1-1-2015

Ethernet Over Plastic Optical Fiber for Use in the Control System Network for Automotive Applications

Ryan Nicholas Nazaretian

Ethernet over plastic optical fiber for use in the control system network for automotive applications

By

Ryan Nicholas Nazaretian

A Thesis
Submitted to the Faculty of
Mississippi State University
in Partial Fulfillment of the Requirements
for the Degree of Master in Science
in Electrical and Computer Engineering
in the Department of Electrical and Computer Engineering

Mississippi State, Mississippi

December 2015

Ethernet over plastic optical fiber for use in the control system network for automotive applications

By

Ryan Nicholas Nazaretian

Approved:

G. Marshall Molen
(Major Professor)

Nick H. Younan
(Committee Member)

Bryan A. Jones
(Committee Member)

Randolph F. Follett
(Committee Member)

James E. Fowler
(Graduate Coordinator)

Jason Keith
Dean
Bagley College of Engineering

Name: Ryan Nicholas Nazaretian

Date of Degree: December 11, 2015

Institution: Mississippi State University

Major Field: Electrical and Computer Engineering

Major Professor: G. Marshall Molen

Title of Study: Ethernet over plastic optical fiber for use in the control system network for automotive applications

Pages in Study: 81

Candidate for Degree of Master in Science

Plastic optical fiber (POF) for use in automotive applications is not a new concept and has been used in some vehicles for infotainment media distribution within the Media Oriented Systems Transport protocol. However, the use of POF for the control network's physical layer is a concept that has not been implemented in automotive applications. Many aspects of a vehicle can be improved by implementing POF as the physical backbone for the control network.

Currently, the Controller Area Network (CAN) is used as the primary backbone control network protocol for most automobiles as it is inexpensive and reliable. However, CAN is limited to 500 kbps in most vehicles and is easily accessible. Ethernet may provide the improvements of speed and security needed in today's feature rich and connected vehicles. The feasibility of implementing Ethernet over POF as the control network for automotive applications is the topic of this research investigation.

DEDICATION

The work provided in this thesis is dedicated to my parents, Sue and Nicky Nazaretian, my sister, Sarah Nazaretian, and my wife, Chelsea Nazaretian, who always stand beside me and believe that I can make a difference in the world.

ACKNOWLEDGEMENTS

I would like to thank Dr. Marshall Molen for enlightening me to the Mississippi State University Advanced Vehicle Technology Competition team, EcoCAR 2, and EcoCAR 3, as well as always being a great mentor by providing advice, feedback, and criticism to me as well as being a great friend.

I would also like to thank the Mississippi State University's EcoCAR 2 and EcoCAR 3 team for allowing me to participate as well as provide funding for graduate school. I want to recognize Dr. Randolph Follett, Dr. Kaylan Srinivasan, Dr. Sundar Krishan, Stephen Hayes, Casey McGee, Wesley Haney, Jeremy Walker, Dalton Childers, Matt Tidwell, and Charles Boyd, for being great friends and team members during my time on the team. The EcoCAR 3 team would not be great if it were not for those faculty advisors and team members who put in tireless work for the betterment of the team.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

ABS	Anti-lock Braking System
ADAS	Advanced Driver Assistance System
ADC	Analog to Digital Converter
AVTC	Advanced Vehicle Technology Competition
AWG	American Wire Gauge
CAN	Controller Area Network
CAVS	Center for Advanced Vehicular Systems
CRC	Cyclic Redundancy Check
DHCP	Dynamic Host Configuration Protocol
DLC	Data Length Code
ECU	Electronic Control Unit
EMI	Electromagnetic Interference
EPA	Environmental Protection Agency
ESKII	Microchip PIC32 Ethernet Starter Kit II
FSM	Finite State Machine
GUI	Graphical User Interface
I/O	Input and Output
I^2C	Inter-Integrated Circuits

IC	Integrated Circuit
ISO	International Organization for Standardization
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IP	Internet Protocol
LED	Light Emitting Diode
LIN	Local Interconnect Network
LLC	Logical Link Control
MAC	Media Access Control
MOST	Media Oriented Systems Transport
MSU	Mississippi State University
OBDII	On Board Diagnostics II
OSI	Open Systems Interconnection
OUI	Organizationally Unique Identifier
PHY	OSI Physical Layer Interface
POF	Plastic Optical Fiber
PVC	Polyvinyl chloride (plastic)
SAE	Society of Automotive Engineers
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
V2V	Vehicle to Vehicle

CHAPTER I

INTRODUCTION

1.1 Introduction

Today, the use of electronics in vehicles has increased at a rapid pace somewhat analogous to the growth of semiconductors as described by Moore's law [1]. Within the past ten years, the integration of in-vehicle navigation systems, rearview cameras, stereo systems with a plethora of input options, adaptable cruise control, Advanced Driver Assistance Systems (ADAS), and telematics have occurred. The drive to produce complex hybrid vehicles has further increased the use of electronics, and therefore, the amount of wire required for interconnections as well as the increase of bandwidth requirements for the Controller Area Network (CAN).

Current generation vehicles are more connected with vehicle-to-vehicle (V2V) communication, ADAS, autonomy, Internet of Things (IoT), and further hybridization. In fact, the main drive behind this thesis relates to EcoCAR 3, a US Department of Energy Advanced Vehicle Technology Competition (AVTC), where a stock 2016 Chevrolet Camaro is to be transformed into a production-ready performance-hybrid with a new infotainment center, ADAS incorporation, and other innovations where the overabundance of wires begs for a better solution.

There are several vehicle communication networks available today including CAN, FlexRay, Media Oriented Systems Transport (MOST), and Local Interconnect Network (LIN).

- CAN – The Controller Area Network is one of the most common communication networks in today's vehicles, supporting up to 1 Mbps of bandwidth [2]. CAN is a bus type network meaning that multiple devices can connect to the network at the same time using the multi-drop bus topology such as used for Inter-Integrated Circuits (I^2C) as shown in Figure 1.1. CAN uses differential pairs for electrical signals to filter out the effect of electromagnetic interference (EMI) and common mode noise [3].

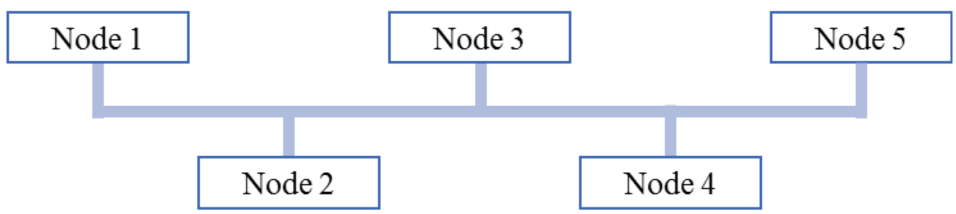

Figure 1.1 Example of Multi-drop bus topology as may be used with CAN or I^2C

- FlexRay – FlexRay is a time-deterministic communication protocol used for safety-critical or high-performance vehicle systems such as drive-by-wire, active suspension, or ADAS. FlexRay has a maximum bandwidth of 10 Mbps and uses either single or dual differential twisted pairs. The network topology for FlexRay is flexible, supporting the multi-drop topology, star topology (shown below), or a hybrid network comprised of both multi-drop and star network topologies [4].

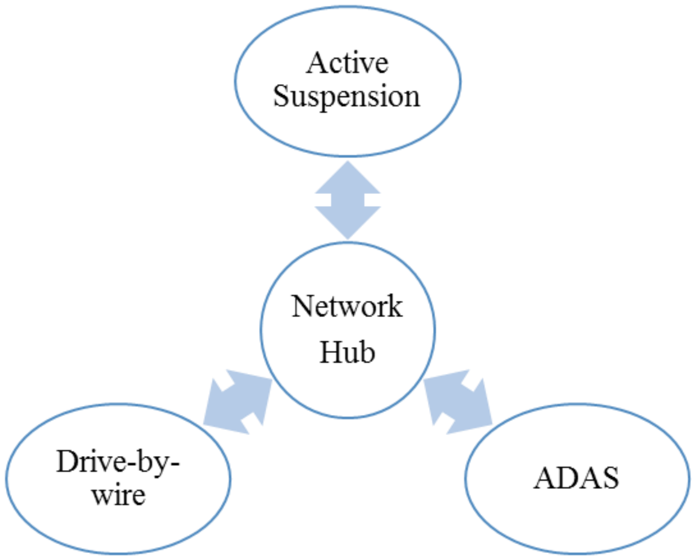

Figure 1.2 Example of a FlexRay star topology

- MOST – The Media Oriented Systems Transport protocol was designed specifically for media distribution in automotive applications. MOST is a time deterministic network developed in a ring topology as shown in Figure 1.3. MOST is unique from the other network types listed because it was designed to operate over either electrical wiring or fiber optics [5].

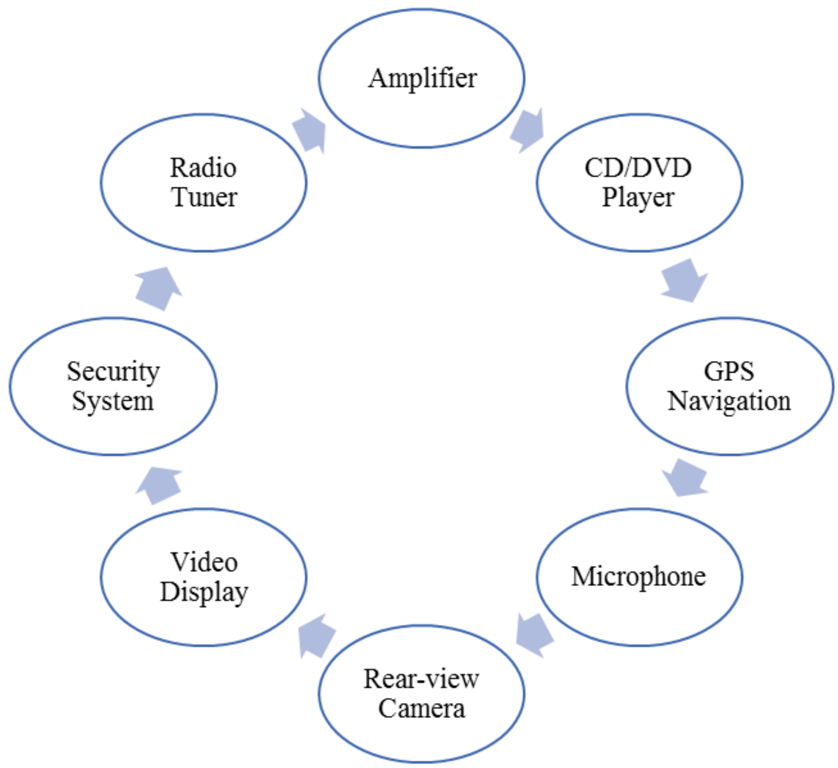

Figure 1.3 Example of a MOST ring network topology used in a vehicle

- LIN – The Local Interconnect Network is implemented in many vehicles as an alternative to CAN to lower the cost of the electronics. LIN operates using a single wire and a return on the vehicle chassis in the multi-drop topology and often utilizes the serial universal asynchronous receiver/transmitter (UART) peripheral included with many microcontrollers. Additionally, LIN is a specialized network with a master/slave configuration, meaning only one device can act as the controller [6].

1.2 Inspiration

The work described in this thesis was developed for the Mississippi State University EcoCAR 3 team's innovation topic for Year 1 of the competition. The inspiration of the goals for this thesis was brought about from experience in EcoCAR and EcoCAR 2, the previous two AVTCs before EcoCAR 3.

In the EcoCAR competition, the team's 2008 Chevrolet Captiva, redesigned as a series hybrid, had significant EMI issues due to the high voltage brushless DC electric motors and wiring located in the front of the vehicle. During an EMI event, which frequently happened when generating power, there would be a loss of engine control. One major advantage of using fiber optics is its immunity to EMI.

In EcoCAR 2, the MSU team had significant installation and reliability issues with control system wiring connecting the front of the vehicle to the rear of the vehicle. The control system wiring to the engine bay consisted of 44 wires with a length of 4 m that weighed 3.5 kg. However, this does not take into account the three CAN cable assemblies that incorporate twisted pairs, aluminum shielding, and thick insulation. The physical size of 44 wires all located within a small area in the trunk made packaging difficult and the number of wires created many potential failure points in the vehicle. These potential failure points caused many of the control system problems for the team in Year 3 of the EcoCAR 2 competition.

Ethernet over Plastic Optical Fiber (POF) has the potential to alleviate some the issues of EMI, packaging, and weight. The switch to Ethernet over POF would have removed the chance for EMI to affect the control system in EcoCAR. The switch would have also increased the bandwidth for the EcoCAR 2 vehicle's controls network from

500 kbps to 100 Mbps, allowing more signals to be transmitted on the network while also reducing the need for additional wires to be installed throughout the vehicle. Additionally, the POF cable is one-quarter of the weight of the CAN wiring it would replace.

1.3 Literature Review

The literature review looked at various web articles and peer-reviewed papers regarding issues with CAN, uses of fiber optics in the automotive industry, and use of Ethernet in automotive applications.

An article by Kathy Pretz discussed how vehicles have become more sophisticated with the advancement of technologies included in current vehicles and mentioned how the electrical and control system is the third heaviest and third costliest component in today's vehicles, behind the chassis and engine [7]. The main idea of this article was to replace CAN with Ethernet. Pretz's article described a way to accomplish this using IEEE standard P802.3bp to allow communication using less than three twisted pairs to establish gigabit link speeds.

Peter Hank also agrees with the point that Ethernet can meet the intense data transmission demands of today's vehicles [8]. He also mentions that the switch to Ethernet can reduce costs, an opinion also supported by Damon Lavrinc's article [9], which describes how the switch to Ethernet could reduce wiring harness costs by 80% while also reducing wiring harness weight by 30%. Damon's article also declared Ethernet as 'instrumental' to improving fuel economy for the new Environmental Protection Agency (EPA) fuel economy standards set for 2025 due to the weight savings a faster network interface can provide by removing signal wires [10]. A paper by

Alexandre Vasile discussed fiber optics and identified copper as an expensive resource and that there is a need for faster networks within vehicles [11].

Chung-Wei Lin identified security problems with CAN in which the network can be hacked in four different attack scenarios described in his paper [12]. The first scenario is “modification,” where an unauthorized node installed between two existing nodes splits the CAN bus into two busses and acts as a gateway between the busses. This method would be difficult to accomplish since it would require physical access to the wiring for an electronic control unit (ECU) in order to accomplish the attack. The second scenario is “fabrication,” where an unauthorized node could generate messages on the CAN bus. If these messages use the ID of another device, such as a wheel speed sensor for example, the anti-lock braking system (ABS) may react by disabling the brakes. Such an occurrence was demonstrated by hacking activists Charlie Miller and Chris Valasek [13, 14]. The third scenario is “interception,” where an unauthorized node gains access to the network and intercepts messages sent on the bus. The data acquired, such as location, speed, direction (compass), fuel level, or any other information that is transmitted over the CAN bus can be used by an attacker. For example, in the “fabrication” example, the attacker could wait until the vehicle is at a high speed before disabling the brakes. Finally, the fourth scenario is “interruption,” where an unauthorized node blocks all messages on the bus. With CAN, this is easy to do by shorting the CAN High and CAN Low signals together or by flooding the bus with high priority messages. Wei Lin also stated how implementing secure methods of data transfer in CAN is difficult because of CAN’s lack of global time used in many encryption techniques and lack of bandwidth needed for many authentication processes.

The literature review identified some deficiencies to improve upon. Such deficiencies include the lack of security with CAN, the need for higher bandwidth data communication, the need to reduce the weight of a vehicle's wiring, and the need to reduce the cost of the wiring harness.

1.4 Goal

The goal for this thesis project is to explore the feasibility of using Ethernet over POF as an automotive control network by developing a working prototype and providing an analysis of the cost, installation considerations, security, bandwidth, and weight of such a network. The prototype is demonstrated with two objectives. The first objective is to control a headlight assembly that includes a low beam, high beam, running light, and turn signal. The 12 V bulbs are controlled by Ethernet over POF that is capable of providing feedback from the light driver chips such as current or short-circuit conditions. The second objective is to achieve backwards compatibility by implementing a CAN and Ethernet gateway so that CAN controlled hardware, such as the battery pack in the EcoCAR 3 vehicle, can be controlled using Ethernet over POF. Both of the solutions to these objectives are shown together in Figure 1.4.

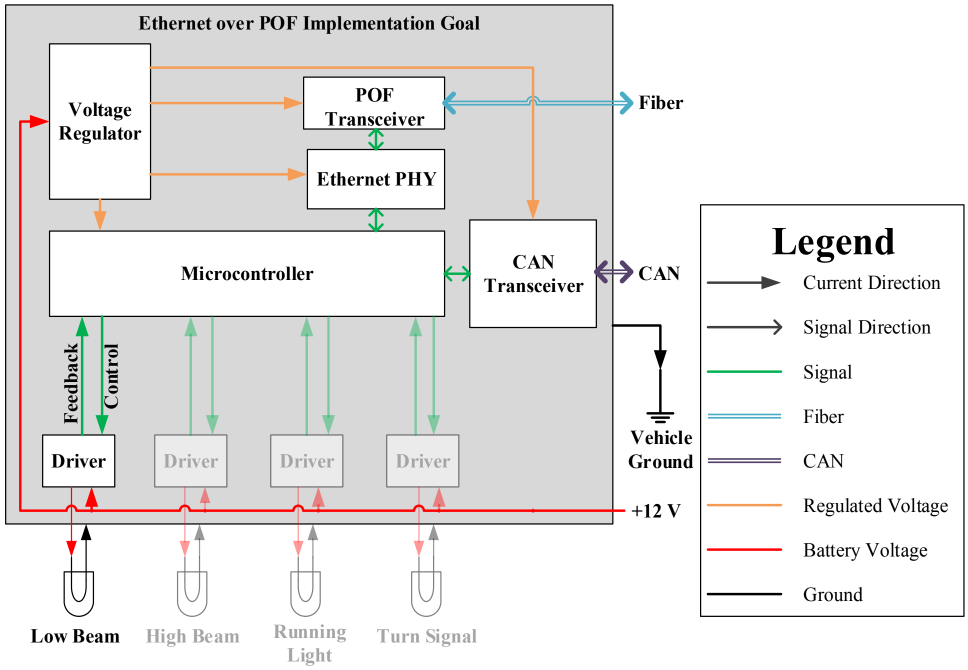

Figure 1.4 Ethernet over POF Implementation Goal

CHAPTER II

DATA COMMUNICATION NETWORKS

2.1 The Controller Area Network

A Controller Area Network (CAN) is a broadcast communication protocol released in 1986 by Robert Bosch GmbH [15]. CAN is a low cost network employing a half-duplex, serial communication protocol implemented with a physical layer utilizing differential twisted pair wires, as described in Figure 2.1.

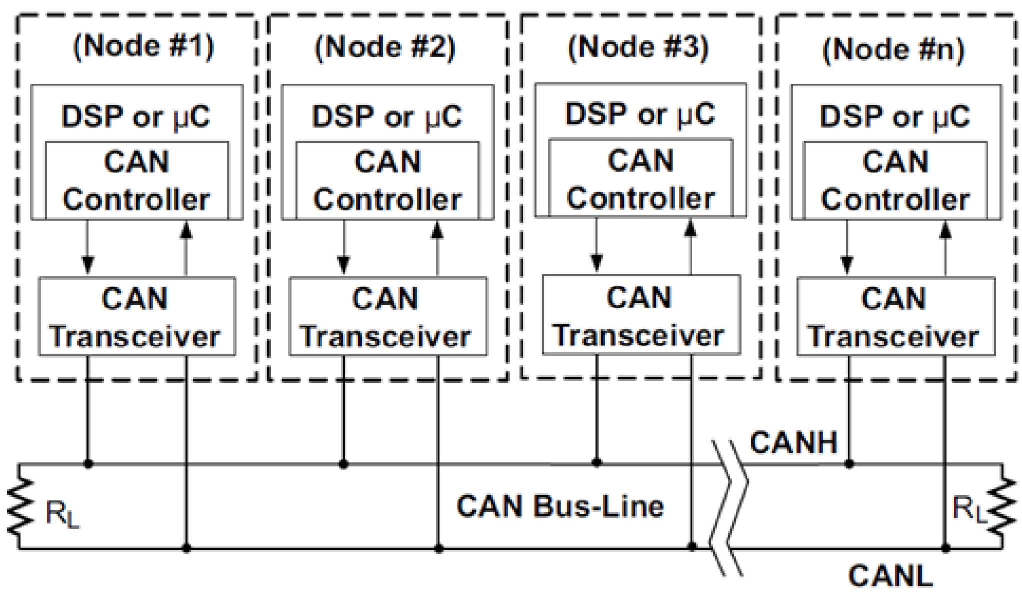

Figure 2.1 CAN Physical Layer [2]

As a broadcast network, every node on the CAN bus has access to all the data being transmitted. The individual messages are identified with either an 11-bit or a 29-bit identifier (ID). A particular node can either read or ignore a message depending on the message's ID. The priority of the message in regards to gaining access to write to the network is specified based on this ID with priority given to the message with the lowest numerical ID [2].

2.1.1 A CAN Message

A CAN message consists of a data frame that includes seven fields and a space field: Start of Frame, Arbitration Field, Control Field, Data Field, Cyclic Redundancy Check (CRC) Field, Acknowledgement (ACK) Field, End of Frame, and Interframe Space [16]. The standard 11-bit ID CAN message is shown in Figure 2.2.

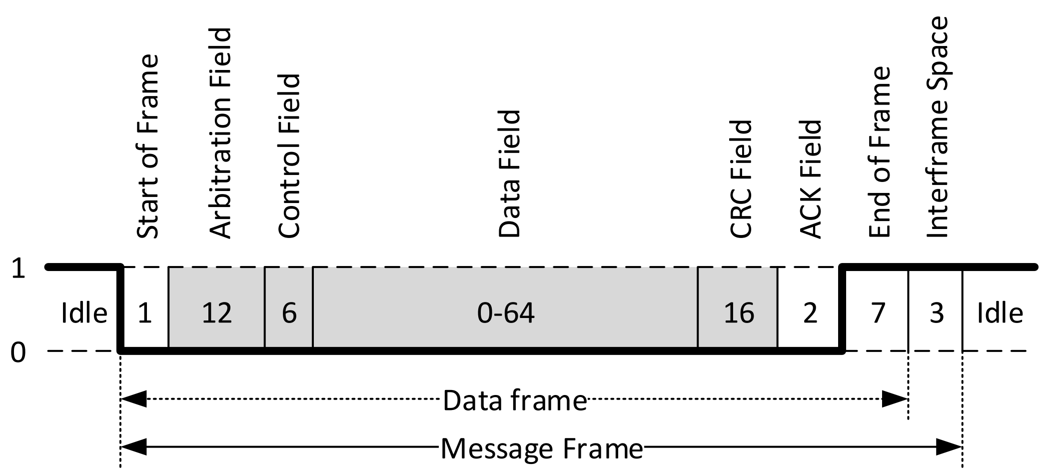

Figure 2.2 CAN Message Data Frame [16]

The arbitration field contains the ID as well as a Remote Transmission Request used to request the remote node to transmit data. The Control Field contains the data length code (DLC) determining how many bytes of data follow. The Data Field is a 1 to 8 byte field that contains the data to transmit. The CRC is used to verify that the data transfer was free of errors. Finally, the ACK field acknowledges that the message was received by transmitting a dominant bit during the second half of the ACK field.

A CAN message is serially transmitted on the bus, sending up to 8 bytes of data. The recipient decodes the messages into sensible information such as voltage, speed, temperature, counters, or flags, for example.

In Figure 2.3, a screenshot of a CAN display using a software product called CANoe and manufactured by Vector Software GmbH shows CAN messages transmitted on the bus. A Vector CANcaseXL analyzer, popular in the automotive industry, can both read and decode CAN messages as well as transmit messages on the bus using the CANoe software.

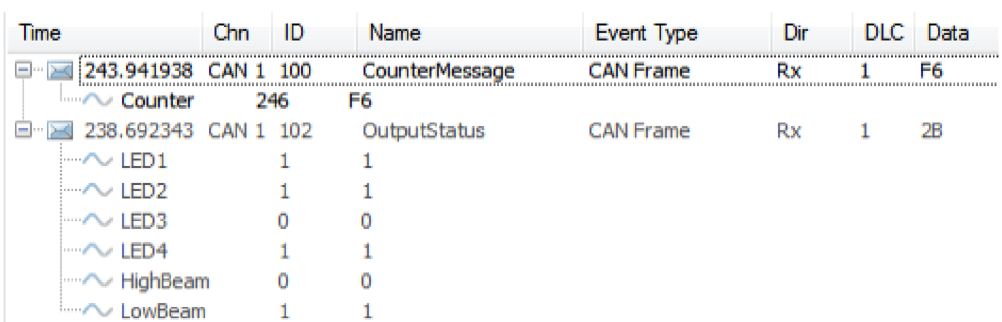

Figure 2.3 CAN messages interpreted with CANoe software

Figure 2.3 shows two messages transmitted from a microcontroller. The first message in the figure, identified as “CounterMessage,” has a DLC of 1, indicating that only one byte of data was sent. The data in this example is converted from the hexadecimal representation, 0xF6, to the decimal representation, 246.

The second message, “OutputStatus,” also has a DLC of 1. However, this message is broken down into six signals from the hexadecimal number 0x2B, which is 0b101011 in binary representation, where the first ‘1’ is for the low beam and the last ‘1’ is for LED1.

2.1.2 CAN Electrical Properties

CAN operates over a differential twisted pair cable with a characteristic impedance of 120 Ω and is terminated at each end with 120 Ω resistors to prevent signal reflections on the bus. Microcontroller can communicate on the bus by utilizing CAN transceivers.

The CAN transceiver implements the physical layer of the Open System Interconnection (OSI) model following the ISO11898-2 standard [17]. The transceiver provides CAN transmit and receive signals for the microcontroller’s communication while providing the differential CAN High and CAN Low signals to communicate on the CAN bus.

2.1.2.1 Differential Pair

CAN has the ability to achieve a high transmission bandwidth over long distances because of its differential pair implementation. The use of a differential pair cancels out the effects of EMI; thereby reducing the number of transmit errors.

CAN interprets the differential signals as containing either dominant or recessive bits. To interpret as a recessive bit, both the high and low signals must be at the same voltage. To interpret a dominant signal, both high and low signals must be at different voltages. In Figure 2.4, the signals are plotted from the same voltage reference level of 0 V located at the bottom of the waveform. As shown in Figure 2.4, the recessive voltage level is approximately 2.5 V, while the dominant levels for CAN Low and CAN High are about 1.5 V and 3.5 V, respectively.

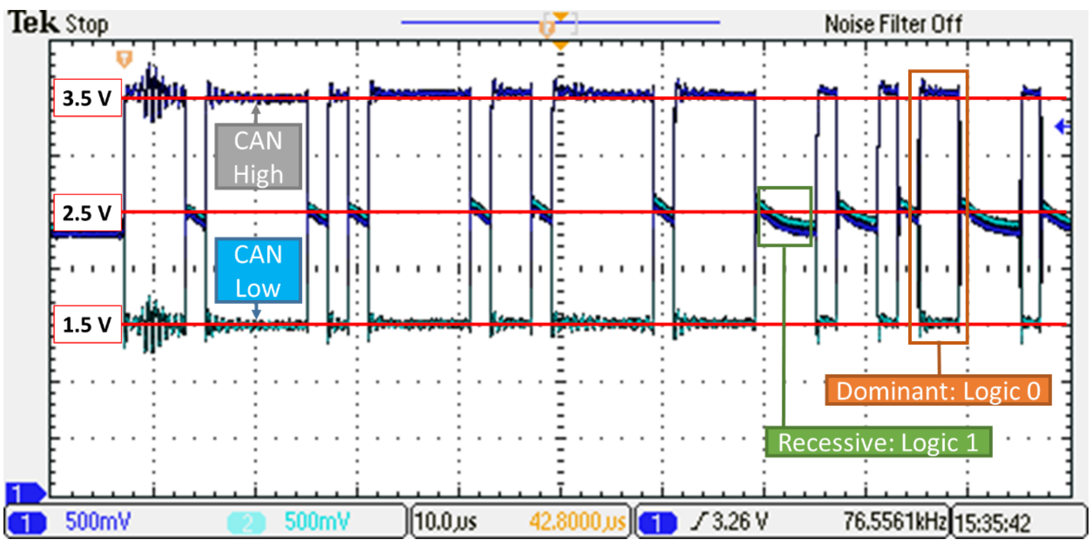

Figure 2.4 Screenshot of CAN data using an oscilloscope

2.1.3 CAN Topology

CAN is a bus network, where all devices or nodes connect to the same two differential paired conductors. Typically, the CAN bus is considered a single line structure where each node is daisy chained from one node to another as illustrated in Figure 2.5. It is important to note that the limit of the number of devices on the bus is not five nodes as may be interpreted from the figure below, but is limited by the electrical current a CAN transceiver can drive.

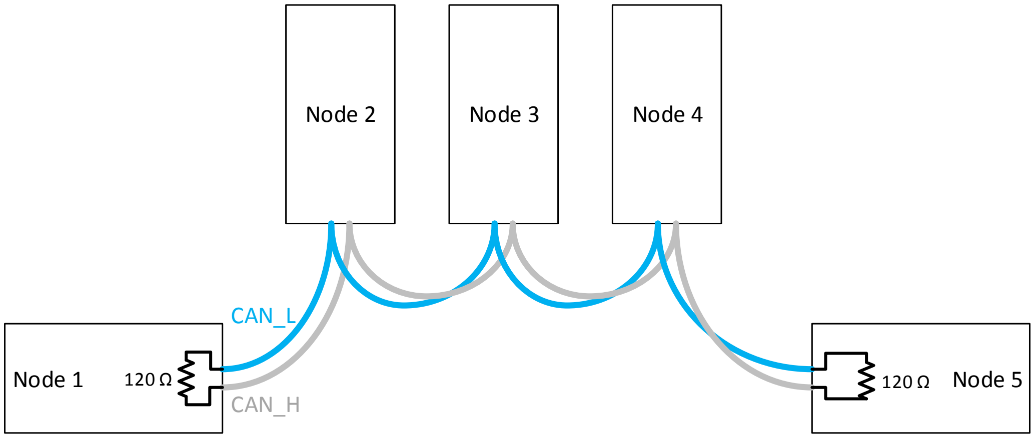

Figure 2.5 Example of CAN bus topology

The CAN bus is analogous to a trunk line that connects two major points each located at an end of the line. Branches or “stubs” are located at various points on the line to provide access to the system. In the case of the CAN bus, the stubs must be sufficiently short so that they do not influence the integrity of the system. The stubs must be unterminated so that they do not appear to be alternate paths for the bus [2].

2.1.4 Example use of CAN in a vehicle

Every vehicle sold in the United States since 1996 has included an On Board Diagnostics II port (OBD II) based on the Society of Automotive Engineers (SAE) standard J-2284/3, which incorporates several communication interfaces into a standard connector interface located within three feet of the driver [18]. Part of a 2008 revision of the SAE J-2284/3 standard requires the use of CAN based on the International Organization for Standardization (ISO) 15765-4 standard, which defines the high-speed CAN for vehicle applications to operate at 500 kbps when used with the OBD II port [19].

Vehicle CAN busses can transmit hundreds of messages and thousands of signals on their networks. For example, a stock vehicle such as EcoCAR 2's 2013 Chevrolet Malibu transmits over 80 separate messages on one of its CAN busses. Many vehicles contain multiple networks to reduce network utilization or to separate critical component control from non-critical component control. A sample log from the Malibu is shown in Figure 2.6. Specific database files are needed to decode the messages from CAN into signals, which have been redacted from Figure 2.6 due to confidentiality agreements with the manufacturer.

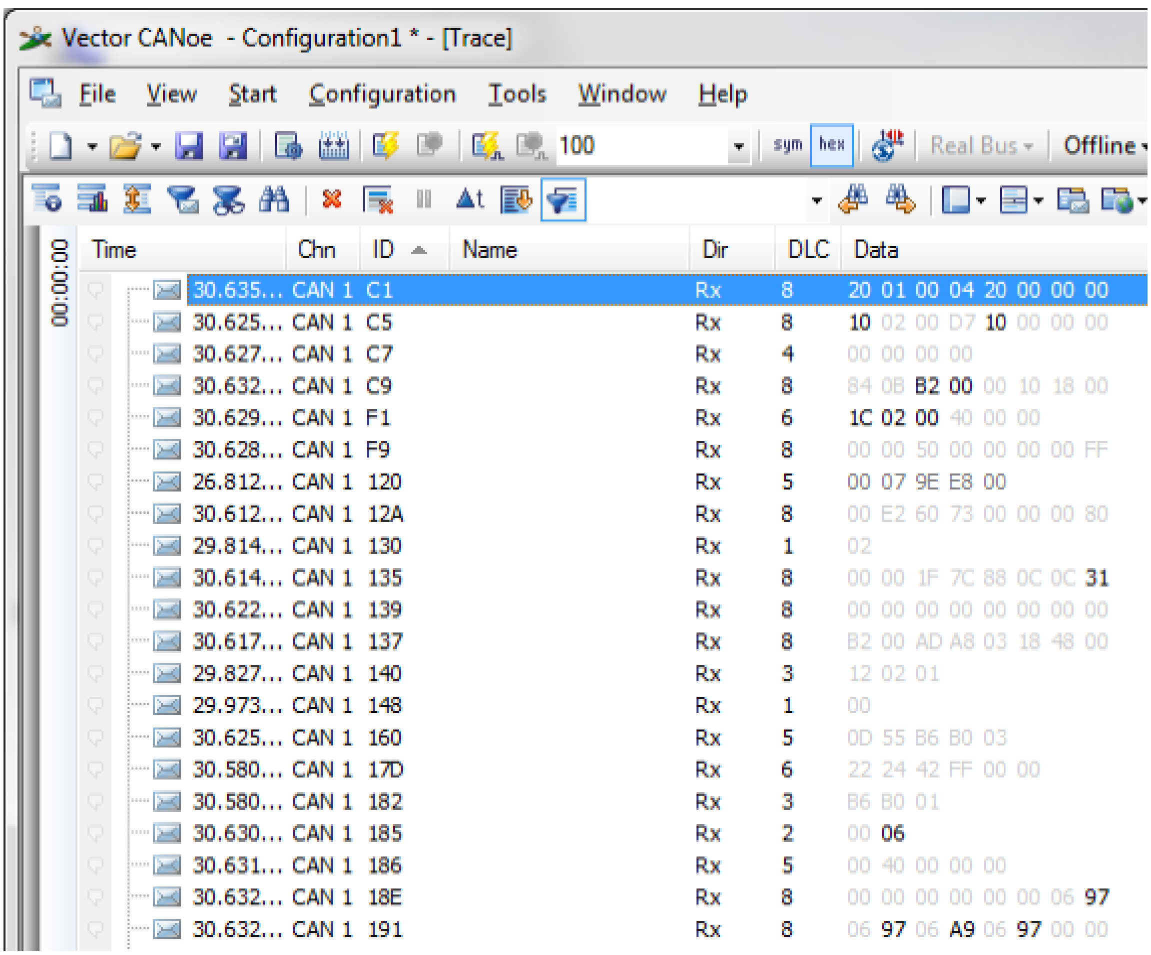

Figure 2.6 Use of CAN in a vehicle using Vector CANoe

2.2 Fiber Optics

Fiber optics is a physical media in which light is used to transmit data. Alexander Graham Bell, inventor of the telephone, first conceived the method of using light to transmit data in 1880 by using sunlight reflecting onto a diaphragm that vibrated with sound [20]. Reflections from the sunlight would travel through an open medium to a photosensitive parabolic reflector made of selenium that would vibrate, recreating the sound, marking the invention of the photophone [20]. Later, the invention of the light emitting diode (LED), laser diode, and glass optical fibers with attenuation under 20 dB/km enabled the use of fiber optics as a network medium in which to transmit data.

Today, fiber optics operate using a few different standard wavelengths such as 850 nm, 1310 nm, and 1550 nm; the cable length is one of the determining factors. Each wavelength exhibits a different signal attenuation over the fiber optic cable as described in Table 2.1. The table shows that any of the three wavelengths will perform well for an automobile.

Table 2.1 Fiber Optic Cable Attenuation at Different Signal Wavelengths

Wavelength (nm)	Attenuation (dB/km)
850	3
1310	0.5
1550	0.2

2.2.1 Fiber Optics in the Automotive Industry

Fiber optics have been used in the automotive field before. The MOST protocol can work over fiber to distribute media and control devices throughout a vehicle for multimedia purposes.

The MOST network's ring topology would not be an appropriate topology for the control network of an automobile. A ring topology's resilience is poor due to its multiple points of potential failure. With CAN, if one node fails, the bus will likely still work; however, with a fiber optic ring network, if one node fails, the entire network will fail. To improve reliability, the fiber optic control network would need to be implemented as a star topology, such as shown in Figure 2.7.

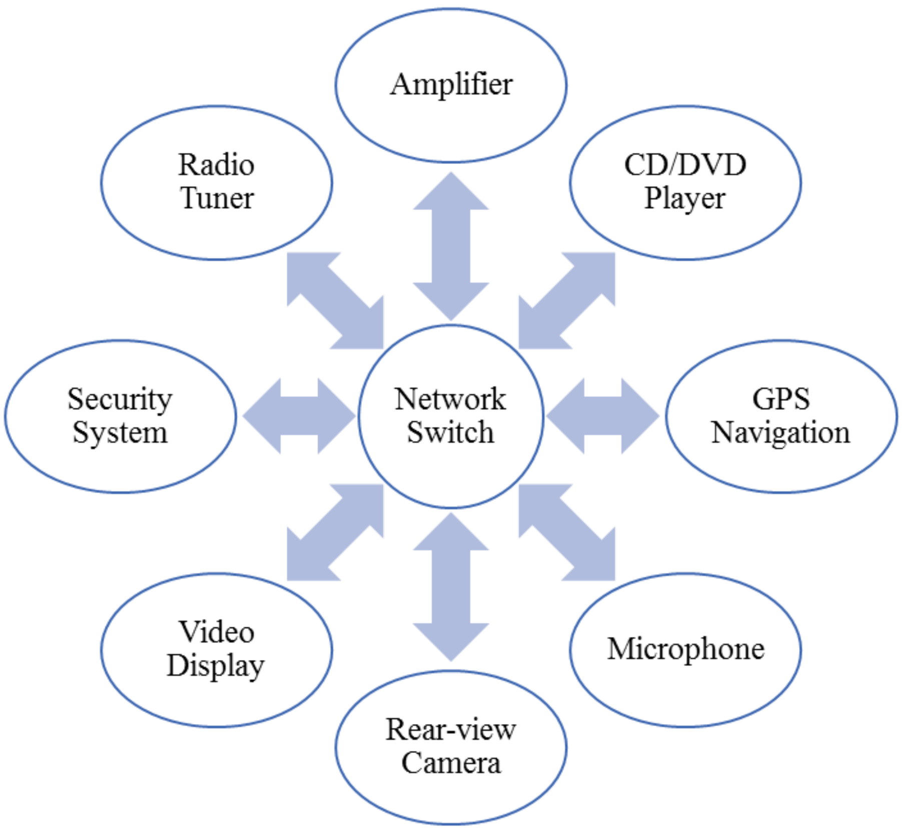

Figure 2.7 Example of a fiber control network star topology

The star topology reduces the number of points of potential failure from n-nodes (total number of nodes in the network) down to one node. That single point-of-failure is the network switch, which is responsible for receiving and sending the data to the appropriate device.

2.2.1.1 Advantages

Fiber optics have the potential to improve many aspects of the current control network such as immunity from EMI, reduction of weight, increase of link speed, reduced packaging space, and improved security.

2.2.1.1.1 Immunity from EMI

Since light is not affected by EMI, fiber optics systems are immune to EMI, except at the source and detector where electronics convert the light back to electrical signals.

2.2.1.1.2 Reduction of Weight

The weight of electrical wiring in today's vehicles has increased with the inclusion of additional emissions, entertainment, and safety systems. On average, there is 120 kg of wire in a typical modern vehicle, according to several sources [9, 21, 22, 23].

2.2.1.1.3 Increase of Bandwidth

Greater bandwidth is required with the introduction of additional safety systems, such as surround view cameras. In fact, many of the cameras included in current vehicles use Ethernet because of the needed bandwidth [24].

2.2.1.1.4 Reduced Packaging Space

Integrating Ethernet over POF allows many more signals to be transmitted because of the increased bandwidth. Wiring for sensors can be terminated locally within the subsystem area rather than being routed throughout the vehicle, reducing the number of long wires needed to transmit the signals.

2.2.1.1.5 Improved Security

CAN is not a secure means to transmit data. In order to hack CAN, all that is needed is to tap into the CAN High and CAN Low signal wires, which is very easy to do using the OBD II connector. There is also no way to check if an illegitimate device connects to the CAN, where it can fabricate and intercept unauthorized messages in the vehicle.

The interruption from unplugging a fiber optic cable can easily be detected. Furthermore, when using a network switch (rather than a hub), the messages only go to the intended recipient and are not accessible by other devices on the network.

Encryption can also be easily and securely implemented with Ethernet over POF using well-known methods. There are many types of encryption methods already supported with Ethernet including Secure Sockets Layer (SSL) and Transport Layer Security (TLS) [25].

2.2.1.2 Disadvantages

There will be some disadvantages with such a pronounced change in how a vehicle’s control system works and communicates. Such disadvantages include increased cost, increased complexity, reduced repairability, environmental constraints, and a possible reduction of consumer acceptability.

2.2.1.2.1 Increased Cost

The biggest disadvantage to switching the automotive control system from copper wiring to fiber will be increased cost. With an average of 50 ECUs in an average mid-priced vehicle, even a small increase in cost of the nodes will increase the cost of the control system about tenfold. [21].

2.2.1.2.2 Increased Complexity

The use of Ethernet as well as the use of fiber optics to replace most signal wires will increase the complexity of the vehicle's control system. Typically, point-to-point wiring is used for most remote signals within a vehicle and are then connected to an ECU to distribute the signals throughout the vehicle using CAN. The focus of this project is to modularize the vehicle's body sections such as headlight assemblies, door assemblies, and engine assemblies to use individual electronic control units (ECUs) to reduce the number of long cables that are often used for these assemblies. The switch to modularized assemblies will increase the engineering effort by requiring more printed circuit boards (PCBs) to be fabricated as well as more network engineering to ensure the control messages are sent to the appropriate ECU in a minimal amount of time.

2.2.1.2.3 Reduced Repairability

The move to fiber optics will reduce the repairability of the vehicle's control system. While copper wire can be cut and spliced with common tools, fiber optics need specialty cutters and crimpers. Additionally, splicing will attenuate the signal significantly, which may cause enough signal loss to prevent successful transmission. However, with widespread adoption, these specialty tools may become more common.

2.2.1.2.4 Environmental Constraints

Accommodating environmental factors such as heat, dust, moisture, and oils will be more challenging with fiber optics than with copper wire networks. Currently automotive grade fiber optic connectors and cables are limited in use to the interior of the vehicle where they are used in MOST networks for entertainment. The environmental conditions in the engine bay change drastically from very cold in the winter (-40 °C) to very hot right after the engine is cut off on a hot summer day (125 °C) [26]. Additionally, the flammability of materials has to be taken into account if fiber optics are used in the engine bay where the extensive use of polyvinyl chloride (PVC) for jacket material would cause dioxins when burned [27].

2.2.1.2.5 Consumer Acceptability

With the increased cost, higher complexity, and reduction of repairability, consumers may find the benefits of fiber optics to outweighed by the disadvantages, and avoid purchasing vehicles with fiber optic control systems.

2.3 Ethernet

The use of POF is only half the answer to redesigning the automotive control network since POF is the physical medium in which to transmit data. An appropriate protocol is required that is compatible with POF. A commercially viable solution with a much documentation and industry usage is Ethernet over POF. Ethernet was invented in 1973 by Robert Metcalfe [28]. The specifics of the Ethernet protocol are defined in IEEE standard 802.3 [29].

2.3.1 Network Communication

Ethernet can also be described using the OSI 7-layer model [30]. The OSI model describes the 7-layers from high level to low level as Application, Presentation, Session, Transport, Network, Data Link, and Physical. The work described in this thesis uses the four lowest layers: the Transport, Network, Data Link, and Physical layers. All messages are transmitted through the control system should using either the User Datagram Protocol (UDP) or Transport Control Protocol (TCP) in the Transport Layer. UDP is a simple protocol that offers low overhead and low latency, but has no acknowledgement on delivery. TCP guarantees delivery and message order, but requires a three-way handshake requiring significant overhead, inherently taking longer to transmit the data [30]. Typically, UDP is a great choice for data that is streamed from the source at a pre-determined interval that is delay-sensitive, such as what current vehicle controllers do when connected by CAN. Additionally, if a message needs to be broadcast throughout the entire network, communication through Multicast UDP is available.

In order for communication to be possible between devices, data has to be encapsulated by the sending device and decapsulated by the receiving device. At the Application layer, the data packet is called a “message”. The message is encapsulated at the Transport layer by adding a TCP/UDP header to the message thus becoming a TCP/UDP “segment”. The segment is further encapsulated by adding a Network layer header containing the source and destination IP address to create a “packet”. The packet is then encapsulated by adding a Data Link layer header containing the source and destination media access control (MAC) address to create a “frame”. The MAC address is a globally unique 6-byte ID consisting of 3 bytes for an Organizationally Unique

Identifier (OUI), which identifies the manufacturer such as Apple, Broadcom, Cisco, Microchip, and Realtek to name a few. The MAC address is often represented as MM:MM:MM:DD:DD:DD in hexadecimal where the three MM bytes contain the OUI and the three DD bytes contain the network interface controller (NIC) ID. Three bytes allows up to 2^{24}-1 or about 16 million unique IDs, meaning there can be 16 million unique OUIs and 16 million unique NIC IDs for each OUI. Both of these combined allows for 2^{48}-1 or about 281 trillion unique devices. The frame is converted into a bit stream at the Physical Layer and then transmitted on the physical medium.

The Physical layer consists of the physical electronics that are used to transmit the data through the medium; whether it be traditional Ethernet using an RJ45 connector and quad twisted pair copper wire, Ethernet through POF, or wireless through Wi-Fi. However, the Physical layer is not just hardware; the Physical layer controller has software registers controlled by the microcontroller or computer, meaning that the Physical layer has some software attributes.

2.3.2 An Ethernet Message

At the transport layer, a UDP segment acts as a liaison between the upper layers and the lower layers, enabling end-to-end communication. The header that is added to the data payload by the Transport layer is shown in 0. The header contains the source and destination port, length of the data, and the data's checksum. The gray shading represents the payload from the user.

Byte 0	Byte 1	Byte 2	Byte 3
Source Port		Destination Port	
Data Length		Checksum	
Data			

Figure 2.8 UDP Segment Format (Transport Layer)

The segment is then encapsulated using an Internet Protocol (IP) version 4 (IPv4) or 6 (IPv6) header in the Network Layer. Only IPv4 will be discussed, since it was used for this thesis. The IPv4 header, shown in 0, contains the source and destination IPv4 addresses. An IPv4 address contains 4-bytes separated into four parts such as 192.168.0.1, for example, allowing up to 2^{24}-1 or about 16 million unique addresses.

<table>
<tr><th colspan="2">Byte 0</th><th colspan="2">Byte 1</th><th colspan="2">Byte 2</th><th colspan="2">Byte 3</th></tr>
<tr><td>Version</td><td>Internet Header Length</td><td colspan="2">Type of Service</td><td colspan="4">Total Length</td></tr>
<tr><td colspan="4">Identification</td><td colspan="2">Flags</td><td colspan="2">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td colspan="2">Protocol</td><td colspan="4">Fragment Offset</td></tr>
<tr><td colspan="8">IPv4 Source Address</td></tr>
<tr><td colspan="8">IPv4 Destination Address</td></tr>
<tr><td colspan="8">Options</td></tr>
<tr><td colspan="8">Data</td></tr>
</table>

Figure 2.9 IPv4 Packet Format (Network Layer)

The IPv4 packet from 0 is then encapsulated by the Data Link layer that formats the packet into an IEEE 802.3 frame, shown in 0, which is the last step before the message is physically transmitted to the destination device.

<table>
<tr><th>Byte 0</th><th>Byte 1</th><th>Byte 2</th><th>Byte 3</th></tr>
<tr><td colspan="4">Preamble</td></tr>
<tr><td colspan="3">Preamble (continued)</td><td>Start of Frame Delimiter</td></tr>
<tr><td colspan="4">Destination MAC Address</td></tr>
<tr><td colspan="2">Destination MAC Address (continued)</td><td colspan="2">Source MAC Address</td></tr>
<tr><td colspan="4">Source MAC Address (Continued)</td></tr>
<tr><td colspan="2">Length of Data Field</td><td colspan="2">Data</td></tr>
<tr><td colspan="4">Data</td></tr>
<tr><td colspan="4">Frame Check Sequence</td></tr>
</table>

Figure 2.10 IEEE 802.3 Frame Format (Data Link Layer)

The Data link layer passes the frame to the Physical layer where the frame is physically transmitted as bits to the remote device. The network switch will forward the frame to the appropriate recipient, and the recipient will then decapsulate the frame by stripping away the headers and footers as well as performing error checking to ensure the data was transferred successfully. With all the headers and footers removed, the receiving application is buffered with the transmitted data where it can process the data.

CHAPTER III

ANALYSIS

In this chapter, an analysis is presented that compares point-to-point electrical wiring, CAN, and Ethernet over POF concerning cost, installation considerations, security, bandwidth, and weight.

3.1 Cost

The cost of materials is a big driving force in the decision of auto manufacturers for Ethernet over POF to be successful. The costs included in this analysis are from current prices published by well-known electronic distributors such as Allied Electronics [31], Digi-Key [32], Mouser [33], Newark [34], and Waytek [35].

3.1.1 Point-to-point Wiring Cost

Point-to-point wiring is known as primary wiring or hook-up wire in the automotive industry. Primary wiring has the advantage of not needing special termination such as the termination resistors and transceivers required by CAN. However, the disadvantage is that each signal requires an individual wire. This does not include the power and ground needed for the electronic device since each device will need power and ground regardless of communication type. From the analysis performed, automotive grade 18 AWG wire that is rated for the temperature and abrasion specifications needed for automotive applications has an average cost of $0.49 / m.

3.1.2 CAN Cost

CAN requires the use of a twisted pair cable. While point-to-point wiring can only transmit one signal, CAN cables can transmit thousands of signals without increasing the number of wires. The analysis concluded that the average cost for CAN cable is $3.35 / m. However, each ECU needs a CAN transceiver that has an average cost of $0.56.

3.1.3 Ethernet over POF Cost

Ethernet over POF offers much lower cable cost, but the transceiver is considerably more expensive. There are considerably fewer suppliers for the POF transceivers and controllers than for CAN transceivers. Since Ethernet typically uses a star network topology, each end of the POF cable needs a transceiver, which again increases the cost.

Cost analysis was performed for the transceiver, PHY controller, and POF cable. The analysis found that the average cost for the transceiver, PHY controller, and POF cable is $18.06 / transceiver, $1.58 / controller, and $1.04 / m, respectively. Therefore, the total cost per node is $19.64 with a cable length cost of $1.04 / m.

3.2 Installation Considerations

Installations considerations involve how the control system is installed in the vehicle and the guidelines that must be accommodated. Considerations include the installation space, bend radius constraints, and environmental constraints.

3.2.1 Point-to-point Wiring Installation Considerations

Point-to-point electrical wiring requires the largest installation space of the networks considered since each signal requires its own individual wire. If the control system requires only a few wires, such as older vehicles that did not contain ECUs, then it is appropriate to use point-to-point wiring. Newer vehicles have ECUs to decrease the emissions, and improve performance, safety, and comfort. If newer vehicles only used point-to-point wiring, it would be impractical to manage the large number of wires.

With point-to-point wiring, the bend radius is not a significant concern. Obviously, sharp bends in wires that conduct substantial current should be avoided because of possible damage to the insulation and an ensuing fault, but for signal wires, the bend radius does not affect the signal.

As for environmental constraints, copper wire has been used for all automotive applications. Copper has a melting point of 1083 °C [36] making it appropriate to use in the engine bay of a vehicle which can have temperatures up to 1050 °C [37]. The copper itself is usually is not the issue with the wire; instead it is the surrounding insulation. Most wire insulation, otherwise known as the jacket, is made of polyvinyl chloride (PVC) that is rated up to 105 °C. High temperature rated wire may use a braided fiberglass jacket to increase the allowable operating temperature. Corrosion is a big environmental factor to consider. Copper is very prone to corrosion and oxidation; therefore, tinned copper wiring is very common. Additionally, weatherproof connector housings are found in the engine bay to protect terminals from moisture and debris.

3.2.2 CAN Installation Considerations

CAN operates over a single twisted pair cable that is much larger than a single wire. However, the ability to carry multiple signals over a single cable allows more signals to be transmitted using less physical space. Currently CAN is the most widely used standard used to communicate between different ECUs within the vehicle. There is a no required minimal bend radius for CAN cable, but it is not recommended to be bent tightly, just like point-to-point wiring.

The environmental constraints for CAN cable are very similar to point-to-point wiring because CAN cable utilizes copper wire. One other environmental factor to consider is EMI if the CAN cable is placed too close to high voltage motors or their power cables such as those used in hybrid vehicles.

3.2.3 Ethernet over POF Installation Considerations

Ethernet over POF operates over fiber optic strands that are much smaller than point-to-point wire or CAN cable. To protect the fibers, three layers of material are used around the core fiber: 1) cladding – used to reflect the bouncing light waves, 2) buffer material – used for strain relief and core protection, and 3) the jacket – the outside material that keeps the inner three components protected.

Bend radius is a concern with fiber optics. The inner core is made of a glass or plastic fiber that can shatter or snap if pinched or deformed. The signal is also attenuated when bent because it has to make more reflections off the cladding surface. The recommended bend radius for POF is 25 mm in order to achieve less than 0.5 dB of signal loss.

Environmental factors change regarding POF compared to the copper wire. POF is typically rated from -55 °C to 85 °C; however, some high-temperature POF is rated from -55 °C to 105 °C. These temperature ranges are representative of POFs sold by numerous sources and detailed in appendix A. However, POF has a huge advantage with regard to corrosion. Unlike copper, which is highly prone to corrosion, POF does not react with moisture or other common chemicals within a vehicle. Like with point-to-point and CAN, keeping the connection points clean and dry is important since dust and moisture can significantly attenuate the signal.

3.3 Security

Security has become a huge news media topic lately. Access to the control system has been attributed to different points of attack including:

- Telematics systems [13] – Services such as On-Star, UConnect, Starlink, HondaLink, Ford Telematics, BlueLink, and many others can be vulnerable.
- OBD II Port [14] - Adapters purchased for diagnostic purposes or those provided by insurance companies or employers to track employee activity provide direct CAN bus access.
- Media player sources – Infected MP3 songs, Bluetooth, Wi-Fi can run malicious programming on the vehicle's media player.
- Wireless Key Fobs – Unsecure key fobs can be analyzed and their signals replicated to gain access to a vehicle. This is less common than previously, due to the three-way handshaking authentication used in today's vehicles.

3.3.1 Point-to-point Wiring Security

Older vehicles that used point-to-point electrical wiring for all of their signals were very vulnerable to being "hot-wired". Hacking a vehicle with point-to-point wiring requires physical access. Today, if a system designed to include telematics with point-to-point wiring was hacked, only the signals whose wires were compromised would be affected, which makes it more secure than CAN in that regard.

3.3.2 CAN Security

CAN is the least secure system to use when compared to point-to-point and Ethernet over POF. The main security feature of CAN is "security-through-obscurity" such that manufacturers do not release CAN message information that is not relevant to OBD-II communication in order to prevent unauthorized manipulation of CAN bus messages. However, the capability to control almost every function of a vehicle through the CAN interface is present since most ECUs communicate using CAN and diagnostic features often have full, unbounded control of vehicle functions. Decoding CAN messages takes some time and skill, but is completely possible to do. The worst security issue with CAN is when manufacturers connect CAN directly to a cellular telematics system, allowing any CAN message to be generated or read remotely. A security flaw using this method was demonstrated using a 2014 Jeep Cherokee [13]. Another security flaw was recently discovered through a device called the Mobile Device dongle commonly used by insurance companies or by employers keeping track of their assets [38].

3.3.3 Ethernet over POF Security

Neither Ethernet nor POF are secure unless appropriate procedures are adopted from the inception. Ethernet, of course, has been used to connect computers to networks and the internet for decades. Viruses and other malware can spread quickly through the internet, infecting computers worldwide. How, then, do manufactures secure Ethernet? One method used only opens the software ports needed for communication. Another method is to enforce encrypted links so that raw data is never transmitted through the network. One great advantage of using Ethernet is that messages are only sent to the intended recipient, automatically removing many of the hacking opportunities present with bus systems.

3.4 Bandwidth

Bandwidth, the amount of data that can be transmitted over the network within a specific time, only involves networks; therefore, point-to-point wiring is not relevant for comparison in this section.

3.4.1 CAN Bandwidth

CAN bandwidth is generally 500 kbps because of OBD-II standards. The throughput can reach up to 1 Mbps using the CAN 2.0 standard. A new CAN standard, CAN FD, is able to support speeds up to 8 Mbps as long as the cable can support such speeds [39].

3.4.2 Ethernet over POF Throughput

Ethernet over POF is capable of reaching speeds up to 10 Gbps, but the hardware and software needed to take advantage of such speeds does not exist yet in the automotive world. The low cost Ethernet over POF that has been examined for this thesis operates at 100 Mbps. Currently, automotive manufacturers are able to control vehicles using a variety of different protocols such as CAN, FlexRay, and LIN, without exceeding the bandwidth limits of any one of the network interfaces. However, faster protocols such as Ethernet are being used for video transmission in vehicles that implement camera based security systems that require higher bandwidth to operate.

3.5 Weight

Weight of the control system is a main challenge that this thesis intends to address because the control system in modern vehicles is increasing in weight that decreases fuel economy. While the simple switch from CAN to POF will not considerably reduce the weight, the number of point-to-point wires that POF can remove can yield a significant weight reduction.

3.5.1 Point-to-point Wiring Weight

Electrical point-to-point wiring is the majority of the wiring weight in current vehicles, since all power wires going to devices such as the fuse panel, starter, battery, headlights, power windows, and many other components are point-to-point wiring of different wire gauges. The 18 AWG wire used for comparison in this thesis weighs 20 g/m. However, larger power wires, for example, are used for high power devices such as the starter, radiator fans, blower motors, headlights, fuel pump, and the rear defroster.

3.5.2 CAN Weight

CAN cable is significantly heavier than 18 AWG wire since it has twisted pair conductors, a shield wire, foil shielding, and thick insulation. The Waytek CAN cable used by the MSU AVTC team measures 48 g/m, significantly heavier than the 18 AWG wire used in point-to-point connections. Additional electronics such as CAN transceivers, microcontrollers, discrete electronics, and an enclosure that make up an ECU add to this weight. Identifying an average weight for a controller is difficult because of the various applications in which they are employed, such as controlling the engine, door locks, climate control system, or the stereo system.

3.5.3 Ethernet over POF Weight

The POF cable is the lightest material out of the three described, at just 13.3 g/m. However, because Ethernet over POF it is implemented as a star network, there are more cables required.

Overall, CAN and Ethernet over POF will weigh the same at the nodes because of the needed electronics. The advantage of the latter is that Ethernet over POF has is that it can transmit more signals than CAN, and therefore, can reduce the amount of wire needed by only supplying a node with power, ground, and Ethernet through POF.

CHAPTER IV

PROTOTYPE

4.1 Prototype Goal

The prototype's goal is to indicate whether it is feasible to integrate Ethernet over POF into an inexpensive microcontroller to accomplish various tasks such as switching high current loads, reading sensors, or transmitting data to another device. The prototype must be able to have a quick boot time of less than 2 sec, which excludes any embedded computer solutions with a Linux operating system.

To provide backwards compatibility, a hybrid control network is designed that supports both Ethernet over POF as well as CAN. CAN support is needed for devices such as EcoCAR 3's battery pack, as well as OBD II connector compatibility.

4.2 Component Selection

The components in Table 4.1 were selected for use in this thesis project based on ease of integration and risk.

Table 4.1 Component Selection Table

Description	Manufacturer	Part
Microprocessor Board	Microchip	PIC32 Ethernet Starter Kit II (ESKII)
POF Transceiver	Avago	AFBR-5803AZ
Ethernet PHY	IC Plus	IP101G
CAN Transceiver	Microchip	MCP2551
Incandescent Bulb High-Side Driver	STMicroelectronics	VN5E010AH

The Microchip PIC32 Ethernet Starter Kit II (ESKII) is unique as the PIC32 microprocessor integrated in the ESKII supports Ethernet, CAN, and MPLAB Harmony. MPLAB Harmony allows for rapid development by providing pre-built drivers and software that can be selectively integrated into the code [40, 41]. The ESKII is also unique because the development board offers various off-the-shelf PHY controllers that can be interchanged. This allows for firmware development using a known working hardware and software configuration; this is a great tool to have when developing both custom hardware and software when issues occur.

The IC Plus IP101G Ethernet PHY was selected for a few reasons. First, this PHY supports both the Reduced Media-Independent Interface for Ethernet, which is the only type of Transport layer to Physical layer interface supported by the PIC32 while also supporting 100Base-FX used for fiber optic communication [42]. The IP101G is also unique in that the PIC32 ESKII has a commercially available daughter board that uses this PHY. Additionally, Avago actively supports the IP101G by providing application notes on integrating the PHY with their transceivers [43, 44].

The ST VN5E010AH high-side MOSFET driver is distinct among high-side drivers in that it features 10 mΩ of on-state resistance, dissipating only 0.21 W of heat when driving a 55 W load such as a headlight [45]. Currently, automobile manufactures commonly use mechanical relays that utilize an electromagnet to switch physical contactors within an enclosure. Relays are larger, more expensive, and have a higher rate of failure than solid-state devices because of their mechanical components.

4.3 Hardware Overview

The hardware used in this investigation was integrated into a prototype setup using the PIC32 ESKII and I/O expansion board, breadboard, high-side driver, and POF daughter board. Figure 4.1 shows an overview of the prototype system that was developed. Appendix B contains more detailed schematics and board layout designs.

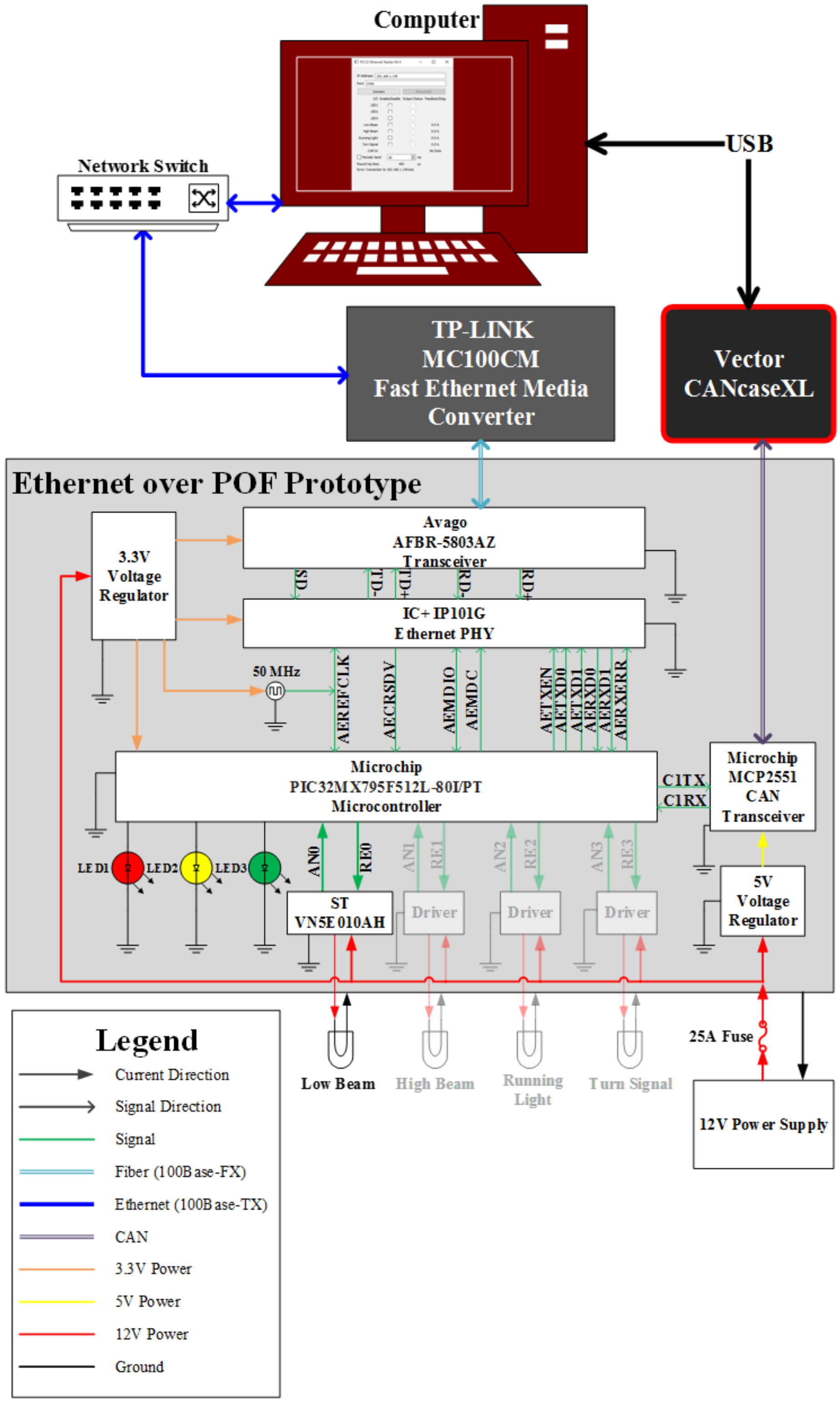

Figure 4.1 System overview

The prototype design hardware in the figure is included within the gray “Ethernet over POF Prototype” box with all the programmed signals shown. The high-side driver is mounted off-board on a custom milled circuit board with wires attaching it to a fuse panel. Since the high-side driver’s board was custom milled and assembled, only one driver was tested. The lack of all high-side drivers should not affect the prototype results since the programming was still designed as if the other drivers were installed.

The computer and prototype were connected to the same network to enable an Ethernet connection at a speed of 100 Mbps. In addition, the Vector CANcaseXL that was connected to the same computer, allowing for CAN communication between the computer and prototype at a rate of 500 kbps.

Figure 4.2 shows a descriptive photo of the prototype setup. The fiber optic daughter board plugs directly into the microcontroller board. The microcontroller board then connects to the I/O expansion board that is powered from the fuse panel. The breadboard contains the CAN transceiver and some discrete components used for the high-side driver’s interface to the microcontroller. The CAN transceiver is connected directly to the Vector CANcaseXL.

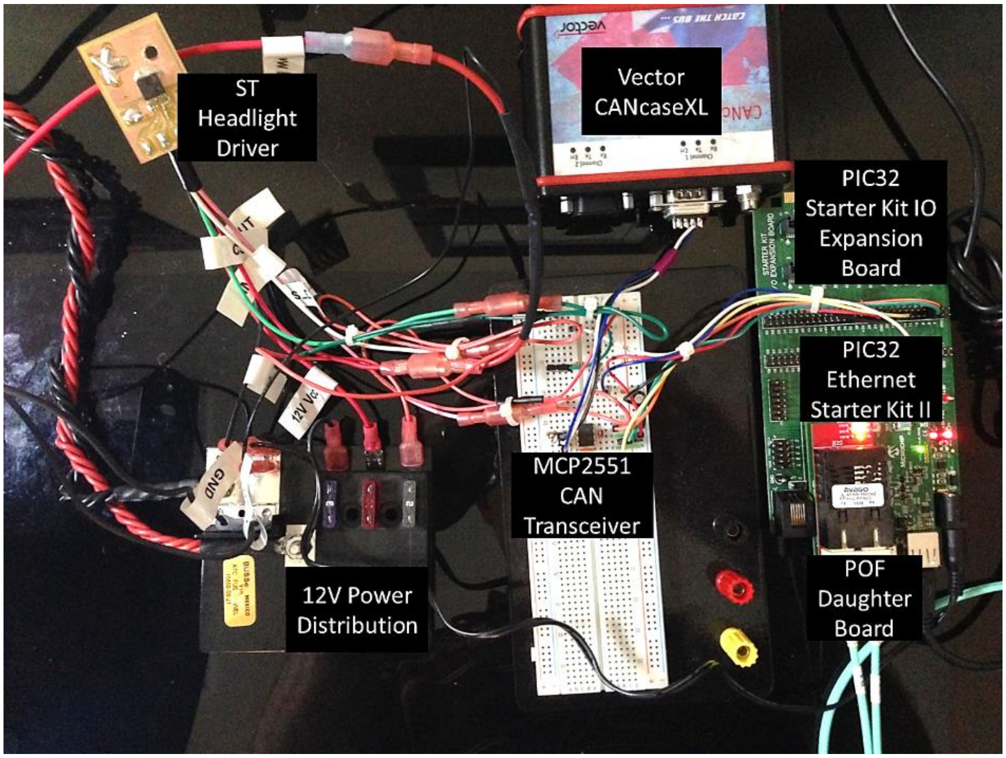

Figure 4.2 Prototype descriptive photo

The overall test setup overview is shown in Figure 4.3. The headlight module can be seen at the bottom left and the 12 V power supply can be seen at the top. At the right is the TP-LINK Media Converter used to convert 100Base-FX to 100Base-TX, which converts the fiber Ethernet from the microcontroller to the copper Ethernet cable that is connected to the computer.

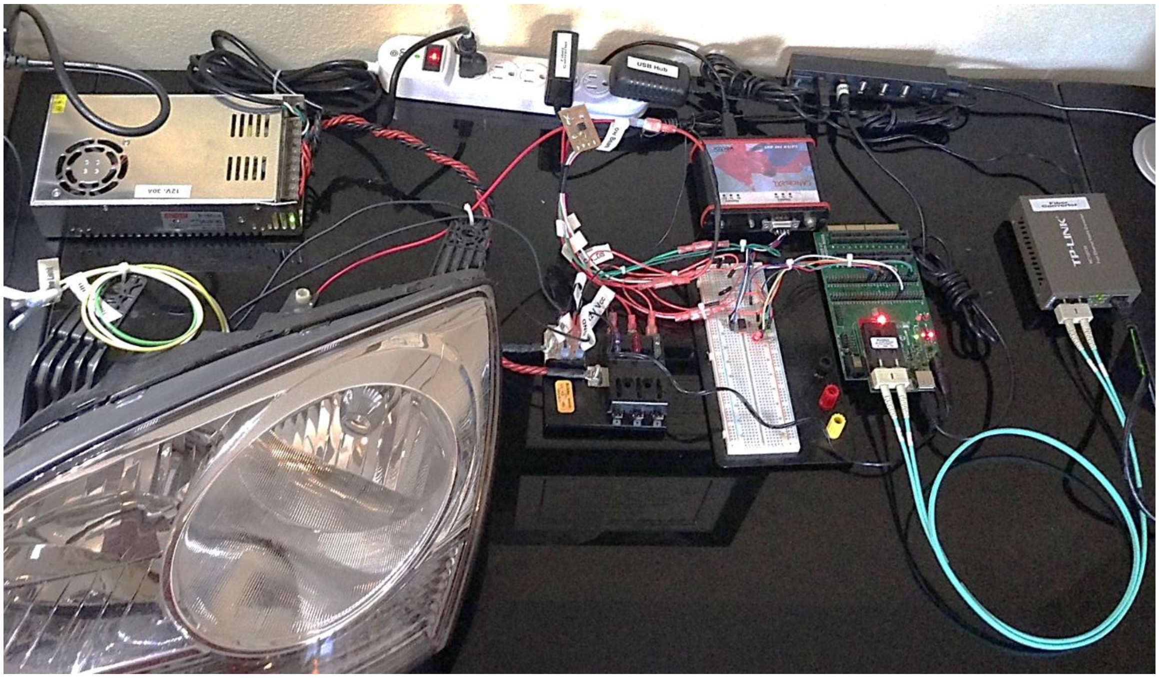

Figure 4.3 Test setup overview photo

4.4 Software Overview

Two software programs were developed for the prototype. The first program controls the prototype through an Ethernet connection using the control computer. The second program is the firmware on the microcontroller that receives messages from the computer from Ethernet and CAN, controls the outputs of the prototype, and reads feedback data from the high-side drivers.

The software communicates over Ethernet between the control computer and the microcontroller board. UDP segments are transmitted in both directions. The segment sent from the control computer contains the control information to switch the output drivers. The segment sent from the microcontroller contains status information including output states, driver current, driver diagnostics, and CAN data. The information from the microcontroller is transmitted to the control computer where it is displayed on the program's GUI window as shown in Figure 4.4. Appendix C contains more detailed information regarding the software.

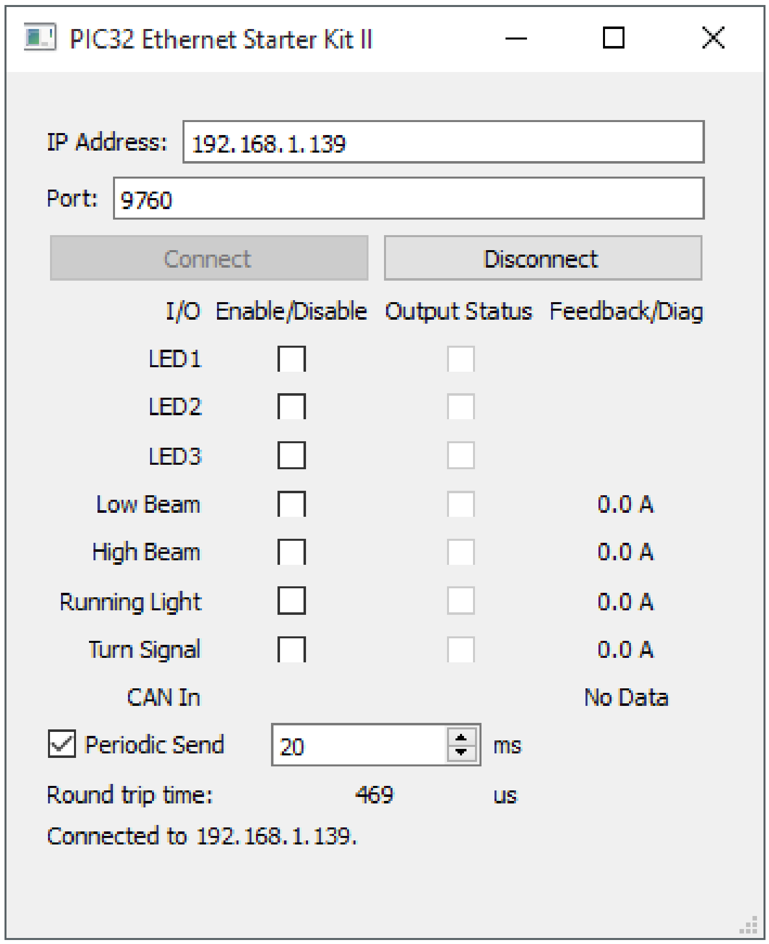

Figure 4.4 Python GUI connected to PIC32

In the left column of the GUI is the I/O list, which names what output is being affected or described. The "Enable/Disable" column will switch the particular output on if checked and off if unchecked. The "Output Status" column reports whether the output should be on or off based on the microcontroller's firmware. Finally, the

“Feedback/Diagnostic” column will display the output current for the high current outputs that use the high-side driver, or any fault that will prevent the driver from operating correctly. The last row of this table is the “CAN In” status, which reads a 16-bit integer sent by the CANcaseXL over the CAN bus.

At the bottom is a periodic send checkbox with a prescribed transmission rate. The frequency with which the Python script sends data to the microcontroller is described by the periodicity specified. Directly below the periodic send is the round trip time that measures the time it takes to send a command and receive a status message.

CHAPTER V

RESULTS

5.1 Analysis Results

The results from chapter III are summarized in Table 5.1, where green indicates the best choice and red indicates the worst choice, based on the results from the analysis.

Table 5.1 Analysis Summary

	Point to Point	CAN	Ethernet over POF
Cost	(red)	(green)	(yellow)
Installation Considerations	(red)	(green)	(green)
Security	(yellow)	(red)	(green)
Bandwidth	(red)	(yellow)	(green)
Weight	(red)	(yellow)	(green)

5.1.1 Cost

The parameters in Table 5.2 were used to evaluate the relative costs of the different control system methods. The parameters were selected using the number of signals present in the EcoCAR 2 Chevrolet Malibu, the average length of a compact sedan and, and a representative number of electronic controllers for a modern vehicle [21, 46].

Table 5.2 Cost Analysis Parameter List

Parameter	Measurement
Signals	500
Vehicle Length	4.5 m
Nodes	50

The cost calculation for each network type is shown in Table 5.3. The resulting costs shown in this table clearly identify CAN as being significantly less expensive than both point-to-point and Ethernet over POF. Point-to-point is the most expensive of the three methods as individual wires, with an approximate length of 4.5 m, are used for each of the 500 signals. Ethernet over POF was only marginally less expensive because of its high transceiver cost. CAN was the least expensive by a large margin because of CAN's low cost transceivers. With such a large difference between CAN and Ethernet over POF, this will severely limit the acceptability of Ethernet over POF.

Table 5.3 Cost Calculations

Type	Equation	Cost
Point-to-point	$500 \times 4.5\,m \times \$0.49$	$1102.50
CAN	$(50 \times \$0.56) + (4.5\,m \times \$3.35)$	$43.08
Ethernet over POF	$(50 \times \$19.64) + (4.5\,m \times \$1.04)$	$986.68

5.1.2 Installation considerations

CAN and Ethernet over POF are very similar in regards to installation considerations. CAN had an advantage with regard to the temperature range, but POF is superior with regard to EMI immunity. Point-to-point was the worst choice due to the physical space requirements and weight.

5.1.3 Security

Ethernet over POF has the potential to be the most secure of the networks considered, since data is only sent to the intended recipient, which prevents other nodes from reading unauthorized data. Ethernet also has available encryption methods such as SSL and TLS. Point-to-point wiring is also secure, because if one signal is hacked, it could only affect that signal; however, it is very easy to hack an individual signal. CAN has the worst security because nearly every ECU in the vehicle is connected to the CAN bus, it is easy to access the bus through the OBD II connector, and there is no way to authenticate a node on the bus.

5.1.4 Bandwidth

Ethernet over POF has the most available bandwidth since it is capable of transmitting 100 times the data as CAN, allowing Ethernet over POF to transmit millions of signals per second. Point-to-point wiring is limited to a single signal.

5.1.5 Weight

Ethernet over POF uses the lightest cable and its network has the least weight. Since both CAN and Ethernet over POF require electronics at each node in order to operate, the weight of each node was ignored and just the weight of the cable was taken into account. The increased bandwidth by implementing Ethernet over POF can remove the need for multiple networks, a solution that vehicle manufactures use to keep the bus utilization of CAN low enough to allow the lower priority messages to transmit without being blocked. Point-to-point wiring is the heaviest, since each signal requires its own signal wire.

5.2 Prototype Results

The prototype was designed to communicate using Ethernet over POF, toggle a headlight, as well as support bidirectional CAN communication. The prototype proved that it is possible to integrate Ethernet over POF with an embedded microcontroller.

Figure 5.1 shows the communication between the control computer and the microcontroller using Ethernet over POF, as well as the microcontroller transmitting messages to the CAN bus, as analyzed with the CANcaseXL.

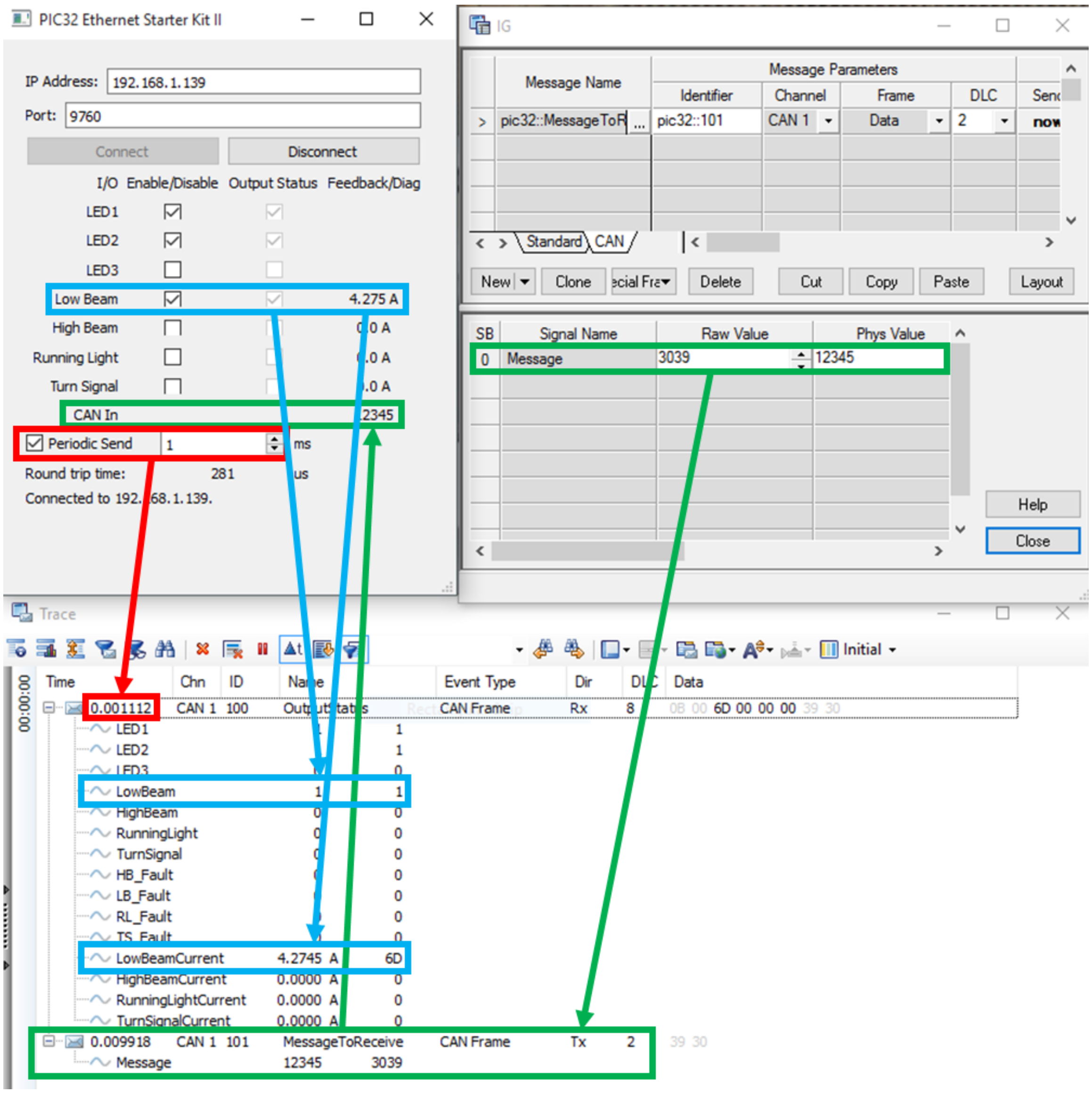

Figure 5.1 Software interaction from Python GUI to PIC32 to CAN

Emphasized in blue in the data log is the low beam output command sent from the control computer to the microcontroller. As can be seen in Figure 5.2, the low beam is illuminated. The current sensor on the ST driver provides a signal to the PIC32's ADC that then records a current of 4.28 A for this demonstration. The output status is sent back to the Python GUI as well as the CANoe software.

Emphasized in green in Figure 5.1 is the CAN message that is sent from CANoe using the CANcaseXL to the microcontroller. The microcontroller retransmits the message's contents back to the control computer using Ethernet over POF.

Emphasized in red is the periodic time parameter that sets the rate at which the control computer transmits a message. As can be seen in the CANoe screen at the bottom of the figure, the period at which the message is transmitted from the microcontroller is 0.001112 sec, or approximately 1 ms. The 1 ms period demonstrates that the microcontroller along with Ethernet over POF can communicate at rates fast enough for the vehicle control network, which send messages to the CAN bus at a rate up to 1 kHz.

Figure 5.2 Low beam on, controlled from Python GUI

CHAPTER VI

CONCLUSION

6.1 Conclusions

The thesis presented an analysis of current issues within automotive control systems and a potential way to fix some of those issues. The results from the analysis section indicated that the cost of implementing Ethernet over POF presents a significant disadvantage over CAN. However, in terms of installation considerations, security, bandwidth, and weight, fiber optics met or exceeded the criteria from point-to-point wiring and CAN. The results from the prototype section indicated that Ethernet over POF could be integrated into a low-powered microcontroller, such as the Microchip PIC32 used in the prototype.

The concept of using Ethernet over POF extends beyond controlling headlights. The intention of this project was to describe a way to create a new backbone network that could be used to control all electrical components of a vehicle. Presented in Figure 6.1 are the different ECUs that would utilize the Ethernet over POF control network.

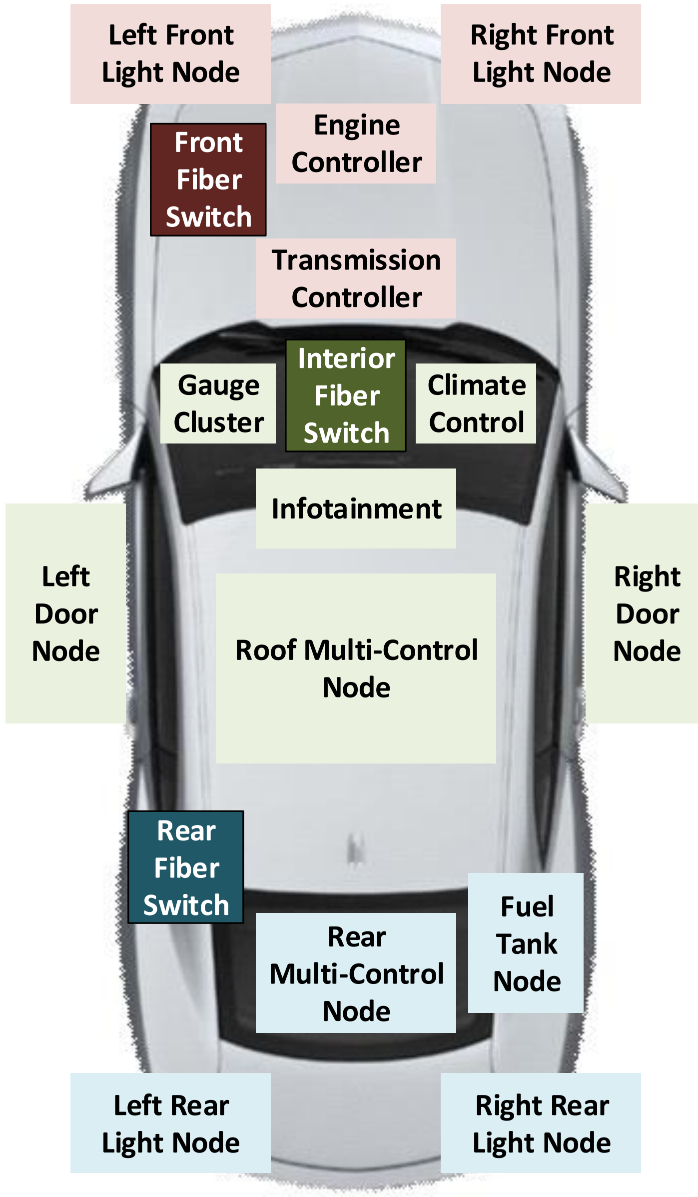

Figure 6.1 Possible nodes in a modern vehicle

These nodes would connect using a multi-switch network to reduce the length of POF cable installed within the vehicle with statically assigned IP addresses for each node. The network would include a front fiber switch, interior fiber switch, and a rear fiber switch as illustrated in Figure 6.2 to reduce the length of POF routed within the vehicle by splitting the star network into different locations.

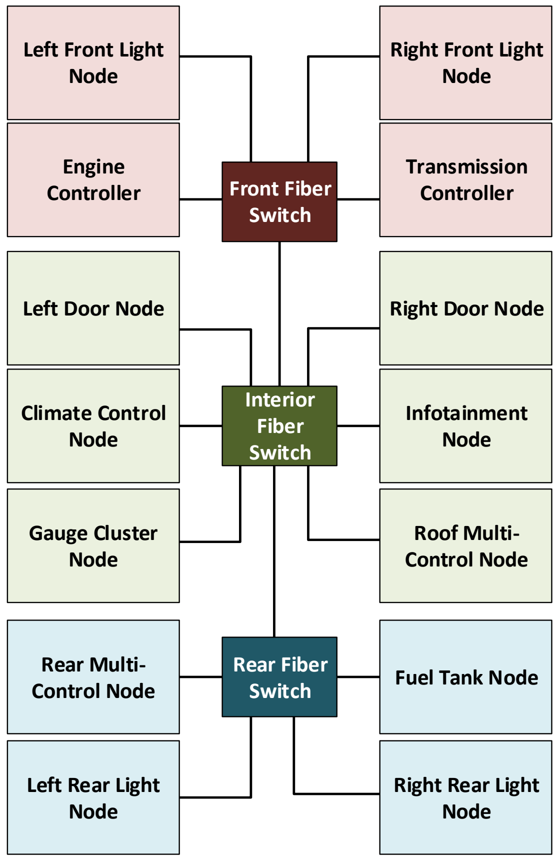

Figure 6.2 POF network diagram

One potential disadvantage of adopting Ethernet over POF would be the changes that would be made to industry standard methods. Currently, OBD II is required to contain CAN on the diagnostic port. However, Ethernet is not a standard that is integrated within the OBD II standard. Either a new OBD standard would have to be adopted or backwards compatibility would have to be designed into the Ethernet network to support the required OBD II protocols through a gateway.

6.2 Future Work

Two outstanding issues remain from the work in this thesis that require additional work before the idea of using Ethernet over POF can be integrated into vehicles. The first outstanding issue is cost reduction since the adoption cost is significantly higher than CAN due to the transceivers. The second outstanding issue is improving the security, which was mentioned within the thesis, but never implemented in the prototype.

6.2.1 Cost reduction

The cost of implementing Ethernet over POF is far too high because of the cost of the fiber optic transceivers. There may be a cost reduction with widespread adoption due to supply and demand. However, the introductory cost may prevent the widespread adoption of Ethernet over POF in automotive. CAN transceivers may be inexpensive because the demand for CAN transceivers is high. CAN transceiver manufacturers are able to pay for the engineering efforts that went into making the CAN transceivers. Fiber optics are not as widely implemented as CAN; therefore, the demand is much lower, leaving manufacturers to increase the cost of the fiber optic transceiver to make up for the engineering time that went into developing the transceivers.

Rather than waiting for the prices of POF transceivers to decrease, possible ways to reduce the cost can be investigated. The materials inside the transceiver do not justify the cost. The transceiver consists of plastic components, a small PCB populated with an LED and photodiode, a photodiode quantizer, an LED driver and discrete components as can be seen in Figure 6.3.

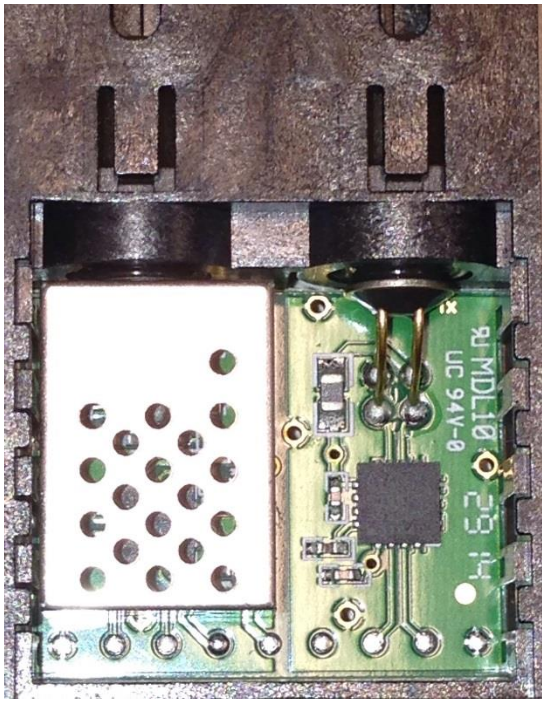

Figure 6.3 Avago AFBR-5803AZ with cover removed

One way to reduce the cost would be to change the interface between the transceiver and the PHY. With such a short trace length, there seems to be no advantage of using a differential pair between the transceiver and the PHY. An LED can be driven by a current source on the microcontroller and the photodiode's input can be read using a high-speed digital input.

6.2.2 Security improvement by using encryption

While the use of encryption was mentioned within this thesis, it was never implemented. MPLAB Harmony has an option to integrate wolfSSL into PIC32 firmware as shown below [25]. wolfSSL provides embedded SSL encryption to microprocessors that communicate using TCP/IP. Integrating this into a vehicle network would provide much higher security as message contents could not be read nor fabricated by an unauthorized node.

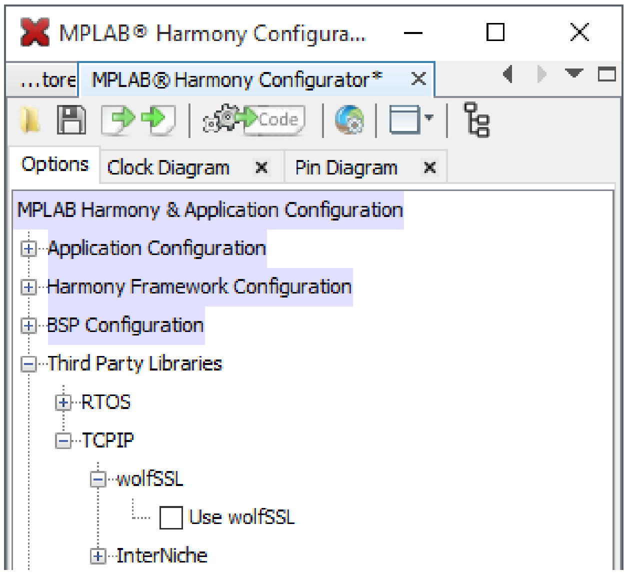

Figure 6.4 wolfSSL integration option in MPLAB Harmony v1.06 Configurator tool

REFERENCES

[1] "50 Years of Moore's Law," Intel, [Online]. Available: http://www.intel.com/content/www/us/en/silicon-innovations/moores-law-technology.html.

[2] S. Corrigan, January 2009. [Online]. Available: http://www.ti.com/lit/an/slla270/slla270.pdf. [Accessed 4 April 2015].

[3] CAN in Automation, "CAN Specification 2.0 part B," 26 11 2002. [Online]. Available: http://www.can-cia.org/fileadmin/cia/specifications/CAN20B.pdf.

[4] National Instruments, "FlexRay Automotive Communication Bus Overview," National Instruments, 21 August 2009. [Online]. Available: http://www.ni.com/white-paper/3352/en/.

[5] MOST Cooperation, "MOST Network," MOST Cooperation, 2015. [Online]. Available: http://www.mostcooperation.com/technology/most-network/.

[6] National Instruments, "Introduction to the Local Interconnect Network (LIN) Bus," National Instruments, 3 11 2011. [Online]. Available: http://www.ni.com/white-paper/9733/en/.

[7] K. Pretz, "Fewer Wires, Lighter Cars," IEEE, The Institute, 12 April 2013. [Online]. Available: http://theinstitute.ieee.org/benefits/standards/fewer-wires-lighter-cars.

[8] P. Hank, "Automotive Ethernet: In-vehicle networking and smart mobility," 18-22 March 2013. [Online]. Available: http://dx.doi.org/10.7873/DATE.2013.349.

[9] D. Lavrinc, "In-Car Ethernet Isn't What You Think," Wired.com, 16 October 2012. [Online]. Available: http://www.wired.com/2012/10/in-car-ethernet/. [Accessed December 2014].

[10] EPA, "Regulations & Standards: Light-Duty," EPA, 18 June 2015. [Online]. Available: http://www.epa.gov/otaq/climate/regs-light-duty.htm.

[11] A. Vasile, "Trends in fiber optics applications for automotive industry," 9-13 May 2007. [Online]. Available: http://dx.doi.org/10.1109/ISSE.2007.4432904.

[12] C.-W. Lin, "Cyber-Security for the Controller Area Network (CAN) Communication Protocol," 14-16 December 2012. [Online]. Available: http://dx.doi.org/10.1109/CyberSecurity.2012.7.

[13] A. Greenberg, "Hackers Remotely Kill a Jeep on the Highway—With Me in It," Wired, 21 July 2015. [Online]. Available: http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/.

[14] A. Greenberg, "Hackers Reveal Nasty New Car Attacks--With Me Behind The Wheel (Video)," Forbes, 12 August 2013. [Online]. Available: http://www.forbes.com/sites/andygreenberg/2013/07/24/hackers-reveal-nasty-new-car-attacks-with-me-behind-the-wheel-video/.

[15] CiA, "CAN History," [Online]. Available: http://www.can-cia.org/index.php?id=systemdesign-can-history. [Accessed 13 4 2015].

[16] Robert Bosch GmbH, Automotive Handbook, 8th Edition, Postfach, Germany: Rober Bosch GmbH, 2011.

[17] P. Richards, "A CAN Physical Layer Discussion," 2002. [Online]. Available: http://ww1.microchip.com/downloads/en/AppNotes/00228a.pdf.

[18] SAE, "High-Speed CAN (HSC) for Vehicle Applications at 500 KBPS," SAE, 2 March 2010. [Online]. Available: http://standards.sae.org/j2284/3_201003/.

[19] SAE, "High-Speed CAN (HSC) for Vehicle Applications at 500 KBPS," SAE International, 2 3 2010. [Online]. Available: http://standards.sae.org/j2284/3_201003/.

[20] Fiber-Optics.Info, "The Nineteenth Century - History of Optical Fiber Technology," Fiber-Optics.Info, 5 December 2008. [Online]. Available: http://www.fiber-optics.info/history. [Accessed July 2015].

[21] K. Pretz, "Fewer Wires, Lighter Cars," The Institute, IEEE, 12 April 2013. [Online]. Available: http://theinstitute.ieee.org/benefits/standards/fewer-wires-lighter-cars. [Accessed December 2014].

[22] G. Rushton, Interviewee, Body Controls Systems Engineering. [Interview]. December 2014.

[23] "Freescale MCUs simplify car wiring system, cut vehicle weight," Automotive EE Times, 20 March 2013. [Online]. Available: http://automotive-eetimes.com/en/65279-freescale-mcus-simplify-car-wiring-system-cut-vehicle-weight.html?cmp_id=7&news_id=222902820. [Accessed 12 May 2015].

[24] ixia, "Automotive Ethernet: An Overview," May 2014. [Online]. Available: https://www.ixiacom.com/sites/default/files/resources/whitepaper/ixia-automotive-ethernet-primer-whitepaper_1.pdf.

[25] wolfSSL, "Documentation," wolfSSL, [Online]. Available: https://www.wolfssl.com/wolfSSL/Docs.html. [Accessed July 2015].

[26] D. R. Sparks, "Application of MEMS Technology in Automotive Sensors and Actuators," in International Symposium on Micromechatronics and Human Science, Nagoya, Japan, 1998, November 25-28.

[27] "Polyvinyl Chloride (PVC)," Tox Town, [Online]. Available: http://toxtown.nlm.nih.gov/text_version/chemicals.php?id=84.

[28] IEEE, "History of Ethernet," IEEE, 2013. [Online]. Available: http://standards.ieee.org/events/ethernet/history.html.

[29] IEEE, "IEEE 802.3 Ethernet Working Group," IEEE, 14 August 2015. [Online]. Available: http://www.ieee802.org/3/.

[30] C. M. Kozierok, 20 September 2005. [Online]. Available: http://www.tcpipguide.com.

[31] Allied Electronics, [Online]. Available: http://www.alliedelec.com/.

[32] Digi-Key Electronics , [Online]. Available: http://www.digikey.com/.

[33] Mouser Electronics, [Online]. Available: http://www.mouser.com/.

[34] Newark, [Online]. Available: http://www.newark.com/.

[35] Waytek Incorporated, [Online]. Available: http://www.waytekwire.com/.

[36] "Melting Points of Various Metals," Online Metals, [Online]. Available: http://www.onlinemetals.com/meltpt.cfm.

[37] Borg Warner, "Exhaust Cas Temperature - 1050 °C - An Engineering Challenge," September 2000. [Online]. Available: http://www.turbos.bwauto.com/tools/download.aspx?t=document&r=105&d=327.

[38] A. Greenberg, "Hackers Cut a Corvette's Brakes Via a Common Car Gadget," Wired, 11 August 2015. [Online]. Available: http://www.wired.com/2015/08/hackers-cut-corvettes-brakes-via-common-car-gadget/.

[39] National Instruments, "Understanding CAN with Flexible Data-Rate (CAN FD)," National instruments, 1 August 2014. [Online]. Available: http://www.ni.com/white-paper/52288/en/.

[40] Microchip, "PIC32 Ethernet Starter Kit II," Microchip, 3 February 2014. [Online]. Available: http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320004-2.

[41] Microchip, "MPLAB Harmony," Microchip, [Online]. Available: http://www.microchip.com/pagehandler/en_us/devtools/mplabharmony/home.html.

[42] IC Plus Corporation, "IP101G Single Port 10/100M MII/RMII/TP/Fiber Fast Ethernet Transceiver," IC Plus Corporation, [Online]. Available: http://www.icplus.com.tw/pp-IP101G.html.

[43] Microchip, "IC Plus IP101G PHY Daughter Board," Microchip, 7 February 2014. [Online]. Available: http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=AC320004-2.

[44] Micorchip, "IC Plus IP101G PHY Daughter Board Information Sheet," 7 September 2014. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/50002210A.pdf.

[45] ST Microelectronics, "ST VN5E010AH," September 2013. [Online]. Available: http://www.st.com/web/en/resource/technical/document/datasheet/CD00240716.pdf.

[46] C. Woodyard, "Vehicles keep inching up and putting on pounds," USA TODAY, 16 July 2007. [Online]. Available: http://usatoday30.usatoday.com/money/autos/2007-07-15-little-big-cars_N.htm.

[47] Avago, "Interfacing Avago Technologies Fast Ethernet Optical Transceivers with various PHY ICs, Application Note 5582," 24 September 2014. [Online]. Available: http://www.avagotech.com/docs/AV02-4572EN.

[48] Avago Technologies, "AFBR-5803Z/5803TZ/5803AZ/5803ATZ," 25 July 2007. [Online]. Available: http://www.avagotech.com/docs/AV02-0606EN.

[49] Microchip, "PIC32MX5XX/6XX/7XX Family Data Sheet," March 2013. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/61156H.pdf.

[50] Vector, "CAN Gets Even Better," April 2013. [Online]. Available: https://elearning.vector.com/portal/medien/cmc/press/Vector/CAN_FD_ElektronikAutomotive_201304_PressArticle_EN.pdf. [Accessed 4 April 2015].

[51] Robert Bosch GmbH, "CAN 2.0 Specification," 1991. [Online]. Available: http://www.kvaser.com/software/7330130980914/V1/can2spec.pdf.

[52] T. Schmidt, "CRC Generating and Checking," 2000. [Online]. Available: http://ww1.microchip.com/downloads/en/AppNotes/00730a.pdf.

[53] Microchip Technologies, "Microchip MCP2551," 2010. [Online]. Available: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en010405.

[54] Digikey, "Texas Instruments SN65HVD1050QDRQ1," Digikey, [Online]. Available: http://www.digikey.com/product-detail/en/SN65HVD1050QDRQ1/296-19876-2-ND/1120747. [Accessed 5 8 2015].

[55] Quotation of HDTR-03-33F12LF, Hi Optel, 2015.

[56] FEMA, "Highway Vehicle Fires (2008–2010)," Topical Fire Report Series, vol. 13, no. 11, p. 4, 2013, January.

[57] D. Kuo, Interviewee, Sales & Marketing at IC+. [Interview]. 11 8 2015.

[58] Waytek, [Online]. Available: http://www.waytekwire.com/item/CB20-11/20GA-Can-Bus-Cable-J1939-11/.

[59] Waytek, "Waytek CAN Bus Data Cables," [Online]. Available: http://www.waytekwire.com/datasheet/CB20-11.pdf.

[60] Znixter, "Cabling Installation Pocket Reference Guide," September 2014. [Online]. Available: https://www.anixter.com/content/dam/Anixter/Guide/12H0005X00-Anixter-Installation-Pocket-Reference-Guide-BOOK-W%26C-EN-US.pdf.

[61] Digikey, " Avago Technologies US Inc. AFBR-5803AQZ," Digikey, [Online]. Available: http://www.digikey.com/product-detail/en/AFBR-5803AQZ/516-2346-ND/2210743.

[62] Freescale Semicondutor, "Lamp Inrush Current Computation Tool for Optimal eXtreme Switch Usage," June 2013. [Online]. Available: http://www.freescale.com/files/analog/doc/app_note/AN4049.pdf?fsrch=1&sr=1.

[63] CadSoft, "CadSoft Home Page," [Online]. Available: http://www.cadsoftusa.com/.

[64] Microchip, "Starter Kit I/O Expansion Board," Microchip, 5 March 2012. [Online]. Available: http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=DM320002&utm_source=MicroSolutions&utm_medium=&utm_term=&utm_content=&utm_campaign=PIC32+Expansion+Board.

APPENDIX A

POF TEMPERATURE ANALYSIS

An analysis of different plastic optical fiber operating temperatures was performed using specifications from various manufacturers and models of POF. A study of what core, cladding, and jacket materials increase or decrease specified operating temperature was performed that yielded lackluster results with the exception of cross-linked polyethylene for jacket material. Cross-linked polyethylene was found in “high-temperature” rated POF, and was rated 20 °C higher than any other POF cable. Table A.1 shows acronyms that are used to describe the material listed in 0. Some material details could not be found and are omitted from 0.

Table A.1 POF Material Acronyms

Material Type	Acronym
Polyethylene	PE
Polyvinyl Chloride	PVC
Fluorinated Polymer	FP
Polymethyl-Methacrylate	PM
Chlorinated Polyethylene	CPE
Polycarbonate	PC
Cross-linked Polyethylene	XLPE

Table A.2 POF Temperature Analysis

Manufacturer	Model	Core Material	Cladding Material	Jacket Material	Low Temperature (°C)	High Temperature (°C)
Industrial Fiber Optics	GHV4002	PM	FP	PVC	-40	85
Industrial Fiber Optics	MH4002	PM	FP	PE	-55	85
Industrial Fiber Optics	MHV4002	PM	FP	PVC	-55	85
Industrial Fiber Optics	GHCP4002	PM	FP	CPE	-55	85
Industrial Fiber Optics	GH4002	PM	FP	PE	-55	85
Industrial Fiber Optics	SHCP4002	PM	FP	CPE	-55	70
Industrial Fiber Optics	VST4002	PM	FP	PE / PVC	-40	70
Industrial Fiber Optics	SHV4002	PM	FP	PVC	-40	70
Industrial Fiber Optics	SH4002	PM	FP	PE	-55	70
Industrial Fiber Optics	SH3002	PM	FP	PE	-55	70
Industrial Fiber Optics	FH4001-V 1	PC	FP	XLPE	-55	105
Industrial Fiber Optics	BH2001 1	PM	FP	XLPE	-55	105
Industrial Fiber Optics	BH4001 1	PM	FP	XLPE	-55	105
Tripp Lite	N549-01K				-20	70
Avago	HFBR-RUS500Z				-55	85
TE Connectivity	501232-3			PE	-40	85

[1] High temperature rated

APPENDIX B

HARDWARE DESIGN

B.1 PIC32 Ethernet Starter Kit II Fiber Optic Daughter Board Design

A PIC32 ESKII fiber optic daughter board was created, since an off-the-shelf product was not available. The interface between the PIC32 microcontroller and PHY operates at 50 MHz, which meant that breadboard development was not possible due to the EMI generated from using unshielded components, unequal wire lengths, and the capacitive effects of breadboard traces at this frequency. The lack of breadboard prototyping required the use of a printed circuit board (PCB) that would integrate the microcontroller, PHY, and POF transceiver into a daughter board compatible with the PIC32 ESKII.

Figure B.1 shows the interface to communicate with the microcontroller using a dual in-line package that inserts directly into the PIC32 ESKII daughter board interface.

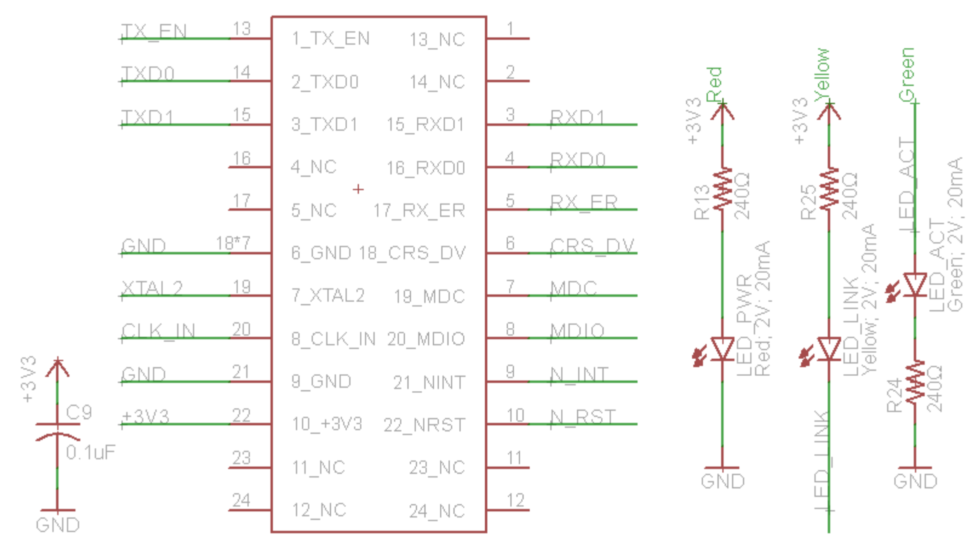

Figure B.1 Microprocessor board-to-board interface and indicators schematic

The PHY circuit that enables communication between the microcontroller and the physical interface is shown in Figure B.2. This circuit enables the PHY's fiber mode using R19 on pin 19: FX_HEN pulled to +3.3 V. The Reduced Media Independent Interface (RMII) is enabled using pin 4 – COL/RMII pulled to +3.3 V. The RMII mode is required for the Microchip PIC32 microprocessor used. To enable the 50 MHz clock input setting of the PHY, pin 2 – X1 is connected to ground and pin 10 – 50M_CLKI is connected to the 50 MHz clock generated located on the microcontroller's main board. Most of the other components are discrete components used to set the LED mode and decoupling capacitors.

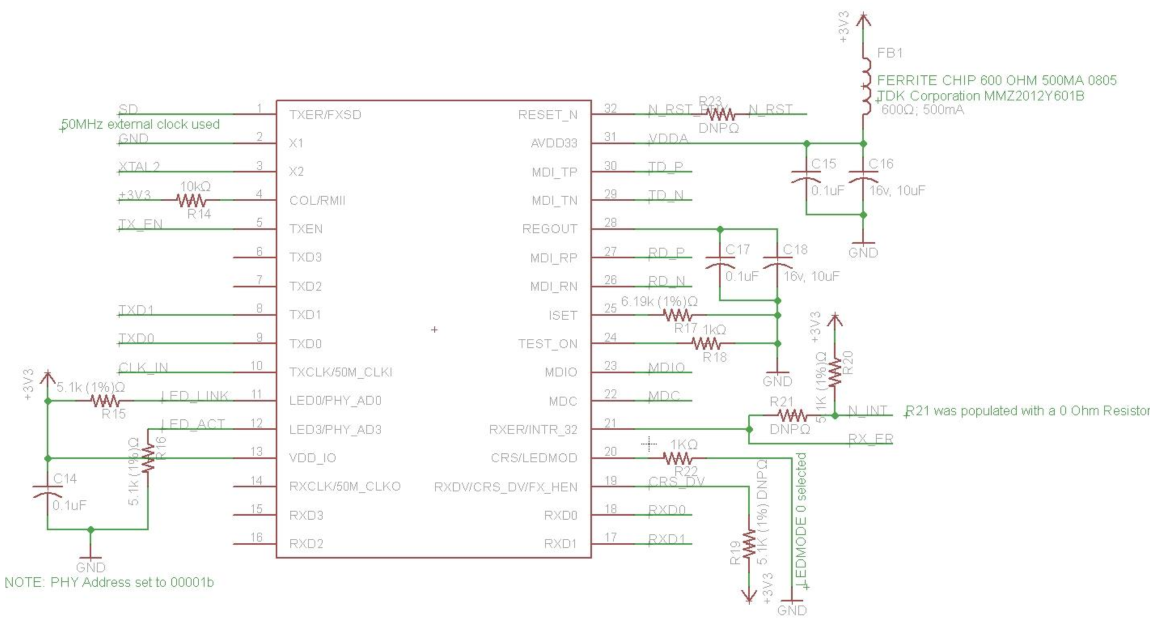

Figure B.2 Ethernet PHY interface schematic

The interface between the PHY and POF Transceiver can be seen in Figure B.3. Special considerations are necessary for this interface, since the interface uses differential signals. The differential signals required proper split-load termination for data integrity. The split-load termination component values are from recommendations in an Avago application note and the Avago AFBR-5803AZ datasheet [47, 48]. Additional decoupling components and values were referenced from the application note and datasheet [47, 48].

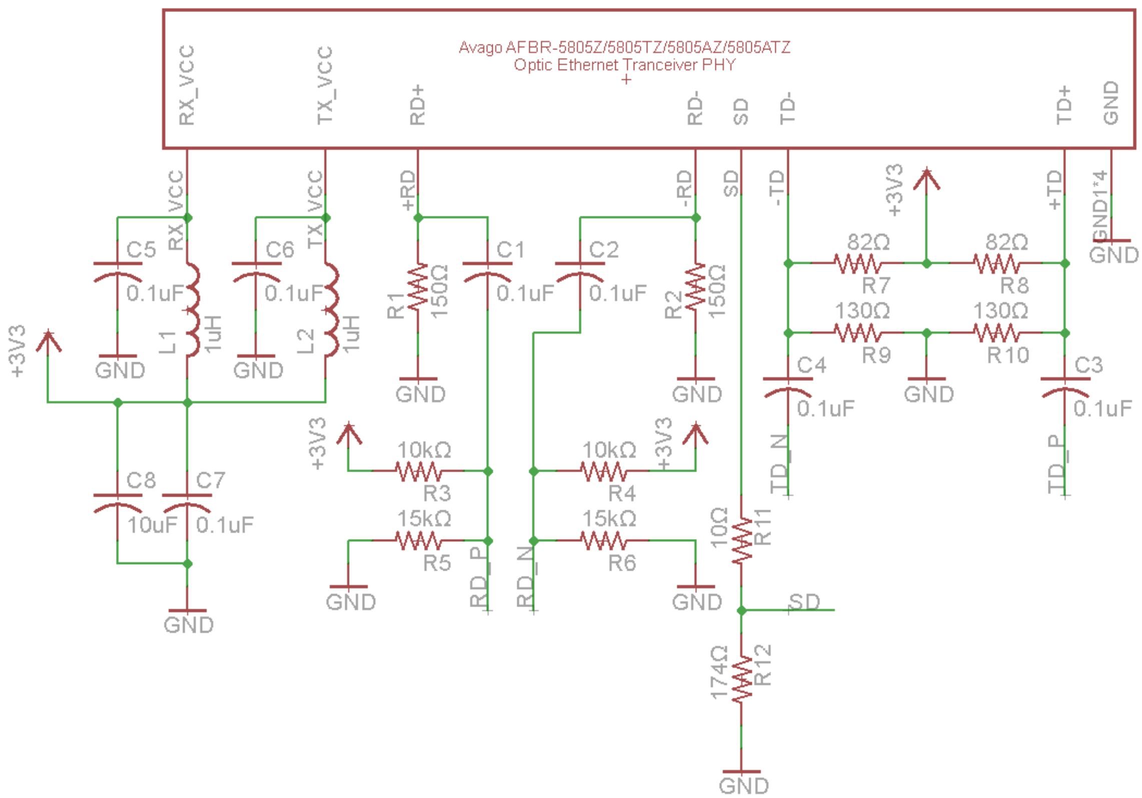

Figure B.3 POF Transceiver interface schematic

B.2 PIC32 ESKII Fiber Optic Daughter Board Printed Circuit Board Layout

A circuit board design was created as shown in Figure B.4, which is a two-layer board with the dimensions of 2.45 cm wide by 6.04 cm long. The dual in-line package to the Microchip ESKII is near the top of the design, shown with the green drill holes. The top layer contains the IC Plus IP101G PHY controller. Special care was taken with the clock trace from the ESKII to the PHY to eliminate the need for through-board vias and sharp curves that would cause the trace to act as an antenna and create interference from the 50 MHz signal in the trace. The PHY also connects to the Avago transceiver, which is located near the bottom with the horizontal line of green drill holes. The differential traces from the transceiver to the PHY are routed together and have trace meandering to keep the trace lengths equal to ensure that the signals in each trace reach the other end at the same time. Other considerations in the design include the top layer that contains the ground plane and the bottom layer that contains the +3.3 V power plane with an exception near the center of the board, which supplies the ground connection to the PHY. These power and ground pours provide a stable voltage source for all of the components on the daughter board. The polygon fills for the power and ground layers are removed from Figure B.4 for better clarity.

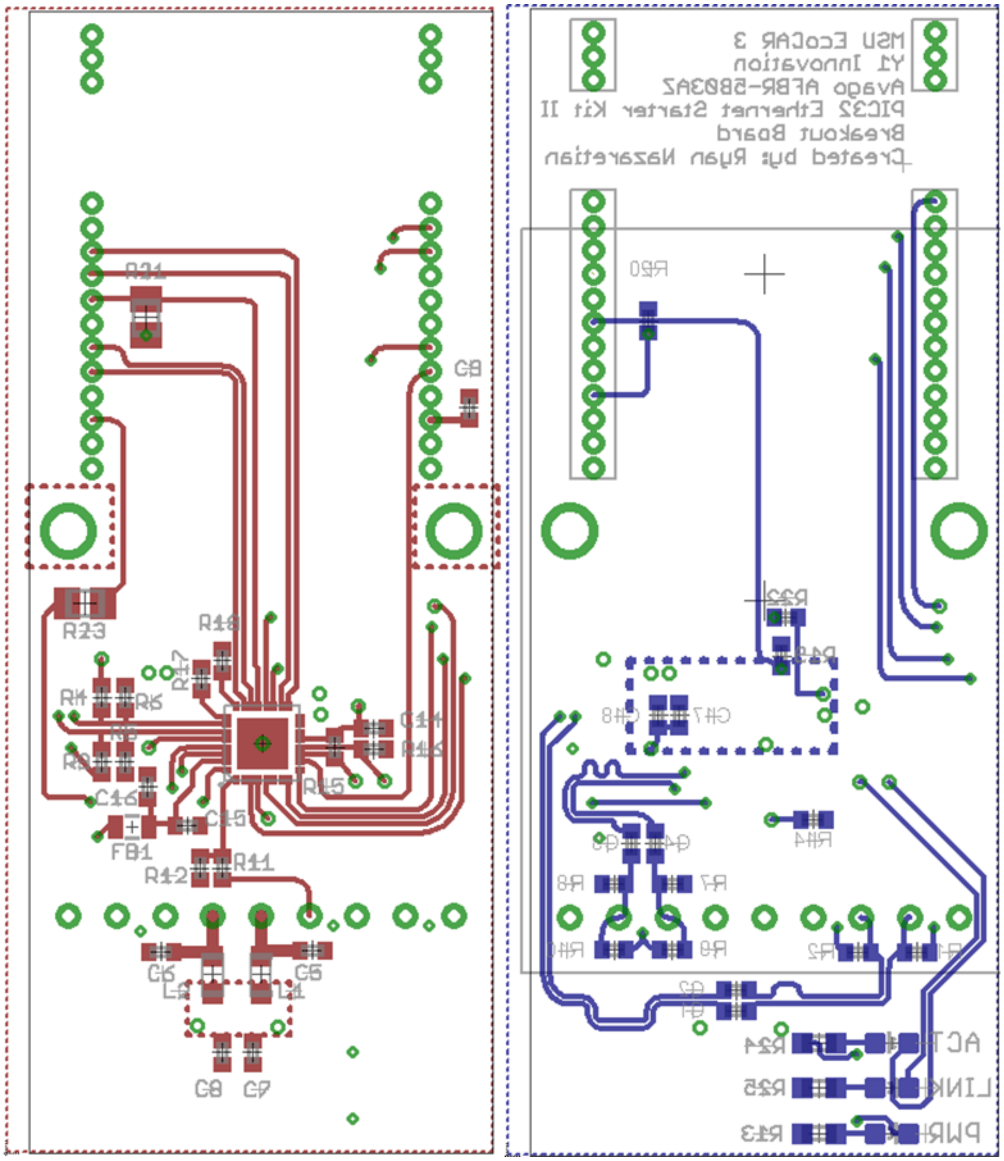

Figure B.4 Top layer (left) and bottom layer (right) of the ESKII POF Daughter Board

The printed circuit board (PCB) layout in Figure B.4 was ordered as a blank board, shown in Figure B.5, which needed to be populated with components as shown in Figure B.6. One difference between the board described in Figure B.4 and the board shown in Figure B.5 exists. Figure B.4 contains a jumper (0 Ω resistor) at R23 (left side below large mounting hole) that disabled the N_RST signal. The original schematic contained this signal; however, Microchip cut this trace from their daughter boards because resetting the PHY caused initialization issues. This issue was found after the boards were ordered, so the trace was cut in the prototype.

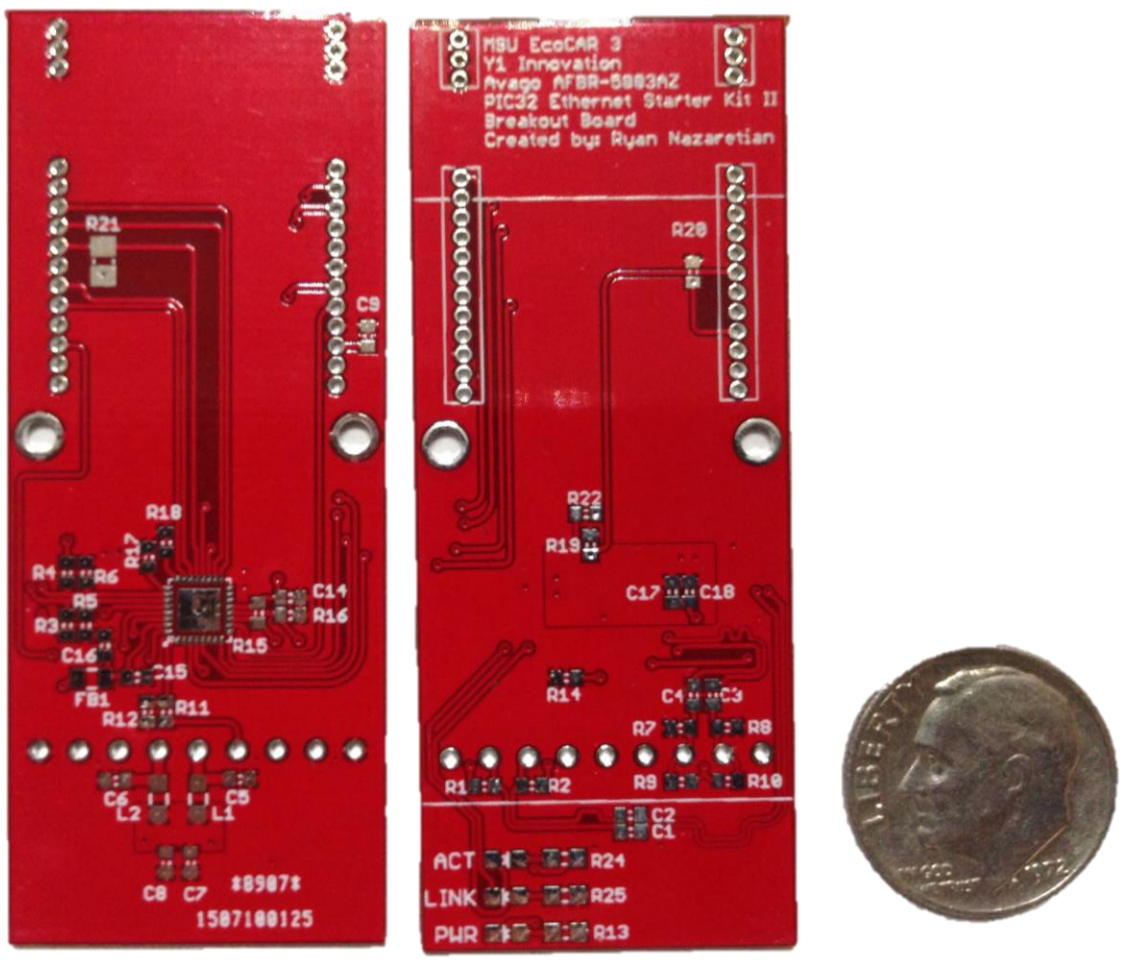

Figure B.5 Blank fiber optic daughter boards PCBs with a dime for size comparison.

Figure B.6 Assembled fiber optic daughter board

B.3 Other hardware schematics

Additional circuitry not captured by the daughter board schematics included the CAN transceiver, the high-side driver, a fuse panel, the PIC32 ESKII schematic, and PIC32 I/O Expansion schematic.

B.3.1 CAN Transceiver

The CAN transceiver enables CAN communication from the microcontroller to a vehicle's CAN bus. The CAN transceiver integration schematic is shown in Figure B.7

with the 120 Ω termination resistor. A 0.1 µF capacitor (C1) was added for decoupling purposes.

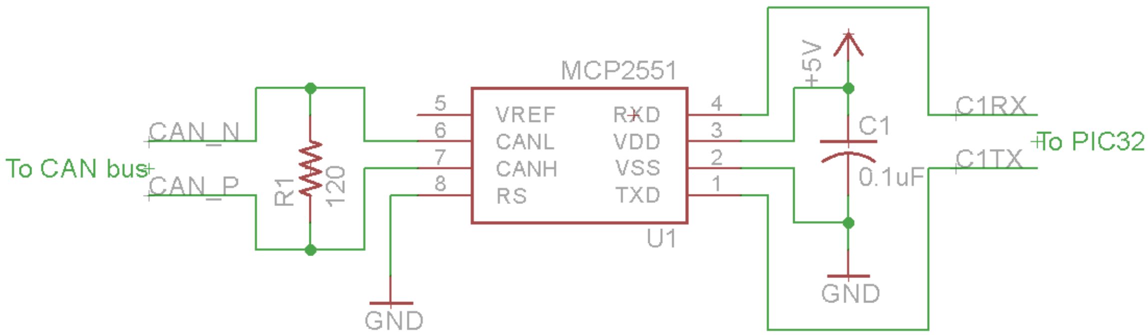

Figure B.7 CAN transceiver integration schematic

B.3.2 High-side driver

The high-side driver enabled the microcontroller's low 25 mA maximum current outputs to drive high current loads such as a headlight [49]. The high-side driver also featured current sensing and diagnostics to detect open or shorted loads. The integration schematic can be seen in Figure B.8.

Consideration had to be made when interfacing the current sensor with the microcontroller. When a driver fault occurred, the driver would output a regulated 8 V, which would exceed the CMOS voltage input level limit of the microcontroller. To alleviate this problem, a voltage divider circuit using R4 and R5 was developed that reduced the 8 V down to 3.18 V. For additional protection, a 10 kΩ resistor was placed between the voltage divider's output and the microcontroller. A Zener diode or transient voltage suppressor (TVS) diode at the microcontroller's current sense input would improve the safety of the circuit if the input voltage went above the microcontroller's 3.6

V voltage limit (+0.3 V above V_{DD}) [49]. Neither of those components were on-hand when assembling the prototype and the risk of damaging the microcontroller was low enough to disregard the concern.

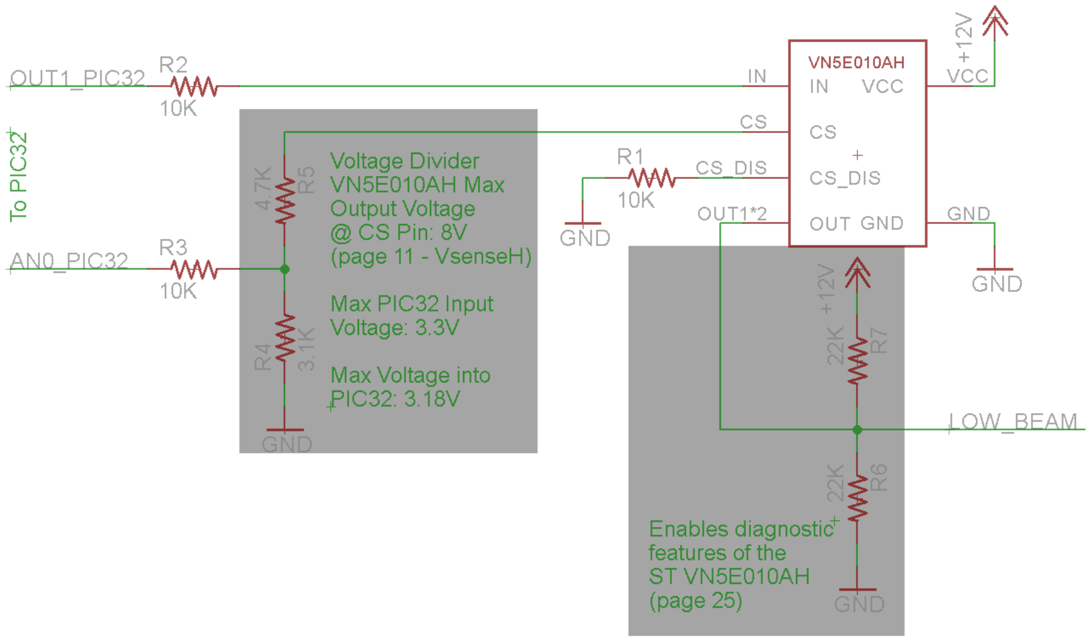

Figure B.8 High-side driver integration

To enable the diagnostic features of the high-side driver, two 22 kΩ resistors (R6 and R7) created a 50/50 voltage divider at the output of the driver. With these two resistors installed and no load connected, the driver's output pin should read around 6 V when powered by a 12 V supply, which allows the driver's diagnostic feature to detect if there is a short circuit (reads exactly 0 V) or an open circuit (read 6 V) before engaging the driver.

APPENDIX C

SOFTWARE OVERVIEW

C.1 PC Side Software – Python

The PC side software is responsible for establishing a link between the PC and the microcontroller over Ethernet, continuously pinging the microcontroller to ensure the microcontroller is connected, measuring the round trip time from sending data to the microcontroller and receiving data back from the microcontroller, sending control data to the microcontroller and receiving feedback from the microcontroller. The PC side software was written in Python using WinPython 64-bit version 3.4.3.4 and the graphical user interface (GUI) shown in Figure C.1 was designed using Qt4 Designer included with the WinPython release. The software contains two execution threads. One thread controls the GUI and sends UDP messages to the PIC32. The second thread waits for received UDP messages. The separate threads keep the GUI responsive in the event that the PIC32 stops responding, allowing for a better debugging experience. The GUI is shown in Figure C.1 and described in section 4.4.

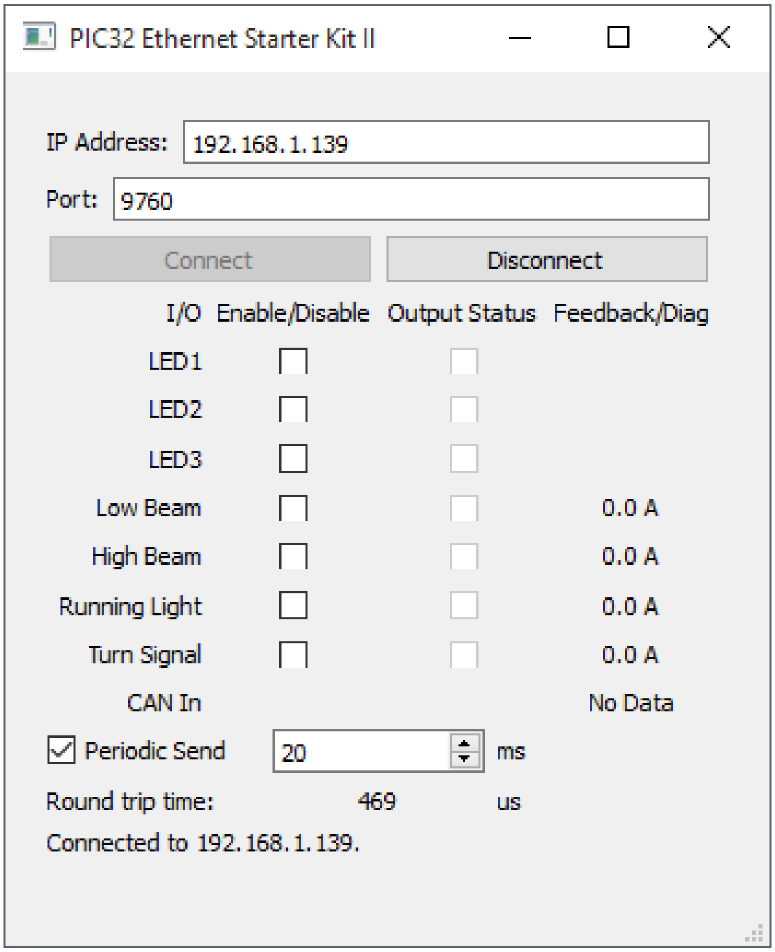

Figure C.1 Python GUI connected to PIC32

C.2 Embedded Side Software – C

The embedded software is programmed in the C programming language for the PIC32 using the Microchip MPLAB X integrated development environment using the Microchip XC32 compiler for the 32-bit Microchip PIC microcontrollers.

A new rapid prototyping solution from Microchip, called MPLAB Harmony, allows a developer to select prebuilt drivers and software libraries to integrate into the project [41]. The MPLAB Harmony Configurator is shown in Figure C.2.

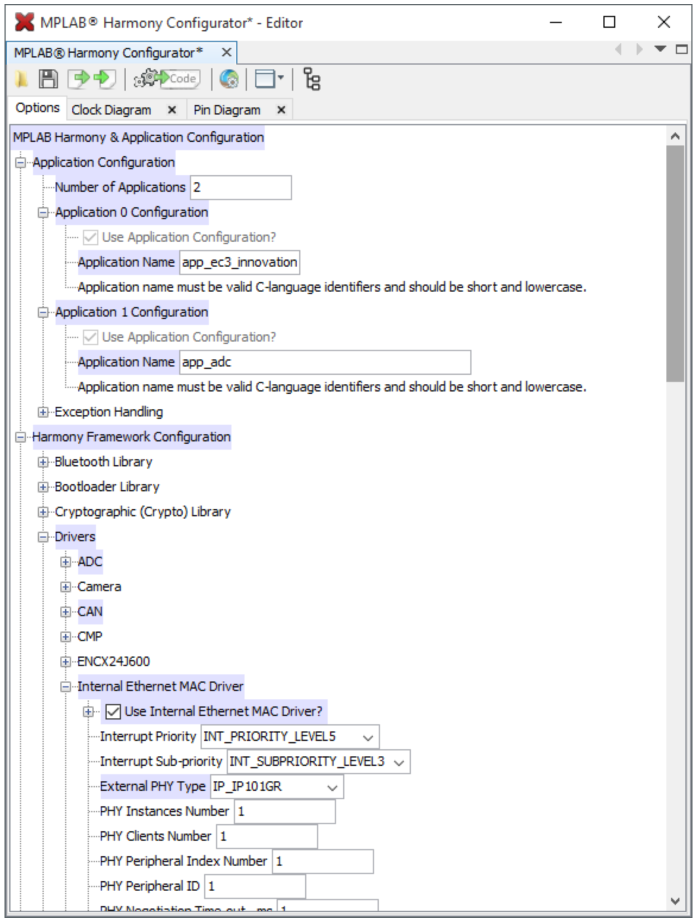

Figure C.2 MPLAB Harmony Configurator

MPLAB Harmony encourages the use of multiple finite-state machines (FSM), which allow for much more complex code to be easily developed. Microchip refers to these FSMs as "apps", since each FSM can be interpreted as separate thread with independent states and actions.

The project for this thesis uses two 'apps'. The first app is used for the UDP/CAN communication that checks for queued CAN data and reads the latest analog to digital converter (ADC). ADC data is collected from the second app, the ADC app, which switches the ADC's multiplexer to one of the four different inputs connected to the current sense outputs of the high-side drivers and periodically reads the ADC's voltage.

The UDP/CAN communication thread starts by waiting for a UDP connection where it will then bind a socket to this section of code. Once the socket is bound, it receives how many bytes of data are queued to be read. If the number of bytes is greater than zero, the application will read this data into a character buffer. The character buffer is then passed to a function created called "handleMessage".

The "handleMessage" function takes the data received, checks the data length, and unpacks the data if the received length is correct. In this case, the function is expecting 8 bytes of data, mainly to replicate the maximum length of a CAN message. However, only 1 byte of data is used, which contains a mask to toggle the various outputs. The first operation performed is to read the high-side driver current values from the ADC application. The ADC's contents contain a voltage proportional to the current passing through the driver or the fault status if the output of the driver is 8V (or 3.18 V into the microcontroller). The function then interprets the data and sets a bit in the fault mask for the particular output if a fault is found. Next, if there is no fault reported by the high-side driver, the function toggles the output. The function then creates an output status message that contains the output states, fault information, and current information. Finally, the function will read any buffered CAN data and append that to the output status message. At this point, the function will return to the UDP/CAN application thread.

Back in the UDP application thread, a UDP and CAN message will be transmitted with the output status data. After the messages are transmitted, the application will close the socket and wait for the next message to be received.