

Automotive CAN Network engineering & automotive instrument cluster control tool

RESEARCH MANUAL

BY

THOMAS JORDAN
INSTITUTE OF TECHNOLOGY CARLOW

APRIL 2021

Abstract

The CAN network is a series of components that are within a vehicle that allows individual components within a vehicle to communicate. My aim with this Project is to take control of the CAN network and adapt the messages within this network to have that vehicle complete tasks it would not normally do. To achieve this goal of reverse engineering and controlling messages on a CAN network I am going to create a desktop application that will allow the user to have the ability to control the CAN bus by sending their own CAN messages onto the CAN bus and throughout the CAN network.

Contents

Abstract	0
Introduction	1
What is needed to complete this project?	2
What is a CAN BUS?	2
<i>Different CAN bus protocols</i>	4
<i>Different CAN bus devices</i>	5

Software languages	6
IDE's	9
API's.....	11
The application.....	11
Summary.....	12
Conclusion	13
References	14

Introduction

The reason for this project is to bring to light the accessibility and functionality we have today when it comes to cars and specifically their electrical systems. I chose this project because I have always loved cars and the automotive industry and technology so why not try and combine them both. The aim for this manual is to research and report on the reverse engineering of the CAN bus network within an automotive vehicle and how those devices and systems operate together and try to create an application that allows the user control of this network. A CAN network is a network built up within a vehicle which gives it the ability to communicate with all the individual systems within a vehicle like the engine management system, driver aids and other systems like the abs, traction control or the stability control. All these different systems in a modern vehicle must communicate for it to operate correctly and that is where the CAN network comes into play. In this report I will cover many different topics which will aid in the development of my knowledge of the automotive vehicle and the development of this application. The topics that will be covered within this report are the CAN bus itself. I will explain what it

is? How it functions? And I am going to give examples of different CAN networks and how they differ from each other. I will then move on to the specific programming language that I will use and explain why I am using that language, what is the benefits and drawbacks of that specific language and I am going to compare that language to some of the popular programming languages that are out there today and why I chose not to go with those languages and explain why you might want to go with one of those languages.

I will then move on to the software that will aid me in creating this application and this will include different IDE's that will help write and compile my application and API's that will allow me to communicate with the devices I will need to interact with to create a working application.

What is needed to complete this project?

WHAT IS A CAN BUS?

The CAN bus is a protocol within a vehicle which takes in all messages being broadcast from all micro controllers and ECU's. These messages are sent on the CAN bus and are picked up by the device within the car that needs it for example for this application the main aim is to take control of the instrument cluster and this device within a vehicle may take in multiple readings from sensors around the vehicle like the wheel rpm (rotations per minute) and that would be sent along the CAN bus and then collected by the ECU corresponding to the speedometer. These instruments within a vehicle may not only take

in one message from the CAN bus it may take in multiple readings from the bus to make the greater calculation to be displayed on the Instrument.

How it functions

The CAN bus splits its traffic into two different signals which are named CAN high (CANH) and CAN low (CANL). When a signal comes across these lines CAN uses differential signaling which means that when a message comes across the high line the voltage on that line is raised from 2.5v to 3.5v and at the same time the CANL line is dropped in voltage by the same amount which would end up being 1.5v from 2.5v. So, with that all messages are given an ID number by their respective micro controller and since a controller can send out multiple messages at once it gives each message an unique ID and this allows the CAN bus to know what message is given priority and the lower the ID number the higher the priority and is then sent on the CAN high signal explained earlier and the higher ID numbers travel along the CAN low signal. An example is the abs ID is given higher priority than the message containing info on the windows.

The messages that are sent throughout the network are sent over the data link layer of the OSI model and so they are called frames and within these frames the ID number is stored along with other more important information like the size of the data that is being sent within the frame which is called the DLC (Data length code) and the data itself which can be up to eight bytes which can mean that if the data section is not always full it is padded out to fill out the rest of the empty bytes.

So, from the research I have done the Instrument cluster for example may take in information from the wheel speed and multiple other factors.

To access this CAN bus and the messages we need a PCAN USB and this would allow us to connect into the OBD port which allows the user to see the messages sent back and forth. (Smith, 2020)

Different CAN bus protocols

Within the realm of the CAN network protocol there are multiple different versions of a CAN protocol and while the main CAN bus protocol has somewhat internationally standardized but there are still some unique circumstances where bespoke or inhouse CAN bus protocols are being used today. In this section of the research report, I will be giving examples of these different CAN bus protocols and how they work and how they differ from the standard CAN bus protocol.

The first alternative CAN bus protocol is the ISO-TP protocol and this method is mainly used in diagnostics and it is mainly a protocol that deals with large size frames within the bus and this protocol deals with this problem by extending the eight byte limit of the frame within the bus and it achieves this by taking the individual frames containing the same data and chaining those together the process is done by getting the first byte of the frame and using it for addressing and allow the other seven bytes to be used for the data within the frame. The maximum supported size of frame within this protocol is 4095 bytes.

The next protocol that is relevant is the CANOPEN protocol this a higher layer protocol that is built based on the CAN bus but is mostly used not within the automotive industry and is more used in the industrial side of businesses. CAN open is mostly used on motorized machinery like robotics more specifically stepper or server motors and this

protocol uses a CAN frame which is different from the normal CAN frame it uses an eleven bit ID in the frame which is the same as the CAN ID but is broken up into two different sections and these sections are the function code and the node ID and in a CANOPEN frame the abbreviation ID in normal CAN is known as the communication objective ID (COB-ID). The COB-ID allows CANOPEN protocol to know which node is sending or receiving data and what is the function of that data.

Different CAN bus devices

For this application to operate the user would need a device that allows them to communicate between the CAN network and the application so for this a device was created and it allows that user to interpret the messages that are sent over the CAN network and may allow them to send or receive messages on the network and these devices come with many features that allow you to take the best advantage of this network.

For this project I will be given one of these devices to aid in the creation of my application and the certain device that is a “pcan USB device” which is created by the company PEAK who specialize in industry tools for specific jobs like this application. Their device is on the higher end of the devices but are still on the consumer side of this industry but the benefit of being on the high end of the industry is that trust in the device having the ability to complete that task that you give it, and it comes with key software that will aid in the development of my application. (GmbH, 2020)

Since the “pcan USB device” is on the higher end of the I thought I would research more devices on the market that take on the same task and compete in the same industry. The

first device I discovered was the CANABLE device that was on the lower end of the price bracket and is an open-source product this means that it uses preexisting software that allows it to access the CAN network. It uses a software called “SOCKETCAN” and this software allows the user to monitor the can network and specifically the messages sent across the CAN bus. This software allows that user to filter specific messages from certain parts of the vehicle and allows you to learn the messages sent over the CAN BUS by monitoring and capturing data live while you operate the car.

The “CANABLE” device allows you to work from multiple operating systems like windows, Linux. While in operation within each operating language in interacts with the CAN bus in different ways for example on windows it operates as a virtual serial port and on Linux it operates as a serial line on the CAN bus and is enumerated as a native interface on Linux. (An Open-Source USB to CAN Adapter - CANable, 2020)

SOFTWARE LANGUAGES

Reason

For this project I am going to have to create an application tool with the use of a programming language and to be able to make the best and chose and judgement about which programming language I am going to use I am going to research and report on three of the most common programming languages used today and why I should pick one over the others.

Findings

I first started off by researching C# which is a programming language which is derived from the c programming language and is a main part of the .net framework and this is one of the great reasons why I have considered C# because of the .net framework because it allows you to do so many things and be adaptable to nearly any situation.

Some of the benefits of using C# and the .net framework would be its great interoperability this allows C# to access and compile any written code that was created with any other .net language like F# or visual basic. This is a great feature of this language because of the adaptability for that application down the road. (moore, 2015)

C# is also an object-oriented programming language which means that the language allows for easier creation of new features and the ease at which future improvements can be made.

C# benefits from being type safe also which means that the code is not able to access any space in mem0ry that it does not have execution permission which can help with security which comes as a benefit to a student like me who's background is in security. (C# Features - javatpoint, 2020)

The next language that I researched was python and this is a high level which is highly adaptable and is extremely popular because of its ease of use.

A reason why python is so popular is the use of dynamic high-level data typing which means that there is less need for support code that would be normally needed to correctly create a program and this feature is extremely useful for experienced users who want to

save time writing excessive code and helps simplify the coding language for new people who have just started coding.

Another benefit of python is that it follows the same object-oriented programming structure that C# also benefits from. This allows for great future proofing and allows the code to be easily maintained by anyone that is not the originator.

Python is mainly used in data science or artificial intelligence/ machine learning but that does not limit python to those fields it can also be used in web and application develop and developing operating systems.

The next language is java and this language is a quite similar to the other languages in that it follows some of the same rules of object-oriented programming. Java has also become a robust and secure language with the use of the JRE and the programs have a less chance of causing errors because of the errors being reported at compilation and runtime checking. (Java Features and Advantages - CSVeda, 2020)

Conclusion

In conclusion I have chosen C# as the main language that I will be working with because of its robust infrastructure behind it and allows me to create my application without having to worry about the language and whether or not it is supported for the tasks that I have to complete.

Another reason why I chose C# was because of great adaptability within the language and all the things and applications it can create like a web or desktop application, it also supports integration and similarities with many other languages which takes the stress out of having to learn a completely new language but the does not mean that it's a language that I have used before and this will bring another great learning aspect to this already intriguing project.

IDE'S

Reason

To fully utilize the C# programming language, I am going to need a software that would allow me to edit and run my application and integrate with any other software that I may need for this application. I have chosen to go with a IDE and not a standalone language editor because a IDE (Integrated development environment) allows me to take control over every aspect of the project while also giving familiarity to the way in which I will be creating this application. I have work with IDE's in the past and they have been a great help to me and to the way I have gone about tackling tasks throughout my time programming. To make the best choice about which IDE I will chose I have chosen three of the most well respected IDE's in the industry and have compare them to see which one

suites my style of programming and which integrates best with the software's that I will be using alongside the IDE.

Findings

The first IDE the I have chosen to research is the IDE created by Microsoft visual studio and this IDE is highly adaptable and versatile with the choice of programming languages that it supports it also supports a wide variety of plugins that aid in its ability to adapt to what every task needs to be completed.

Another benefit of using Visual studio is that is a part of the .net standard and allows you to seamlessly create multiple applications using the same standards and libraries for most of them. (Visual Studio 2019 | Download for free, 2020)

Conclusion

In conclusion I have chosen visual studio because it's a great open-source developer environment that will allow me to create the application that I want to create without any restrictions and allows me to integrate any other software that I may need to use along the way.

It also gives me the opportunity in the future to transition this application maybe to mobile if the technology permits the CAN network to be access wirelessly and this can be done through visual studio and Xamarin which is a developer environment that runs alongside visual studio to simultaneously create a mobile application for both apple and android.

API'S

Reason

This Project needs to use an API to allow the application created to communication with the pcan USB that was mentioned early and allows that application to gain access to the correct transmission protocols needed to communication with the CAN bus.

Findings

When researching for the correct API to use with the help of my supervisor he pointed me in the direction of the PEAK “pcan basic API” and this API is built specifically for the PEAK family of devices and integrates seamlessly with the pcan USB.

When researching this API, the functions that were listed compliments the tasks that I need to complete. Some of the functions included reading and writing functions which will be needed for the CAN frame data to be reverse engineered and changed to represent different values. (GmbH, 2020)

THE APPLICATION

To be able to access these messages I will need to create an application that allows me to take control of these messages. To implement this application, I have been researching which languages to create this tool. The main language that I've being researching is C#. The reason why I have chosen this language was that it is a new language to me that seems to be very versatile and lightweight and very approachable and allows for multiple approaches to a single problem. To be able to communication with the CAN bus through my application I will need to implement an API which was created by PEAK who are a

company who create software and processes which are built for implementation within industry. (GmbH, 2020)

This API when implemented would allow the application to perform multiple functions within the CAN bus when called. Some of the functions that can be called using the API are the ability to set the bit rate of the CAN bus, filter individual messages sent within the CAN bus and can check the status information given by the CAN bus I also have the ability to most importantly read and write can messages to the can bus so I can take control of the Instrument cluster. The application will have some basic controls at the start like taking control of the rev limiter, speedometer, and indicator signals.

(GmbH, 2020)

SUMMARY

In summary of what I have covered within this research report I have stated my intention to create an application that will allow a user to take control of the instrument cluster within an automotive vehicle. To best create this application, I had to research the key concepts behind the CAN bus protocol and the network surrounding that and how it operates. To be able to bring this application to the public I had to decide on how to best approach the application so that meant researching what was the best software to use and why not use a different software version.

CONCLUSION

My conclusion on this research manual is that I have learned a great deal about this project and how it is going to play out and what resources I am going to need to complete this project.

On what I have covered within this manual I have covered what is the main aim of this project and why I am undertaking it. I have talked about what a CAN network is and it works and how a CAN bus functions and how it transmits its messages around the network and I also spoke about different types of CAN protocols and how they differ from the CAN bus protocol itself, I talked about different CAN devices that allow you to communicate with the network and how they differ between price points.

I then went into the different software's that I will need to help me in creating this application, I researched different possible programming languages and the benefits and drawbacks. I then talked about the IDE and API that I will be using and the features and reasons why I chose those pieces of software for this project.

This project is a show case of some of the aspects of what I have learned over the last four years and some new techniques that I will pick up along the way and I see this application showing those learnings to the fullest and I hope this is a successful project.

References

1. GmbH, P., 2020. *PCAN-USB: PEAK-System*. [online] Peak-system.com.
Available at: <<https://www.peak-system.com/PCAN-USB.199.0.html?&L=1#>>
[Accessed 25 October 2020].
2. GmbH, P., 2020. *PCAN-Basic: PEAK-System*. [online] Peak-system.com.
Available at: <<https://www.peak-system.com/PCAN-Basic.239.0.html?&L=1>>
[Accessed 25 October 2020].
3. Smith, C., 2020. *The carhacker's handbook*. 2nd ed. [ebook] no Starch press,
p.chapter 5. Available at:
<<https://docs.alexomar.com/biblioteca/thecarhackershandbook.pdf>> [Accessed 21
October 2020].
4. chan, R., 2019. *The 10 Most Popular Programming Languages, According To
The Microsoft-Owned Github*. [online] Business Insider. Available at:
<[https://www.businessinsider.com/most-popular-programming-languages-github-
2019-11?r=US&IR=T#8-shell-3](https://www.businessinsider.com/most-popular-programming-languages-github-2019-11?r=US&IR=T#8-shell-3)> [Accessed 10 November 2020].
5. www.javatpoint.com. 2020. *C# Features - Javatpoint*. [online] Available at:
<<https://www.javatpoint.com/csharp-features>> [Accessed 11 November 2020].

6. moore, w., 2015. *Benefits Of Python Over Other Programming Languages - Invensis Technologies*. [online] Invensis Technologies. Available at: <<https://www.invensis.net/blog/benefits-of-python-over-other-programming-languages/>> [Accessed 10 November 2020].
7. CSVeda. 2020. *Java Features And Advantages - Csveda*. [online] Available at: <<https://csveda.com/java/>> [Accessed 10 November 2020].
8. Visual Studio. 2020. *Visual Studio 2019 | Download For Free*. [online] Available at: <<https://visualstudio.microsoft.com/vs/>> [Accessed 11 November 2020].
9. JetBrains. 2020. *Features - Rider*. [online] Available at: <<https://www.jetbrains.com/rider/features/#web-development>> [Accessed 11 November 2020].
10. 8devices.com. 2020. *Korlan USB2CAN - 8Devices*. [online] Available at: <https://www.8devices.com/products/usb2can_korlan> [Accessed 10 November 2020].

11. GmbH, P., 2020. *PCAN-USB: PEAK-System*. [online] Peak-system.com.

Available at: <<https://www.peak-system.com/PCAN-USB.199.0.html?&L=1#>>

[Accessed 15 October 2020].

12. Canable.io. 2020. *An Open-Source USB To CAN Adapter - Canable*. [online]

Available at: <<https://canable.io/>> [Accessed 6 November 2020].