

Controller Area Network (CAN 통신과 임베디드 시스템을 이용한 자동차 감시 시스템 구현

양승현*, 이석원
호서대학교 정보제어공학과

An Implementation of Monitoring System of Vehicle Using CAN Communication and Embedded System

Seung-Hyun Yang*, Seok-Won Lee
Department of Information & Control Engineering, Hoseo Univ

Abstract - CAN communication can minimize the interfacing lines between equipments because it is composed of only the input and output lines, also is used for automatic system including vehicle, aircraft, railway vehicles and robot because the reliability of data is high by the capability of data-related error detect and correcting function. It can also improve the low-reliable and inefficient system which is composed of the existing Wiring Harness(W/H), so in case of vehicle, it is used in place of the present ECU as the new electro-control unit. In this paper, we constructed the electro-control unit of vehicle by using CAN communication and implemented system that could monitor the condition of vehicle through the web or mobile by connecting the electro-control unit to imbedded system. Such a system is expected to be helpful to the intelligent vehicle and the adoption of ACC(Adaptive Cruise Control).

1. 서 론

자동차 산업의 비약적인 발전으로 인해 운전의 편리성이나 안전성, 자동차 내부의 자동화 등의 고급화, 고기능화에 대한 연구개발이 이루어지고 있다. 이러한 자동차의 사양들이 다양해지고 고기능화 되어지기 때문에 자동차의 집적적인 통합제어 방식보다는 분산제어 방식에 대한 많은 연구들이 진행 중이다. 차량의 전기전자 부품을 전기적으로 연결하는 Wire Harness는 차량에 대한 중요한 신호 전달 매개체로 신경과 같은 역할을 한다. 하지만 제어대상의 증가와 향상에 따라 전선의 양이 많아져 부피가 커지고 무게 또한 중량이기 때문에 차량에 효율적 배치나 구성이 어려워진다. 이러한 문제점을 해결할 수 있는 차량배선 방법 중 국제적으로 표준화가 진행되고 있는 Controller Area Network (CAN)은 대표적인 표준화 프로토콜이다. CAN 통신은 입력선과 출력선으로만 구성되어 장치간의 연결선을 최소화 할 수 있으며, 데이터 연계층의 에러 검사와 수정 기능으로 데이터 신뢰도가 높아 자동차를 비롯한 항공기, 철도, 로봇을 비롯한 자동화 시스템에 이용되고 있다. 또한 기존의 Wiring Harness(W/H)에 의한 제어기 구성의 낮은 신뢰성과 비효율적인 시스템 구성을 개선 할 수 있어 자동차의 경우 현재 집적 적이고 통합적인 제어방식을 대체하며 새로운 전자제어 장치로 사용되어 지기도 한다.

본 연구에서는 CAN 통신을 활용해 자동차의 분산 제어를 위한 전자제어장치를 구성하고, 이를 관리하고 감시할 수 있는 Intel 사의 XSCALE 기술기반의 RISC 프로세서인 PXA255 칩을 이용하여 메인 제어기를 구현하였다. 또한 메인 제어기에 RTOS(Real Time Operation System)인 임베디드 리눅스를 설치해 웹 서버를 구현하여 원격지에 존재하는 자동차의 모니터링 시스템을 구현하여 자동차 분산 제어 시스템을 인터넷을 통한 웹이나 PDA와 핸드폰과 같은 모바일 기기로 감시할 수 있

는 시스템을 구현 하였다.

2. 시스템 구성

2.1 CAN(Controller Area Network) 제어기

국제규격(ISO11898)인 ISO에 등록되어 있는 CAN은 Bosch 자동차 회사에 의해 개발되었다. 효율적인 제어를 수행하고자 국제 표준화가 이루어진, 높은 안전성과 함께 분산 실시간 제어를 효율적으로 지원하는 Serial(직렬) 통신 프로토콜이다. 이러한 방식은 정보교환을 입력선(CAN_RX)과 출력선(CAN_TX)을 직렬로 하기 때문에 전기적 결선이 많은 자동차 부품간의 통신 연결을 획기적으로 줄일 수 있다.

현재 자동차는 하나의 ECU(Electronic Control Unit)가 자동차에서 검출되는 엔진 온도나 자동차의 속도, 스로틀 각 등의 센서신호를 비롯한 각종 자동차의 전반적인 내용들을 살피고 상황에 맞는 신호를 전자적으로 출력해 자동차가 동작하는데 있어 중요한 역할을 한다. 하지만 자동차의 발전으로 인해 자동화, 지능화(Intelligent) 되기 때문에 하나의 ECU로 모든 자동차의 전자장치를 제어 하기에는 역부족이다. 그래서 1986년 자동차 업체인 벤츠의 요구로 인해 독일의 보쉬는 서로 다른 세 개의 ECU간의 통신을 위한 통신 장치 개발을 하였다.

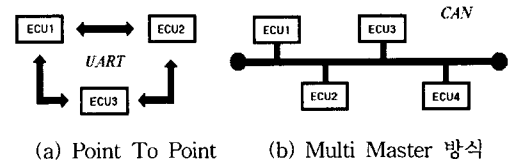


그림 1. ECU 간의 데이터 전송방식

초기에는 그림 1의 (a)과 같이 UART 방식을 고려하였으나 속도나 데이터 전송거리 문제 등이 적합하지 않아 (b)과 같이 Multi Master 방식의 CAN 통신 프로토콜이 제안되었다. CAN의 속도와 거리간의 문제는 UART에 비해 월등히 앞서며, 실시간으로 데이터를 처리하기 위해서는 고속의 데이터 전송을 필요로 하는데 CAN에는 속도는 물론 긴급한 데이터 전송을 위해 관련된 메시지의 식별자에 의해 전송의 우선순위를 정해서 송수신 되도록 하기 때문에 높은 신뢰성을 보인다.

CAN 통신 프로토콜에서 메시지는 일정한 형태의 포맷을 갖고 제한된 길이의 데이터를 가지는 버스에 실린 정보로 정의된다. 실제 CAN 데이터 전송은 메시지를 통하여 이루어지는데, 이 메시지는 데이터 프레임(Data Frame), 원격 프레임(Remote Frame), 오류 프레임(Error Frame), 마지막으로 과부하 프레임(Overload Frame)의 4가지 형태로 구분된다. 각각의 프레임은 식별자(ID)의 길이에 따라 11 비트의 식별자를 가지는 표준 포맷(CAN 2.0A)과 29비트의 ID를 갖는 확장된 포맷

(CAN 2.0B)이 이용된다. 데이터 프레임과 원격 프레임만이 표준 및 확장 형식 사용이 가능하며, 한 번에 전송할 수 있는 데이터량은 8 바이트로 한정된다.

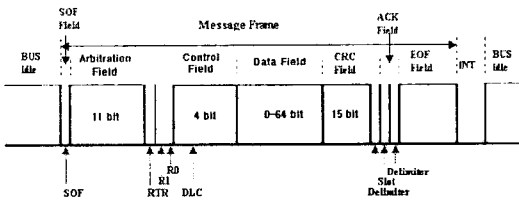


그림 2. 표준 포맷(Standard Format) 메시지 프레임

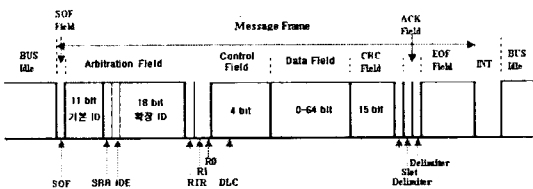


그림 3. 확장된 (Extended Format) 메시지 프레임

CAN 제어기는 16비트 마이크로프로세서로 128K Flash 메모리와 4K Byte의 RAM을 가지며 다양한 I/O, 10bit의 A/D변환기, Serial I/O를 가지며 싱글 CAN 2.0A/B가 내장된 Fujitsu사의 MB90595를 사용하였다. 또한 CAN 드라이버(Transceiver)로는 Phillip사의 110노드 연결까지 지원되는 PCA82C250를 사용하여 CAN 제어기 보드를 설계 제작하였다.

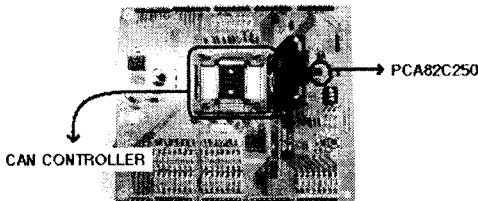


그림 4. CAN 제어기 및 드라이버 제작

2.2 임베디드(Embedded) 시스템

자동차의 ECU와 측정대상의 다양한 센서들에 분산 연결된 CAN 제어기들에서 출력되는 신호들을 입력받아 처리하고 웹(Web)으로 전송하기 위해서는 임베디드 시스템 보드에 무료 RTOS(Real Time Operating System)인 임베디드 리눅스를 설치하고, 웹 서버(Server)를 포팅해야 한다.

2.2.1 임베디드 웹서버 구현

웹 기반 프로그램이란 클라이언트의 웹 브라우저를 이용해 인터넷에 연결되어 있는 원격지의 서버로 기기를 관리한다는 의미를 가지고 있다. 이러한 웹서버를 구현하고 클라이언트로 서버를 제어하기 위해서는 세 가지 요소를 갖추어야 한다.

1) 클라이언트 풀(Client Pull)과 CGI

웹 페이지의 화면을 자동으로 반복표시(refresh)해주는 방법에는 크게 클라이언트 풀(Client Pull) 방식과 서버 푸시(Server Push) 방식이 있다. 원격지에 있는

자동차의 상황을 실시간적으로 모니터링하기 위해 클라이언트 풀 방식을 사용하여 한번 CGI로 구성된 페이지를 열게 되면 반복적으로 페이지가 변경되게 하는 방법이 유용하다. 클라이언트의 요청이 있을 때마다 서버에 새 새로운 CGI 프로세스의 생성과 종료로 유발시켜주기 때문에 서버의 동작에 부하를 주지 않는다.

2) 자바애플릿

Sun Microsystems에서 개발된 자바언어(Java Language)는 객체지향 언어로 내장형 시스템을 위해 플랫폼에 구애 받지 않는 언어로 처음 개발되었다. 자바가 플랫폼에 구애받지 않기 위해서는 동작할 플랫폼에 자바 가상머신(JVM, Java Virtual Machine)이 내장되어 있어야 한다. 하지만 임베디드 시스템에 자바 가상머신을 올리려면 많은 리소스를 필요로 하기 때문에 자바 애플릿(Java Applet)을 이용한다. 이는 클라이언트 상에만 자바 가상머신만 있으면 된다.

3) GoAhead 웹 서버

본 연구의 프로그램 구현시 사용한 웹 서버는 오픈소스인 GoAhead 웹 서버를 사용하였다. 웹 서버는 PXA 255기반의 리눅스 플랫폼상에서 실행될 수 있도록 교차 컴파일하여 사용하였다. 웹 서버를 포팅해야만 임베디드 보드가 하나의 서버가 되기 때문에 원격지에서 클라이언트를 통해 접속할 수 있다.

2.2.2 임베디드 웹서버용 보드

웹서버용 보드는 리눅스와 임베디드 웹서버를 포팅한 것으로 그림 5에 보이고 있다.

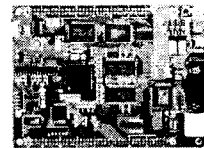


그림 5. 임베디드 웹 서버용 보드

PXA255 RISC ARM 400[Mhz]인 CPU를 사용하고 있으며, 64[Mbyte] Nand-Flash, 32[Mbyte]의 SDRAM과 부트(Boot) Flash로 512[Kbyte]의 메모리를 가지고, 이더넷 컨트롤러로 CS8900 Chip을 사용하고 있다. 또한 PCMCIA 카드를 연결할 수 있도록 확장보드를 설계하여 무선 랜카드 연결이 가능하도록 하였다. 그림 6은 임베디드 보드에 GoAhead 웹서버를 포팅하고, 외부의 클라이언트로 서버에 접속한 그림이다.

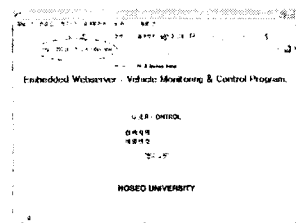


그림 6. 임베디드 웹 서버에 접속한 화면

그림 6은 임베디드 웹서버의 IP와 index.html를 웹 브라우저의 창에 입력해 자동차에 접속한 화면이다. 첫 화면에 접속하게 되면 사용자임을 검증하는 화면이 나오도록 하였다. 미리 입력된 사용자가 아니면 자동차의 상황을 모니터링 할 수 없고, 제어 또한 할 수 없도록 하였다.

2.3 ECU (Electronic Control Unit)

자동차에 존재하는 전자제어장치인 ECU에는 자동차에서 사용되고 있는 센서신호들과 스로틀각 등 자동차의 전반적인 내용들을 살피고 상황에 맞는 신호를 전자적으로 출력해줘 자동차 운행에 많은 영향을 준다. 이러한 ECU의 단자를 이용하면 자동차의 상태를 파악하기 용이하기 때문에 본 연구에서는 ECU의 신호를 CAN 제어기에 연결해 ECU의 정보를 임베디드 시스템에 전송하게 하였다.

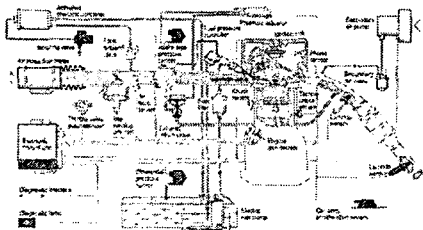


그림 7. 자동차 내부의 ECU 연결도

그림 7은 ECU가 자동차 내부곳곳의 신호들을 파악하고 제어하는 모습을 보여주고 있다.

2.4 전체 시스템 구성도

본 연구에서는 그림 8과 같이 실험 차량의 ECU에 CAN 제어기를 연결해 다양한 자동차의 정보들을 취득하고 제어할 수 있도록 하였다. 또한 자동차의 곳곳에 CAN 제어기와 드라이버를 이용해 센서 신호 취득이나 On/Off 신호를 처리 하도록 하였다.

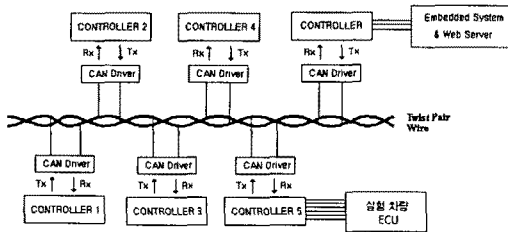


그림 8. 자동차 감시를 위한 시스템 구성도

자동차에 존재하는 전자제어장치인 ECU는 자동차에서 사용되고 다양한 신호를 파악할 수 있기 때문에 필수적으로 CAN 제어기를 인터페이스 시킨다.

3. 실험

본 연구의 검증을 위한 실험에서 그림 9와 같은 임베디드 시스템에 무선 랜카드를 설치하여 자동차의 상황을 TCP/IP로 인터넷상의 웹으로 모니터링 하도록 하였으며, 그림 10과 같은 위치에 분산되어 있는 CAN 제어기들과의 통신을 위해 마스터용 CAN 보드를 설계하여 임베디드 시스템과 인터페이스 하였다. 분산되어 있는 CAN 제어기들은 자동차의 모니터링을 위한 입력 기능 뿐 아니라 자동차를 제어하기 위한 출력도 가능하기 때문에 원격으로 자동차를 제어할 수도 있다. 자동차의 상황을 파악하기 위해서 ECU에 연결된 마스터 CAN 보드가 ID와 요청사항을 보내게 되면 각각의 CAN 보드들은 정



그림 9. 임베디드 시스템 & Master용 CAN 보드

해져 있는 측정 데이터를 추출해 전송하게 되고 마스터 보드와 연결된 임베디드 시스템은 그 내용을 무선 랜카드를 통해 웹으로 전송한다.

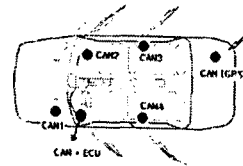


그림 10. 분산 되어진 CAN 제어기 위치

그림 11은 분산되어진 CAN 제어기를 중 원하는 제어기를 선택하고 출력을 기다리면 나타나는 웹브라우저 화면이다. 이때는 CAN 제어기를 두 개까지 선택하도록 하였다.

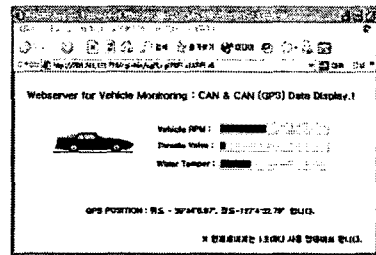


그림 11. 데이터 출력화면

4. 결론

최근에 무인운행이나 ACC (Automatic Cruise Control) 와 같이 자동차가 자동화나 지능화(Intelligent) 되어가기 때문에 기존 단일 ECU 제어방식과 같은 통합제어 방식은 비효율적이고 신뢰성이 저하되는 현상이 발생한다. 따라서 본 논문에서는 분산제어가 가능하고 Wire Harness와는 반대로 시리얼로 데이터를 송수신 할 수 있어 경량이며, 신뢰성이 우수한 CAN 제어기와 임베디드 시스템을 이용해 정지 및 운행 중인 자동차를 원격지에서 인터넷을 이용 모니터링 할 수 있도록 구현 하였다. 또한 PDA와 같은 모바일 기기로도 모니터링이 가능하기 때문에 운행 중에도 운전자는 자동차의 상황을 감시하여 이상 유무를 확인토록 하였다.

[참고 문헌]

- [1] C. D. Leidigh, "Web Based Management of Network Device", ESCC 2001, no 204, Chicago, 2001
- [2] L.Q.kong, J.Malee and T.Korte, "A Simple Architecture for Real-Time Web Based Device Control and Monitoring", ESCC 2001, no. 230, Chicago, 2001.
- [3] E.A.Bertz, "By-wire cars turn the corner",IEEE Spectrum, pp. 68-73, April 2001.
- [4] 정차근, "Controller Area Network(CAN) 통신 프로토콜에 의한 자동차 신호 및 센서 제어 시스템의 개발", 신호처리시스템학회 3권3호, 54-62, 2002