

BENCHMARKING, CHARACTERIZATION AND TUNING OF SHELL  
ECOMARATHON PROTOTYPE POWERTRAIN

A Thesis

presented to

the Faculty of California Polytechnic State University,

San Luis Obispo

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science in Mechanical Engineering

by

Eric J. Griess

October 2015

©2015

Eric J. Griess

ALL RIGHTS RESERVED

## COMMITTEE MEMBERSHIP

TITLE: Benchmarking, Characterization and  
Tuning of Shell EcoMarathon  
Prototype Powertrain

AUTHOR: Eric J. Griess

DATE SUBMITTED: October 2015

COMMITTEE CHAIR: Patrick Lemieux, Ph.D.  
Associate Professor of Mechanical  
Engineering

COMMITTEE MEMBER: Joseph Mello, Ph.D.  
Lecturer of Mechanical Engineering  
Faculty Advisor for Supermileage  
Team

COMMITTEE MEMBER: John Fabijanec  
Lecturer of Mechanical Engineering

## ABSTRACT

### Benchmarking, Characterization and Tuning of Shell EcoMarathon Prototype Powertrain

Eric J. Griess

With the automotive industry ever striving to push the limits of fuel efficiency, the Shell EcoMarathon offers a glimpse into this energy conserving mindset by challenging engineering students around the world to design and build ultra-efficient vehicles to compete regionally. This requires synchronization of engineering fields to ensure that the vehicle and powertrain system work in parallel to achieve similar goals.

The goal for Cal Poly – San Luis Obispo’s EcoMarathon vehicle for the 2015 competition is to analyze the unique operating mode that the powertrain undergoes during competition and improve their current package to increase fuel efficiency. In this study, fuel delivery, ignition timing and engine temperature are experimentally varied to observe trends in steady state fuel consumption. A developmental simulation is then implemented with these trends to analyze potential differences in transient and steady state tuning targets. The engine is then tuned to finalized tuning targets and performance compared with benchmark values.

## ACKNOWLEDGMENTS

I am indebted to the fantastic faculty at Cal Poly, my ever supportive family and incredible friends. I move forward from this point knowing that I would be nowhere close to where I am today without them.

Thank you *Patrick Lemieux* for perpetuating Cal Poly's invaluable 'learn by doing' approach by allowing enough freedom to learn how to define project goals and maintain focus on the scope throughout development. Your instruction in engine design has played an integral role in further developing my passion for the subject. Thanks to *Jim Gerhardt*, who was always ready to offer invaluable experience during dynamometer installation and when everything started failing. Additionally, my gratitude goes especially to *Don Williams*, whose support of the engines program at Cal Poly made this project possible.

Possibly the largest daily contributions came from my graduate colleagues. Their expertise and vast diversity of thought processes helped reveal solutions I would never have considered. A gratuitous thanks to *Michael Eichermueller* for late-night, impromptu brainstorming sessions, *Victor Villa* for unproductive laughs, and a special thank you to *Corinne Warnock* for endless moral support.

But most of all, I'd like to thank my family for motivating me to get places I've only dreamed of going.

# TABLE OF CONTENTS

	Page
LIST OF TABLES .....	ix
LIST OF FIGURES .....	x
CHAPTER	
1   INTRODUCTION .....	1
1.2   OPERATING MODE.....	2
1.3   TEAM GOALS .....	3
1.4   REQUIREMENTS.....	3
1.4.1   Team Requirements.....	4
1.4.2   Competition Requirements .....	4
2   PROJECT OVERVIEW.....	5
2.1   SCOPE.....	5
2.2   METHOD .....	5
2.3   RESOURCES .....	6
3   TEST SETUP .....	7
3.1   DYNAMOMETER .....	7
3.1.1   Mounting.....	7
3.1.2   Power Supply .....	8
3.1.3   Controller .....	10
3.1.4   Cooling Supply.....	11
3.2   ENGINE .....	13
3.3   DRIVE.....	16
3.4   ENGINE CONTROL UNIT (ECU).....	19
3.5   FUEL SYSTEM.....	21
3.6   FLOW SENSOR.....	22
3.7   SAFETY .....	27
3.8   CALIBRATION.....	29
3.8.1   Dynamometer .....	29
3.8.2   Engine Temperature .....	31
3.8.3   Manifold Temperature .....	34
3.8.4   Flow Meter.....	34

3.8.5   Test Repeatability.....	41
4   BENCHMARK TESTING.....	56
4.1   BASELINE PERFORMANCE.....	56
4.1   LAMBDA VARIATION.....	59
4.2   IGNITION VARIATION.....	61
4.3   TEMPERATURE VARIATION.....	63
4.4   TRANSIENT TEMPERATURE.....	67
4.5   ENGINE IDLE / START.....	69
5   VEHICLE SIMULATION DEVELOPMENT.....	71
5.1   DISCLAIMER.....	71
5.2   MODEL ASSUMPTIONS.....	71
5.3   MODEL STRUCTURE.....	73
5.3.1   Engine Speed Controller.....	73
5.3.2   Output Torque Calculation.....	75
5.3.3   Vehicle Dynamics Calculation.....	77
5.4   PRELIMINARY VERIFICATION.....	79
5.4.1   Convergence.....	79
5.4.2   Vehicle Dynamics.....	81
5.4.3   Clutch Characteristic.....	84
5.4.3   Trend Study.....	86
6   RESULTS, DISCUSSION.....	90
6.1   LAMBDA, IGNITION TARGETS.....	90
6.2   ENGINE TUNE.....	95
7   RECOMMENDATIONS.....	99
7.1   ENGINE TUNING.....	99
7.2   DRIVETRAIN.....	100
7.3   SIMULATION DEVELOPMENT.....	103
7.4   VEHICLE OPTIMIZATION.....	104
8   CONCLUSION.....	106
8.1   DYNAMOMETER.....	106
8.2   EXPERIMENTAL PERFORMANCE.....	107
8.3   NEXT STEPS.....	109
REFERENCES.....	110

## APPENDICES

A1. SYSTEM SPECIFICATIONS.....	112
A2. DYNAMOMETER SETUP, SAFETY .....	113
A3. MAGTROL DYNAMOMETER TUTORIAL.....	114
A4. SIMULATION RUN TUTORIAL .....	119
A5. SIMULATION EQUATIONS .....	121
Torque, Inertia, Efficiency .....	121
Aerodynamic Drag.....	125
Rolling Resistance .....	126
A5.1 BURN / COAST CONTROL .....	126
A5.2 CENTRIFUGAL CLUTCH .....	127
A5.3 MODIFIERS .....	131
Density .....	131
Lambda.....	132
Ignition .....	133
Temperature Variation .....	134
A5.4 OUTPUT CALCULATIONS .....	135
Overall Fuel Economy .....	135
Energy .....	135
Average Values.....	142
Engine Start Counter.....	143
A6. HAND CALCULATIONS.....	145
A7. FUEL SYSTEM SCHEMATIC .....	148



## LIST OF TABLES

Table	Page
Table 1. Baseline duty cycle range of engine .....	39
Table 2. Drive system iteration summary .....	42
Table 3. Summary of simulation assumptions .....	72
Table 4. Simulation-based lambda and ignition targets .....	92
Table 5. Simulated fuel economy difference with tuned engine.....	98
Table 6. Summary of example modifications and improvements.....	104

## LIST OF FIGURES

Figure	Page
Figure 1. Cal Poly Urban Concept Vehicle .....	1
Figure 2. Cal Poly Prototype Vehicle .....	2
Figure 3. Final dynamometer and engine system.....	7
Figure 4. Dynamometer mounting plate .....	8
Figure 5. Wiring Configuration for Power Supply and Torque/Speed Conditioner.....	9
Figure 6. Power Supply and Torque/Speed Conditioner Mounted .....	10
Figure 7. Connections on the DSP 6001 Controller .....	10
Figure 8. Front panel of DSP6001 controller and associated functions .....	11
Figure 9. Open cooling system.....	12
Figure 10. 20 Micron water filtration unit .....	12
Figure 11. Cooling inlet and exit on dynamometer.....	13
Figure 12. One piece aluminum engine mount .....	14
Figure 13. Existing mount and exhaust .....	14
Figure 14. Revised steel exhaust runner .....	15
Figure 15. Third engine mount.....	15
Figure 16. Boring of 32mm coupler .....	16
Figure 17. Broaching of 10mm key for 32mm coupler .....	17
Figure 18. Three different alignment requirements for coupler system .....	17
Figure 19. Supporting ratchet strap used to assist in coupler alignment .....	18
Figure 20. Radial and angular alignment using straight edge (yellow elastomer shown).....	19
Figure 21. ECU and electronics panel .....	20
Figure 22. User interface to Megasquirt 2 ECU provided by TunerStudio MS ...	21
Figure 23. Final fuel system configuration .....	22
Figure 24. Fuel volumetric flow rate based on volumetric efficiency values.....	23
Figure 25. McMillan Model 101 20-100 mL flowmeter implemented on dynamometer.....	24
Figure 26. Flow meter internal parts diagram.....	25
Figure 27. Upstream plumbing of flow meter.....	26
Figure 28. Safety cage with chain drive setup .....	27
Figure 29. Front control panel. (Left to Right): Emergency cutoff, ECU power, fuel pump, ignition power, starter, and throttle control.....	28
Figure 30. Applying fixed weight to calibration arm of dynamometer.....	29
Figure 31. Dynamometer error in in both CW (+) and CCW (-) directions .....	30
Figure 32. Engine coolant temperature sensor location.....	31
Figure 33. Coolant temperature sensor calibration setup .....	32
Figure 34. Coolant temperature sensor calibration verification .....	33
Figure 35. Manifold temperature sensor calibration validation .....	34
Figure 36. Fuel injector test setup .....	35
Figure 37. Oscilloscope output of flowmeter during test .....	36

Figure 38. Factory supplied flow meter calibration compared to corrected calibration .....	37
Figure 39. Flow meter error with increasing duty cycle .....	38
Figure 40. Fuel flow measurement comparison .....	40
Figure 41. Repeatability variation throughout drive system iterations .....	43
Figure 42. Engine speed variation between chain and direct drive .....	44
Figure 43. First chain drive iteration .....	45
Figure 44. Welded centrifugal clutch .....	46
Figure 45. Crankshaft failure plane on output side .....	47
Figure 46. Torsion spring tensioner setup .....	47
Figure 47. Torque, power variation with no temperature control .....	49
Figure 48. Air mover with butterfly valve and ducting for temperature control .....	50
Figure 49. Torque output comparison with temperature control @ 4250RPM ....	51
Figure 50. Double-idler iteration of chain tensioner .....	52
Figure 51. Direct drive setup (Top View).....	54
Figure 52. Melted elastomer thrown radially .....	55
Figure 53. Stiffer elastomer installed with lubrication .....	55
Figure 54. Baseline performance of engine .....	57
Figure 55. Lambda and ignition values in benchmark tune .....	58
Figure 56. Torque and BSFC variation with lambda (4000 RPM, $\Phi = 0^\circ$ ) .....	60
Figure 57. Torque variation with ignition timing (4000 RPM, $\lambda=1.0$ ) .....	62
Figure 58. Torque, BSFC variation with engine temperature (4000 RPM, $\lambda=1.0$ , $\Phi=0^\circ$ ) .....	64
Figure 59. Engine temperature rise rate at various RPM.....	68
Figure 60. Temperature fall rate through natural convection .....	69
Figure 61. Fuel consumption between idling and starting conditions .....	70
Figure 62. Simplified Simulink model structure.....	73
Figure 63. Engine speed controller in Simulink.....	73
Figure 64. Output torque calculation in Simulink .....	75
Figure 65. Centrifugal clutch model.....	76
Figure 66. Vehicle dynamics calculation in Simulink.....	77
Figure 67. Final Simulink portion of vehicle model .....	79
Figure 68. Simulation convergence with maximum step size .....	80
Figure 69. Run time variation with different solvers .....	81
Figure 70. Vehicle dynamics during launch conditions .....	82
Figure 71. Large scale vehicle dynamics.....	83
Figure 72. Intermediate burn and coast behavior .....	84
Figure 73. Engine and output speed behavior during launch.....	85
Figure 74. Torque characteristic during vehicle launch .....	85
Figure 75. Vehicle mass trend study .....	86
Figure 76. Rolling resistance trend study .....	88
Figure 77. Gear ratio trend study.....	89
Figure 78. Normalized fuel economy with lambda and ignition variation .....	91
Figure 79. BSFC and torque loss with lambda variation .....	93
Figure 80. Loss comparison with simulated fuel economy trend .....	94

Figure 81. Torque, Power, BSFC comparison with baseline and simulation .....	96
Figure 82. BSFC improvement with target lambda and ignition values .....	97
Figure 83. Drive system comparison .....	101
Figure 84. Output side of crankshaft .....	102
Figure 85. Tractive force equation implementation in Simulink .....	124
Figure 86. Aerodynamic drag force implementation in Simulink .....	125
Figure 87. Rolling resistance force implementation in Simulink.....	126
Figure 88. Engine switching function in Simulink.....	127
Figure 89. Centrifugal clutch used in prototype vehicle.....	128
Figure 90. Clutch model characteristics .....	130
Figure 91. Engine speed control implementation in Simulink.....	131
Figure 92. Simulink calculation for torque multiplier for lambda variation.....	133
Figure 93. Temperature variation implementation in Simulink .....	134
Figure 94. Obtaining variables for fuel economy calculation in Simulink.....	135
Figure 95. Calculation of net engine energy .....	137
Figure 96. Energy losses from engine calculated in Simulink.....	138
Figure 97. Simulink implementation of energy lost from torque factors .....	139
Figure 98. Clutch slip energy calculation in Simulink model.....	140
Figure 99. Inertia energy calculation in Simulink .....	141
Figure 100. Simulink implementation of aerodynamic energy calculation .....	142
Figure 101. Rolling resistance energy calculation in Simulink .....	142
Figure 102. Example of average value calculation .....	143
Figure 103. Engine start counter schematic .....	143

## 1 | INTRODUCTION

The primary objective of the Shell EcoMarathon event is to achieve the best fuel efficiency possible, quantified with miles per gallon (MPG). The regulations that exist on vehicle design ensure student safety and challenge design, while allowing freedom for engineering innovation. Vehicles are split into categories based on class and fuel type. The Urban Concept class is designed to reflect real world fuel efficiency, in which vehicles look similar to passenger cars and the fuel efficiency measurement involves stop and go driving.



Figure 1. Cal Poly Urban Concept Vehicle

The vehicles in prototype class, however, are allowed more freedom in vehicle shape to achieve more impressive fuel economy numbers. Instead of stop and go operation, a large, 6-mile course is laid out and the operating strategy between the start and finish is governed only by a required average speed of 15 mph and a maximum time limit.



Figure 2. Cal Poly Prototype Vehicle

Since Cal Poly primarily develops vehicles for the prototype class, the powertrain package developed in this study is for the prototype vehicle in an effort to maximize relevance.

## 1.2 | OPERATING MODE

The engineering challenges for the powertrain are unique to this class due to the specific operating mode utilized to achieve extreme fuel efficiency targets. Conventionally, teams use a ‘burn and coast’ method throughout the course. This involves starting the engine immediately at the beginning of the course and allowing it to propel the vehicle to a maximum velocity, then cutting engine power and coasting until minimum velocity. The engine is then restarted by the driver, and this process is repeated until the end of the course.

The main design challenges teams experience with the powertrain under these conditions are largely due to the vast number of variables involved during transient engine operation, making specific design goals difficult to establish.

### 1.3 | TEAM GOALS

Cal Poly's EcoMarathon team lead, Sean Michel, explicitly defined the goals for the 2015 prototype vehicle in order of importance:

1. Ease of Operation/Engine Tuning
2. Steering Geometry
3. Vehicle Testing/Driver Training
4. Drivetrain system design
5. Carbon fiber technology
6. Wheels, Hubs and tire testing

As previously mentioned, a full engine tune presents itself as a large potential improvement because the team did not have access to an engine dynamometer during vehicle development in 2013. The existing engine lead, Dorian Capps, was forced to resort to driver feedback for the duration of testing. The oxygen sensor was not operational until competition, by which time there was not an opportunity to modify the tune extensively. Developing the engine with little to no feedback from the engine suggests that the existing tune is far from optimal.

### 1.4 | REQUIREMENTS

In order to integrate this study with the team's vehicle development, there are both team and organization requirements that are shared. These cover a large range of design criteria from fuel selection to project scheduling.

#### 1.4.1 | Team Requirements

Within the scope of this study, the primary requirement for the team is to iterate engine design from the previous year to increase fuel efficiency to achieve a better result in the competition. Additional requirements include:

- Similar packaging dimensions
- Minimizing overall weight
- Project finished by January, 2015 to install and test.

Beyond these, the team is flexible with engine choice, subsystem design, electronics, and auxiliary systems. However, there are still competition requirements that limit freedom of design.

#### 1.4.2 | Competition Requirements

Although the team must comply with chassis, electrical, and fuel requirements, the only limitation for the engine is a 4-stroke combustion cycle. Maximizing efficiency has implicitly driven trends in engine design, such as use of small engine displacements (35-50cc) and retaining as much engine heat as possible.



## 2 | PROJECT OVERVIEW

This section fully defines the scope of the project, methodology, and resources available at the time of this study.

### 2.1 | SCOPE

The scope of this project is to successfully benchmark the engine, develop and utilize a vehicle simulation to define engine tuning targets, and perform the tune. Only air-fuel ratio ( $\lambda$ ), ignition, and engine temperature will be varied, while torque, brake specific fuel consumption (BSFC), and transient temperature behavior will be studied and characterized.

At the conclusion of this project, an engine with tuned fuel delivery and ignition timing tables is delivered, accompanied by a vehicle simulation that provides a foundation for future development. Although the engine is benchmarked and tuned with the existing Megasquirt2 unit, the team is responsible for implementing the final tune onto their new Engine Control Unit (ECU) unit and associated hardware that is currently under development by the team.

### 2.2 | METHOD

The development path for this project is as follows:

1. Dynamometer system fabrication
2. Benchmark testing and trend analysis
3. Vehicle simulation development
4. Comparison of tuning targets
5. Engine tune results and summary

To increase efficiency, the simulation was developed in conjunction with dynamometer fabrication and calibration. The list above only represents the linear path of these benchmarks, and is followed as an outline for this report.

### 2.3 | RESOURCES

The main resources available for this study include Don Williams and the Mechanical Engineering Department for funding, Professor Patrick Lemieux's project direction, Jim Gerhardt for technical support, Dorian Capps for existing powertrain knowledge, Sean Michel for competition information, and Chad Bickel for initial simulation development support.

Existing hardware included a dynamometer, table, control unit, and access to the propulsion and internal combustion engines lab. All fabrication and machining was done in the Cal Poly Mustang 60 machine shop.

### 3 | TEST SETUP

Before benchmark testing could begin, dynamometer components were implemented to support existing and future small engines. This section summarizes the installation and calibration processes involved to ensure supporting systems allowed for safe, accurate and repeatable testing conditions.

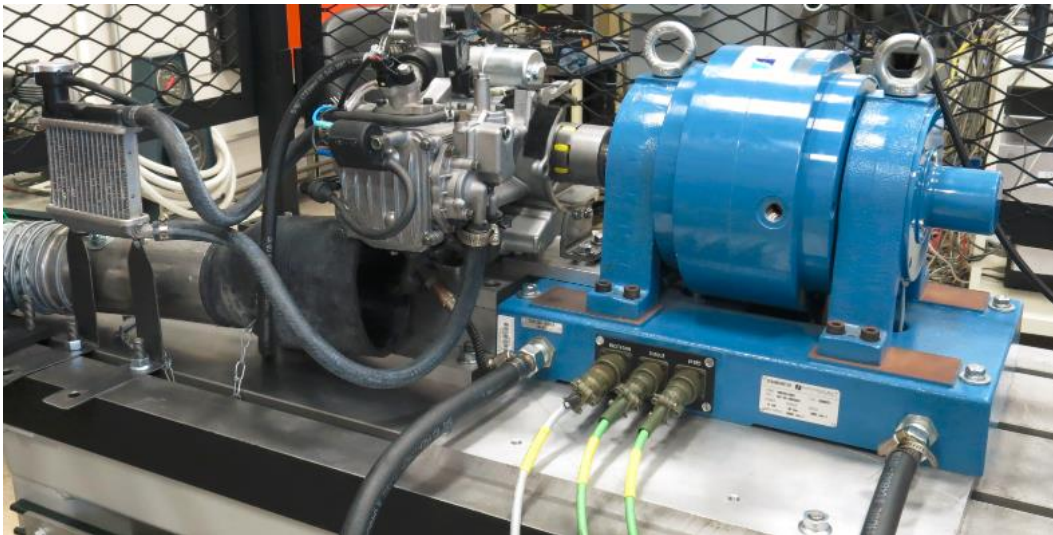


Figure 3. Final dynamometer and engine system

#### 3.1 | DYNAMOMETER

For the existing Magtrol WB115 water brake dynamometer and controller, mounting, electronics, and water supply were necessary for operation.

##### 3.1.1 | Mounting

The dynamometer configuration was dictated by the direct drive system after iterations of the chain system proved unreliable (discussed in later sections). As pictured in the figure below, both engine and dynamometer were mounted for their drive axes to remain collinear.

The Magtrol dynamometer mounts were manufactured specifically to metric standards, but the T-slot table was standard. Because of this, an adapter plate was necessary and is shown (in chain drive configuration) in the figure below.

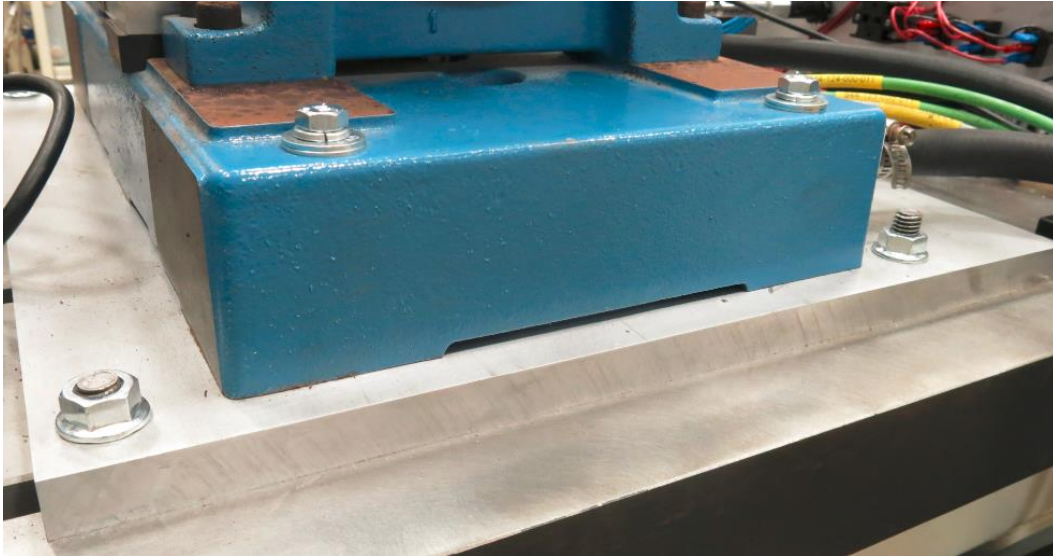


Figure 4. Dynamometer mounting plate

Slightly undersized, standard bolts were used through the 10mm holes to allow for clearance and slight angle adjustment. The dimensions are available through the dynamometer manual listed in the reference section [8].

### 3.1.2 | Power Supply

A power supply and torque/speed conditioner were supplied with the dynamometer and configured as follows:

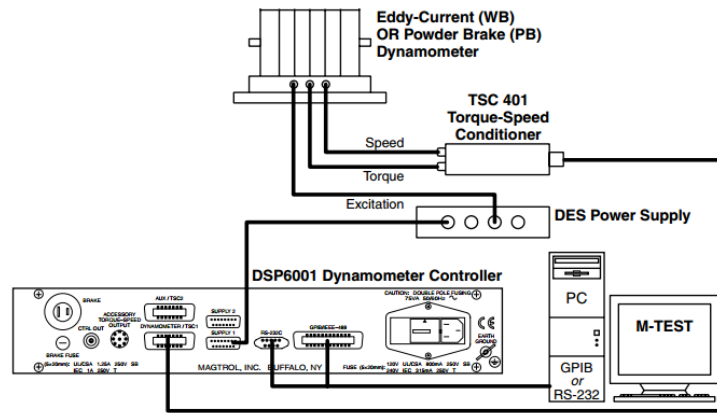


Figure 5. Wiring Configuration for Power Supply and Torque/Speed Conditioner

The power supply utilizes 110V outlets to supply the necessary current to the coils of the brake of the dynamometer, and is specifically designed to provide the best response time for transient load application. This component is controlled by the dynamometer controller discussed in the next section.

The TSC 401 torque-speed conditioner acts as an instrumentation amplifier, and both amplifies and filters the torque signal for more usable data. It also controls power to the speed sensor in the dynamometer.



Figure 6. Power Supply and Torque/Speed Conditioner Mounted



Figure 7. Connections on the DSP 6001 Controller

### 3.1.3 | Controller

The Magtrol DSP 6001 controller acts as a user control interface, I/O interface, and data acquisition system in conjunction with Magtrol’s M-TEST 7.0 software. PID settings, torque and speed control, and additional features are controllable by the user via front panel or software interface.

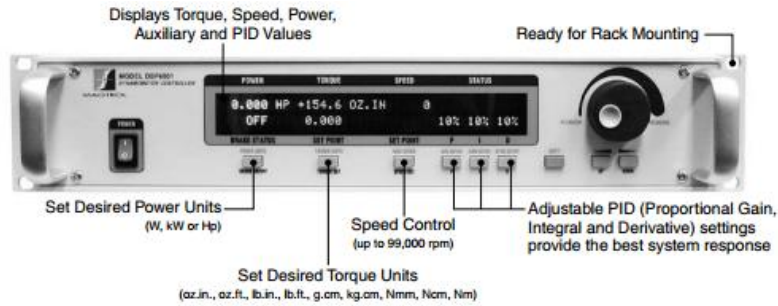


Figure 8. Front panel of DSP6001 controller and associated functions

### 3.1.4 | Cooling Supply

An internal cooling system is integrated to reject excess heat created by torque absorption in the coils. Even through the power produced by this engine is well within the limits of the WB115, the dynamometer support systems were designed for future compatibility with other test engines that produce significantly more power.

For the cooling system, an open system was chosen to integrate with existing facilities. A 100 micron maximum particle size is recommended to reduce corrosion, but a 20 micron water filter with an adequate flow rating was easily available and was therefore installed on the water inlet.

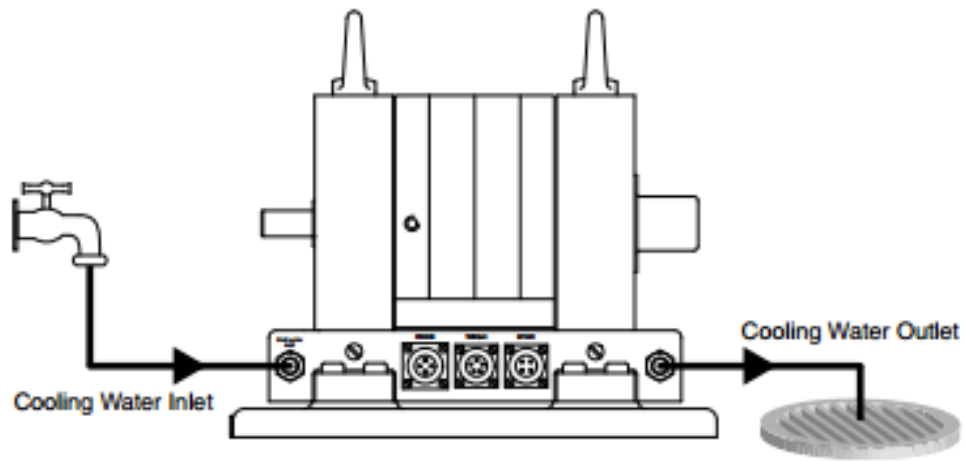


Figure 9. Open cooling system



Figure 10. 20 Micron water filtration unit



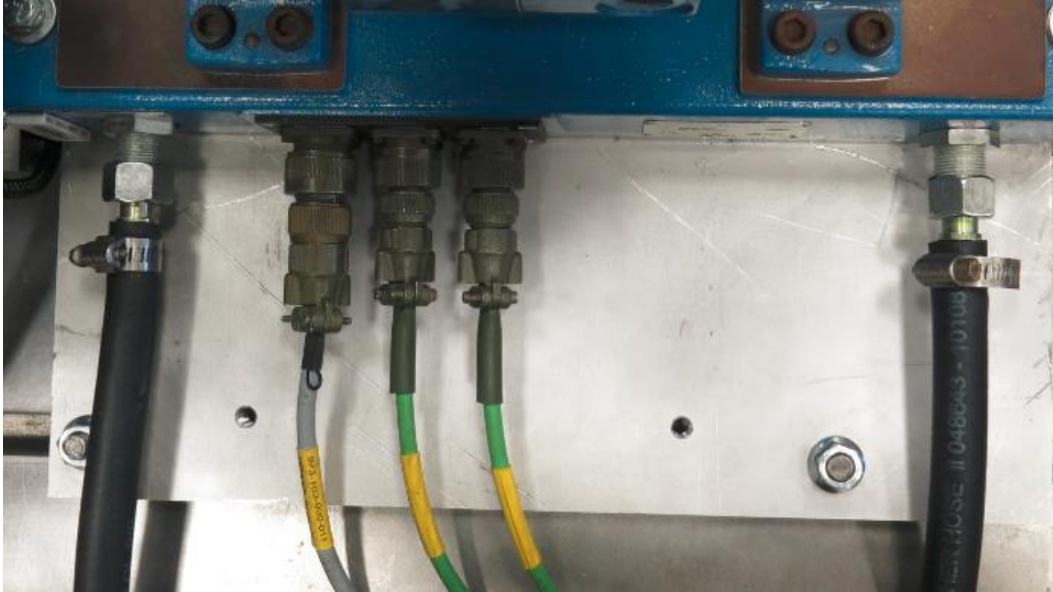


Figure 11. Cooling inlet and exit on dynamometer

A garden hose input was used due to ease of implementation with existing water supply in lab, and heater hose was used for the inlet and outlet to protect against potential high temperature conditions.

### 3.2 | ENGINE

Before mounting the engine, modifications were required to ensure reliability. Previously, the team used a one-piece aluminum mount that connected the exhaust directly to the rear engine mount. Although this setup may be adequate for competition conditions where the engine is run for short periods of time, it was not likely that aluminum could withstand steady-state operating temperatures for an extended period and the expansion of aluminum could introduce asymmetrical stress loading of the engine case, cylinder and head.



Figure 12. One piece aluminum engine mount



Figure 13. Existing mount and exhaust

To address higher operating temperatures and durations, the exhaust portion of the mount was removed and replaced with a steel tube of equivalent length and diameter. Though this may slightly affect the benchmark tune, it was a necessity to prevent potential damage.



Figure 14. Revised steel exhaust runner

With the exhaust mount removed, an additional mount was necessary to stabilize the engine for testing. The third engine mount locked all degrees of freedom for the engine, and was located near the output shaft to minimize local vibration to decrease chance of dynamic coupling misalignment.

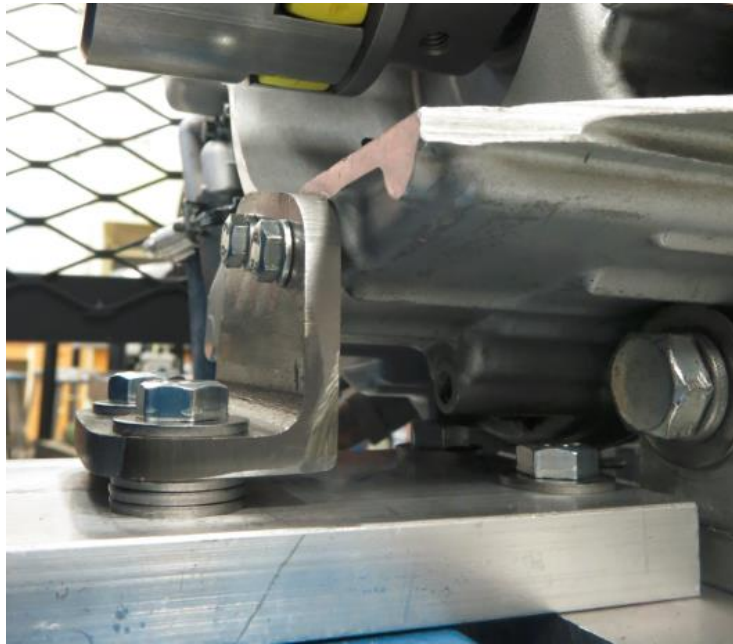


Figure 15. Third engine mount

### 3.3 | DRIVE

The requirement the drive system design was to maximize repeatability between tests while enduring test conditions. However, this proved one of the more difficult aspects of the project. After many iterations with a chain setup, a direct drive system was implemented for final testing. Each iteration is discussed in the calibration section.

Minimum requirements for the coupling system include 8000 RPM capability and the ability to withstand up to 10 ft·lb of torque (to protect against instantaneous loading conditions). Lovejoy curved jaw couplings with their red ‘spider’ elastomer damper were selected. Due to the large 32mm shaft on the dynamometer, a larger size 28 coupler was used. The lack of inventory necessitated machining of the bore and key of the 32mm coupler. Jim Gerhardt led this process with excellent results, allowing for a much smoother alignment process.



Figure 16. Boring of 32mm coupler



Figure 17. Broaching of 10mm key for 32mm coupler

After installing the couplers on the dynamometer engine, the two couplers were aligned within manufacturer's specifications, outlined in Lovejoy's installation manual [11].

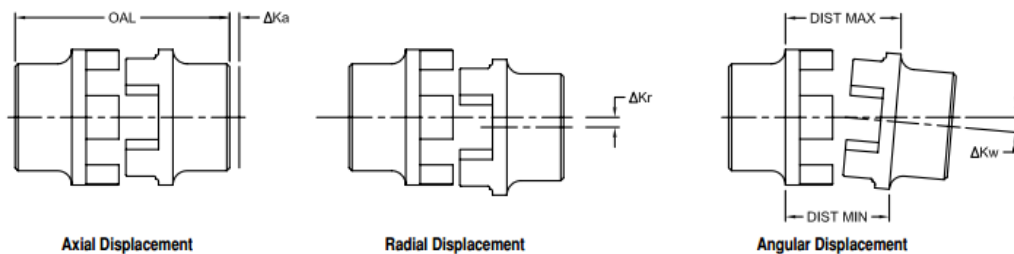


Figure 18. Three different alignment requirements for coupler system

The maximum axial, radial, and angular tolerances for this coupler are 0.01 in, 0.01 in, and 1 degree, respectively. Since the engine mount is designed for many degrees of freedom, a ratchet strap was used to help initially align the engine while engine mounting was finalized.



Figure 19. Supporting ratchet strap used to assist in coupler alignment

After the engine mounts were tightened, the base engine mount was left loose to allow final adjustment. Axial, radial and angular tolerances were measured with a straight edge and feeler gauges.

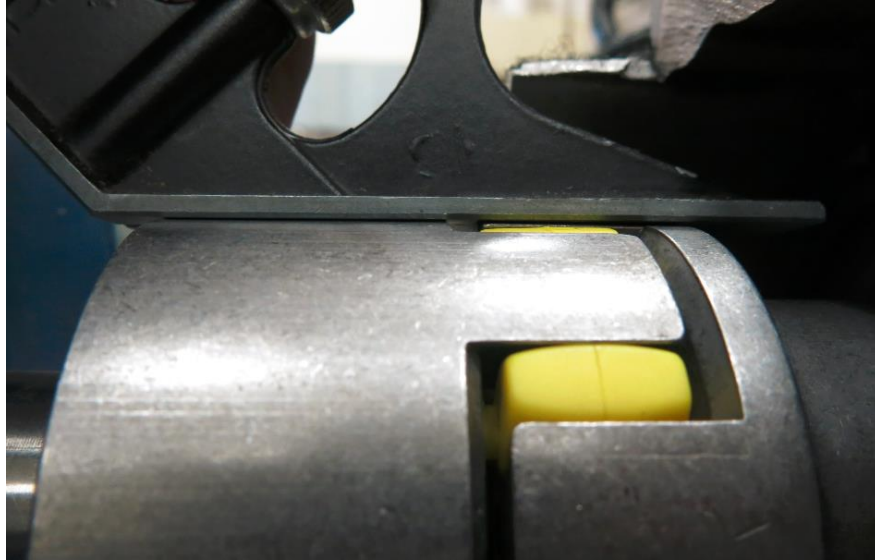


Figure 20. Radial and angular alignment using straight edge (yellow elastomer shown)

### 3.4 | ENGINE CONTROL UNIT (ECU)

The Megasquirt 2 ECU unit was previously utilized by the team to manage engine functions. In order to accurately benchmark the engine with existing fuel and ignition tables, the same ECU unit was used for testing. The electrical issue with the crank sensor caused a fuel cut condition at 5000 RPM, preventing the engine from reaching the factory 7500 RPM limit. Considering that the existing clutch engaged around 3000RPM, this essentially limited the operating range of the engine to 3000-5000 RPM, or 25% of the factory engine speed range. This issue is discussed further in a later section.

The ECU and associated relays and fuses were mounted on a separate panel and wired according to the schematic available in the Megasquirt installation guide

[18]. Alongside the ECU, a relay board manufactured by Megasquirt was utilized to simplify wiring and troubleshooting.

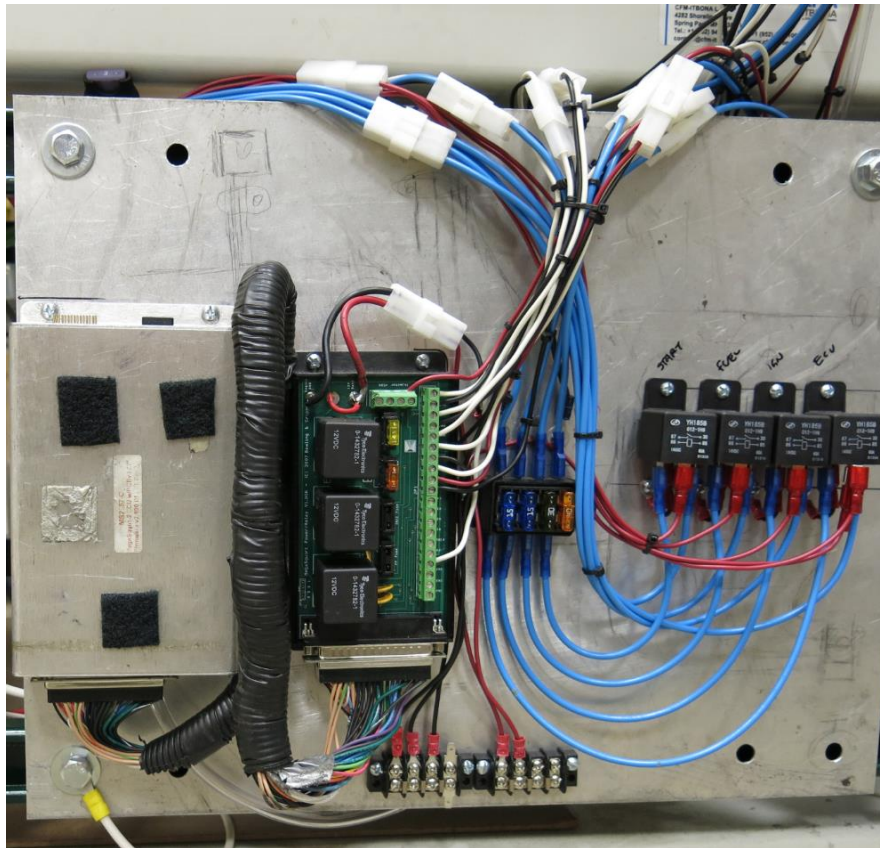


Figure 21. ECU and electronics panel

The ECU integrates a direct pressure reference, shown at the bottom left. This reference was obtained from the intake manifold after the throttle plate, and was also spliced to the fuel pressure regulator. Manifold pressure obtained from this reference was used for all load calculations during tuning.

Communication with the ECU was accomplished through a serial to USB cable, and the program TunerStudio MS provided the user interface for



manipulating ECU values. The figure below shows the basic interface used during testing.

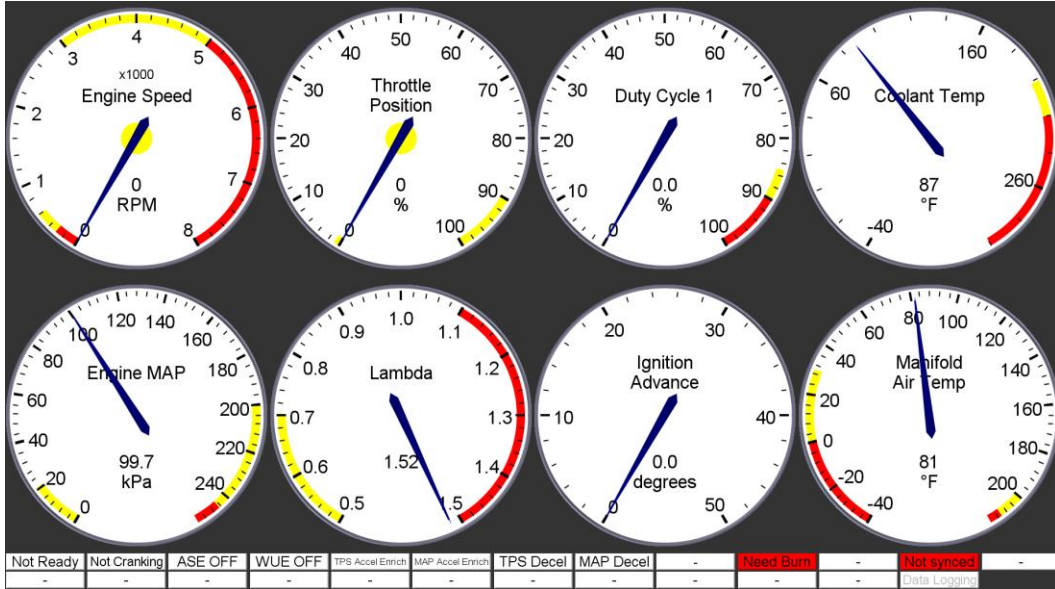


Figure 22. User interface to Megasquirt 2 ECU provided by TunerStudio MS

### 3.5 | FUEL SYSTEM

EcoMarathon competition vehicles do not use a fuel pump due to increased weight, complexity and safety concerns. Instead, a fuel line is pressurized with a bike pump via Schrader valve, and the fuel rail pressure is maintained with a fuel pressure regulator. Replicating this condition was detrimental to extended test conditions, and should carry little to no effect on results. Instead, a universal fuel pump was used in conjunction with pre-and post-pump filters, a fuel pressure regulator, check valve and flow meter.



Figure 23. Final fuel system configuration

The plumbing for the fuel system was modified several times during flow meter calibration tuning and is discussed in a later section. The plumbing schematic is available in Appendix A7.

### 3.6 | FLOW SENSOR

In order to size the flow meter, the existing volumetric efficiency vs. engine speed used in the Megasquirt ECU (maximum load / atmospheric pressure) was used and the following equation calculated to find fuel flow rate in ml/min [A2]:

$$\dot{V}_f = 0.3 \left( \frac{\rho_{air}}{\rho_{fuel}} \right) \frac{VE * RPM * V_d}{\lambda * AFR_{stoich}} \quad (1)$$

$\dot{V}_f$  = Fuel Volumetric Flow Rate (m<sup>3</sup>/s)

VE = Volumetric Efficiency (%)

RPM = Engine Speed (RPM)

$V_d$  = Engine Volume ( $m^3$ )

$\lambda$  = Lambda

$AFR_{stoich}$  = Stoichiometric Air-Fuel Ratio

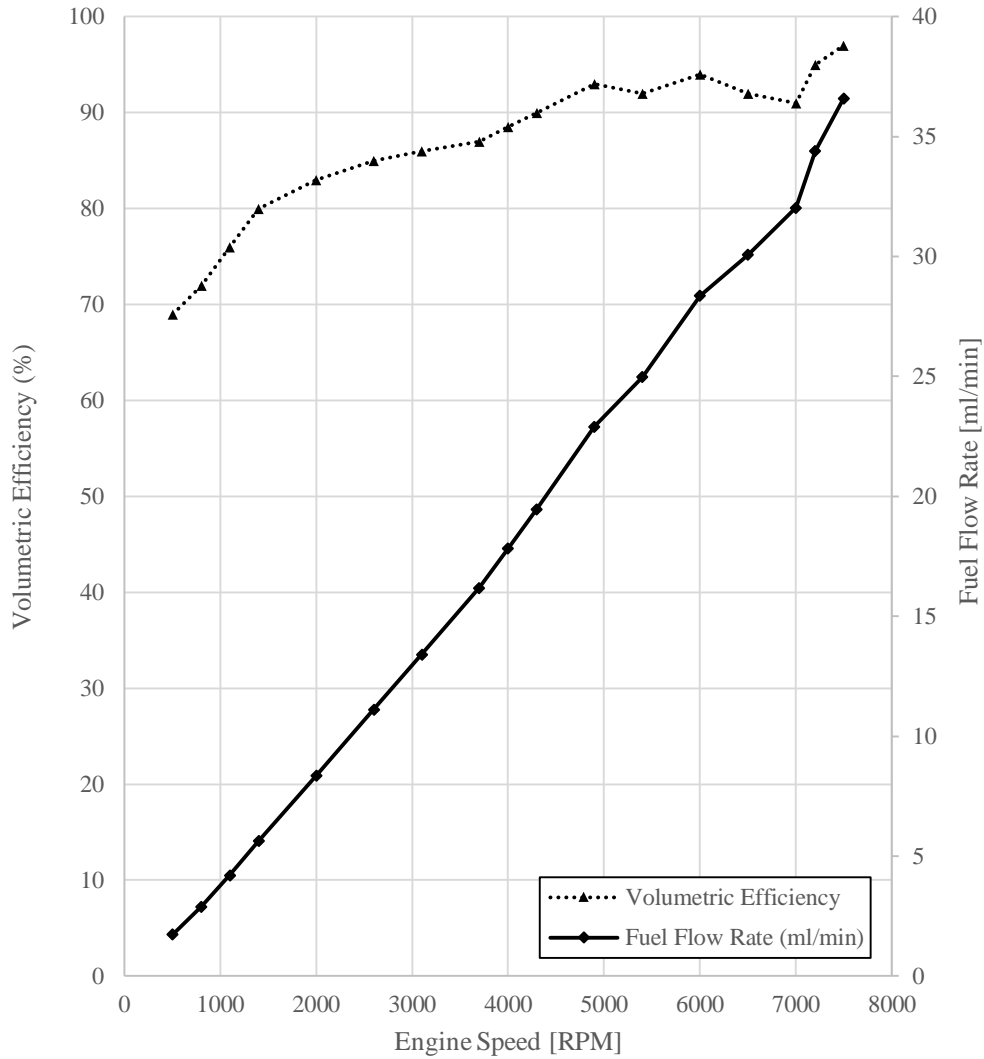


Figure 24. Fuel volumetric flow rate based on volumetric efficiency values

Under full load conditions, the fuel flow rate ranges from 8.4 ml/min to 36.6 ml/min. Additionally, an idle condition at 2000 RPM results in 3.6 ml/min of fuel consumption. In this miniscule range of fluid flow, flow meters become extremely

expensive due to the tolerances required for the sensitivity. Since those flow meters were well outside of the project budget, a 13-100 mL McMillan Model 101 was chosen and further tested to study accuracy below its flow range.



Figure 25. McMillan Model 101 20-100 mL flowmeter implemented on dynamometer

Although the 13-100 mL range is prohibitive at speeds below 3500 RPM and idle, it was possible to calibrate the sensor below rated flow ranges to a certain extent [4]. The linearity of data in the flow range for this application is studied in the calibration section.

The Model 101 utilizes a small Pelton wheel turbine mounted on sapphire bearings and shaft. This turbine directs flow into a nozzle that allows the turbine speed to remain proportional to the volumetric flow rate. The turbine itself has alternating black and white colors on the wheel, allowing an infrared sensor to detect these pulses to calculate wheel speed [5].

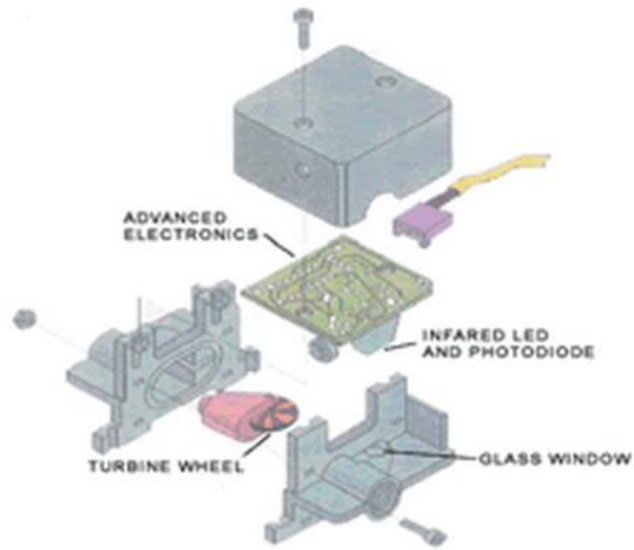


Figure 26. Flow meter internal parts diagram

Although the manual does not specify plumbing requirements, general requirements for turbine flow meters are to plumb with straight pipe for: [6]

- 10-15 diameters upstream minimum
- 5 diameters downstream minimum
- 20 pipe diameters for 90 deg. Elbow, tee or filter
- 25 pipe diameters for partially open valve
- 50 pipe diameters for elbows in multiple planes, spiraling flow



Figure 27. Upstream plumbing of flow meter

Eight inches of hard 3/16" brake line was used upstream to ensure flow straightness, allowing an L/D ratio of 43 – well over the recommended 20. Downstream, 12 inches of straight 3/8" ID rubber fuel line was used for an L/D ratio of 32.

A National Instruments NI9219 4-channel analog input module was used to interface with the flowmeters 0-5V output. Using the National Instruments Measurement & Automation Explorer (NI MAX), the flow meter was interfaced with M-TEST order to acquire instantaneous fuel flow rate data in addition to speed, torque and power.

However, the flow meter returned unreliable flow rates at lower engine speeds, so the ECU output signal was used to calculate fuel flow instead. This is further discussed in the following section on sensor calibration.

### 3.7 | SAFETY

With the dynamometer operating at high rotational speeds in an open lab environment, a safety cage was fabricated to protect users in event of component failure. The cage was mounted directly to the table, and features expanded steel doors and covers for visibility and accessibility. Additionally, existing exhaust evacuation and ducting systems were utilized to prevent buildup of fumes in the closed lab.

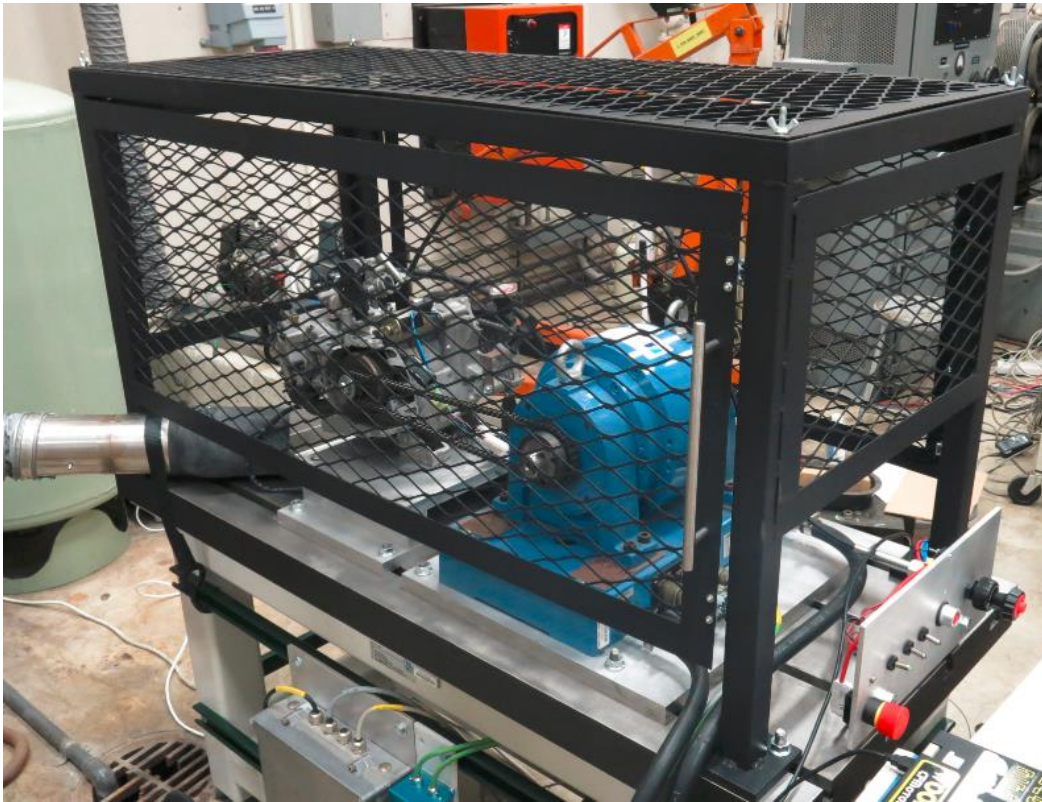


Figure 28. Safety cage with chain drive setup

The electrical system for all components excluding the flow meter passes through a master cut-off switch in case of an emergency. This push-button switch was mounted within close proximity of the user. ECU, fuel and ignition switches are all LED lit to ensure correct operation, and the starter switch is momentary to prevent a starter overrun condition.

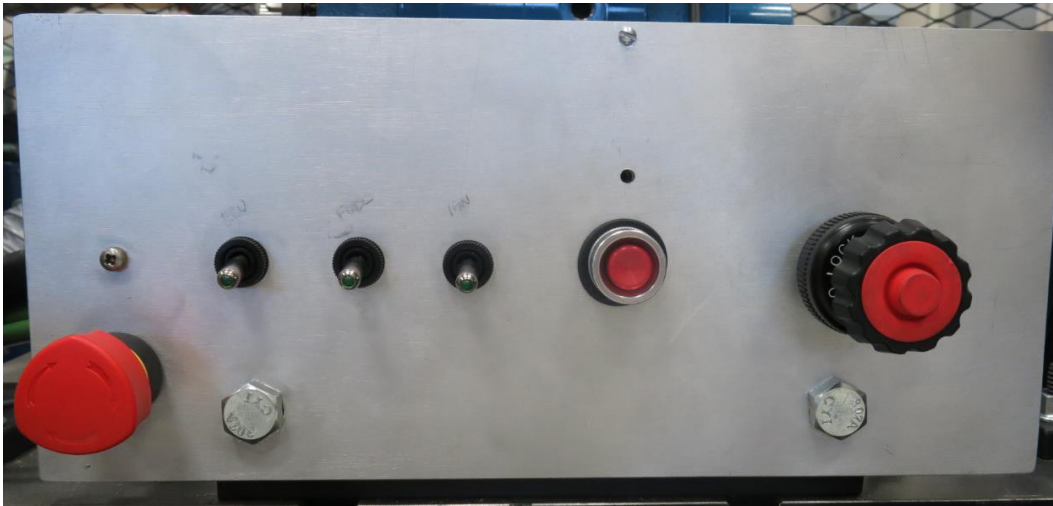


Figure 29. Front control panel. (Left to Right): Emergency cutoff, ECU power, fuel pump, ignition power, starter, and throttle control

All engine systems were powered by an automotive battery and interfaced through a quick disconnect connection to allow convenience of recharging batteries. This also allows the user to quickly remove power from the electronics to return it to its lowest energy state. The battery is constantly charged during testing to ensure constant voltage levels. A direct power supply wasn't utilized due to the high current draw from the starter motor (~40A).



### 3.8 | CALIBRATION

In order to ensure the accuracy of data and obtain uncertainty values, full calibration was performed for the dynamometer, flow meter, engine temperature sensor, and manifold temperature sensors. Additionally, data repeatability was studied to ensure reliable results throughout benchmark testing. This section summarizes the calibration method, tools, and results obtained for each.

#### 3.8.1 | Dynamometer

Calibrating the dynamometer involved applying a known torque through fixed weights and distances. After applying a known weight, the gain on the torque/speed transducer was then adjusted in order to match the displayed torque with actual torque.

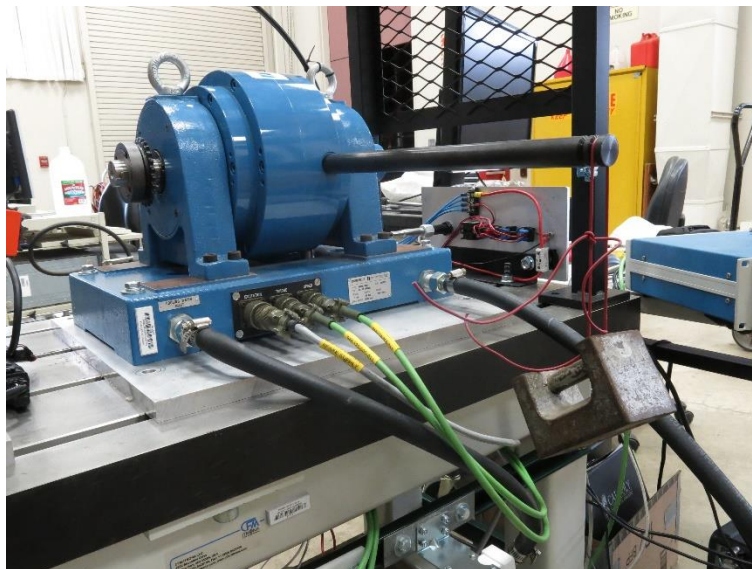


Figure 30. Applying fixed weight to calibration arm of dynamometer

The calibration weights were first verified on a scale to  $\pm 0.005$  lb accuracy. Magtrol indicated a 10 kg weight applied on the calibration arm would provide the rated

torque of 50 Nm [8]. From there, the calibration arm length was determined. 5, 10, and 15 lb. weights were used to measure torque in clockwise and counterclockwise directions, and the gain adjusted accordingly to minimize error.

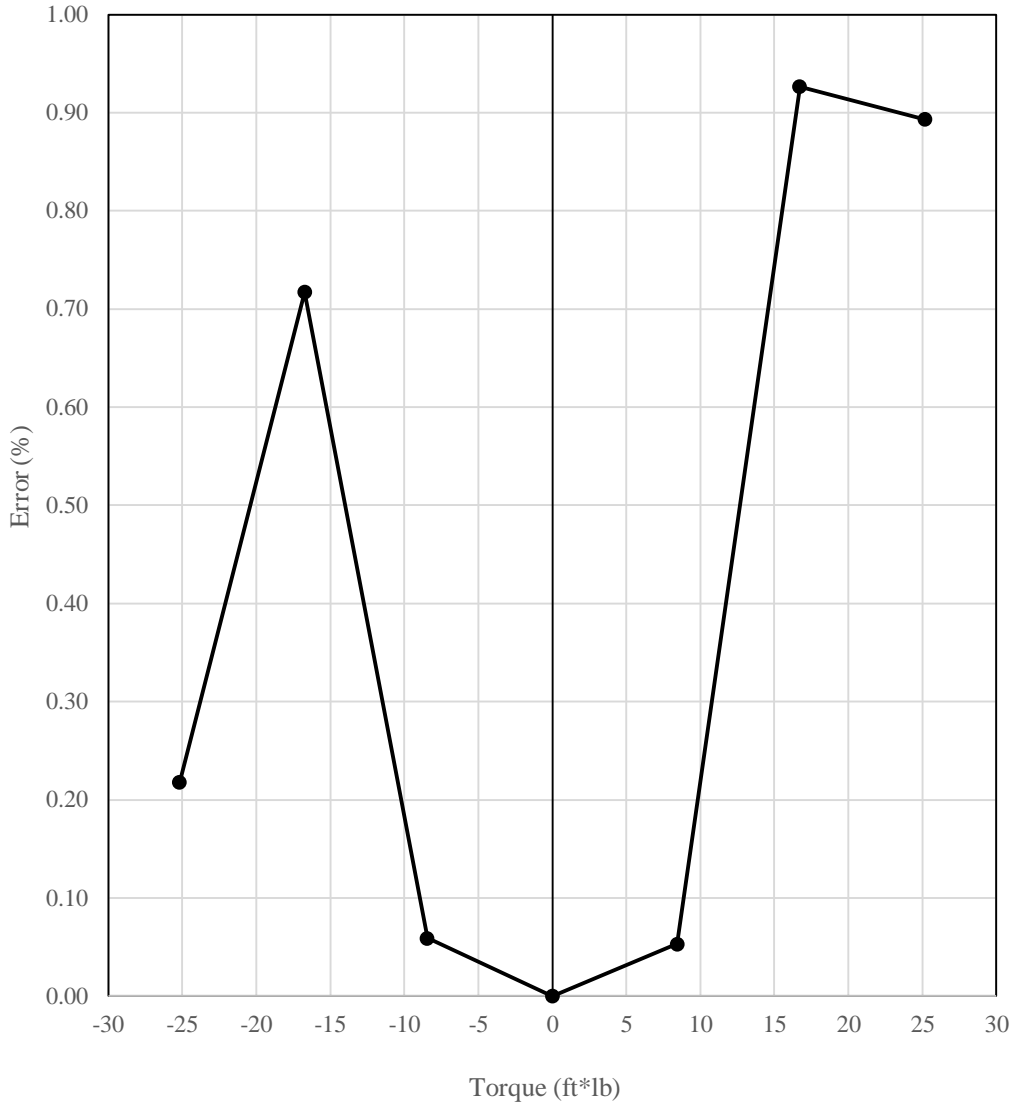


Figure 31. Dynamometer error in in both CW (+) and CCW (-) directions

The figure above indicates the dynamometer maintains an error under 1% below 25.2 ft\*lb in both directions. Since this particular engine operates within 10 ft\*lb,

the torque reading during testing of this engine is accurate within 0.1% variation from actual torque value.

### 3.8.2 | Engine Temperature

Engine temperature was measured through coolant temperature at the water pump inlet, as shown in the following figure.

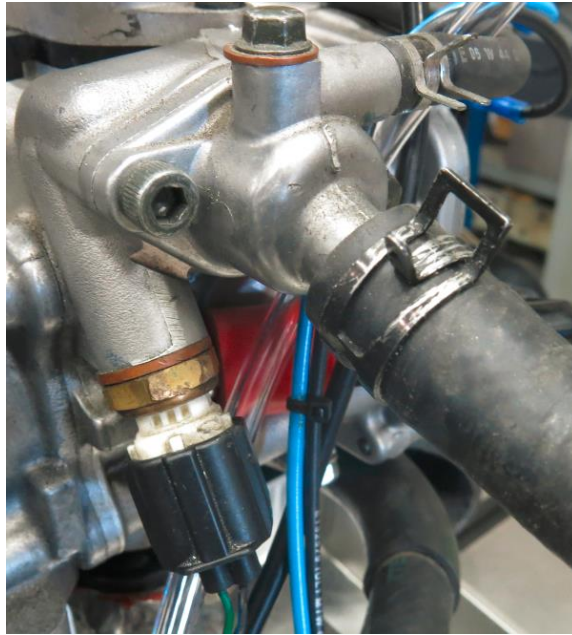


Figure 32. Engine coolant temperature sensor location

Tunerstudio creates a calibration curve for the sensor based off of 3 input points relating temperature corresponding resistance using the Steinhart-Hart equation. This equation is generally used to model the resistance of a semi-conductor at different temperatures [7].

$$\frac{1}{T} = A + B \ln(R) + C(\ln(R))^3 \quad (2)$$

T = Temperature (K)

R = Corresponding Resistance  
(Ohms)

A,B,C = Stein-Hart Coefficients

The engine temperature sensor was calibrated by immersing it in heated water while resistances and temperatures were measured as the temperature of the water decreased from near boiling to room temperature.

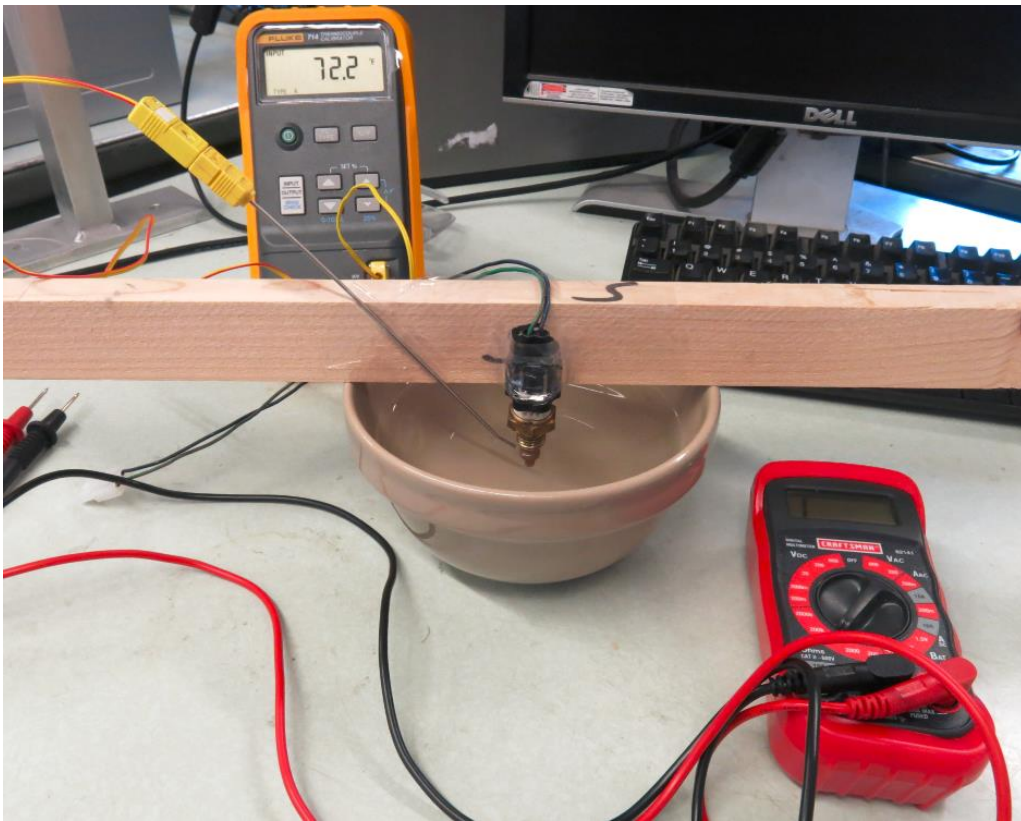


Figure 33. Coolant temperature sensor calibration setup

Then, the corresponding temperature and resistance values were recorded, and compared to the Steinhart-Hart equation.

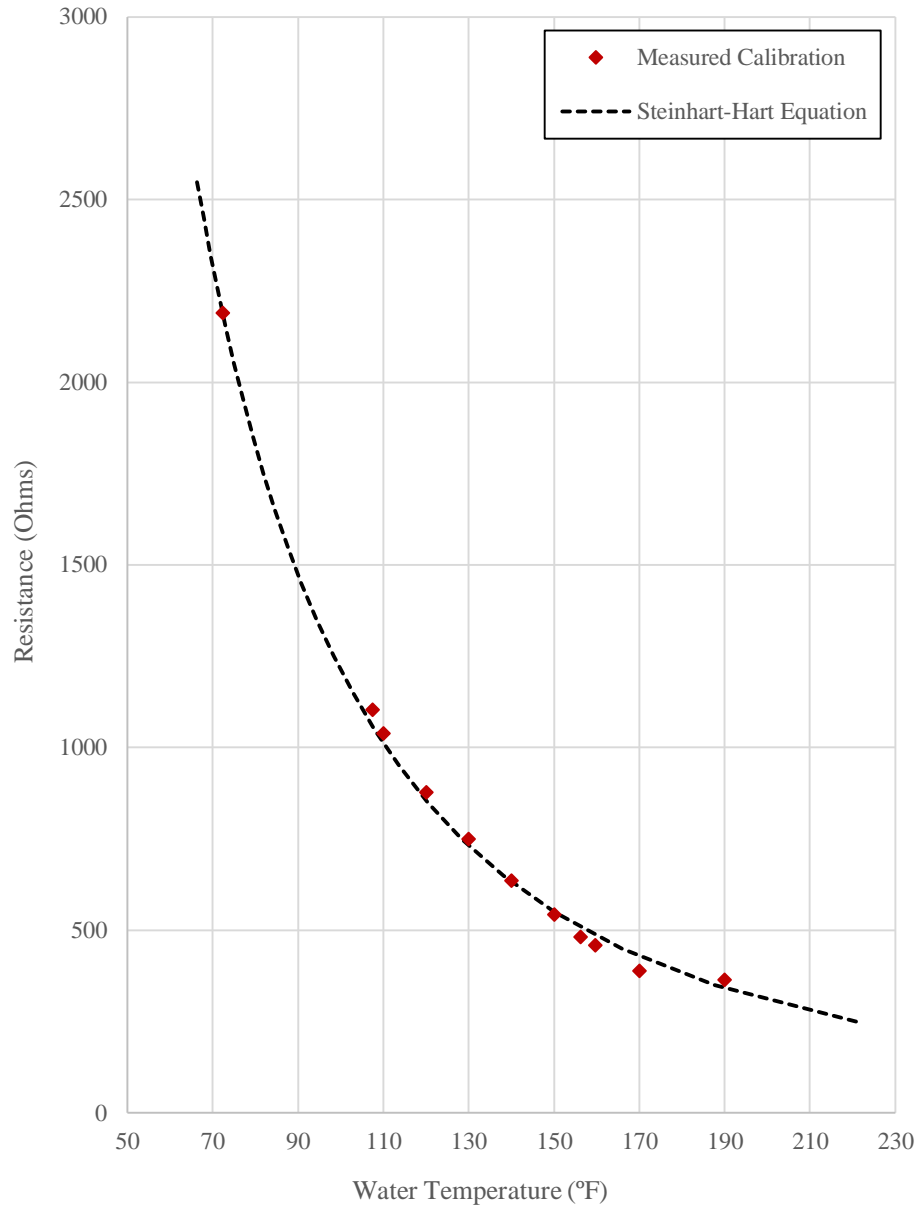


Figure 34. Coolant temperature sensor calibration verification

The figure above shows that the Steinhart-Hart equation used to generate calibration tables based on 3 data points provides a valid calibration curve that follows measured data points.

### 3.8.3 | Manifold Temperature

The manifold temperature sensor calibration table is also calculated with the Steinhart-Hart equation. The following figure compares data points taken and the calibration curve created by Megasquirt, showing again that the 3-point curve generator in the software is valid.

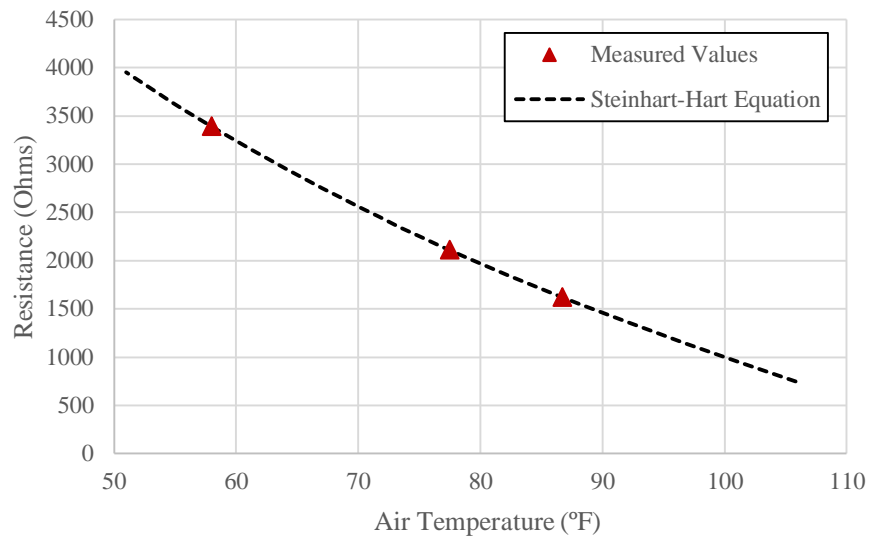


Figure 35. Manifold temperature sensor calibration validation

### 3.8.4 | Flow Meter

Flow meter calibration was performed with the intent of meeting a recommended accuracy of below 1%, specified in SAE J1349 [9]. Although the flow meter was delivered with a calibration curve, it was performed with deionized water at atmospheric conditions. Turbine flow meters are very sensitive to changes in

density, so a full calibration was performed to create a unique calibration for gasoline.

Megasquirt 2 incorporates an output test mode in which injectors can be manually fired given total output time, injection time, and number of injections. An output time of 30ms, corresponding to 4000RPM, was held constant while duty cycle (ratio of injector open time to total output time) was varied from 10% to 80% to cover the majority of the fuel injector's operating range. After mounting the injector above a graduated cylinder, the injector was fired with 40psi constant differential fuel pressure for a fixed amount of time. A rubber tube and plastic cap were also used between the injector and graduated cylinder to minimize variation in measurements due to loss of fuel vapor.

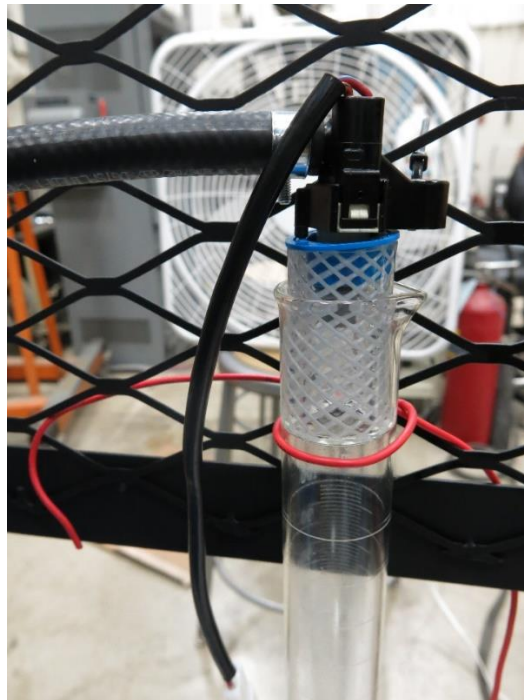


Figure 36. Fuel injector test setup

During calibration, the injector output is set to fire for a fixed duty cycle for an amount of time that allows a large portion of the graduated cylinder to be filled. While the injector was firing, the average voltage output from the flow sensor was acquired from the oscilloscope available in the NI MAX interface.

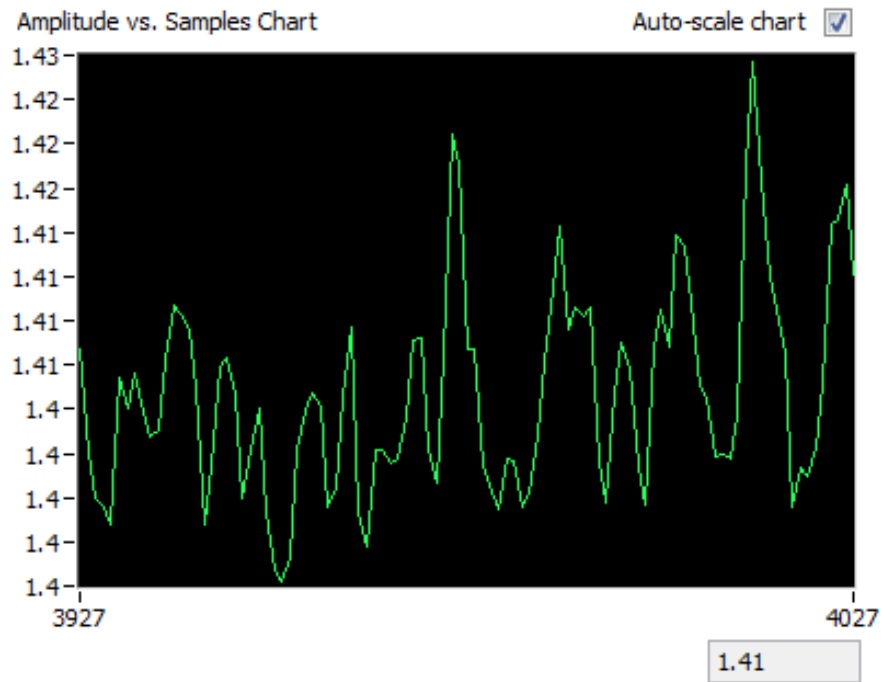


Figure 37. Oscilloscope output of flowmeter during test

After the test is complete, the total volume is divided by the time corresponding to the number of injections to find the volumetric flow rate. The raw voltage value from the flow meter is then related to that particular flow rate. This process is repeated for the range of duty cycles, and a new calibration curve was created that relates flow meter voltage to the corresponding volumetric flow rate of fuel.



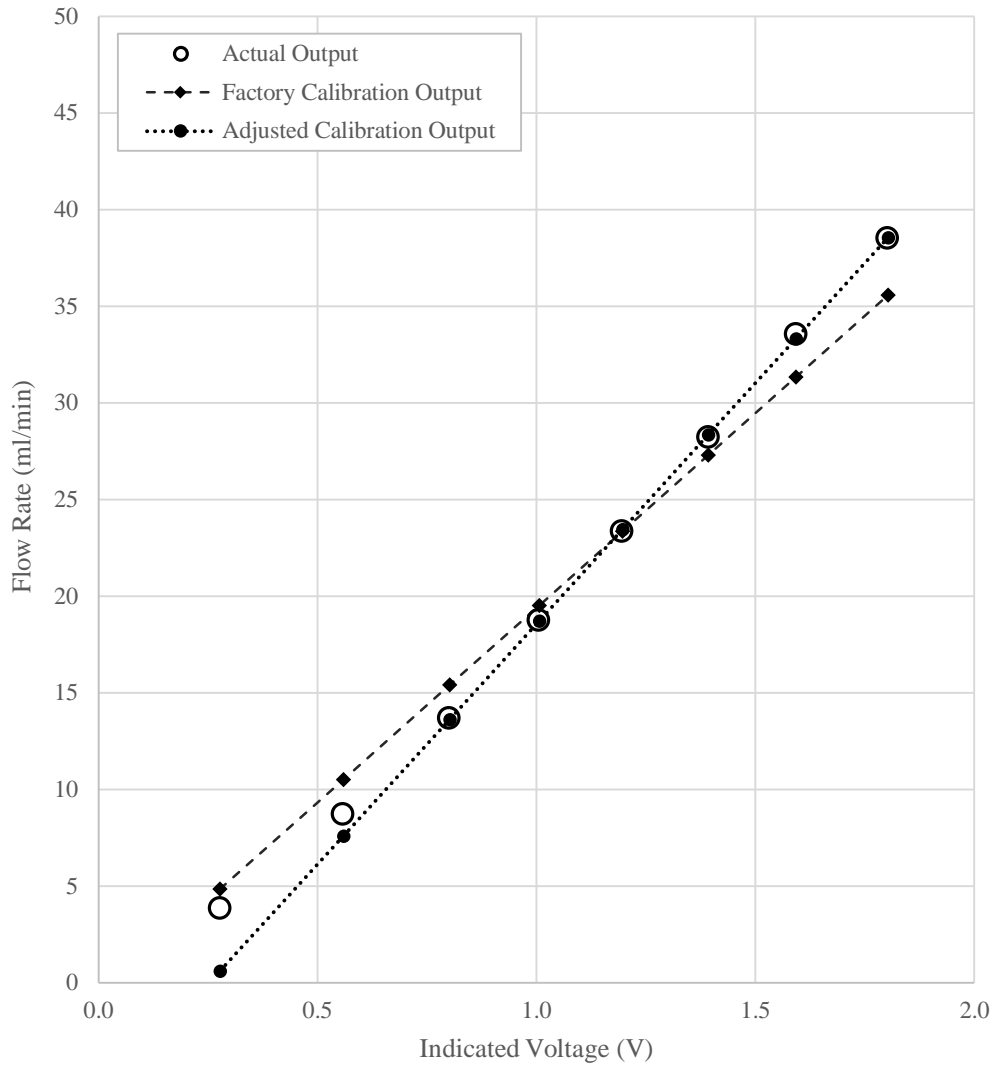


Figure 38. Factory supplied flow meter calibration compared to corrected calibration

The figure above reflects the non-linearity of the flow meter output at flows below its specified 13 mL/min flow range specification. The next figure compares measurement error between the factory and corrected calibrations.

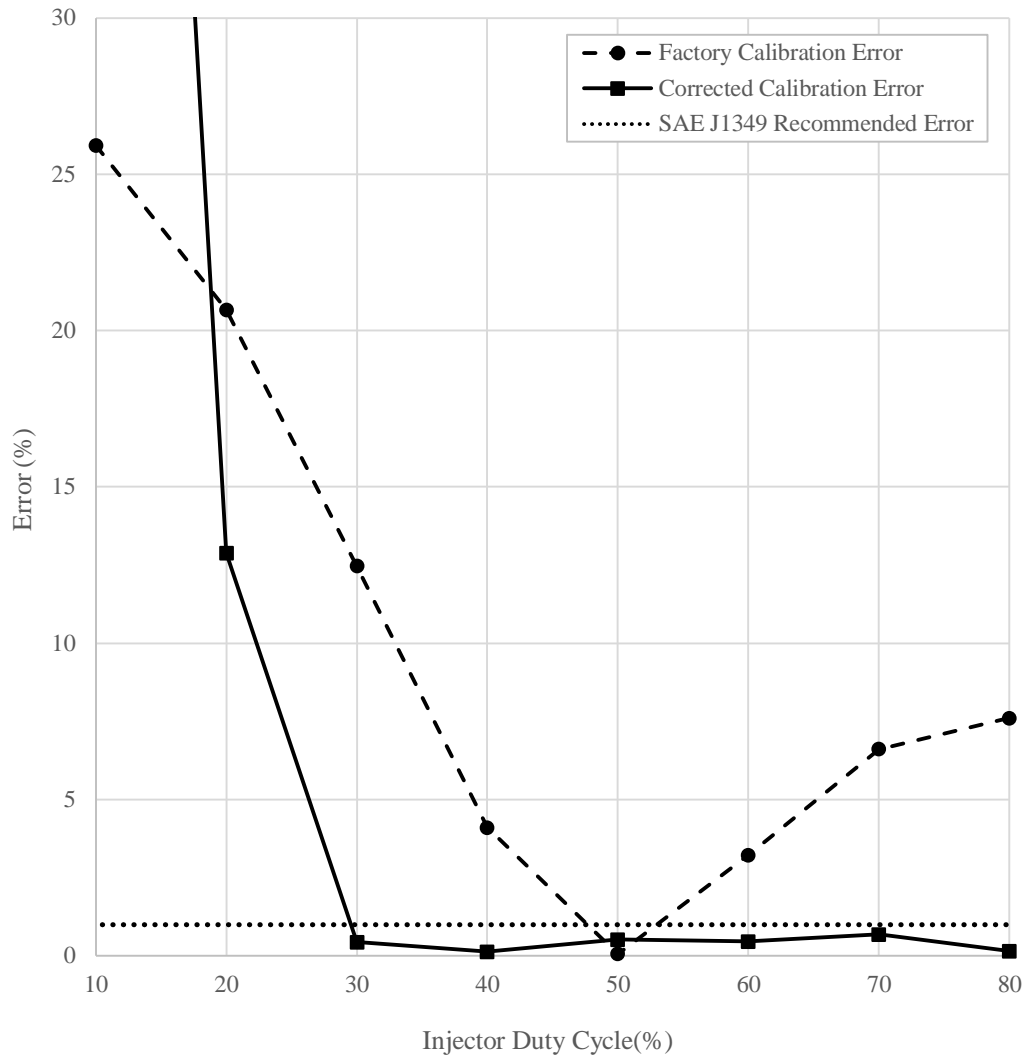


Figure 39. Flow meter error with increasing duty cycle

The corrected calibration curve sacrificed 87% error at 10% duty cycle in order to increase the linearity at duty cycles above 30%. Unfortunately, the range of acceptable output (>30% duty cycle) overlapped minimally with the current duty cycles of the engine at WOT shown in the following table.

Table 1. Baseline duty cycle range of engine

RPM	500	2000	3000	3500	4000	4500	5000	6000	6500	7000	7500
Duty Cycle (%)	4.6	13.2	19	22.1	25.3	28.9	32.5	38.9	41.1	43.7	49.5

Megasquirt allows the user to input an assumed volumetric efficiency table to modify fuel flow throughout the engine speed range. The data acquired above is calculated from the VE values, engine displacement, and  $\lambda = 0.85$  (an average value seen during benchmark testing).

The shaded values indicate the engine operating range in which error would be unacceptable if measured with the existing fuel flow meter. Although a different fuel flow meter more capable of high accuracy in this flow range was desired, the associated price was not justifiable for this project. Instead, fuel flow for the majority of testing was measured average duty cycle output of the ECU, obtained from video recordings of steady state testing.

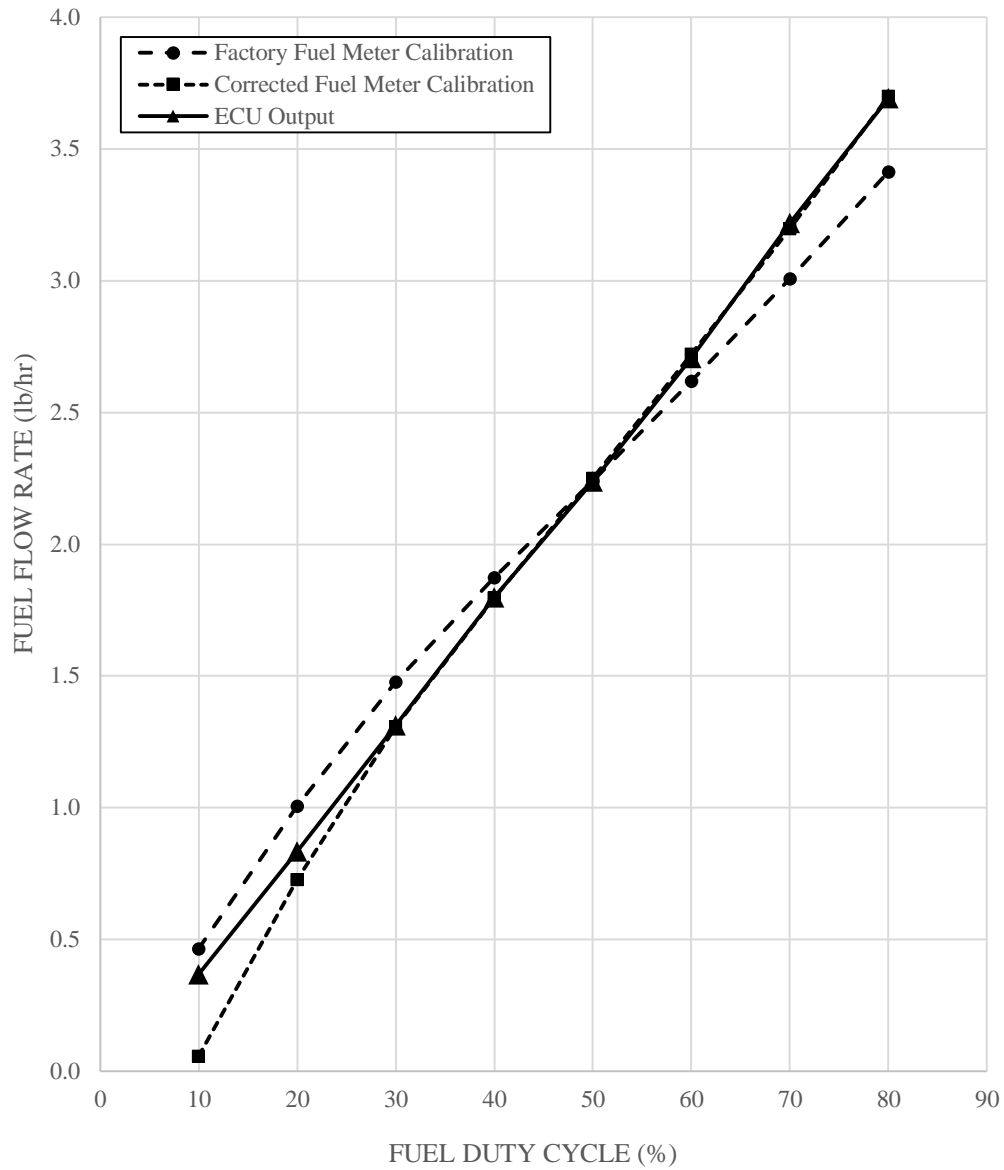


Figure 40. Fuel flow measurement comparison

The figure above shows the fuel flow rate is directly proportional to the flow rate through the following relationship:

$$\dot{m}_{fuel} = 0.474 * (Duty Cycle) - 0.1114 \quad (3)$$

This equation, however, indicates that there is a non-linearity in injector behavior somewhere between 0 and 10% duty cycle. It was assumed that this non-linearity occurs at very small duty cycle values ( $<1\%$ ), where the response time of fuel flow through the injector is too slow to follow the square wave signal of the rapid open and closing of the injector. The range of possible non-linearity for the injector was not tested due to large variations in measurements from fuel vaporization and excessive run times to get an adequate volume of fuel for measurement.

By utilizing this relationship, resolution of the ECU to control the injector duty cycle became the main source of uncertainty, along with the resolution of the values it displays through the user interface where data is taken from.

### 3.8.5 | Test Repeatability

Engine speed variation, chain dynamics, and engine temperature fluctuation all presented a challenge for meeting the 1% repeatability requirement for brake torque output set by SAE J1349. This section outlines how subsystems were iterated to meet this standard.

The following table and figures summarize the drive system iterations and their respective repeatability and absolute variation in speed and torque.

Table 2. Drive system iteration summary

Iteration	System Summary
1	Chain Drive, No Tensioner, No Temperature Control
2	Chain Drive, Torsional Tensioner, No Temperature Control
3	Chain Drive, Torsional Tensioner, Temperature Control
4	Chain Drive, Double Idler Tensioner, Temperature Control
5	Chain Drive, Stiff Double Idler Tensioner, Temperature Control
6	Direct Drive, Stiff Spider Coupling, Temperature Control

For repeatability testing, steady state was achieved at 4000 RPM with engine temperature held within  $167 \pm 2$  °F, well within the SAE standard of  $\pm 3.6$  °F (1%). After one minute of data was captured at 1 Hz, the load was removed, the engine allowed to idle, then the test was immediately repeated for a total of 10 tests. In this context, repeatability was defined as:

$$R = \left( \frac{Y_{avg,1} - Y_{avg,10}}{Y_{avg,10}} \right) * 100 \quad (4)$$

$Y_{avg,10}$  = Average value of all 10 averaged values

$Y_{avg,1}$  = Average value of single test

The following figure summarizes the variation in overall repeatability throughout drivetrain iterations.

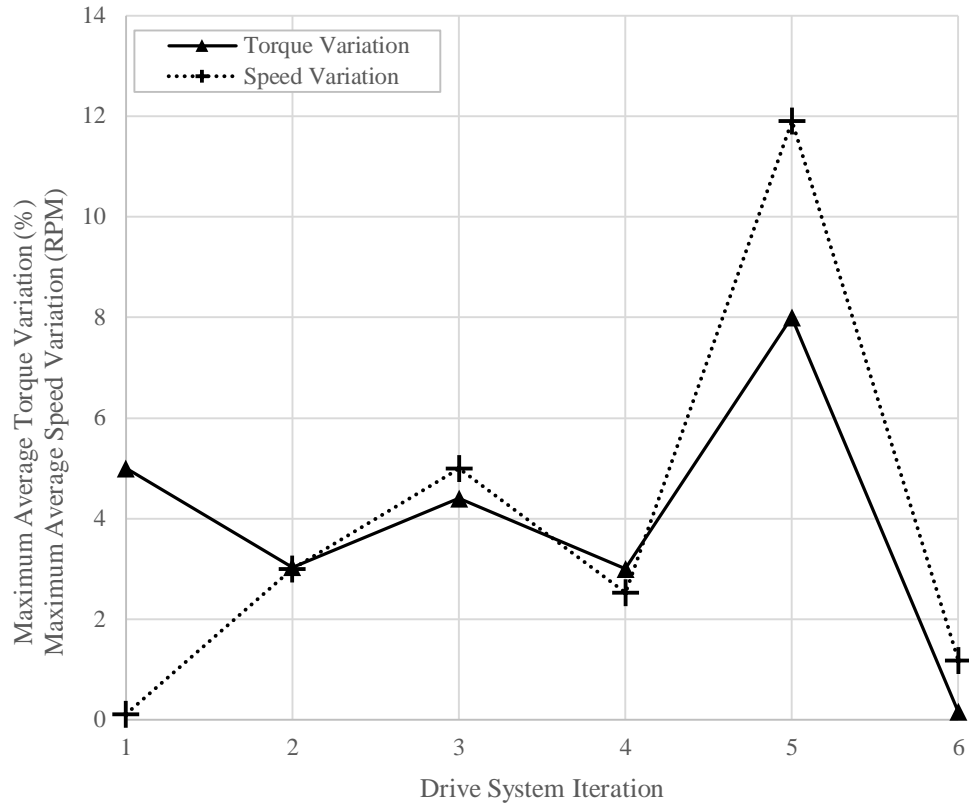


Figure 41. Repeatability variation throughout drive system iterations

Although the direct drive system was initially more difficult to set up, it provided more reliable results with most data falling well within the SAE J1349 requirements.

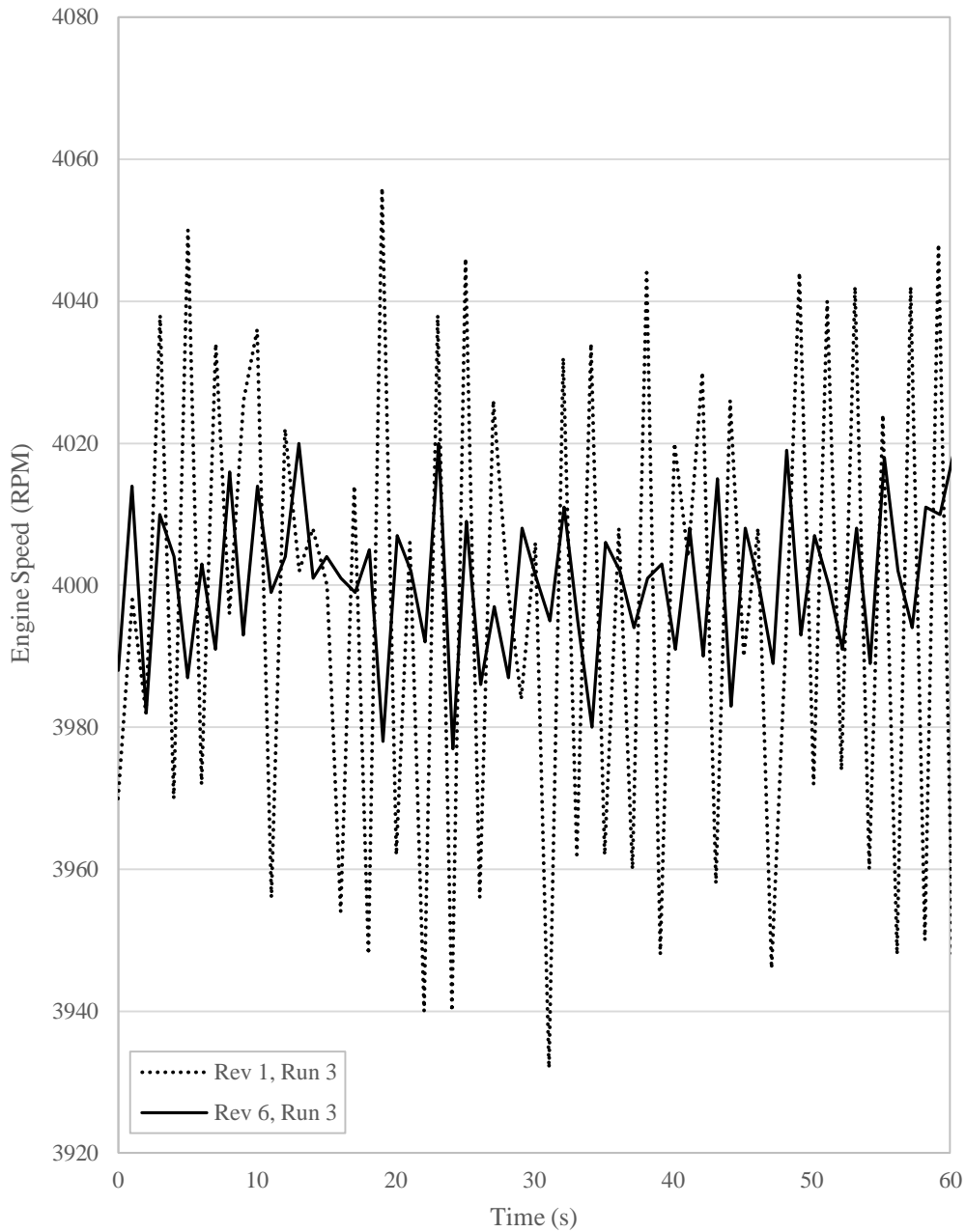


Figure 42. Engine speed variation between chain and direct drive

Engine speed with the direct drive system varied quickly by nearly  $\pm 20$  RPM throughout steady state tests. The improved repeatability most likely stemmed from the large improvement over the chain drive system, which caused



instantaneous engine speed to vary by sometimes more than  $\pm 60$  RPM. At this point, it was believed that the engine speed variation was due to the quick acceleration and deceleration of rotating components within small displacement single cylinder engines.

The next sections outline each iteration of the drive system with more detail, with emphasis on potential reasons for their associated lack of accuracy or repeatability.

### Iteration 1: Chain Drive

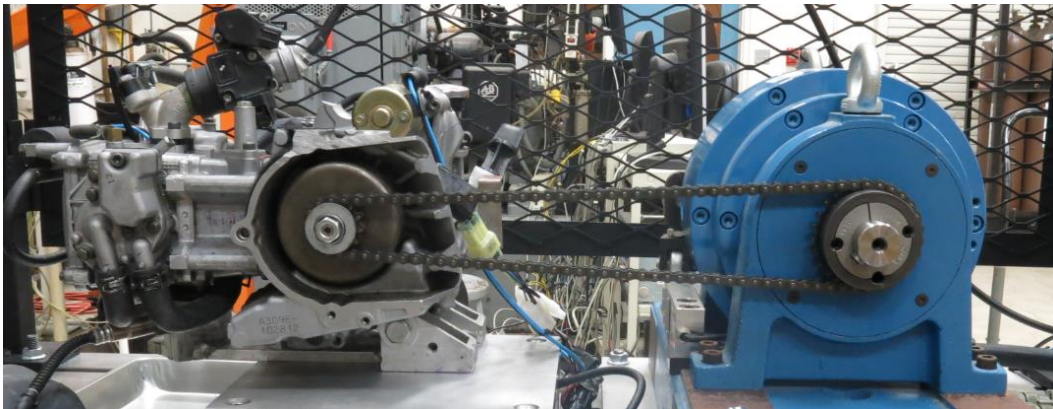


Figure 43. First chain drive iteration

The centrifugal clutch allowed the engine to idle without spinning the dynamometer enabling a more stable engine start and idle condition that is necessary under competition conditions. However, it introduced another source of relatively movement and was welded shut to allow testing at lower engine speeds. For this reason, the central hub and outer housing were welded together.



Figure 44. Welded centrifugal clutch

For alignment, the clutch hub and dynamometer face was checked for squareness against their respective mounts. From there, the assembly was rotated by hand and the engine was moved parallel to the dynamometer in order to reduce the resistive torque. In hindsight, a more accurate method of ensuring proper alignment may have alleviated some chain vibration issues that are described in the following sections.

During tests, the vibrations caused by the oscillating tension of the chain and harmonics of the system had negative consequences on both the physical system and data. The chain vibrated in combinations of vertical and horizontal directions, nearly causing contact with the dynamometer itself. Additionally, the crankshaft sheared at the clutch mounting point, necessitating a full rebuild before further testing.



Figure 45. Crankshaft failure plane on output side

The failure was believed to be caused by a combination of the sharp, high frequency loads seen through the welded clutch, and the transverse load applied by chain tension.

Revision 2: Torsion-spring chain tensioner



Figure 46. Torsion spring tensioner setup

Along with reinstalling the centrifugal clutch pads to reduce torque during startup and shut down, a torsional spring tensioner was installed to maintain chain tension

and reduce chain vibration. This setup only supported the chain on the loose side, or the side that doesn't experience tension when engine power is applied. In certain engine speed ranges, however, the tight side of the chain experienced extreme oscillation most likely due to multiple modes of its natural frequency.

Due to the lack of temperature control, the engine temperature varied from 140 to 190 °F throughout this 60 second test, causing the drift in torque output by as much as 5.6% as shown in the following figure.

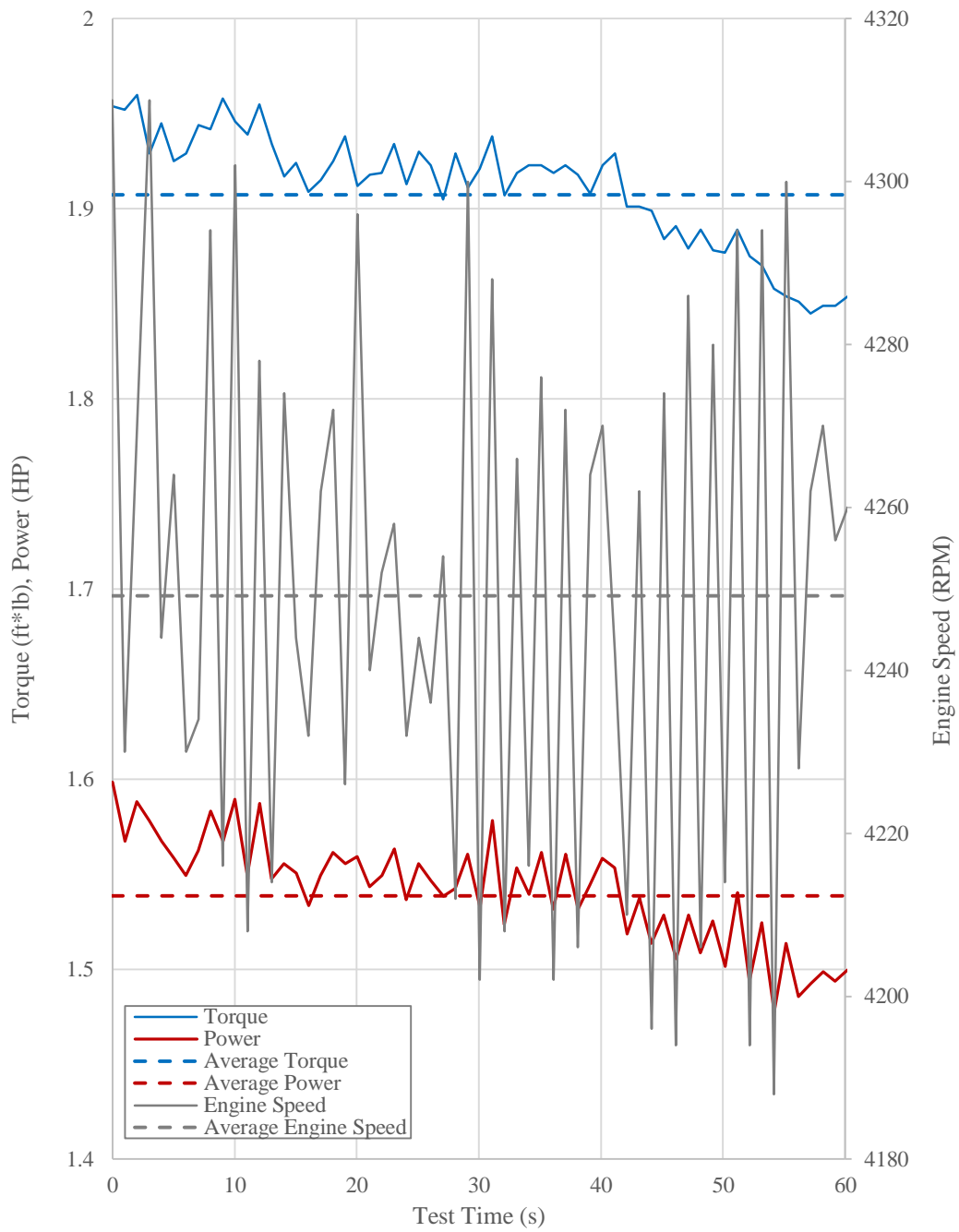


Figure 47. Torque, power variation with no temperature control

Revision 3: Temperature control, torsional chain tensioner



Figure 48. Air mover with butterfly valve and ducting for temperature control

A centrifugal blower was mounted at engine level with ducting to direct the air over the radiator. A butterfly valve was installed, which was controlled by the operator with a rod acting on a lever. Though seemingly crude, it was highly effective. With 3 different blower speeds and infinite control over the butterfly valve, engine temperature was regulated within  $\pm 2$  °F for every test, and it was able to keep engine temperature below 130 °F under full load conditions. The following figure shows that the temperature variation in the previous test did indeed cause the torque data to drift, since there is much less drift with controlled engine temperature.

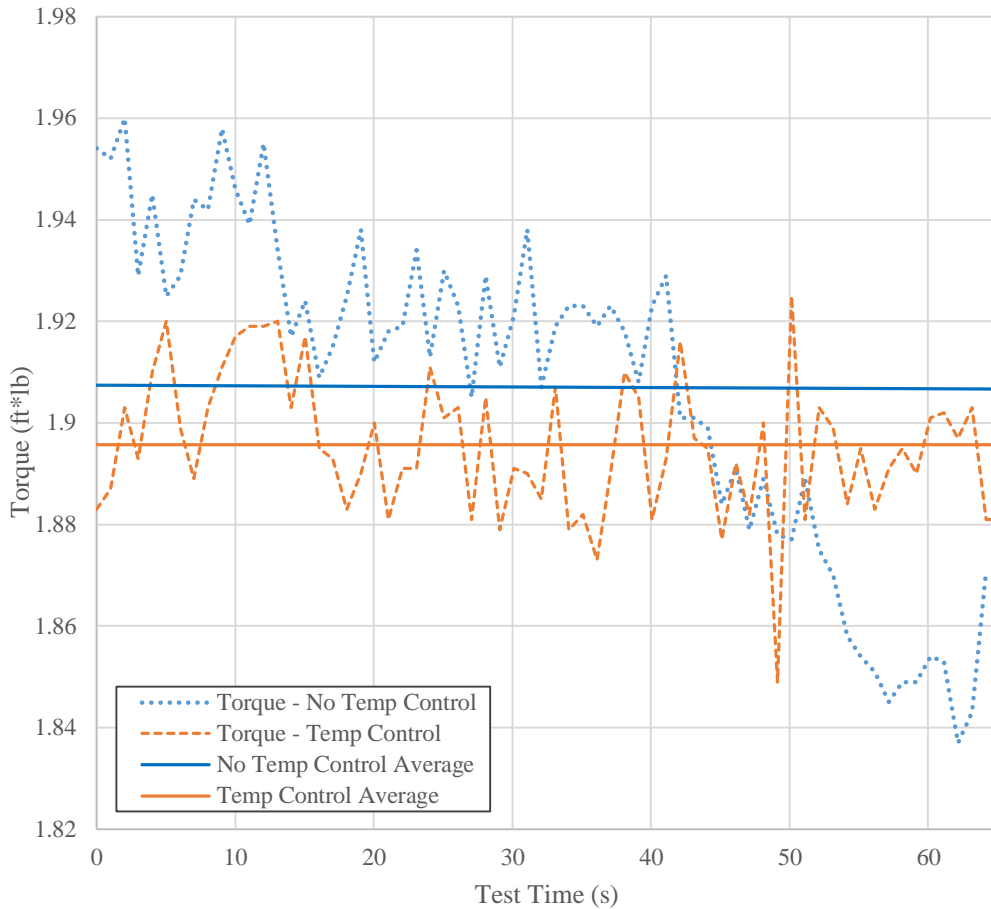


Figure 49. Torque output comparison with temperature control @ 4250RPM

With temperature control, the absolute value of average torque variation (outside of outlying data points) remained within 1% of the average value. This compares to 5.6% percent without temperature control. Since duty cycle and ignition timing remained constant throughout both tests, it is safe to assume that the difference in torque variation (~4.6%) is due to transient temperature behavior of the engine.

Revision 4: Loose Double Idler, Temperature Control

In order to attenuate the chain's oscillation on the tight side, a system was designed with a second idle sprocket, and is depicted in the following figure.

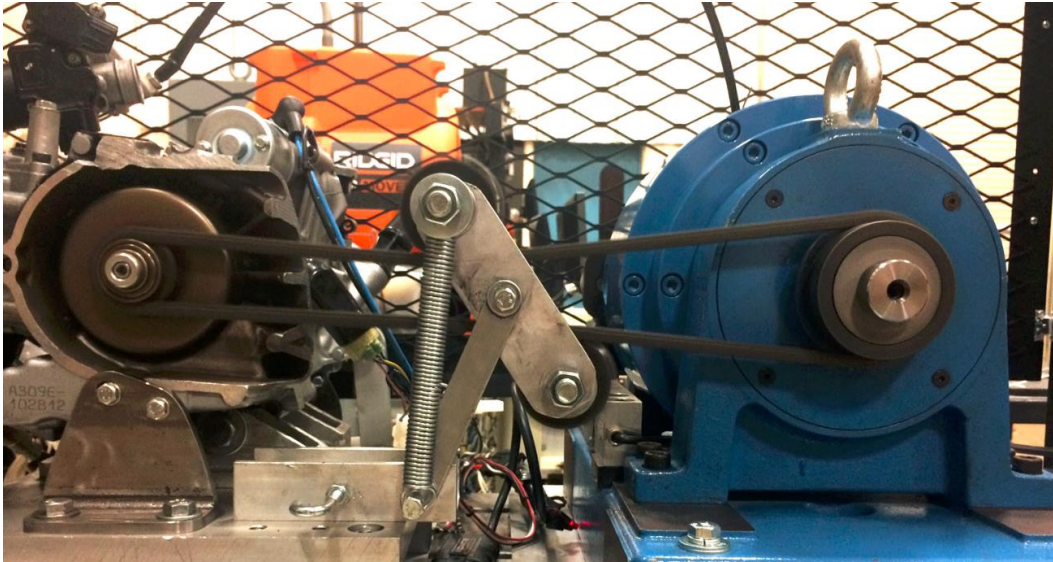


Figure 50. Double-idler iteration of chain tensioner

A pair of two force members held the idler arm in place, and were free to rotate with thrust bearings at each mounting point. With the spring tension applied through a moment applied to the idler arm, it was suspected that the stability of torque readings would increase due to tension force applied to both tight and loose sides of chain as the spring oscillates to maintain sprocket contact. This would result in tension force acting on both sides of the dynamometer sprocket, ideally canceling much of the torque variation.

Unfortunately, the idler was allowed too many degrees of freedom and failed to adequately dampen chain vibration. Although results improved, violent movement and quick prevented extended testing at higher engine speeds.



#### Revision 5: Stiff Double Idler, Temperature Control

For this iteration, thrust bearings were installed on all joints to reduce overall movement of the tensioner, and were tightened only to the point where lateral movement was very restricted.

Though chain vibration was reduced drastically, it caused the chain to heat up to the point where the lubrication between links liquefied and caused the chain to become very hot and stiff. With the additional stress from links bending around the small front sprocket, the crankshaft saw another failure.

#### Revision 6: Direct drive system

Multiple crankshaft failures caused by the loading condition of the chain drive forced the implementation of a direct drive system the majority of forces act in torsion, instead of transversely.

For the direct drive system, high speed couplers with an elastomer coupling were chosen. The dynamometer and engine setup were then modified to accommodate the alignment and spacing requirements.

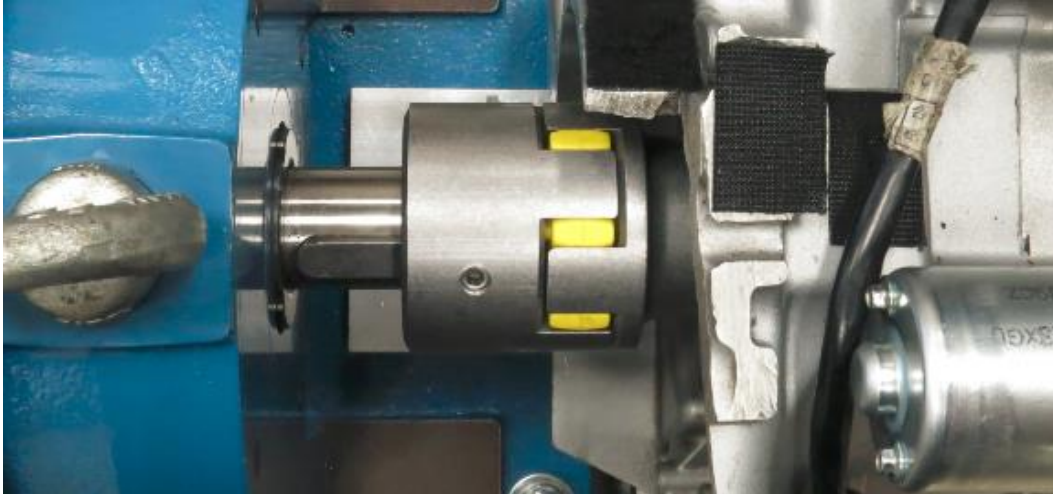


Figure 51. Direct drive setup (Top View)

This setup caused the crankshaft to rotate relative to the flywheel and coupler on the engine, loosening them on both ends. To alleviate this, Loc-tite slip fit compound was used on the tapered flywheel shaft, along with lock washers and permanent Loc-tite for each nut.

The yellow elastomer seen in the figure above had a hardness rating of 92 Shore A [11]. During testing, the relatively movement of the jaws heated the elastomer to the point of slightly melting, which caused material to be thrown out radially.

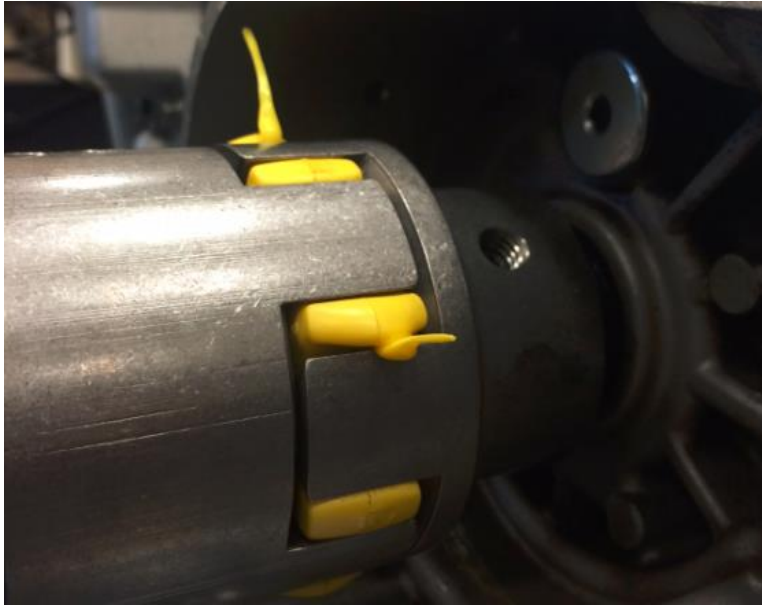


Figure 52. Melted elastomer thrown radially

To address this issue, a stiffer elastomer with a 98 Shore A rating was installed with Vaseline lubrication to reduce friction heating.



Figure 53. Stiffer elastomer installed with lubrication

## 4 | BENCHMARK TESTING

The goal of benchmark testing was to first characterize the torque, BSFC and temperature gain for this particular engine, then develop a relationship for these as lambda, ignition, and engine temperature were varied. The results were then used in the vehicle simulation to help identify engine tuning targets.

### 4.1 | BASELINE PERFORMANCE

Due to the crank sensor fault limiting the engine to 5000 RPM, data was acquired from 3000-4800 RPM. Each data point was acquired under steady state conditions in accordance to SAE J1349.

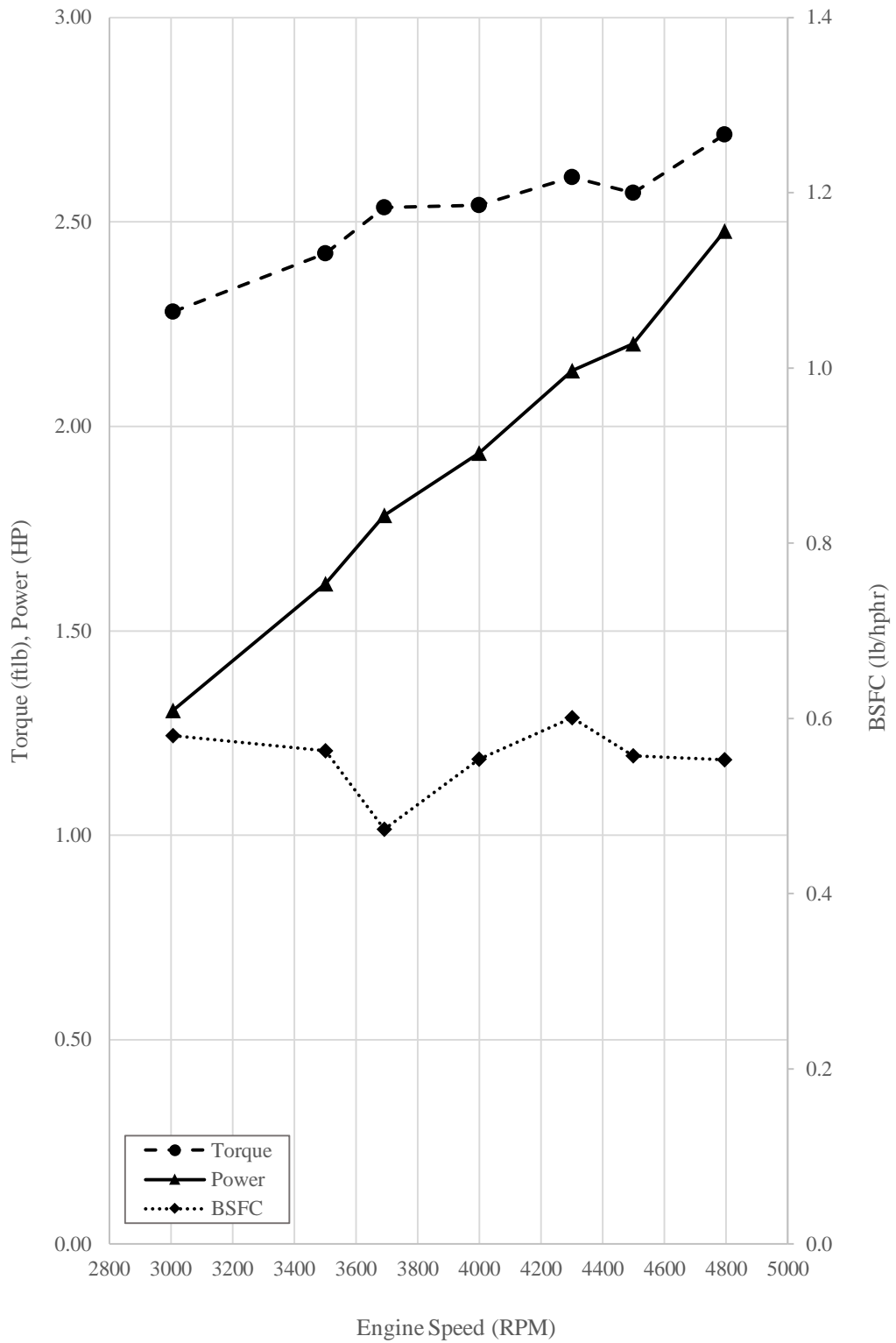


Figure 54. Baseline performance of engine

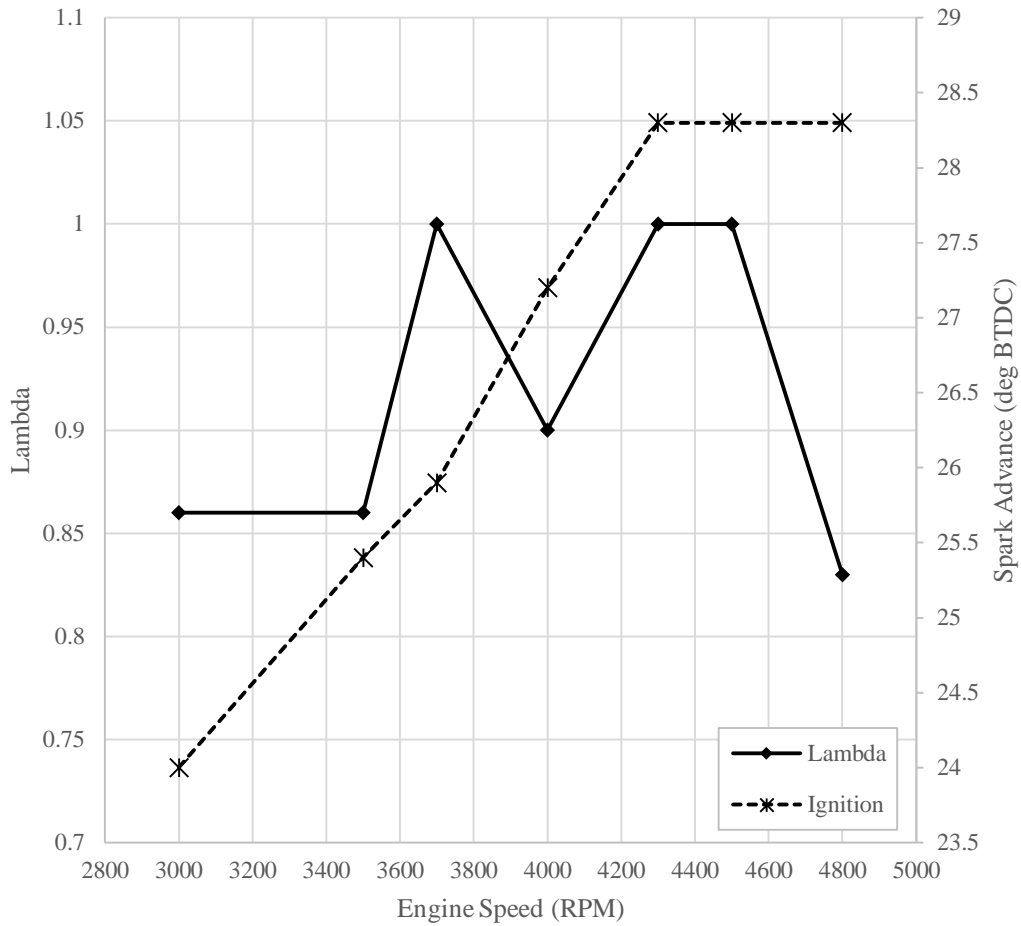


Figure 55. Lambda and ignition values in benchmark tune

Torque values varied by only 8% across the operating range, and BSFC values were larger than expected - likely attributed to relatively rich operating conditions. The dip in BSFC at 3700 RPM was most likely due to the stoichiometric AFR, instead of 14% rich at both 3000 and 3500 RPM. The largely rich tune was caused by the aforementioned lack of AFR feedback during previous tuning. Fuel duty cycles were previously manipulated through subjective driver input, and riders would feel that more power was produced from the engine with a slightly rich mixture.

#### 4.1 | LAMBDA VARIATION

Due to the uniqueness of engine design, combustion duration, mixing properties, and cycle-to-cycle characteristics vary greatly between applications. Since these play a large role in determining the maximum torque and minimum BSFC points, experimentally obtaining the relationship between air-fuel-ratio and torque was a crucial exercise to increasing fuel efficiency of the engine.

With the engine operating at steady state, wide open throttle (WOT) @ 4000 RPM, the fuel duty cycle was varied to allow lambda to range from 0.8 to 1.2, ignition was held constant at 24 degrees before top-dead-center (BTDC) ( $\Phi = 0^\circ$  indicating maximum brake torque (MBT) timing), while engine temperature was held at  $167 \pm 3^\circ$  F.

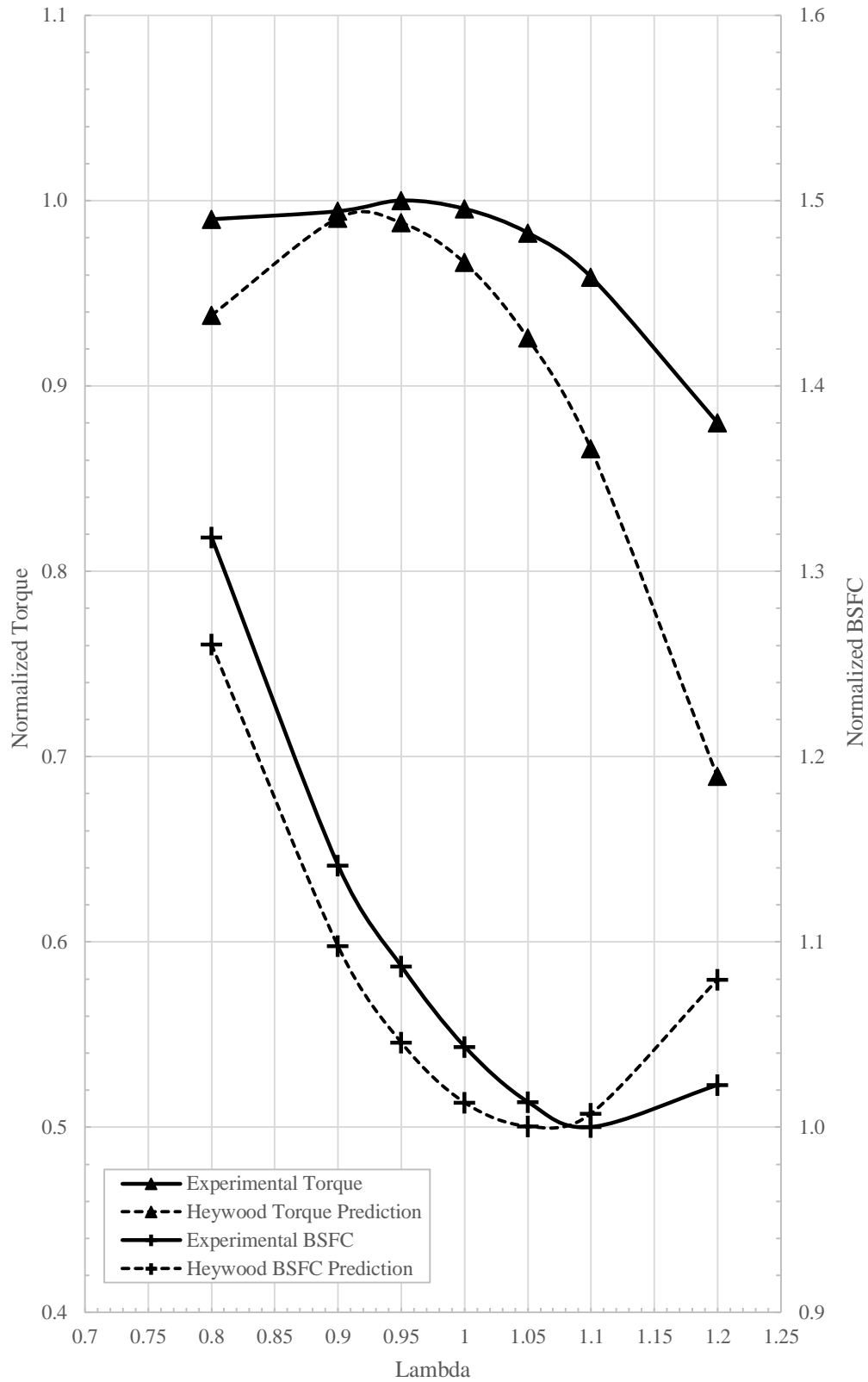


Figure 56. Torque and BSFC variation with lambda (4000 RPM,  $\Phi = 0^\circ$ )



Minimum BSFC was observed at  $\lambda = 1.1$ , which differed from the 5% lean condition in Heywood's general prediction. Lower temperatures associated with lean combustion prevent the creation of excess oxygen through dissociation of products. Even though this results in less net mechanical work, the complete combustion of the all fuel increases the fuel conversion efficiency of engine, thereby decreasing BSFC. Longer combustion durations play a large role in causing the BSFC to increase again after  $\lambda=1.0$  [2].

Maximum torque was seen at a slightly rich condition ( $\lambda= 0.95$ ) which parallels the widely accepted trend. The higher temperatures present from combusting a larger mass a fuel at rich conditions causes dissociation of some of the products ( $\text{CO}_2$ ,  $\text{H}_2\text{O}$ ), releasing excess oxygen that allows extra fuel to undergo partial combustion. Although this results in the highest net mechanical work, some fuel is left unburned. At mixtures richer than that, too much excess fuel and increased combustion durations due to slower flame speeds start to decrease efficiency [2].

Lambda values of  $\lambda = 0.85$  were common throughout benchmark testing, which from this trend, implies that BSFC of the engine could be improved nearly 23% by tuning for  $\lambda = 1.1$ . This is further analyzed and discussed in the results section.

#### 4.2 | IGNITION VARIATION

Due to the difference in flame speed compared to varying engine speed, it is important to find individual spark advance values that maximize combustion

pressure at just after top dead center (TDC), creating a maximum brake torque condition (MBT). Since MBT ignition timing varies with engine speed, it is normalized relative to MBT timing throughout this study.

Ignition advance was varied  $\pm 4$  degrees from MBT conditions, while lambda and engine temperature were held constant at  $\lambda=1.0$  and  $167 \pm 3^\circ \text{ F}$ .

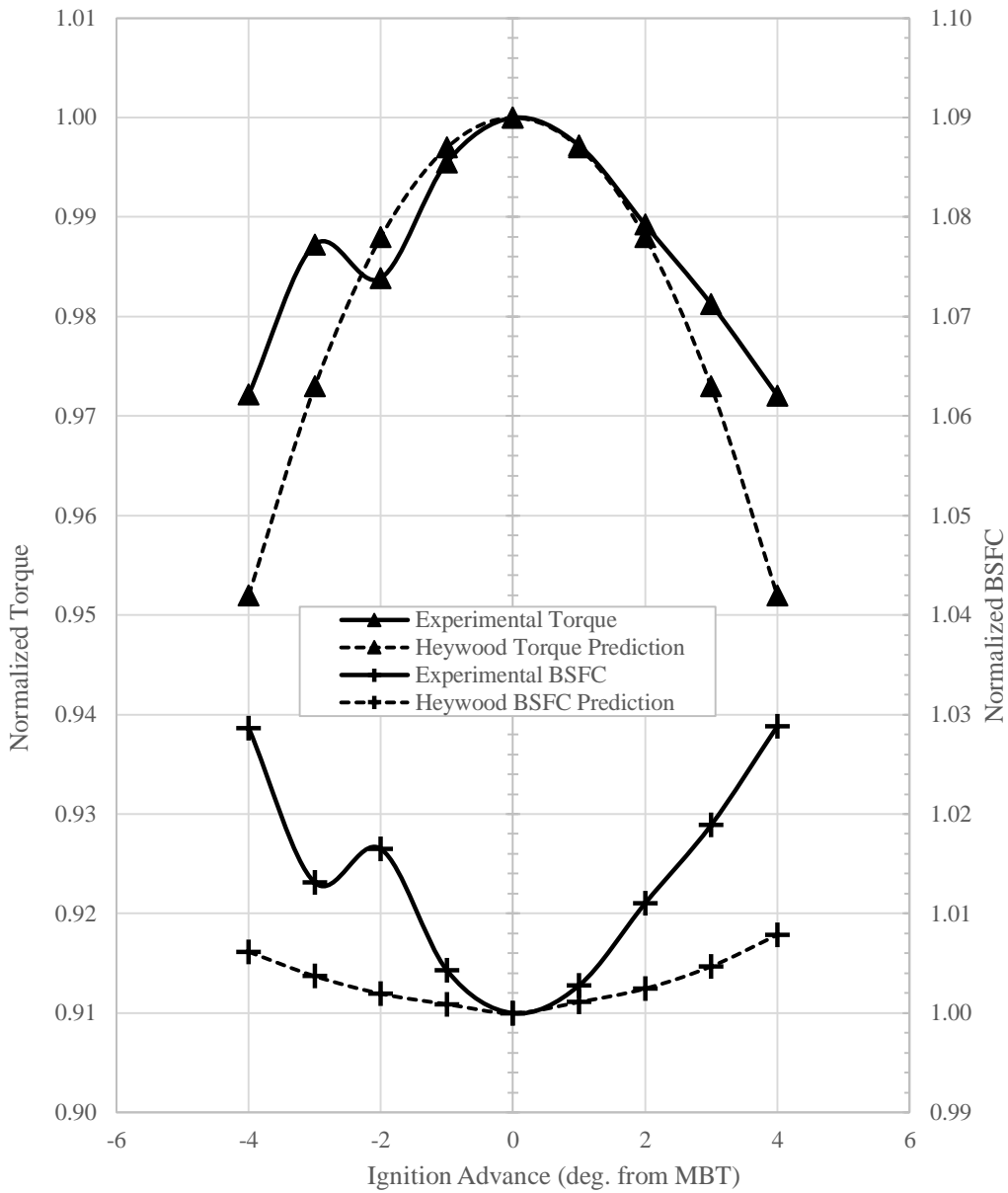


Figure 57. Torque variation with ignition timing (4000 RPM ,  $\lambda=1.0$ )

Both torque and BSFC variations remained relatively symmetric about MBT, varying about 3% in either direction. It is important to note that the change in both BSFC and torque are much smaller with variations in ignition, implying that air-fuel-ratio is much more important than ignition when considering improving fuel economy.

The obvious break in the parabolic trend seen is at  $-2^\circ$  from MBT, where an uncharacteristic torque decrease also causes an increase in BSFC at that point. This outlying point existed throughout multiple tests. Since lambda, engine temperature, manifold temperature, and battery voltage were held constant throughout the tests, this break in trend is believed to be a unique hardware issue. Nevertheless, this curve was used during simulation in future sections since all relevant hardware used in competition was identical to the test setup.

#### 4.3 | TEMPERATURE VARIATION

The largely transient nature of engine operating mode introduced varying engine temperature as a variable, so this test was performed to understand the effect of various engine operating temperatures on both torque output and BSFC.

First, the engine components and oil were warmed by performing a one minute WOT test at 4000 RPM and  $167 \pm 3^\circ$  F. Tests were then performed at  $20 \pm 2^\circ$  F intervals by changing the flow rate of air over the radiator with the centrifugal blower, while ignition and lambda were held at MBT and  $\lambda = 1.0$ .

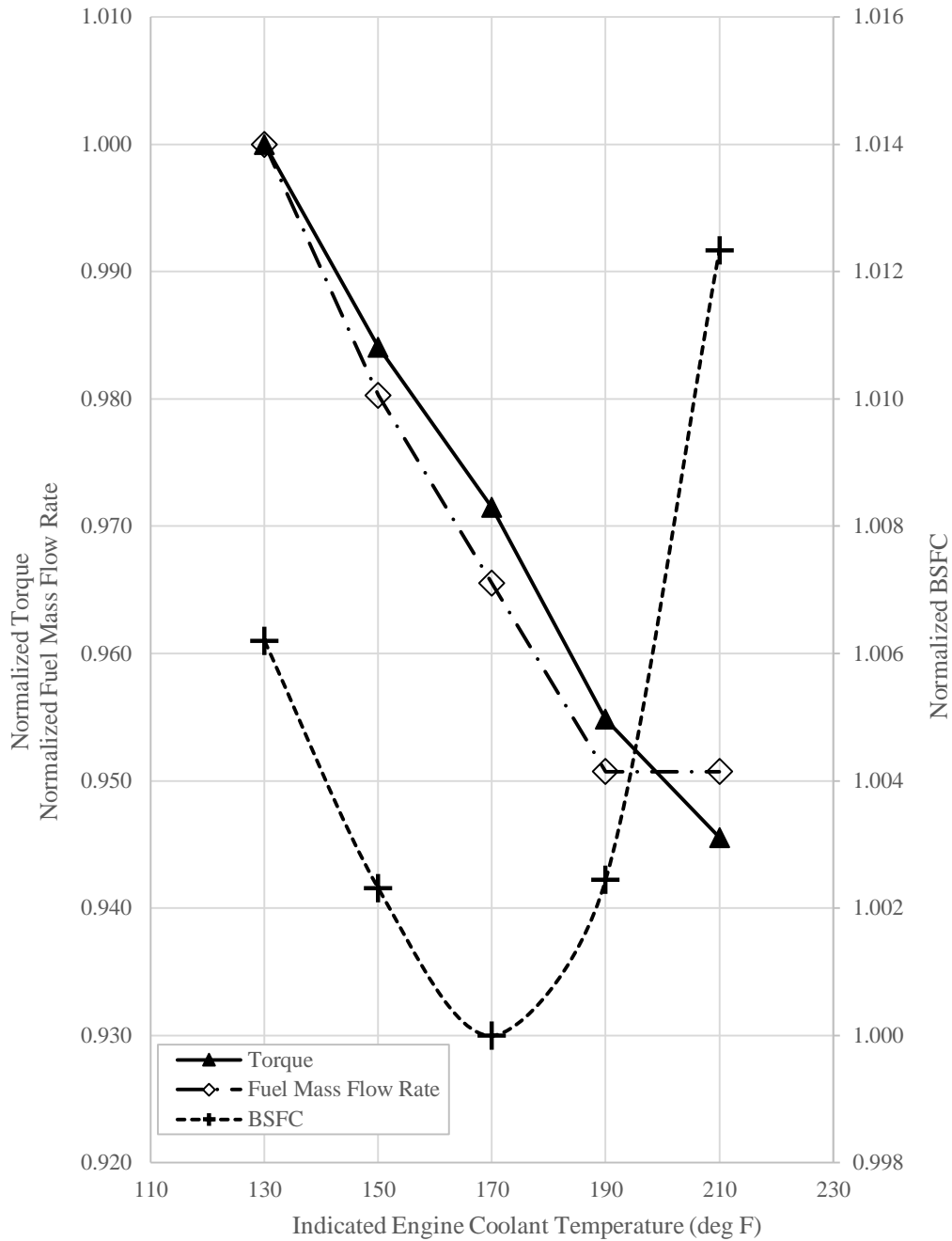


Figure 58. Torque, BSFC variation with engine temperature

(4000 RPM,  $\lambda=1.0$ ,  $\Phi=0^\circ$ )

Torque output decreased by 5.4% across the temperature range, while BSFC reached a minimum at 170 °F, then increased to 1.2% of its minimum at 230 °F.

$$VE = \frac{2\dot{m}_a}{\rho_{a,i}V_d} = \frac{2}{V_d} \left( \frac{\rho_a}{\rho_{a,i}} \right) \dot{V}_a \quad (5)$$

- VE = Volumetric Efficiency
- $\rho_{a,i}$  = Inlet Air Density (kg/m<sup>3</sup>)
- $\rho_a$  = Engine Air Density (kg/m<sup>3</sup>)
- $V_d$  = Engine Displacement (m<sup>3</sup>)
- $\dot{V}_a$  = Volumetric Air Flow Rate (m<sup>3</sup>/s)

The equation for volumetric efficiency, presented in the equation above, offers insight into the fuel mass flow rate trend seen in Figure 58. Equation (1) shows that volumetric efficiency is directly proportional to mass flow rate of fuel at a constant AFR. With increasing engine temperature, decreasing fuel flow rate implies that intake air density is also decreasing, assuming constant inlet conditions, volumetric flow rate of air, and engine displacement during steady state testing. If air density decreases linearly with increasing temperature, this relationship suggests that fuel mass flow should also decrease linearly with increasing engine temperature. Although this behavior is observed from 130 – 190 °F, fuel mass flow rate remains constant from 190 – 210 °F, which may be due to other factors such as increasing engine friction.

$$BSFC = \frac{\dot{m}_f}{P} = \frac{1}{\omega_e} \left( \frac{\dot{m}_f}{T_e} \right) \quad (6)$$

$$\begin{aligned} \dot{m}_f &= \text{Fuel Mass Flow Rate} \\ &\text{(kg/s)} \\ T_e &= \text{Engine Torque (Nm)} \\ \omega_e &= \text{Engine Speed (rad/s)} \end{aligned}$$

The trend for BSFC can be studied by analyzing its definition, shown in the equation above. At a constant engine speed, the change in BSFC is directly proportional to the different in rates of change of mass flow rate and torque. The following relationship is suggested to represent this trend.

$$\frac{d}{dT}(BSFC) \propto \frac{d}{dT}(\dot{m}_f) - \frac{d}{dT}(T_e) \quad (7)$$

This shows that BSFC will decrease if the rate of fuel mass flow rate is greater than the rate of torque output variation, and vice versa. BSFC remains constant if the rates are constant. Between 130 – 170 °F in Figure 58 the slope of mass flow rate and torque decrease diverge, with mass flow rate decreasing at a faster rate. This explains the initial decrease in BSFC, up to the minimum at 170 °F after which the slopes begin to converge again. Since fuel flow rate remains constant between 190-210 °F while torque continues to decrease, BSFC increases more rapidly.

It is important to note that while large variations in engine temperature have non-negligible effects on engine torque, BSFC changes minimally. For example, if the engine was operating at 210 °F instead of 170 °F, BSFC sees an expected increase of 1.2%. That is less than half the effect that ignition timing has

within the tested range, and nearly 28 times less of an effect on BSFC than running at  $\lambda = 0.8$  instead of  $\lambda = 1.1$ . Although it is further discussed in the results section, it is clear that if the ECU can compensate for variations in volumetric efficiency to maintain an ideal AFR, varying temperature should not play a large role in overall efficiency.

#### 4.4 | TRANSIENT TEMPERATURE

In order to incorporate temperature effects into the simulation, temperature rise and fall rate were necessary to track engine temperature. Before the exercise, it was unknown whether the rise rate would be relatively constant or be a function of power input.

To find this trend, all coolant was removed from the engine and the air system was left unpressurized through an open radiator cap. Then, load was applied at 4000 RPM and air was blown over the engine to keep it at a low temperature while  $\lambda = 1.0$  and  $\Phi = 0^\circ$  were achieved. Then, the air blower was turned off and the data was recorded while the engine was allowed to run up to 210 °F.

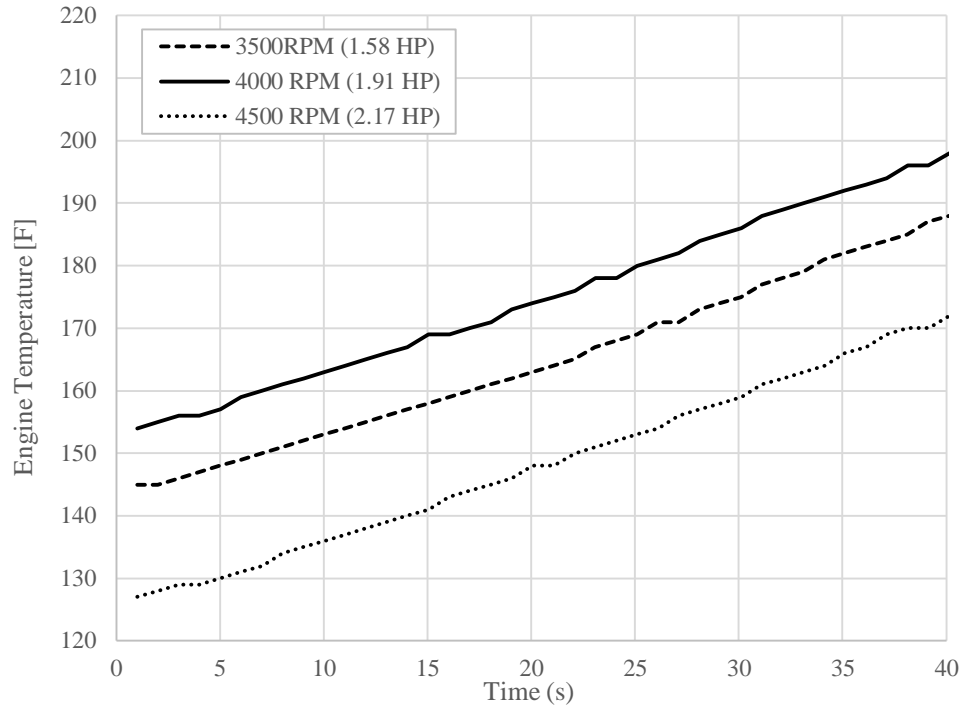


Figure 59. Engine temperature rise rate at various RPM

Although data was not taken from similar initial temperatures, the figure above shows that the temperature rise rate (slope) remained relatively constant independent of power input. For the sake of simplicity in the simulation, a constant value of 1.2 °F/s was used during engine burns. For the engine temperature fall rate, the engine was turned off and was allowed to cool down only through natural convection.



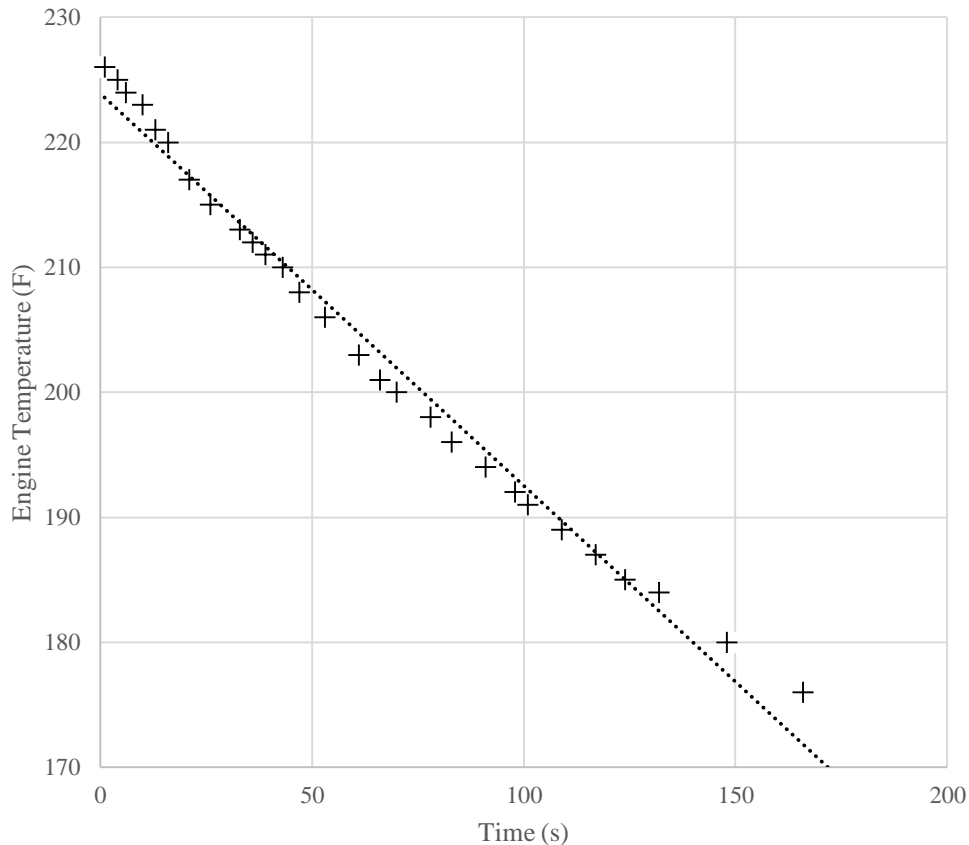


Figure 60. Temperature fall rate through natural convection

Also for simplicity, the engine fall rate was assumed to be linear with a value of -3.1 °F/s. Both temperature rise and fall rates, however, should vary with changing convection rates depending on competition conditions. With the engine enclosed in a tightly packaged carbon structure, these slopes will most likely noticeably change. It is suggested for the team to record engine temperature during burning and coasting conditions to adjust these slopes for use in the simulation.

#### 4.5 | ENGINE IDLE / START

To justify the “burn and coast” method used in this competition, cranking and idle duty cycles were compared to quantify a crossover time at which one is more fuel

efficient than the other. For the engine idle condition, the engine was warmed up and allowed to idle at 160°F, resulting in a duty cycle of 5.7% and fuel consumption of 0.134 mL/s. For the engine start condition, the ignition was disconnected and the engine cranked at a similar engine temperature to capture a 3.4% duty cycle. After reconnecting the ignition and running the engine to clear excess fuel, the time required to start the engine was measured five times and averaged at one second, resulting in 0.042mL fuel consumption. This resulted in a 0.31s crossover time.

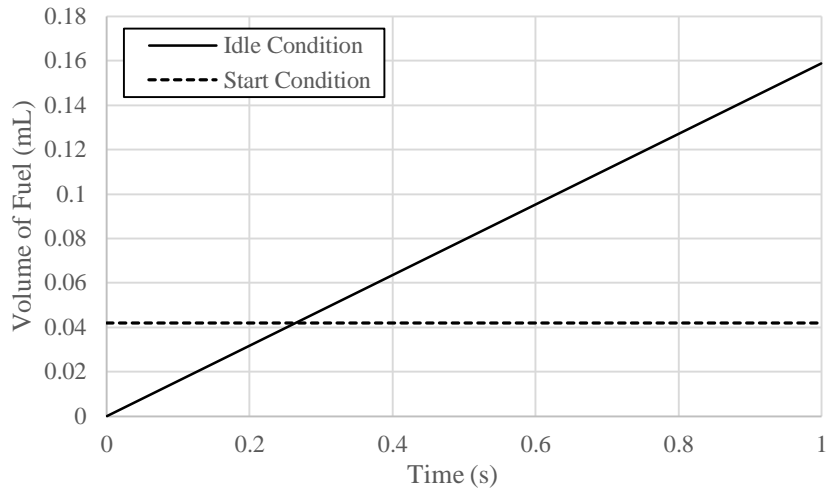


Figure 61. Fuel consumption between idling and starting conditions

This crossover time shows that idling the engine for times less than 0.31 seconds is more efficient than cutting and restarting the engine. The existing tune had very little cranking and temperature compensation, which accounts for most of the additional fuel necessary during starting. Due to the short crossover time and relatively long coast times discovered later in this study, it is safe to conclude that the burn-and-coast method is more efficient than idling the engine between acceleration events.

## 5 | VEHICLE SIMULATION DEVELOPMENT

During steady state operation, minimizing fuel consumption entailed modifying ECU parameters to reach target values for AFR and ignition timing obtained from the trends observed in baseline testing. In order to investigate potential differences in target parameters for minimizing fuel consumption under steady state testing and transient competition conditions, a straight-line vehicle model was developed.

### 5.1 | DISCLAIMER

It is important to note that vehicle data was not available at the time of this study, therefore the simulation tool is not verified. Even though final fuel economy figures were similar to those achieved during the 2014 competition, there are still a multitude of unknown variables that affect the outcome of the simulation. While the absolute accuracy of the simulation remains to be verified by the team, governing equations and engine parameter relationships were carefully developed to study general trends in performance (outlined in Appendix 5).

### 5.2 | MODEL ASSUMPTIONS

The following table outlines different parameters assumed in the model and their expected behavior during competition.

Table 3. Summary of simulation assumptions

<b>Parameter</b>	<b>Expected</b>	<b>Assumed</b>
Course Geometry	Circuit with turns	Straight Line
Rolling Resistance	Dependent on local surface conditions	Constant
Clutch Engagement	Oscillatory based on load	Fixed speed
Environment	Varies with wind, temperature, humidity	Constant
Engine Bay Conditions	Changes with packaging, natural convection	Constant

The straight-line model assumption was made to both simplify development and to standardize results. Since courses have changed various times, a straight line model eliminates the variation between courses in order to isolate trends with vehicle design instead of course design.

Acquiring the rolling resistance under competition conditions may not be possible for the team, since surface quality varies greatly. Additionally, this average would change if the vehicle path differed between laps. Because of this uncertainty, rolling resistance was the chosen scaling factor to obtain the fuel economy figure seen during competition the previous year.

While the nature of the centrifugal clutch may cause its speed and torque output to oscillate during launch conditions, a constant slipping speed was chosen based on rough values of acceleration times provided by the team. The development of the clutch model is outlined in Appendix 5.

### 5.3 | MODEL STRUCTURE

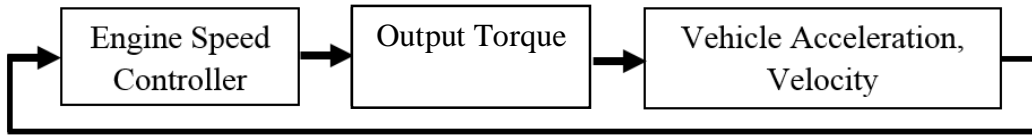


Figure 62. Simplified Simulink model structure

The figure above shows the general operating scheme of the simulation. Velocity was chosen as the closed-loop variable since it is easily related back to engine speed through the final gear ratio and wheel radius. These primary sections of the vehicle simulation are discussed in the following section, but a more comprehensive look at model development is available in Appendix 5.

#### 5.3.1 | Engine Speed Controller

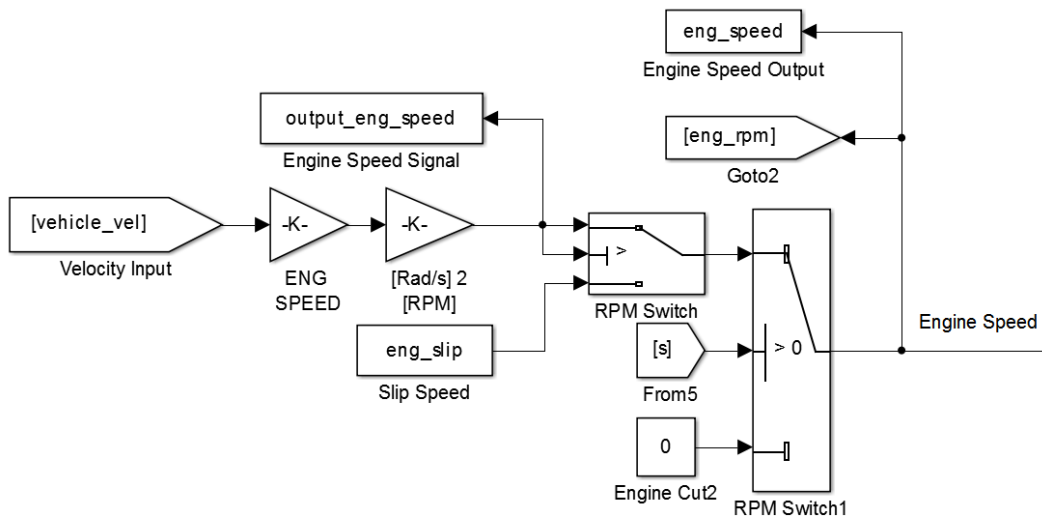


Figure 63. Engine speed controller in Simulink

From the vehicle velocity, the engine speed signal is calculated and converted to RPM. The logic of the first switch is as follows:

- If the calculated engine speed is greater than the designated slip speed of the clutch, then the calculated engine speed is used.
- If the calculated engine speed is less than the designated slip speed of the clutch, then the clutch slip speed is used.

Simply put, this introduces a relative motion between the engine input and the clutch output to simulate clutch slippage. If the vehicle is traveling at a speed below the stall speed of the clutch, then the constant slipping speed is used. Once the vehicle is traveling at a speed that results in an engine speed faster than the slip speed of the clutch, then the calculated engine speed is used.

After the engine speed is instantaneously chosen, the signal enters a simple switch that controls the engine cut condition. The engine burn/coast switching variable 's' allows the engine speed to be calculated from the vehicle velocity during a burn, but switches the engine speed to zero when the vehicle is coasting.

Torque tables that are input into the simulation use zero speed, zero torque initial values. Switching the engine speed to zero for coasting simulates an engine cut by forcing the engine to produce zero torque.

### 5.3.2 | Output Torque Calculation

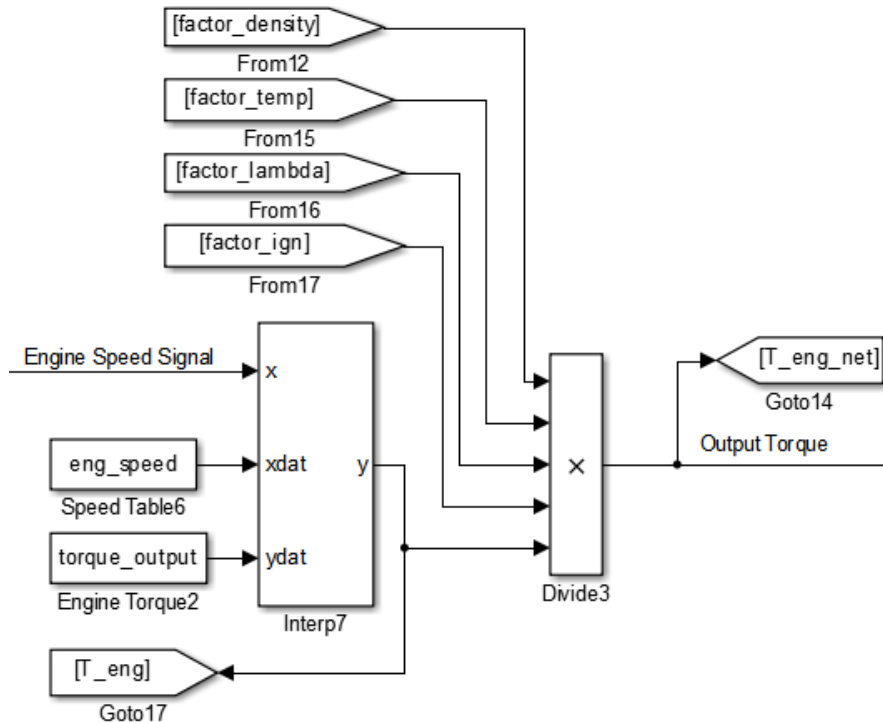


Figure 64. Output torque calculation in Simulink

In order to calculate output torque values, the engine speed signal described in the previous section is input into a linear interpolation table available in Simulink. From the engine speed and torque array, the simulation can calculate the output torque based on any engine speed. Keep in mind that the output torque is being used, instead of the raw engine torque. The output torque is the curve that incorporates the clutch engagement behavior at lower RPM.

With the torque output value now available, modifiers are applied. These modifiers are all based off experimental trends that are normalized to act as multipliers. For example, if  $\lambda = 0.8$  at that particular engine speed, then lambda trend in Figure 56 shows that torque output is 0.99 of its normalized maximum

value. This factor is then adjusted based on desired simulation lambda values, then used as 'factor\_lambda'. The adjustment of this factor is described in detail in Appendix 5.

The same procedure is used for all other factors except 'factor\_density'. This factor is an adjustment for environmental conditions, where air density in competition conditions is divided by density during engine testing conditions. This factor remains constant for the entirety of the simulation.

Note that the factors are applied to the output torque instead of directly to the engine torque. The following figure explains why that may introduce a small source of inaccuracy.

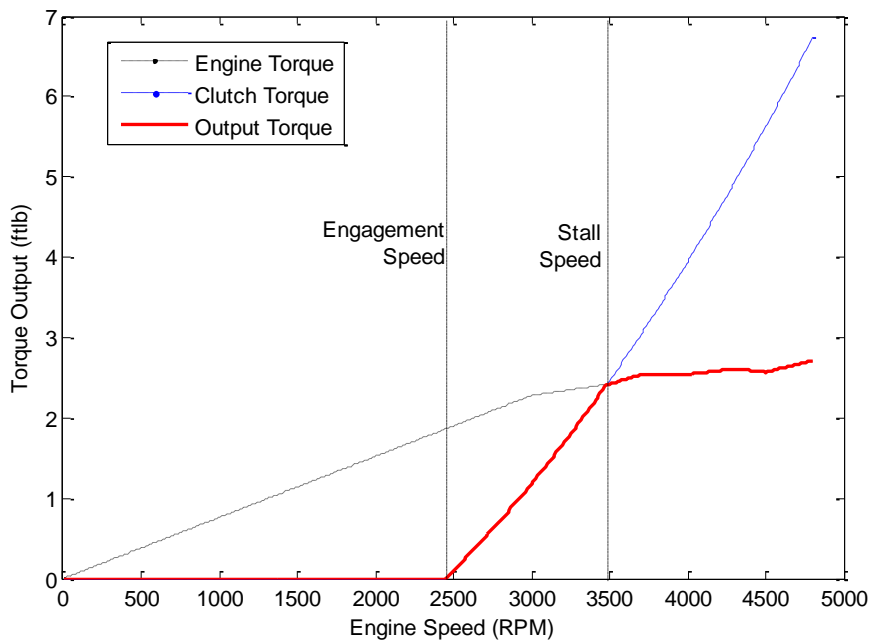


Figure 65. Centrifugal clutch model



The figure above shows the engine, clutch, and total output torque characteristic of the centrifugal clutch with increasing engine speed (further discussed in Appendix 5). Since all the factors directly affect the combustion, they should be applied to the engine torque (black dotted line) instead of the output torque (red). In attempting to do that, however, the model became unstable and a simple solution could not be found.

By applying the factors to the output torque instead, the stall speed of the clutch is slightly skewed. Although the factors are inherently small with small variations from ideal lambda, ignition, temperature and density values, it could have an effect if extreme variations were to be analyzed. During this study, however, only values close to ideal were used, therefore causing the variation in clutch stall speed to be negligible. Additionally, the clutch model may have uncertainties involved that make analyzing small deviations from stall speed futile.

### 5.3.3 | Vehicle Dynamics Calculation

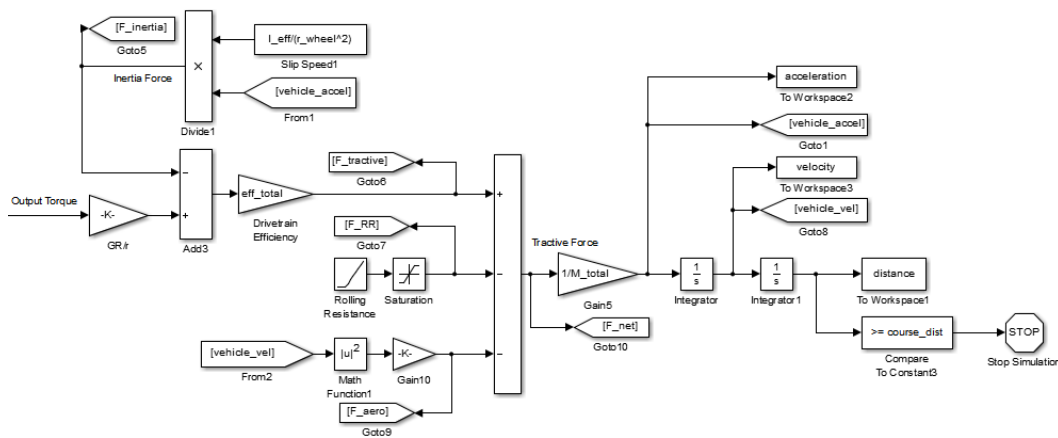


Figure 66. Vehicle dynamics calculation in Simulink

Previously calculated output torque is now multiplied by the final (and only) gear ratio, and divided by the radius of the wheel to acquire tractive force. This force represents the longitudinal force applied at the rear tire of the vehicle if no drag forces are involved.

From there, inertia forces are calculated with the vehicle's acceleration. Then, as further discussed in Appendix 5, drivetrain efficiency is applied to both engine and inertial forces.

Aerodynamic and rolling resistance drag are now applied by subtracting them from the engine and inertial forces, resulting in the *net* tractive force available at the wheel. Keep in mind that the engine is the only positive force during the simulation (besides inertial effects during deceleration), causing the vehicle to accelerate. The others are negative/drag forces, causing deceleration.

Dividing this tractive force by mass allows the vehicle acceleration to be calculated. Integrating that once results in vehicle velocity, and integrating once more gives vehicle distance. As previously mentioned, the vehicle velocity as the feedback variable in the simulation, while the distance is used to end the simulation once the total course distance has been traveled (6 miles during this study).

## 5.4 | PRELIMINARY VERIFICATION

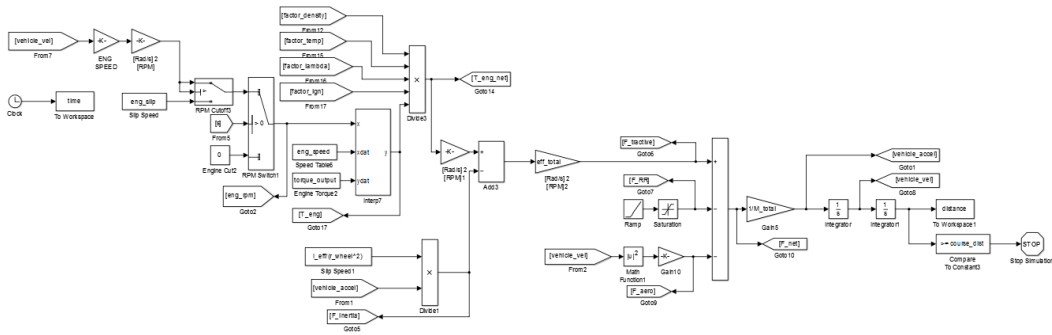


Figure 67. Final Simulink portion of vehicle model

In this section, components of the simulation are studied to verify expected behavior. Tools for simulation-based vehicle design are also introduced through 2-D trend analyses.

### 5.4.1 | Convergence

Before simulation data is taken, a convergence study was performed to choose a solver and step size. A variable step solver was chosen for step size flexibility and to decrease necessary run times [12]. ODE45, ODE23, and ODE113 were compared in the following figures.

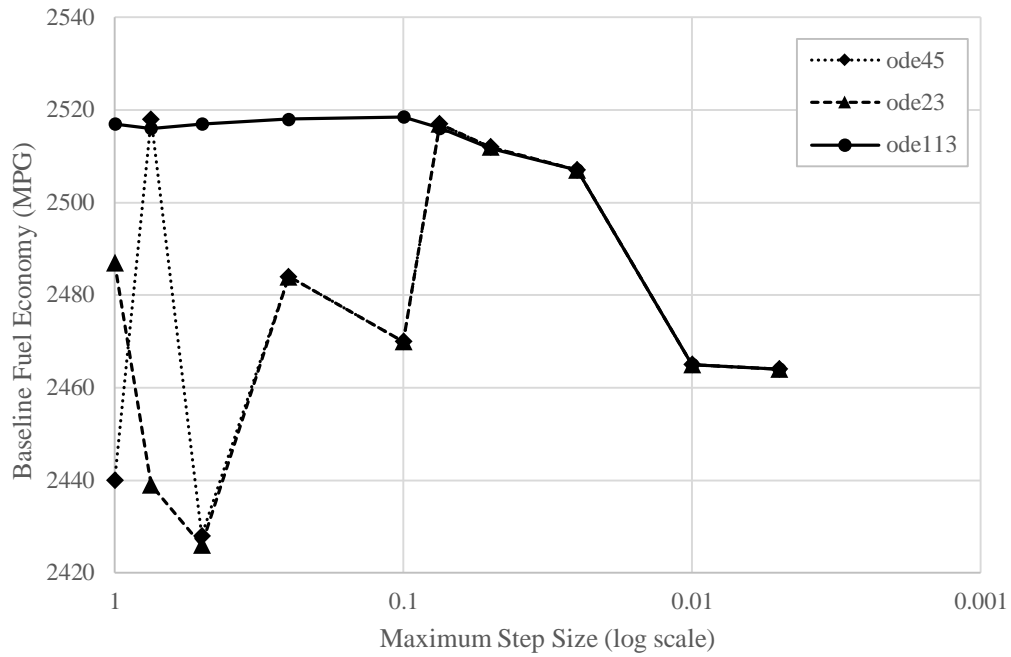


Figure 68. Simulation convergence with maximum step size

From the figure above, all three solvers achieved similar results at a step size of 0.75, while they all converged on a value of 2465 MPG at 0.01 maximum step size.

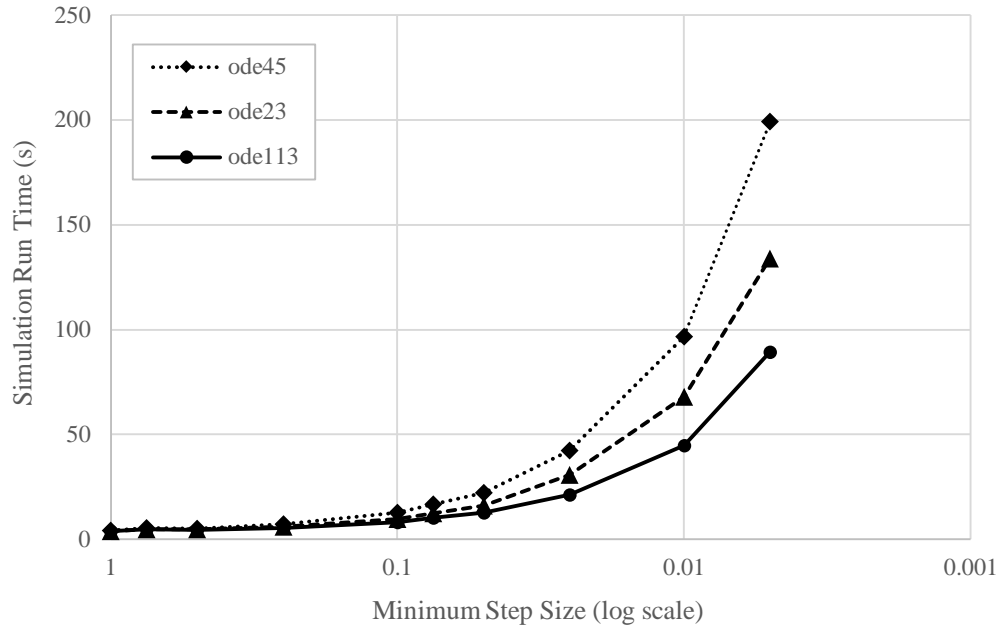


Figure 69. Run time variation with different solvers

Since all solvers return nearly identical values below a maximum step size of 0.75 and start converging at 0.01, 0.01 was chosen as the maximum step size and ode113 as the solver to minimize run times. The relative tolerance was held constant at  $1E-4$ , implying that each computed state is accurate to within 0.01% [12].

#### 5.4.2 | Vehicle Dynamics

Since quantitative vehicle data was not available at the time of this study, the verification of vehicle dynamics are based on logical behaviors and trends that the team lead observed during competition conditions.

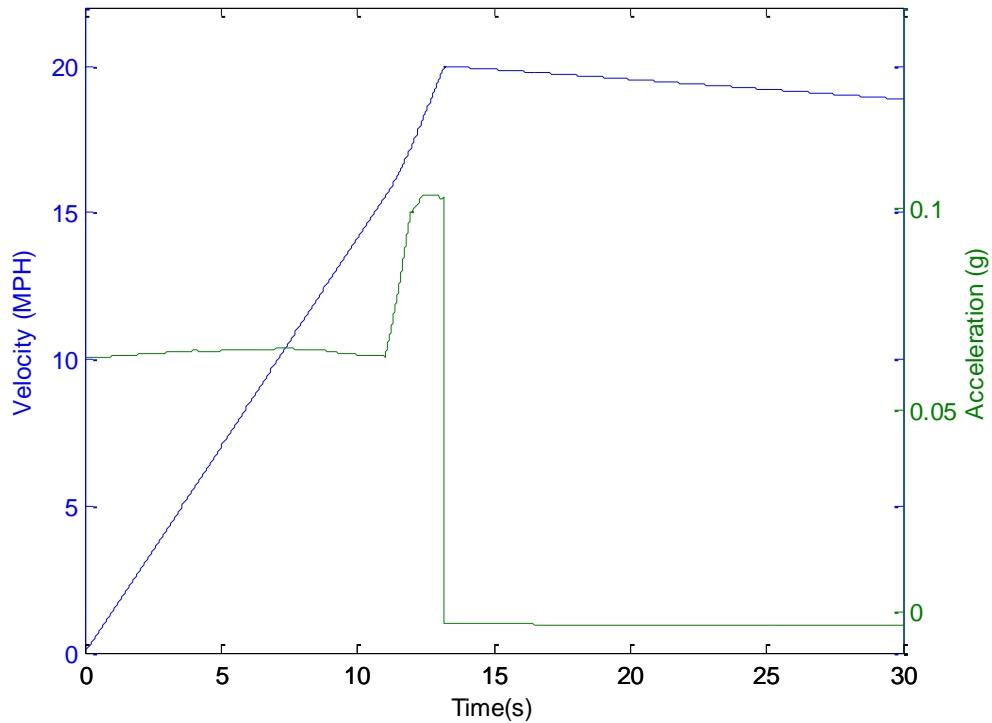


Figure 70. Vehicle dynamics during launch conditions

The figure above is mainly a representation of clutch characteristics on vehicle that were previously analyzed. With the constant slip speed assumption for the clutch, acceleration remains relatively constant, causing linearly increasing velocity. The slight variation in acceleration is attributed to various vehicle loads including ramping rolling resistance and increasing aerodynamic forces. After the clutch slips enough to allow it to fully stall (~12 seconds), acceleration increases rapidly as engine speed is allowed to increase to propel the vehicle to maximum velocity. Velocity follows the trend by increasing more rapidly to the maximum velocity, when the engine is cut (~13 seconds). At that time, acceleration jumps to a negative value and velocity decreases.

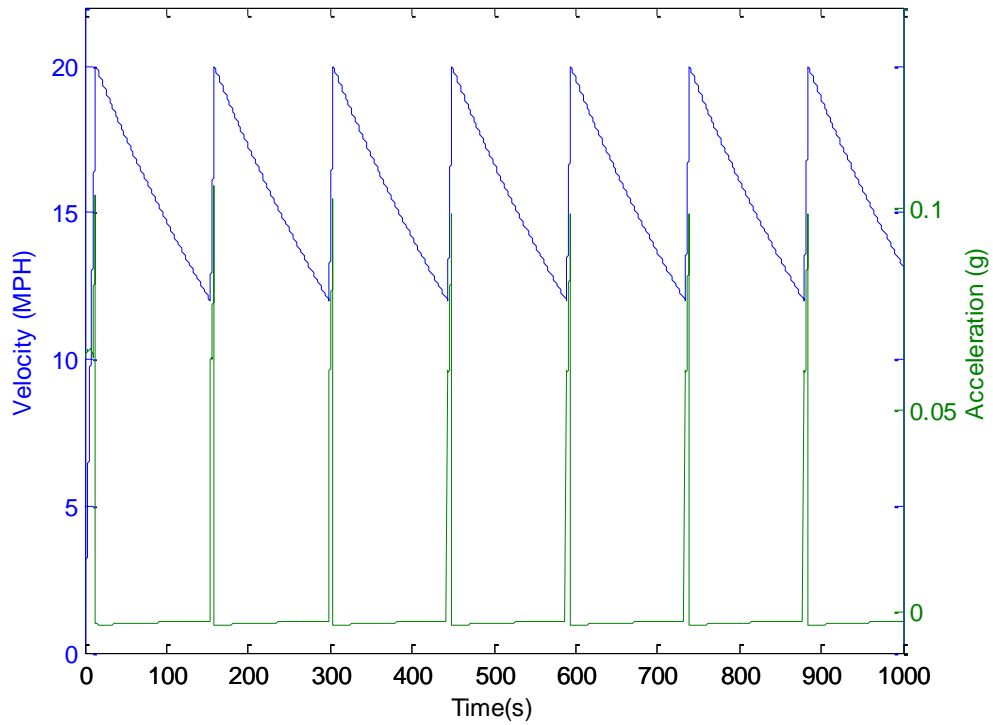


Figure 71. Large scale vehicle dynamics

In this figure, a larger time scale is studied to verify simulated driver input and resulting vehicle behavior. The saw-tooth behavior of vehicle velocity accurately reflects the burn and coast method, but the small burn to coast time ratio (BCR) causes the vehicle acceleration to appear nearly instantaneous.

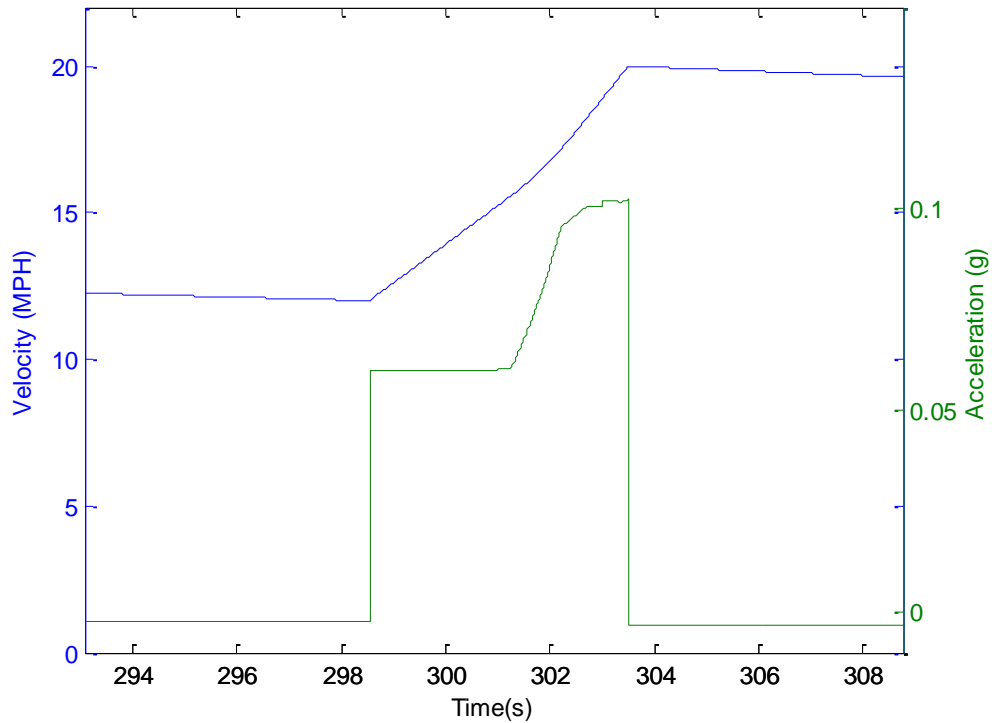


Figure 72. Intermediate burn and coast behavior

Upon reaching minimum velocity, the engine switch signal is set to burn, causing the vehicle to experience similar behavior to launch conditions except from a rolling start. The rolling start decreases the slip time, and at this point it becomes obvious that tuning the engagement speed of the clutch or vehicle speed range directly changes the amount of time the clutch is slipping. The simulation shows that roughly 50% of burn time is spent wasting heat energy through slippage.

#### 5.4.3 | Clutch Characteristic

Although the clutch was discussed in the vehicle dynamics verification, powertrain characteristics were specifically studied to verify clutch input and output behavior.



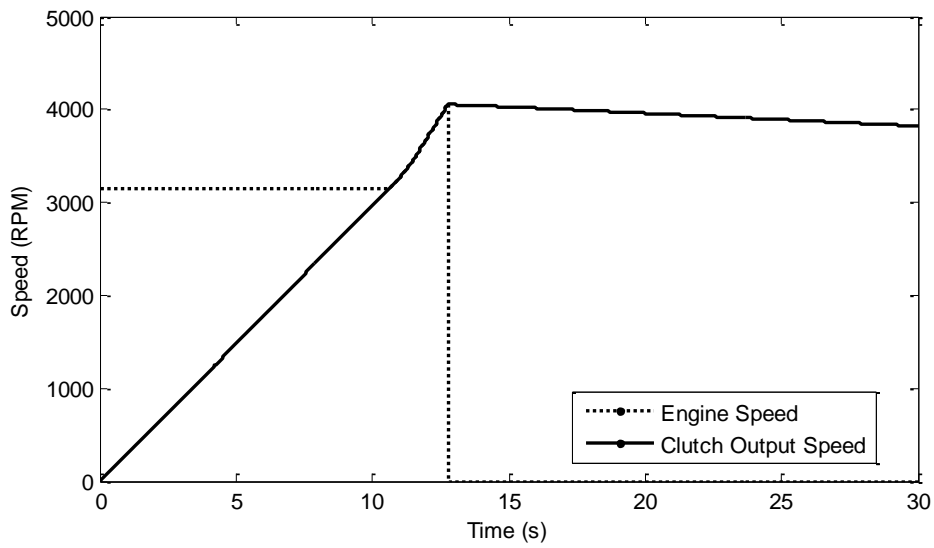


Figure 73. Engine and output speed behavior during launch

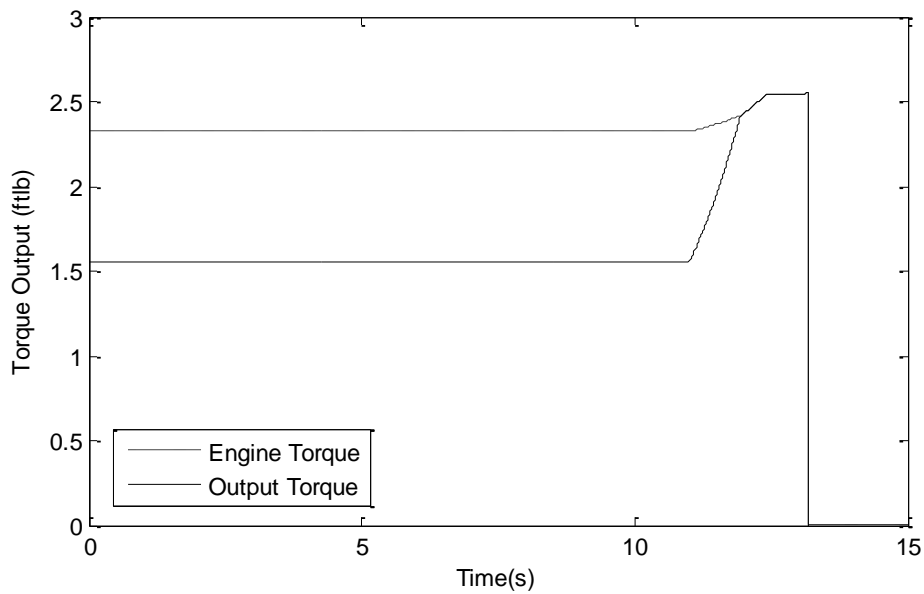


Figure 74. Torque characteristic during vehicle launch

The figures above show the powertrain behavior during launch from standstill, and also show that the clutch does indeed follow behavior expected from the assumed model. After starting, the engine operates at full throttle but is held at a constant

slip speed while the clutch speed increases linearly to its stall point. This slip speed is somewhat arbitrary and is one of the model assumptions that should be modified by the team after engine data from the competition is available. At the time of this study, the slip speed was chosen to result in an initial launch time that the team viewed as relatively accurate.

### 5.4.3 | Trend Study

Another useful exercise in verification is a trend study, in which select variables are varied and fuel economy trends are observed. Variables that should result in obvious trends were used to verify simulation behavior, while more variables with more intricate relationships are also introduced.

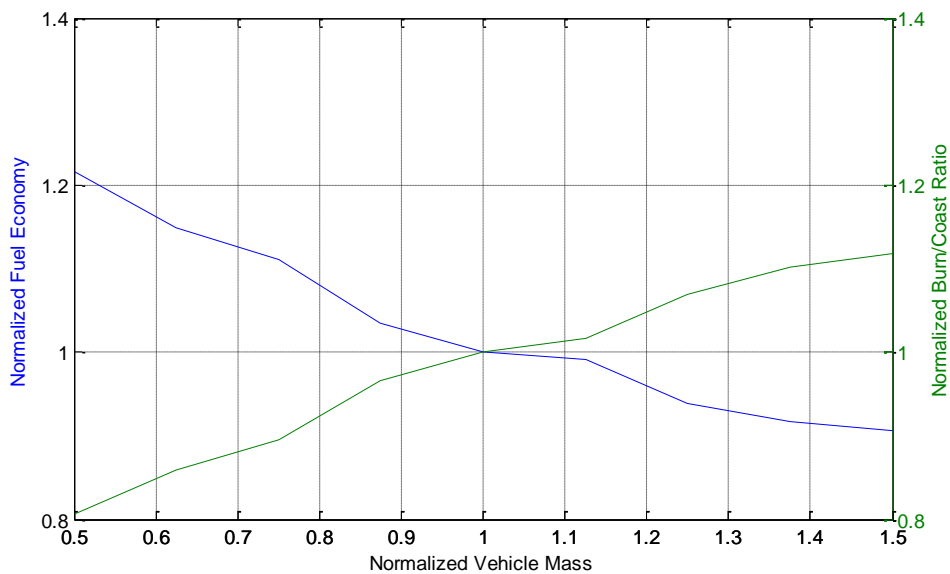


Figure 75. Vehicle mass trend study

Here, it is apparent that fuel economy decreases nearly linearly with increasing vehicle mass. To describe this relationship, the term *sensitivity* is introduced, and defined as follows:

$$S_{variable} = \left( \frac{\text{Normalized Change in Output Variable}}{\text{Normalized Change in Input Variable}} \right) \quad (8)$$

This term describes the approximated linear slope of the figure above, and may become a valuable design tool for the team. For example, the approximate sensitivity for vehicle mass is:

$$S_{v,mass} = \left( \frac{-0.07}{0.5} \right) = (-0.14) \quad (9)$$

Applied in vehicle design context, estimating the economy of decreasing vehicle mass by 5% would involve applying it to the linear equation:

$$\begin{aligned} MPG_2 &= MPG_1 \left[ 1 + S_{v,mass} \left( \frac{\% \text{ Change}}{100} \right) \right] \\ &= (1.007) MPG_1 \end{aligned} \quad (10)$$

Implying that decreasing vehicle mass by 5% results in a 0.7% improvement in overall fuel economy. For simplicity, it may be easier to think of sensitivity as percent change in fuel economy for every percent change of the input variable.

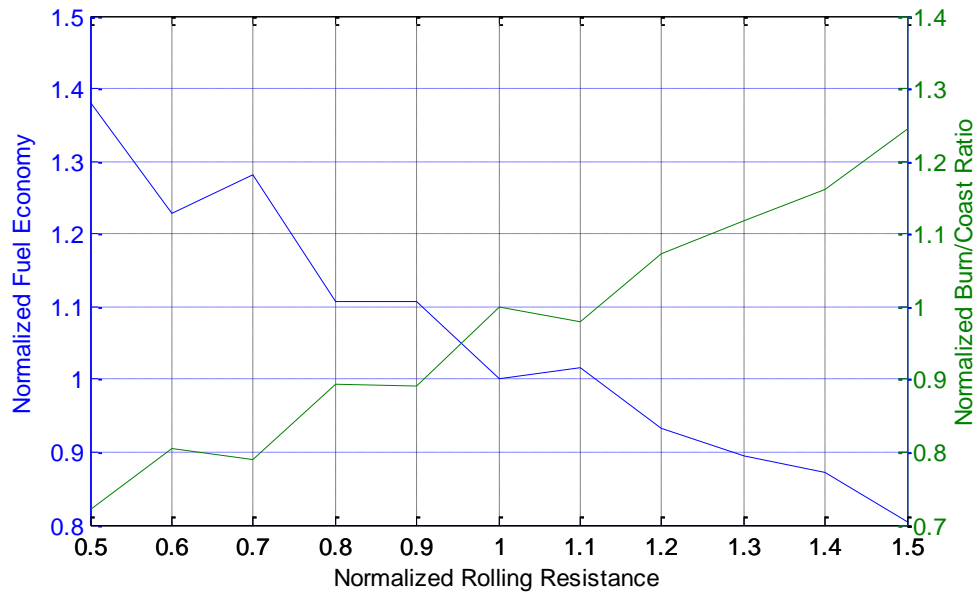


Figure 76. Rolling resistance trend study

However, for trends with obvious discontinuities such as that shown in the figure above, a linear trend could be assumed for a rough sensitivity estimate. These discontinuities occur because the number of burns required changes with large variations in rolling resistance. While the start number stays constant at low rolling resistances (0.6-0.7 of rolling resistance value), fuel economy increases with increasing rolling resistance, caused by locally decreasing BCR. Although there are a number of reasons why this may happen (higher engine temperatures, different engine operating range), it is up to the team to study specific behavior as it is outside the scope of this project.

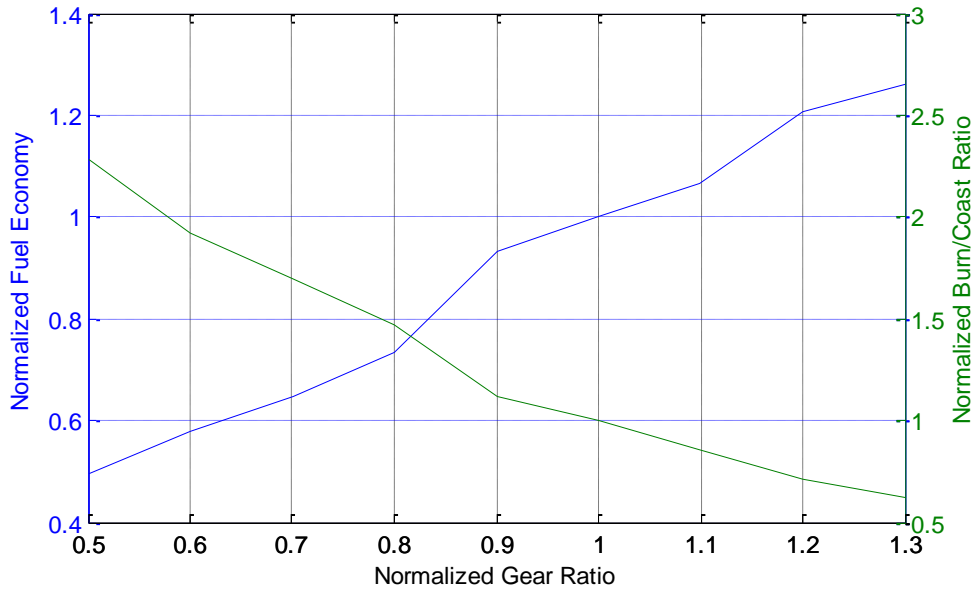


Figure 77. Gear ratio trend study

In this final trend study, it is apparent that increasing gear ratio results in greatly increased fuel economy, with a sensitivity of 0.83 – nearly 6 times more sensitive than vehicle mass.

At this point, the simulation is as accurate as possible without additional data from competition. Vehicle dynamics, torque modifiers, clutch characteristics and two-dimensional trends all follow expected behavior, so the next step is to apply a three-dimensional trend study with lambda and ignition to define target values to tune the engine for to maximize fuel economy under competition conditions.

## 6 | RESULTS, DISCUSSION

In this section, lambda and ignition targets suggested by the simulation are compared to minimum BSFC targets from previously acquired trends. After final values are selected, the engine is tuned and performance is compared with results found during benchmark testing.

### 6.1 | LAMBDA, IGNITION TARGETS

A nested for-loop in the simulation code allowed lambda and ignition values to be iterated through a two dimensional matrix. Lambda was varied from 0.8 to 1.2, and ignition varied from -4 deg. from MBT to 4 degrees after MBT.

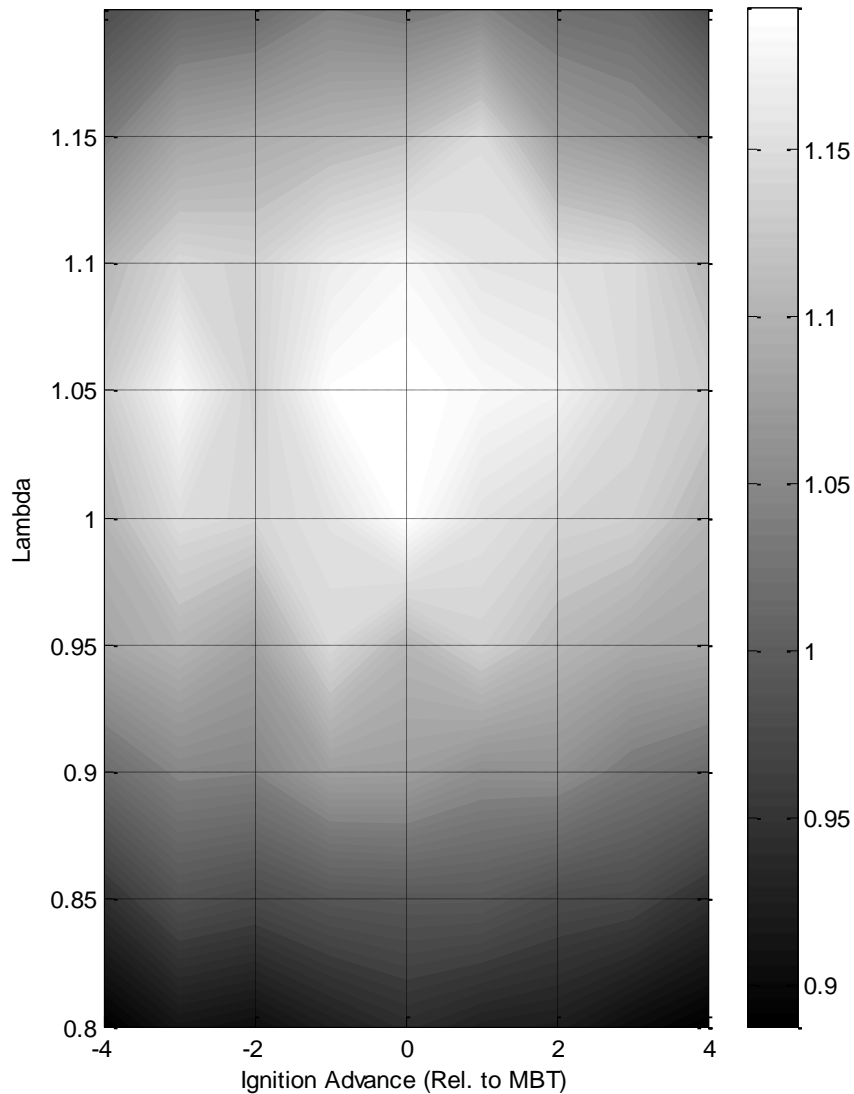


Figure 78. Normalized fuel economy with lambda and ignition variation

The white highlights in the figure above represent increased overall fuel economy. Reflecting familiar trends, the lambda variation has a much larger effect on overall performance, with an expected 12% decrease at  $\lambda = 0.8$  to a 17% improvement at  $\lambda = 1.05$  at MBT. The lambda and ignition targets that the simulation estimates to result in highest fuel economy are summarized in the following table.

Table 4. Simulation-based lambda and ignition targets

Component	Value
Lambda	1.05
Ignition (degrees from MBT)	0
Estimated MPG Improvement (%)	19.7

Although MBT timing was expected to maximize fuel economy, the simulation estimates that maximum improvement occurs when the fuel mixture 5% richer than the  $\lambda = 1.1$  seen in Figure 56.

Since the simulated lambda target lies between minimum BSFC and maximum torque values, it was hypothesized that the simulation found the optimum fuel efficiency to be a lambda value that resulted in the best balance between these points. To further investigate this, normalized values shown in Figure 56 were plotted together as percent loss and shown in the following figure.



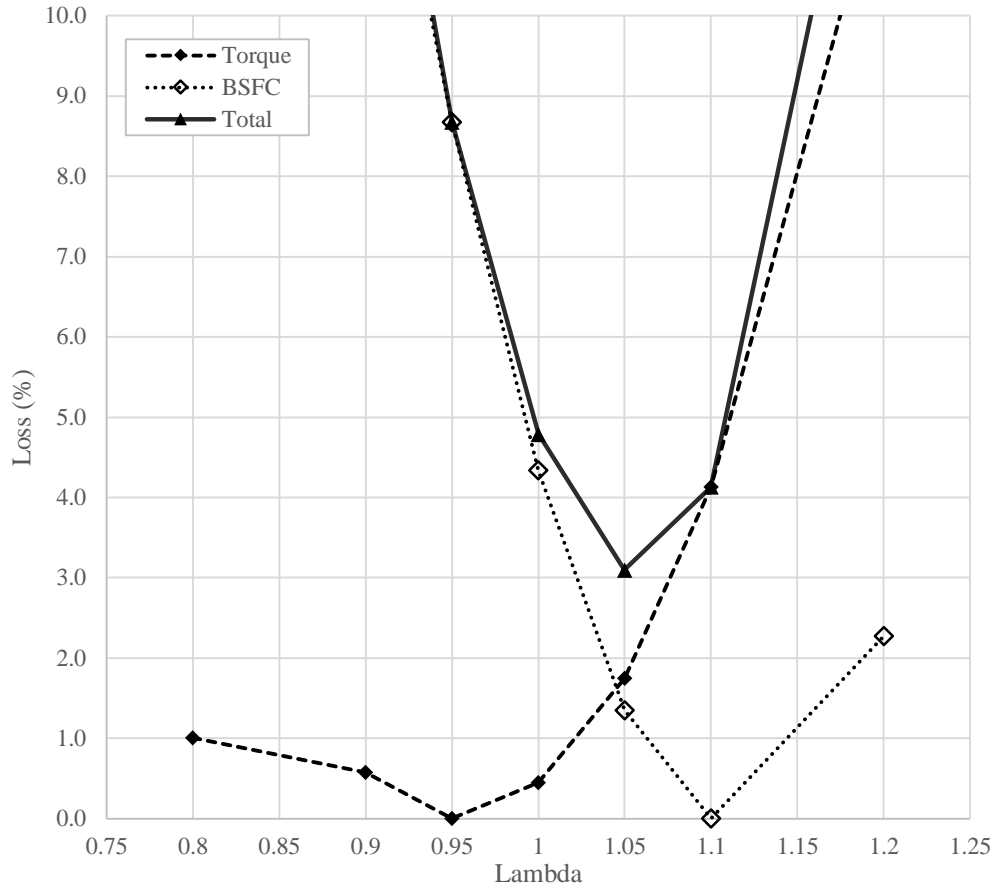


Figure 79. BSFC and torque loss with lambda variation

The loss of both BSFC and torque are added to find total loss between the two parameters. The minimum total loss is achieved at  $\lambda = 1.05$ , the same value that the simulation estimates to maximize fuel economy. Although this stands as a theory, it is believed that minimizing total loss between torque output and BSFC is the core trend of fuel economy during competition conditions, which differs from  $\lambda = 1.1$  condition that minimizes BSFC at steady state.

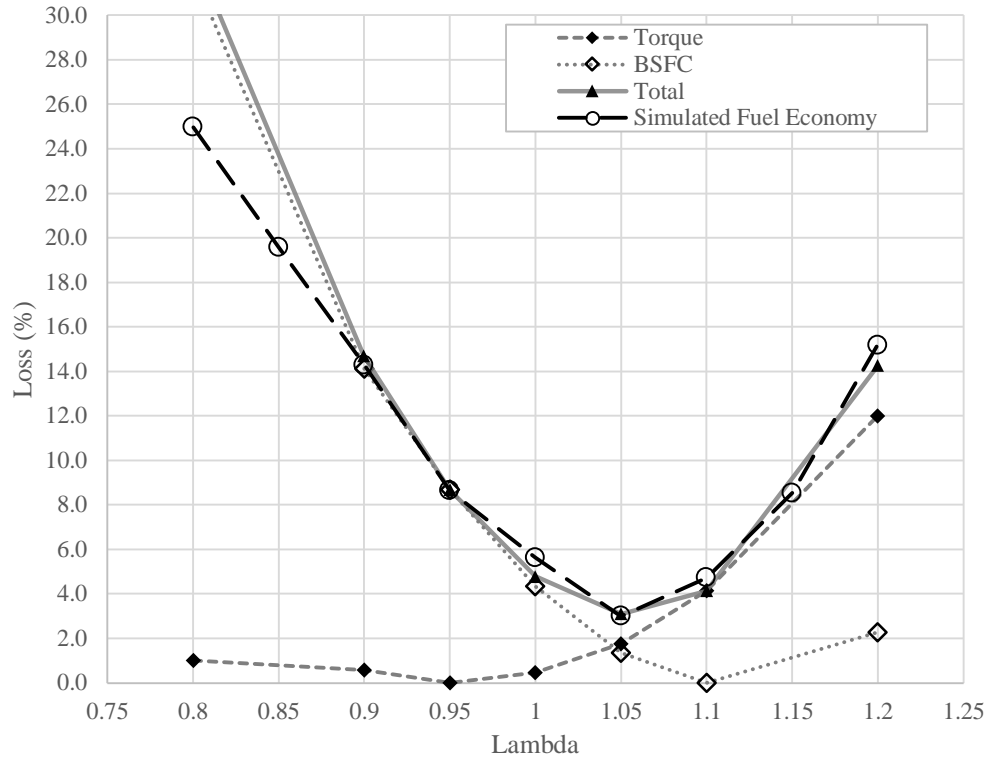


Figure 80. Loss comparison with simulated fuel economy trend

The figure above is identical to the previous figure with the addition of normalized simulated fuel economy variation with lambda and fitted to the total loss curve (assuming 3% loss  $\lambda = 1.05$ ). This further supports the hypothesis that the total loss curve between torque and BSFC determines the maximum overall fuel efficiency instead of minimizing BSFC during steady state testing. Additionally, the total loss curve shown above is also believed to reflect the overall fuel economy trend with variations in lambda. Though the simulation follows the total loss trend closely at low variations from  $\lambda = 1.05$ , increasing discrepancy farther away are suspected to be due to the total loss curve being obtained only for lambda variation at one operating speed, while the simulation accounts for relationships that may exist between and play a larger role with more extreme lambda values.

This theory played a partial role in choosing  $\lambda = 1.05$  and  $\Phi = 0^\circ$ . Another reason for selecting  $\lambda = 1.05$  was because the torque and BSFC trend with varying lambda is suspected to vary between different operating speeds for many possible reasons, including unique flow characteristics and varying engine friction characteristics. However, it is well established in literature that operating at a slightly lean mixture will result in lowest BSFC at every operating speed, which suggests  $\lambda = 1.05$  as an estimated average value that will come close to minimizing BSFC at all points. Choosing MBT ignition timing was rather straightforward since it minimizes BSFC and torque output, as illustrated in Figure 57. Torque variation with ignition timing (4000 RPM ,  $\lambda=1.0$ )

## 6.2 | ENGINE TUNE

With ignition and lambda targets defined, the engine was tuned at the same speeds studied during benchmark testing. In the following figure, torque, power, and BSFC are compared between the baseline values, results from the tuned engine, and simulation predictions based on interpolation of the fixed torque and BSFC trends observed at 4000 RPM during benchmark testing.

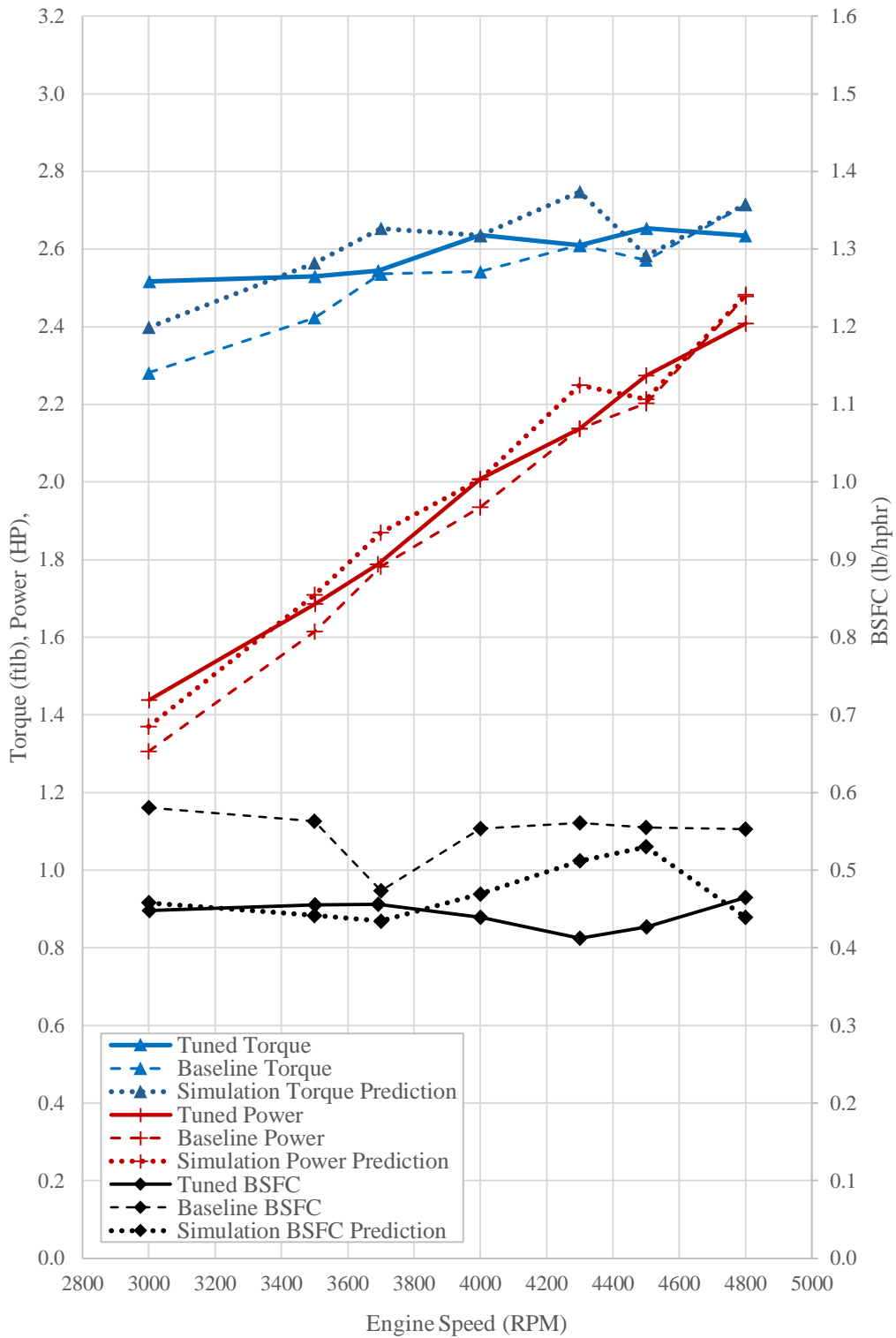


Figure 81. Torque, Power, BSFC comparison with baseline and simulation

Although torque is decreased by 3% at 4800 RPM, improvements of up to 10% at 3000 RPM are seen across the rest of the operating range. These improvements directly relate to the trend studied in Figure 56. More importantly, it is apparent that BSFC values are drastically reduced across the operating range, with the exception of 3700 RPM.

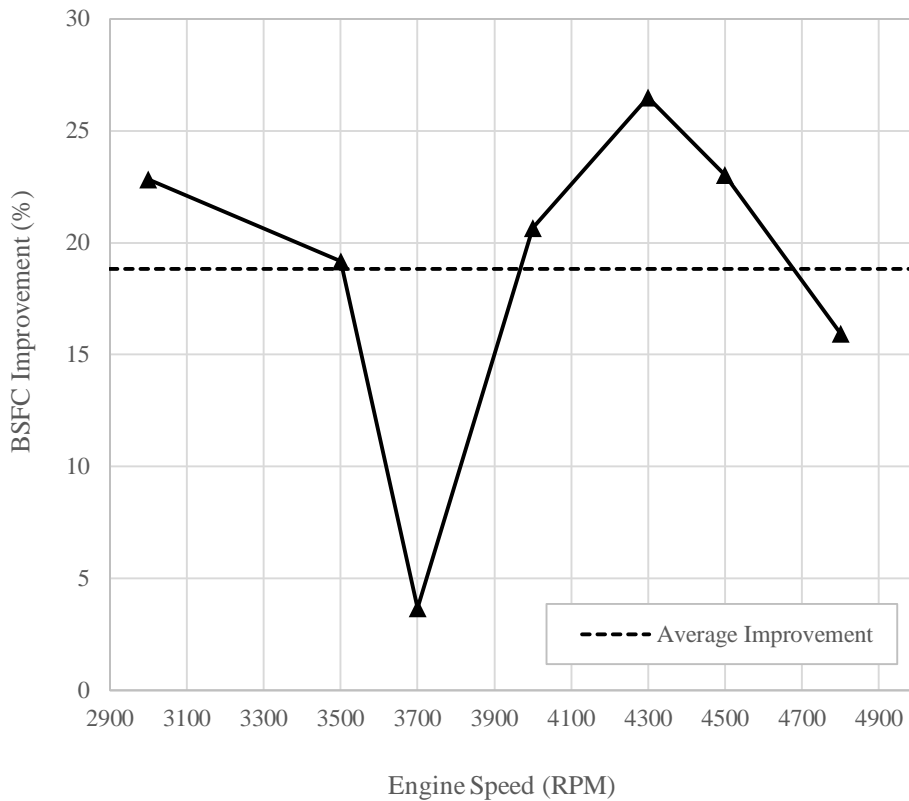


Figure 82. BSFC improvement with target lambda and ignition values

BSFC sees nearly an 18.8% average improvement (reduced BSFC), and 4300 RPM sees the greatest BSFC improvement of 26.5%. However, the especially large difference between simulated and actual BSFC improvement at that speed reveals several important conclusions.

First, the trend seen in Figure 56 mostly likely does not accurately represent trends at all operating speeds, implying that BSFC varies with lambda more significantly than the experimental trend at 4000 RPM. The AFR from the baseline tune was merely 5 percent richer than the target at 4300 RPM, theoretically suggesting a 4% improvement in BSFC if  $\lambda = 1.05$  is achieved instead of the  $\lambda = 1.0$  baseline value.

To relate the improvements in engine performance to overall vehicle fuel economy in the competition, the tuned torque and BSFC points were entered in the simulation and compared to the baseline value. Interestingly, the estimated overall fuel economy improvement was very similar to the average BSFC improvement seen at steady state.

Table 5. Simulated fuel economy difference with tuned engine

Baseline Engine Simulation	1334 MPG
Tuned Engine Simulation	1569 MPG
Estimated Improvement	17.5 %

## 7 | RECOMMENDATIONS

This section summarizes the potential shortcomings of this study, suggests improvements for the engine and support systems, and summarizes a quick exercise in simulation-supported vehicle design.

### 7.1 | ENGINE TUNING

After benchmark testing, it was apparent that AFR had more than 10 times the effect on BSFC than ignition, and 27 times more than temperature variation effects. Additionally, Figure 58 showed that over a 90 °F engine temperature range, mass flow rate of fuel needed to decrease by 5% in order to maintain a constant lambda value due to decreased volumetric efficiency at higher temperatures. With these two trends in mind, it can be concluded that holding the mass flow rate of fuel constant throughout a range of engine temperatures will cause a significant increase in BSFC, so temperature compensation to account for varying volumetric efficiency is necessary to minimize BSFC in all operating conditions.

The test procedure for acquiring this temperature compensation curve would involve choosing a constant temperature to test at and reaching the desired lambda value during steady state at that point. Then, steady state should be achieved at different temperatures, and the target lambda achieved once again by *only varying temperature compensation values*.

Another concern involved with observing trends is the assumption that the trends observed at one operating point apply to others. For the ‘simulation’ data in Figure 81. Torque, Power, BSFC comparison with baseline and simulation, one trend was interpolated at all data points to find the estimated change in

performance. The large difference between the simulated and experimental data, particularly in the 4000-4600 RPM range, suggest that the trend is not an accurate prediction at these points.

This observation introduces some complications for testing, as the experiments conducted during benchmark testing would have to be repeated for each operating point in order to maximize fuel efficiency at all speeds. Although temperature compensation curves should be constant across all speeds due to the linear nature of decreasing volumetric efficiency, it is up to the team to verify this hypothesis.

It is also recommended that the team to address the artificial engine speed limitation imposed by either a hardware or software error involving the crank sensor. Both baseline and tuned performance curves show both torque and power increasing up 4800RPM, or the artificial cutoff point. Solving this issue would allow further engine testing to be conducted to explore more possible ranges of reduced BSFC. At this point, it is thought to be a software issue since the engine speed operates normally until 5000 RPM, where the engine stops increasing speed in a very similar fashion to a redline-induced fuel and/or ignition cut.

## 7.2 | DRIVETRAIN

Although the existing centrifugal clutch and chain drive system were initially used to power the dynamometer, two serious issues necessitated the move to a direct system.



The first issue was the inefficiencies introduced by the clutch and chain, which are compared to the direct drive system in the figure below.

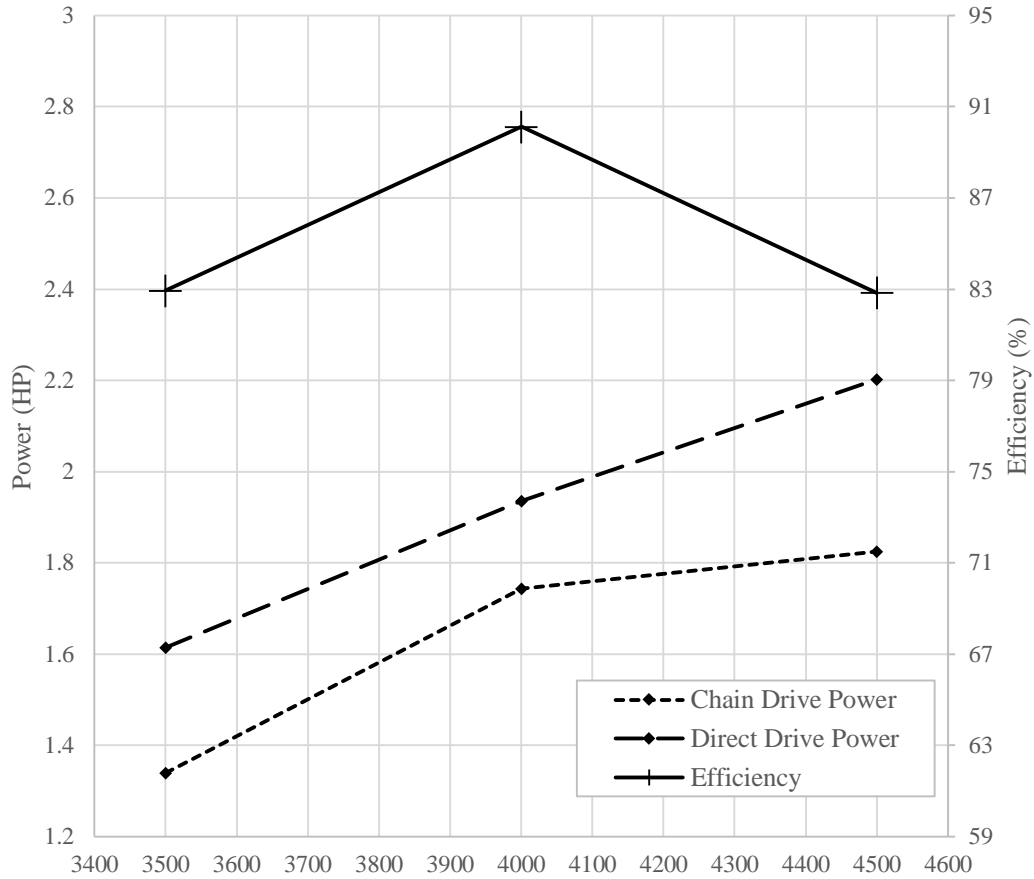


Figure 83. Drive system comparison

The figure above shows that the chain drive (Iteration 1) was 85% efficient on average, which in turn skewed BSFC numbers by the same percentage. Since drivetrain efficiency is directly proportional to the torque produced by the engine, it is important for the team to aim to improve this, perhaps by considering a direct drive system.

The second issue was a combination of the imbalance associated with the centrifugal clutch pads and the transverse loading condition caused by the chain

that caused 4 crankshaft failures throughout testing. The clutch is attached to the end of the output side of the crankshaft, shown below.

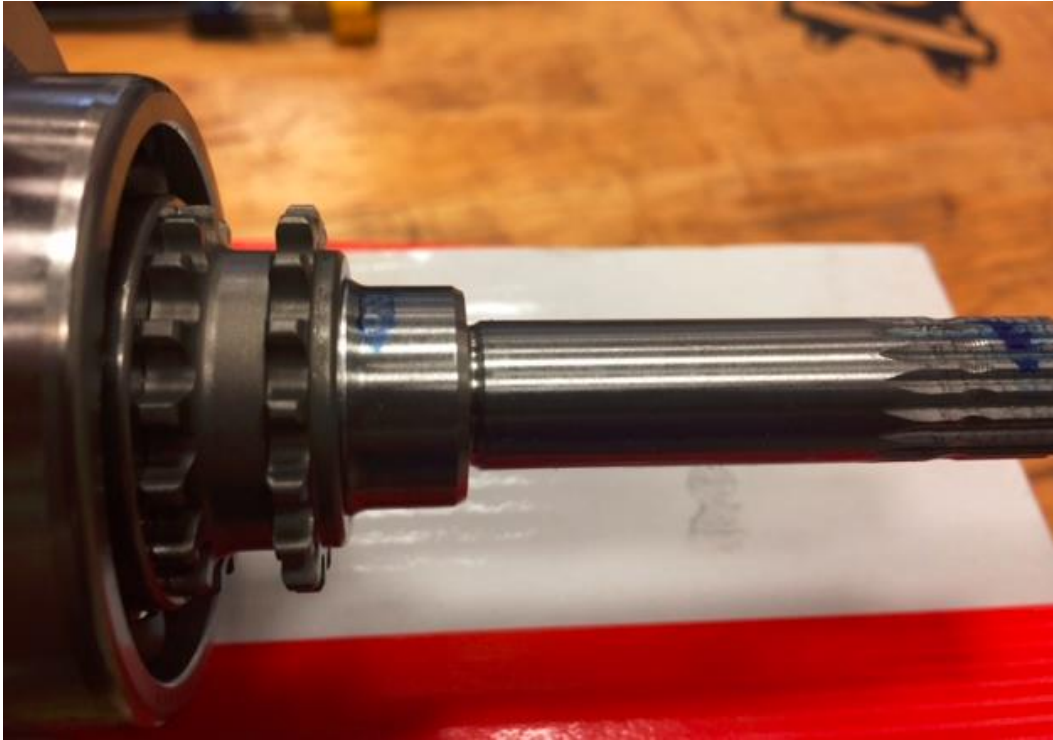


Figure 84. Output side of crankshaft

The bearing support on the output side is located just outside of the timing sprocket, followed by a long, constant diameter shaft with an undercut feature between them. The undercut feature was the location of all crankshaft failures, which is believed to be aggravated by the cantilever loading condition.

It is peculiar that the team did not see any similar failures during competition, but since the dynamometer used the same drive system, it is an important issue for the team to resolve to prevent future failures. Unfortunately, a single cause for this failure is unknown at this time, but reducing the cantilever load and unbalanced mass will most likely prevent a similar failure event.

### 7.3 | SIMULATION DEVELOPMENT

As previously mentioned, verification of the simulation is necessary to rely on its absolute accuracy for design decisions. The following variables should be acquired to improve the simulation:

- More accurate Coefficient of Drag
- Clutch slip speed
- Overall drivetrain efficiencies
- Rolling resistance across various surfaces
- Engine bay environment constants
- Environmental conditions (temperature, pressure, wind speed)

Additionally, future iterations could involve integrating accurate course maps with direction changes and associated drag forces involved.

Although most of the variables in the simulation are as accurate as possible at the time of this study, the baseline simulation still needed to be scaled to achieve the 1300 MPG number the team saw during last year's competition. Even with the lambda and ignition trends applied, it is believed that the initial overestimation were caused mostly by dynamic volumetric efficiency conditions not being accounted for during engine testing and extreme variations in rolling resistance values most likely seen during competition.

Since the tire supplier provides a rolling resistance value of 0.002 taken at constant, smooth conditions, it chosen as the scaling factor since its uncertainty was the largest. The simulation results in this study reflect an average rolling resistance

value of 0.007, 3.5 times higher than the manufacturer’s specification. With competition held on public roads with sections of rough areas, including cobblestones, it may not be far off from an actual average.

Similar baseline results were important because burn times vary greatly with varying drag forces acting on the vehicle. If burn-coast-ratios were relatively small, improvements in BSFC would not be accurately represented.

#### 7.4 | VEHICLE OPTIMIZATION

As an extra exercise beyond the scope of this project, a shallow vehicle optimization was performed using general trends seen throughout this study. Modifications to the baseline vehicle model are outlined in the following table.

Table 6. Summary of example modifications and improvements

Modification Order	Variable	Baseline Value	New Value	Projected Fuel Economy
0	BASELINE			1334 MPG
1	Engine Tune	Base	Tuned	1568 MPG
2	Gear Ratio	11.4	13.5	1890 MPG
3	Clutch Pad Mass	.118 kg	.160 kg	2137 MPG
4	Drivetrain Efficiency	0.85	0.9	2240 MPG
<b>Overall Improvement</b>				<b>+ 68 %</b>

Since one of the largest sensitivities seen was gear ratio, it was increased to match maximum engine speed to maximum vehicle velocity speed through the following equation:

$$GR_{max} = r_w \left( \frac{\pi}{30} \right) \left( \frac{RPM_{max}}{V_{max}} \right) \quad (11)$$

$GR_{max}$  = Maximum Gear Ratio

$r_w$  = Wheel Radius (m)

$RPM_{max}$  = Maximum Engine Speed  
(RPM)

$V_{max}$  = Maximum Velocity (m/s)

This engine has a relatively flat torque curve, so increasing gear ratio also increases tractive force at the tire, thereby increasing acceleration and decreasing burn times. It is important to note that this relationship would be negated once engine torque decreased at a faster rate than tractive effort increased.

Increasing the mass of the clutch pads effectively decreases the time the clutch is slipping, since the normal force increases exponentially with linearly increasing engine speed. Although these numbers should be verified under competition conditions, it is valuable to see expected results relative to the baseline simulation.

## 8 | CONCLUSION

Due to the multiple components involved in this study, this section will outline conclusions made for dynamometer setup/testing and experimental results. Final conclusions from the simulation are avoided to prevent potential misdirection due to the remaining uncertainties in the simulation.

### 8.1 | DYNAMOMETER

The dynamometer system worked most accurately by minimizing the degrees of freedom between the engine and motor through the use of direct drive. Along with vastly improved repeatability, the direct drive system allowed accurate engine data acquisition without the losses involved with the centrifugal clutch and chain drive, which caused an average 15% power loss throughout the operating range. Constant power loss was assumed during the earlier stages of testing, but Figure 83 proves that assumption to be inaccurate with efficiency varying by 9% across measured speeds. If the final system included the chain drive with this assumption, accurate engine characterization would have been impossible.

Due to the flexible nature of the chain, the vibrations associated with power application exacerbated fluctuations in engine speed by nearly 300% more than the direct drive, causing variations in output parameters that decreased the reliability of steady state data. Although various methods for alleviating the vibration issue did generally increase the quality of the data, the lack of crankshaft failures during direct drive testing suggests that the pre-existing chain drive with a centrifugal clutch was very detrimental to the supporting components.

Even with the direct drive system, relative motion between couplings caused accelerated degradation of the elastomer damper, necessitating frequent lubrication to maintain longevity. The alignment also played a large role in the longevity of the direct drive system, but the process was more exact than the process involved in the chain drive system. Combined with relatively large distance between drive axes and lack of appropriate tools and more precise measurements, the chain drive introduced more opportunity for misalignment, potentially decreasing repeatability significantly between setups.

## 8.2 | EXPERIMENTAL PERFORMANCE

The trend analyses in this study reflected both expected and unexpected behavior. The lambda trend showed that  $\lambda = 1.1$  achieved a minimum BSFC, which understandably varied from Heywood's prediction of  $\lambda = 1.05$  due to different engine variables between his and this study. A more universal trend of minimum BSFC and maximum torque at  $\Phi = 0^\circ$ , or MBT timing, was verified multiple times with the exception of an outlying data point at  $\Phi = -2^\circ$ . The cause of this data point was hypothesized to be hardware related, but further investigation is necessary.

The temperature variation experiment revealed interesting and crucial information for improving overall vehicle performance. Most likely under the assumption that increasing engine temperature decreases BSFC, even to extreme values, competing teams generally design towards heavily insulated systems to minimize heat lost to surroundings. Figure 58 shows decreasing engine temperature indeed decreases BSFC, but at some point (170°F in this case) other

factors cause the BSFC to rise again. Since BSFC is directly proportional to torque and fuel flow at a constant engine speed, the parabolic BSFC behavior is attributed to variations in rates of change of both of these variables. Although the fuel flow trend directly reflects varying volumetric efficiency, more tests involving motoring friction at various engine temperatures are required to confidently conclude why the minimum BSFC point exists at that point.

Perhaps more important than the minimum BSFC observation during the temperature study is the relationship between fuel flow during this test and the lambda trend. As previously stated, a nearly 10% decrease in volumetric efficiency across the tested range would require a similar decrease in fuel mass delivered to maintain a constant AFR. Since the previous engine tune did not incorporate a compensation for this, the AFR varied by a significant amount. In this study, a constant fuel delivery (set at 120 °F, 4000 RPM) would cause a 10% rich condition at 210°F, causing BSFC to increase by nearly 32% in addition to the 1.2% increase caused by component temperatures. From this, it is obvious that temperature compensation for varying volumetric efficiency is far more beneficial to decreasing fuel consumption than controlling engine temperature. After the compensations are programmed, however, controlling engine temperature could potentially decrease BSFC by a further 1.2%.

After final ignition and AFR targets were chosen, the full tune on the engine resulted in a 19% average BSFC improvement/decrease. Although most ignition values were retarded nearly 6°, the largest improvements were caused by the change in AFR. In Figure 81. Torque, Power, BSFC comparison with baseline and



simulation, the trends observed at 4000RPM were extrapolated to estimate the performance improvement (simulation lines). Large variations between predicted and actual values at certain speeds suggest that the benchmark trends are different at each engine speed, with differences being more severe in in the 4100-4600 RPM range. Further trend studies at these speeds may be beneficial to improve the accuracy of lambda targets throughout the operating range, but should result in relatively small BSFC improvements.

### 8.3 | NEXT STEPS

With the final tune presented to the team, the next step is to apply a temperature compensation to all points. The trend found during baseline testing could be applied to all speeds, but a more comprehensive approach involves acquiring the same trend at all operating points and introducing a 2 dimensional compensation table.

For future design, simulation development and engine subsystem design should help the team achieve even higher fuel efficiency. By manipulating air flow components such as intake and exhaust, the BSFC curve could be manipulated to achieve a lower average BSFC across the transient operating range of the engine. Nevertheless, the tuning and development performed in this study should return the largest benefit with this engine.

## REFERENCES

- [1] *2015 Official Rules*. Tech. N.p.: Shell, n.d. Print.
- [2] Heywood, John B. *Internal Combustion Engine Fundamentals*. New York: McGraw-Hill, 1988. Print.
- [3] Gillespie, T. D. *Fundamentals of Vehicle Dynamics*. Warrendale, PA: Society of Automotive Engineers, 1992. Print.
- [4] *McMillan Flowmeter Model 101 Installation and Operating Instructions*. N.p.: McMillan Co., 2007. PDF.
- [5] "Liquid Microturbine Technology." McMillan Co., 2015. Web. 03 Jan. 2015.
- [6] Livelli, Greg. "Flowmeter Piping Requirements." *Flowmeter Piping Requirements*. Flowcontrolnetwork, 26 Sept. 2010. Web. 03 Jan. 2015.
- [7] John S. Steinhart, Stanley R. Hart, Calibration curves for thermistors, Deep Sea Research and Oceanographic Abstracts, Volume 15, Issue 4, August 1968, Pages 497-503, ISSN 0011-7471
- [8] *Magtrol IWB115 Manual*. N.p.: Magtrol, Inc, 205. PDF.
- [9] "SAE J1349." *Surface Vehicle Standard* (2011): n. pag. Print.
- [10] John, Richard. Centrifugal Clutch Construction. Aspro, Inc, assignee. Patent US 4226320A. 22 May 1978. Print.
- [11] *Lovejoy Curved Jaw Coupling Catalog*. N.p.: Lovejoy, Inc, n.d. PDF.
- [12] "Choose a Solver." *Choose a Solver*. Mathworks, Inc, n.d. Web. 21 Jan. 2015.
- [13] Yamaha YW50 Service Manual
- [14] F.Grappe, R.Candau, B.Barbier, M.D.Hoffman, A.Belli & J.-D. Rouillon (1999) Influence of tyre pressure and vertical load on coefficient of rolling resistance and simulated cycling performance, *Ergonomics*, 42:10, 1361-1371, DOI: 10.1080/001401399185009
- [15] *A Handbook for the Rolling Resistance of Pneumatic Tires*. N.p.: Industrial Development Division Institute of Science and Technology, The University of Michigan - Ann Arbor, 1979. PDF.
- [16] Jawad, B., E. Marck, D. Tingley, J. McCoy, A. Ondes, E. Poota, and V. Floma. *Best Practice for an SAE SUPERMILEAGE Vehicle*. Tech. no. SAE2001-01-2469. N.p.: Lawrence Technological U, n.d. SAE. Web.

- [17] Ciocia, K., C. Floren, and A. McGraw. *Cal Super Mileage Vehicle Engine Displacement Reduction and Electronic Fuel Injection*. N.p.: University of California - Berkeley, 13 Dec. 2004. PDF.
- [18] Murray, J. *MS2/V3.0 Hardware Guide*. N.p.: Megasquirt, 2014. PDF.
- [19] Murray, J. *TunerStudio MS Lite Reference Guide*. N.p.: Megasquirt, 2014. PDF.
- [20] Murray, J. *Megasquirt-2 Product Range – Setting Up*. N.p.: Megasquirt, 2014. PDF.

## APPENDICES

### A1. SYSTEM SPECIFICATIONS

<b>Engine</b>	
Scooter Model	ZUMA 50F
Year	2012
Compression Ratio	12:1
Bore	38 mm
Stroke	43.5 mm
Fuel Delivery	Port Injection
Oil System	Wet Sump
Oil Type	Amsoil 10W-40
Crank Sensor	VR
Thermostat	Removed
Coolant Temp Sensor	DENSO 5YP1
Oxygen Sensor	BOSCH Wideband
Wideband Controller	Innovate LC-1
<b>Magtrol Dynamometer + Software</b>	
Model	WB115
Software	M-TEST 5.0
Encoder	60 ppr
Quadrature Input	Disable
Torque Filter	3 Hz
Nominal Speed	2865 RPM
Max Speed	5000 RPM
Max Torque	50 Nm
Scale Factor	50
Torque Ratio	1.0
Speed Ratio	1.0
<b>Fuel System</b>	
Fuel Type	87 Octane Gasoline
Operating Pressure	45 psi
Pressure Reference	Intake Manifold
Flow Meter Model	McMillan 101
<b>Drive System</b>	
Couplers	Lovejoy
Type	Curved Jaw
Size	28
Elastomer	Shore 48A
Dyno Shaft Size	32 mm
Dyno Key Size	10 mm
Engine Shaft Size	3/4 in
Engine Key Size	3/16 in

## A2. DYNAMOMETER SETUP, SAFETY

### Initial Setup Procedure

- Ensure water system inlet is connected and outlet is set up in drain
- Secure exhaust fan extension adequately over engine exhaust
- Identify fire extinguisher location
- Check fuel level, fill as necessary
- Ensure that coupler is adequately lubricated
- Perform full alignment of drive system
- Torque mounting hardware, check other hardware often
- Check dyno wiring and ensure nothing is laying on the ground
- Ensure all switches on dash are in off position (down)
- Ensure throttle is fully closed
- Check that emergency switch is closed (twist out if necessary)
- Retrieve car battery and place at a distance away from dyno
- Plug in battery and charger and ensure cable lays flat on floor
- Flip fuel switch, check or adjust fuel pressure as necessary. Turn off afterwards.

### Power on procedure

- Open M-TEST software, load appropriate program settings
- Open ECU interface, check communication
- Turn on ECU, check ECU data transfer
- Turn on Dynamometer controller
- Put on ear protection if not acquired already
- Turn on exhaust fan and water supply

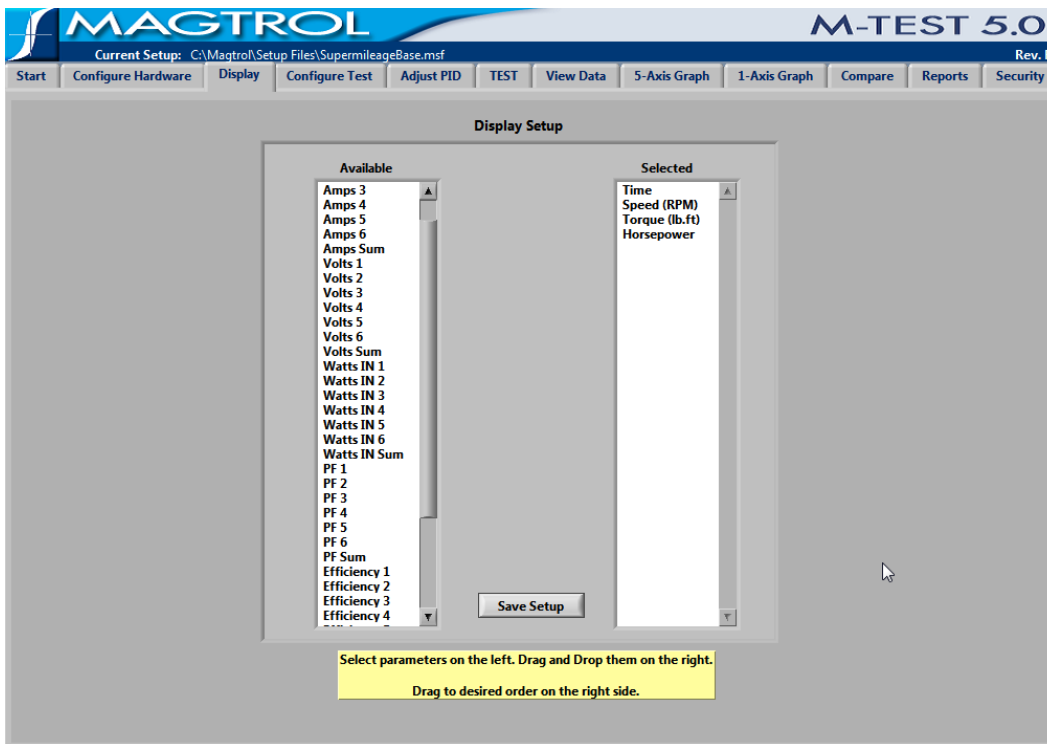
### Power Down Procedure

- Turn off fuel and ignition switch simultaneously to kill engine
- Turn off ECU when data is acquired
- Turn off dynamometer controller
- Turn off exhaust fan and water
- Unplug battery charger and battery, return battery to location

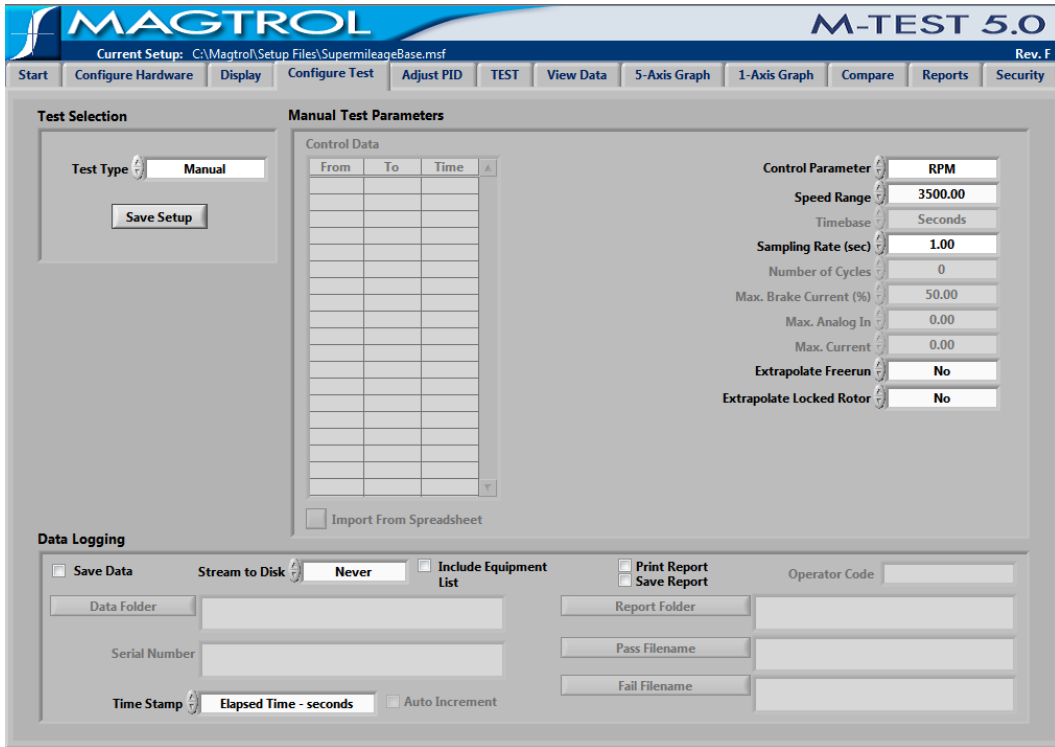
### A3. MAGTROL DYNAMOMETER TUTORIAL

This tutorial is for steady state testing, as applied during this study. For transient and drive cycle testing, please see the Magtrol manual.

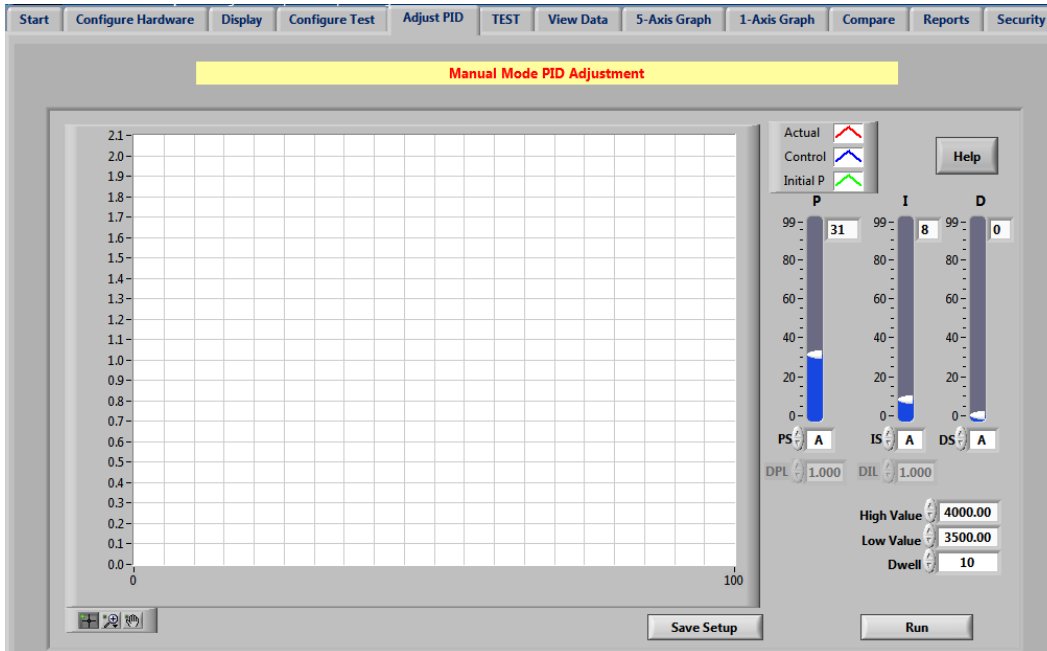
1. Follow initial setup and power up procedure
2. On 'Display' Tab, drag variables of interest to 'Selected' area.



3. Open 'Configure Test' tab. Ensure test type is 'Manual', control parameter to 'RPM' and speed range to desired steady state speed. Set sampling rate to 1Hz and do not extrapolate free run or locked rotor.



4. Open PID control tab in M-TEST. Skip to step 10 if PID is already tuned.  
 Setup up desired alternating speeds for the controller, and ‘dwell time’,  
 which is time spent at each speed (10 seconds used during this study)



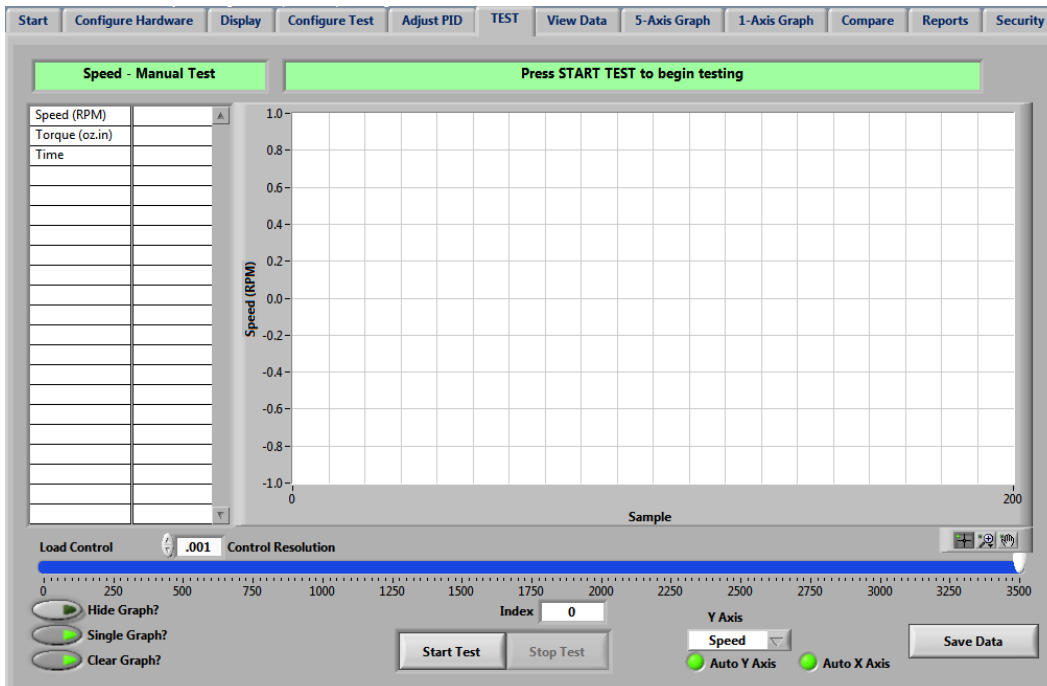
5. Start engine and let the engine warm up to operating temperature.
6. Open throttle and bring engine to RPM above upper RPM PID limit.
7. Click 'Run' and continue opening up throttle position to at least 10%.
8. Manipulate controller values until desired engine response is adequate.

Some slight overshoot may be desired for quicker response. The response shown in the following image shows varying RPM even at steady state, a characteristic of the engine during this study.





9. Close the engine throttle and click 'Stop' to end the PID testing.
10. Open the 'Test' tab and ensure that the load control slider is set to maximum (desired) test speed.



11. When ready, click on 'Start Test' and open the throttle simultaneously.  
While the load is being applied, continuously open the throttle gradually until the load is fully applied.
12. Vary the blower speed to reach desired steady state temperature.
13. When steady state is desired, click 'Record data' and Magtrol will start recording data. Verify by ensuring that the index value is increasing at the data sampling rate.
14. After sufficient time has passed without fluctuations (1 minute for SAEJ1349), press 'Stop Test' and close throttle immediately afterwards to prevent engine from over-revving.
15. Press 'Save Data' and save to desired directory.
16. The steady state test is now complete.
17. If testing is done, follow shut down procedure.

#### A4. SIMULATION RUN TUTORIAL

The code is commented to guide the user through modification of any variables.

There are 3 types of simulation: Single output, 2-D output, and 3-D output. Only the 3-D is covered, since the single is very straight forward and using the 2-D simulation is more simple than 3D.

##### 3-D Optimization Code

1. Open directory with both SM\_SENS\_3D.m and SUPERMILEAGE\_SIMULATION.slx
2. In the script file, edit VAR and VAR2 matrices to variables of interest.
3. Change values of both variables to VAR(n) and VAR2(j). It is important to keep track of index variables n and j.
4. Run. The simulation will display an overall simulation time estimate once, then will display select variables and iteration count.

```
*****  
Iteration: 16 of 81  
Lambda: 1.1  
Ignition: -3  
Fuel Economy: 1527.8513 MPG  
  
Energy Conversion Efficiency: 26.8274 %  
Engine Loss Total: 4.0427 %  
- Density Losses: 2.5044 %  
- Temperature Losses: -0.71397 %  
- Lambda Losses: 2.7408 %  
- Ignition Losses: -0.47484 %  
  
Average BSFC: 0.45909 lb/hphr  
Average Engine Speed: 3340.2775 RPM  
Average Tractive Force: 2.2014 lbf  
Average Engine Burn Temp: 113.5748 degrees F  
Burn time: 8.7359 %  
Engine Starts: 22
```

5. After simulation is complete, plot output variables of interest. For example, to view VAR and VAR2 against overall MPG in a filled contour

```
plot: contourf(xx, yy, MPG_3D, 60)
```

## A5. SIMULATION EQUATIONS

The forces acting on the vehicle include torque production, aerodynamic drag, rolling resistance, drivetrain efficiencies and inertia effects. Equations are developed using Gillespie's straight line acceleration modeling techniques [3]. Implementation of these equations into the Simulink environment are also outlined.

### **Torque, Inertia, Efficiency**

Tractive force at the driven wheel determines vehicle acceleration, velocity and distance. Engine torque acting through the gear ratio is the only force that moves the vehicle forward, while most other forces act negatively as drag forces.

Tractive force due to the engine torque, while constant at steady state, is affected in transient operation by the load applied. This load is determined by both the vehicle's acceleration rate and effective inertia of drivetrain components. The simplified tractive force equation that accounts for these effects is:

$$F_x = \frac{T_e N_{tf} \eta_{tf}}{r} - \{(I_e + I_t) N_{tf}^2 + I_d N_f^2 + I_w\} \frac{a_x}{r^2} \quad (12)$$

$F_x$  = Tractive Force (N)

$T_e$  = Engine Torque (Nm)

$r$  = Wheel Radius (m)

$\eta_{tf}$  = Overall Drivetrain  
Efficiency

$N_{tf}$  = Final Drive Ratio

$I_e$  = Engine Inertia ( $\text{kg}\cdot\text{m}^2$ )

$I_t$  = Transmission Inertia  
( $\text{kg}\cdot\text{m}^2$ )

$I_d$  = Drivetrain Inertia ( $\text{kg}\cdot\text{m}^2$ )

$I_w$  = Wheel Inertia ( $\text{kg}\cdot\text{m}^2$ )

$a_x$  = Longitudinal Acceleration  
(m/s<sup>2</sup>)

However, Gillespie's simplified application of drivetrain efficiency does not act on the rotating components of the drivetrain. Specifically, the positive tractive force experienced by the angular momentum of rotating components during negative acceleration (coasting) are not subject to drivetrain efficiencies, when indeed they are. To account for this, drivetrain efficiency was assumed to apply the entire tractive force equation, instead of just the torque produced. The equation was also modified to tailor it to the prototype vehicle.

$$F_x = \left[ \frac{T_e N_f}{r} - \{ (I_e + I_{cl} + I_{ch}) N_f^2 + (I_{fw} + I_{rw}) \} \frac{a_x}{r^2} \right] \eta_{tf} \quad (13)$$

$N_f$  = Gear Ratio

$I_e$  = Engine Inertia (kg•m<sup>2</sup>)

$I_{cl}$  = Clutch Inertia (kg•m<sup>2</sup>)

$I_{ch}$  = Chain Inertia (kg•m<sup>2</sup>)

$I_{fw}$  = Front Wheel Inertia

(kg•m<sup>2</sup>)

$I_{rw}$  = Rear Wheel Inertia

(m/s<sup>2</sup>)

Modifications include dropping the intermediate transmission ratio term, and adding chain, clutch and wheel inertias where appropriate. The equations used for each term are as follows:

### Final Drive Ratio ( $N_f$ )

The final drive ratio is defined as the torque ratio of the rear wheel to the clutch – or ratio of sprocket teeth in a chain driven system.

$$N_f = \frac{\text{Rear Sprocket Size (teeth)}}{\text{Front Sprocket Size (teeth)}} \quad (14)$$

### Clutch Inertia

Inertias of most rotating components are simplified to a rotating cylinder. Although the clutch contains two independent shoes, the rotating cylinder assumption was also made in the following equation:

$$I_{cl} = \frac{1}{2} m_{cl} r_{cl}^2 \quad (15)$$

$m_{cl}$  = Mass of Clutch + Shoes (kg)

$r_{cl}$  = Outside Radius of Clutch  
Assembly (m)

### Chain Inertia

The chain inertia was assumed to act as a mass acting at the radius of the front sprocket, so the cylinder equation was also used:

$$I_{ch} = \frac{1}{2} m_{ch} r_{spr}^2 \quad (16)$$

$m_{ch}$  = Mass of Chain (kg)

$r_{spr}$  = Pitch radius of front  
sprocket (m)

## Wheel Inertia

The mass of the wheel was assumed to act at around the circumference.

$$I_w = \frac{1}{2} m_w r^2 \quad (17)$$

$m_w$  = Mass of Chain  
(kg)  
 $r$  = Radius of wheel  
(m)

These terms were then fully defined in Matlab and implemented in Simulink through the following method:

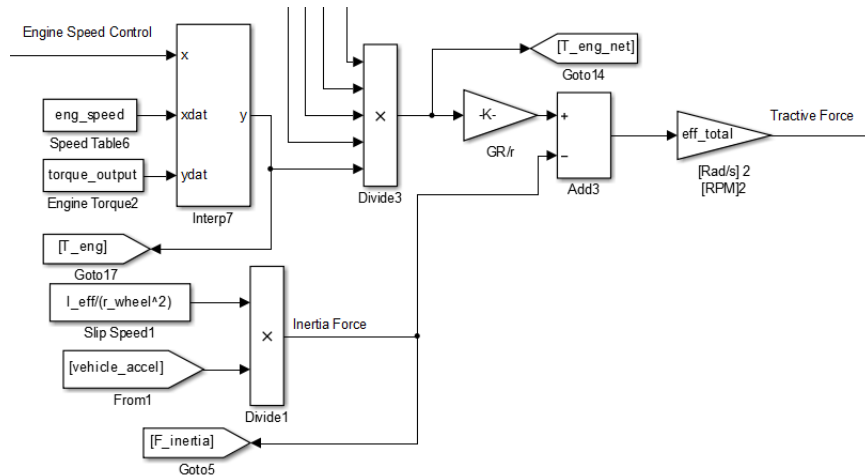


Figure 85. Tractive force equation implementation in Simulink

Starting with desired speed, an interpolation table finds the corresponding output torque value. That torque is then multiplied by the torque output modifiers (not pictured). The inertia portion of the equation begins with introducing the Matlab calculated  $I_{eff}$ , which is defined as:



$$I_{eff} = \{(I_e + I_{cl} + I_{ch})N_f^2 + (I_{fw} + I_{rw})\} \frac{1}{r^2} \quad (18)$$

This term is multiplied by the current longitudinal acceleration of the vehicle to find the inertial component of tractive force.

### Aerodynamic Drag

The simplified aerodynamic drag equation shown below was used in this simulation, neglecting surface drag effects for simplicity.

$$F_d = \frac{1}{2} C_d A v^2 \quad (19)$$

$F_d$  = Force of Aerodynamic Drag

$C_d$  = Coefficient of Drag

$A$  = Projected Frontal Area (m<sup>2</sup>)

$v$  = Vehicle Velocity (m/s)

For simplified implementation, all constant terms were joined in Matlab, then multiplied by the square of the velocity term in Simulink.

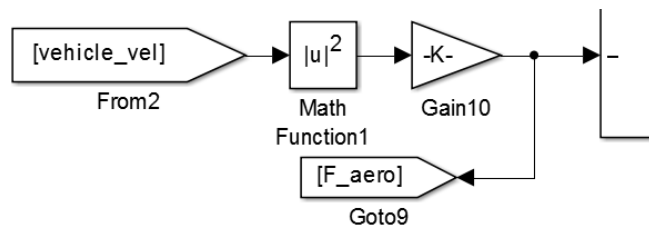


Figure 86. Aerodynamic drag force implementation in Simulink

## Rolling Resistance

Rolling resistance provided a challenge since it is normally considered to be velocity independent, acting as a constant force in the opposite direction of velocity [3].

$$F_{RR} = WC_{RR} \quad (20)$$

$F_{RR}$  = Force of Rolling  
Resistance (N)

$W$  = Weight of Vehicle (N)

$C_{RR}$  = Coefficient of Rolling  
Resistance

In the model, however, that assumption would imply the vehicle would experience negative acceleration at rest, so a velocity dependency exists. To alleviate this issue, a ramp was used that allowed the rolling resistance force to be gradually applied as the vehicle accelerated.

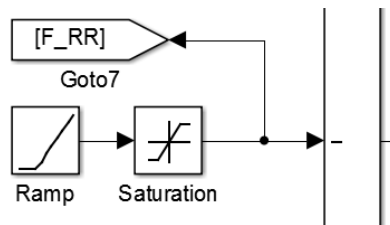


Figure 87. Rolling resistance force implementation in Simulink

### A5.1 BURN / COAST CONTROL

Since engine speed is directly linked to the rear wheels through the fixed gear ratio, vehicle velocity was used as the closed loop parameter. To ensure that no torque is produced by the engine during coasting, the input torque tables include a zero value

at zero engine speed. Therefore, engine cut/coast condition is simulated by setting the engine speed to zero.

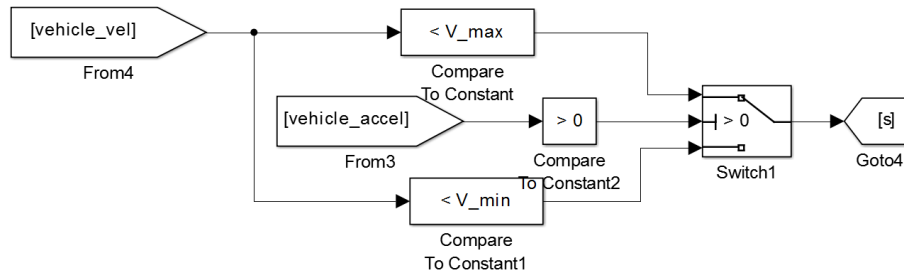


Figure 88. Engine switching function in Simulink

The variable 's' is the binary signal that controls whether the engine is running or not. The value of the switch is true (1) under either condition:

- Velocity is less than maximum velocity and acceleration is positive.
- Velocity is less than minimum velocity and acceleration is negative.

Conversely, the signal is false (0) if either of the following conditions are met:

- Velocity is greater than maximum velocity and acceleration is positive.
- Velocity is greater than minimum velocity and acceleration is negative.

## A5.2 CENTRIFUGAL CLUTCH

A large part of the operating mode for the engine is controlled by the relative motion between the output shaft of the engine and the rear sprocket of the wheel. This relative motion is controlled by the centrifugal clutch and is studied in this section.



Figure 89. Centrifugal clutch used in prototype vehicle

The clutch operates under the concept of centrifugal force; two identical shoes are connected together with springs and have a contact surface with equal radius as the inside of the clutch housing. As the engine speed increases, the force acting at the center of mass for each clutch shoe pushes each outwards radially towards the contact surface. At the clutch's *engagement speed*, contact is initially established with the housing.

From there, the normal force is determined by the difference in centrifugal force (outwards) and the radial force (inwards) pulling the shoes away from the surface. Multiplying this net normal force with the coefficient of friction and the radius of contact yields an output torque that is seen by the rear wheel.

However, the clutch's output torque can only increase to full 'stall' torque, which is the point where the friction torque of the clutch matches the torque generated by the engine. Knowing this, the following clutch characteristic was developed [A5].

$$F_N = m_{sh}(r_m + r_{gap}) \left\{ RPM \left( \frac{\pi}{30} \right) \right\}^2 - 4K_s r_{gap} \quad (21)$$

$$T_{clutch} = F_N N_{sh} \mu_f r_c \quad (22)$$

- $F_N$  = Normal Force Applied by Shoe (N)
- $m_{sh}$  = Mass of Shoe (kg)
- $r_m$  = Radius of Shoe Center of Mass (m)
- $r_{gap}$  = Static Radial Clearance (m)
- RPM = Engine Speed (RPM)
- $K_s$  = Individual Spring Constant (N/m)
- $T_{clutch}$  = Clutch Torque Output (Nm)
- $N_{sh}$  = Number of Clutch Shoes
- $\mu_f$  = Coefficient of Friction
- $r_c$  = Contact Radius (m)

The loop that defines the output torque is based on the torque produced from the increasing normal force with engine speed. If that torque is smaller in magnitude than the torque the engine produces, then the output torque is equal to the clutch torque. If that torque is larger, then the output torque is equal to the engine torque.

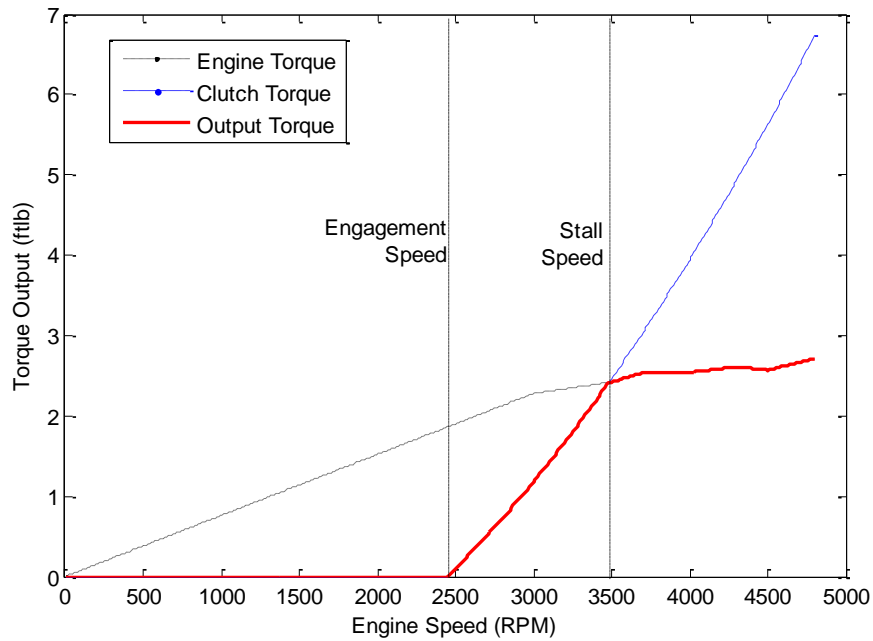


Figure 90. Clutch model characteristics

To apply this to the vehicle simulation, the dynamic clutch behavior under initial loading must be understood. When engine speed increases, the load applied to the engine also increases, which causes the engine speed to decrease. This results in decreased normal force on the pads, which decreases load on the engine, and therefore results in an increased engine speed once again [10]. Depending on the characteristic of the system, the clutch can allow the engine to maintain a constant engine RPM throughout slippage until stall torque is reached. Instead of designing a control system for this oscillatory behavior, a constant engine speed is assumed during slipping.

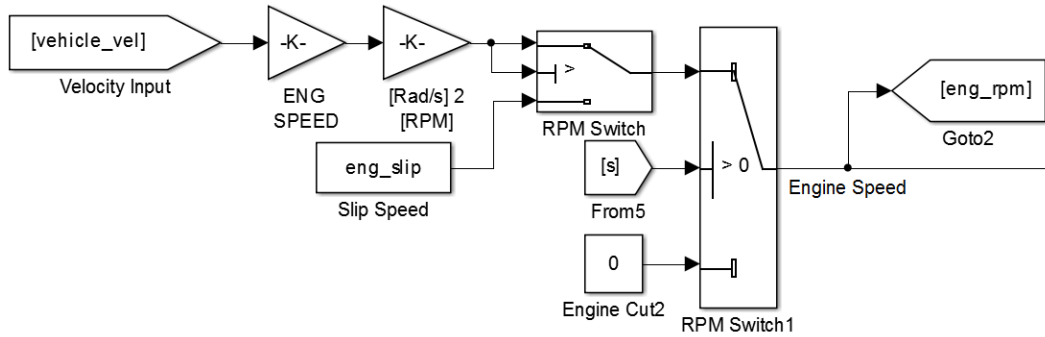


Figure 91. Engine speed control implementation in Simulink

In the figure above, engine speed is first back calculated from vehicle velocity through the gear ratio and wheel radius (ENG SPEED gain). The RPM switch compares this value with the fixed clutch slip speed (eng\_slip). If the calculated value is less than the slip speed, then the slip speed is used. Otherwise, the calculated value is used. The following switch (RPM Switch1) sets the output speed value to zero if the engine cut signal (s) is zero (coast condition).

### A5.3 MODIFIERS

This section outlines how the established relationship between AFR, lambda, and engine temperature are utilized to actively modify torque output, BSFC, and temperature transients.

#### **Density**

This factor relates the torque output during test conditions and actual torque output due to local conditions during the competition. Equations for this correction assume constant conditions, and uses engine bay temperature instead of outside temperature.

$$f_{dens} = \left( \frac{\rho_{a,engbay}}{\rho_{a,test}} \right) \quad (23)$$

$f_{dens}$  = Density Factor for Torque  
 $\rho_{a,engbay}$  = Air Density Inside Engine Bay  
 (kg/m<sup>3</sup>)  
 $\rho_{a,test}$  = Air Density During Engine Testing  
 (kg/m<sup>3</sup>)

## Lambda

As established by Heywood and verified during benchmark testing, variations in lambda have a non-negligible effect on output torque, BSFC, and heat transfer rate. This modifier was implemented by deducing an appropriate equation from these trends, normalizing them, then multiplying this factor by the existing torque, BSFC, and temperature rise rate. Since lambda and ignition are input as tables instead of a constant value, a series of 2D lookup tables is required for the following calculation.

$$f_{lambda} = \left( \frac{C_{lambda,sim}}{C_{lambda,test}} \right) \quad (24)$$

$f_{lambda}$  = Density Factor  
 $C_{lambda,sim}$  = Correction Factor With Simulation Lambda  
 Value  
 $C_{lambda,test}$  = Correction Factor With Test Lambda Value

This equation normalizes the correction factors for use in the simulation, and is implemented in the following fashion:



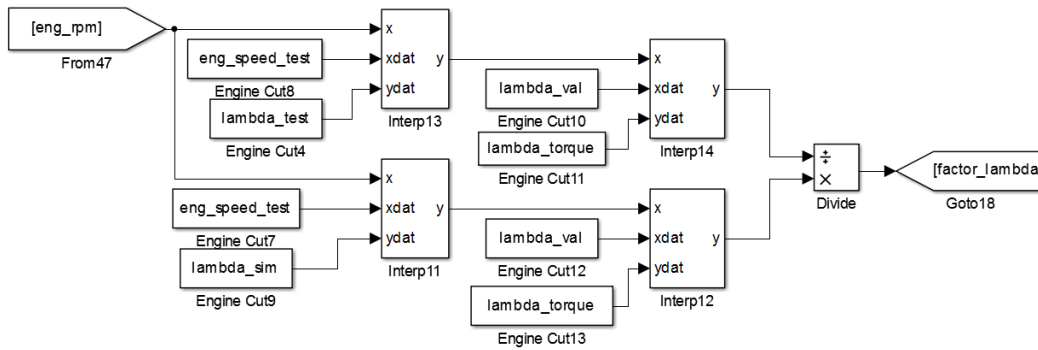


Figure 92. Simulink calculation for torque multiplier for lambda variation

In the figure above, active engine RPM is used to look up what both test and simulation lambda values are at that point. They are then input into their correction factor tables that were calculated in Matlab to output each independent factor, then divided to result in the final torque multiplier. This method is used for lambda and ignition variation for torque and BSFC.

### Ignition

The ignition function was developed relative to MBT conditions, which vary at each operating point. Therefore, a table is utilized that establishes actual MBT values, then the table of actual ignition values are shown relative to MBT values. Another 2D lookup table outputs another multiplier that is factored back into the torque production in propelling the vehicle.

$$f_{ign} = \left( \frac{C_{ign,sim}}{C_{ign,test}} \right) \quad (25)$$

$f_{ign}$  = Ignition Factor

$C_{ign,sim}$  = Correction Factor With Simulation Ignition Value

$C_{ign,test}$  = Correction Factor With Test Ignition Value

### Temperature Variation

From trends analyzed previously, engine temperature rise is assumed to be linearly proportional to engine power. With this, the simulation uses a discrete counter that only counts up every second. When the engine is burning, the value of the counter (time) is multiplied by the instantaneous temperature rise rate, which is interpolated with respect to power output using 2D table lookup function.

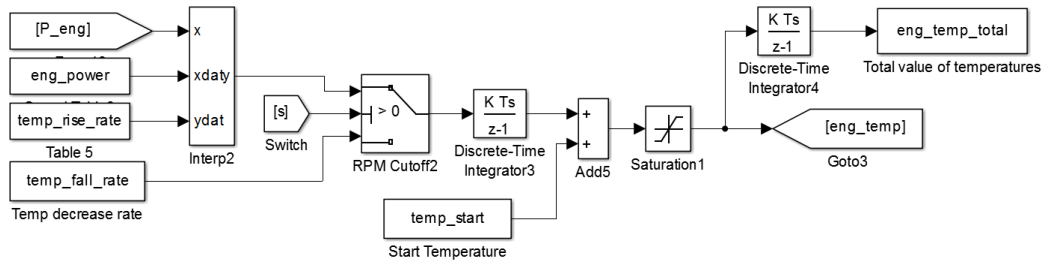


Figure 93. Temperature variation implementation in Simulink

The temperature rise rate is affected by lambda, ignition, and insulation factor, while the temperature fall rate is only affected by the insulation factor. This factor acts as a gain on the instantaneous slope of the transient temperature, and represents the effects of increasing or decreasing insulation around the engine. Increasing this factor increases the temperature rise rate proportionally while decreasing the temperature fall rate inversely proportionally. This is further analyzed in the future preliminary verification section.

## A5.4 OUTPUT CALCULATIONS

For thoroughness, all functions used in calculating fuel economy and other relevant variables are developed in the following sections.

### Overall Fuel Economy

Fuel economy is calculated by dividing the overall course distance by the volume of fuel used. The following figure shows how variables related to fuel consumption are calculated in the simulation.

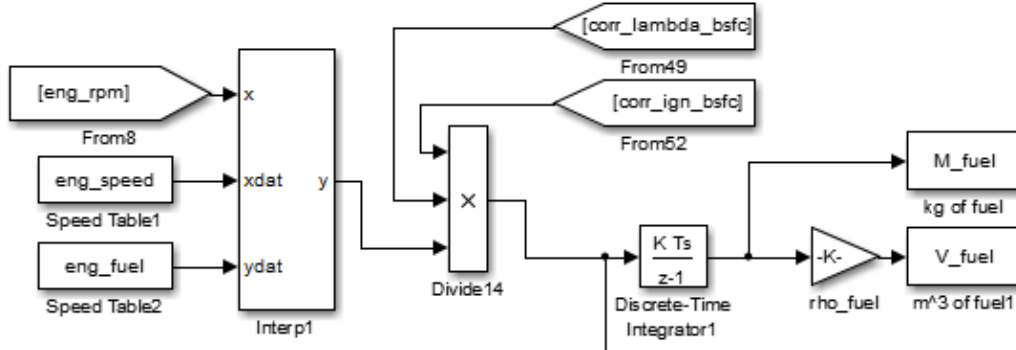


Figure 94. Obtaining variables for fuel economy calculation in Simulink

First, engine speed (eng\_RPM) is used to look up fuel consumption value at that operating point. From there, ignition and lambda values are multiplied to find the corrected fuel consumption rate. That value is then integrated to track mass of fuel consumed, then divided by the density of fuel to find volume.

### Energy

A useful exercise in the study of the overall vehicle behavior during competition conditions was an energy analysis, in which the distribution of input and output energy was gathered.

In the simulation, energy variables were acquired by translating all forces to tractive force at the rear wheel. With this, the variable was multiplied by the instantaneous vehicle velocity to give power output. Integrating this power with respect to time over the course of the simulation resulted in energy consumption of the associated component.

Although some energy components are not directly related to engine modification, it was more straightforward to include all losses associated with the vehicle, and may provide direction for future vehicle design. The following subsections outlines the equations and Simulink implementation used in the energy analysis.

### Fuel Conversion Efficiency

Fuel conversion efficiency is the ability of the engine to convert heat energy from fuel combustion to mechanical energy, and is calculated with the following equation:

$$\eta_{fuel} = \left( \frac{E_{eng}}{LHV_{fuel} m_{fuel,t}} \right) * 100 \quad (26)$$

- $\eta_{fuel}$  = Fuel Conversion Efficiency
- $E_{eng}$  = Gross Energy Produced by Engine (J)
- $LHV_{fuel}$  = Lower Heating Value of Gasoline (J/kg)
- $m_{fuel,t}$  = Total Mass of Fuel Used (kg)

The gross engine energy was found by using a 2-D lookup table in Simulink:

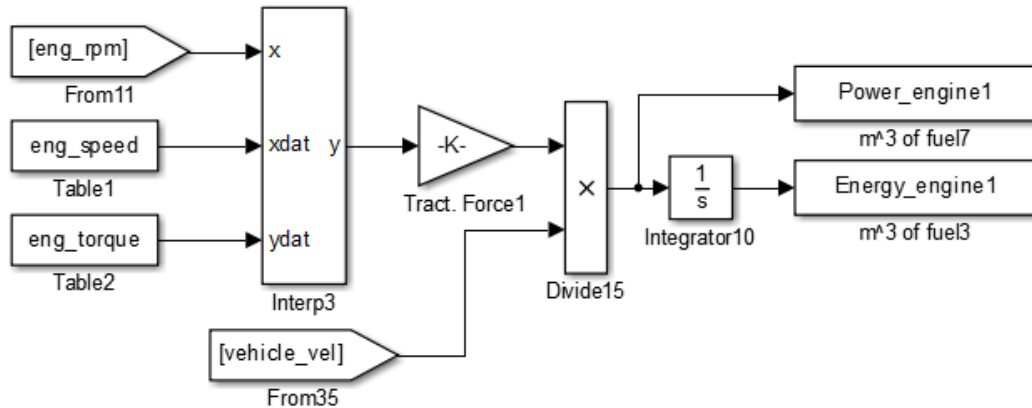


Figure 95. Calculation of net engine energy

Tractive force is acquired from engine torque by the gain (K), which is equal to the gear ratio divided by the radius of the rear wheel.

### Engine Losses

Engine losses that are defined in this simulation are energy losses from the torque modifiers, including:

- Density factor (local air density)
- Temperature factor (variations in engine temperature)
- Lambda factor (difference from ideal lambda value)
- Ignition factor (difference from MBT conditions)

Since temperature, lambda, and ignition factors are the most relevant to this study, overall losses are also divided into individual energy losses to study the impact of engine tuning and temperature effects. The overall energy loss due to the variation in the variables outlined above are:

$$E_{loss} = \int_0^t (T_{eng} - T_{net}) \left( \frac{N_f}{r} \right) v dt \quad (27)$$

$T_{eng}$  = Gross Engine Torque Output (Nm)  
 $T_{net}$  = Net Engine Torque (Nm)  
 $v$  = Vehicle Velocity (m/s)

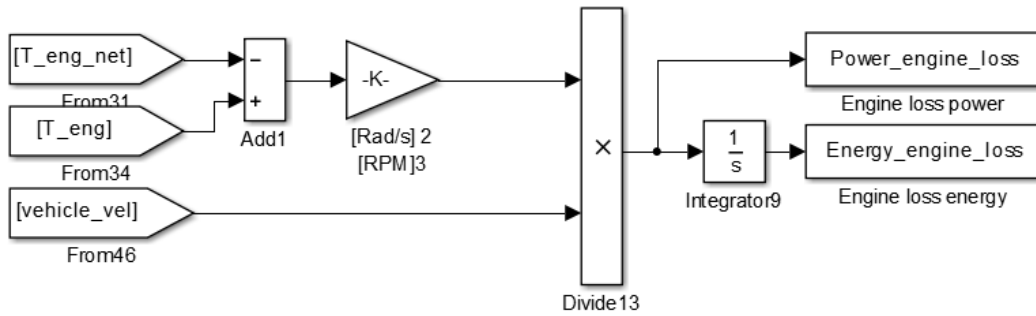


Figure 96. Energy losses from engine calculated in Simulink

From here, energy consumption of each factor is calculated as a portion of the energy loss with the following equations:

$$E_{factor} = \int_0^t (1 - f) T_{eng} \left( \frac{N_f}{r} \right) v dt \quad (28)$$

$$\%_{factor} = \left( \frac{E_{factor}}{E_{loss}} \right) * 100 \quad (29)$$

$E_{factor}$  = Energy Consumed by Factor (J)  
 $f$  = Torque Multiplier

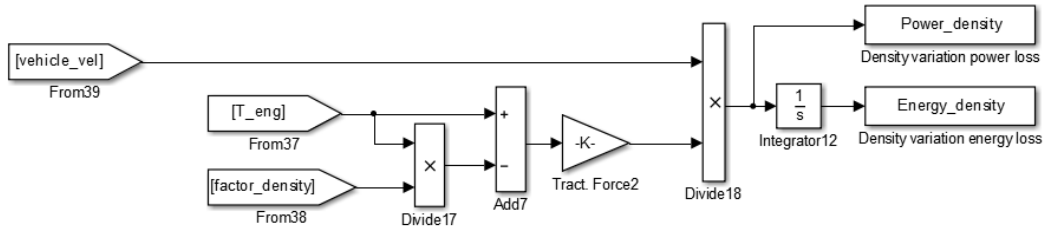


Figure 97. Simulink implementation of energy lost from torque factors

The method in the figure above is identical for the temperature, lambda, and ignition factors also.

### Clutch Losses

The team hypothesized that most of the available energy in their system was being wasted as heat energy during clutch slippage. Although the constant-speed clutch engagement model is a simplification, it helped quantify their hypothesis. The equation used for clutch loss was:

$$E_{slip} = \int_0^t (T_{eng} - T_{out}) \left( \frac{N_f}{r} \right) v dt \quad (30)$$

$$\%_{slip} = \left( \frac{E_{slip}}{E_{eng}} \right) * 100 \quad (31)$$

$E_{slip}$  = Energy lost due to slippage  
(J)

$T_{out}$  = Clutch output torque (Nm)

The torque slip was calculated by subtracting the clutch torque output from the brake engine torque found through an interpolation table, then translating it to tractive force using the same gain.

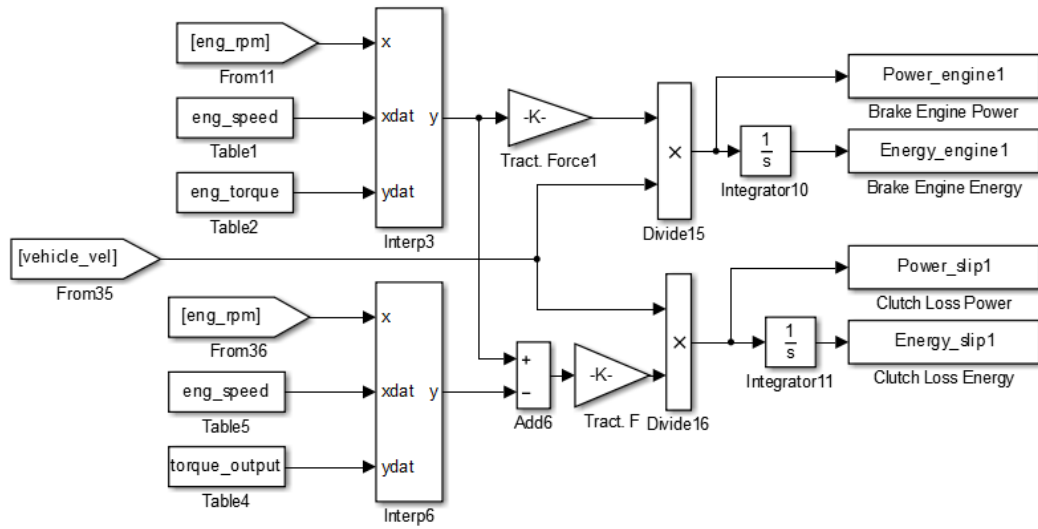


Figure 98. Clutch slip energy calculation in Simulink model

### Drivetrain Inertia Losses

Knowing that some teams implement large flywheels to recover kinetic energy during coasting, this energy component showed how much inertia energy is recovered, and may help quantify that design and its effect on overall performance. Additionally, manipulating gear ratio to change the engine's operating range will change the inertia of the drive components.

From the inertia force component in equation (13), inertia energy is calculated using:

$$E_{inertia} = \int_0^t (F_{inertia})v dt \quad (32)$$



$$F_{inertia} = \frac{I_{eff}}{r^2} a \quad (33)$$

$$\%inertia = \left( \frac{E_{inertia}}{E_{eng}} \right) * 100 \quad (34)$$

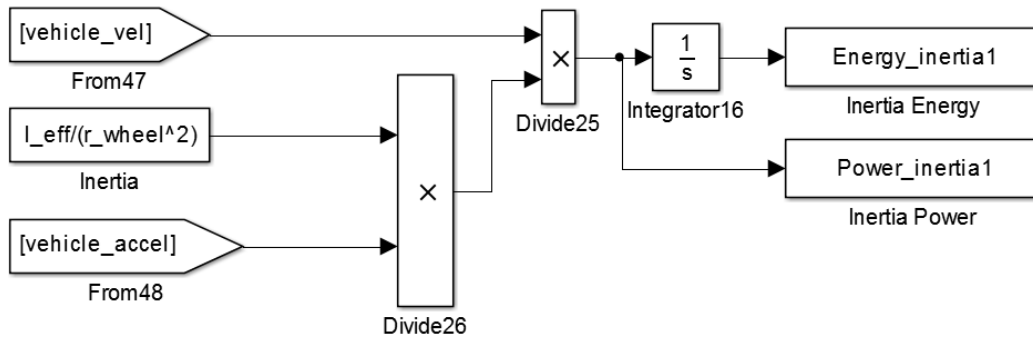


Figure 99. Inertia energy calculation in Simulink

### Aerodynamic Losses

A large portion of prototype vehicle manufacturing time is spent reducing the coefficient of drag, so aerodynamic losses may be an important variable for the team to study. Within the scope of the engine, all drag forces manipulate the burn times and therefore operating temperature range.

$$E_{aero} = \int_0^t (F_{aero}) v dt \quad (35)$$

$$\%inertia = \left( \frac{E_{inertia}}{E_{eng}} \right) * 100 \quad (36)$$

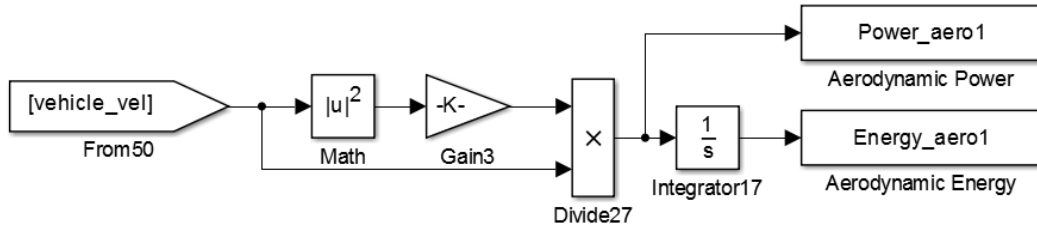


Figure 100. Simulink implementation of aerodynamic energy calculation

### Rolling Resistance Losses

This component reflects losses due to tire properties and surface quality. Like aerodynamic losses, it will affect engine operating time and temperatures.

$$E_{RR} = \int_0^t (F_{RR})vdt \quad (37)$$

$$\%_{RR} = \left( \frac{E_{RR}}{E_{eng}} \right) * 100 \quad (38)$$

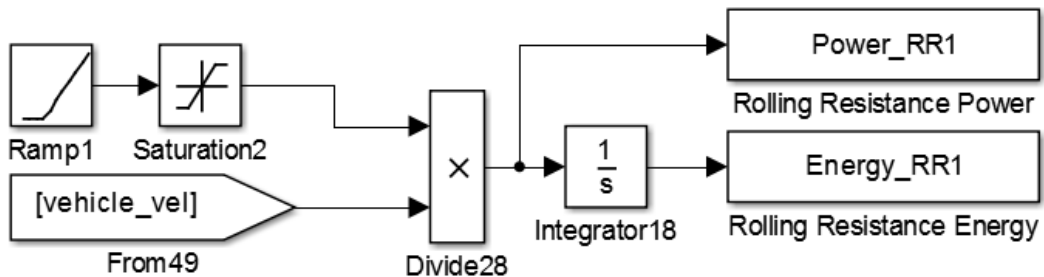


Figure 101. Rolling resistance energy calculation in Simulink

### **Average Values**

Average values for variables such as engine speed, BSFC, and temperature are used to observe changes in operating points for the engine. Since the simulation solver

utilized variable time steps, a discrete integration was utilized to take values at every second. The output variable is then divided by the time associated with it to find the average.

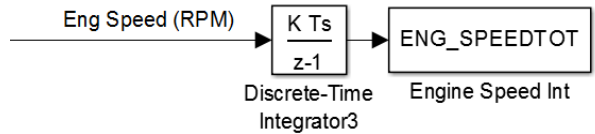


Figure 102. Example of average value calculation

### Engine Start Counter

The number of engine starts reflects how many burns the vehicle undergoes during competition, and also affects overall fuel efficiency by adding the fuel required for starts to the mass of fuel used.

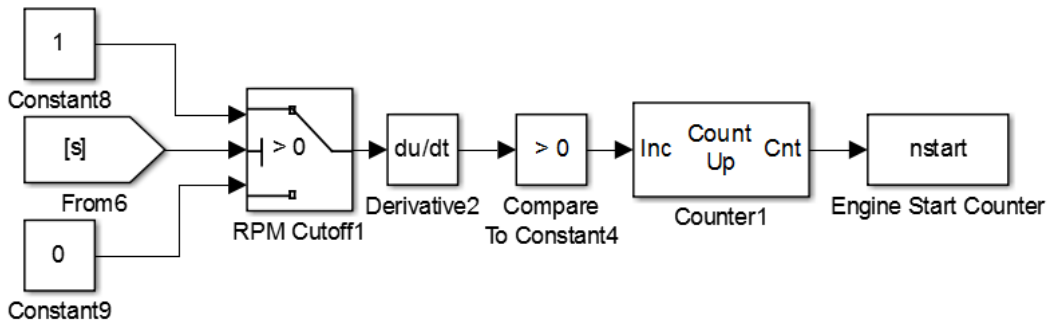


Figure 103. Engine start counter schematic

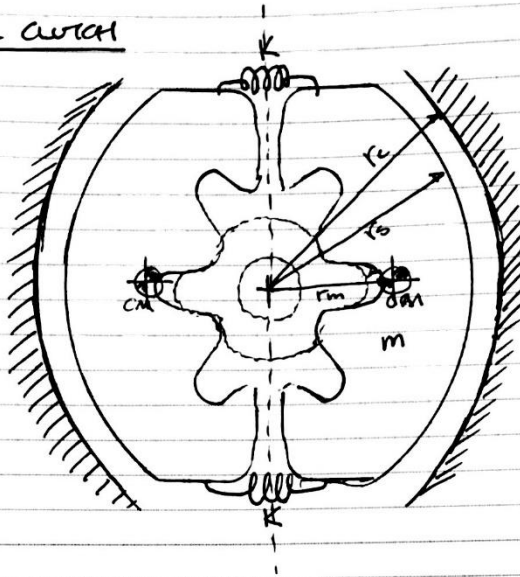
This switch creates a square function that rises to 1 when the engine is burning and drops to zero when it isn't. Taking the derivative of that creates a spike (undefined slope) each time the switch changes output. The compare to constant block outputs a 1 if true, and 0 if false. This allows the counter increment input to receive a '1'

every time the engine is switched on (rising edge), but ignores the falling edge of the switching function.

## A6. HAND CALCULATIONS

### CENTRIFUGAL CLUTCH

NORAM  
GE CLUTCH  
- LIGHT SHOES  
- RED SPRINGS



$$r_c = 44.15 \text{ mm}$$

$$r_s = 42.75 \text{ mm}$$

$$m = 0.1175 \text{ kg}$$

$$K_R = 33785 \text{ N/m}$$

$$K_B = 23836 \text{ N/m}$$

$$\mu_f = 0.4$$

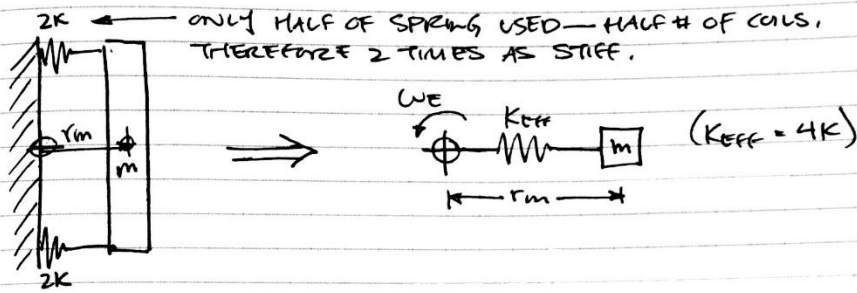
\* BACK CALCULATED

BEFORE I DEVELOP THE MODEL, I NEED TO EXPLAIN HOW THE SPRING STIFFNESSES WERE CALCULATED. FROM NORAM'S WEBSITE, I OBTAINED THE ENGAGEMENT SPEED VALUE FOR EACH COMBINATION OF SHOES & SPRINGS.

SPRING COLOR	LIGHT SHOES ENGAGE	HEAVY SHOES ENGAGE
BLACK	2100 RPM	1820 RPM
RED	2450 RPM	1950 RPM

"THIS ENGAGEMENT SPEED" REFERS TO THE INITIAL CONTACT BETWEEN THE SHOES AND CLUTCH. I ENTERED/DEVELOPED THE EQUATIONS ON THE FOLLOWING PAGE, AND USED GOAL SEEK ON EXCEL BY VARYING  $K$  SO THAT THE GAP ( $r_c - r_s$ ) = 0 @ THE ENGAGEMENT SPEED. THE MEASUREMENTS FOR DIMENSIONS WERE NOT EXACT, BUT WILL BE ITERATED WHEN THE NEW CLUTCH COMES IN.

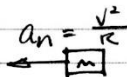
→ FROM HERE, A ~~GUARTEEER~~ MODEL WAS DEVELOPED.



FRD



NAD



$$F_k = m \left( \frac{v^2}{r_m} \right) \quad \text{WHERE} \quad v = r_m \omega \rightarrow v^2 = r_m^2 \omega^2$$

$$= m r_m \omega^2 \quad \text{WHERE} \quad \omega = \text{RPM} \left( \frac{\pi}{30} \right)$$

$$F_k = m r_m \left[ \text{RPM} \left( \frac{\pi}{30} \right) \right]^2 \quad \leftarrow \text{FORCE OF SPRINGS TOWARDS CENTER / NORMAL FORCE AGAINST CLUTCH}$$

→ ULTIMATE GOAL IS TO FIND  $F_N$  FOR OUTPUT TORQUE.

- WHEN CLUTCH BARELY ENGAGES,  $T=0$  ( $F_N=0$ )
- NEED TO SUBTRACT SPRING FORCE @ ENGAGEMENT (FIXED)

→ NORMAL FORCE OF SHOES

$$F_N = F_R - F_k \quad \leftarrow \text{FIXED} \quad \text{DEFINE } r_{\text{GAP}} = (r_c - r_s)$$

$$= m (r_m + r_{\text{GAP}}) \left[ \text{RPM} \left( \frac{\pi}{30} \right) \right]^2 - K_{\text{EFF}} (r_{\text{GAP}})$$

$$F_N = m (r_m + r_{\text{GAP}}) \left[ \text{RPM} \left( \frac{\pi}{30} \right) \right]^2 - 4K (r_{\text{GAP}})$$

$$\therefore T_{\text{CLUTCH}} = \mu F_N (r_c) \quad \text{OUTPUT TORQUE OF ENGINE (2 CLUTCH SHOES)}$$

Continued to page

→ GIVEN VE, ENGINE SPEED, ENGINE VOLUME, CAN CALCULATE  $\dot{m}_a$ . THEN INPUT AFR OR  $\lambda$ , THEN  $\dot{m}_f$  CAN BE FOUND, THEN BSFC.

$$\dot{m}_a = \left(\frac{VE}{100}\right) \left(\frac{REV}{MIN}\right) \left(\frac{1 CYCLE}{2 REV}\right) \left(\frac{VOL}{CYCLE}\right) \left(\frac{MASS}{VOL}\right) \left(\frac{60 MIN}{1 HR}\right)$$

↑ RPM                      ↑ ENGINE (DISPLACEMENT)                      ↑ DENSITY

$$= \left(\frac{VE}{100}\right) (30) (RPM) (V_{ENG}) (\rho_{AIR})$$

$$\dot{m}_a = 0.3 (VE) (RPM) (V_{ENG}) (\rho_{AIR})$$

→ MASS OF FUEL

~~$\dot{m}_f$~~  START W/ LAMBDA ( $\lambda$ )

$$\lambda = \frac{AFR_{R.}}{AFR_{ST.}} = \frac{\dot{m}_a}{\dot{m}_f} = \frac{\dot{m}_a}{(AFR_{ST.}) \dot{m}_f}$$

$$\dot{m}_f = \frac{\dot{m}_a}{\lambda (AFR_{ST.})} = \frac{0.3 (VE) (RPM) (V_{ENG}) (\rho_{AIR})}{(\lambda) (AFR_{ST.})}$$

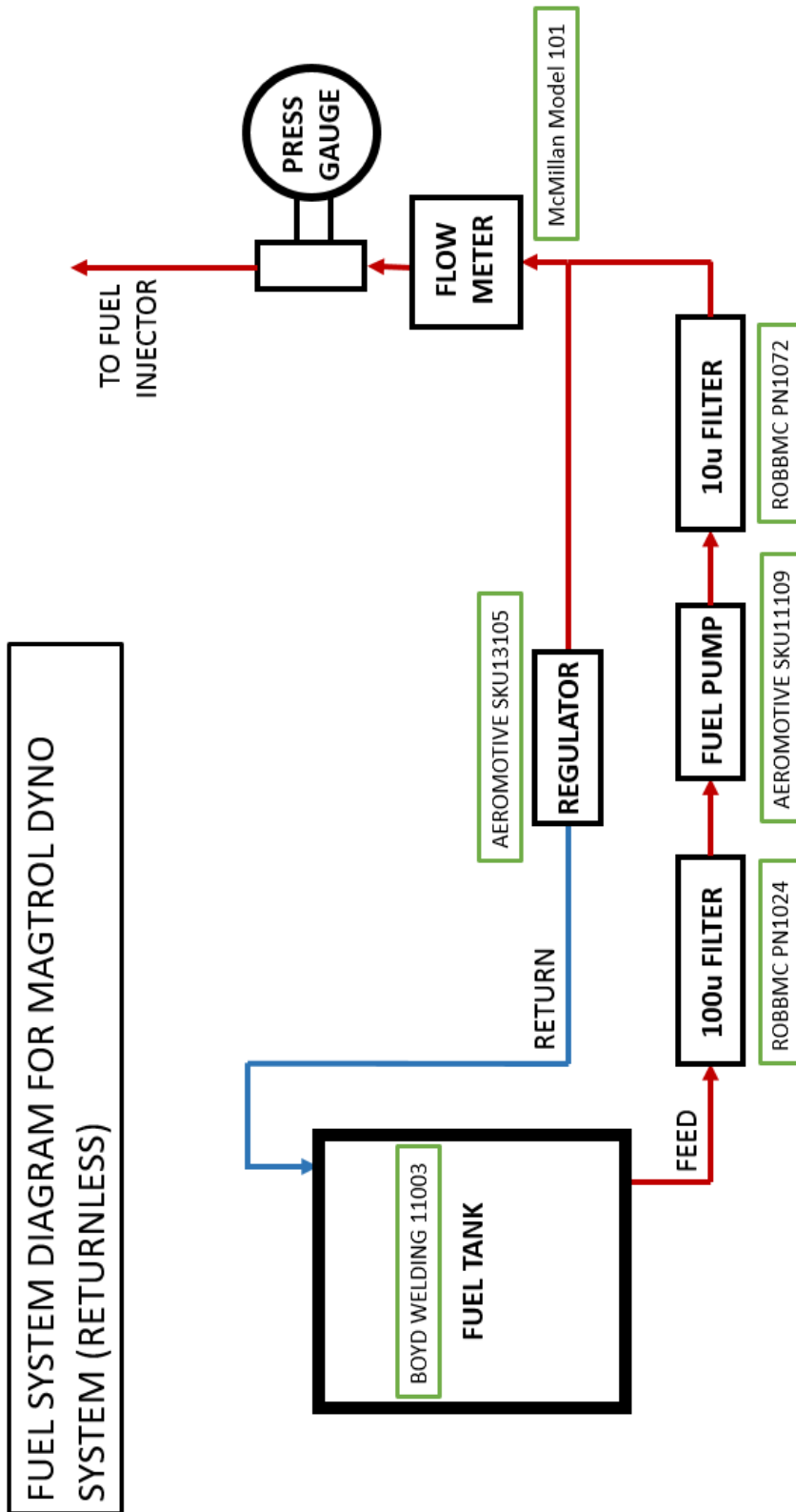
→  $BSFC = \frac{\dot{m}_f}{P} = \frac{\dot{m}_f}{T\omega}$  WHERE  $\omega = \frac{REV}{MIN} \left(\frac{2\pi RAD}{1 REV}\right) \left(\frac{1 MIN}{60 S}\right)$

$$\therefore BSFC = \frac{0.3 (VE) (RPM) (V_{ENG}) (\rho_{AIR})}{\left(\frac{\pi}{30}\right) (T) (RPM) (\lambda) (AFR_{ST.})}$$

$VE = \%$   
 $V_{ENG} = m^3$   
 $\rho_{AIR} = kg/m^3$   
 $T = Nm$   
 $BSFC = kg/whr$

$$BSFC = \left(\frac{9}{\pi}\right) \frac{(VE) (V_{ENG}) (\rho_{AIR})}{(T) (\lambda) (AFR_{ST.})}$$

A7. FUEL SYSTEM SCHEMATIC





## A8. SIMULATION CODE

```
% SUPERMILEAGE STRAIGHT LINE MODEL
% Developed by: Eric Griess
% Last Revised: 1.26.15
% Developed for completion of thesis project

%% PREFACE
% This simulation is developed in conjunction with engine testing as a
% platform for a generalized sensitivity analysis. The main purpose is to
% create a straightforward, repeatable model that can estimate the
% relationships between the variables that play a role in achieving peak
% overall fuel efficiency. Note that absolute accuracy is not the goal
% (but
% may be during future development). Instead, relative relationships are
% analyzed in order to narrow design goals for the team.

%% OUTLINE
% The simulation is split into the following sections:
%
% 1. Environmental Constants / Calculations
% 2. Course properties / definition
% 3. Vehicle Constants
% 4. Engine Information
% 5. Clutch Model
% 6. Simulation run (Simulink)
% 7. Post-sim calculations / unit conversions
% 8. Simulation outputs
% 9. Figures / Plots
%
%% MODEL ASSUMPTIONS
% The following are assumptions used at the time of model development:
%
% 1. Vehicle - Straight-line model
% 2. Course - Constant rolling resistance
% 3. Clutch - Simplified engagement model
% 4. Driver - Perfect controls based on Vmax/min
% 5. Vehicle - No-slip condition
% 6. Engine - Constant environmental constants
%
%*****
clear all
clc
close all
tic

%% CONVERSION FACTORS;
N_LB = 4.448; % N to lbf
FTLB_NM = 1.355817995; % ftlb to Nm
NM_FTLB = 1/FTLB_NM; % Nm to ftlb
HP_W = 745.699872; % HP to W
RPM_RADS = .10472; % RPM to rad/s
RADS_RPM = 1/RPM_RADS; % rad/s to RPM
LBHPPHR_KGWS = 1/5.91835E6; % lb/hp*hr to kg/W*s
KGWS_LBHPPHR = 1/LBHPPHR_KGWS; % kg/W*s to lb/hp*hr
LBHR_KGS = .453592/3600; % lb/hr to kg/s
M3_GAL = 264.172; % m^3 to gal
M3_L = 1E3; % m^3 to Liters
M3_CC = 1E6; % m^3 to cc
KG_LB = 2.20462; % kg to lb
MS_MPH = 2.23694; % m/s to MPH
```

```

%% ITERATION VARIABLES
% Here, both variable arrays are established for the 2-D trend analysis.
In order to vary these, change your step, low/high limits, then go
through the code and the appropriate VAR(n/j) to the variable of
interest. The analysis is done through a nested for loop and separated by
indexing variables (n) and (j). Be sure to keep track of which index
variable is being used during modification.

% Initialize variation for array
VAR_step = .05;
VAR_low = .8;
VAR_high = 1.2;
VAR = (VAR_low:VAR_step:VAR_high);
VAR_length = length(VAR);

% Initialize variation2 for array
VAR2_step = 1;
VAR2_low = -4;
VAR2_high = 4;
VAR2 = (VAR2_low:VAR2_step:VAR2_high);
VAR2_length = length(VAR2);

% Simulation time estimation
% Time for single iteration [s]
t_single = 101;
% Multiply by number of elements in iteration matrix
t_guess = VAR_length*VAR2_length*t_single;
% Minutes Calculation
m_guess = floor(t_guess/60);
% Seconds Calculation
s_guess = t_guess - m_guess*60;
% Display estimate for overall run time
disp(['Predicted Run Time: ',num2str(m_guess),' min, ',num2str(s_guess),'
sec']);

% Initialize output arrays
% In order to increase the efficiency of this code, any variable of
interest needs to be initialized here by creating a matrix of equal size
to the iteration matrix (n x j)
MPG_LOOP = zeros(VAR2_length,VAR_length);

AVG_ENGSPEED_LOOP = zeros(VAR2_length,VAR_length);
AVG_ENGTEMP_LOOP = zeros(VAR2_length,VAR_length);
AVG_ENGTORQUE_LOOP = zeros(VAR2_length,VAR_length);
AVG_BSFC_LOOP = zeros(VAR2_length,VAR_length);
AVG_TRACT_LOOP = zeros(VAR2_length,VAR_length);

BCR_LOOP = zeros(VAR2_length,VAR_length);
NSTART_LOOP = zeros(VAR2_length,VAR_length);
CONV_EFF_LOOP = zeros(VAR2_length,VAR_length);

LOSS_ENG_LOOP = zeros(VAR2_length,VAR_length);
LOSS_DENS_LOOP = zeros(VAR2_length,VAR_length);
LOSS_TEMP_LOOP = zeros(VAR2_length,VAR_length);
LOSS_LAMBDA_LOOP = zeros(VAR2_length,VAR_length);
LOSS_IGN_LOOP = zeros(VAR2_length,VAR_length);
LOSS_CLUTCH_LOOP = zeros(VAR2_length,VAR_length);
LOSS_AERO_LOOP = zeros(VAR2_length,VAR_length);
LOSS_RR_LOOP = zeros(VAR2_length,VAR_length);
LOSS_DRIVETRAIN_LOOP = zeros(VAR2_length,VAR_length);

```

```

LOSS_INERTIA_LOOP = zeros(VAR2_length,VAR_length);

%% SECTION 1: ENVIRONMENTAL CONSTANTS
g = 9.80665; % Gravity [m/s]
altitude = 71; % Altitude [m]
% Local Pressure [Pa]
p_local = 101325*(1-2.25577E-5*altitude)^5.25588;
temp_local = 25+273; % Local Temperature [K]
R_air = 286.9; % Gas Constant Air [J/kgK]
% Local Air Density [kg/m3]
rho_air_local = p_local/(R_air*temp_local);
rho_fuel = 725.4; % Density of gasoline [kg/m3] % Lower
Heating Value of Fuel [J/kg]
LHV_fuel = 43.44E6;

%% SECTION 2: COURSE PROPERTIES / DEFINITION
course_dist_mi = 6; % Course Distance [mi]
course_dist = course_dist_mi*1609.39; % Course Distance [m]

% VELOCITY RANGE DEFINITION
V_aim = 15; % Target Velocity [MPH]
V_range = 5; % Velocity Range (+/-) [MPH]
V_min_MPH = 2; % Minimum Velocity [MPH]
V_max_MPH = 20; % Maximum Velocity [MPH]

% CONVERSION FOR SIMULATION
V_min = V_min_MPH*0.447; % Minimum Velocity [m/s]
V_max = V_max_MPH*0.447; % Maximum Velocity [m/s]

%% SECTION 3: VEHICLE CONSTANTS
M_vehicle = 40; % Mass of Vehicle [kg]
M_rider = 57; % Mass of Rider [kg]
M_total = M_vehicle+M_rider; % Total Mass [kg]
W_total = M_total*g; % Total Weight [N]

% TRANSMISSION
GR = 11.4; % Final Gear Ratio
N_wheel = 2; % Number of Wheels
N_chain = 2; % Number of Chains
m_wheel = 0.4; % Mass of Wheel [kg]
r_wheel = 0.24; % Radius of Drive Wheel [m]
m_clutch = .95; % Mass of Clutch Assembly [kg]
r_clutch = 0.05; % Outer Radius of Clutch [m]
m_chain = 0.2; % Mass of Chain [kg]
r_sprocket = 0.1; % Radius of drive sprocket [m]

% INERTIA
I_eng = 0.01; % Engine Inertia [kg*m2]
I_clutch = m_clutch*r_clutch^2; % Clutch Inertia [kg*m2]
I_chain = m_chain*r_sprocket^2; % Chain Inertia [kg*m2]
% Wheel Inertia [kg*m2]
I_wheel = N_wheel*(m_wheel*(r_wheel^2-.02^2));
% Effective Inertia @ Engine [kg*m2]
I_eff = (I_eng+I_clutch+I_chain)*GR^2+(N_wheel*I_wheel);

% DRIVETRAIN EFFICIENCY
eff_chain = .90; % Chain Efficiency
eff_hub = .98; % Hub Efficiency
eff_total = .85; % Total Drivetrain Efficiency

% AERODYNAMICS

```

```

FD_A = 0.36845; % Frontal Vehicle Area [m2]
CD = 0.12; % Coefficient of Drag
FD_C = 0.5*rho_air_local*FD_A*CD; % Drag Constant for Simulation

% ROLLING RESISTANCE
% Value obtained from Michelin Prototype Tire Specifications
RR_C = 0.002*3.5; % Coefficient of Rolling Resistance[kg/kg]
RR_F = W_total*RR_C; % Rolling Resistance Force [N]

counter = 0; % Counter used to count iterations during simulation

for j = 1:length(VAR2); % Parent for-loop varying variable 2 (j)

for n = 1:length(VAR); % Child for-loop varying variable 1 (n)
%% SECTION 4: ENGINE INFORMATION
% First, we need to establish a uniform matrix that we can interpolate
all inputs to in order to get correct matrix sizes. For that, an
increment matrix is established, then all input tables are
interpolated/scaled to fit.
step = 10; % Step Size [RPM]
RPM_max = 4800; % RPM Limit
incr = (0:step:RPM_max); % Increment Matrix
L = length(incr); % Length of increment matrix

% ENGINE GEOMETRY
bore_mm = 38; % Engine Bore [mm]
bore = bore_mm/1000; % Engine Bore [m]
stroke_mm = 43.6; % Engine Stroke [mm]
stroke = stroke_mm/1000; % Engine Stroke [m]
volume = (pi/4)*bore^2*stroke; % Engine Volume [m3]

% TEST CONDITIONS
% - These are local conditions during dyno testing in order to establish
a torque loss factor based on local engine bay conditions during
competition.
air_temp_test = 300; % Dyno local air temp [K]
elevation_test = 71; % Dyno local elevation [m]
% Dyno local pressure [Pa]
p_test = 101325*(1-2.25577E-5*elevation_test)^5.25588;
% Local air density
rho_air_test = p_test/(R_air*air_temp_test);

% ENGINE TEST DATA
% This is raw engine data from the dynamometer / ECU. Ensure that each
matrix starts with zero.
eng_speed_test = [0 3000 3500 3700 4000 4300 4500 4800]; % [RPM]
eng_torque_test = [0 2.28 2.42 2.54 2.54 2.61 2.57 2.71]; % [ftlb]
eng_BSFC_test = [0 .580 .563 .473 .553 .601 .557 .553]; % [lb/hphr]
lambda_test = [0 0.86 0.86 1 0.9 1 1 0.83];
ign_test = [0 3 3 3 3 3 3 3] ;

% ENGINE START CONDITION
% Approximation of volume of fuel used to start engine.
V_start = 42E-9; % Volume of fuel used to start [m3]

% ENGINE BAY DENSITY CORRECTION
% This uses the air temp in engine bay (manifold air temp) to calculate
the torque lost due to the change in density from both competition
elevation and differences in local pressure.
air_temp_engbay = 310;
% Local Air Density [kg/m3]

```

```

rho_air_engbay = p_local/(R_air*air_temp_engbay);
% Density correction factor
corr_density = rho_air_engbay/rho_air_test;

%***** LAMBDA PROPERTIES *****
lambda_sim = zeros(1,length(lambda_test));
lambda_sim(1,:) = VAR(n);
% ^ lambda_sim is what lambda you want to simulate. From there, the
equations are normalized to this value to find the adjusted torque
correction factor.

% TORQUE/BSFC MODIFICATION
% Lambda values during test
lambda_val = [0.8 0.9 0.95 1 1.05 1.1 1.2];
% Normalized Torque Variation
lambda_torque = [.99 .994 1 .996 .983 .959 .880];
% Normalized BSFC Variation
lambda_bsfc = [1.318 1.141 1.087 1.043 1.014 1 1.023];

%***** IGNITION PROPERTIES *****
ign_sim = zeros(1,length(ign_test));
ign_sim(1,:) = VAR2(j);

% TORQUE MODIFICATION
% Ignition values during testing
ign_val = [-4 -3 -2 -1 0 1 2 3 4];
% Normalized torque variation
ign_torque = [.972 .987 .984 .996 1 .997 .989 .981 .972];
% Normalized BSFC variation
ign_bsfc = [1.029 1.013 1.016 1.004 1 1.003 1.011 1.019 1.029];

% CORRECTED TORQUE, BSFC, POWER CURVES
% Now, the engine curves are scaled up to the increment matrix
eng_speed = incr; % Corrected RPM Matrix
% Corrected Torque [ft*lb]
eng_torque_ftlb = interp1(eng_speed_test,eng_torque_test,incr);
% Corrected BSFC [lb/hphr]
eng_BSFC_lb = interp1(eng_speed_test,eng_BSFC_test,incr);
% Engine Power [HP]
eng_power_HP = eng_speed.*eng_torque_ftlb./5252;
% Engine Fuel Consumption [lb/hr]
% * Imperial to Metric Unit Conversions
eng_fuel_lbhr = eng_BSFC_lb.*eng_power_HP;
% Corrected Engine Torque [Nm]
eng_torque = eng_torque_ftlb.*FTLB_NM;
% Corrected BSFC [kg/Ws]
eng_BSFC = eng_BSFC_lb.*LBHPPHR_KGWS;
% Corrected Engine Power [W]
eng_power = eng_power_HP.*HP_W;
% Corrected fuel consumption [kg/s]
eng_fuel = eng_fuel_lbhr.*LBHR_KGS;

% **TEMPERATURE CORRECTION
% This is a function that describes torque/bsfc variation with
temperature
% Test points for engine temperature [F]
temp_array = (80:10:280);
% Percent of maximum torque available
temp_torque = -.0007.*temp_array+1.0886;
temp_bsfc = 6e-8.*temp_array.^3-3E-
5.*temp_array.^2+0.0033.*temp_array+.87;

```

```

temp_start = 120;           % Engine temp @ start [F]
temp_max = 260;           % Maximum Engine Temperature [F]
% Minimum Engine Temperature (Ambient engine bay temp) [F]
temp_min = (air_temp_engbay-273.15)*1.8+32;

% The following variables control the rate the engine temp rises and
falls. Along with a baseline, an insulation factor is introduced to
simple modification. Both temperature rise and fall are modeled as linear
relationships with engine POWER produced.
% Effect of insulation (% increase temp rise)
ins_factor_raw = 0;
% Insulation multiplier
ins_factor = (100+ins_factor_raw)/100;
% Temp rise rate calculated from engine power [degF/s]
temp_rise_rate = 1.2*ins_factor;
% Fixed temp fall rate during coasting [degF/s]
temp_fall_rate = -.31/ins_factor;

%% SECTION 5: CLUTCH MODEL
% The goal of this section is to develop a clutch output torque vs.
engine speed relationship. The team currently uses a centrifugal clutch,
so creating a model is relatively straightforward.

% Engine speed conversion for calculations [rad/s]
eng_speed_rads = eng_speed.*RPM_RADS;

% CLUTCH CONSTANTS
clutch_N = 2;           % Number of clutch shoes
clutch_shoe_m = .1175; % Mass of each clutch shoe [kg]
clutch_uf = 0.3;       % Coefficient of friction
clutch_k = 21724.4;    % Clutch spring stiffness [N/m]
clutch_gap = 0.0014;  % Radial static clearance [m]
clutch_r = 0.04415;   % Inner radius of clutch contact surface [m]
clutch_r_com = 0.0143; % Radius to center of mass of clutches [m] %
Radial displacement of clutch shoes @ contact [m]
clutch_disp = clutch_r_com+clutch_gap;
% Radial force pulling pads in during contact [N]
clutch_FK = 4*clutch_k*clutch_gap;

% CLUTCH CALCULATIONS
% Centrifugal force on clutch [N]
clutch_FN = (clutch_shoe_m*(clutch_disp).^2)-clutch_FK;
% Initial contact speed of clutch shoes [RPM]
clutch_engage = sqrt(clutch_FK/(clutch_shoe_m*clutch_disp))*RADS_RPM;

% CLUTCH TORQUE OUTPUT
% This is a basic if statement that sets the clutch output torque to zero
if the engine speed is less than engagement speed, then calculates torque
based on centrifugal forces for any speed above engagement.

% Initialize clutch torque matrix [Nm]
clutch_torque = zeros(1,L);

for i = 1:L
    if eng_speed(i) >= clutch_engage
        clutch_torque(i) = clutch_N*clutch_uf*clutch_r*clutch_FN(i);
    else
        clutch_torque(i) = 0;
    end
end

```

```

% ENGINE TORQUE OUTPUT
% This loop sets the output torque of the powertrain (engine+clutch). If
the clutch clamping force is less than the engine, then the output torque
is defined by the clutch slippage force. If the clamping force is greater
than the torque output of the engine, then the output torque is defined
by the engine torque.

% Initialize total torque output matrix [Nm]
torque_output = zeros(1,L);

for i = 1:L
    if clutch_torque(i) > eng_torque(i);
        torque_output(i) = eng_torque(i);
    else
        torque_output(i) = clutch_torque(i);
    end
end

% SLIPPAGE VARIABLES
% This sets a mostly arbitrary engine speed during clutch slippage, and
calculates the clutch_loss array, which is only the difference in input
and output torque during slippage.
eng_slip = clutch_engage+700;
% Torque difference during slippage [Nm]
clutch_loss = eng_torque - torque_output;

%% SECTION 6: RUN SIMULATION
% Time to run the simulation. It is a straight-line acceleration model
based around controlling engine speed. Using velocity feedback, engine is
speed is switched from 0 (coast) to fixed clutch engagement speed (burn).
The centrifugal clutch is assumed to act at constant speed during
slippage.

sim('SUPERMILEAGE_SIMULATION');

%% SECTION 7: POST SIMULATION CALCULATIONS / UNIT CONVERSIONS

% TIME
min = time./60; % Reference time [min]
hr = time./3600; % Reference time [hr]
time_burn = burn(end); % Time engine is burning [s]
time_slip = time_slip(end); % Time clutch is slipping [s]
time_coast = time(end)-time_burn; % Coast time for BCR calc [s]

% DISTANCES
burn_dist = dist_burn(end); % Total distance engine burning [m]
slip_dist = dist_slip(end); % Total distance clutch slipping [m]

% FUEL
% Total fuel used [m3]
V_fuel_tot = (V_fuel(end)+(nstart(end)*V_start));
V_fuel_L = V_fuel_tot*M3_L; % Total fuel used [L]
V_fuel_CC = V_fuel_tot*M3_CC; % Total fuel used [cc]
V_fuel_gal = V_fuel_tot*M3_GAL; % Total fuel used [gal]
M_fuel = V_fuel_tot*rho_fuel; % Total fuel used [kg]

% ENERGY CALCULATIONS
% These calculations gather variables in the 'energy variables' portion
of the simulation in order to look at overall energy distribution of the
system.

```

```

E_fuel = M_fuel*LHV_fuel;      % Energy available from fuel [J]
E_net = Energy_net(end);      % Energy after all losses [J]
E_eng = Energy_engine(end);   % Energy available from engine [J] %
Energy difference due to torque corrections [J]
E_eng_loss = Energy_engine_loss(end);
E_slip = Energy_slip(end);    % Energy from clutch slippage [J]
E_aero = Energy_aero(end);    % Energy from aerodynamic forces [J]
E_RR = Energy_RR(end);       % Energy from rolling resistance [J] %
Energy from drivetrain inefficiencies [J]
E_eff = Energy_eff(end)
% Energy from drivetrain inertia [J]
E_inert = Energy_inertia(end);

E_density = Energy_density(end); % Energy from density losses [J]
E_temp = Energy_temp(end);      % Energy from temperature losses [J]
E_lambda = Energy_lambda(end);  % Energy from lambda losses [J]
E_ign = Energy_ign(end);        % Energy from ignition losses [J]

loss_clutch = E_slip/E_eng*100; % Energy lost from clutch [%]
% Energy lost from engine [%]
loss_engine = (E_eng_loss/E_eng*100);
% Energy lost from aerodynamics
loss_aero = E_aero/E_eng*100; [%] % Energy lost from rolling resistance
[%]
loss_RR = E_RR/E_eng*100;
% Energy lost from drivetrain inefficiencies [%]
loss_eff = E_eff/E_eng*100;
% Energy lost from inertia forces [%]
loss_inertia = E_inert/E_eng*100;

% ENGINE LOSSES
% Energy lost due to density difference during comp [%]
loss_density = E_density/E_eng*100;
% Energy lost due to temperature variation [%]
loss_temp = E_temp/E_eng*100;
% Energy lost due to lambda value [%]
loss_lambda = E_lambda/E_eng*100;
% Energy lost due to ignition value [%]
loss_ign = E_ign/E_eng*100;

% OVERALL FUEL ECONOMY
% Total volume of fuel (V_fuel_gal) is calculated by taking total volume
+ fuel used to start engine.
% Overall fuel economy [miles/gallon]
MPG = course_dist_mi/V_fuel_gal;

% OTHER CALCULATIONS
burn_avg = burn(end)/min(end);
% Average burn time [burn/min]
eng_speed_avg = eng_rpm_total(end)/(burn(end));
% Average engine speed during burns [RPM]
bsfc_avg = BSFC_tot(end)/burn(end)*KGWS_LBHPHR;
% Average BSFC operating point [lb/hphr]
torque_avg = T_eng_tot(end)/burn(end)*NM_FTLB;
% Average torque output of engine [ftlb]
tractive_avg = F_net_tot(end)/time(end)*N_LB;
% Average tractive force @ tire [lb]
temp_avg = eng_temp_burn_tot(end)/burn(end);
% Average engine temp during burn [F]
conv_eff = E_eng/E_fuel*100;
% Conversion efficiency [%]
burntime = burn(end)/time(end)*100;

```



```

% Burn time percentage [%]
BCR = time_burn/time_coast;
% Burn-Coast Ratio

counter = counter + 1; % Increment counter for iteration display

disp(['Iteration: ',num2str(counter),' of
',num2str(length(VAR)*length(VAR2))]);
disp(['Lambda: ',num2str(VAR(n))]);
disp(['Ignition: ',num2str(VAR2(j))]);
disp(['Fuel Economy: ',num2str(MPG),' MPG']);
disp(' ');
disp(['Energy Conversion Efficiency: ',num2str(conv_eff),' %']);
disp(['Engine Loss Total: ',num2str(loss_engine),' %']);
disp(['- Density Losses: ',num2str(loss_density),' %']);
disp(['- Temperature Losses: ',num2str(loss_temp),' %']);
disp(['- Lambda Losses: ',num2str(loss_lambda),' %']);
disp(['- Ignition Losses: ',num2str(loss_ign),' %']);
disp(' ');
disp(['Average BSFC: ',num2str(bsfc_avg),' lb/hphr']);
disp(['Average Engine Speed: ',num2str(eng_speed_avg),' RPM']);
disp(['Average Tractive Force: ',num2str(tractive_avg),' lbf']);
disp(['Average Engine Burn Temp: ',num2str(temp_avg),' degrees F']);
disp(['Burn time: ',num2str(burntime),' %']);
disp(['Engine Starts: ', num2str(nstart(end))]);
disp(' ');
disp('*****');

% STORE OUTPUT VARIABLES FOR EACH ITERATION
MPG_LOOP(j,n) = MPG(end);
AVG_ENGSPEED_LOOP(j,n) = eng_speed_avg(end);
AVG_ENGTEMP_LOOP(j,n) = temp_avg(end);
AVG_ENGTORQUE_LOOP(j,n) = torque_avg(end);
AVG_TRACT_LOOP(j,n) = tractive_avg(end);
AVG_BSFC_LOOP(j,n) = bsfc_avg(end);
BCR_LOOP(j,n) = BCR(end);
NSTART_LOOP(j,n) = nstart(end);

CONV_EFF_LOOP(j,n) = conv_eff(end);

LOSS_ENG_LOOP(j,n) = loss_engine(end);
LOSS_DENS_LOOP(j,n) = loss_density(end);
LOSS_TEMP_LOOP(j,n) = loss_temp(end);
LOSS_LAMBDA_LOOP(j,n) = loss_lambda(end);
LOSS_IGN_LOOP(j,n) = loss_ign(end);

LOSS_CLUTCH_LOOP(j,n) = loss_clutch(end);
LOSS_AERO_LOOP(j,n) = loss_aero(end);
LOSS_RR_LOOP(j,n) = loss_RR(end);
LOSS_DRIVETRAIN_LOOP(j,n) = loss_eff(end);
LOSS_INERTIA_LOOP(j,n) = loss_inertia(end);
end
end

[xx,yy] = meshgrid(VAR2,VAR); % Establish 3D results array
MPG_3D = MPG_LOOP';

t = toc; % Finish timer
disp(['Simulation Run Time: ',num2str(t),' s']);

```