Transactions of KSAE, Vol. 28, No. 12, pp.851-858 (December, 2020)

pISSN 1225-6382 / eISSN 2234-0149
DOI http://dx.doi.org/10.7467/KSAE.2020.28.12.851

차량 보안을 고려한 게이트웨이의 라우팅 방법에 대한 연구

박 진 서*[1] · 이 성 준[2] · 서 일 홍[3]

인피니언 테크놀로지스 아시아 퍼시픽 전략기획팀[1] · 한양대학교 융합공학과[2] · 한양대학교 융합전자공학부[3]

The Study of Routing Methods on the Secure Vehicle Gateway System

Jin Seo Park*[1] · Songjun Lee[2] · Il hong Suh[3]

[1]*SMA, Infineon Technologies Asia Pacific Pte Ltd, 8 Kallang Sector, Singapore 349282, Singapore*
[2]*Department of Electrical and Electronic Engineering, Hanyang University, Gyeonggi 15588, Korea*
[3]*Engineering College Department of Electronic Engineering, Hanyang University, Seoul 04763, Korea*

(Received 10 September 2020 / Revised 22 September 2020 / Accepted 25 September 2020)

Abstract : Vehicle networks are becoming increasingly complex due to the demands of autonomous driving and connectivity. This increment of data complexity requires higher bandwidth communication, and automakers are using Ethernet-based communications for this purpose. In order to use existing in-vehicle communication with high-speed Ethernet communication, an extended gateway system is used in the vehicle. The system uses Ethernet as the interface for the external controller and the gateway, and CAN as the interface for the communication between the internal controllers. Vehicle gateway provides an interface for exchanging vehicle data and connecting in real time in a heterogeneous communication environment between the existing CAN network protocol and the external Ethernet protocol. In this routing operation, time delay occurs, and in the in-vehicle network environment where communication time is important, this delay time must be considered and measured during the development process. Since the involvement of the external communication network causes a security vulnerability, a security function that guarantees the integrity of the message against cybersecurity risks should be considered. In this paper, we investigate the effects of delayed gateway routing on CAN frames and Ethernet frames, including security systems. For the above investigation, quantitative comparison and analysis are performed by using two routing methods: direct routing method without data modification when routing from CAN frame to Ethernet frame, and indirect routing method for selecting and transmitting the necessary data. For security, CMAC is used in order to ensure the integrity of the CAN network messages and Ethernet messages. Generation and verification of the CMAC uses the hardware security module(HSM) that is built into the Infineon 32-bit MCU, TC397.

Key words : Secure communication(보안 통신), HSM(하드웨어 보안 모듈), CMAC(암호 기반 메시지 인증 코드), AUTOSAR(오토사), Tunneling(터널링)

1. 서 론

게이트웨이는 차량내부의 다양한 제어기들을 기능별 분류하고 효율적으로 관리하는 데 사용된다. 특히, 제어기 수가 증가함에 따라 게이트웨이의 통신 채널수도 증가하는 추세다. 또한 증가하는 데이터 양을 처리하기 위해 높은 통신속도가 요구된다. 이러한 추세에 따라 CAN 통신은 점차 CAN-FD로 이동하고 있으며 10개 이상의 채널이 필요한 게이트웨이 시스템이 소개되고 있다.

다양한 제어기들이 외부와 통신하기 위해, 게이트웨이에 이더넷 또는 기가비트 이더넷이 사용된다. 최근의 게이트 웨이는 16개 이상의 CAN /CAN-FD 채널과 이더넷 /기가비트 이더넷 채널을 요구하기도 한다.

Fig. 1은 차량용 게이트웨이의 예를 보여주고 있다. 엔진제어와 변속기제어 등 구동 관련된 제어기들이 구동관련 도메인으로 하나의 CAN 버스에 연결되어 있고, 제동, 조향, 현가 등 샤시관련 제어기들이 샤시 도메인으로 별도의 CAN 버스에 연결되어 있다. 외부와 무선통신을 하기 위한 텔레매틱스는 많은 양의 데이터 송수신을 위해 빠른 통신을 지원해야 하므로 이더넷을 통해 게이트웨이

*Corresponding author, E-mail: jerry.park@infineon.com

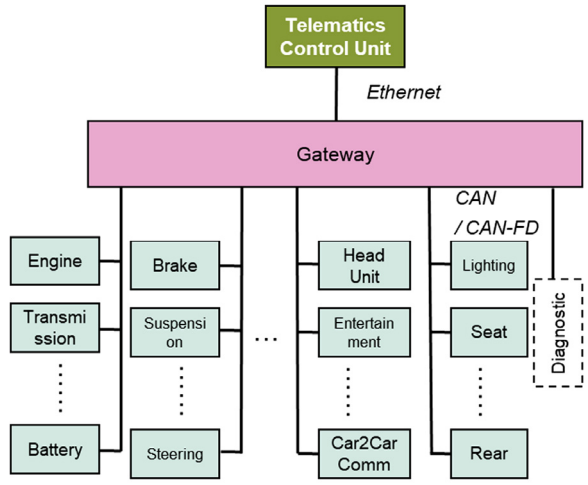

Fig. 1 The structure of vehicle gateway[8)]

와 연결된다.

이와 같이 게이트웨이가 텔레매틱스와 개별제어기들의 인터페이스 역할을 하는 구조에서 외부 해커가 개입하여 텔레매틱스를 통해 게이트웨이 및 개별제어기들에 허가되지 않은 데이터를 전송할 수 있다. 또한 해커가 차량내부에서 게이트웨이의 통신버스에 침입한다면, 제어기들간의 통신을 위한 버스에 허가되지 않은 데이터를 삽입할 수 있다. 게이트웨이 구조에서 보안의 취약점으로 인해 자동차 제조사들은 다양한 보안방법을 검토하고 있고 적용을 위해 HSM(Hardware Security Module)기반 게이트웨이 보안 시스템을 개발하고 있다.

그러나 보안시스템을 적용하는 경우 게이트웨이 내부 코어의 부하가 증가하고 라우팅 지연시간이 증가하게 된다. 시스템 부하증가로 인한 RTOS (RealTime OS)오류 및 라우팅 지연으로 인한 통신 지연 가능성이 있으므로 이를 단계별 분석을 통해 시스템 설계 시 고려되어야 한다.

본 논문은 게이트웨이의 지연시간을 분석하기 위해 게이트웨이 라우팅 지연시간을 단계별로 분리하여 정량적으로 분석한다. 또한 자동차 게이트웨이에 적용가능한 라우팅 방법들을 정의하고 제안된 각 방법들의 지연시간을 실험적으로 보여준다. 이와 같이 제안된 방식과 정량적인 결과를 이용하여 게이트웨이 설계자는 효율성, 보안레벨등 시스템 요구사양에 맞게 최적화된 게이트웨이 컨셉설계를 할 수 있다. 그리고, 설계자는 요구되는 시스템 리소스를 정의하고 지연시간을 예측할 수 있다.

2. 게이트웨이의 메시지 보안과 전송 방법

현재 자동차 게이트웨이에서 사용되는 라우팅방법은 크게 두가지로 분류된다. 데이터 전송 방법에 따라 하나의 버스에 실리는 메시지를 수정없이 원본을 다른버스에

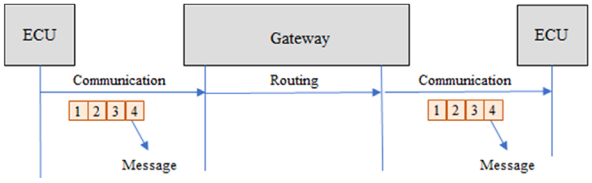

Fig. 2 Direct routing : Transfer without data change[10)]

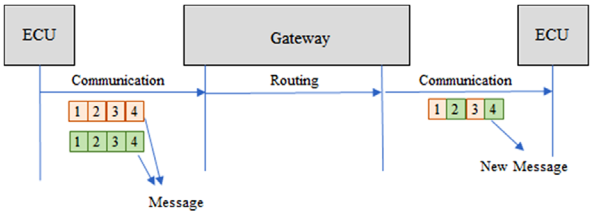

Fig. 3 In-direct routing : Transfer after data selection

Table 1 Two message transmission methods at the gateway

Routing	Description	Implementation
Direct	Transmit all received data to other bus without change	Store all received data in the gateway buffer
In-direct	Selectively transmit to other bus among received data	Store only some data among the received data in the gateway buffer

전송하는 방법과, 일부 메시지만 선별하거나 데이터를 가공하여 다른 버스로 전송하는 방법이 있다. 전체 메시지를 수정없이 원본을 다른 버스로 전송하는 방법을 직접 라우팅 방법이라 하고(Fig. 2 참조), 메시지의 일부를 선별하여 재조합하거나 가공하여 다른 버스에 전송하는 방법을 간접 라우팅 방법이라 한다(Fig. 3 참조).

라우팅은 CAN-to-CAN과 같이 동일한 유형의 통신간에 이루어 질 수 있고,CAN-to-Ethernet과 같이 서로 다른 유형의 통신간에 이루어질 수도 있다.

2.1 MAC(Message Authentication Code) Use Case

수신자가 메시지를 수신할 때, 올바른 발신자가 보낸 메시지인지 혹은 허가받지 않은 해커가 보낸 메시지인지 판별하기는 매우 어렵다. Fig. 4는 Attacker에 의해 인증되지 않는 Fake Message가 전송되는 것을 보여준다.

따라서 메시지를 안전하게 전송하기 위해 무결성과 인증이라는 두 가지 속성과 절차가 요구된다. 메시지의 무결성은 메시지가 위변조되지 않았음을 입증하고, 메시지 인증은 메시지가 올바른 발신자로부터 송신된 것임을 의미한다.

메시지 인증 코드 (MAC)는 데이터가 변조되었는지 확인하기 위한 방법이다.[12)] 수신자와 발신자만 알고 있는 개인보안 키를 사용하여 MAC 값을 생성하고, 이 값을 이용하여 아래 Fig. 5와 같이 전송 된 메시지의 무결성을 확

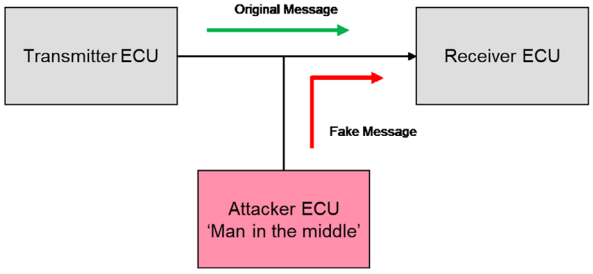

Fig. 4 Fake message sent by attacker

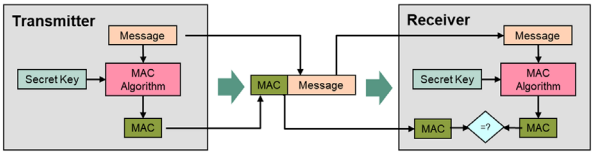

Fig. 5 Message authentication and verification for secure communication[2)]

인 및 인증할 수 있다.[3)]

Fig. 5의 발신자는 송신할 메시지와 개인보안 키를 사용하여 MAC값을 생성하고, 송신할 메시지와 MAC값을 함께 송신한다. 수신자는 메시지와 MAC을 함께 수신한 후, 수신된 메시지와 개인보안 키를 사용하여 MAC값을 직접 생성한다. 수신자가 수신한 MAC값과 직접 생성한 MAC값을 비교하여 일치하면 메시지의 무결성과 인증 절차를 마치게 된다.

MAC값을 계산하기 위해 다양한 알고리즘이 존재한다. 본 논문에서는 현재 자동차 보안에서 가장 대중적으로 적용되는 Blockchipher기반의 CMAC방식을 사용하여 실제 차량 시스템과 동일한 환경으로 구현 및 시험한다.

2.2 CMAC (Cipher-based Message Authentication Codes)

AUTOSAR의 'Specification of Secure Onboard Communication'은 NIST SP800-38B에 따라 AES-128 기반 CMAC 알고리즘을 사용하여 MAC을 계산한다.[9)] CMAC(Password Based Message Authentication Code)는 비밀키와 결합 된 블록암호를 사용하여 메시지 인증 코드를 계산한다. 그러므로, CMAC를 사용하여 메시지의 무결성과 인증을 확인할 수 있다. CMAC는 오류에 의한 데이터 변경뿐 아니라 의도적인 위변조도 감지할 수 있다. 최근 자동차 제조사는 CMAC을 안전하고 효율적으로 계산하기위해 Microcontroller에 내장된 HSM(Hardware Security Module)을 사용한다.

Fig. 6은 HSM 내장형 Microconroller의 아키텍처를 보여준다. HSM 은 Firewall에 의해 고립되어 있는 구조로써, 기존 Core들은 극히 제한된 통로 이외에는 HSM의 내부를 접근할 수 없다. 이와 반대로 HSM은 Firewall외부의

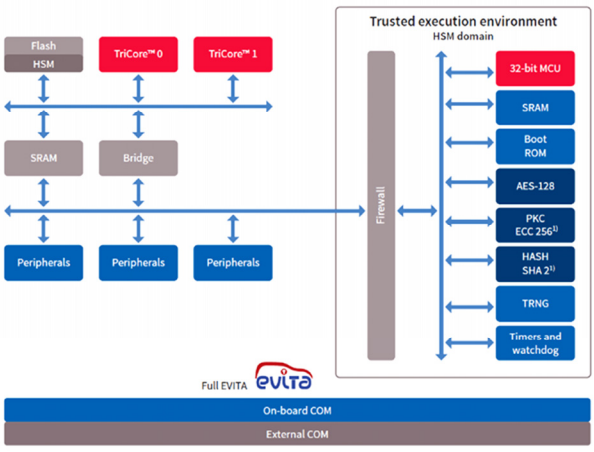

Fig. 6 Hardware Security Module (HSM)[11)]

영역을 자유롭게 접근할 수 있다. HSM 내부에는 AES-128과 같은 암호화모듈을 내장하고 있다.

2.3 게이트웨이의 라우팅

라우팅은 게이트웨이 제어기의 가장 기본적인 기능 중 하나이다. 하나의 통신 버스에서 다른 통신 버스로 메시지 및 데이터를 전송하는 작업을 의미한다. 동일한 유형의 버스간에(예. CAN-to-CAN) 또는 서로 다른 유형의 버스간에(예. CAN-to-Ethernet) 데이터를 라우팅 할 수 있다. 모든방식의 라우팅은 제한된 시간안에 정확하게 메시지가 전달되는 것이 매우 중요하다. 최근에는 제어기 수의 증가, 데이터양의 증가, 보안을 위한 격리 등의 이유로 통신 버스수를 늘리고 있고, 이더넷과 같은 고속 통신 프로토콜이 적용되고 있다. 이와 같은 변화에도 네트워크간의 정보교환은 항상 정확하고 빠른 통신이 요구된다.

게이트웨이가 서로 다른 유형의 버스에 대한 라우팅을 수행할 때, 한 유형의 프레임이 다른 유형의 프레임으로 캡슐화 되어야 하며, 일반적으로 작은 대역폭의 버스에서 더 큰 대역폭을 가진 다른 버스로 전달되는 경우가 해당된다(예. CAN-to Ethernet). 이어서 IP 또는 MAC주소를 사용하여 캡슐화 된 데이터가 최종수신제어기에 전달된다. 이와 같은 작업을 터널링이라 한다. 수신제어기는 캡슐화된 데이터패키지를 해제시키고 이 데이터를 시스템 RAM에 저장한다.

실시간 터널링을 하는 이유는 ECU를 직접 연결하지 않고도 한 도메인에서 다른 도메인(예: 인포테인먼트 시스템에 표시되는 자동차 속도)으로 데이터를 공유하는 것이다. 이와 같은 방식은 시스템의 유연성을 높이고 케이블사용을 줄일 수 있다. 도메인 아키텍처 및 Gbit Ethernet 백본 외에 ADAS, 센서 퓨전, 인포테인먼트 도메

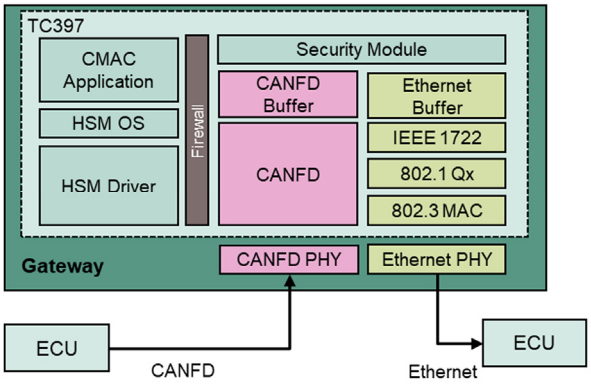

Fig. 7 Routing structure of CAN-to-Ethernet[5)]

인의 도입으로 이와 같은 사용예는 더욱 증가하고 중요해지고 있다. 실시간 터널링의 중요한 요소는 라우팅 대기 시간이다(즉, 패킷이 송신측에서 수신측으로 라우팅되는 데 걸리는 시간).

Fig. 7은 게이트웨이가 CAN-to-Ethernet 라우팅을 수행하기 위한 시스템 아키텍처를 보여준다. 게이트웨이는 CAN통신을 통해 메시지를 수신한 후, HSM을 통해 메시지의 인증과 무결성을 확인한다. 이어서 이더넷에 맞게 터널링작업을 수행하여 이더넷용 버스로 송신을 한다.

2.4 게이트웨이 라우팅 분석의 필요성

서로 다른 유형의 버스에 대한 라우팅 수행과 높은 보안레벨의 적용 등으로 지연시간과 코어 부하가 크게 증가하고 있다. 컨셉설계 단계에서 코어부하와 지연시간의 원인에 대한 정확한 분석과 예측이 없다면, 자동차 제조사의 시스템 요구사양에 최적화된 게이트웨이 라우팅 사양설계가 이루어 질 수 없다. 또한, 요구성능을 만족하기 위한 부하 예측 및 적합한 코어 제품을 선택하는 것도 어렵다. 그러므로, 체계적이고 정량화된 분석/설계 프로세스를 확립하고 적용하는 것이 필요하다.

3. 게이트웨이 라우팅 설계 및 구현

3.1 게이트웨이의 보안 전송 방안

본 논문은 게이트웨이가 메시지를 라우팅하기 위해 적용 가능한 세가지 방식을 제안한다. 첫째, 직접 라우팅 시 수신 메시지의 MAC무결성을 확인하지 않는 방식(Fig. 8 참조), 둘째, 직접 라우팅시 수신 메시지의 MAC검증을 포함하는 방식(Fig. 9참조), 셋째, 간접 라우팅시 수신 메시지의 MAC검증과 메시지의 데이터를 선별 및 가공 후 메시지로 계산한 새로운 MAC을 생성하여 송신하는 방식(Fig. 10 참조) 등이다.

본 논문은 게이트웨이의 메시지 무결성 확인과 인증

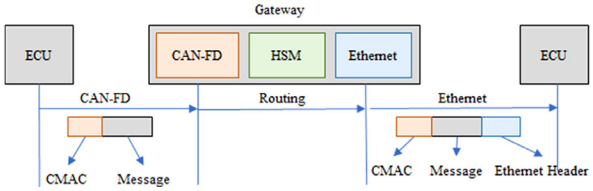

Fig. 8 Direct routing (without CMAC verification)

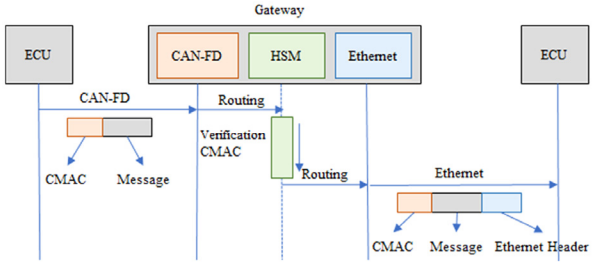

Fig. 9 Direct routing (with CMAC verification)

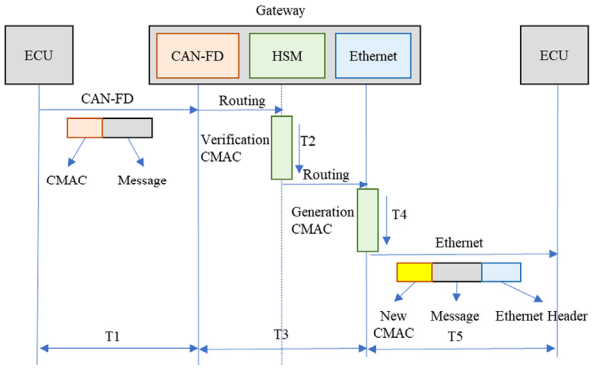

Fig. 10 In-direct routing (with CMAC verification and generation)

절차를 위해 CMAC을 사용한다. 본 논문이 제안한 세가지 라우팅 방안은 Table 2에 상세히 설명된다.

3.2 게이트웨이 지연시간 분석

게이트웨이 지연시간을 분석하기 위해, 지연시간을 전송단계에 따라 분리하고 각 단계별 시간을 실험과 계산을 통해 파악해야한다. Fig. 11은 안전한 메시지 전송을 위한 게이트웨이 시스템의 전송단계를 간략하게 보여준다. Fig. 11의 T1, T2, T3, T4 및T5는 단계별 지연 시간을 나타낸다. 게이트웨이의 단계별 지연 시간은 Table 3에 상세히 설명되어 있다. T1과 T5는 통신속도와 메시지 프레임 크기를 계산하여 파악할 수 있고, T2와 T4는 HSM과 코어의 성능과 소프트웨어구조에 따라 다른 결과를 얻으므로 실험을 통해 파악할 수 있다. T3는 코어의 성능, 소프트웨어구조, 라우팅을 위한 IP종류 등에 따라 다른 결과를 얻으므로 실험을 통해 파악할 수 있다. 게이트웨이의 총전송 지연 시간은 앞서 제시된 세가지 전송모드에 따라 각각 다르게 나타난다.

Table 2 Three message routing methods considering security

Routing	Implementation
Direct (without CMAC verification)	Store all incoming data on the gateway Sending the same data in different communication methods
Direct (with CMAC verification)	Store all incoming data on the gateway after CMAC integrity verification Sending the same data in different communication methods
In-direct (with CMAC verification and generation)	Store all incoming data on the gateway after CMAC integrity verification CMAC is created after selecting required data Sending in different communication methods

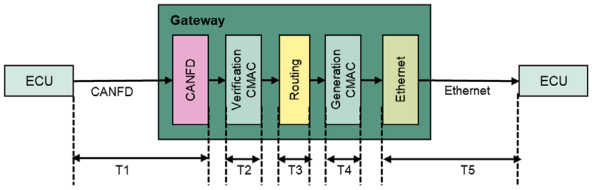

Fig. 11 Routing process of CANFD to Ethernet[6)]

Table 3 Latency by routing steps

Latency	Description
T1	CANFD message transmission time (ECU to Gateway)
T2	CMAC verification time
T3	Routing time in Gateway
T4	CMAC generation time
T5	Ethernet message transmission time (Gateway to ECU)

CMAC 검증없이 직접 라우팅하는 경우 CAN 및 다른 크기의 CAN-FD 메시지는 IEEE 1722 이더넷 프레임으로 즉시 캡슐화된다. 이어서 CMAC 검증없이 1722 이더넷 프레임으로 전송된다. 이 경우 T2와 T4 시간을 요구하지 않고, Microcontroller내부의 HSM 모듈은 사용되지 않는다.

CMAC 검증을 포함한 직접 라우팅 전송의 경우, CAN-FD 메시지는 이더넷 프레임에 캡슐화되고 CMAC 값에 대한 검증 후 전송된다. 이 경우, CMAC검증을 위해 T2시간이 요구되고 HSM모듈이 사용된다. CMAC 검증을 위한 추가 지연 시간, T2는 64 바이트 페이로드 CAN-FD 프레임의 경우 42 us이다.

CMAC을 검증 및 생성해야 하는 간접 라우팅 전송의 경우, CAN-FD 메시지는 이더넷 프레임에 캡슐화되고, 수신된 CMAC값을 검증하고 변경된 송신 데이터에 맞는 신규 CMAC값을 재생성 후 전송된다. 그러므로, 간접 라우팅은 CMAC값 검증을 위한 T2와 신규 CMAC값 계산을 위한 T4시간이 요구된다.

Table 4는 각 라우팅 방법에 따른 지연시간을 보여준다.

Table 4 Total latency according to routing method (Description)

Routing	Total latency
Direct (without CMAC verification)	T1 + T3 + T5
Direct (with CMAC verification)	T1 + T2 + T3 + T5
In-direct (with CMAC verification and generation)	T1 + T2 + T3 + T4 + T5

4. 실 험

4.1 하드웨어 실험 환경

본 논문은 실험환경 구현을 위해 Realtek의 RTL 9047AA 이더넷 스위치와 함께 AURIX™ TC397를 사용한 두개의 Infineon 자동차 게이트웨이 평가 보드를 사용하였다. Aurix마이크로 컨트롤러 TC397는 데이터 전송을 위해 1Gbps RGMII포트와 스위치에 연결하여 사용했다.[1)]

CAN 프레임은 2개의 Peak PCAN-USBPro FD 장치에 의해 생성된다. CAN-FD 프레임은 2Mbps Data 프레임(CAN의 경우 1Mbps) 및 1Mbps Arbitration Phase로 구성된다. 2개의 PCAN 장치는 게이트웨이 보드 #1의 8개의 CAN 포트에 연결된다. 마지막으로, 로직 분석기는 지연시간을 측정한다(Fig. 12 T1참조). 게이트웨이 보드 #1이 수신 한 CAN 프레임은 로컬라우팅 테이블을 사용하여 라우팅되고 CAN 포트에서 이더넷 인터페이스로 포맷된다. 사용된 이더넷 프레임 형식은 IEEE 1722-2016 표준에 의해 정의된 ACF(AVTP Control Format)이다(Fig. 12 T3 참조).[2)] ACF는 Time Sensitive Networking Ethernet을 사용하여 다양한 제어 메시지를 시간 동기화 또는 비동기화 방식으로 전송하기 위한 유연한 프레임 워크를 제공한다. 본 논문은 실험을 위해 CAN / CAN FD ACF 메시지를 사용한다. Fig. 13은 IEEE 1722 프레임에 CAN프레임이 캡슐화 되어있는 것을 보여준다. CAN 프레임이 CAN/CAN-FD ACF 이더넷 프레임에 캡슐화되어 있고 프레임이 1Gbps 이더넷 인터페이스를 통해 게이트웨이 보드 #2로 전송된다(Fig. 12 T5참조). 게이트웨이 보드 #2는 이더넷 프레임을 수신하고, 이 프레임에 포함된 CAN 프레임을 추출하여 라우팅 테이블을 확인한 후 CAN 프레임을

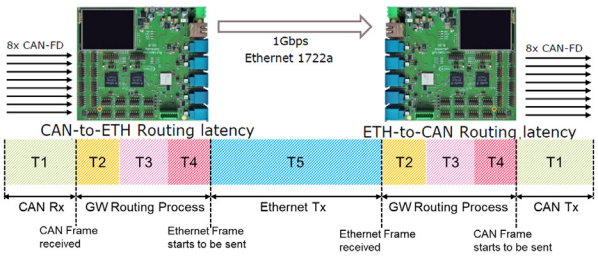

Fig. 12 Gateway routing experiment environment

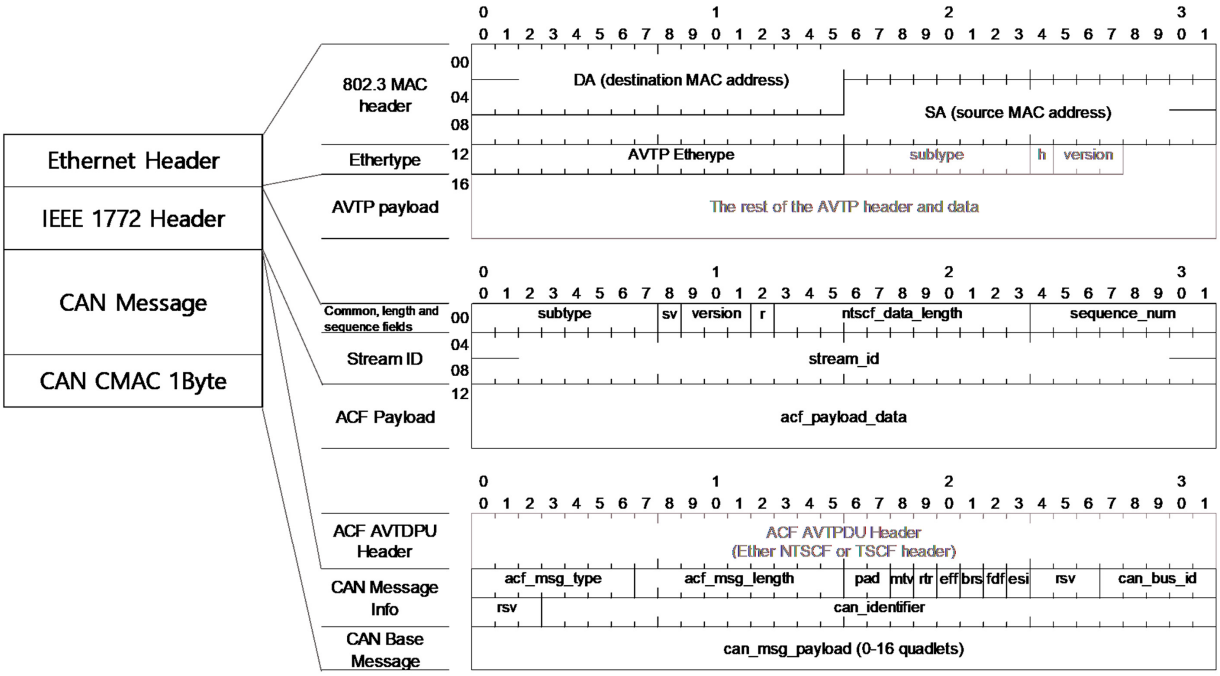

Fig. 13 CAN/CANFD AVTP format (IEEE 1722-2016)[4]

Table 5 Routing experiment environment

CPU clock		300 MHz
Number of channels (CAN/CANFD)		8
CAN	Baudrate	1 Mbps
	Payload size	1/8 Bytes
CANFD	Baudrate (Arbitration)	1 Mbps
	Baudrate (Data)	2 Mbps
	Payload size	1/8/64 Bytes
Ethernet speed		1 Gbps
HW acceleration		CRC

오른쪽 CAN 인터페이스로 라우팅한다. 실험 환경 설정의 개요는 Fig. 12에서 볼 수 있다. Aurix TC397은 6개의 TriCore CPU 가 내장되어 있지만, 이 실험 환경은 1 CPU 만사용한다. 이 실험환경을 구현하기 위해 이더넷 스택에 Infineon Low-Level Drivers(ILLD)와 오픈소스 라이브러리 LwIP(Lightweigh IP)를 사용한다. 하드웨어 가속은 CRC 계산을 위해 사용된다. Table 5는 사용된 자원을 요약하여 보여준다.

4.2 실험 결과

Fig. 14와 Fig. 15는 전송 지연(T2)에 대한 고려 없이 페이로드 크기에 대한 평균라우팅 지연 시간을 보여준다. CAN과 CAN-FD비교시 라우팅 지연은 큰차이가 없고, 페이로드의 크기에 따른 라우팅 지연차이도 매우 제한적인 것을 볼 수 있다. CAN은 최대 8bytes의 Payload를 지원

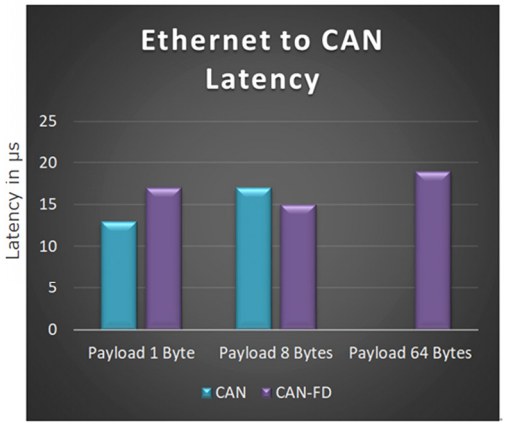

Fig. 14 Comparison routing latency of Ethernet to CAN/CANFD

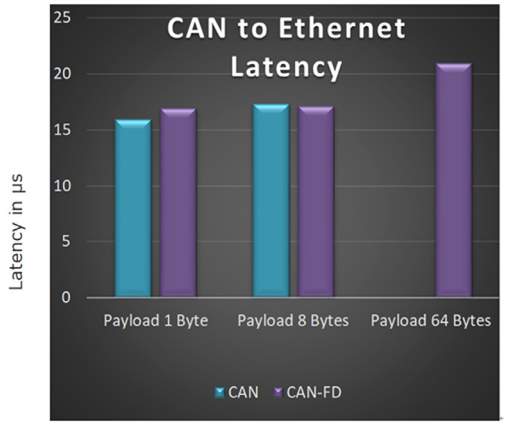

Fig. 15 Comparison routing latency of CAN/CANFD to Ethernet

Table 6 Total latency according to routing method (Test result [us])

Routing	T1	T2	T3	T4	T5	Total latency
Direct (without CMAC verification)	350		20		2	372us
Direct (with CMAC verification)	350	42	20		2	414us
In-direct (with CMAC verification and generation)	350	42	20	42	2	456us

하므로(Table 5참조) Fig. 14와 Fig. 15는 64bytes Payload에 대해 CAN-FD의 결과값만을 표시한다.

이어지는 실험에서 게이트웨이 성능은 64 바이트 CAN-FD 메시지가 이더넷 프레임으로 캡슐화되어 전송되는 시간을 측정했다. Table 6은 라우팅 방법에 따라 측정된 시간을 포함한다.

결과적으로 게이트웨이에서 고정된 지연 시간은 T1, T3 및 T5이고, 라우팅 방법에 따라 고려되야 하는 지연시간은 CMAC 검증 시간 및 CMAC 생성 시간 인 T2, T4이다. 위 실험은 각각의 T2 및 T4가 42 μs 시간이 소요됨을 보여준다. 그러므로, 라우팅 방법에 따른 변동 시간은 CMAC 사용여부에 따라 다르게 나타난다.

5. 라우팅 적용 방안

게이트웨이의 라우팅을 설계 및 적용할 때, 자동차 제조사는 정의된 사양에 따라 본 논문이 제시한 3 가지 전송 방법 중 가장 적합한 방법을 선택해서 사용해야 한다.

Fig. 8, Fig. 9, Fig. 10에서 제안된 세가지 방법에 따른 각각의 총 지연시간분석을 아래 Fig. 16, Fig. 17, Fig. 18에서 보여준다.

Table 7 Pros and Cons according to routing methods

Routing	Pros	Cons
Direct (without CMAC verification)	Gateway doesn't perform security related functions Fast routing speed	Can't check the integrity of received message Transfer unnecessary data together Increasing communication inefficiency
Direct (with CMAC verification)	Received message integrity check	Transfer unnecessary data together Increasing communication inefficiency
In-direct (with CMAC verification and generation)	System security level is raised by regenerating CMAC	Increased latency due to CMAC verification and regeneration

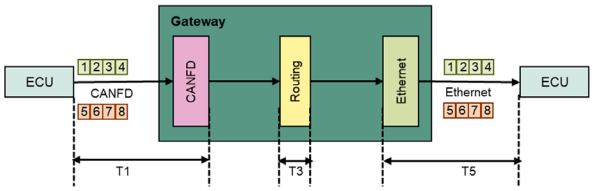

Fig. 16 Direct routing (without CMAC verification)

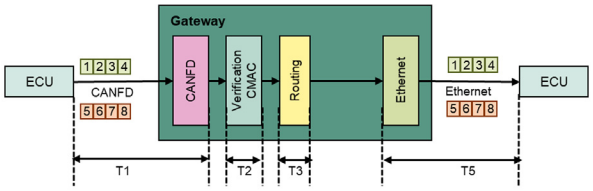

Fig. 17 Direct routing (with CMAC verification)[7)]

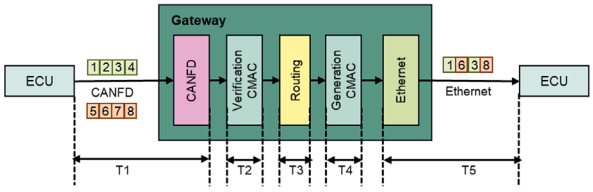

Fig. 18 In-direct routing (with CMAC verification and generation)[7)]

Fig. 16의 직접 라우팅 방법은 시스템의 복잡도는 낮고, 메시지를 재구성 및 가공할 필요가 없고, 보안기능을 수행할 HSM을 필요로 하지 않는다. 그러므로 데이터에 의한 버스부하가 높지 않고 가격에 민감한 저가 시스템에 적용하기 적합한 방법이다.

Fig. 17의 직접 라우팅 방법은 시스템의 복잡도는 낮고, 메시지를 재구성 및 가공할 필요가 없다. 그러므로 데이터에 의한 버스부하가 높지 않고 입력받은 메시지에 대한 보안검증 수행해야 하는 시스템에 적용할 수 있다.

Fig. 18의 방법은 데이터의 높은 무결성이 요구되고, 많은 데이터양에 의해 높아지는 버스부하를 관리하기 위해 필요한 데이터를 선택하여 전송한다. 이 경우 많은 메시지를 실시간으로 라우팅해야 하고 동시에 보안기능을 수행해야 한다. 그러므로, HSM을 기능을 포함한 고성능 프로세서가 적용된 시스템에 적용하기 적합한 방법이다.

6. 결 론

본 논문은 자동차용 게이트웨이에 적용 가능한 세가지 라우팅 방법을 제시하였다. 메시지 라우팅의 지연시간을 분석하기 위해 단계별로 지연시간을 분리하였고 각 단계별 시간을 계산 및 실험을 통해 파악하였다. 이를 통해 본문에서 제시한 세가지 라우팅방법이 적용되기 적합한 환경을 정의해 보았다. 이와 같은 연구를 통해 게이트

웨이를 컨셉단계부터 설계하기 위한 프로세스를 정립할 수 있었다. 본 논문에서 제시된 지연시간 분석 및 코어성능 분석 프로세스를 따르면, 자동차 시스템 요구사양에 적합한 최적화된 라우팅 방법을 선택하고, 적절한 코어 제품을 선정하는 데 도움이 될 것이다.

References

1) B. Steurich, K. Scheibert, A. Freiwald and M. Klimke, “Feasibility Study for a Secure and Seamless Integration of Over the Air Software Update Capability in an Advanced Board Net Architecture,” SAE 2016-01-0056, 2016.
2) Q. Hu and F. Luo, “Review of Secure Communication Approaches for In-vehicle Network,” Int. J. Automotive Technology, Vol.19, No.5, pp.879-894, 2018.
3) J. Park, D. Kim, S. Hong, H. Lee and E. Myeong, “Case Study for Defining Security Goals and Requirements for Automotive Security Parts Using Threat Modeling,” SAE 2018-01-0014, 2018.
4) IEEE Std 1722TM. IEEE Standard for a Transport Protocol for Time Sensitive Application in Bridged Local Area Networks, 2016.
5) Q. Zou, W. Chan, K. Gui, Q. Chen, K. Scheibert, L. Heidt and E. Seow, “The Study of Secure CAN Communication for Automotive Applications,” SAE 2017-01-1658, 2017.
6) K. Kawahara, Y. Matsubara and H. Takada, “A Simulation Environment and Preliminary Evaluation for Automotive CAN-Ethernet AVB Networks,” arXiv preprint arXiv:1409.0998. 2014.
7) E. Seo and H. Kim, “Security of Self-Driving Car from the Point of View of In-Vehicle System,” Transactions of KSAE, Vol.26, No.2, pp.240-253, 2018.
8) C. Park and S. Kee, “Implement of Autonomous Driving System in the Intersection Area Equipped with Traffic Lights,” Transactions of KSAE, Vol.27, No.5, pp.379-387, 2019.
9) AUTOSAR. Specification of Crypto Service Manager, AUTOSAR CP Release 4.3.0, 2016.
10) M. Dworkin, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, NIST Special Publication 800-38B, 2005.
11) Infineon Technologies AG, https://www.infineon.com/dgdl/Infineon-Automotive-Application-Guide-2019-ABR-v01_00-EN.pdf, 2019.
12) E. Wang, W. Xu, S. Sastry, S. Liu and K. Zeng, “Hardware Module-Based Message Authentication in Intra-vehicle Networks,” ACM/ IEEE 8th International Conference on Cyber-Physical Systems (ICCPS), Pittsburgh, PA, pp.207-216, 2017.