

Instituto Tecnológico y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018,
publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática
Especialidad en Sistemas Embebidos



CAN-Ethernet Gateway for Visualizing CAN Network Packets in Wireshark

TRABAJO RECEPCIONAL que para obtener el **GRADO** de
ESPECIALISTA EN SISTEMAS EMBEBIDOS

Presenta: **JUAN CARLOS RUIZ OCHOA**

Tutor **SERGIO NICOLAS SANTANA SANCHEZ**

Tlaquepaque, Jalisco. Julio de 2023

CAN-Ethernet Gateway for Visualizing CAN Network Packets in Wireshark

Juan Carlos Ruiz Ochoa
Postgraduate in Embedded
Systems

ITESO, Instituto Tecnológico de
Estudios Superiores de Occidente
Tlaquepaque, Jalisco, Mexico
jruiz.ochoa@iteso.mx

Sergio Nicolas Santana Sanchez
Postgraduate in Embedded
systems

ITESO, Instituto Tecnológico de
Estudios Superiores de Occidente
Tlaquepaque, Jalisco, Mexico
sergioni@iteso.mx

Abstract— In automotive control systems, the analysis of CAN messages is key to identify and correct errors. This paper presents a CAN-Ethernet Gateway, which allows knowledge acquisition from the analysis CAN systems. The FRDM-K64F board is utilized as it supports both CAN and UDP communication protocols. In addition, a CAN transceiver is used to establish the connection with the Vector CAN interface. Therefore, all gateway functionality is visualized in the computer. For testing, CANoe with an ITESO license is used to enable the simulation of the CAN system. The Gateway successfully captures CAN messages of different sizes and facilitates their analysis in Wireshark by displaying each data in a structured format, which enables enhanced and efficient monitoring of automotive systems.

Keywords—CAN-Ethernet Gateway, CANoe, Wireshark

I. INTRODUCTION

In automotive electronics, interconnected systems use the serial communication protocol CAN (Controller Area Network) to transmit packet messages with bitrates up to 1 Mbit/s. This protocol offers numerous advantages, including high-speed networking capabilities and cost-effective multiplex wiring solutions. CAN transfers bits between interconnected systems, and each connection is represented by a node. These bits represent message data, facilitating communication through the CAN bus [1]. In fact, analyzing these messages becomes crucial, particularly in detecting potential system issues.

To acquire knowledge in CAN analysis and CAN message data frames, CANoe is a development and software tool that is used to create and simulate a wide range of messages according to specific requirements. In this context, CANoe allows a better understanding of the structure of CAN messages by generating custom messages using CAPL (Communication Access Programming Language).

In CAN bus networks, analyzing the data packets requires the use of gateways, which play a vital role in connecting two networks with different transmission protocols. However, these devices often require the purchase of software licenses for analysis, leading to increased costs. Open-source software is an affordable option used for continuous code development and improvement, eliminating the need to wait for software updates. Furthermore, users have complete control over the source code, providing absolute freedom to use the program for any purpose and adapt it to specific requirements [2]–[4].

Wireshark is an affordable option which excels in its ability to analyze packet captures and capture speed compared to other alternatives [5]. Therefore, the aim of this project is to implement a CAN-Ethernet Gateway device utilizing Wireshark as the preferred alternative for the analysis of CAN messages.

II. SYSTEM DESIGN

The main gateway device in this setup is the NXP board FRDM-K64F, which is equipped with a CAN transceiver. It captures CAN messages from systems that utilize the CAN protocol, such as the automotive system. Therefore, the Vector CAN interface is used to simulate a CAN message from any system. These CAN messages are then transmitted to the Wireshark software for analysis over Ethernet using the UDP (User Datagram Protocol). The gateway system design is illustrated in “Fig. 1”.

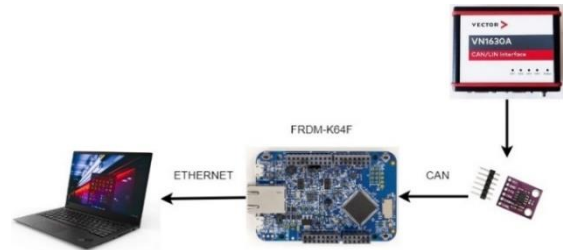


Fig. 1. Gateway system design.

A. FRDM-K64F board

The NXP Freedom K64 (FRDM-K64F) hardware is a freedom development board created for rapid prototyping of microcontroller-based applications. The FRDM-K64F board is built on the ARM Cortex M4 core evaluating Kinetis K series devices. This board also has a maximum operation frequency of 120 MHz, 1MB of flash, 256 KB RAM, a full speed USB controller, Ethernet controller, secure digital host controller, analog and digital peripherals due to the inclusion of the MK64FN1M0VLL12 MCU [6].

The FRDM-K64F incorporates the FlexCAN module, which implements version 2.0 B of the CAN protocol. This module supports both standard and extended message frames. The board also features an embedded RAM specifically dedicated to the FlexCAN module, serving as storage for the message buffers [7].

The Ethernet MAC on the FRDM-K64F enables high-speed communication within the range of 10 Mbps to 100 Mbps, in accordance with the IEEE802.3-2002 standard. It also includes a hardware acceleration block that optimizes the performance of network controllers providing TCP/IP (Transmission Control Protocol / Internet Protocol) and UDP (User Datagram Protocol) protocols [7].

B. Transceiver CAN

To enable the reception of CAN messages, the FRDM-K64F board requires an interface between a CAN protocol controller and the physical bus. This interface is provided by the MCP2551 transceiver, which is specifically designed for this purpose. This device enables differential transmission and reception capabilities for the CAN protocol controller. The MCP2551 operates at speeds of up to 1 Mb/s and is compatible with the ISO-11898 standard, including 24V requirements [8].

C. Vector CAN Interface

The interfaces serve as a bridge between Vector tools such as CANoe and CANalyzer. These devices offer a wide range of functionalities, including bus analysis, bus behavior simulation, issue diagnosis, calibration, and flash programming [9].

The vector CAN interface VN1630A used in this work was provided by the Instituto Tecnológico de Estudios Superiores de Occidente (ITESO) laboratory. For this project the interface is used exclusively for testing and simulating a CAN system and not for analyzing CAN messages.

III. FIRMWARE DESIGN

The main program used to create the Firmware is the MCUXpresso IDE by NXP, which is embedded on the FRDM-K64F board. Additionally, Wireshark was modified with LUA scripting to display the CAN message data in a clear format. CANoe is also utilized, where the simulated CAN messages are created using CAPL and the Vector CAN interface. The structure of firmware is represented in “Fig. 2”.

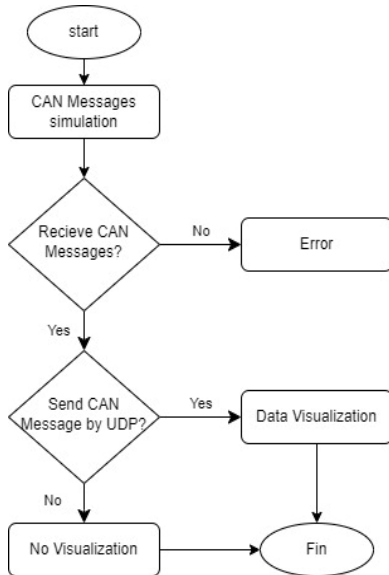


Fig. 2. Firmware structure.

A. MCUXpresso IDE

The MCUXpresso is the NXP IDE that provides advanced editing, compiling, and debugging features. It also offers Software Development Kit (SDK) examples that can be used in projects to facilitate development.

In this paper, the FlexCAN and UDPecho FreeRTOS examples were utilized to develop the firmware. To enable the transmission of CAN messages over UDP in real-time, both functionalities can be combined. The original FlexCAN example simulated CAN messages in a loop, and these were printed on the terminal. However, modifications of this original example were performed to the firmware. Firstly, the simulated CAN messages were sent over UDP in real-time using FreeRTOS. Subsequently, the firmware was enhanced to capture actual CAN messages and send them over UDP. The firmware incorporates conditional compilation, which facilitates user interaction by offering a choice between simulated option or captured external messages.

B. Wireshark

Wireshark is a network sniffing tool with a user-friendly graphical interface and high capabilities for capturing and filtering network traffic. Also, Wireshark is freely available to users and is versatile as it can scan types of networks, including Ethernet, WI-FI, and even Bluetooth [5].

To improve visualization of captured information, LUA embedded in Wireshark with C language programming is used to modify the display according to specific needs. This modification separates the data frame and displays it in different parts of the message such as packet payload, CAN ID, packet format, packet length, etc. This allows the user to easily visualize the packet message and interpret the data effectively.

C. CANoe

CAPL enhances the functionality of the Vector CANoe tool, serving as an event-controller programming language specifically designed for simulating, testing and diagnostics. It enables the creation of network node models within CANoe, thereby extending its capabilities and enabling advanced functionalities for comprehensive testing and simulation scenarios.

Different CAN messages were created in CAPL with varying time periods. These messages are sent only once when the key “a” is pressed, utilizing the Vector CAN interface. This setup effectively simulates a CAN system, allowing for the emulation of realistic communication scenarios.

IV. TESTING RESULTS

For testing, the computer is connected to the system on the FRDM-K64F board using both USB and Ethernet, as well as the

Vector CAN interface to visualize the system's functionality. "Fig. 3" illustrates the test environment.

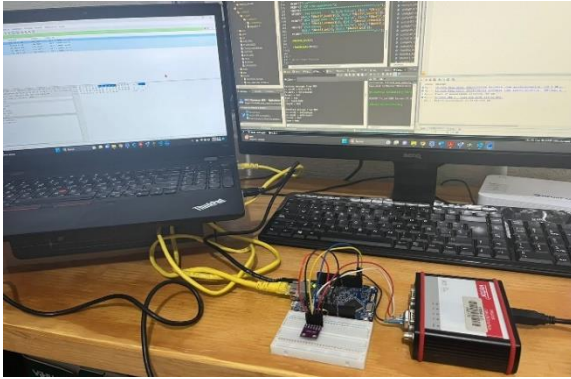


Fig. 3. Test environment.

When the key "a" is pressed, the CANoe software displays CAN messages and provides an indication of any errors. In such cases, it is necessary to examine the connections or check if the FDRM-K64F board is running properly. CAN messages are shown in "Fig. 4".

Time	Chn	ID	Name	Event Type	Dir	DLC	Data ...	Data
0.201455	CAN 1	123	Message_1	CAN Fra...	Tx	8	8	11 22 33 44 55 66 77 88
0.301607	CAN 1	191	Message_2	CAN Fra...	Tx	8	8	9C AE 71 7C 35 E6 A7 8D
0.401663	CAN 1	46B	Message_3	CAN Fra...	Tx	8	8	8C ED 9C A6 95 1E 64 A7
0.501767	CAN 1	130	Message_4	CAN Fra...	Tx	8	8	51 48 24 11 54 9A 5E 5D
0.601631	CAN 1	5A3	Message_5	CAN Fra...	Tx	8	8	E2 35 F4 73 4B 64 2B 23

Fig. 4. Simulation of CAN messages.

To maintain better control during the test, the MCUXpresso terminal prints the CAN message and indicates the data that will be transmitted before sending it over UDP. "Fig 5" is an example of a CAN message on the terminal.

```

Installed SDKs Problems Console Terminal x Image Info Debugger Console Offline Peripherals
COM4 x
*****
UDP Echo example
*****
IPv4 Address : 192.168.0.102
IPv4 Subnet mask : 255.255.255.0
IPv4 Gateway : 192.168.0.100
*****
==FlexCAN Transceiver functional example -- Start==

Received message from MB9
rx word0 = 0x11223344
rx word1 = 0x55667788
format = 0x0
type = 0x0
id = 0x1230000
length = 0x8

```

Fig. 5. MCUXpresso terminal.

The last step involves using Wireshark, where the data of the CAN message is displayed for its analysis, and it can be visualized in "Fig. 6". For easier analysis, a filter can be applied by adding "CAN_SNIFF" to only display messages of that specific protocol. Once filtered, the desired message can be selected and deployed for further examination.

Time	Source	Destination	Protocol	Length	Info
3975.198.011753	192.168.0.1...	192.168.0.100	CAN_SNI...	607	→ 50000 Len=16
3976.198.033813	192.168.0.1...	192.168.0.100	CAN_SNI...	607	→ 50000 Len=16
3977.198.050551	192.168.0.1...	192.168.0.100	CAN_SNI...	607	→ 50000 Len=16
3978.198.080901	192.168.0.1...	192.168.0.100	CAN_SNI...	607	→ 50000 Len=16


```

Frame 3881: 60 bytes on wire (480 bits) captured (60 bytes) over interface 0
Ethernet II, Src: MS-NLB-PhysServer-10:00:00:00:00:00, Dst: 08:00:2c:00:91:00, Len: 1139, Protocol: Internet Protocol Version 4, Src: 192.168.0.100, Dst: 192.168.0.100
User Datagram Protocol, Src Port: 7, Dst Port: 50000
<Wireshark Lua fake item>
CAN sniffer protocol.
  Packet timestamp: 0xb419
  Packet length: 0x08
  Packet type: DATA (0x00)
  Packet format: STD (0x00)
  Packet reserved: 0x00
  CAN ID: 0x00001203
  Packet payload: 0x44332211
  Packet payload: 0x88776655

```

Fig. 6. CAN message data.

V. CONCLUSION

The gateway device enhances the overall understanding of the data and supports efficient troubleshooting and problem solving in automotive systems. Also, the Gateway device offers a cost-effective and easily manipulable solution. It not only facilitates the acquisition of knowledge about the CAN protocol, but it also enables the visualization of data in an organized manner simplifies the analysis process, which plays a crucial role in the automotive industry as it allows to monitor the functionality of systems and address any potential issues.

REFERENCES

- [1] Robert Bosch, "CAN Specification," Sep. 1991. Available: <http://esd.cs.ucr.edu/webres/can20.pdf> Accessed 7 June 2023
- [2] M. Tomazin and M. Gradisar, "Free/Open Source Software in Slovenian primary and secondary schools — Survey 2007," in ITI 2008 - 30th International Conference on Information Technology Interfaces, Jun. 2008, pp. 177–182. doi: 10.1109/ITI.2008.4588403.
- [3] M. Tomazin and M. Gradisar, "Cost-benefit analysis of free/open source software in education," in Proceedings of the ITI 2009 31st International Conference on Information Technology Interfaces, Jun. 2009, pp. 473–478. doi: 10.1109/ITI.2009.5196129.
- [4] A. J. Turner, "The development and use of open-source spacecraft simulation and control software for education and research," in 2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06), Jul. 2006, p. 7 pp.-. doi: 10.1109/SMC-IT.2006.82.
- [5] P. Goyal and A. Goyal, "Comparative study of two most popular packet sniffing tools-Tcpdump and Wireshark," in 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN), Sep. 2017, pp. 77–81. doi: 10.1109/CICN.2017.8319360.
- [6] NXP Semiconductors, "FRDM-K64F Freedom Module User's Guide," .2016,8. Available: <https://www.nxp.com/webapp/sps/download/> Accessed 21 July 2023.
- [7] NXP Semiconductors "K64 Sub-Family Reference Manual," .2014,1. Available: <https://www.mouser.com/datasheet/2/813/K64P144M120SF5RM-1074828.pdf> Accessed 23 July 2023
- [8] Microchip Technology Inc., "MCP2551 High-Speed CAN Transceiver" MCP2551, 2001. Available <https://ww1.microchip.com/downloads/en/devicedoc/20001667g.pdf> Accessed 21 July 2023
- [9] Vector, "VN1600 Fact Sheet," VN1600, Feb 2023. Available: https://cdn.vector.com/cms/content/products/VN16xx/docs/VN16xx_FactSheet_EN.pdf Accessed 21 July 2023