Fieldbus 네트워크 접속기의 하위계층 구현

°김현기, 이전우, 황선호, 이혁희, 채영도
한국전자통신연구소 자동화시스템 연구실

Lower Layer Implementation of a Fieldbus Network Interface

°H.K.Kim, J.W.Lee, S.H.Hwang, H.H.Lee, Y.D.Chae
Electronics & Telecommunications Research Institute
Automation Systems Section

ABSTRACT

Fieldbus is a low level serial digital network which will be used for factory automation. This paper describes lower layer implementation of a Fieldbus network interface. Physical layer provides hardware interface between IBM-PC and Fieldbus. Also, physical layer uses manchester coding, shielded twisted pair lines and RS-485 electrical standard. Data link layer includes Intel's iDCX96 real time executive for 8797 one chip microcontroller and Fieldbus data link protocol software.

1. 서 론

현대의 생산 현장에는 복잡한 공정을 자동으로 제어하기 위해 많은 센서(sensor)와 액추에이터(actuator)들이 단위 기계의 내부 및 공장 필드에 배치되어 있다. 이러한 센서, 액추에이터의 point to point 연결은 그 수가 증가 할수록 많은 공간을 차지하게 되고 더욱 복잡하게 되어 유지 및 보수에 많은 문제점이 생긴다. Fieldbus는 이러한 문제점을 해결하기 위하여 기존의 아날로그 신호 전송방식을 다중화된 디지탈 신호 전송방식의 직렬 데이타 버스로 대체한 것으로 설치의 간편성 및 효율적인 유지, 보수를 도모하고자 한다. 이 네트워크를 사용할 경우 기존 방식보다 케이블 및 설치 비용을 줄일 수 있을 뿐 아니라 기기간에 보다 정확하고 많은 양의 데이타를 교환할 수 있으며, 기기를 서로 접속하거나 분리 시키는 것이 쉬워지는 등 여러가지 장점이 있다. [1]

Fieldbus는 실시간 통신을 위해서 컴퓨터 통신 표준 모델인 ISO(International Standards Organization)의 OSI 7 계층 구조에서 계층 1, 2, 7 만을 사용한 통신 표준 모델을 정하였다. 즉 Fieldbus는 물리 계층, 데이타링크 계층 및 응용 계층으로 구성된 3 계층 구조를 가진다. [2]

Fieldbus의 표준화는 ISA(Instrument Society of America)와 IEC(International Electrotechnical Commission)에서 하나의 공통된 국제 표준화를 추진중에 있으며, 물리 계층 및 데이타링크 계층은 현재 "Draft Standard"가 발표된 상태이며 응용 계층의 표준화는 미흡한 편이다. [3][4] 한편, 프랑스, 독일 및 노르웨이에서는 FIP, PROFIBUS 및 ISIBUS라는 이름으로 각기 Fieldbus의 시제품을 개발한바 있다. [5]

본 논문에서는 Fieldbus의 특성을 갖는 EFbus(ETRI Fieldbus)의 규격을 바탕으로 Fieldbus 네트워크 접속기의 하위계층에 해당되는 물리 계층 및 데이타링크 계층에 대한 구현 내용을 기술 하고자 한다. [6][7][8]

2. 하위계층 프로토콜

2.1 물리 계층

물리 계층은 ISO의 7 layer 모델상에서 계층 1에 해당되며 데이타링크 계층으로 부터 심볼을 받아 전송매체에 전송할 신호를 인코딩하고 전송매체로 부터 받은 신호를 디코딩하여 비트 레벨에서 에러를 점검한다. Fieldbus는 디지탈 신호를 사용하는데, 이때 통신 특성을 향상 시키기 위해서 전송 신호에 출력 신호를 같이 전송할 수 있는 맨체스터 (Manchester) II 코딩 (Coding) 방식을 이용한다. Fieldbus의 물리 계층 규격은 다음과 같다.

- 데이타 전송속도 : 31.25 Kbps (공정 제어 분야)
  1 Mbps (생산 자동화 분야)
- 전송로 구성 (Topology) : 버스 topology
- 변조 방식 : Baseband 방식 (맨체스터 II 코딩 방식)
- 전송 매체 : Shielded twisted pair 케이블

2.2 데이타링크 계층

데이타링크는 변수값 교환과 메시지 전달의 2가지 서비스를 제공한다. 변수는 비교적 적은 양이면서 주기적으로 또는 자주 교환되는 데이타이고, 메시지는 비

교적 많은 양이면서 교환 회수가 적은 데이타이다. 변수값 교환 서비스는 시스템내의 각 변수가 identifier와 관련지워 진다. Configuration시에 각 entity들은 변수값의 producer 또는 consumer로 지정되고, 통신시 대상 entity는 identifier로 지정된다. 메시지 전달 서비스에서는 메시지 송수신과 관련된 응용 entity에 한개 이상의 address가 지정된다. 별도의 connection 과정 없이 메시지내에 source address와 destination address를 포함 시킴으로써 메시지 전달과 함께 통신 entity들이 연결된다.

Medium access 방법은 버스 제어기 (Bus Arbitrator : BA) 라고 하는 device가 identifier frame을 전송 함으로써 각 entity들에 medium access 권한을 부여한다. 데이타 교환은 먼저 버스 제어기가 identifier frame을 broadcast 하면, identifier와 관련된 데이타를 제공하는 producer device 가 데이타를 담은 response frame을 broadcast하고 모든 consumer device들은 response frame을 수신하여 필요한 데이타를 얻는다. 이렇게 1회의 교환이 완료되면 버스 제어기는 configuration시의 규칙에 따라 그 다음 순서의 데이타 교환을 시작한다.

3. 접속기의 하드웨어

3.1 전체의 구조

Fieldbus 네트워크 접속기의 하드웨어는 Fieldbus의 물리 계층과 관련된 부분으로 호스트 (host)인 IBM PC와 전송 매체(medium)에 인터페이스 된다. 구현된 Fieldbus 네트워크 접속기는 다른 노드의 Fieldbus 인터페이스 기기들과 데이타를 교환하며 호스트가 필요로 하는 정보를 주고 받게 된다.

Fieldbus 네트워크 접속기는 프로세서 및 메모리 회로, 호스트 접속회로, 전송매체 접속회로들로 구성되며 접속기의 전체적인 구조는 그림 3.1과 같다.

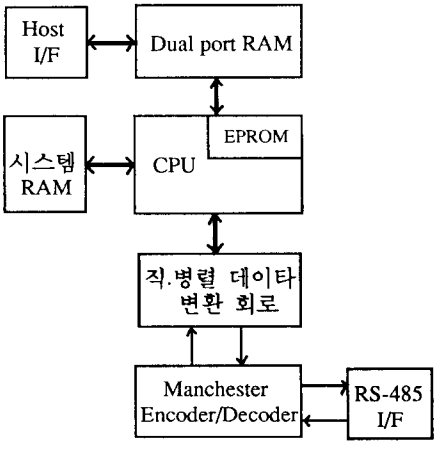

그림 3.1 Fieldbus 접속기의 하드웨어 블록 구조

프로세서는 12 MHZ에서 동작되는 INTEL사의 16 비트 원칩 마이크로콘트롤러 (One-chip Microcontroller)인 8797BH를 사용하였으며 자체내에 8 Kbyte EPROM과 시리얼 입출력 포트를 내장하고, 데이타의 입출력 제어, 인터럽트 처리등 주변 하드웨어를 제어하며 호스트와 전송 매체간에 데이타 교환을 수행한다. 호스트 인터페이스 부분은 전송 매체에서 수신된 데이타를 호스트에 전달하거나 호스트에서 전송 매체로 전송할 데이타를 수신하는 부분으로 Dual-port RAM을 이용하였다. 전송 매체 인터페이스 부분은 전송 매체상에 전송할 데이타를 맨체스터 인코우더와 RS-485 드라이버(driver)를 통해 twisted pair 케이블로 전송하고, 이 케이블로 전송되어 오는 데이타를 RS-485 리시버 (receiver)와 맨체스터 디코우더를 통해 수신 한다. 이때 데이타의 전송 속도는 EFbus 규격에 따라 31.25 Kbps로 정하였다.

3.2 프로세서 및 메모리 회로 [9]

이 부분은 8797BH 원칩 마이크로콘트롤러와 이에 의해 입출력되는 여러 제어 신호들에 대한 처리를 담당하는 프로세서 및 EPROM, 시스템 RAM, Dual-port RAM을 가진 메모리들로 구성된다.

8797BH CPU는 칩 내부에 8 Kbyte EPROM, 232 byte RAM, 256 byte register file, A/D converter, 시리얼 인터페이스, 입출력 포트, 타이머등을 내장하고 있으므로 부속 소자들의 수를 상당량 줄일 수 있으며 64 Kbyte 까지 메모리 addressing이 가능하다.

시스템 ROM은 8797BH 원칩 마이크로콘트롤러에 내장되어 있는 8 Kbyte의 내부 EPROM을 사용하며 여기에는 데이타링크 계층에 해당되는 소프트웨어가 실리게 된다.

시스템 RAM은 32 Kbyte 영역을 사용하며 Dual-port RAM을 통하여 호스트와 전송 매체간에 전송 되어진 정보를 처리하기 위한 영역으로 사용되어 지므로 충분한 영역이 확보 되어야 한다. 시스템 데이타 버스가 16 비트 이므로 8 비트 메모리 디바이스를 even address와 odd address로 구분하여 프로그램 또는 데이타를 저장시킨다.

Dual-port RAM은 Fieldbus 네트워크 접속기와 호스트 간의 정보 교환에 사용되며 공유 RAM 영역은 4 Kbyte로 2개의 V61C32 Dual-port RAM을 사용한다.

3.3 호스트 접속회로

Fieldbus 접속기와 호스트 시스템의 접속은 2개의 2 Kbyte Dual-port RAM을 사용하여 4 Kbyte의 공유 RAM 영역을 확보하고 폴링 (polling) 과 레지스터(register)에 의한 방법으로 구현 하였으며, 메모리 영역중의 2 byte를 송수신 레지스터로 사용하여 Fieldbus 접속기와 호스트 시스템간의 여러 command와 status를 주고 받는데 사용 하였다. 이에 대한 회로도는 그림 3.2와 같다.

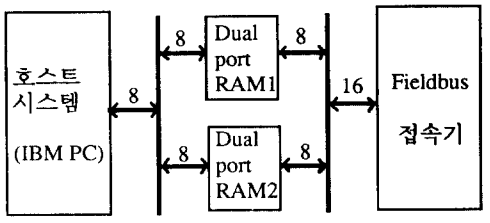

그림 3.2 호스트 접속 회로도

Dual-port RAM은 칩 내부에 중재 회로를 가진 고속 RAM으로 두 시스템 버스가 완전히 분리되어 두 시스템간의 정보를 교환한다.

3.4 전송매체 접속회로

이 곳은 Fieldbus의 물리 계층에 해당되는 부분으로 Fieldbus 접속기와 전송 매체간에 데이타의 송수신을 담당하는 회로이다. 여기에는 직병렬 데이타 변환 회로, CRC 발생기/검사기, 맨체스터 인코우더/디코우더 (MED), RS-485 드라이버/리시버 및 이의 주변 회로들로 구성되며 그림 3.3과 같다.

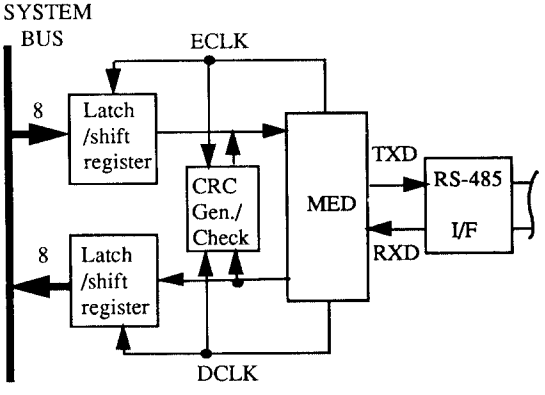

그림 3.3 물리 계층 접속 회로도

Latch/shifter register는 8 비트 병렬 입력 데이타를 직렬 출력 데이타로, 또는 직렬 입력 데이타를 병렬 출력 데이타로 변환해 주는 역할을 한다. CRC 발생기/검사기는 직렬 디지탈 데이타 처리 시스템에서 데이타 송수신시 에러 점검을 위한 곳으로 데이타 전송 완료후 CRC 검사기 frame이 발생되고 데이타 수신 완료후 에러 여부를 체크하게 된다. MED는 디지탈 신호를 코딩(coding) 하는 부분으로 이 코딩 방식은 전송 신호에 클럭 신호를 함께 전송하는 방법으로 직렬 데이타 통신에서 MED는 NRZ(Non-Return to Zero) 데이타를 맨체스트 코드로, 또는 맨체스터 코드를 NRZ 코드로 변환해준다. Fieldbus에서 사용되는 전송매체는 네트워크에서 transmitter와 receiver간의 물리적 경로로서 shielded twisted pair 케이블을 사용한다. 이 케이블과의 인터페이스는 EIA의 직렬 통신에 사용되는 전기적 규격인 RS-485 방식을 사용한다.

- 버스 방식 : 멀티 드롭 (multi-drop)
- 최대 접속 노드수 : 32개
- 전송매체 : shielded twisted pair 케이블
- Termination : 120 Ω
- 인터페이스 : SN75174/175 line driver/receiver

4. 접속기의 소프트웨어

4.1 전체의 구조

Fieldbus 접속기의 소프트웨어는 데이타링크 계층에 해당되는 소프트웨어로 INTEL사에서 제공하는 iDCX 96 real time multitasking executive 상에서 동작된다. 데이타링크 계층 소프트웨어는 응용 계층으로 부터 request primitive를 받아서 그 primitive에 해당되는 서비스 제공에 필요한 PDU를 만든 다음 물리 계층을 통하여 다른 노드로 전달한다. 또한, 다른 노드로 부터 PDU를 받아 이를 분석, 처리하고 응용 계층에 indication 또는 confirm primitive를 전달한다. 이 기능들은 다음과 같이 4개의 task로 구성된다.

- 응용 계층 인터페이스 task (task1)
- 물리 계층 인터페이스 task (task2)
- Elementary cycle 관리 task (task3)
- Micro cycle 관리 task (task4)

위 4개의 task 중 task1과 task2는 producer/consumer entity가 되고 task3와 task4는 bus arbitrator entity가 된다. 각 task의 priority는 task4가 가장 높고 task3, task2, task1의 순서로 주어진다. Task 모듈간의 통신은 공유 메모리와 메시지 교환 방식을 이용하며 task간 통신구조는 그림 4.1과 같다.

4.2 응용 계층 인터페이스 task

응용 계층으로 부터 버퍼 읽기, 버퍼 쓰기, 비주기적 버퍼 전달, 메시지 전달등의 request primitive를 받아서 필요한 처리를 수행한다. 또한, 물리 계층 인터페이스 task로 부터 새로운 데이타의 수신통보, 이전에 update된 데이타의 처리 통보를 iDCX96이 제공하는 메시지로 받아서 각 통보에 대한 indication이나 confirmation primitive를 응용 계층에 전달한다.

4.3 물리 계층 인터페이스 task

이 task는 버스 제어기 (BA)의 기능을 포함하는지의 여부에 따라 기능이 다르다. BA 기능을 수행하지 않는 노드일 경우에는 물리 계층으로 부터 IDentifier frame(ID frame)을 받아서 분석하고, 그에 따라 지정된

변수값이나 메시지를 response frame으로 만들어 물리 계층에 전달한다. 이렇게 1회의 통신이 끝나면 그 사실을 응용 계층 인터페이스 task에게 알린다. 또한 데이타 수신시에도 데이타를 해당 버퍼에 보관하고 그 사실을 응용 계층 인터페이스 task에게 알린다. 데이타 송수신 여부의 결정은 ID table에 의하여 결정 되는데 ID table에는 각 ID에 대하여 해당 노드가 consumer인지 또는 producer인지가 결정된다. BA 기능을 수행하는 노드일 경우에는 elementary cycle 관리 task로 부터 ID를 받아서 ID frame을 만들어 물리 계층에 전달하는 기능을 더 갖게 된다.

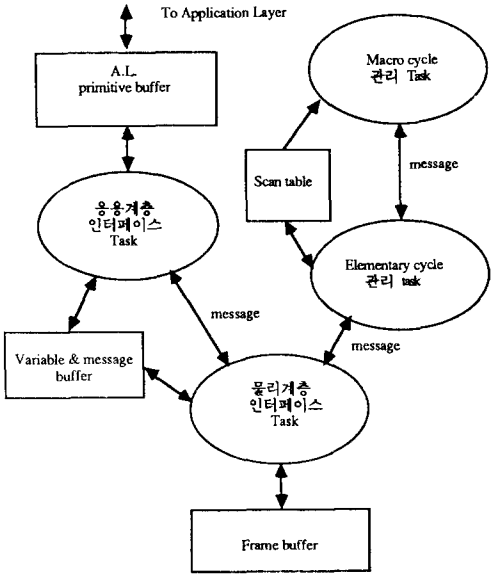

그림 4.1 Task간 통신구조

4.4 Elementary cycle 관리 task 및 Macro cycle 관리 task

이 두 task는 BA의 기능을 수행하는 task이다. Elementary cycle 관리 task는 Macro cycle 관리 task가 정해 주는 cycle내에서 버스상에 교환될 변수 ID 또는 메시지와 관련된 ID를 결정한다. Macro cycle 관리 task는 버스 운용 cycle을 periodic, aperiodic, messaging, stuffing의 4가지중 하나로 결정해 준다. periodic cycle에서는 주기적으로 교환되는 데이타들의 통신이 이루어 지며, aperiodic cycle에서는 비 주기적으로 교환되는 데이타들의 통신이 이루어 진다. 또한 messaging cycle 에서는 메시지 전달 서비스가 수행된다. stuffing cycle은 periodic cycle의 timing 조정을 위해 존재하며 이동안 버스는 idle 상태가 된다.

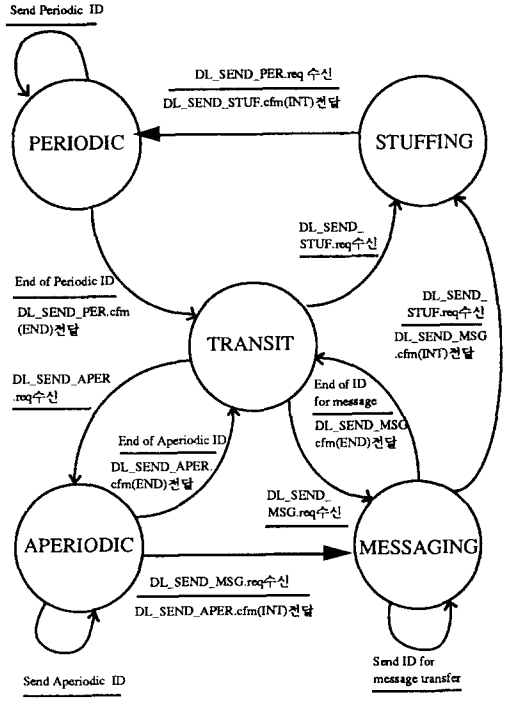

그림 4.2 Elementary cycle 관리 task의 state transition

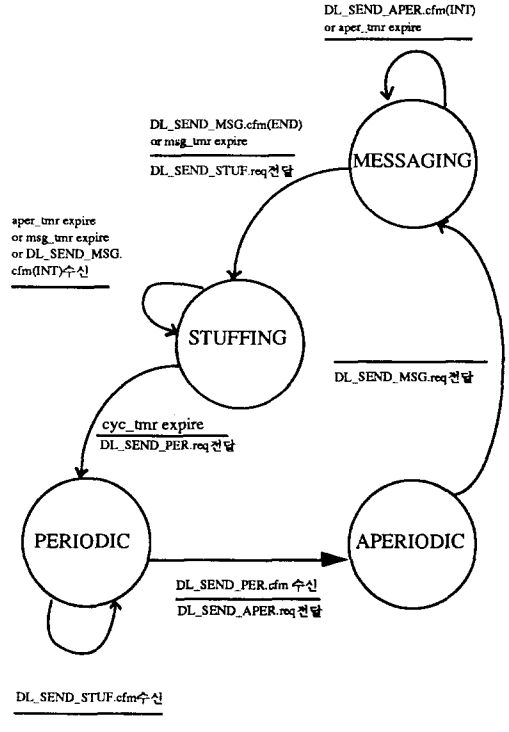

그림 4.3 Macro cycle 관리 task의 state transition

5. 실 험

본 논문에서 구현한 Fieldbus 네트워크 접속기는 IBM PC를 호스트로 하는 인트페이스 보드이다. Fieldbus 접속기는 자체의 local processor, local memory, 호스트와의 데이타 교환을 위한 공유 메모리, Fieldbus를 연결하는 전송 매체 인터페이스 회로를 가지며, Fieldbus 프로토콜 소프트웨어는 CPU의 ROM에 firmware로 탑재 되어 있다. Local processor로는 16 비트 원칩 마이크로콘트롤러인 8797BH를 사용하였으며 호스트와의 통신용인 공유 메모리는 Dual-port RAM을 사용하여 구성 하였다. 전송 매체 인터페이스 부분은 MED, CRC 발생기/검사기, line 드라이버/리시버등을 포함한다. 실험은 2대의 Fieldbus 네트워크 접속기를 제작하였으며, 구현된 접속기의 실험을 위한 네트워크 구성은 그림 5.1과 같다.

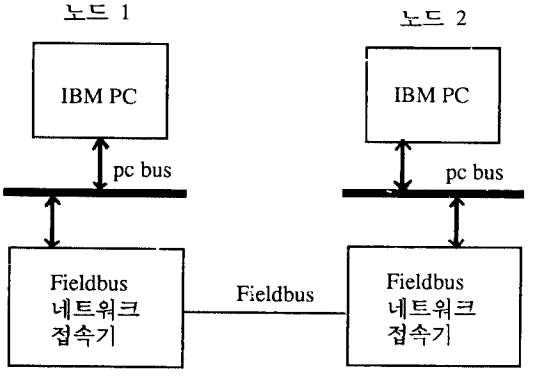

그림 5.1 Fieldbus 네트워크 구성도

위 그림은 2대의 Fieldbus 접속기를 각각 IBM PC에 장착하고 데이타링크 소프트웨어를 올린후 접속기들을 Fieldbus 네트워크로 구성한 것이다. 실험 결과 Fieldbus를 통하여 노드1과 노드2 시스템간의 프레임 송수신시 데이타 전송이 오류없이 수행됨을 확인 하였다.

6. 결 론

본 논문에서는 EFbus(ETRI Fieldbus)를 바탕으로 하여 제작 된 Fieldbus 네트워크 접속기의 하위 계층에 해당되는 물리 계층 및 데이타링크 계층에 대한 구현 내용을 기술 하였다. 이를 토대로 접속기를 Fieldbus 네트워크로 구성시 노드간의 데이타 송수신이 오류 없이 수행됨을 확인 하였다. Fieldbus는 공정 제어 및 생산 자동화 분야에 응용되며, 데이타 크기가 짧고 빠른 응답의 실시간 처리를 필요로 하며, 또한 기기간에 전달되는 데이타는 일정한 주기를 갖고 cyclic하게 전달된다. Fieldbus 네트워크 접속기의 특징중 물리 계층은 맨체스터 II 코딩의 데이타 전송 방식과 RS-485 방식을 사용한 twisted pair 케이블의 사용이며, 데이타링크 계층은 centralized MAC 방식과 producer/consumer 개념의 데이타 교환 방식을 사용한 것이다. 그리고 향후 과제로 응용 계층을 포함한 Fieldbus 네트워크 접속기가 개발되어야 하며 물리 계층과 데이타링크 계층의 네트워크 소프트웨어 및 하드웨어가 하나의 소형 칩으로 제작되어야 한다. 또한, 네트워크 management 기능과 optical fiber medium 사용에 대한 연구도 수행되어야 한다.

참고 문헌

[1] P. Pleinevaux 외, "Time Critical Communication Network : Fieldbuses", IEEE Network, pp.55-63, May. 1988

[2] Bruce Armitage외, "Fieldbus:An Emerging Communications Standard", Microprocessors and Microsystems, pp.555-562, Vol12, December. 1988

[3] "Draft Fieldbus Standard : Physical Layer", ISA SP50, Oct,1989

[4] "Specification for the Physical Layer", IEC SC65C WG6, July. 1990

[5] "FIP Technical Description", FIP

[6] 황선호 외 "EFBUS Specification", ETRI, TM, 4. 1990

[7] 이전우 외 "공장자동화용 데이타버스(EFBUS)의 설계에 관한 연구", 한국자동제어학술회의 논문집,10. 1990

[8] 채영도 외 "공장자동화를 위한 인터페이스 기술 개발에 관한 연구", 최종보고서, 과학기술처, 11. 1990

[9] "Embedded Controller Handbook", Intel, 1988