and Evaluation Program." Draft. U.S. Department of Energy FreedomCAR & Vehicle Technologies Program Advanced Vehicle Testing Activity, 2007.
- [3] EcoCar. "2010 EcoCar Competition Finals Year 2 - Integration and Outreach." US DOE and GM, 2010.
- [4] Langer, Gary. ABC News. 13 February 2005. 14 March 2011
- [5] Quality One International. Quality-One. 2011. 14 March 2011 <<http://www.quality-one.com/services/fmea.php>>.
- [6] Dr. William Vesely, SAIC. Fault Tree Handbook with Aerospace Applications. Handbook. NASA. Washington: NASA, 2002.
- [7] II, Clifton A. Ericson. "Fault Tree Analysis." Project. 1999.

Appendix B

ASME 2011 International Mechanical Engineering Congress & Exposition Paper

IMECE2011-65495

**PROGRESSION OVER THE YEARS: THE ECOEAGLES STUDENT-DESIGNED BIODIESEL
PLUG-IN HYBRID ELECTRIC VEHICLE**

Sean Carter

Jenna Beckwith

Dr. Marc Compere

Dr. Darris White

Brandon Smith

Zachary Karstetter

Brian Harries

Department of Mechanical Engineering
Embry-Riddle EcoEagles
Embry-Riddle Aeronautical University
600 S. Clyde Morris Blvd.,
Daytona Beach, FL 32114 USA

ABSTRACT

The Embry-Riddle Aeronautical University (ERAU) EcoEagles are participating in the EcoCar: The NeXt Challenge competition. The competition is a three-year collegiate event where 16 teams from North America compete to build a more efficient and better performing GM production vehicle. The three year collegiate competition is sponsored by the Department of Energy (DOE), General Motors (GM), and Argonne National Labs (ANL). The advanced vehicle technology competition has a history, and has been organized and ran for the past 20 years.

The competition challenges collegiate minds to reduce the environmental impact of a Chevrolet EcoCAR by minimizing fuel consumption and reducing emissions while retaining the vehicle's performance, safety, and consumer appeal. The main focus of the competition is to use real world vehicle development strategies and processes that would meet GM's standard practices and safety protocols. All of the sponsors of the competition provide teams with engineering tools, equipment needed to create a realistic vehicle, and project design support to the teams throughout the competition. The ERAU team, the EcoEagles, has successfully devised a Plug-In Hybrid Electric Vehicle (PHEV) propulsion system that meets those requirements.

The electrification of the powertrain and the use of biodiesel fuel are central themes in the EcoEagles' strategy for improving fuel economy and tailpipe emissions. The team selected an electric range of approximately 25 miles based on the average commuter driving less than 33 miles per day [1]; meaning that most of the vehicle operation will be conducted using either fully electric or electric-assisted propulsion. The vehicle design consideration was accomplished by

implementing a 1.3L GM Turbo Diesel coupled with a 2-Mode electrically variable transmission (EVT) and an A123 Lithium-Ion Iron-Phosphate 330V 12.8kWhr battery pack. The EcoEagles design will reduce petroleum energy consumption by 78%, improve fuel economy by 66%, and reduce well-to-wheel greenhouse gas (WTWGHG) emissions by 30%.

The paper will focus on the 99% production readiness. The paper will also discuss and include vehicle test data supporting the energy efficiency, emissions, and performance / utility capabilities of the vehicle as determined by the first two years of vehicle development. The vehicle architecture and background information will also be presented to help the reader understand why the given architecture was chosen and how it might compare to the Chevrolet EcoCAR. Performance predictions made from simulations will be contrasted against those from the Hardware-in-the-Loop (HIL) development. Finally, on-road testing will also be compared with the same predictions with the goal of showing why the model-based, HIL enhanced, and vehicle technical specifications (VTS) did or did not agree.

INTRODUCTION

EcoCAR: The NeXt Challenge is a supported effort by the DOE, GM, and National Resources Canada in order to promote the development of cleaner, more efficient vehicles as part of a comprehensive educational program. The EcoEagles team represents ERAU in this three-year competition. The design and technical goals for the competition are to reduce petroleum energy consumption, reduce well-to-wheel green house gas (WTWGHG) emissions, and increase vehicle energy efficiency, all while maintaining consumer

acceptability in the areas of utility, safety, and performance. Vehicle electrification was identified as a key technology for this project because of the availability and efficiency of electricity and electric power systems. The modern automobile is the result of over a century of technological evolution.

Multiple types of propulsion technologies have been attempted with electric, hybrid-electric, and plug-in hybrid vehicles; most of these technologies began development in the late 1800s. GM developed an experimental plug-in hybrid vehicle, called the XP-883, in 1969 [2]. Despite notable efforts to increase the degree of vehicle electrification, the cost, weight, and complexity of these systems has prevented widespread market acceptance. Current advances in battery and control system technologies, along with increased awareness of the environmental impact of petroleum energy use, have resulted in new opportunities for hybrid vehicle development and powertrain electrification. The EcoEagles' PHEV system features a high degree of vehicle electrification including: an all-electric driving range of 25 miles, all electric accessories, and plug-in charging.

VEHICLE ARCHITECTURE SELECTION AND DEVELOPMENT

The EcoEagles team has adhered to a simplified version of GM's Global Development plan. The development process can be divided into four phases: concept evaluation, design, prototype, and pre-production. At the project initialization, the EcoCAR competition organizers and GM provided a description of the project goals and a list of minimum requirements for the vehicle, representing the Document of Strategic Intent. Based on the information, the EcoEagles team organized themselves into vehicle development groups and began the process of defining the requirements for our vehicle through research and evaluation of design concepts.

The team selected a conceptual powertrain configuration based on the competition requirements and VTS. The VTS were used to determine the engineering specifications for the vehicle and related components, which drove the selection of each component in the PHEV system. The selection of the powertrain configuration and components marked the completion of the conceptual evaluation phase of the project. Year one concluded before the Fall 2009 term, during which time they used a range of design tools to evaluate solutions to structural, thermal, and control system challenges.

The integration of the vehicle components and prototype phase was accomplished during year two, before the Fall 2010 term. The EcoEagles managed this through use of software-in-the-loop (SIL), hardware-in-the-loop (HIL) testing, and computer aided drafting (CAD). Students worked closely with sponsors to ensure their designs promoted vehicle safety and proper operation. The team this past year was working on refinement and optimization of the mule vehicles systems. The mule vehicle was a prototype test vehicle, with working, yet unrefined powertrain systems. The prototype phase concludes at the 65% design review, which was the Year Two competition in May 2010. The pre-production phase of the

competition includes the refinement of the mule vehicle into a production ready vehicle. This phase will conclude at the 99% design review, which was the Year Three competition in June 2011. If the PHEV design were slated for production, there would be an additional production phase in the VDP to include manufacturing and final refinements.

The goal of the powertrain configuration process was to determine the optimal propulsion system configuration that could be built with the resources available to the ERAU team. Preliminary research indicated that fuel cell vehicle and electric vehicle technologies are not currently sufficient to meet the minimum range, weight, and volume requirements for this project. The remaining options allowed by the competition requirements include a range of hybrid and plug-in hybrid configurations, and fuel selection of B20, E85, and H2.

To evaluate potential design, the Powertrain Systems Analysis Toolkit (PSAT) from ANL was used. PSAT provides a graphical user interface to Simulink, predefined hybrid-electric vehicle configurations, and many preconfigured OEM component models, making it an ideal tool for the rapid development of vehicle models. The baseline model of the vehicle used the following parameters, which were provided to the team from GM and ANL:

- Vehicle Mass: 1742 kg
- Engine Power: 123 kW
- Mechanical Accessory Load: 0 Watt
- Electrical Accessory Load: 300 Watt
- Road Load Equation: $F = 112.85N + 4.60 * v + 0.542 * v^2$

For plug-in hybrid vehicles, a Utility Factor (UF) is used to measure the percentage of travel that uses electrical energy and is one indication of the degree of vehicle electrification. To evaluate the influence of utility factor on vehicle performance, baseline PHEV models were created in PSAT.

Approximately 50% of daily travel distances are less than 25 miles [3]. The team originally selected a charge depleting range of 18 miles and the final distance of 25 miles was selected based on battery constraints and consultation with the battery module manufacturer, A123 Systems.

A123 Systems produces an energy storage system (ESS) that meets energy storage requirements, while meeting the packaging and weight requirements for the vehicle. The final configuration, consisting of four 25S2P battery modules totaling 12.8 kWh at 330V, is capable of 25 miles of all-electric operation.

B20 architectures have better fuel economies and lower greenhouse gas emissions, but higher petroleum energy use than E85 architectures. B20 has a higher energy density when burned in a diesel engine; so little fuel is required to go further. E85 is more of a biofuel than B20, however, it requires more to burn to obtain the same distance B20 could obtain. B20 was selected as the fuel source using a weighted average decision matrix. These effects were then ranked based

on their importance in the EcoCAR competition regard to scoring.

Another factor in the fuel selection was the list of supported engines, which included 1.3 L turbo diesel, 2.0 L turbo diesel, 1.6 L gas, and 1.8 L gas engines. The 1.3 L turbo diesel engine could be packaged with a wide range of hardware, including GM's front wheel drive two-mode transmission, without significant chassis modifications. Since all four engines met the minimum torque and power requirements determined for this project, the 1.3 L turbo diesel engine was selected.

VEHICLE CONTROL SYSTEM: HARDWARE

The ERAU EcoEagles control system architecture consists of several controllers communicating on two separate high-speed controller area network (CAN) busses. These two vehicle buses are the General Motors Local Area Network (GMLAN) and the Powertrain Extended Bus (PTEB). Those busses are located on the traction power inverter module (TPIM) side of the Gateway. The ERAU High Speed (HS) and ERAU PTEB busses, shown in Figure 1, interface with the National Instruments (NI) control boards. The architecture layout was designed to allow the team to isolate the new battery (BPCM) and engine (ECM) controllers from the stock vehicle.

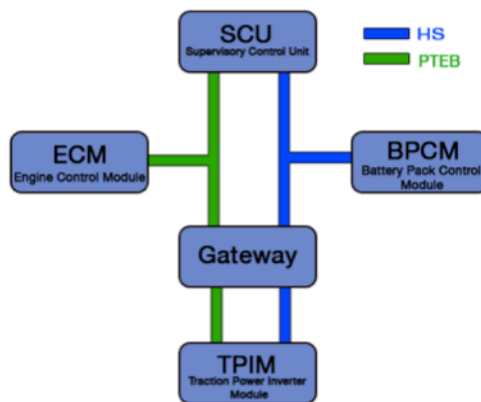


Figure 1: Control System Architecture

The following list includes the controllers used by the EcoEagles or the controllers that are interfaced in order to communicate with the vehicle and ensure safe operation.

- Supervisory Control Unit (SCU) – The controller board chosen for the SCU is a NI Single Board Reconfigurable Input / Output-9642 (sbRIO-9642). The sbRIO-9642 is used to implement ERAU's main control strategy. Driver inputs and the status of powertrain components are received in order to initiate safe and efficient operating modes. The SCU also controls subsystem and control module power through analog and digital outputs. These subsystems include: selective catalytic reduction (SCR), fuel management, engine cooling, and battery cooling systems.
- Gateway (GW) – A NI sbRIO-9602 was used for the gateway. The primary functions of the gateway are to isolate the new control modules from the stock controllers, and to add additional processing power to run safety critical algorithms at acceptable speeds.
- Engine Control Module (ECM) – The ECM is the OEM controller for the 1.3 L SDE, responsible for direct control over engine operations. The ECM is controlled through analog outputs from the SCU and CAN messaging on the power train expansion bus (PTEB). Messages from the ECM are allowed to pass through the gateway during engine operation.
- Traction Power inverter Module (TPIM) – The TPIM is the OEM controller for the two-mode hybrid transmission. Control of the two-mode is implemented through CAN communication between the SCU and the TPIM.
- Battery Pack Control Module (BPCM) – A123 Systems controller that monitors battery pack operation. Communicates over CAN with the SCU and the BRUSA charger. Maintains safe operation of the battery during driving and charging conditions

The sbRIO-9642 and sbRIO-9602 boards were chosen for their form factor, hardware capabilities, and for the utilization the competition and the EcoEagles team required. The boards are shown in Figure 2 and 3 on the next page.

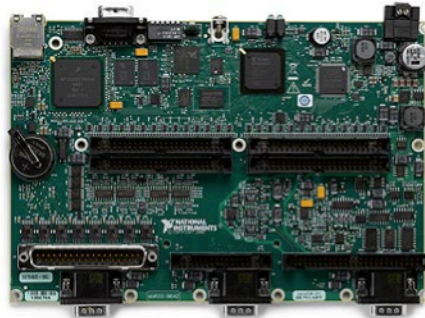


Figure 2: NI sbRIO – 9642



Figure 3: NI sbRIO – 9602

VEHICLE CONTROL SYSTEM: GOALS AND MODES

The control system was designed and optimized for safe and efficient operation of the EcoEagles powertrain. The unique powertrain configuration of the 2-Mode transmission coupled with the 1.3 L turbo diesel engine allows the EcoEagles to propel the vehicle in several different modes of operation. These modes are dependent on the current operating conditions of the vehicle, including state of charge (SOC), driver torque demands, vehicle speed, and engine speed. The main modes of operation are Mode 1 engine off, Mode 1 engine on, and Mode 2 engine on. Mode 1 operation is for low speed driving under 27 mph and is where the all-electric driving takes place. The EcoEagles vehicle acts as a series hybrid during Mode 1 operation; the engine is turned on only when SOC is low or just prior to shifting into Mode 2. Mode 2 of vehicle operation is like a power-split hybrid, splitting the power between the transmission and engine for charging or helping propel the vehicle. Control of mode operation is done using two coupled state machines. These state machines utilize LabVIEW’s state-chart functionality. One state chart controls engine modes and operation, while the second is used to control transmission shift modes.

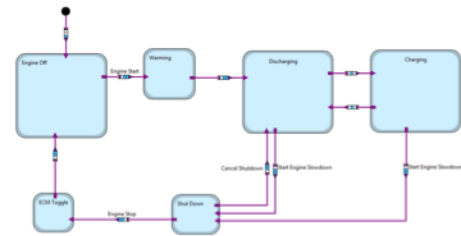


Figure 4: Engine operating modes state-chart

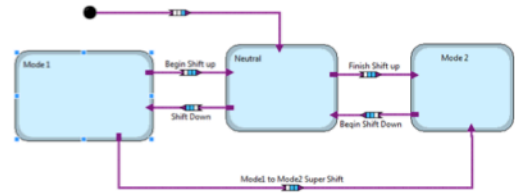


Figure 5: Transmission shift modes state-chart

The engine states include: engine off, engine warming, engine on, and engine idle for shutdown. Inside the engine on state, there are different modes of operation depending on the state of charge of the battery. These internal states are engine on SOC low, engine on SOC high and engine on. In the engine on state, the power demand at the wheels is used to determine the power input from the engine. The engine operation is controlled using the analog pedal signal from the SCU and CAN messaging to the TPIM to control engine RPM. The operating states of the diesel engine were determined by analyzing the brake specific fuel consumption data. The data was then linearized with a 95% fit to create functions for calculating the throttle position and RPM to produce the necessary power. In the *engine on SOC low* state the power demands at the wheel are overshot in order to recharge the battery to a safe operating condition. In the *engine on SOC high* state the power produced by the engine is less than the power needed at the wheels in order to deplete the battery.

The shift states include Neutral, Mode 1, and Mode 2. In most operating conditions, a shift from Mode 1 to Mode 2 occurs at 25 mph. The shift down from Mode 2 to Mode 1 occurs at 17 mph. If all of the conditions for the upshift are not met, then the car is put into the neutral state until conditions are met to return to either a Mode 1 or Mode 2 state. The Intelligent Driver Efficiency Assistant (IDEA) is another integral part in deciding the SOC low engine on and off state. The SCU has strict conditions, where if they are met the IDEA car computer will decide whether or not to turn the engine on based off of key factors such as vehicle location, route travel, speed, and driver history. This system is utilized to help reduce the overall diesel usage and to make sure the vehicle performs just as expected when travelling the same route.

VEHICLE CONTROL SYSTEM: SAFETY

HIL setup and integration played some key roles when developing the control strategy for the 2-Mode transmission and 1.3 L turbo diesel engine. The 2-Mode transmission is a very complex system and in order to insure safe operation, software and hardware in the loop simulations are required. Bench testing was done through SIL and HIL systems by utilizing ERAU's control boards and the programming computers as a means to test and see what would happen with purely driving input and with other controller inputs, such as the TPIM. Some areas of testing in a controlled simulation environment were, power train failures including broken input shafts, shift strategy, 12V system dips, and ground fault detection. These simulations allowed ERAU to instigate faults to test control strategies and failure modes without endangering students or vehicle components.

An input shaft failure results in an almost complete loss of power train functionality and requires immediate servicing. Due to the severity of this failure it was important to run simulations to ensure driver safety and mitigate powertrain damage. A control strategy was developed to identify that the failure had occurred, by comparing the difference in rotational speeds of the engine and the input shaft and then take immediate action to protect the vehicle. When the fault occurs the engine and fuel pump are turned off immediately to avoid the unloaded engine from accelerating uncontrolled. While the ECM is turned off, the SCU fakes the ECM, limits the top speed, and shifts the transmission to a neutral state until the vehicle has slowed down enough to allow mode 1 operation. These simulation tests created a safe environment to implement and refine these fault mitigation tests. Unfortunately ERAU had a chance to validate these failure modes during an actual input shaft failure, where these failure modes were safely implemented on the vehicle itself.

The mule vehicle itself has proven to be the most valuable HIL when considering consumer acceptability. HIL and SIL simulations do not simulate the way it feels to actually drive the car; therefore, real driving time was very important for year 3 refinements. Drive time is invaluable specifically when considering engine operational modes, regenerative braking and shifting strategies. To ensure safe shifting strategies, within a controlled environment, drive testing was used to see how well the vehicle performed under different driving styles and conditions. Simulations were used to validate safe operation of regenerative braking modes, and then real driving tests were done to collect data used to optimize the regenerative system for consumer acceptability and efficiency.

Even after careful modification to the 12V system, the addition of the extra components degraded its robustness and created intermittent faults that proved difficult to predict in simulations. The NI control boards and the BPCM are system critical modules that are particularly affected by fluctuations in the 12V system. These intermittent failure conditions were discovered through vehicle testing. To mitigate the effect of these 12V failures, software simulations were conducted to

control the safe shut down of the vehicle in the event of a 12V fluctuation causing a loss of communication to the BPCM. To prevent these failures from occurring the TPIM's voltage regulating capabilities were used to maintain higher charge on the 12V system during vehicle operation. This causes slightly higher energy consumption but these losses seem negligible compared to the huge gains in consumer acceptability by avoiding these faults.

Another idea to be integrated, after validation, will be the incorporation of a capacitor parallel with the 12V bus to help reduce the amount of surging current on vehicle startup. The idea came from a problem with communication and contactor control from the BPCM. The voltage dip occurred when the BPCM would commence with closing contactors upon vehicle startup. This would sometimes dip the 12V bus low enough to reset or drop communication with the vehicle. Tying in the capacitor with the 12V bus would enable the BPCM to properly close contactors within the battery pack and help maintain power for proper communication on the CAN buses.

Another problem was occurring within the distribution and disconnect enclosure (DDE). The ground fault detection system was not functioning properly, but the A123 system had an integrated ground fault detection system as well. After the choice was made to use the A123 ground fault detection system, rather than the external Bender system, ground faults could be instigated in a simulation environment. If a ground fault were to occur, the SCU ensures that the 2-Mode does not power on and notifies the driver while ensuring the car will not be able to crank or allow any operation of the vehicle powertrain. Driver notification was validated at the Environmental Protection Agency (EPA) and at the GM proving grounds and was proven to effectively disable the vehicle from any powertrain operation.

OTHER FEATURES AND UNIQUE VEHICLE ATTRIBUTES

Other aspects that have been modified to the vehicle to improve consumer acceptability and overall drive quality are the IDEA system, airbag placement within the rear springs, and slight aerodynamic modifications. Currently, the IDEA system is implemented into the car and ready for optimization. The IDEA system will enable the driver to save on fuel when driving the usual to and from work routine seen in a typical day. The IDEA system will analyze when the driver accelerates, where is home located for the driver, and typical traffic patterns seen in the area. This system will improve the overall efficiency of the vehicle by helping the driver use less fuel when travelling, and by preparing the vehicle for any measure of performance when the driver most needs it, like accelerating onto a freeway.

The airbag implementation was done to help stiffen the rear springs of the vehicle. The battery pack weighs considerably more than the stock pack and has lowered the ride height of the vehicle toward the rear. To help improve drive quality, and to improve traction, the airbag

implementation helps stiffen the ride so the vehicle will have a closer stock driving quality.

Computational Fluid Dynamic (CFD) analysis was performed on the stock Chevrolet EcoCAR to validate the provided data as well as locate areas on the vehicle that would be good candidates for improvement. The implementation of aerodynamic changes could help offset the detrimental effects of the reduction in powertrain power and added weight from ESS and component integration onto the car. Initial CFD analysis proved to have less than 1% difference in the coefficient of drag from GM provided data; an additional case was run, to prevent boundary layer build up under the vehicle, with a moving floor and corresponding vehicle components rotating, yielding an approximate difference of 3.5% from the stock GM data. The prominent areas of concern are the fog light recesses and bumper cutouts, front air dam, and the rear D-pillars, all of these areas can be modified without drastically changing the stock appearance of the EcoCAR.



Figure 6: Bumper Cut-Out Aerodynamic Modification

Stream traces were used to see if the flow through the bumper cutouts were vital for engine cooling or not; the cutouts that were allowing flow to pass that was not affecting cooling were covered to help smooth out the overall shape of the front bumper. The fog light recesses in the front bumper show areas of higher pressure than the surrounding areas due to the recirculation of flow into the opening; the current enhancement covers the fog light openings creating a smooth bumper with better flow characteristics and retaining the ability for addition of actual fog lights in the stock location.



Figure 7: Air Dam Aerodynamic Modification

The front air dam was extended using a strong lightweight carbon fiber addition, while still maintaining VTS ride height. The extension on the air dam should help direct more flow around the vehicle keeping unnecessary flow from under body components while still allowing enough flow for brake cooling purposes. Flow through underbody components results in unwanted high-pressure regions and turbulent flow on the bottom side of the car; eliminating these high-pressure regions could potentially increase down force and handling characteristics.



Figure 8: D-Pillar Aerodynamic Modification

D-pillar winglets were also redone to sharpen the stock body curve to increase the pressure behind the vehicle. The increase of pressure behind the vehicle acts as a pushing force on the vehicle aiding forward movement by reducing power required to maintain speed. These aerodynamic enhancements underwent experimental coast down testing and are still being validated to observe a reduction in coefficient of drag. The overall goal of these modifications is to reduce the coefficient of drag while enhancing or maintaining dynamic performance and consumer acceptability.

The IDEA system computer that was installed in place of the radio required an adapter bezel so that it would fit in the center dash. The bezel was designed by taking the appropriate measurements to create a clean presentation of the computer and make it as flush as possible with both the computer and the center dash of the vehicle. The final product in CATIA is below:

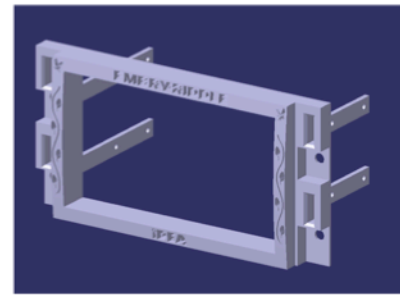


Figure 6: IDEA computer bezel in CATIA

The arms extending from the back were designed to be screwed into the sides of the computer. The holes on the sides are for connections between the bezel and the car and are the same as those originally on the car's radio. To take advantage of the 3D Printing process's level of detail, the front was customized for Embry-Riddle's EcoCar Team.



Figure 7: Completed bezel in vehicle

The bezel was produced by COHO Designs, a Rapid Prototyping company located in Palm Coast, FL. The method used by COHO Designs is Fused Deposition Modeling (FDM). We chose this for having our bezel produced to highlight the capabilities of the Rapid Prototyping process and to maximize its customizability. The small level of detail this process allowed us to create a unique product.

The Selective Catalytic Reduction (SCR) system has been designed and implemented over the past year. First the individual components were tested and calibrated on a test bench setup. Then a control code was developed that utilizes sliding mode control strategy and real time information from various sensors and engine data. A circuit board was designed and printed to use for final packaging on the vehicle. The entire system was moved from the test bench and mounted to the vehicle.

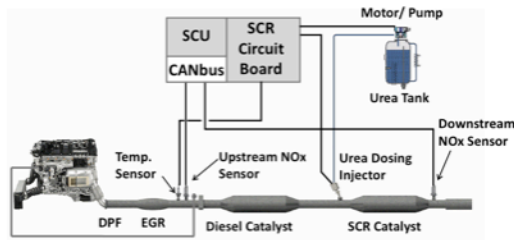


Figure 8: System Components

Gaining control of the initial components proved to be a challenge. With a purchased Volkswagen SCR system, we attempted to gain control of the individual components. The urea injector needed to be controlled with a pulse width signal

(PWM) and developing circuitry to handle relatively high power, very fast switching was undertaken. Consistent control was gained using a special injector chip. The injector chip was also suitable for running the heater of the SCR system. The original motor on the VW components was proprietary and the proper control signal could not be obtained. A new motor was specked out and was mounted in place of the original motor. A Pololu Simple Motor Controller controls the new motor.

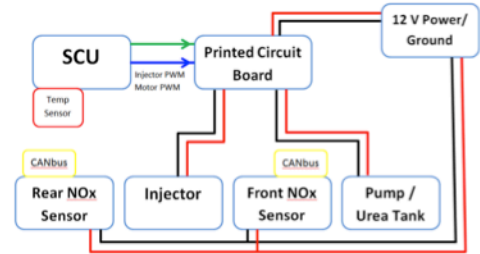


Figure 9: System Control Circuitry

As testing and calibration was being done with all of the controlled components on the test bench, a control strategy was also being developed. Using feedback from the engine, a temperature sensor and both an upstream and downstream NOx sensor, a sliding mode control strategy was developed. The control strategy will use this real time data to control the PWM of the urea injector to reduce NOx as much as 90% while allowing for only minimal ammonia slip.

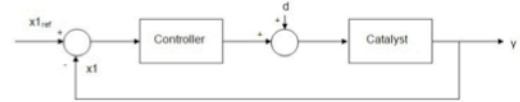


Figure 10: Schematic representation of the control loop.

The system was mounted to the vehicle so that emissions testing could begin at the EPA facility on their SEMTECH unit. Initial calibration of the entire SCR system began with the data from the EPA dynamometer. Because our system had not previously been used, it has taken some time to saturate the catalyst with ammonia. Due to the catalyst saturation affect, NOx reductions of only 50% were achieved at the EPA facility.

PREDICTED VTS GOALS

The EcoEagles developed their VTS through the efforts of PSAT, SIL, and HIL development. PSAT played a vital role in choosing the architecture and allowing the EcoEagles to plan accordingly for the three-year competition. Through PSAT, the team was able to determine which architectures would be most beneficial to develop and the pros and cons between them.

The SIL and HIL development allowed clarification, verification, and validation of the vehicle operating parameters. Bench testing the SCU and GW with fault mitigation has helped ensure a safe vehicle, but the competition results show the EcoEagles how well the VTS was predicted. Testing in the areas of braking, acceleration, lane changing, emissions, towing capacity, cargo capacity, passenger capacity, and starting time are among the few that were tested at the GM proving grounds in Milford, Michigan.

The EcoEagles had to research into the different types of hybrid technologies to achieve the team's current successes. The team mainly researched into the series, parallel, and compound technologies to develop the current architecture. The EcoEagles noticed that series hybrids are great for low to mid speed applications and utilize the engine only to recharge the battery on the go. However, high speeds, like the interstate, lower the efficiency of the motor and reduce the overall efficiency of the system by almost keeping the engine on consistently to keep a charge. Performance is also lost unless the vehicle is equipped with a powerful enough motor. The more powerful the motor, however, the more weight added to the car and the lower your miles per gallon gasoline equivalent. The team realized that even a parallel and compound system could have the same problems. Parallel

hybrids are unique by allowing the engine to help provide torque, and at a near constant RPM. The con of using a parallel hybrid is that most of the energy used from the powertrain system is petrol or diesel. There are also a lot of mechanical losses when converting the petrol energy into electrical and then having the vehicle use that energy, with its own efficiency losses by use of the electrical. This could be a costly factor when driving around the city because an engine performs better at a near constant and higher RPM for better fuel economy.

That could also be said for a series or compound hybrid. Compound hybrids are a combination of both series and parallel technologies. At low speeds, the compound hybrid will act as a series hybrid, and at high speeds a parallel. You get the best of both worlds and the cons tend to lower in risk with this kind of powertrain. An example of this powertrain is the 2-Mode that was chosen for the EcoEagles architecture. Coupling the diesel with the 2-Mode has allowed the team to benefit from the recharging capabilities of a series, and allow the more engine-based propulsion from a parallel at higher speeds. The combination of the 2-Mode with the diesel engine was ultimately set to meet the emissions requirements while trying to minimally meet the performance that came stock with the vehicle.

Specification	Competition Requirements		EcoEagles	EcoEagles Competition
	Production Vue	Competition Requirement or Target	Projected	Measured
EcoCar				
Accel 0-60	10.6 s	≤14 s	11.8 s	≥20 s*
Accel 50-70	5.7 s	≤10 s	8.6 s	
Towing Capacity	680 kg (1500 lb)	≥680 kg @ 3.5% grade, 20 min @ 72 kph (45 mph)	680 kg	N/A
Cargo Capacity	.83m ³	Height: 457 mm (18") Depth: 686 mm (27") Width: 762mm (30")	Height: 457 mm (18") Depth: 686 mm (27") Width: 762mm (30")	Height: 457 mm (12") Depth: 686 mm (27") Width: 762mm (30")
Passenger Capacity	5	≥4	5	5
Braking 60-0	38-43 m (123-140 ft)	<51.8 m (170ft)	46m (151 ft)	38.4m* (126 ft)
Mass	1758 kg (3875 lb)	2268kg (5000 lb)	1974 kg	2043
Starting Time	≤2 s	≤15 s	10 s	1 s
Fuel Consumption CAFÉ unadjusted, Combined, Team: UF Weighted	8.3 L/100 km (28.3 mpgge)	7.4L/100 km (32mpgge)	6.25 L/100 km (0.43 UF) (37.7 mpgge)	N/A
Charge Depleting Fuel Consumption	N/A	N/A	5.23 L/100 km (45.26 mpgge)	N/A
Charge Sustaining Fuel Consumption	N/A	N/A	7.4 L/100 km (32 mpgge)	N/A
Charge Depleting Range	N/A	N/A	38.54 km (23.95 mi)	N/A
Petroleum Use	0.85 kWh/km	0.77 kWh/km	0.40 kWh/km	N/A
Emissions	Tier II Bin 5	Tier II Bin 5	Tier II Bin 4	N/A
WTW GHG Emissions	250 g/km	224 g/km	158 g/km	N/A

* indicates calculation from CAN message data

Table 1: Vehicle Technical Specifications (VTS)

As work was accomplished through the three years, the vehicle has shown good signs of meeting the VTS requirements that were predicted from PSAT through HIL, SIL and actual vehicle testing. The VTS has changed over the years and the final VTS was shown in Table 1.

Because of the heavier vehicle, due to the battery pack and added hardware, performance was expected to be lost in key areas. Those areas are mainly the acceleration, braking distance, and overall startup time. The following table shows the results from over the past year and competition. The numbers will be further discussed in the following section.

PERFORMANCE TESTING AND RESULTS

The EcoEagles did dynamic testing over at the Daytona International Speedway (DIS) and as part of the competition, at the EPA and Milford Proving Grounds (MPG) in Ann Arbor and Milford, Michigan respectively. The team's goals at DIS were to accomplish acceleration testing, braking, aerodynamic drag from a coast down of 60 mph, and possibly some handling tests if the track organizers will allow. The team managed to acquire coast down data, slightly aggressive acceleration data, and moderate braking test data. The data allowed the EcoEagles to approximate acceleration, braking, and possible charge depleting range. The following table shows some of the results.

Category	Project VTS	Estimated	Actual Data
EV Range (km)	23.95	~20	~22
Overall Vehicle Range (miles)	320	~280	~290
Acceleration (0-60, 50-70) (sec)	11.8, 8.6	~15, ~10	~13, ~10
Braking Distance (60-0) (ft)	151	~170	~160

Table 2: Projected vs. Actual Test Data

Some of the emissions testing done at the EPA has also proven useful. The team learned more about the catalytic converter and what needs to be done to improve the overall emissions of the vehicle. The team learned that the catalyst requires time to soak into the catalytic converter to allow proper control over the NOx emissions and ammonia slip. The EcoEagles also observed noticeable areas of RPM ranges that would best keep NOx emissions low and help maintain peak performance for the vehicle architecture. Some of the data taken that has improved the team's overall emissions development, and improved fuel consumption is shown in Table 3.

RPM	O2 (%)	Exhaust Flow (kg/hr)	Exhaust Temp (°C)	Power (kW)	Fuel Consumption (lt/s)
1100	4.5	51.3	52.1	9.1	3
1100	4.6	51.7	53	8.6	2.95
1100	4.6	51.9	54.2	9.4	2.95
1300	6.4	57	62	10	3.1
1300	3.8	65	78.1	13	4
1300	4.5	65.8	92	13	4
1500	4.8	80	103	17	4.8
1500	3	86	116	20	5.7
1500	3.1	85	124	20	5.7
1800	5.6	80	140	17	4.7
1800	4.2	120	162	28	7.3
1800	4.3	120	190	28	7.3
1800	-	-	-	28	7.3
2250	6.9	150	215	31	8
2250	6.9	150	220	31	8
2250	6.9	150	220	31	8
2250	6.9	150	230	31	8

Table 3: Collection of Emissions Data at EPA.

RPM	Throttle (%)	CO (ppm)	CO2 (%)	NO (ppm)	NO2 (ppm)	Nox total (ppm)
1100	10	80	12	896	8.5	904.5
1100	20	70	11.81	730	13.1	743.1
1100	30	80	11.83	548	430	978
1300	20	100	10.6	34.2	730	764.2
1300	30	130	12.19	36	1000	1036
1300	40	110	12.11	358.1	648	1006.1
1500	30	110	11.47	365	590.6	955.6
1500	40	110	12.76	336	520	856
1500	50	120	12.76	463.7	410	873.7
1800	30	130	10.41	109	223	332
1800	45	110	11.99	518	280	798
1800	60	100	11.85	627.9	267.9	895.8
1800	75	100	11.8	670	283	953
2250	40	100	10.02	601.4	308.5	909.9
2250	50	100	9.96	601	312.6	913.6
2250	60	110	9.97	601	308.9	909.9
2250	70	100	9.98	606	306.2	912.2

Table 3 (Cont.): Collection of Emissions Data at EPA.

So far, all of the data has not validated the VTS. All of the tests conducted currently have shown that the vehicle needs improvement. This was expected prior to arriving at the MPG for the final competition. Once the team and vehicle were both ready at MPG, the vehicle was tested under 99% buyoff conditions. The vehicle was scheduled to go through braking, acceleration, noise quality, drive quality, on-road safety, and autocross testing events. The VTS shown previously in Table 1 has the final results of the competition. The EcoEagles were not able to fully validate the predictions made previous years. The team encountered a few problems that unfortunately caused the vehicle to not perform as planned.

The first problem that arose was the turbo on the turbo diesel. The turbo seals eventually broke and allowed air pressure to drop considerably. This forced the engine to perform at a quarter of the power available. The EcoEagles vehicle architecture highly depends on the engine for higher speed applications. Replacement of the turbo is apparent, but then the question arose of what caused this problem.

The second problem that arose was the diesel particulate filter (DPF). The DPF was clogged with particulate to the

point that no air was flowing through the exhaust. This problem was the source for the first problem the EcoEagles encountered. To fix the problem, the DPF was baked in a high temperature oven and cleaned. The next question was what caused the DPF to clog.

The third and final problem that arose and stopped the EcoEagles from performing in the competition was the control code. It was noticed that the team did not account for the regenerative capabilities of the engine to burn off the DPF. A code fix to detect when the engine needed to burn off the DPF would allow the exhaust to stay unclogged and prevent the turbo from having too much back pressure and causing a failure. After realizing these problems, the EcoEagles have learned valuable lessons and plans to incorporate solutions into future DFMEA and troubleshooting scenarios.

CONCLUSIONS AND FUTURE WORK

The competition is over and the vehicle is safely parked back at the Green Garage here at ERAU. The EcoEagles managed to place 11th out of 16 teams. In terms of customer appeal and static consumer acceptability, the team managed to place 3rd. The vehicle was presented well and a lot of GM engineers were impressed with the technologies the team was incorporating into the vehicle.

Dynamic consumer acceptability was not a big success, but a greater learning experience for future competitions. The team realized what occurred, how to fix the problem, and managed to maintain a safe driving environment when these failures did occur. Overall the project was a success and will allow the team to strive further in the future competition of EcoCAR 2: Plugging Into the Future.

ACKNOWLEDGEMENTS

The EcoEagles would like to thank all of the sponsors of the EcoCAR competition, mainly GM, ANL, and NI for their support and countless hours helping all the teams that needed assistance. The team would also like to thank Dr. White and Dr. Compere for advising the team wherever it was needed, and for times when the team needed extra support. Last, but not least, Charles Hua was a great help, and a wonderful mentor from GM, and the team thanks him for all of his effort.

REFERENCE

- [1] Langer, Gary. *ABC News*. 13 February 2005. 14 March 2011
- [2] Norbye, Jan P., Dunne, Jim. "GM Takes the Wraps Off its Steam Cars and a commuter Car with Hybrid Drive." *Popular Science* July 1969: 86-87. Print.
- [3] Gonder, J., T. Markel, A. Simpson, M. Thornton. "Using GPS Travel Data to Assess the Real World Driving Energy Use of Plug-In Hybrid Electric Vehicles (PHEVs)." (2007), NREL Report Number: NREL/CP-540-40858.