

CAN Protocol을 이용한 CAN 통신 시스템 설계 및 구현

이서경*, 이재용*, 김동현**, 최광주**, 정재일*

*한양대학교 전자통신컴퓨터공학부

**미디어 연구소, LG전자¹⁾

e-mail: seokyunglee@mnlab.hanyang.ac.kr, jijung@hanyang.ac.kr

CAN Communication System using CAN Protocol

Seo-Kyung Lee*, Jae-Yong Lee*, Dong-Hyun Kim**

Kwang-Joo Choi**, Jae-Il Jung*

*Dept of Electronic and Computer Engineering, Han-Yang University

**Media Lab., LG Electronics

요 약

자동차 산업은 관련 기술의 진전과 함께 비약적인 발전을 거듭하여 신뢰성 및 안정성의 확보뿐만 아니라, 운전자의 편리성과 같은 새로운 기능 구현을 위한 연구개발이 가속화되고 있다. 최근 각종 텔레매틱스 서비스를 위해 차량 내 기능의 신뢰성과 성능을 향상시키기 위한 자동 조절의 필요성이 증가하고 있으며, 각종 기능에 대한 전자통신 시스템을 사용한 제어 및 네트워크의 통합화가 급속히 이루어지고 있다. 따라서 CAN(Controller Area Network)과 같은 네트워크 개념의 도입으로 차내 전선사용의 감소뿐만 아니라 제어 및 고장의 진단을 용이하게 하고 차량 내 안전성의 개선 및 자동차 품질과 비용 절감을 기대할 수 있다. 본 논문에서는 CAN Protocol을 분석하고 차량 내 제어 데이터 전송을 위해 CAN을 통한 통신 시스템을 구현 및 검증하였다.

1. 서론

차내 네트워크 성능향상에 따라 전자통신 시스템은 복수화·비대화되지만 공간이 한정적인 자동차에서의 시스템 크기는 한계에 가까워지고 있다. 이러한 문제를 해결하는 수단 중 하나로 종래의 1 대 1 네트워크가 아니라 버스형 네트워크 시스템이 도입되었다. 버스형 네트워크로 구성되면 시스템 내의 데이터 공유가 가능해질 뿐만 아니라 배선이 단순화되어 차체 중량의 감소와 생산성을 향상시킬 수 있다[1]. 따라서 CAN(Controller Area Network) 네트워크 도입으로 차내 전선사용의 감소뿐만 아니라 제어 및 고장의 진단을 용이하게 하고 차량 내 안전성의 개선, 조립 시간의 단축 등과 같은 높은 자동차 품질을 기대할 수 있다[2][3].

본 논문에서는 CAN Protocol을 분석하고 차량

내 제어 데이터 전송을 위해 CAN을 통한 통신 시스템을 구현한다.

이하, 본 논문의 구성은 다음과 같다. 2장에서는 CAN Protocol을 분석한다. 3장에서는 CAN 통신 시스템의 구조와 시스템 설계에 대한 실행을 통하여 특정 노드에서 메시지가 송수신 됨을 검증한다. 4장에서는 결론으로 마무리한다.

2. CAN Protocol 분석

2.1 CAN 통신 프로토콜의 기본개념

CAN은 차량의 안전성, 편리성, 승차감의 개선, 배기 가스량의 삭감 및 연비의 향상 등과 같은 기능을 향상시키고, 효율적인 제어를 수행하고자 국제표준화가 이루어진 높은 레벨의 안전성과 함께, 분산 실시간 제어를 효율적으로 지원하는 직렬 통신 프로토콜의 하나이다. 이와 같은 CAN의 가장 큰 특징을 살펴보면, 제어 노드를 버스 형식으로 접속하고 있어서 어떤 노드에서라도 메시지 송신이 가능한 멀티

1) 본 연구는 LG전자 DM 사업본부 미디어 연구소의 지원에 의해 수행되었음

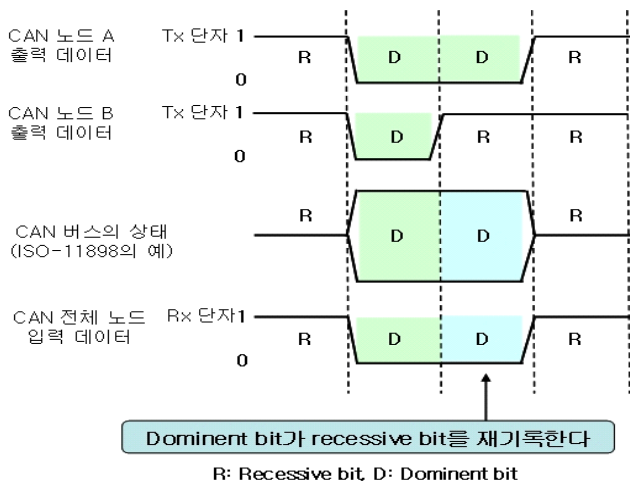
마스터 방식의 버스 구성을 가지며, 송신 메시지에 식별자를 붙여 버스 상의 전 노드에 브로드캐스팅 방식으로 메시지를 송신하여 수신측에서는 식별자를 보고 그 메시지가 자기에게 필요한 것인가를 판단한다. 또한 높은 전송 신뢰성을 확보할 수 있다. 즉, 최대 1Mbps의 데이터 전송율을 보장하면서 에러 검출과 에러 제한의 기능이 뛰어나다[2].

2.2 CAN Protocol 분석

CAN에서는 두 가지 비트 레벨이 있다. 도미넌트(Dominant) 비트와, 리세시브(Recessive) 비트가 있으며, 도미넌트 비트가 논리 "0", 리세시브 비트가 논리 "1"이다. 이 두 가지 비트 레벨을 통해서 CAN 제어 데이터가 전송된다.

2.2.1 데이터 충돌 시 비트 상태

CAN에서는 기본적으로 동시에 데이터 출력이 가능하다. 즉 데이터 충돌이 일어날 수 있다. (그림 1)에서 도미넌트 비트와 리세시브 비트의 충돌 예를 나타내었다. 다른 데이터가 동시에 출력된 경우에는 도미넌트 비트가 리세시브 비트를 덮어쓸 수 있다. 이러한 관계에 의해 메시지의 우선순위가 부여되며 멀티마스터를 실현하고 있다[4].

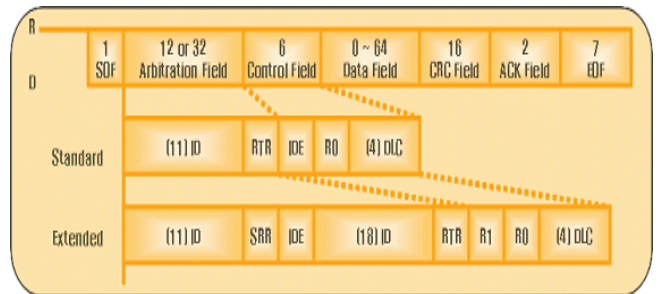


(그림 1) Dominant 비트와 Recessive 비트의 충돌

2.2.2 CAN 프로토콜 프레임의 구성

CAN은 통신을 위해 데이터 프레임, 원격 프레임, 에러 프레임, 오버로드 프레임, 인터프레임 스페이스의 5가지 형태의 프레임을 제공한다. 데이터 프레임은 데이터를 전송할 때 CAN 컨트롤러가 전송하는 프레임이며 원격 프레임은 데이터를 요구할 경우에 사용되며 데이터 프레임에서 데이터 영역만 제

외된 것이다. 에러프레임은 감지된 에러를 알리는데 사용되며 오버로드 프레임은 연속된 프레임 사이에서 지연시간을 늘리기 위해 사용된다. (그림 2)는 CAN 2.0A 표준 포맷과 CAN 2.0B 확장 포맷에 대한 메시지 프레임의 구성을 나타낸 것이다.



(그림 2) CAN 2.0A/B 메시지 프레임

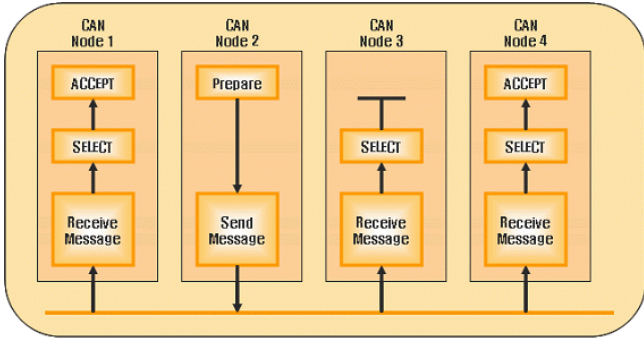
CAN 프로토콜은 2가지의 메시지 프레임 포맷을 지원한다. 즉, 11 비트의 식별자(ID)를 갖는 표준 포맷(CAN 2.0A)과 29 비트의 ID를 갖는 확장된 포맷(CAN 2.0B)이 있다. 메시지 전송을 위한 메시지 프레임은 7개의 필드로 구성된다. 표준형의 메시지 포맷에서 메시지는 SOF(Start Of Frame)로 시작하고, ID와 RTR(Remote Transmission Request)비트를 갖는 중재필드(Arbitration Field)가 그 뒤를 잇는다. 그리고 제어 필드(Control Field)에는 표준 포맷과 확장된 포맷을 식별하기 위한 IDE(Identifier Extension) 비트를 갖는다. 다음으로 0~8 bytes인 가변량의 데이터를 갖는 데이터 필드와 비트 에러 검출을 위한 CRC 필드, ACK 슬롯과 ACK delimiter 비트를 갖는 ACK 필드 등으로 구성되고, 메시지의 마지막은 EOF(End Of Frame)로 구성된다[5][6].

3. CAN 통신 시스템 구현 및 검증

3.1 CAN 통신 시스템 구조

CAN에 의한 데이터 전송은 고유번지의 지정 없이 Identifier라고 하는 식별자에 의해 엔진의 rpm이나 온도와 같은 메시지의 내용이 식별되도록 한다. 이때 식별자는 메시지의 우선순위가 그 내용을 정의한다. 임의의 제어기에서 하나 이상의 노드에 메시지를 전송하고자 할 때는 전송할 데이터와 그들의 식별자를 할당된 CAN 노드에 보내도록 한다. 각 CAN 노드에서 버스 사용이 가능하게 되면, CAN 네트워크 내의 모든 노드들은 이 메시지의 수신기가 된다.

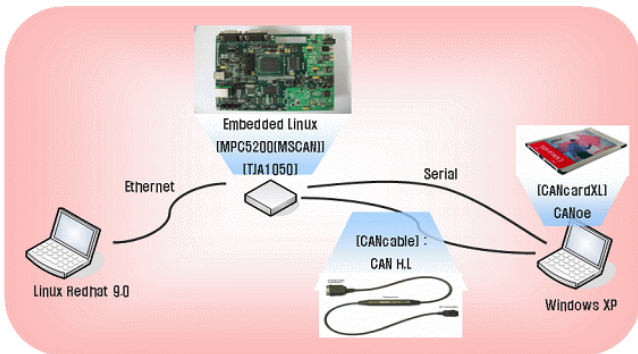
(그림 3)은 CAN 통신 시스템 구조를 나타낸 것이다. 즉, 하나의 CAN 노드에서 전송된 메시지는 그 메시지 내에 포함된 식별자를 다른 노드가 보고 필요하면 받아들이고 그렇지 않으면 수신하지 않는 필터링 과정을 거쳐 전송된다.



(그림 3) CAN 통신 시스템 구조

3.2 CAN 통신 시스템 구성도

(그림 4)는 CAN 통신 시스템 구성도이다. 통신 준비절차는 다음과 같다. Embedded Linux가 porting된 MPC5200 CAN board가 power on이 되면, Linux가 설치된 호스트에서 CAN Driver module과 통신 애플리케이션이 업로드 되어 통신 준비 단계를 끝낸다. 그 뒤, CAN 메시지 프레임 전송을 위해 호스트에서 원격으로 보드에 접속을 하고, 통신 애플리케이션을 실행시켜 각 메시지의 필드를 설정하여 전송한다.



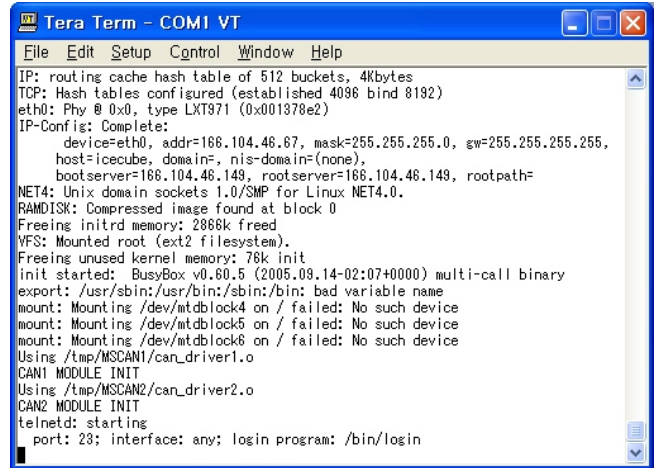
(그림 4) CAN 통신 시스템 구성도

3.3 CAN 통신 시스템 실행 및 검증

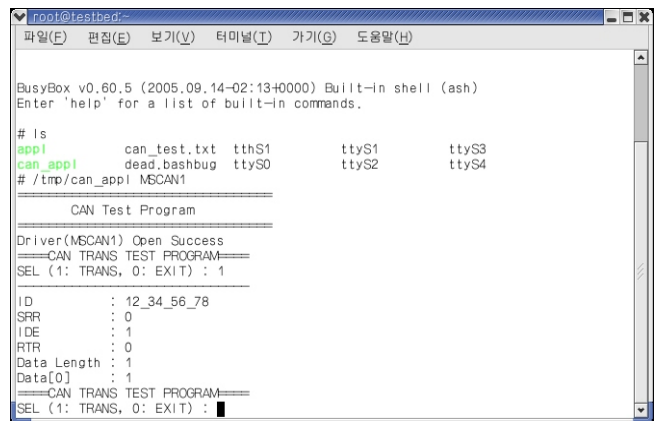
3.3.1 CAN 통신 시스템 실행

(그림 5)에서 나타내는 것과 같이 데이터 프레임 전송을 테스트하기 위하여 Linux호스트에서 MPC5200 CAN board 에 접속하여 통신 애플리케이션을 실행시킨다. 통신 애플리케이션 실행 후, 각 메

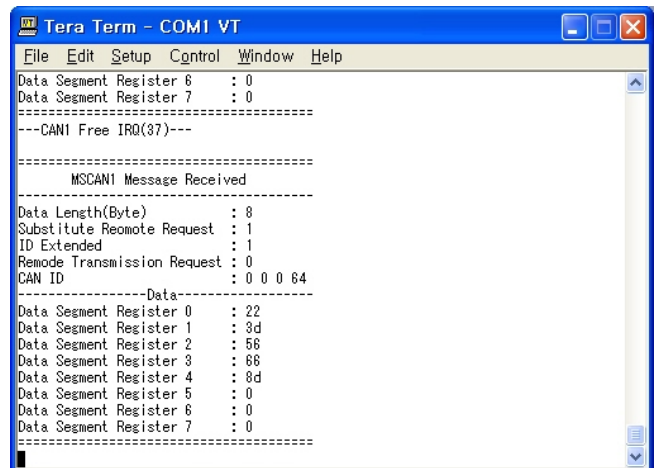
시지의 필드들을 입력하여 전송한다. 이 메시지들은 각 수신노드에 브로드캐스팅으로 전송되지만, 각 수신 노드는 필터링 작업을 거치기 때문에 자신의 원하는 특정 메시지만 수신하게 된다. 이것을 다음 (그림 6)]과 (그림 7)에서 보여준다.



(그림 5) Target Board Booting



(그림 6) 통신 애플리케이션 실행 및 데이터 전송



(그림 7) MPC5200 CAN Board 수신 상태

3.3.2 CAN 통신 시스템 검증

CAN 통신 검증 툴로는 CANoe를 사용하였다. CANoe(CAN open environment)는 CAN 버스 시스템을 분석하고 테스트하기 위한 통합개발환경으로써, 시스템 개발과 관련된 모든 사람들에게 사용 가능하다. CANoe의 시뮬레이션 기능을 이용해 구현되어 있지 않은 노드 또는 전체 시스템을 시뮬레이션할 수 있으며, 이를 사용해 개발하려는 시스템의 통신 특성을 확인하거나 보다 이상적인 테스트 환경을 구현할 수 있는 도구이다[7][8]. 따라서 실제 CAN board에서 전송된 메시지를 CANoe에서 수신하여 보여줌으로써 (그림 8)과 같이 CAN 통신 시스템을 검증한다.

Time	Chn	ID	Name	Dir	DLC	Data
0.005210	1	12345678x		Rx	8	11 22 33 04 00 55 3d 22
1.003820	1	64x		Tx	8	22 3d 56 66 8d 00 00 00

(그림 8) CANoe 송수신 상태

4. 결론

본 논문에서는 CAN Protocol을 분석하고 차량 내 제어 데이터 전송을 위해 CAN을 통한 통신 시스템을 구현 및 검증하였다. 브로드캐스팅 방식으로 전송되는 메시지들을 각 수신기에서 필터링을 하여 원하는 메시지들만을 수신하게 하였으며, 실제로 전송 및 수신을 CAN board 상에서 정확히 동작하는가를 CANoe를 통하여 검증하였다.

향후 본 논문에서 제안된 시스템을 다른 Real-Time Operating System 환경에서 개발할 것이며, CAN board에서 CAN 버스를 통한 ECU등의 전자 장비들을 위한 애플리케이션을 개발할 것이다.

참고문헌

- [1] 멀티마스터 방식의 차내 네트워크 프로토콜: 월간 전자기술 2003.08
- [2] <http://www.can.bosch.com>
- [3] CAN 통신 프로토콜을 사용한 자동차 Network

의 다중화 기법의 개발에 관한 연구, 정차근, 한국 통신학회, 2001년 하계 학술대회

- [4] Electronic Control in Electric Vehicle Based on CAN Network, Lamia CHAARI, IEEE SMC WPICS, 2002
- [5] INTERNATIONAL STANDARD ISO 11898: 1993
- [6] CAN Specification, BOSCH, Part A, 1991
- [7] CANoe/Denoe User Manual version 5.1, Vector Informatik GmbH, 2005, 5
- [8] CANcardXL and CANcabs User Guide, Vector Informatik GmbH, 2005. 5