

<https://doi.org/10.7236/IIBC.2016.16.6.311>

IIBC 2016-6-39

CAN, CAN FD, FlexRay를 이용한 자동차용 진단시스템에 관한 연구

A Study on Automotive Diagnostic System using CAN, CAN FD, FlexRay

손창관*, 오세춘*, 김의룡*, 김신령**, 김영곤***

Chang-Koan Son*, Se-Chun Oh*, Eui-Ryong Kim*, Sin-Ryeong Kim**,
Young-Gon Kim***

요약 최근의 자동차 내부에는 차량의 제어, 진단기능 및 ECU 프로그램 업그레이드 등의 다양한 목적으로 차량 내부 네트워크를 사용하고 있는데 현재 CAN과 FlexRay가 가장 대표적인 네트워크이며 차세대 네트워크로는 CAN FD와 차량용 이더넷의 사용이 적극 연구되고 있다. 본 논문에서는 진단기능 및 ECU 프로그램 업그레이드 관점에서 CAN, CAN FD, 그리고 FlexRay 네트워크 적용시의 각 네트워크별 다운로드 시간과 관련된 특성에 대해서 상호 비교 평가를 하는 것을 목적으로 하였다. 시뮬레이션 결과 CAN FD 네트워크가 현재 실질적인 사용가능 속도인 500Kbps, 2Mbps 조건에서도 가격대비 성능측면에서 다른 네트워크들을 제치고 차세대 네트워크로 가장 적합함을 판단할 수 있었다.

Abstract Recent vehicles are using car internal network for various purposes such as vehicle control, diagnostic functions, and ECU program upgrade. Currently CAN and FlexRay are the most representative networks. In the next-generation network, the use of CAN FD and car ethernet is actively studied. In this paper, we aimed to compare and evaluate the diagnostic function and the program of the ECU from the upgrade view on characteristics related to download time for each network when CAN, CAN FD, and FlexRay network are applied. As a result of the simulation, it was possible to determine that the CAN FD network is currently the most suitable for the next-generation network by suppressing other networks in terms of cost performance even under conditions of 500 Kbps and 2 Mbps which are practically usable speeds

Key Words : Automotive, CAN, CAN FD, FlexRay, diagnostics

1. 서 론

전자산업의 발달로 근래의 자동차는 수많은 ECU(Electronic Control Unit, 전자제어장치)를 사용하고 있다. 초창기의 ECU들은 단순히 자동차 내의 수많은 센

서, 스위치, 액추에이터(모터, 램프 등) 간을 연결하는 전기배선(wiring harness)들을 줄이기 위한 목적으로 사용되었으나 근래에는 엔진 제어와 변속기 제어 등의 성능 향상, 차시(chassis) 제어, ABS, TCS 등을 통한 안정성 확보는 물론 기타 바디(body) 제어 및 편의장치로의 적

*정회원, 한국산업기술대학교 컴퓨터공학과

**중신회원, 동서대학교 정보통신과

***중신회원, 한국산업기술대학교 컴퓨터공학과

접수일자: 2016년 10월 11일, 수정완료: 2016년 11월 11일

게재확정일자: 2016년 12월 9일

Received: 11 October, 2016 / Revised: 11 November, 2016

Accepted: 9 December, 2016

***Corresponding Author: ykkim@kpu.ac.kr

Dept. of Computer Engineering, Korea Polytechnic University, Korea

용이 활발히 이루어 졌다. 이러한 네트워크들은 차량 내부의 ECU들 간의 정보교환은 물론 차량 외부와 연결되어져 차량 진단용 정보의 교환 및 ECU의 업그레이드를 위한 프로그램 데이터의 다운로드 등의 기능도 지원하고 있다.

이렇게 다양한 기능들을 처리하기 위해서 많은 ECU들이 차량에 적용되면서 이들 ECU들은 그 특성에 따라 다양한 종류의 차량용 네트워크(IVN : In-Vehicle Network)로 연결되어져 있다. CAN(Controller Area Network)[1][2]의 경우가 가장 보편적으로 많이 사용되는 IVN이지만 최대 1Mbps의 낮은 버스속도, 한번에 8바이트밖에 전송할 수 없는 프레임 포맷 및 전송시간을 예측할 수 없는 ID 기반 우선순위를 사용하는 이벤트 트리거(event-trigger) 방식이기 때문에 근래에는 최고 10Mbps의 높은 버스속도를 지원하며 TDMA(Time Division Multiple Access) 방식을 이용해 전송시간을 보장하는 FlexRay[3][4]가 도입되어 광범위하게 적용되고 있다. FlexRay는 많은 장점에도 불구하고 실제 차량 적용 시의 높은 비용 때문에 최근에는 차량용 이더넷을 이용해 FlexRay를 대체하려는 연구[5][6]가 적극적으로 진행되고 있다. 이더넷의 경우 point-to-point 방식의 연결이기 때문에 많은 ECU들이 한꺼번에 하나의 스위치로 연결되는 부담을 줄이기 위해 그림 1처럼 이더넷을 기본 백본(backbone) 네트워크로 사용을 하며 별도의 서브네트워크들을 구성하고 이를 게이트웨이(gateway)를 통해서 이더넷과 연결하는 연구[5][6]들이 진행되고 있으며 서브네트워크를 위한 IVN으로는 CAN의 특성을 개선한 CAN FD(CAN Flexible Data-Rate)[7]가 적극 검토되고 있다. 또한 기존의 CAN을 이용한 차량 진단 시스템의 확장을 위해 최근에는 이더넷을 이용한 진단(DoIP : Diagnostic Over Internet Protocol)과 관련해서 ISO 13400 표준이 제정되었다[8].

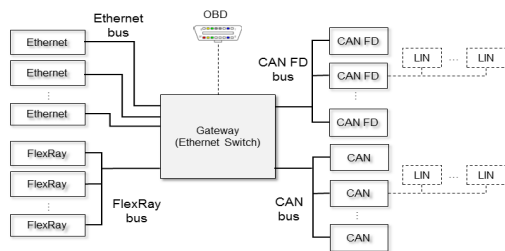


그림 1. 이더넷 백본 시스템
Fig. 1. Ethernet backbone system

본 논문에서는 차량용 이더넷 백본 네트워크의 서브네트워크로 사용될 수 있는 CAN과 FlexRay 그리고 새롭게 제안된 CAN FD에 대해서 진단 시스템 및 ECU 프로그램 다운로드 기능을 중심으로 데이터 전송 시의 버스 효율에 대해서 비교하고 CAN FD의 적용 가능성에 대해 판단하는 것을 목적으로 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 다루고 있는 CAN, CAN FD 및 FlexRay에 대해서 간략히 설명을 실시하였고 3장에서는 진단 시스템 및 대용량 데이터 전송과 관련하여 각 네트워크 별 데이터 통신의 효율성을 시뮬레이션하기 위한 조건들을 설정하였고 4장에서는 시뮬레이션 결과를 통해 각 네트워크별 버스 효율을 비교하고 평가하였다.

II. CAN, FlexRay, CAN FD

1. CAN(Controller Area Network)

CAN은 1993년 ISO 표준으로 제정된 자동차용 표준네트워크로 시간과 관계없이 전송할 메시지가 발생할 때만 해당 메시지를 버스로 출력하는 이벤트 트리거 기반의 통신 프로토콜이다. 최대 1Mbps까지의 전송속도를 지원하며 여러 개의 노드(ECU)들이 한 쌍의 트위스트 페어 케이블을 사용하여 멀티드롭 형태로 버스가 구성된다. 각 노드들은 복수개의 송신 메시지들을 가질 수 있는데 각각의 메시지들은 고유 ID를 갖게 되며 ID의 우선순위에 따라 버스 사용 우선권을 갖는 CSMA/CA(Carrier Sense Multiple Access with Collision Avoidance) 방식으로 버스를 관리한다.

CAN 규격에는 ID가 11비트인 표준 데이터 프레임 포맷(그림 2)과 ID가 29비트인 확장 데이터 프레임 포맷(그림 3)이 있다[2].

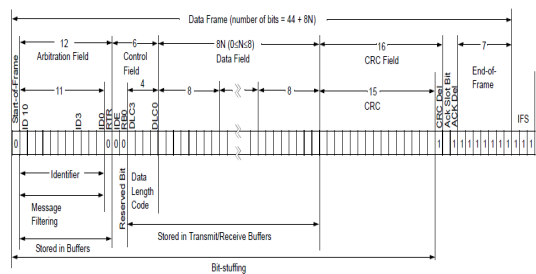


그림 2. CAN의 표준 데이터 프레임 포맷
Fig. 2. CAN standard data frame format

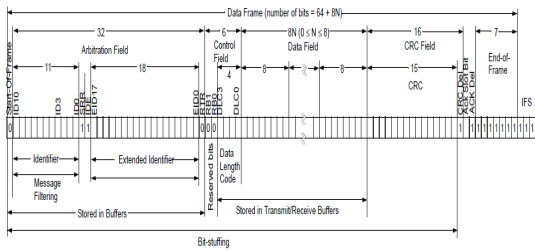


그림 3. CAN의 확장 데이터 프레임 포맷
 Fig. 3. CAN extended data frame format

모든 CAN 메시지는 프레임 단위로 구분되며 프레임은 여러 개의 필드로 구분되고 DLC(Data Length Code) 필드에 해당 프레임에서 전송할 데이터 바이트 수를 지정하여 최대 8바이트까지의 데이터를 가변적으로 전송할 수 있다.

CAN 노드에서의 수신은 임의의 노드로부터 전송이 시작되면 모든 노드의 CAN 컨트롤러들은 메시지 ID를 기반으로 설정되는 acceptance filter를 이용해서 수신여부를 판단하여 수신이 허용된 ID의 메시지만 수신한다. 비록 CAN은 낮은 전송속도와 이벤트 트리거 기반의 전송방식에서 발생하는 구조적인 문제점을 가지고 있으나 근래에는 CAN 컨트롤러를 내장하는 다양한 마이크로컨트롤러(MCU)[2]들이 많은 반도체 업체에서 생산되고 있기 때문에 낮은 가격과 부품선택의 폭이 넓어져서 현재는 자동차뿐 아니라 다양한 산업분야에서 폭넓게 사용되어지고 있다.

2. FlexRay

CAN은 기존의 자동차 제어부문에 폭넓게 사용되고 있었으나 brake-by-wire, steer-by-wire 등의 x-by-wire 제어 시스템의 등장은 기존의 제어 방식보다 훨씬 더 빠르고 정확한 시간에 제어가 되어야하기 때문에 CAN을 더 이상 이러한 분야에 적용하기에는 불가능하게 되었다. x-by-wire 등의 고속 제어에 적합한 통신 프로토콜을 개발하기 위해 2000년에 주요 완성차 업체들, 자동차 부품 업체들이 컨소시엄을 구성해 FlexRay에 대한 본격적인 개발을 시작했으며 2009년도에 FlexRay 규격은 ISO로 표준화 되었다. FlexRay는 전송시간의 예측성을 확보하기 위해 TDMA 기반의 통신을 하며 채널당 최대 10Mbps의 전송속도를 지원하고 필요시 두 개의 채널로 동일한 데이터를 보내 한 채널에서 문제가 발생해도 또 다른 채널로 올바른 데이터를 확보할 수 있는 fault

tolerance 특성을 갖는 것이 가장 큰 특징이다.

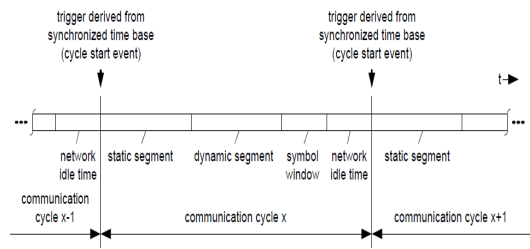


그림 4. FlexRay 통신 사이클
 Fig. 4. Basic communication cycle of FlexRay

그림 4는 FlexRay의 사이클에 대한 구성이며 정적 세그먼트(Static Segment), 동적 세그먼트(Dynamic Segment), 심볼 윈도우(Symbol Window), 네트워크 유희 시간(Network Idle Time)으로 구성된다.

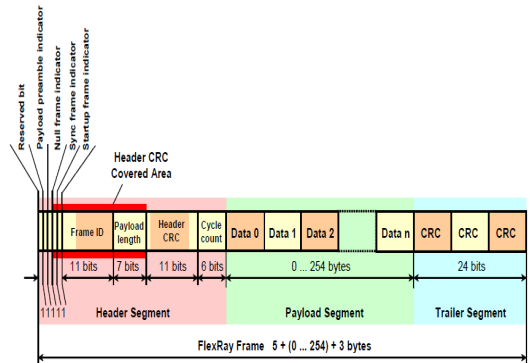


그림 5. FlexRay의 프레임 포맷
 Fig. 5. FlexRay frame format

그림 5는 정적 세그먼트와 동적 세그먼트에서 공통으로 사용하는 프레임 포맷이며 하나의 프레임에 최대 254 바이트까지의 데이터를 전송할 수 있다.

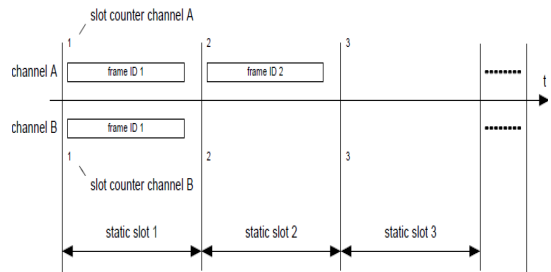


그림 6. FlexRay 정적 세그먼트의 구조
 Fig. 6. Structure of the FlexRay static segment

정적 세그먼트 구간(그림 6)은 동일한 시간 길이를 갖는 슬롯(slot)들로 구성되고 각 슬롯 구간마다 정해진 노드만이 출력 데이터의 유무와 관계없이 무조건 버스의 사용 권리를 갖기 때문에 전송시간의 예측이 가능한 구간이다. 특정 슬롯의 할당시간이 완료되면 슬롯번호는 1이 증가되어 다음번 슬롯번호를 할당받은 노드에게 버스의 사용권을 넘겨준다.

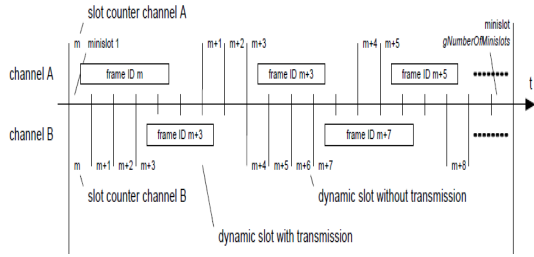


그림 7. FlexRay 동적 세그먼트의 구조
Fig. 7. Structure of the FlexRay dynamic segment

동적 세그먼트 구간(그림 7)은 짧은 시간을 갖는 미니 슬롯(mini-slot)들로 구성이 된다. 동적 구간에서도 슬롯번호를 기준으로 할당된 노드가 버스를 사용하는데 송신할 데이터가 없을 경우에는 매우 짧은 미니슬롯 시간이 경과한 후 자동적으로 슬롯번호가 증가되어 다음 슬롯번호에 할당된 노드가 데이터를 출력할 수 있도록 한다. 동적 세그먼트에서는 출력이 있을 경우에만 출력을 진행하고 출력을 할 경우에도 정적 세그먼트와는 다르게 payload length 필드에서 설정한 길이만큼의 데이터만을 전송하고 버스를 반납하는 구조로 되어있기 때문에 동적 세그먼트에서의 프레임 길이는 가변적이 되며 이러한 이유로 동적 세그먼트 구간을 이벤트 트리거 기반 FTDMA(Flexible TDMA)라고 한다.

심볼 윈도우는 네트워크의 유지와 시작 등을 위해, 네트워크 유희 시간은 노드간의 클럭을 동기화 용도로 사용된다. 사이클 시간 및 노드의 할당 등은 모두 FlexRay 시스템 설계 시에 고정된 값으로 결정된다.

FlexRay가 최초로 상용화된 차량은 2007 BMW X5 SAV 모델[9]로서 전자식 Damper 제어 시스템에 사용되었으며 이 시스템은 5개의 노드를 스타 구조로 연결해 10Mbps 버스 속도, 정적 세그먼트 3ms, 동적 세그먼트 2ms, 최대 16바이트의 정적 세그먼트 페이로드로 고정되어 있으며 정적 세그먼트는 2.5ms, 5ms, 10ms, 20ms, 40ms 주기의 task용 데이터 통신을 위해 사용되었고 동

적 세그먼트는 이벤트성 메시지, 진단기능 및 ECU 프로그램의 다운로드 등의 비주기적인 데이터 통신에 사용된다.

이렇듯 FlexRay는 x-by-wire 시스템에 잘 적응되는 구조를 가지고 있어서 많은 장점이 있으나 CAN과 대비해 높은 가격 때문에 많은 자동차 업체들은 전면적인 적용보다는 일반 제어에는 비교적 저가인 CAN을 사용하고 x-by-wire 제어에는 FlexRay를 적용하여 두 네트워크를 게이트웨이로 연결하여 사용하고 있다.

3. CAN FD

CAN FD는 2013년 보쉬사에서 제안하여 현재 ISO로 표준화가 된 규격으로 기존의 CAN을 기반으로 설계가 된 IVN이다. 기존 CAN과의 큰 차이점은 한 프레임 내에서 전송할 수 있는 데이터의 최대 개수를 기존 8바이트에서 최대 64바이트까지로 확대하고, 늘어난 데이터 전송 시간을 줄이기 위해 실질적인 데이터의 전송구간(data field)은 버스 속도를 최대 8Mbps로 높여서 프레임의 전송시간을 최소화하는 개념이다. 그림 8은 CAN FD의 기본 데이터 프레임 포맷(11비트 ID), 그림 9는 확장 데이터 프레임 포맷(29비트 ID)이다[10].

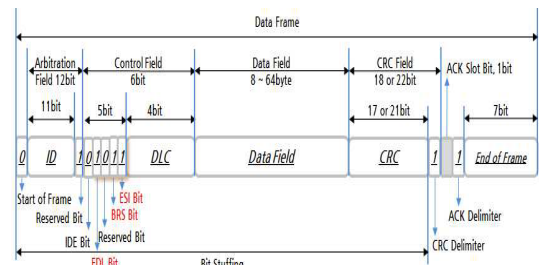


그림 8. CAN FD의 표준 데이터 프레임 포맷
Fig. 8. CAN FD standard data frame format

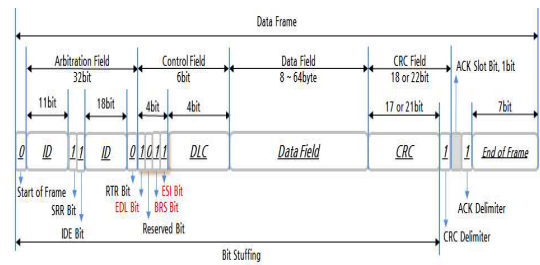


그림 9. CAN FD의 확장 데이터 프레임 포맷
Fig. 9. CAN FD extended data frame format

CAN FD의 경우는 기존 CAN을 기반으로 개선된 IVN이므로 기존의 CAN과 함께 연결해 사용할 수 있다는 큰 장점이 있다. 따라서 향후 적극적인 활용이 예상되는 차량용 이더넷 백본 시스템의 서브네트워크로 사용될 가능성이 매우 높다.

III. 3장 시뮬레이션 조건

1. 프레임 길이

시뮬레이션을 실시하기 위해 각 네트워크별로 데이터 크기에 따라 달라지는 프레임의 길이를 정리하면 수식 1~5와 같은데 CAN과 CAN FD는 bit stuffing 개념을 사용하여 신호의 안정성을 확보하고 있으나 추가되는 stuffing bit의 개수는 프레임 데이터의 내용에 따라 많은 편차가 있을 수 있고 상대적으로 비트수가 크지 않기 때문에 본 논문에서 실시하는 시뮬레이션에서는 이를 무시하였다.

수식 1은 CAN 표준 프레임의 길이(비트)이며 수식 2는 CAN 확장 프레임의 길이(비트)이다. CAN은 하나의 프레임에 최대 8바이트의 데이터를 전송하기 때문에 8바이트가 초과하는 부분들은 새로운 프레임으로 나누어져 전송되어야 하기 때문에 대용량 데이터 전송에 있어서는 매우 불리하다.

$$\text{프레임 길이} = 47\text{bit} + (8 \times \text{DLC}) + \frac{34 + (8 \times \text{DLC})}{5} \quad (1)$$

$$\text{프레임 길이} = 67\text{bit} + (8 \times \text{DLC}) + \frac{54 + (8 \times \text{DLC})}{5} \quad (2)$$

FlexRay의 정적 세그먼트는 고정길이이며 본 논문에서 사용하는 동적 세그먼트에서의 프레임 길이는 최대 254바이트의 payload(데이터)를 포함해서 수식 3과 같이 표현된다.

$$\text{프레임 길이} = 64\text{bit} + (8\text{bit} \times \text{payload length}) \quad (3)$$

CAN FD는 한 프레임에서 최대 64바이트까지 전송이 가능하기 때문에 데이터 프레임의 길이는 stuffing bit를 무시하면 수식 4(11비트 ID)와 수식 5(29비트 ID)로 표현할 수 있다.

$$\text{프레임 길이} = 29\text{bit} + (23 \text{ or } 27\text{bit} + \text{data bit}) \quad (4)$$

$$\text{프레임 길이} = 48\text{bit} + (23 \text{ or } 27\text{bit} + \text{data bit}) \quad (5)$$

수식 4, 5에서 괄호가 없는 부분은 기본 버스속도인 1Mbps가 적용되는 비트들이며 괄호 부분은 확장 버스속도인 8Mbps가 적용되는 구간이고 괄호안의 숫자 23은 전송 데이터가 16바이트 이하, 27은 16바이트를 초과할 때 적용되는 CRC(Cyclical Redundancy Check)용 비트 수이다. 비록 CAN FD의 최대속도가 기본속도 1Mbps, 확장속도 8Mbps로 정의되어 있지만 상용화된 IC의 특성상 현재는 500Kbps, 2Mbps의 속도로 사용하고 있다.

2. 시뮬레이션 조건

본 논문에서는 그림 1의 구성처럼 이더넷 기반의 백본 시스템에서 DoIP의 최대 전송 데이터 크기인 1,500바이트와 ECU 프로그램 업데이트를 위한 128K바이트의 프로그램 데이터를 각 네트워크로 다운로드하는데 필요한 각 네트워크의 버스 점유시간 시뮬레이션을 실시하였다.

표 1에서 보이듯이 CAN과 CAN FD의 경우 차량 기본 제어용 데이터와 버스를 경합하면서 사용되므로 버스 효율을 50%로 산정하였다. CAN의 최고속도는 1Mbps이나 실제 현장에서는 안정적인 사용을 위해 최고 500Kbps의 사용을 권장하고 있으므로 500Kbps 버스속도를 적용하였고 CAN FD의 경우는 기본 버스속도 1Mbps, 확장 버스속도 8Mbps 이외에 추가적으로 현재 사용이 가능한 최대 버스속도인 기본 버스속도 500Kbps, 확장 버스속도 2Mbps를 사용한 시뮬레이션도 포함하였다. FlexRay는 10Mbps 속도에 정적세그먼트 3ms, 동적 세그먼트 2ms로 설정하였고 대부분의 제어 데이터가 정적 세그먼트에서 처리되므로 동적 세그먼트의 버스 점유율을 80%로 할당하였다.

표 1. 시뮬레이션 조건

Table 1. Simulation condition

조건	세부 항목		
	IVN 종류	버스 속도 (Mbps)	버스 점유율
조건 1	CAN 표준 프레임	0.5	50%
조건 2	CAN 확장 프레임	0.5	50%
조건 3	CAN FD (11bit ID)	0.5 / 2	50%
조건 4	CAN FD (29bit ID)	0.5 / 2	50%
조건 5	CAN FD (11bit ID)	1 / 8	50%
조건 6	CAN FD (29bit ID)	1 / 8	50%
조건 7	FlexRay	10	80%

IV. 시뮬레이션 결과

그림 10은 1,500바이트의 데이터를 처리하기 위한 CAN, CAN FD, FlexRay의 버스 점유시간을 표 1의 조건으로 시뮬레이션을 실시한 결과이며 표 2는 1,500바이트 전송 조건과 128K바이트의 ECU 소프트웨어를 다운로드할 경우의 시뮬레이션 결과를 요약한 표이다.

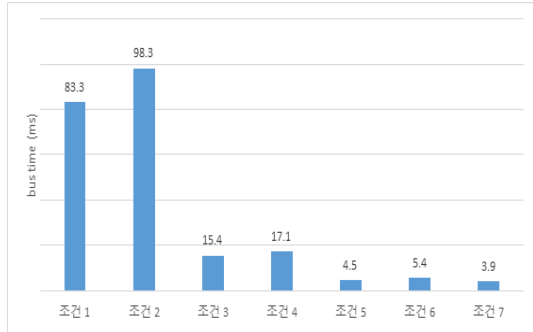


그림 10. 1,500byte 처리 시의 버스 소요시간
Fig. 10. Bus time for 1,500bytes transmission

표 2를 보면 CAN(조건 1, 2)의 경우 버스속도도 낮지만 한 프레임에서 최대 8바이트만 전송할 수 있으므로 8바이트 이상일 경우 복수개의 8바이트 단위의 메시지로 구분하여 전송하기 때문에 다른 IVN보다 매우 높은 버스 점유시간이 필요하게 되어 매우 비효율적임을 알 수 있었다.

표 2. 시뮬레이션 결과
Table 2. Simulation result

조건	버스 점유시간		
	1,500byte (ms)	128Kbyte (sec)	상대비율
조건 1	83.3	7.27	21.5
조건 2	98.3	8.59	25.4
조건 3	15.4	1.34	4.0
조건 4	17.1	1.50	4.4
조건 5	4.5	0.39	1.2
조건 6	5.4	0.47	1.4
조건 7	3.9	0.34	1.0

CAN FD의 경우 500Kbps의 기본 버스속도와 2Mbps의 확장 버스속도를 사용할 경우(조건 3, 4)에는 FlexRay와 대비하여 약 4~4.4배 정도의 버스 점유시간의 차이가

있으나 만약 CAN FD의 이론적인 최대 버스속도인 1M/8Mbps가 구현될 경우(조건 5, 6)에는 FlexRay 기준 약 1.2~1.4배로 그 격차가 매우 줄어들게 되어 FlexRay와의 가격대비 성능을 고려할 때 CAN FD의 사용이 훨씬 유리함을 알 수 있었다.

또한 CAN FD는 500Kbps의 기본 버스속도와 2Mbps의 확장 버스속도를 사용할 경우(조건 3, 4)에 CAN과 대비해 약 5배 이상의 성능향상을 가져오며 이론적인 최대 버스속도인 1M/8Mbps가 구현될 경우(조건 5, 6)에는 CAN 대비 약 18배의 향상이 있기 때문에 FlexRay와는 다르게 비교적 적은 비용으로 기존 CAN을 대체할 충분한 성능을 가지고 있음을 알 수 있었다.

V. 결론

최근의 차량용 IVN 연구에 있어서 가장 많이 검토가 되고 있는 것이 TDMA 기반의 이더넷이다. 이는 기존의 다양한 산업분야에서 검증된 기술인 이더넷을 이용해 현재는 100Mbps, 향후에는 1Gbps의 버스속도로 크기가 작은 데이터를 사용하는 제어용 네트워크는 물론 ADAS(Advanced Driver Assistance Systems) 및 멀티미디어 신호 전송을 담당하는 인포테인먼트(infotainment) 네트워크를 통합하는 차량용 백본 네트워크로의 활용성이 높기 때문이다. 이더넷 백본 시스템에서는 중앙 집중 처리 및 point-to-point 연결방식의 문제점 등을 개선하기 위한 서브네트워크의 사용이 필수적인데 CAN, CAN FD, FlexRay 등이 검토되고 있다.

본 논문에서의 시뮬레이션 결과에 의하면 CAN FD는 최고 버스속도인 1Mbps/8Mbps로 구현되지 않고 현재 구현이 가능한 0.5M/2Mbps의 속도으로도 기존의 CAN과 함께 버스를 구성할 수 있는 장점과 FlexRay 대비 상당히 저렴한 가격으로 인해 향후 차세대 서브네트워크로 각광을 받을 것으로 판단되었다.

References

[1] Robert Bosch GmbH."CAN Specification Version 2.0," 1991.
[2] Microchip, MCP2515 data sheet, 2007, from

<http://ww1.microchip.com/downloads/en/DeviceDoc/21801e.pdf>

- [3] FlexRay Consortium, "FlexRay Communications System Protocol Specification Version 3.0.1," Oct. 2010, from <https://svn.ipd.kit.edu/nlrp/public/FlexRay/FlexRay%E2%84%A2%20Protocol%20Specification%20Version%203.0.1.pdf>
- [4] JeongHoon Park, Chan-Woo Moon, "Design and Implementation of a FlexRay-CAN gateway for Real-Time Control," The Journal of the Institute of Internet, Broadcasting and Communication, Vol.14 No.2, pp.53-58, 2014.
- [5] Jin Ho Kim, Suk-Hyun Seo, Nguyen-Tien Hai, "A Gateway Framework for In-Vehicle Networks Based on CAN, FlexRay, and Ethernet," IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, vol. 64, no. 10, pp.4472-4486, Oct. 2015.
- [6] Charles M. Kozierok, Colt Correa, Robert B. Boatright, Jeffrey Quesnelle, "Automotive Ethernet: The Definitive Guide," Intrepid Control Systems, 2014.
- [7] Robert Bosch GmbH, "CAN with Flexible Data-Rate Specification Version 1.0," 2012.
- [8] Young Seo Lee, Jin Ho Kim, Jae Wook Jeon, "Automotive Diagnostic Gateway using Diagnostic over Internet Protocol," IEIE Transactions on Smart Processing & Computing 3(5), pp.313-318, 2014.10.
- [9] Anton Schedl, "Goals and Architecture of FlexRay at BMW," Vector FlexRay Symposium Stuttgart, Mar. 2007.
- [10] Jeongmin Yang, Youngsu Kwon, Kyoung-Sun Shin, Jin-Ho Han, "Implementation and verification of CAN 2.0 A, B, FD Integrated IP Based on Fault-Tolerant Vehicle Processor," The Institute of Electronics Engineers of KOREA, pp.1832-1835, 2015.6.
- [11] JeongHoon Park, Chan-Woo Moon, "Design and Implementation of a FlexRay-CAN gateway for Real-Time Control," The Journal of The Institute

of Internet, Broadcasting and Communication(IIBC), Vol. 14, No. 2, pp.53-58, Apr. 30, 2014.

- [12] Seong-Mi Park, Sang-Hyeok Lee, Sung-Jun Park, Bae-Ho Lee, "A Study of Parallel Operation of Module Power using CAN Communication", Journal of the Korea Academia-Industrial cooperation Society(JKAIS), Vol. 12, No. 8 pp. 3603-3609, 2011.

저자 소개

손 창 관(정회원)



- 2012년 2월 : 한국산업기술대학교 컴퓨터공학과(공학사)
 - 2014년 2월 : 한국산업기술대학교 컴퓨터융합학과(공학석사)
 - 2014년 3월 ~ : 한국산업기술대학교 컴퓨터공학과 박사수료
- <주관심분야: 소프트웨어공학, 정보통신시스템, 객체지향 분석 및 설계>

오 세 춘(정회원)



- 1984년 2월 : 고려대학교 전자공학과(공학사)
 - 2015년 3월 ~ : 한국산업기술대학교 컴퓨터공학과(석사과정)
 - 1999년 9월 ~ 2010년 1월 : 삼성전자 LCD 총괄 상무이사
- <주관심분야: 소프트웨어공학, 정보통신시스템, 객체지향 분석 및 설계>

김 의 룡(정회원)



- 2010년 2월 : 한국산업기술대학교 컴퓨터공학과(공학사)
 - 2012년 2월: 한국산업기술대학교 IT융합학과(공학석사)
 - 2016년 2월 : 한국산업기술대학교 컴퓨터공학과(공학박사)
- <주관심분야: 소프트웨어공학, 정보통신시스템, 객체지향 분석 및 설계>

김 신 령(중신회원)



- 1983년 2월 : 경북대학교 전자공학과 (공학사)
- 1985년 2월 : 연세대학교 본대학원 전자공학과(공학석사)
- 1990년 2월 : 연세대학교 본대학원 전자공학과(공학박사)
- 1992년 2월 ~ : 동서울대학교 정보통신과 부교수

<주관심분야 : 정보통신시스템, 부호화 방식>

김 영 곤(중신회원)



- 1983년 2월 : 경북대학교 전자공학과 (공학사)
- 1985년 2월 : 연세대학교 본대학원 전자공학과(공학석사)
- 2000년 2월 : 한국과학기술원 전산학과(공학박사)
- 1985년 ~ 2007년 : KT 수석연구원

- 2007년 ~ : 한국산업기술대학교 컴퓨터공학과 교수

<주관심분야 : 소프트웨어공학, 정보통신시스템, 객체지향 분석 및 설계>