

차량 네트워크에서의 지연 시간 절감 방법 제안

김민규¹, 김종훈¹, 도영수¹, 전재욱¹

¹성균관대학교 정보통신대학

alsrb3421@skku.edu, jhkim2022@g.skku.edu, cok2529@g.skku.edu,

jwjeon@yurim.skku.ac.kr

Suggestion of a method for reducing latency in the vehicle network

Minkyu Kim¹, Jonghun Kim¹, Young-Soo Do¹, Jae-Wook Jeon¹

¹Dept. of Electronic and Electrical Engineering, Sungkyunkwan University

요 약

본 논문에서는 차량 ECU(Electronic Control Unit)에서 제한된 메모리 공간의 효율적인 사용과 추가적인 프로세싱 시간(메모리 할당과 해제)을 감소시키는 방안을 제안한다. 메모리 할당을 하고 사용이 된 이후 바로 해제하는 기존의 방식 대신, 버퍼에 넣어두고 이후에 메모리 할당하는 상황이 생기면 버퍼에 넣어둔 할당된 공간을 재사용하는 방법을 사용하여 네트워크 전환 과정에서의 데이터 처리시간 차이를 비교 및 분석하고자 한다.

1. 서론

기술이 발전됨에 따라 차량의 기능이 다양해지고, 이로 인해 처리해야 하는 데이터가 많아지면서 최근 차량에는 100개 이상의 전자제어장치(ECU, Electronic Control Unit)가 포함되어 있다. 이들 간의 원활한 데이터 통신을 위하여 차량용 내부 네트워크(IVN, In-Vehicle-Network)에서는 ECU들을 연결하기 위해 CAN, CAN-FD, FlexRay, Ethernet, LIN 등의 통신 프로토콜이 사용된다. 각각의 네트워크방식마다 사용되는 영역이 다르기에 차량 업체는 고성능 ECU를 활용한 중앙 게이트웨이 기반 구조를 채택하여 서로 다른 통신 프로토콜을 연결해주고 변환하였다.[1] 하지만, 하나의 게이트웨이에서 여러 가지의 내부 차량 통신을 제어 및 데이터 처리해야 하기에 부하가 게이트웨이에 집중될 수 있다.

이와 같은 상황에서 현재 사용되는 도메인 기반 네트워크 구조에서는 이전의 중앙 게이트웨이 기반 구조와 비교해 서로 다른 도메인 간의 융복합 기능 개발을 쉽게 하도록 DCU(Domain Control Unit)라는 개념이 생겼다.[2] 또한, 이전에는 단순 운송수단으로만 차량을 대했지만, 시대가 변해 차량 내 멀티미디어 정보가 많아지면서 처리해야 할 데이터의 수가 많아졌다.[3] 이 경우, 차량 내부에서 어떤 네트워크를 다른 네트워크방식으로 전환하는 과정이 진행되는 도중에 지연이 생기면 인명사고가 발생할 수

있다. 게다가, 차량 내에서 사용되는 임베디드 시스템의 메모리 공간 크기 또한 한정되어 있기에 메모리 동적 할당을 무제한으로 할 수 없으며, 메모리를 할당하고 해제하는 과정에서 추가적인 지연 시간이 발생할 수 있다.

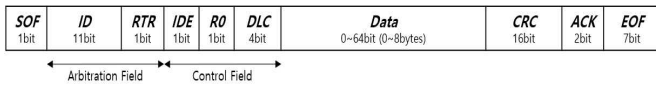
위와 같은 문제점들을 해결하기 위해 제한된 메모리 공간의 효율적인 사용과 추가적인 프로세싱 시간(메모리 할당과 해제)을 감소시키는 방안을 제안하고자 한다. 본 논문에서는 차량에서 사용되는 알고리즘인 FIFO(First-In-First-Out)와 SP(Static-Priority)에 제안하는 방법을 적용하여 네트워크 전환 과정에서의 데이터 처리시간 차이를 비교 및 분석하고자 한다.

2. 배경

2.1 CAN Protocol

차량용 통신 방식 중 하나인 CAN은 1986년 독일계 회사 Bosch에서 개발되었으며 좋은 안정성과 확장성으로 인해 차량의 특성을 잘 반영하였기에 현재까지 약 40년간 사용되고 있다. CAN protocol에서는 구리선 두 줄을 이용한 Unshielded Twisted Cable을 사용하기에 가볍고 가격이 저렴하지만 잡음에 강한 특성을 가진다. 또한, 구리선 양쪽 끝에 말단저항을 직렬로 연결하고 커패시터를 통한 Low pass filter 역할과 동시에 반사효과를 방지할 수 있

다. 최대 1Mbps의 속도로 전송할 수 있으며, 한 번에 보낼 수 있는 데이터의 양은 최대 8 Byte이다.



(그림 1) CAN Frame

2.2 Ethernet Protocol

최근, 차량에는 사용자의 편의를 위한 ADAS (Advanced Driver Assistance System)와 같은 기술들이 도입되면서 차량 내부에서 처리해야 할 데이터 양과 통신 대역폭이 증가하고 있기에 CAN 네트워크만으로는 부하를 감당할 수 없게 되었다. 따라서 자동차 제조업체는 충분한 대역폭을 제공하고 높은 대역폭의 데이터 이동을 처리하기 위해 차량용 Ethernet을 도입하고 있다. Ethernet은 최대 100Mbps 이상의 속도로 전송할 수 있으며, 한 번에 보낼 수 있는 데이터의 양은 최대 1500 Byte이다.



(그림 2) Ethernet Frame

3. 데이터 전송 방법

3.1 스케줄링 알고리즘

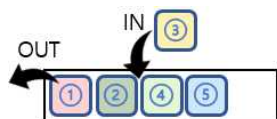
스케줄링(Scheduling)이란 처리할 프로세스들의 우선순위를 정하는 방법이다. 본 논문에서는 두 가지의 스케줄링 알고리즘을 사용하였다.

FIFO(First-In-First-Out)는 선입선출, 즉 먼저 입력된 프로세스가 가장 먼저 처리되고, 처리가 끝날 때까지 다음 프로세스는 대기 상태에 놓이게 되는 스케줄링 방식이다.



(그림 3) FIFO Algorithm

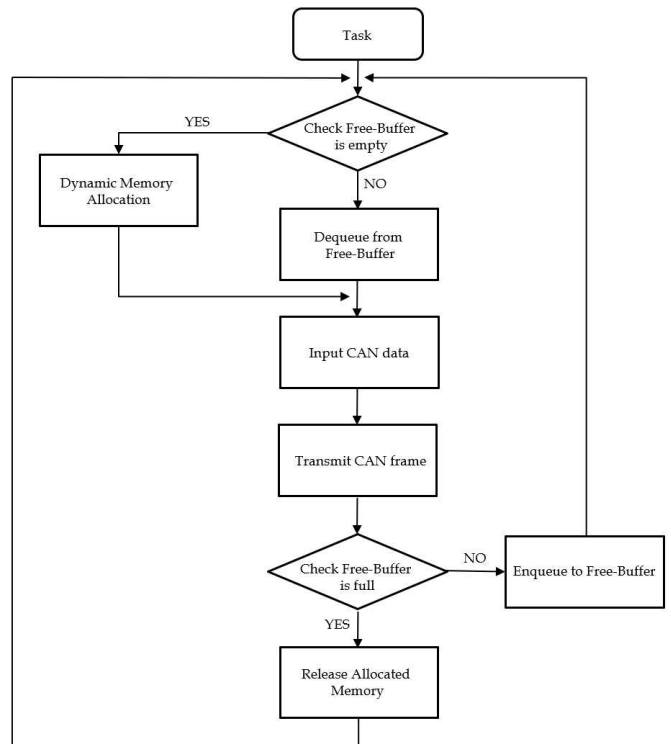
SP(Static-Priority)는 프로세스 간의 우선순위를 비교하여 우선순위가 높은(숫자가 작은) 메시지부터 전송하는 스케줄링 방식이다. 본 실험에서는 CAN frame의 ID를 SP Algorithm의 우선순위로 결정하였다.



(그림 4) SP Algorithm

3.2 제안한 방법(Free-Buffer)

사용자가 메모리를 동적 할당하면 사용자가 이를 해제하기 전에는 메모리 공간이 계속 남아있다. 또한, 프로세스가 실행될 때 프로그램이 정해진 힙 영역의 크기를 초과하면 메모리가 더는 할당되지 않는다. 하지만, 게이트웨이를 사용하여 Ethernet frame을 CAN frame으로 변경하면 CAN frame은 크기가 고정되어 정해진 최대 크기를 넘어가는 경우가 생기지 않는다. 메모리 할당 및 해제시간을 최소화하기 위해 메모리 할당을 하고 사용이 된 이후 바로 해제하는 기존의 방식 대신, 버퍼에 넣어두고 이후에 메모리 할당하는 상황이 생기면 버퍼에 넣어둔 할당된 공간을 재사용하는 방법(프리버퍼)을 제안한다.

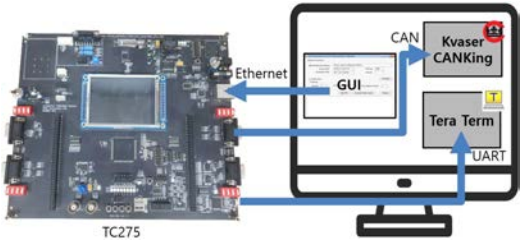


(그림 5) 제안한 방법의 알고리즘 구상도

제안한 방법(그림 5 참조)은 Task가 입력될 때 프리버퍼가 비어있는지를 판단한 뒤 프리버퍼에 잔여 공간이 없으면 새로 메모리 공간을 할당하고, 잔여 공간이 있으면 기존의 공간을 재사용하는 방식이다. CAN 메시지를 전송한 이후에 프리버퍼가 가득 차 있지 않으면 메모리 할당을 해제하지 않고 임시로 프리버퍼에 저장한다. CAN 메시지를 Linked-List 방식을 이용해 정렬하기에 프리버퍼는 원형 큐 방식의 포인터 배열로 구현하였으며, CAN frame 10개 분량이 들어갈 수 있는 크기로 설정하였다.

4. 실험 환경

본 실험은 Infineon 사의 MCU가 내장된 TC275 임베디드 보드를 사용하여 진행되었다. 원하는 데이터의 Ethernet frame을 PC에서 TC275로 보내기 위해 GUI를 제작했으며, CAN frame 수신 확인을 위해 Kvaser사의 CANKing 프로그램을, 프로세스 처리시간을 확인하기 위해 UART Protocol을 이용했으며 이는 Tera Term 프로그램으로 확인하였다.



(그림 6) 실험 환경을 나타낸 그림.

5. 실험 결과

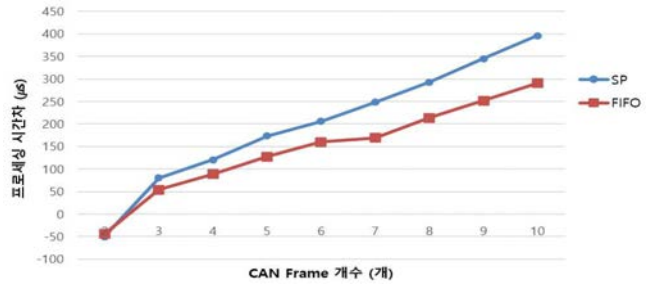
<표 1>은 10회 간의 실험을 통해 도출된 데이터의 평균값으로 작성되었다. GUI에서 Ethernet frame의 Payload(Data) 영역에 CAN frame 개수를 2개에서 10개의 CAN frame을 Packet 단위로 담아 TC275로 전송하였다.

프로세스 처리시간은 ‘TC275의 Ethernet 수신부터 Sorting 과정을 거친 후 CANKing 프로그램에서의 CAN frame 수신까지의 시간’으로 정의했다. 또한, 시간차는 버퍼 미사용 시의 처리시간에서 버퍼 사용 시의 처리시간을 뺀 값이다.

CAN Frame 개수 (개)	FIFO			SP		
	버퍼 미사용	버퍼 사용	시간차	버퍼 미사용	버퍼 사용	시간차
2	7,387	7,431	-44	7,433	7,484	-51
3	17,115	17,061	54	25,700	25,620	80
4	39,030	38,941	89	49,567	49,426	141
5	52,433	52,306	127	74,754	74,581	173
6	70,663	70,503	160	89,607	89,400	207
7	85,666	85,497	169	106,782	106,531	251
8	109,795	109,581	214	120,329	120,047	282
9	134,769	134,517	252	143,178	142,834	344
10	158,786	158,495	291	158,882	158,486	396

<표 1> 프로세스 처리시간을 나타낸 표. (단위 : μ s)

(그림 7)에서 Ethernet frame에 들어있는 CAN frame이 2개인 경우를 제외하면 FIFO와 SP Algorithm 모두에서 프리버퍼를 사용한 경우 기존 대비 전체 처리시간이 감소하였음을 확인할 수 있었다. 또한, 프리버퍼를 사용하였을 때 CAN frame 개수가 늘어날수록 전송 시간이 약 30μ s 정도 단축되었음을 확인할 수 있다.



(그림 7) 알고리즘별 프로세싱 시간차

6. 결론

본 논문에서는 차량용 게이트웨이에서의 네트워크 부하와 메모리 할당 및 해제에 소요되는 시간을 감소시키기 위해 프리버퍼라는 제안방법을 사용했다. CAN frame이 2개인 경우에는 프리버퍼의 프로세싱 시간이 새로운 메모리를 할당하는 시간보다 오래 소요되었지만, 이외에는 프리버퍼를 사용한 경우가 프로세싱 처리시간 측면에서 유의미한 영향을 끼쳤음을 알 수 있다. 이는 프리버퍼를 사용하면 Ethernet frame에 포함된 CAN frame이 증가할수록 더욱 좋은 효율을 나타낼 것으로 예상할 수 있다.

7. Acknowledgement

이 연구는 정부(교육부-산업통상자원부)의 재원으로 한국산업기술진흥원의 지원을 받아 수행된 연구임(P0022094, 2022년 미래형자동차 기술융합 혁신 인재양성사업)

참고문헌

[1] J. H. Kim, S. -H. Seo, N. -T. Hai, B. M. Cheon, Y. S. Lee and J. W. Jeon, "Gateway Framework for In-Vehicle Networks Based on CAN, FlexRay, and Ethernet," in IEEE Transactions on Vehicular Technology, vol. 64, no. 10, pp. 4472-4486, Oct. 2015

[2] Dingwang Wang, "Automotive Domain Controller", 2020 International Conference on Computing and Information Technology, University of Tabuk, 2020, Volume: 01, Issue: ICCIT- 1441, Page No.: 443-447

[3] Y. S. Do, M. Ho Kim and J. W. Jeon, "Time synchronization method between CAN-FD nodes," 2021 IEEE Region 10 Symposium (TENSYP), pp. 1-5, 2021