UW EE 499 3-Credit Research Report
Winter 2021

CAN Network Analysis & Validation

Author: Austin Gilbert
Student ID: 1737454
Date: March 15th, 2021

I. Table of Contents:

II. Table of Figures:

1.0 - Competition Background Information:

The University of Washington EcoCAR team is a research organization made up of both graduate and undergraduate students competing in the national four-year EcoCAR Mobility Challenge. Sponsored by General Motors and the US Department of Energy, this competition tasks students to redesign and integrate a 2019 Chevrolet Blazer into a hybrid, autonomous vehicle without sacrificing the performance of the stock vehicle.

During the first year, the UW EcoCAR team centered on proposing and finalizing the design architectures for the vehicle. Throughout the year, multiple sub-teams considered many different design pathways including having two such teams for separate hybrid configurations: types P4 and P0. Much preparation was made in advance of the vehicle design which was to be completed by the end of fall quarter of year two; this included speculated schematic designs, CAD designs, and planned integration of such designs into the vehicle. Out of three proposed plans for the vehicle's propulsion system design, a P4 split-parallel hybrid-electric vehicle was chosen.

Entering the second year of the fourth consecutive EcoCAR Mobility Challenge, the University of Washington's Propulsion Integration Team completed the Subsystem Design Report, marking the formal milestone in which 100% of the vehicle design was finalized. During winter quarter of year two, the team shifted focus toward vehicle integration. Specifically, the UW EcoCAR team replaced and integrated the engine and transmission, installed the High Voltage junction box, constructed the rear low-voltage controller shelf, finalized all low-voltage engine bay connections, and updated all schematics and CAD diagrams. Additionally, the team mounted the half shafts, front differential, inverter, radiator, ESS, exhaust, fuel tank, Intel Tank, MRRs, and Mobileye. However, one week away from the pre-competition vehicle inspection on March 12, 2020, the UW EcoCAR lab was shut down due to the COVID-19 pandemic.

All deliverables and physical work on the vehicle were put on hold while the team continued progress virtually on a new pandemic-inspired deliverable called the Vehicle Integration Status Documentation (VISD). In this deliverable, the team summarized the status of all testing and integration progress made before the lab was shut down and created integration plans for when the lab eventually opened back up.

Beginning year three of the competition, the lab was opened to a select few members to complete critical integration work. During this time, the motor was mounted and all non-CAVs low-voltage wiring was completed in the vehicle, making way for the PCM swimlane to begin initial testing. As COVID-19 restrictions began to loosen during this winter quarter, more lab members were allowed into the lab where they completed most CAN networks and all CAVs connections, finalizing most of the required integration tasks. As the lab is likely to fully open sometime during this summer, UW EcoCAR members are beginning to focus on vehicle testing and validation to prepare for the fast-approaching year four finalization phase of the competition.

2.0 – Project Introduction:

Continuing my analysis and testing of the twelve-volt power distribution system, my second research project entailed analyzing and validating both the modified GM-Stock and team-added CAN networks that currently lie untested within the vehicle in addition to writing a complete CAN training module for the team. Since about 75% of the integration work for the vehicle was already completed at the start of this quarter, I decided to use the extra time and specificity that comes with researching to delve into understanding and testing the CAN protocol. In preparation for the next phase of the competition as well as my upcoming graduation from university, I prioritized my research on testing, validation, and education for newer members.

As the team currently lacks the knowledge and training resources to test CAN networks, I specifically created a repeatable test procedure to validate the effectiveness of the CAN networks within the Blazer. To expand the scope of my research, I worked directly with the PCM swimlane in addition to the low voltage sub-team I had spent almost three years on. Completing projects with both teams allowed me to gain a more advanced understanding of sending and receiving CAN messages.

Additionally, I created an in-depth training module to increase overall knowledge of CAN on the team. The current team-provided module only covers a basic level of understanding and is made up of ten or so unorganized documents that overlap and are inconsistent with one another. Therefore, I not only consolidated these trainings into one editable document but added my research and knowledge, creating an in-depth document that serves as the perfect introduction for all applications of the CAN Protocol on the team. With the main goal of improving this sector of EcoCAR and making it easier for others to test and validate during the final two years of the competition, I began work on the first project.

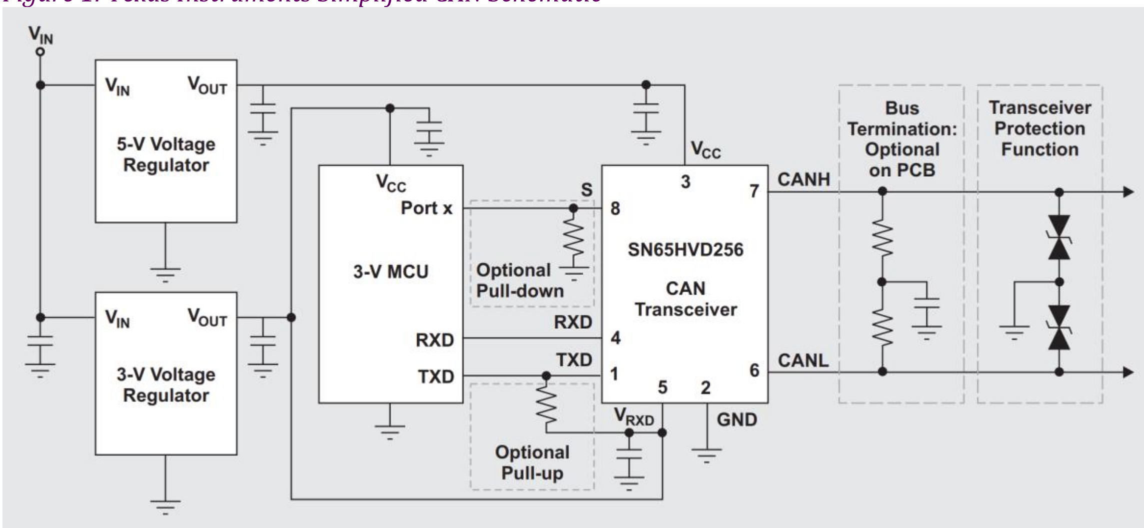
3.0 - Project Description:

3.1 – Test Harness CAN Validation

After researching the topic of CAN physical layer validation, which details the process of probing CAN wires with oscilloscopes and interface tools to analyze the quality of signals, I began looking into procedures to first try validating a simple test harness CAN bus. After looking over many CAN specifications and procedural documents, I found the best possible procedure to physically implement based mainly upon a Texas Instruments research article. Although my final procedure references multiple documents, this article worked well for initial testing because unlike other similar documents, this research report used tools and equipment that I had on hand or that I could find in the lab, saving me and the team money and time.

Specifically, the procedure describes checking all connections for the proper continuity, resistance values, and voltage values; sending and receiving CAN signals or square waveforms across the test harness CAN bus; and receiving an acknowledgment bit after introducing a second controller. *Figure 1* describes a schematic for the example validation circuit found in the research article involving a CAN controller, transceiver, and required power supplies.

Figure 1: Texas Instruments Simplified CAN Schematic



3.2 – In-vehicle CAN Validation

In a similar fashion to the previous section's Test Harness CAN Validation project, the In-vehicle CAN Validation project involved a slightly altered version of the original procedure since the Blazer's CAN networks behave differently than the test harness network. Due to the similarity of team-added CAN networks to the test harnesses I was using in addition to equipment limitations and availability, I focused primarily upon testing the modified GM-stock CAN networks in the vehicle. Using the vehicle's OBDII port and Engine Harness as access points, this project involved repeating the test harness procedure for in-vehicle networks.

3.3 – CAN Physical Layer Validation Procedure

After testing the methods that I researched on a test harness of my creation as well as the CAN buses within the vehicle, I determined the best techniques and applications of doing so. This project covered the creation of a written document detailing the proposed step-by-step procedure describing the summation of my research as well as my in-lab testing. Specifically, this document will allow members of EcoCAR to test, validate, and debug each of the Blazer's CAN networks as the team enters the main testing phase of the competition.

Unlike the documents used in my research and testing, this document outlines a procedure that is specifically tailored to the UW EcoCAR lab. For example, the part of the procedure that discusses receiving an Acknowledge bit makes use of the Vector CAN Logger as well as CANalyzer software. While components such as a CAN controller and transceiver may not be immediately available in the lab, the procedure contains recommendations for this equipment.

3.4 – In-depth CAN Training Module

Building on my research into the CAN Physical Layer, I wrote a consolidated team-wide training module for those throughout the team who would like to gain a basic-to-intermediate level understanding of CAN. Since the current team-provided modules were unorganized and inconsistent, this project was intended to provide an easy-to-use, all-in-one document that covers every possible use of the CAN protocol in UW EcoCAR besides physical layer validation. Starting with the very basics and slowly introducing more material, this document reads much like a chapter of a textbook. Each section is filled with numerous examples, figures, and tables. No prior understanding of CAN or signal processing is required before reading this document; it is made for anyone and everyone in EcoCAR to gain a simplified understanding of the CAN bus.

Quite a bit of research went into the creation of this document from sources such as previous trainings, my knowledge, and several internet articles.

4.0 - Results:

4.1 – Test Harness CAN Validation

With the researched CAN physical layer procedure in mind, I headed to the lab to first create the test harness that I would use to send and receive signals on. However, upon coming to the lab I discovered that there were no more 3-pin Deutsch connectors needed to create my harness and that although more were being ordered, they would arrive the next quarter. However, I also found that the PCM swimlane had already made the test harness I needed (*Figure 2*). Therefore, I spent the extra time I now had in the lab gaining familiarity with the CAN/LIN Vector Interface, which would be helpful later to view the signals I would try to send on my test harness. I was able to open the Interface-connected software called CANalyzer and view plots of CAN data from previous logs (*Figure 3*).

The following week, I joined PCM swimlane member Aidan Yokuda to see if I could send CAN signals across a test harness. Earlier that day, I discovered that the Analog Discovery 2 Module that was lent to me by the University for use in remote lab courses had CAN controller functionality. Since the Analog Discovery 2 Module (AD2) also contains an oscilloscope, function generator, and variable power supplies of up to 5 V, it was perfect for this project. However, upon testing the module, I discovered that although it has a CAN controller, it does not contain a CAN transceiver. Since the AD2 outputs RX (receive) and TX (transmit) signals, it needed a CAN transceiver to translate the TX signal into usable CAN High and Low signals to send onto the test harness. Therefore, I began looking up guides on how to create a CAN transceiver from the integrated circuits I had with me in my personal electronics kit. However, after looking around the lab, I found a box with integrated circuits, one of which was an MCP2551 High-Speed CAN transceiver (*Figure 4*).

With luck on my side, I began the physical validation procedure. The first section of the procedure covered checking all connections for the proper continuity, resistance values, and voltage values with a Digital Multimeter. After ensuring that my test harness was properly connected, terminated, and biased to the transceiver and AD2, I connected the module to my computer and set the user interface (*Figure 5*). With the UI set to display the oscilloscope in logic mode, the logic analyzer in protocol mode, and the protocol controller in CAN mode, I was able to edit CAN frames to send any message I would like onto the test harness.

With this setup, I sent a single CAN frame with ID #7FF and Data "1 2 3 4" and succeeded in receiving a well-defined CAN frame on the oscilloscope as well as an RX signal that matched both the TX and CAN Low Signals. Looking to *Figure 6*, the CAN frame was sent without an Acknowledge or ACK bit because there were no other controllers on the test harness to send such a bit acknowledging the signal was sent. Since the Vector CAN Logger was malfunctioning the day I tested these signals, I could not test for this ACK bit.

Figure 2: Test Harness

Figure 3: Testing CANalyzer Software

Figure 4: MCP2551 Transceiver Connections

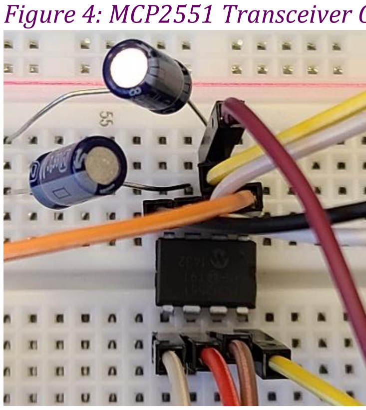


Figure 5: Connected Circuit with Test Harness

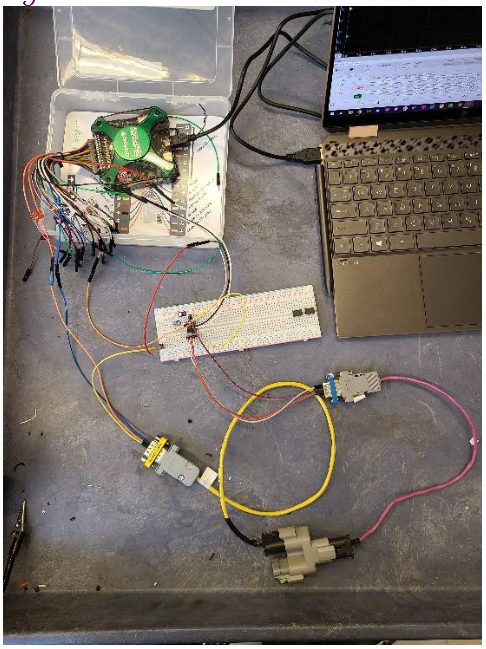
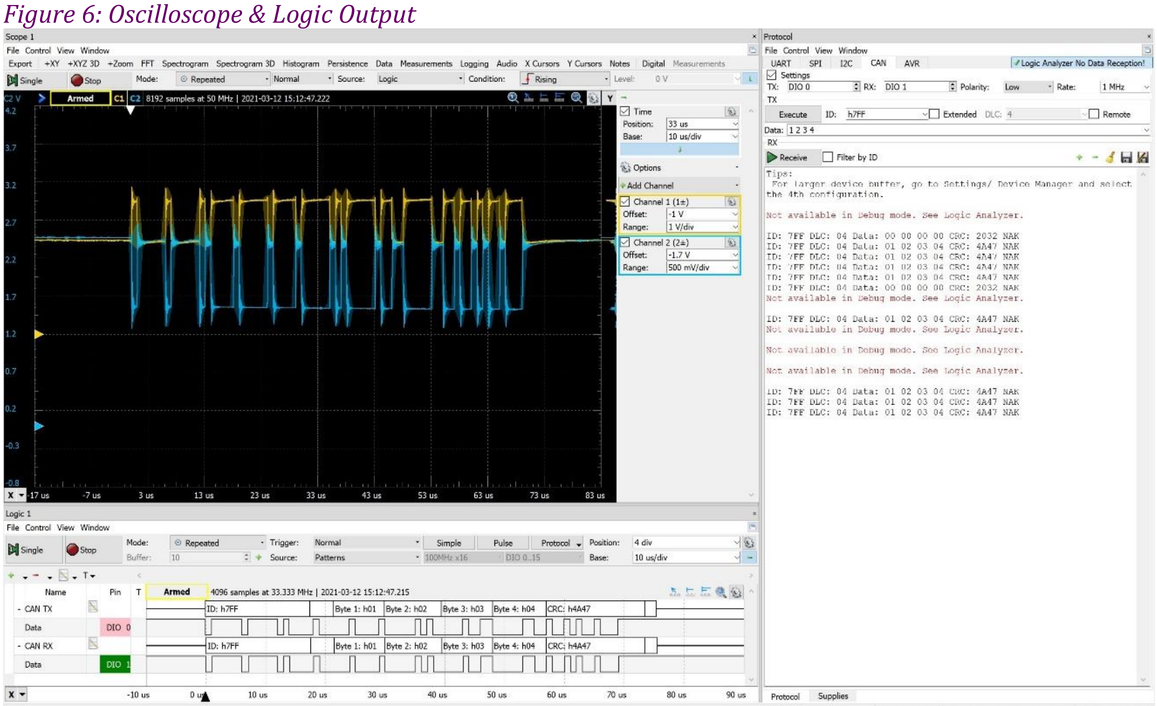


Figure 6: Oscilloscope & Logic Output



4.2 – In-vehicle CAN Validation

Before using the Analog Discovery 2 Module, I met with Aidan Yokuda to see if I could read CAN signals from the car onto an oscilloscope. Due to a lack of available equipment, we used the Fluke PM3384A Autoranging Combiscope in combination with the Vector CAN Logger to see if we could observe CAN signals from the Blazer's OBDII port. Although we could see signals, the 29-year-old oscilloscope was not fast enough to observe an entire CAN frame. Additionally, there was no obvious way to save or print the output we were getting from the oscilloscope, which was necessary for this report. Therefore, we looked to other options for reading CAN signals from the car.

After achieving a successful CAN physical layer validation with the test harness, I was determined to do the same for the in-vehicle CAN buses. Since the PCM swimlane was busy with the team-added CAN networks and since the team-added networks were extremely similar to my test harness, I focused entirely on the GM High-Speed (GMHS) and GM Chassis-Expansion (GMCE) student-modified stock CAN networks. Unplugging the oscilloscope probe wires and CAN output wires, I plugged the oscilloscope end into the GMHS "access point" near the engine harness (*Figure 8*), and the CAN output wires into the OBDII port.

However, this setup caused quite a few problems in terms of testing. First off, although I could receive signals while the car was off, I could not view them on the oscilloscope, and vice-versa when the car was on. However, since these stock networks are sending multiple CAN frames per second, it is almost impossible to override the signals on the bus. Even if overriding signals is possible, sending a signal that is not synchronized with a clock can cause error frames to appear. Since I was limited in the size of my equipment, I was also limited to where on the CAN networks I could test. Since the oscilloscope needed to observe signals at a separate location than transmitting and receiving them, this was the only location in the car where all of the wires could reach.

After spending a day in the lab to try and figure out these issues to no avail, I tested these CAN networks by catching one of the CAN frames on the GMCE network while the car was on (*Figure 9*). Since this testing was completed in the final week of the quarter, I unfortunately could not test this further. However, after a long conversation with multiple leads, I determined the causes of this problem. According to the schematic diagram of the GMHS network, the OBDII port is not connected directly to the CAN bus but to the Serial Gateway Module. This means that when the car is off, the signals will simply bounce off of the ECU's internal 120 Ω resistor without making it to the GMHS "access point" or the rest of the CAN bus.

To solve this problem, two points of access that are not the OBDII port are needed. In this way, the wires will be directly connected, freeing the signals to travel along the bus without any ECUs in the way. Although the oscilloscope reading may be less accurate, probing directly off of the transceiver instead of an alternate location on the bus may prove to be a far more effective way of validating the CAN buses in the vehicle.

Figure 7: Fluke Combiscope GMHS Output

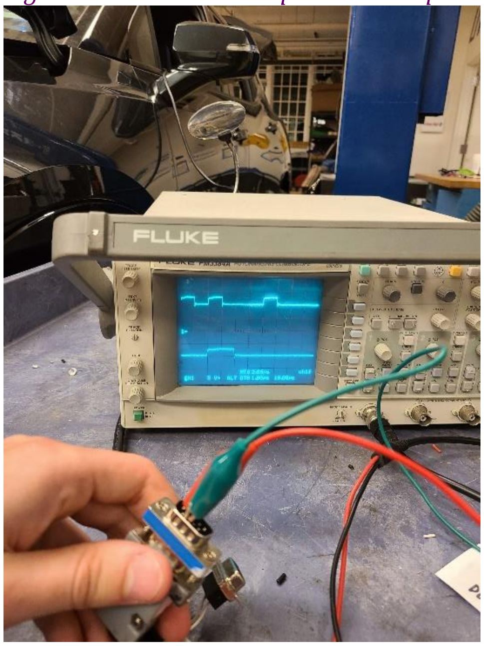
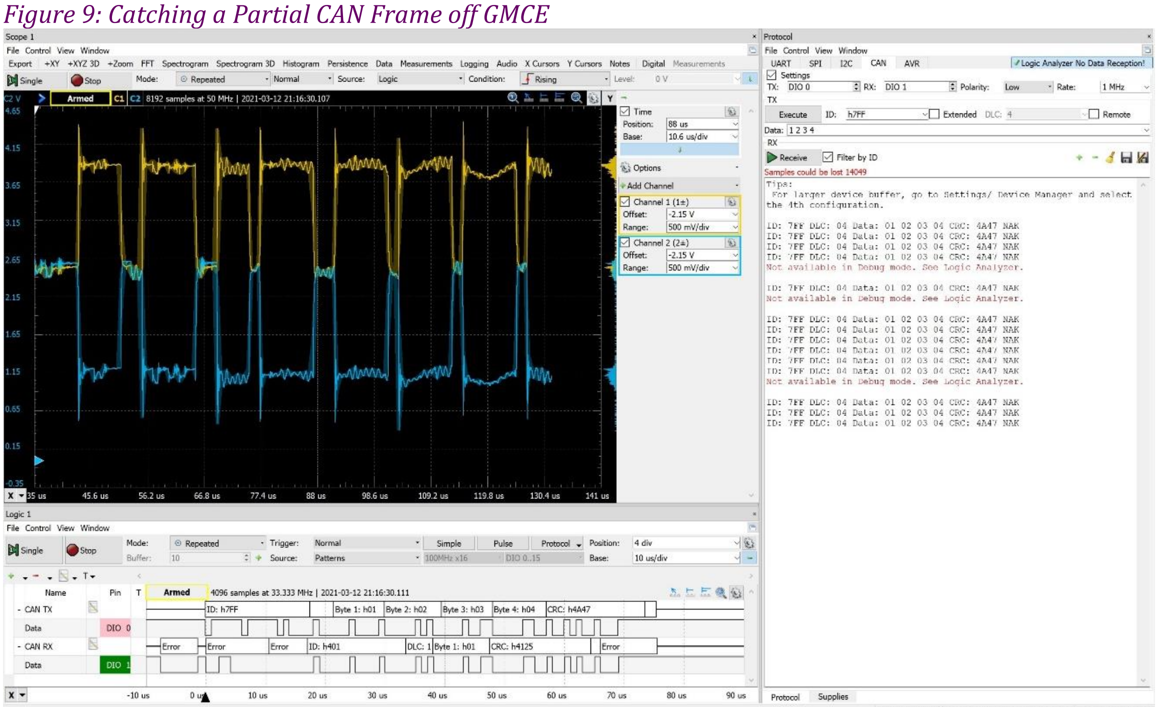


Figure 8: GMHS "Access Point"

Figure 9: Catching a Partial CAN Frame off GMCE



4.3 – CAN Physical Layer Validation Procedure

As a culmination of my findings in both the in-vehicle and test harness CAN validation projects, the CAN Physical Layer Validation Procedure document describes an improved procedure for members of UW EcoCAR to test, validate, and debug each of the Blazer's CAN networks. In building this document, I included an introduction and background to explain the purpose of the document, a requirements section to describe required items and warnings, a step-by-step procedure, and a concluding section with recommendations.

The result of this project is the document itself. This procedural document should be sent along with the report, but I will link it here just in case:

https://docs.google.com/document/d/1zJRt5873404EOn_Wxc7FID5qyw90ZNktEaTulp4ra5o/edit?usp=sharing

4.4 – In-depth CAN Training Module

Over the quarter, I worked on the team-wide CAN Training Document that includes the following sections: an introduction to CAN, how signals are sent over CAN, how to interface with CAN, how to wire a CAN bus, and a summary of important points.

The result of this project is the document itself. This training document should be sent along with the report, but I will link it here just in case:

https://docs.google.com/document/d/1mRqhCepzE6NtlrLOLaOdFv-mbZxQBdMv06M1rLaoSh0/edit?usp=sharing

5.0 - Additional Work Remaining:

While the CAN Training and Physical Layer Validation Procedure documents are complete, additional testing and validation must be completed for all modified GM-stock and team-added CAN networks that currently lie untested within the vehicle. However, before validating any networks, the team must purchase a reliable CAN transceiver, CAN controller, and oscilloscope to have the correct tools for proper physical layer validation. Once all tools have been purchased and obtained in the lab, lab members must follow the CAN Physical Layer Validation Procedure closely for every network in the vehicle.

6.0 - Conclusion:

Overall, this EE 499 research project allowed me to spend more time on a more specific topic within EcoCAR that I had not yet explored. Since I worked somewhat with CAN messages during my General Motors internship this past summer, I wanted to further investigate the electrical portion of the CAN physical layer. Contributing my efforts toward this topic enabled me to gain a better understanding of the CAN Protocol, and more widely, how signals are sent over transmission lines. I will be graduating this spring quarter and I can say with confidence that this project helped me learn so much that I will likely choose to work with the CAN physical layer at my upcoming job as a General Motors Electrical Engineer. I hope that newer members of UW EcoCAR can learn from the procedure and training documents as much as I did from this project.

7.0 - Works Cited:

Alfred, Zach. *EcoCAR: Introduction to CAN*. *Google Drive*, Robert Bosch LLC., 2 Feb. 2019.

"Analog Discovery 2 QuickStart: Video 14 - The Protocol Analyzer." Performance by Franz Kaitlyn, *YouTube*, Digilent, Inc., 3 May 2018, youtu.be/_5Iznbeb4q4.

"CAN Bus." Wikipedia, Wikimedia Foundation, 18 Feb. 2021, en.wikipedia.org/wiki/CAN_bus.

Haase, Jörn, and Gavin C Rogers. "Testing of CAN Signals." *Aerospace Testing International*, 2015, pp. 40–44.

Hareendran, T K. "DIY CAN Transceiver." *ElectroSchematics.com*, AspenCore Inc., 2 July 2015, www.electroschematics.com/diy-can-transceiver/.

"How to Connect the Oscilloscope When Dual-Trace Testing." *Interworld Electronics*, www.interworldna.com/pico/automotive/canbus_scope.php.

McBride, Cooper. "CAN Bus Overview." UW EcoCAR, 2018.

"MCP2551 High-Speed CAN Transceiver." Microchip Technology Inc., 2003.

Monroe, Scott. "Basics of Debugging the Controller Area Network (CAN) Physical Layer." *Analog Applications Journal*, 2013, pp. 18–24.