Automotive Diagnostic Gateway using Diagnostic over Internet Protocol

Young Seo Lee, Jin Ho Kim, and Jae Wook Jeon

The School of Information and communication Engineering, Sungkyunkwan University / Suwon, South Korea
lys673@naver.com, besteng83@gmail.com, jwjeon@yurim.skku.ac.kr

***** Corresponding Author: Jae Wook Jeon

Received January 15, 2014; Revised March 17, 2014; Accepted July 28, 2014; Published October 31, 2014

** Regular Paper*

Abstract: Recently, Ethernet-based Diagnostic Over Internet Protocol (DoIP) was applied to automotive systems, and in-vehicle gateways have been introduced to integrate Ethernet with traditional in-vehicle networks, such as the local interconnect network (LIN), controller area network (CAN) and FlexRay. The introduction of in-vehicle gateways and of Ethernet based diagnostic protocols not only decreases the complexity of the networks, but also reduces the update time for ECU software reprogramming while enabling the use of a range of services, including remote diagnostics. In this paper, a diagnostic gateway was implement for an automotive system, and the performance measurements are presented. In addition, a range of applications provided by the diagnostic gateway are proposed.

Keywords: Automotive gateway, Diagnostic gateway, In-vehicle network, DoIP

1. Introduction

Recently, the number of high performance electronic control units (ECU) installed in vehicles has increased significantly. ECUs provide convenient features to drivers, such as advanced driver assistance systems (ADAS) [1, 2]. Both the number of ECUs for high-performance and the size of the software for reprogramming are increased. This results in the requirement for the network to have a higher bandwidth and provide real-time functionality, but the current CAN-based networks cannot meet these requirements because of a small data size of 8 bytes and low bandwidth of maximum 1 Mbps.

To solve these difficulties, DoIP, which is an Ethernet-based diagnostic protocol, was introduced for use in automotive systems. Ethernet [3-9] can support a maximum data size of 1,500 bytes and with a bandwidth 100 Mbps. Therefore, Ethernet-based DoIP can meet the requirements for high-performance functions. Moreover, automotive systems can provide services, such as diagnosis, calibration and software updates for ECUs, and new applications that support DoIP through the in-vehicle gateway provide convenience to drivers [10].

The external diagnostic devices based on DoIP cannot interface directly with in-vehicle networks, so an in-vehicle gateway [11-14] is essential to integrate different in-vehicle networks with Ethernet. As a result, the external diagnostic devices can be connected to the automotive system. The gateway provides an interface to the external diagnostic devices to access the in-vehicle ECUs through an on-board diagnostics (OBD) terminal. If the external diagnostic device is connected to the Internet, the vehicle can provide a range of telematics services from the external diagnostic devices through the gateway [1, 15].

In this paper, a diagnostic gateway was implemented not only to integrate the different in-vehicle network protocols, such as CAN, FlexRay, and Ethernet, but also to interface the in-vehicle networks with external diagnostic devices. Moreover, the performance of the diagnostic gateway was analyzed and various applications for the diagnostic gateway with external diagnostic devices are proposed.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III describes the implementation of the diagnostic gateway system. Section IV explains the application for the diagnostic gateway. Section V presents the performance analysis of the diagnostics gateway, and Section VI concludes the paper.

2. Related Work

Many studies related to automotive gateways that integrate different networks have been conducted. One study [9] proposed that the future generation of the networks and automotive gateways will focus on methods for data conversion between different protocols, such as CAN, FlexRay and Ethernet.

In [1], the authors implemented a CAN-Ethernet gateway embedded system, which that integrates the CAN and Ethernet networks. Based on the implemented gateway, they [1] also presented methods for packing and unpacking CAN messages into and from Ethernet packets and proposed methods for monitoring, diagnostics and control of the in-vehicle networks.

Another study [11] presented a conversion method of CAN-Ethernet frame transmission in a range of scenarios (one-to-one, buffered, timed and urgency) through a field-programmable gate array (FPGA)-based gateway. Moreover, the study presented the mechanisms of protocol conversion in several cases, and the experiments were conducted with the conversion mechanisms by a real network configuration. As a result, the average end-to-end latency (from 500 Kbps CAN network to 100 Mbps an Ethernet network) was 500 µs.

The Ethernet-based DoIP diagnostic protocols were standardized by the International Organization for Standardization (ISO) [10]. The ISO 13400 standards provide global standardized interfaces for developing Ethernet-based diagnoses of vehicle communication. Each part of the ISO 13400 standards deals with the following information.

- ISO 13400 Part 1: General information and used case definition.
- ISO 13400 Part 2: Network and Transport Layer services.
- ISO 13400 Part 3: Wired vehicle interface based on IEEE802.3.

A previous study [15] on DoIP topic explained how remote online vehicle diagnostics can be realized based on the DoIP. Moreover, in studies [4] and [7], the authors announced the application of Ethernet for vehicle diagnostic in 2008 at BMW. In these studies, the reprogramming time (based on CAN network) of the 4th generation vehicle was compared with the reprogramming time (based on) of a 5th generation vehicle. Ten hours were needed to upload 81 MB in the 4th generation vehicle (based on CAN), whereas only 20 minutes was needed to upload 1 GB of software data in the 5th generation vehicle (based on Ethernet).

In the automotive gateway related studies, the DoIP was introduced, but the research related to the automotive gateways using DoIP is insufficient. Therefore, this paper focused on the implementation of the automotive gateway to support Ethernet-based DoIP. A method is also presented for the transmission of diagnostic messages through connection between the gateway and external diagnostic devices, and show the performance of the diagnostic gateway.

3. Diagnostic Gateway Implementation

This paper proposes the diagnostic gateway that supports CAN, FlexRay and Ethernet. The diagnostic gateway can be connected to the external diagnostic device through the Transmission Control Protocol (TCP), and provides the reliability and integrity of the diagnostic data through the TCP connection.

Diagnostic request messages (from the external diagnostic device to the in-vehicle networks) are packed in the TCP datagram and transmitted to the gateway. The diagnostic gateway unpacks the TCP datagram when the TCP packet is received, and separates the diagnostic information (DoIP data type, data length, target address, data payload) into the TCP datagram. Moreover, the gateway searches the configured routing table that includes a routing parameter, such as the target address, a network type of a target ECU and other network information, and transmits the diagnostic request data to the destination network domain (CAN and FlexRay).

The target ECU receives the diagnostic request messages, which perform a diagnostic operation, and transmit a diagnostic response message. The diagnostic response message is assembled in DoIP format by the diagnostic gateway. A DoIP frame is packed in the TCP datagram, and the diagnostic gateway transmits the TCP packet to the external diagnostic gateway.

The diagnostic gateway implements the basic operation, such as the connection control, routing activation handler, DoIP header handler and routing of diagnostic messages, and adds additional functions, such as security and a software reprograming.

The sub-sections, describe the diagnostic gateway system that has been implemented for the integration of the in-vehicle networks, and a test environment is constructed using the embedded systems.

3.1 System configuration

Fig. 1 presents the system configuration of the diagnostic gateway. The system consists of the following components: one gateway ECU, two ECUs connected to a High Speed CAN (HS-CAN), two ECUs connected to a Low Speed CAN (LS-CAN), two ECUs connected to a FlexRay network, and on ECU connected to an Ethernet network.

The ECU connected to Ethernet and the diagnostic gateway ECU, have been developed using the MPC5668G MCU-based evaluation board. The diagnostic gateway enables the exchange of messages between the ECUs of different networks, which are connected to different network domains and deliver diagnostic messages from an external diagnostic device to each of the ECUs.

The ECUs connected to HS-CAN and LS-CAN are based on the HCS12 MCU. The ECUs connected to FlexRay, are based on XF512. Each exchanges a diagnostic request message and response message from the ECU to external diagnostic devices passing through the gateway. When the ECUs request software reprogramming through the external diagnostic device, the ECUs receive reprogramming messages at a large scale and load these

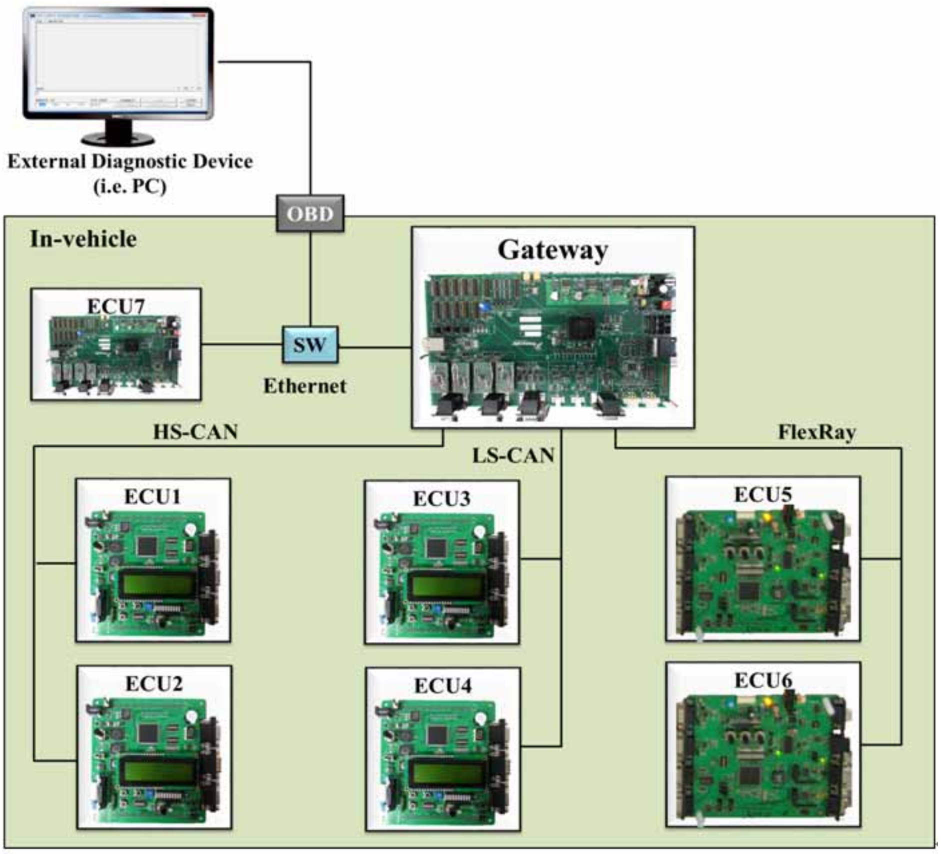

Fig. 1. Implemented embedded system.

into flash memory.

The external diagnostic device was constructed using a PC, which is was to the diagnostic gateway through the OBD terminal and provided diagnostics as well as a range of services to each of the ECUs.

3.2 Software for the Diagnostic Gateway

Fig. 2 presents the software architecture of the diagnostic gateway. The gateway software was configured to use the AUTOSAR3.1 [16] stack. An Ethernet communication stack was implemented using Light Weight IP (LWIP) [17] because the Ethernet communication stack is not supported in AUTOSAR3.1. LWIP is a dedicated, embedded TCP/IP stack. In addition, sub-modules were implemented to provide diagnostic and other services, such as DoIP, reprogramming and security.

The DoIP module can manage the active connection between the external diagnostic device and the gateway, and it routes the diagnostic messages between the external diagnostic device and the in-vehicle networks. Fig. 3 presents the process for routing the diagnostic message to the target ECU from the external diagnostic device. The gateway receives the DoIP request messages from the external diagnostic device, and the received DoIP request messages are delivered to the DoIP module through the TCP/IP stack. The DoIP module separates the protocol version, payload type, and payload length in the DoIP frame, and then checks the target ECU address in the DoIP payload. Finally, the diagnostic user data in the DoIP payload is transmitted to the network domains connected to the target ECU.

If the DoIP frame received from the diagnostic device is a reprogramming type, the DoIP module calls the reprogramming module to process the reprogramming data. The reprogramming module transmits the reprogramming data to the target ECU, but it cannot transmit the reprogramming data to the target ECU directly because the data size of the reprogramming data in the DoIP frame is relatively large compared to the maximum data size that can be transmitted through the CAN or FlexRay. Therefore, to reprogram using data with a large-size, a data division transmission that conforms to the maximum data size is needed. Accordingly, for the transmission process during reprogramming, the transport layer (TP layer) aims to divide the reprogramming data, and the transmission divides the data for the target ECU using continuous transmission control.

Because the in-vehicle networks can be accessed through the gateway, a security function is essential to prevent unwanted access. A security module processes the encryption and decryption of the diagnostic data that is exchanged between the gateway and the external diagnostic device. The gateway using the security module prevents network monitoring and undesired operation by unauthorized users. More detailed descriptions of the related security are included in [18].

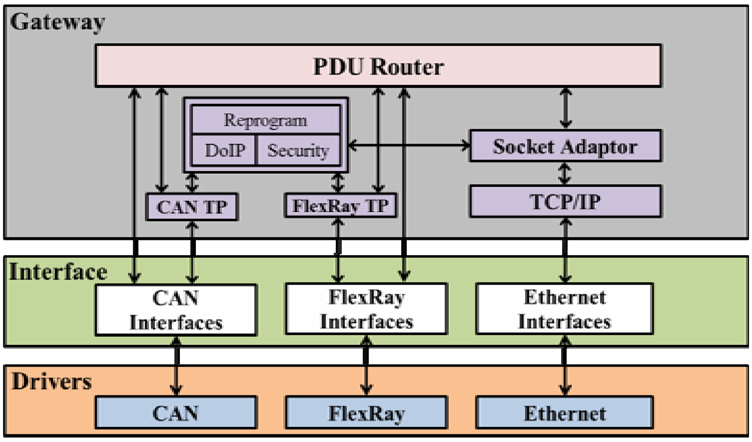

Fig 2. Software configuration for the gateway.

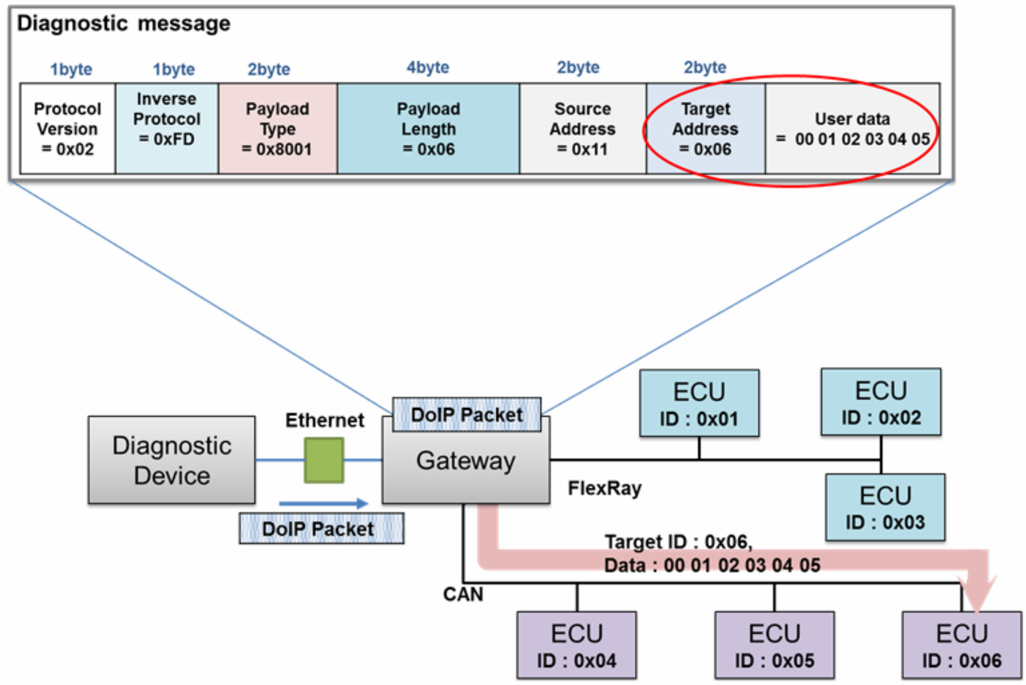

Fig. 3. DoIP messages processing.

4. Application of the diagnostic gateway

The diagnostic gateway system that is implemented can be applied as follows:

- Remote ECU diagnosis – The diagnostic gateway will connect to an external diagnostic device through Ethernet. If the diagnostic device is connected to the Internet, the condition of the vehicle can be monitored and diagnosed in real-time. The real-time monitoring of the vehicle condition allows the diagnostic device to confirm system failures immediately, and can help improve the quality of the vehicle by recording fault information.
- Fast Software update of the ECU – The software reprogramming of ECUs through CAN can be quiet time consuming. In contrast, Ethernet-based DoIP provides the 100 Mbps bandwidth and a 1,500byte message size. Therefore, DoIP can perform parallel reprogramming of in-vehicle ECUs connected to different networks. Parallel processing can be achieved if the diagnostic gateway receives data at a high speed from the external diagnostic devices and then simultaneously transfers the received data to multiple vehicle networks at a relatively slow speed. As a result, it would take much less time to update the software of all in-vehicle ECUs.
- Real time monitoring of traffic congestion – A vehicle using Ethernet can receive real-time services, such as information on the road condition from other vehicles and peripheral devices.

Table 1. Performance of the diagnostic gateway.

Type		Performance	Note
DoIP response	CAN	1 ms	-
	FlexRay	2 * (Cycle time)	-
One channel Re-programming		50 s	HS-CAN (663Kbyte)
Parallel Re-programming		58 s	HS-CAN 2-Ch (each 664Kbyte)

5. Performance analysis

Table 1 presents the performance of the implemented diagnostic gateway. Upon the diagnosis of ECUs, the total time for the diagnostic device to transmit the DoIP request message to the ECU connected to the HS-CAN, and to receive the DoIP response message from the ECU connected to HS-CAN through the diagnostic gateway is approximately 1 ms. Moreover, the total time required to exchange the DoIP request/response messages to the ECU connected to FlexRay is approximately 2 cycle times.

Using the diagnostic gateway, the software reprogramming time for the ECU connected to a one-channel HS-CAN, takes approximately 50 seconds. This time is equivalent to that required to reprogram through CAN-based external diagnostic devices. Therefore, it will take 100 seconds for the serial processing of two ECUs connected to two-channel HS-CAN. On the other hand, the re-programming time, with parallel processing of two ECUs connected to the gateway by two HS-CAN channels, lasts 58 seconds, which saves 42 seconds.

6. Conclusion

CAN-based in-vehicle networks have limitations when high performance systems are implemented (i.e., multimedia, infotainment, ADAS). Furthermore diagnostic services using CAN-based diagnostic protocols take too long because of the low bandwidth, and it is difficult to apply new services that provide convenience to drivers. In contrast, Ethernet-based DoIP provides high bandwidth and large data sizes for large-scale communication, and it can diagnose the vehicle in near real-time. With an external connection to the Internet, an external diagnostic device can request a variety of services, such as the remote diagnosis of ECUs and real-time monitoring of traffic congestion, through the in-vehicle gateway. These application services for high-performance features are made possible for drivers by introducing an in-vehicle gateway. The in-vehicle gateway integrates a traditional in-vehicle network with Ethernet, thereby providing an interface for the external diagnostics device and for the automotive system. Because of the increasing importance of in-vehicle gateways, the interface between the gateway and in-vehicle network should be studied further to optimize communication and as well as provide comfort and safety to the drivers.

Acknowledgement

This research was supported by the MOTIE and KEIT through the ATC program.

References

[1] M. Postolache, G. Neamtu, S. D. Trofin, "CAN-Ethernet Gateway for Automotive Applications," Contorol and Computing (ICSTCC), Oct 2013. pp. 422-427. Article (CrossRef Link)

[2] N. NAVET, Y. SONG, F. SIMONOT-LION and C. WILWERT, "Trends in Automotive Communication System," Proceedings of the IEEE, vol. 93, No.6, June 2005, pp. 1204-1223. Article (CrossRef Link)

[3] B. Triess, T. Hogenmüller, "Ethernet Gateway for Automotive," 2nd ethernet&ip@automotive technology day, Sep 2012.

[4] R. Bruckmeier, "Ethernet for Automotive Applications," Freescale Technology Forum, Orlando, June 2010.

[5] Toulouse, "Is Ethernet the rising star for in-vehicle networks," ETFA2011, Sep 2012. Article (CrossRef Link)

[6] A. Maier, "ETHERNET-THE STANDARD FOR IN-CAR COMMUNICATION," 2nd ethernet&ip@ automotive technology day, Sep 2012.

[7] T. Thomsen, G. Drenkhahn, "Ethernet for AUTOSAR," 1st AUTOSAR Open Conference & 8th Premium Member Conference, Detroit, USA, Oct 2008. Article (CrossRef Link)

[8] H.-T. Lim, L. Völker and D. Herrscher, "Challenges in a Future IP/Ethernet-based In-Car Network for Real-Time Applications," DAC, 2011 48th ACM/ EDAC/IEEE, June 2011, pp. 7-12. Article (CrossRef Link)

[9] L.-L. Bello, "The case for Ethernet in Automotive Communications," RTN2011, Dec 2011, pp. 7-15. Article (CrossRef Link)

[10] ISO 13400, "Diagnostic communication between test equipment and vehicle over Internet Protocol (DoIP),"

[11] A. Kern, D. Reinhard, T. Streichert and J. Teich, "Gateway Strategies for Embedding of Automotive CAN-Frames into Ethernet-Packets and Vice Versa," ARCS 2011, Feb 2011, pp. 259-270. Article (CrossRef Link)

[12] H. Zinner, J. Seitz and T. Waas, "Application and Realization of Gateways between conventional Automotive and IP/Ethernet-based Networks," DAC, 2011 48th ACM/EDAC/IEEE, June 2011, pp. 1-6. Article (CrossRef Link)

[13] Z. Rui, Q. Gui-he and L. Jia-qiao, "Gateway System for CAN and FlexRay in Automotive ECU Networks," ICINA2010, Oct 2010, pp. 49-53. Article (CrossRef Link)

[14] Freescale, "Next generation Automotive Gateways," July 2009. Article (CrossRef Link)

[15] M. Johanson, P. Dahle and A. Soderberg, "Remote Vehicle Diagnostics over the Internet using the DoIP Protocol," ICSNC2011. Article (CrossRef Link)

[16] AUTOSAR, "AUTOSAR Layered Software Architecture," Article (CrossRef Link)

[17] "Light Weight TCP/IP," Article (CrossRef Link)

[18] I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaaniche and Y. Laarouchi, "Survey on security threats and protection mechanisms in embedded automotive networks," DSN-W2013, 2013 43rd Annual IEEE/IFIP, pp. 1-12. Article (CrossRef Link)

Young Seo Lee received his B.S. degree in Information and Communi-cation Engineering from Kongju University, Cheonan, Korea, in 2014, he is currently pursuing a M.S. degree in Electronics, Electrical and Computer Engineering at Sungkyunkwan Univer-sity, Suwon, Korea. His research interests include embedded systems for auto- motive and in-vehicle networks.

Jin ho Kim received his B.S. degree in Computer Engineering in 2007, and M.S. degree in Electronic, Electrical and Computer Engineering in 2009 both at Sungkyunkwan University, Suwon, Korea. He is currently pursuing a Ph.D. degree in Electronics, Electrical and Computer Engineering at Sungkyunkwan University. His research interests include in-vehicle networks and industrial real-time networks.

Jae Wook Jeon (S'82-M'84) received the B.S. and M.S degrees in electronics engineering from Seoul National University, Seoul, Korea, in 1984 and 1986, respectively, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1990. From 1990 to 1994, he was a Senior Researcher with Samsung Electronics, Suwon, Korea. In 1994, he was an Assistant Professor with the School of Information and Computer Engineering, Sungkyunkwan University, Suwon, where he is currently a Professor. His research interests include robotics, embedded systems and factory automation.