FMI-based LIN communication virtual-ECUs simulation using AUTOSAR classic platform

Seungwook Lee, Jeonghun Cho*

Kyungpook National Univ., Kyungpook National Univ.

swookjy@knu.ac.kr, jcho@knu.ac.kr

AUTOSAR 클래식 플랫폼을 이용한 FMI 기반 LIN 통신 가상 ECU 시뮬레이션

이승욱, 조정훈
경북대학교, 경북대학교

Abstract

The number of ECUs in vehicles is increasing and becoming more complex, and about 100 ECUs are used for high-performance vehicles. The automotive industry uses a variety of communication methods for its purpose, and as the number increases, stability in communication between ECUs is becoming important. To verify this, simulations are becoming increasingly complex. In this paper, we devise a virtual simulator implementation for LIN communication using the FMI standard on the AUTOSAR standard platform. FMI is an open standard for dynamic simulation between different simulation tools. The behavior of LIN communication implemented by APIs provided by the AUTOSAR standard platform can be simulated as real-life in a virtual environment using the FMI standard. The FMI-based virtual ECU simulator proposed in this paper is expected to improve productivity by reducing the cost and time required for ECU LIN communication development and verification.

Ⅰ. Introduction

With the recent advent of intelligent vehicles, requirements for the safety and convenience of drivers are increasing. As the functions of vehicles increase and become more complex, the development of an automobile ECU (Electronic Control Unit) has been gradually advanced . As a result, the number of ECUs is increasing, and the importance of reliable communication technology between ECUs is increasing day by day. Currently, automobile ECUs apply various communication methods according to their purpose [1]. Most vehicles use CAN (Controller Area Network) communication as a major network, but LIN (Local Interconnect Network) communication is also actively used as a sub-network line of controllers. LIN communication is mainly used in communication systems between controllers that do not require high performance because it is cheaper to implement than CAN communication [2]. It is used in SPAS (Smart Parking Assist System), Power Window System, Sunroof Control System, WIPE Control System.

The global automotive industry has established and used the AUTOSAR standard software platform to improve the reuse of software. The AUTOSAR standard enables the design of ECU software regardless of hardware and improves efficiency by enabling code reuse [3]. The verification of each ECU software connected to the LIN communication system implemented through AUTOSAR requires an actual hardware-based simulation process. As the number of ECUs increases, the stability of hardware-based integrated simulations decreases, and more cost and time are required [4]. Therefore, in order to efficiently simulate ECU communication, there is a need for a method capable of simulating each modeled ECU and performing the entire simulation connected to it.

To address this problem, we devise a virtual-ECU simulator implementation on the AUTOSAR classy platform using the FMI (Functional Mock-up Interface) [5]. It is expected that the cost of development can be reduced, and the stability of the entire system can be improved by conducting virtual environment simulation without the help of hardware through v-ECU.

This paper briefly examines the AUTOSAR Classic platform and LIN communication layer structure in Chapter 2, and FMI in Chapter 3. Chapter 4 describes how to implement the v-ECU simulator on the AUTOSAR classy platform using FMI. Finally, Chapter 5 concludes the results of this paper.

Ⅱ. AUTOSAR LIN Communication Stack

This chapter introduces the LIN communication structure of the AUTOSAR classy platform.

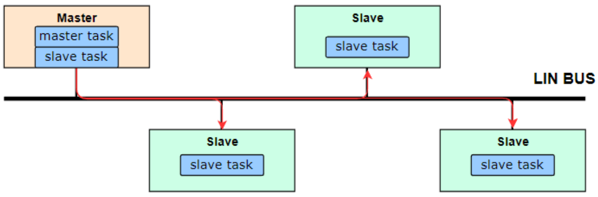

Figure 1. LIN Communication Structure

Figure 1 shows the general flow of message transmission in LIN communication. One master node and one or more slave nodes are connected to a single line LIN BUS. The master task exists only in the master node and determines the transmission timing and transmission target of the frame. The Slave task enters both the Master Node and the Slave Node and receives a frame. When the master node transmits a message to a slave node connected through a BUS, the slave node is a structure that receives a message after checking the identifier.

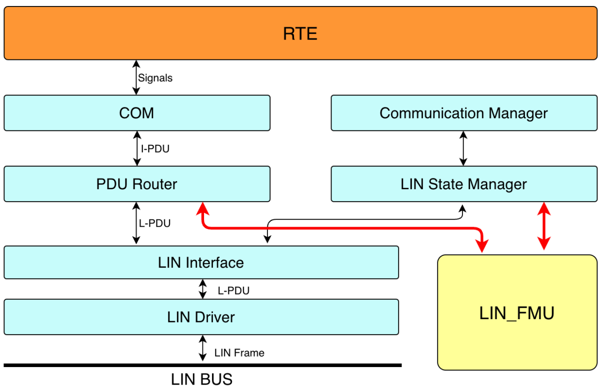

Figure 2. AUTOSAR LIN communication structure with LIN_FMU

AUTOSAR (AUTomotive Open System ARchitecture) is a software standard platform for automotive ECU development and can be divided into three major hierarchies. Application Software, Run-Time Environment, Basic Software. Figure 2 shows the flow of AUTOSAR LIN communication. RTE (Run-Time Environment) is a layer located between ASW (Application Software) Layer and BSW (Basic Software) layer. RTE provides an interface that allows data sent from software components to be accessed in the BSW module. The signal transmitted to the BSW layer through the RTE is transmitted to the COM module [6]. Pack the signal received from the application with the PDU (Protocol Data Unit) and transmit it to the PduR [7]. Upon receiving the PDU, the PDU Router transmits the PDU to the LIN Interface module according to the set routing table [8]. The LIN interface module transmits the received LIN frame data to the LIN driver in the form of a LIN frame [9]. The transmitted PDU is converted from the LIN Driver to the LIN Frame and transmitted through the LIN BUS.

Hardware-based simulation is required to confirm whether the LIN communication operated in the above structure has operated normally.

Ⅲ. Functional Mock-up Interface

The Functional Mock-up Interface (FMI) standard defines an application programming interface (API) for simulation between different dynamic models using XML files and C code. FMI standard is maintained as a Modelica Association Project and support the AUTOSAR standard. The vECU can be implemented as a functional mock-up unit (FMU) using the FMI standard API. Using the generated FMUs, ECUs designed in different systems may be simulated as in real life in a virtual environment.

FMI3.0 define three interface types. CS(Co-Simulation), ME (Model Exchange), SE (Scheduled Execution) [5]. In this paper, we devise a simulator using CS type. The CS type FMU contains its own solver or scheduler. It is a standard interface for the interworking of two or more simulation tools in a simultaneous simulation environment. A solver or scheduler is located inside the FMU, and the Co-Simulation Algorithm can be input from the outside. Since solver can be implemented inside the CS FMU, unlike the ME type, the importer does not have to provide solver.

Ⅲ. LIN_FMU

FMU in the form of an XML file that meets the AUTOSAR standard can be generated through FMI. The generated FMU is an XML document operating as a virtual ECU in an AUTOSAR environment. We implement a virtual ECU for transmission and reception of frames, control of communication states, and control of schedules, which are key functions of AUTOSAR LIN communication.

The red line in Figure 2 represents the flow chart of the simulation for the transmission process of LIN communication. The signal received from the upper layer is packed in Com and put in PDU, and PDU is delivered to LIN_FMU based on PduR routing table. The simulation is carried out in the form of returning information on whether the PDU received from the LIN_FMU was normally received through the upper layers.[10]

The communication state control controls the time point of communication, detects an error, and performs a task. The status of the LIN transceiver hardware can also be controlled in normal, sleep, and standby modes. It is also possible to implement a function that delivers notifications through RTE in the event of an error. When the hardware state is specified in the upper layer, the virtual simulation is performed by changing the state of the LIN_FMU and returning the state.

In schedule control, the master task of the master node transmits a header by defining a schedule table. The order of schedule transfer can be defined by setting the frame and delay to be transmitted, and the Slave Task determines whether to receive or not by viewing the id of the header. It is possible to return whether LIN_FMI performed normally on the requested schedule and the schedule change state, and to determine whether it operates normally in the upper layer.

Ⅳ. Conclusion

Recently, the number of ECUs is increasing because of the high performance of vehicles, and accordingly, simulations are becoming more complicated, and the time and cost required are increasing. It is possible to reduce the time and cost required for testing in the ECU development process through simulation in a virtual environment rather than a hardware-based actual simulation. In this paper, a virtual simulator for LIN communication on the AUTOSAR Classic platform was devised using FMI. It is possible to implement virtual simulators of other ECUs as well as LIN communication through the FMI standard. It is expected that the simulation can be conducted more efficiently, and the stability of the whole system can be improved by applying the virtual simulator.

ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. 1711160343, Development of virtual ECU-based vehicle-level integrated simulation technology for vehicle ECU application software development and verification automation)

REFERENCES

[1] M. S. Alam, S. Iqbal, M. Zulkernine, and C. Liem, “Securing vehicle ECU Communications and stored data,” ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 2019.

[2] J.-B. Lee and S. Lee, “Design and verification of Automotive Lin Controller,” Journal of IKEEE, vol. 20, no. 3, pp. 333- 336, 2016.

[3] Furst, S., & Bechter, M. (2016). Autosar for connected and autonomous vehicles: The Autosar Adaptive Platform. 2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W). https://doi.org/10.1109/dsn-w.2016.24

[4] Altinger, H., Wotawa, F., & Schurius, M. (2014). Testing methods used in the automotive industry: Results from a survey. Proceedings of the 2014 Workshop on Joining AcadeMiA and Industry Contributions to Test Automation and Model-Based Testing - JAMAICA 2014. https://doi.org/10.1145/2631890.2631891

[5] Functional mock-up interface specification. (n.d.). Retrieved November 29, 2022, from https://fmi-standard.org/docs/3.0/

[6] AUTOSAR, “Specification of Communication”, 2022.[Online]. Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SWS_COM.pdf

[7] AUTOSAR, “Specification of PDU Router”, 2022.[Online]. Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SWS_PDURouter.pdf

[8] AUTOSAR, “Specification of LIN Interface”, 2022.[Online]. Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SWS_LINInterface.pdf

[9] AUTOSAR, “Specification of LIN Driver”, 2022.[Online]. Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SWS_LINDriver.pdf

[10] AUTOSAR, “Requirements on LIN”, 2022.[Online]. Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SRS_LIN.pdf