

## 전송효율을 개선한 다중접속 직렬통신 구현

이영석, 한경호  
단국대학교 전기공학과

### Implementation of Multiple Access Serial Communications with Improved Transmission Control

Young suk Lee, Kyong ho Han  
Department of Electrical Engineering, Dankook University

**Abstract** - In this paper, we proposed the implementation of multiple access serial communications with improved transmission control. For serial communications, RS232C protocol is used and the transmitting data and is merged to form data channel. Multiple host access is configured by using the common data channel and ground channel. 8bit data transfer with variable frame size is transferred by using the 16bit host ID. Packet is composed of HEADER, receiver ID, variable length data frame, TAIL and CRC informations. Multiple hosts are allowed to transfer packet with the other hosts through the common communication channel. Byte-stuffing is used to differentiate the transfer rate of PC is performed.

서 사용 없이 선로에 아무런 데이터가 없다면, 원하는 작업을 문제없이 수행할 수 있다. 그러나 네트워크를 다른 PC가 사용하고 있는 경우 네트워크 사용을 시도한다고 하면, 이때 선로에 충돌이 발생하게 된다. 일단 충돌이 발생하면 먼저 사용하고 있던 PC가 계속 사용할 수 있도록 기다린다. 그리고 잠시 후 다시 사용을 시도한다. 이때 각 PC의 대기시간이 서로 달라야 다시 충돌을 일으키지 않는데, 대기 시간은 호스트 내에서 난수 발생기를 이용하여, 데이터 재 전송시 다른 PC의 데이터와 충돌되는 확률을 줄이도록 해결한다.

공통 통신 선로에 여러 호스트를 연결하여 다중접속통신을 구현하기 위해서는 CSMA프로토콜에 의해 각 호스트에서는 일정한 규약을 갖는 데이터 프레임으로 데이터 송, 수신을 구현하여야 한다. 이에 본 논문에서는 아래와 같은 프레임구조로 CSMA에 의한 직렬통신을 구현하였다.

### 1. 서 론

데이터를 전송할 때, 하나의 선을 통해 한 비트(bit)씩 보내는 통신방법을 직렬통신이라 한다. 이는 병렬 전송방식에 비해 데이터 전송속도는 느리지만, 송신, 수신선 2개만으로도 데이터 전송이 가능하므로 비용이 절감되고, 데이터 전송 중 지연차이에 영향을 받지 않아 비교적 먼 거리까지 전송이 가능하다. 이러한 직렬통신의 대표적인 방법으로는 RS232C, RS422, RS423등이 있다.

기존의 시스템구성에서 다수의 호스트가 직렬통신에 의하여 연결되기 위해서는 통신선로수가 많이 필요하게 된다. 이러한 점을 보완하기 위해 공통 통신 채널을 이용한 다중 접속 방식을 구현하였고 기존 RS-232C의 송신, 수신, 접지의 3선으로 사용되었던 직렬통신방식을 2선(데이터, 접지선)을 이용하여 구현하였다.

### 2. 본 론

#### 2.1 충돌검지 다중접속(CSMA)프로토콜

CSMA/CD(Carrer Sence Multiple Access with Collision Detection)는 ANSI/IEEE 802.3에서 규정된 다중 접속시 가장 많이 사용되는 버스/트리 구조의 매체액세스 제어 기법으로, 두 개 또는 그 이상의 호스트가 하나의 공통 전송매체를 공유하는 수단에 의해 액세스가 가능하다.

충돌검지 다중접속방식은 데이터를 송신하려는 호스트가 네트워크상에 다른 호스트가 통신하고 있는지를 조사해 데이터의 이동이 없을 경우에 데이터를 전송하는 구조다. 만약, 동시에 여러 호스트에서 데이터를 전송할 경우 충돌이 발생한다. CSMA/CD는 이 충돌을 감지하는데, 충돌이 발생한 경우에는 일정 시간 대기하다가 다시 신호를 보내 통신을 시도하는 방식이다.

한 PC가 네트워크를 사용하려 한다고 가정하자. 먼저, 네트워크의 선로 상태를 점검한다. 이때 다른 PC에

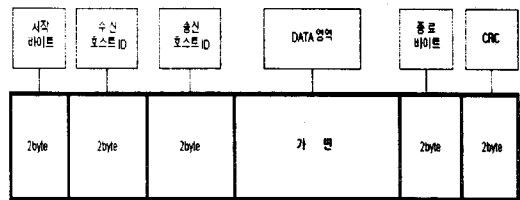


그림 1. 데이터 프레임 구조

데이터 프레임의 구조는, 프레임의 시작을 나타내는 시작 바이트, 수신 호스트 ID(Destination Address) 바이트, 송신 호스트 ID(Source Address) 바이트, 순수 데이터영역, 종료 바이트, 오류검사부호 바이트로 구성된다. 시작 바이트는 2바이트로 구성되며, 데이터 프레임의 시작을 나타낸다. 수신 및 송신 호스트의 ID는 각각 2바이트이고, 각각의 호스트는 고유 주소를 갖고 있다. 이러한 고유 주소는 각 호스트의 주소, 공통 주소 및 그룹 주소 등으로 분할하여 사용한다. 순수 데이터영역은 전송하고자 하는 명령어 및 정보를 싣는 부분으로 가변의 데이터 크기를 갖는다. 오류검사부호 바이트는 앞의 바이트, 즉, 시작 바이트와 송, 수신 호스트 ID바이트, 데이터 바이트의 합과 같으며 전송시 충돌을 검지하는데 사용된다.

데이터 프레임을 전송하고자 하는 호스트에서는 자신의 주소(송신 호스트 ID), 전송하고자 하는 상대 호스트의 주소(수신 호스트 ID), 데이터 및 오류검사부호(CRC)를 위와 같은 프레임구조로 작성한 후, PC의 RS232C 송신 단자를 통해 통신 선로로 데이터 프레임을 전송한다. 이때 데이터 프레임을 전송하는 호스트의 RS232C 수신 단자에는 송신 단자를 통해 전송된 데이터가 전송과 동시에 Loop-back 된다.

#### 2.2 CSMA에 의한 직렬통신 제어방법

CSMA 프로토콜에 의한 RS232C 직렬통신을 구현하기 위하여 우선 기존의 송신, 수신, 접지 3선을 데이터, 접지 2선으로 변형하여 구현하였다.

PC 직렬포트 연결 커넥터

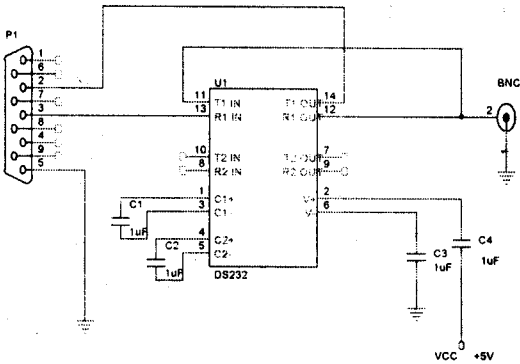


그림 2. 송수신 인터페이스 부분회로도

각 PC에서는 프로그램에 의해 제작된 데이터 프레임이 RS232C의 송신 단자를 통해 공통 통신선로로 전송되며 동시에 수신 단자로 수신된 바이트는 Loop-back 된다. 이에 송신PC에서는 수신 단자로 인터럽트에 의해 바로 Loop-back된 데이터가 송신 단자로 전송한 데이터와 같은가를 비교함으로써, 통신선로에서의 충돌여부를 검사한다.

만약, 충돌 검사를 하여 아무 이상이 없으면 다음 데이터를 위와 같은 방법으로 데이터 프레임에 계속 보내게 되지만, 충돌 검사에서 충돌이 검지 되면 임의의 시간을 기다린 후 다음 데이터를 재 전송하게 된다. 그림3은 다중접속 직렬 통신을 구현하기 위한 각 PC의 입, 출력 장치의 연결도이다.

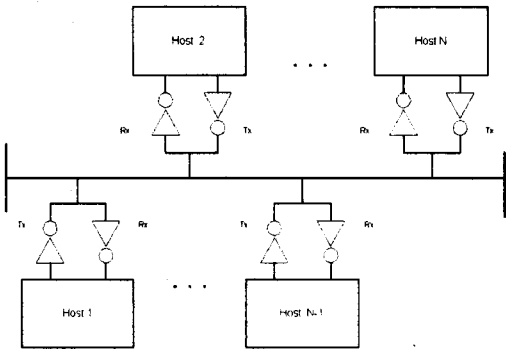


그림 3. 공통 통신 선로 연결도

그림3에서 송신 단자로 전송된 데이터는 공통 통신선로를 통해 각 PC로 연결되며 또한 송신 측 PC의 수신 단자로 직접 연결됨으로 충돌검지를 위한 Loop-back 신호를 제공하게 된다.

그림4는 인터럽트에 의해 데이터가 수신되면서, 동시에 송신데이터의 충돌여부 확인이 가능하도록 구현한 인터럽트 처리문을 나타낸 것이다.

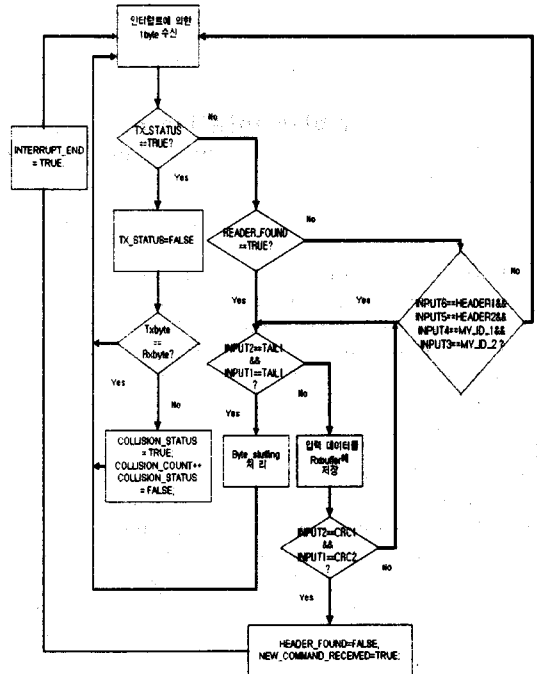


그림 4. 수신 인터럽트 흐름도

그림4를 보면, 인터럽트에 의해 송신된 데이터가 바로 수신된다. 일단 수신 인터럽트 구문에서, 맨 처음 확인해 봐야 할 것은 수신된 데이터가 자기 자신이 보낸 데이터인지, 아니면 다른 PC에서 자신에게 보낸 데이터인지를 확인해 봐야한다. 여러 PC가 공통 선로에 의해 통신을 하기 때문에, 같은 시간대에 하나의 PC만이 데이터를 보낼 수 있으므로, 수신된 데이터가 자신의 것인지 충돌여부를 확인해 봐야한다. 왜냐하면, 본 논문에서는 데이터의 충돌여부를 송신하는 PC가 데이터를 보내는 동시에, 충돌여부를 확인하게끔 하였기 때문이다.

그래서, 현재 데이터를 수신한 PC가 송신중인지 아닌지를 체크해야 한다. 송신 중이었으면 현재 수신된 데이터는 자신이 송신한 데이터이므로 자신이 송신한 데이터와, 수신 데이터가 같은지를 검사한다. 그 동안 다른 PC가 동시에 송신을 하였다면 자신이 보낸 송신데이터는 다른 PC가 송신한 데이터와 충돌되어 데이터 값이 바뀌므로 송, 수신데이터는 서로 다르게 된다.

또한 현재 데이터를 수신한 PC가 송신중이 아닐 경우라면, 현재 수신된 데이터는 다른 PC가 송신한 데이터임을 의미한다. 이러한 경우에는 다음과 같은 사항을 검사해야 한다.

수신된 데이터가 순수한 데이터 영역인지 아니면 프레임 구분자(delimiter)영역인지를 구분해야 하고, 프레임 구분자 영역에서는 수신데이터가 통신데이터 프레임 중, HEADER와 수신 호스트ID를 검사한다. 또한 수신 호스트ID가 PC 자신의 ID와 일치하는가를 확인하는데 이는 수신된 데이터가 다른 PC가 자신에게 보낸 데이터임을 확인하는 것이다.

수신된 데이터가 순수한 데이터 영역일 경우는 Byte-stuffing을 검사해야 한다. Byte-stuffing이란, 바이트를 중간 중간에 끼워 넣는 것인데, 순수한 데이터와 프레임 구분자(delimiter)와 같은 제어 정보를 구분하기 위한 방법이다.

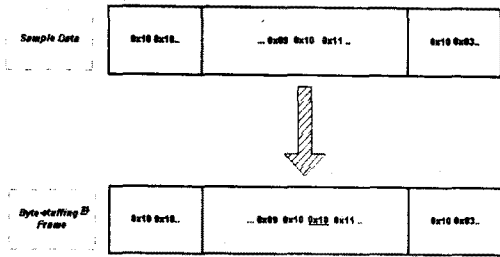


그림 5. Byte-Stuffing의 예

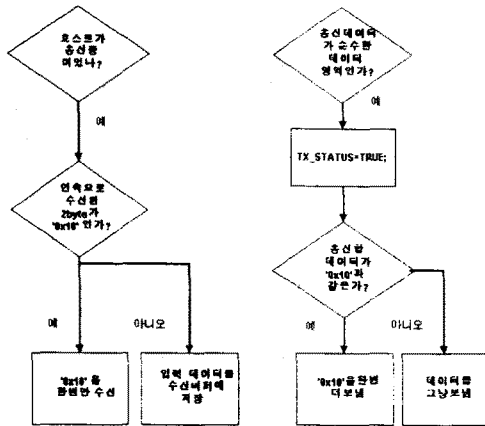


그림 6. 송, 수신시의 Byte-stuffing 흐름도

송신할 때 데이터프레임중 순수한 데이터영역 안에 HEADER와 같은 패턴의 데이터가 있을 경우 그 데이터를 한번 더 송신하여 데이터와 HEADER, TAIL과 구분한다. 수신시에는 HEADER와 같은 패턴의 데이터를 구분해야 하는데 이를 위하여 HEADER가 한번만 존재하는 프레임의 시작을 찾도록 하고 이후에는 HEADER와 같은 패턴의 데이터가 두 번 들어오면 Byte-stuffing된 데이터로 구분하여 두 개의 데이터 가운데 하나는 삭제하도록 한다. 그리고 맨 마지막에는 TAIL1, TAIL2를 체크하고 CRC1, CRC2가 다 들어오면 일단, 데이터 한 프레임의 수신이 완료된다.

데이터 수신과정에 의해 데이터 프레임용 모두 받으면, 데이터 프레임 전체에 대하여 오류검사를 실시하는 데 이를 위하여 전송된 CRC 부호와 수신 프레임에 의해 다시 계산된 CRC 부호를 비교한다. 두가지 CRC 부호가 같으면 데이터 프레임에 전송 오류가 없는 것으로 판단한다. 이처럼 전송된 프레임에 오류가 없으면 전송된 데이터를 수신버퍼에 저장하고, 오류가 있으면 전송된 프레임의 데이터를 무시하고 데이터 전송오류로 처리한다.

인터럽트 마지막에, INTERRUPT\_END라는 Flag를 사용하는데 이 Flag는 평상시에는 리셋되어 있다가 수신인터럽트 과정이 종료되면 세트되면서 인터럽트 처리 과정의 종료를 표시한다.

송신측 PC가 데이터를 연속적으로 보내는 과정에서 하나의 데이터를 보내고 다음데이터를 보낼 때, 확인해야 할 점은 인터럽트처리과정의 종료 여부와 이전에 송신한 데이터가 충돌이 발생하였는지를 체크한다.

### 2.3 CSMA에 의한 직렬통신 프레임 전송 시뮬레이션

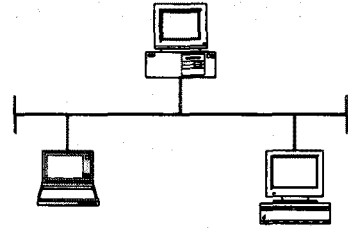


그림 7. 시뮬레이션에서의 PC 연결 상태

시뮬레이션을 위하여 RS232C 직렬통신을 구현하기 위해 그림 2에서와 같이 기존의 송, 수신용 묶은 데이터선과 기존의 접지선을 이용하여 PC를 연결하였다. 이와 함께, 8비트의 완전한 데이터와 가변의 데이터 프레임 길이를 가진, 임의의 데이터를 115,200bps의 전송 속도로서 3대의 PC를 공통선로에 의한 다중접속 직렬 통신방법으로 데이터 전송을 하였다. 이와 함께, Byte-stuffing을 이용하여 순수한 데이터와 프레임 구분자를 구별하였고, 데이터 수신시 이를 제대로 구별하는지를 확인해 보았다. 하나의 통신채널에 의해 여러 대의 PC가 직렬통신을 하기 때문에, 데이터의 충돌이 발생하게 되는데 이러한 데이터의 충돌은 수신 인터럽트에 의해 바로바로 체크하여, 충돌된 데이터의 재전송 여부를 결정하였고, 데이터의 충돌이 걸릴 경우, 데이터의 재전송 시간은 다수의 PC간의 전송 데이터의 재 충돌을 극소화하도록 하였다.

### 3. 결 론

기존의 송신, 수신, 접지의 직렬통신 3선로를 데이터, 접지의 2선으로 줄이고, 각각의 호스트에 ID를 부여함으로써, 충돌검지 다중접속 프로토콜을 이용하여 기존의 1대1 통신매체로 사용되던 RS232C 직렬통신을, 여러 개의 PC를 공통 통신선로에 의한, 다중접속 직렬통신을 구현하였고, 또한 직렬 통신을 위한 하드웨어를 간략화 시켰다.

이러한 방법에 의한 다중 접속 직렬 통신은 통신선로의 수를 줄일 수 있으며 공통통신 선로를 이용하여 다수의 제어기를 연결할 수 있으므로, 제어기의 설치 및 증설시 용이함은 물론, 이에 따른 비용절감의 효과도 기대된다.

### (참 고 문 헌)

- [1] ANSI/IEEE, ANSI/IEEE 802.3, Information technology Local and metropolitan area network, Part3:Carrier Sense Multiple Access with Collision Detection(CSMA/CD) access method and physical layer specifications,1993.
- [2] 방 재희, "C/C++ 시리얼 통신", 영진 출판사, 1995
- [3] 광 회수, "충돌검지 프로토콜에 의한 다중접속 RS232C 직렬통신 구현", 단국대학교, 1999
- [4] 이 옥양, "C 프로그래머를 위한 시리얼 커뮤니케이션", 가남사, 1990
- [5] 김 석주, "C로 하드웨어 주무르기", 가매 출판사, 1997