

TCN(Train Communication Network) 통신 시험용 WTB(Wire Train Bus) Analyzer 개발

The development of WTB(Wire Train Bus) Analyzer for the TCN(Train Communication Network) testing

전성준*

백진성**

손강호***

Jeon, Seong-joon

Paik, Jin-sung

Shon, Kang-ho

ABSTRACT

In Korea, TCN has applied to the Korean High-speed Train (HSR350X) through G7 High-speed Train development project. TCN is the most suitable international standard communication network for distributed control systems that is adopted for high-speed of vehicle, safety and flexibility. TCN is the network exclusively for the high-speed train and electrical trains.

This TCN satisfies the network standards. The network standards are real time communication, fault tolerance design, integrated data system, resistance of environment, automated recognition for modification of vehicle formation and maintenance.

The purpose of this research is applying the development of WTB analyzer which is part of communication network system TCN, to check the communication of high-speed trains and electrical trains.

1. 서 론

차세대 전송기술의 하나인 Fieldbus에 대한 연구가 본격적으로 진척됨에 따라 철도차량 분야에서도 IEC(International Electrical Commission) 와 UIC 위원으로 구성된 Working Group 22 (WG22)에서는 차량내의 부속 기기인 추진, 제동, 서비스 기기 및 출입문 제어 혹은 차량간의 프로그래밍이 가능한 기기들을 연결하기 위한 TCN(Train Communication Network, 이하 TCN)을 정의하고 표준안을 1999년에 IEC 61375로 표준화 하였다. TCN에 대한 표준화는 주로 유럽의 철도차량 부품 제조업체인 Adtranz, AEG, Siemens, Firema, HOLEC 등이 중심이 되어 이루어졌다.

국내에서는 G7 고속전철기술개발사업을 통하여 한국형고속열차(HSR350x) 시제차 7량에 적용되기 시작한 TCN은 고속전철 및 전동차를 위한 전용의 네트워크로써 차량의 고속화와 안전성 및 유연성을 위하여 채택되는 분산제어 방식을 위한 가장 적합한 국제표준의 통신방식이다.

본 연구는 고속열차 통신 네트워크가 갖추어야 할 실시간 통신, 내고장성 설계, 데이터 집적도, 환경에 대한 저항성, 차량 편성 변경 등에 대한 자동인식과 유지 보수성, 네트워크의 표준화 등의 조건을 만족시켜주는 통신 네트워크 시스템인 TCN 중 WTB(Wire Train Bus, 이하 WTB) 통신에 대한 Analyzer를 개발하여 추후 차세대 고속전철 및 TCN 통신 중 WTB 통신을 사용하는 모든 열차의 시험장비로 사용하여 열차의 검수 및 운행 전 점검 시 효율성 증대에 그 목적을 두고 있다.

* 인터콘시스템스(주), 책임연구원, 정회원

** 현대로템(주) 기술연구소 선행제품개발팀 주임연구원

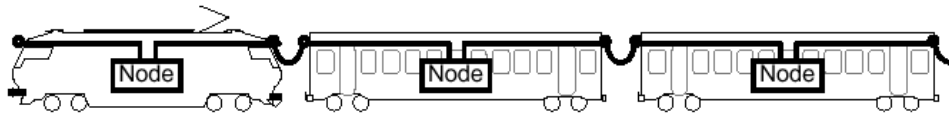
*** 인터콘시스템스(주), 대표이사

2. WTB (Wire Train Bus)

2.1 WTB의 소개

초기의 차량간 통신버스는 시리얼 통신버스로 개발이 되었다. 그후 국제적인 UIC 열차와 같은, 일상적인 기능을 수행하는 도중 열차를 재구성할 수 있는 열차 재구성을 위한 기능이 추가 개발되었다.

차량간 통신버스에서는 UIC 차량 22량까지 구성하고 이를 지원하는 통신 규약인 UIC leaflet 556의 요구조건을 이행하는 것에 대해 명시하고 있다. 차량간 통신버스의 구조는 [그림1]과 같이 차량간 통신을 지원하는 노드와 이 노드간을 연결하는 케이블로 구성되어 있다. 이 노드들 사이의 통신을 WTB라 한다.

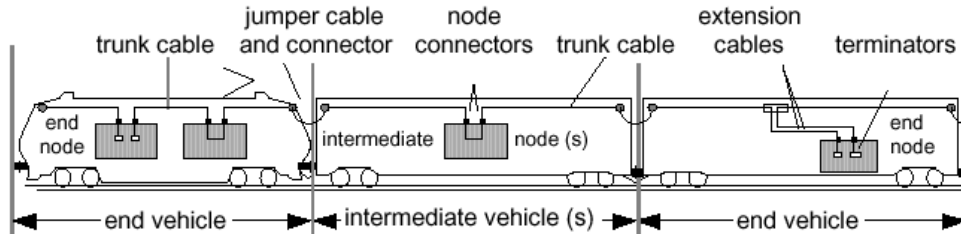


[그림1] Wire Train Bus

차량간 통신버스는 twisted, shielded wire pair 케이블을 이용하여 데이터를 1Mbit/s로 전송하며, 860m 길이 내에서 32개 node까지 지원한다. 그리고 필요에 따라서는 리피터를 사용하여 최대 62개의 노드까지 사용한다. 차량간 통신버스는 [그림2]에서 보는 바와 같이 차량들을 연결한다.

차량간 통신버스를 구성하고 있는 각 노드들은 trunk cable을 사용하여 연결되어 있으며 만약 노드와 trunk cable 사이의 거리가 멀 경우 extension cable을 이용하여 연결한다.

정상 동작 상태에서 열차를 구성하고 있는 차량 내 노드들은 열차의 양 종단에 있는 경우 End_Node가 되며, 그 외의 노드들은 Intermediate_Node가 되어 차량간 통신을 한다.



[그림2] WTB Topology

각 노드들은 2개의 채널을 가지고 있는데, 정상적인 데이터의 전송을 위한 메인 채널과 열차의 재구성등을 파악하기 위한 보조 채널이 있다. 열차의 양 종단에 있는 End_Node의 메인 채널에는 terminator가 연결이 되나, 열차 내부의 Intermediate_Node의 메인 채널에는 terminator가 연결이 되지 않는 채 차량간 통신에 참여하게 된다. 열차의 양 종단에 있는 노드에서만 활성화중인 보조 채널에서는 새로운 노드를 찾는 작업, 열차의 재구성을 검지하는 작업이 이루어진다. 만약 열차의 재구성이 확인이 되면 이를 확인한 노드는 마스터에게 이를 통보하고, 마스터는 열차 재구성이 가능하면 열차를 새로운 구성에 맞추어 재구성하게 된다.

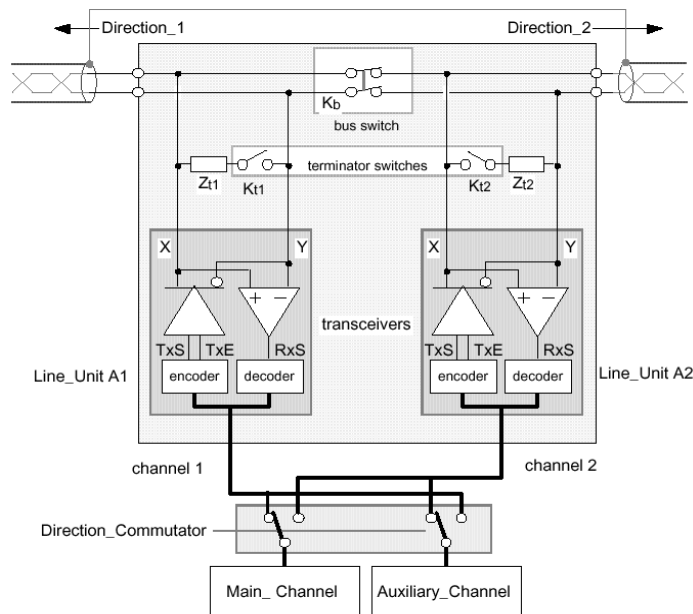
차량간 통신 버스는 하나의 마스터에 의해 운영이 되며, 이 마스터에 의하여 열차 제어 신호의 전송을 위해 사용하는 Process_Data와 승객정보와 같은 비정기적이고 크기가 큰 Message_Data의 전송, 그리고 열차의 노드 관리를 위한 Supervisory Data의 전송이 이루어진다. 열차를 구성하고 있는 노드가 정상상태에서 동작할 경우에는 마스터 권한의 이양은 일어나지 않으며, 열차의 재구성이나 기존 마스터의 고장이 발생할 경우 마스터 권한의 이양은 열차 재구성이라는 과정을 거쳐 발생하게 된다.

차량을 구성하고 있는 노드들은 마스터에 의해 노드 아이디를 할당받게 되므로 열차 재구성 등으로 인한 마스터권한의 이양은 모든 노드들이 새로운 노드 아이디를 할당받게 됨을 의미한다. 하나의 차량간 통신 버스 내에는 하나의 마스터만이 존재할 수 있기에 마스터 노드는 하나이지만 다른 여러 노드들도 열차재구성을 통해 마스터가 될 능력을 가지고 있다.

2.2 노드 구조

노드는 MAU(Medium Attachment Unit, 이하 MAU)를 통해 전송매체에 접근하며 MAU는 Line Unit, Direction Commutator, Main Channel, Auxiliary Channel로 구성되어 있다. Line Unit은 Bus_switch, Terminator_switch, Transceiver, Manchester encoder/decoder, protection circuit으로 구성되어 있다. Direction Commutator는 Main Channel, Auxiliary Channel를 direction 1 또는 direction 2에 연결하는 역할을 한다. Main Channel, Auxiliary Channel를 통해 HDLC frame과 제어신호를 Line unit에 보내고 받게 된다. 이와 같은 구조를 나타낸 것이 [그림3]이다.

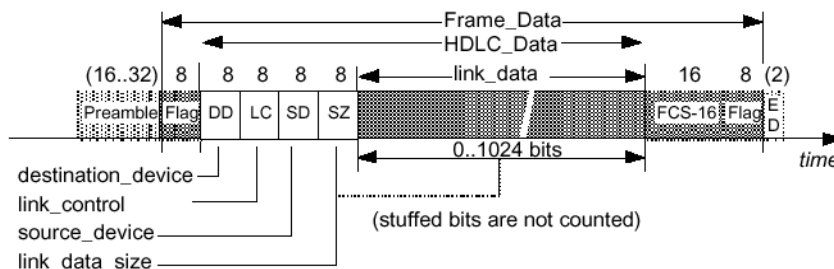
노드는 하드웨어 상에서 bus switch와 terminator switch의 스위치 제어에 의해 Intermediate_Node 또는 End_Node가 된다. End_Node 경우에는 Bus_Switch를 열어 놓고, Terminator_Switch를 닫음으로서 Main Channel은 내부 방향으로 연결시킴으로써 차량간 버스 통신을 하며, 활성화된 Auxiliary Channel은 열차 외부 방향으로 노출되어 있고, 새로운 노드의 검색에 사용된다. Intermediate_Node의 경우에는 Terminator_Switch를 열어두고, Bus_Switch를 닫음으로써 버스를 연결하게 되며 Main Channel만이 활성화되어 차량간 버스 통신에 참여하며, Auxiliary Channel은 새로운 노드 검색이외의 데이터 프레임에 대해 무시한다.



[그림3] WTB MAU with duplicated Line_Unit

2.3 데이터 프레임 구조

모든 프레임의 구조는 HDLC (ISO/IEC 3309) 형태인 [그림4]처럼 되어 있다. 각 프레임은 preamble이 서두에서 시작되며 끝에 End Delimiter가 붙는다. Preamble과 End Delimiter는 Manchester 인코더에 의해서 첨가되고, Manchester 디코더에 의해서 제거된다. Preamble의 길이는 16 또는 32 bit를 지나 기본은 16bit이다. Preamble 과 End Delimiter는 Frame_Data에 속하지는 않는다.



[그림4] WTB Frame (extended ISO/IEC 3309)

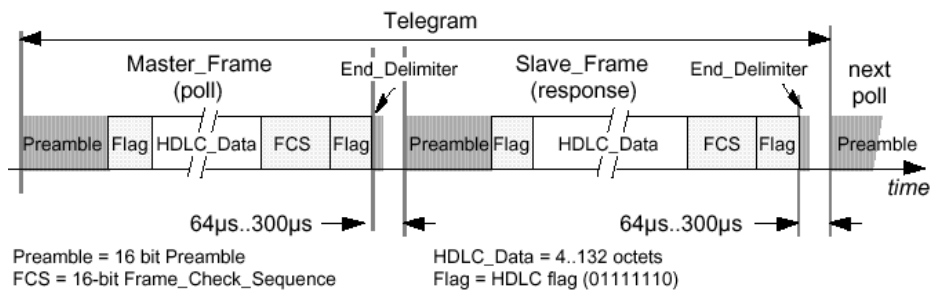
Frame_Data는 8bit의 프래그(01111110)에 의해서 그 범위가 정해진다. HDLC_Data는 8bit에 Destination_Device로 시작이 되며 이것은 현재 프레임이 향하는 목적지 노드(또는 브로드캐스팅)를 나타내는 것으로 HDLC 콘트롤러에 의해서 디코딩 된다. 다음에 오는 8bit의 Link_Control 은 WTB에서 미리 정해진 작업을 위해 정의가 되어있다. Source_Device는 이 데이터 프레임이 전송되는 곳의 주소를 가리키며, Size는 Link_Data에 해당하는 부분의 데이터 크기를 8진수로 나타낸다. Link_Data 뒤에 따르는 16bit의 Frame_Check_Sequence는 HDLC와 동일한 것으로 error를 찾아내는데 이용된다. 마지막에는 8bit에 End_Delimiter가 Manchester 인코더에 의해서 첨가되고 Manchester 디코더에 의해서 제거된다.

2.4 텔레그램

차량간 통신을 하기 위해 마스터는 슬레이브들(하나 또는 여러 슬레이브)에게 Master_Frame을 전송한다. 마스터 프레임은 받아 선택된 슬레이브는 이에 대한 응답으로 해당되는 Slave_Frame을 내보내며, 이때 Master_Frame과 Slave_Frame은 모두 브로드캐스팅을 하게된다. 이와 같은 하나의 통신과정(Master_Frame과 Slave_Frame)을 Telegram이라고 하고 [그림5]에 나타내었다.

텔레그램은 그 내용에 따라 세 가지로 구별이 된다.

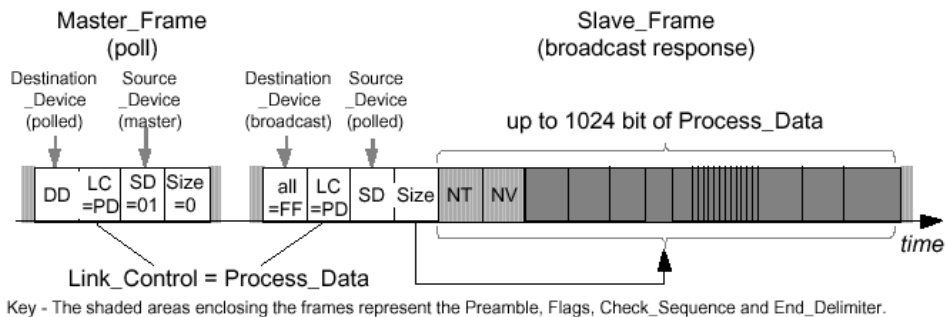
- Process_Data Telegrams
- Message_Data Telegrams
- Supervisory_Data Telegrams



[그림5] WTB Telegram

(1) Process_Data Telegram

마스터가 Process_Data 전송을 위하여 노드를 폴 하면, 폴 된 노드는 Slave_Frame을 브로드캐스트하여 Process_Data 의 전송이 이루어진다. 이와 같은 구조를 [그림6]에 나타내었다. WTB상에 있는 노드들은 Slave_Frame이 브로드캐스팅될 때 Process_Data를 전송 받아 각 노드에 있는 포트를 갱신하게 되며, 이중 원하는 데이터를 찾는다. 한 노드에서 보내는 Process_Data 프레임은 고정된 형식을 따른다. 만약 Master가 Process_Data를 보내고자 할 때에도 먼저 Master_Frame을 보낸 후 Slave_Frame을 보낸다. 이와 같은 것을 self_poll이라고 한다.

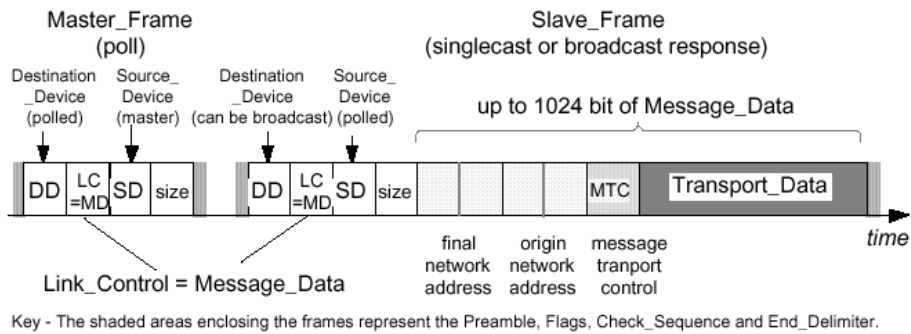


[그림6] WTB Process_Data Telegram

(2) Message_Data Telegram

Master에서 Message List를 근거하여 Message_Data를 위한 Master_Frame이 전송되면 폴 된 노드에 서는 메시지를 포함하고 폴 된 노드의 주소와 이 데이터를 받는 주소로 이루어진 Slave_Frame을 내 보 내게 된다. 이와 같은 Message_Data 텔레그램을 [그림7]에 나타내었다.

이 Slave_Frame은 한 노드(Destination_Node) 또는 브로드캐스팅되어 전체의 노드가 받게 되며, Message_Data는 Process_Data와 달리 그 길이가 고정되어 있지 않기에 여러 Slave_Frame을 받아야 하 나의 데이터가 되는 경우도 있다. 그리고 보낼 메시지가 없는 경우에는 크기를 0으로 내보낼 수도 있다. 만약 마스터가 Message_Data 을 보내고자 할 때는 Process_Date때와 동일하게 먼저 Master_Frame를 보내고 메시지를 포함하고 있는 해당 Slave_Frame을 내보낸다.

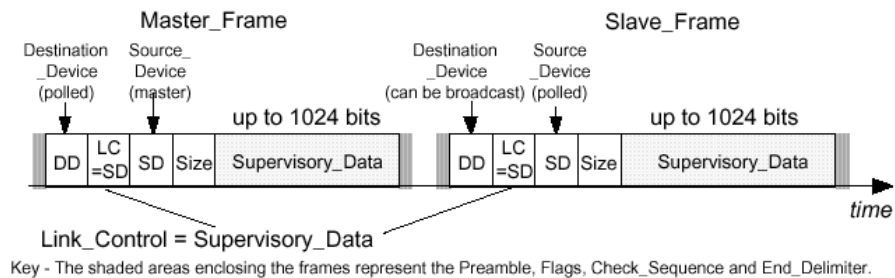


[그림7] WTB Message_Data Telegram

(3) Supervisory_Data Telegram

WTB상에서는 Process_Data와 Message_Data 프레임과 버스의 유지보수를 위한 Supervisory_Data의 전송이 진행되어진다.

이것도 역시 [그림8]과 같이 Master_Frame과 Slave_Frame으로 이루어진다. Master_Frame이 Destination_Device로 브로드캐스팅 될 수 있는데, 이 경우에는 Slave_Frame이 없고 단지 마스터가 Slave_Frame이 전송되어 소비될 시간동안인 time-out만큼 기다린 후 다음 텔레그램이 이루어지게 된다.



[그림8] Supervisory_Data Telegram

3. WTB Analyzer 구현에 관한 연구

철도차량과 같이 수많은 승객을 수송하는 시스템에서는 신뢰성을 중요시하기 때문에 내고장성 설계를 요구하고 있다. 메시지를 차량의 기기에 전달하는 동안에 전송로의 단락, 또는 이상현상이 발생되더라도 네트워크 전체에 영향을 크게 미치지 않도록 네트워크를 설계해야 한다. 열차의 추진장치 및 제동장치와 같이 실시간 통신이 보장되어야 하는 네트워크에서는 통신선과 hardware를 병행하는 기술도 지속적인 연구가 진행되어야 할 것이다. 또한 차량의 기기들이 중앙집중식에서 분산식으로 전환됨에 따라 운전시스템의 신뢰성과 안전성을 어떻게 확보해 갈 것인가 등을 여러 종류의 다양한 소프트웨어와 하드웨어 툴을 이용해 검증해야 할 것이다. 본 논문에서는 이와 같은 요구사항을 만족하기 위한 WTB 통신에 대하여 네트워크의 상태를 진단하고 감시할 수 있는 WTB Analyzer와 모니터링 프로그램을 제작하였고, 시험을 위하여 전동차 및 고속열차에 사용하는 WTB 통신 보드를 이용하였다.

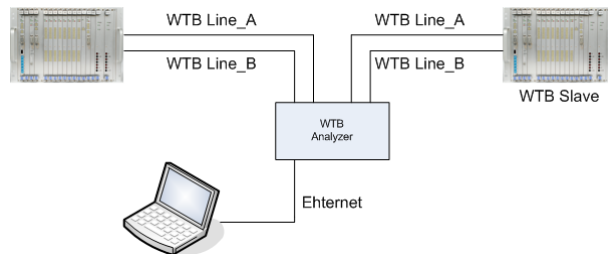
3.1 WTB Board

본 논문에서 사용한 WTB 통신 보드는 TCN통신 규약중 WTB통신을 만족하는 독립적인 보드로써, 전동차 및 고속열차에 장착되어 WTB 통신을 담당한다. 열차제어를 위한 CPU Board와의 인터페이스를 위하여 VMEbus 규격의 인터페이스를 지원한다.

3.2 WTB Analyzer

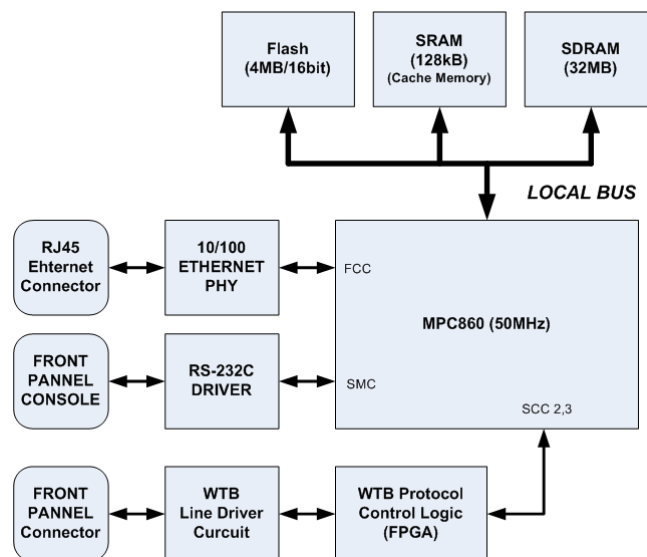
WTB Analyzer는 TCN 통신규약 중 WTB 상에 연결 되어지는 독립적인 디바이스로서, [그림10]과 같이 마스터 디바이스와 슬레이브 디바이스 사이에 연결되어 마스터 디바이스에서 슬레이브 디바이스로 전송되는 마스터 프레임과 마스터 프레임의 응답으로서 전송되어지는 슬레이브 프레임 또는 디바이스 상태정보를 분석하여 원격에 설치되어있는 PC환경의 모니터링 프로그램에서 TCP/IP 프로토콜을 통해 데이터를 전송하는 시스템이다.

WTB Analyzer의 운영구성은 [그림10]과 같다.



[그림10] WTB Analyzer 운영구성도

WTB Analyzer의 프로세서로는 Motorola사의 MPC860T-50MHz를 채택하였다. 운영체제(Real Time Operating System)로 VxWorks가 탑재되며, 사용자 응용 프로그램(User Application Program)과 시스템을 위해 필요한 각종 메모리를 제공한다. 운영체제와 사용자 응용 프로그램이 탑재되는 EPROM/FLASH와 SDRAM으로 구성되어 있는 시스템 메모리가 있다. FLASH 메모리는 2M x 16Bits 용량을 1개 사용한다. SDRAM은 4M x 32Bits 용량을 2개 사용하여 8M x 32Bits를 구성하며, 시스템용 SRAM은 128K x 8Bits를 1개 사용한다. 또한 모니터링을 위한 RS-232C Interface와 PC 환경의 모니터링 시스템과의 인터페이스를 위한 이더넷 포트를 지원한다. [그림11]은 WTB Analyzer의 하드웨어 블록도를 보여주고 있다.



[그림11] WTB Analyzer 하드웨어 블록도

WTB Analyzer Hardware는 [그림11]과 같이 WTB 통신 Data를 Receive할 수 있는 WTB Line

Driver Circuit와 Manchester Code를 TTL level의 신호로 Decode할 수 있는 FPGA 부, Receive한 Data를 처리하기 위한 CPU 부 및 UI Software와 Interface할 수 있는 Ethernet control 부로 구성된다.

(1) WTB Line Driver Circuit

WTB 통신 Node로써 End Node로 설정하거나 Intermediate Node로 설정하기 위해 Relay를 사용한다. 또한 WTB 통신 신호를 외부의 영향으로부터 Isolation 시키기 위해 Transformer를 사용한다. WTB 통신 Data를 Receive하기 위하여 MAX3088을 Receive IC로 사용하였다.

(2) FPGA 부 (Direction & Switch-over Logic & Manchester Decoder)

EPF6016-TI144를 사용하여 CPU(MPC860T)로부터 Control Signal을 Receive하여 이중화된 WTB 통신 중 각 Channel에 대한 Main과 Auxiliary를 Switch-over하고 열차내에서의 각 차량간 통신에서 Master에 대한 방향 설정을 담당한다. 또한 WTB 통신 Line으로부터 전송받은 Manchester Code를 NRZ신호로 Decoding하여 CPU(MPC860T)로 전달한다.

(3) CPU 부

CPU로는 32Bit 프로세서인 MPC860T가 사용되었다.

이 CPU는 일반적인 프로세서로서의 기능은 물론 보드를 제어하기위한 각종 통신용 포트와 CONTROLLER가 내장되어 있어 별도의 CPU와 통신용 CONTROLLER를 사용했을 때 보다 공간을 절약하며 이들을 Access하기위한 BUS 사이클이 최소화되어 보다 고속의 처리를 할 수 있다.

MPC860T는 외부와의 통신 및 모니터용으로 1개의 FEC(Fast Ethernet Controller)와 4개의 SCC 그리고 2개의 SMC를 가지고 있다.

FEC는 10/100Base-T Fast Ethernet Port로 UI software interface 용도로 사용하며 SMC는 RS-232C 통신용으로 사용하고 SCC는 WTB 통신용으로 사용한다.

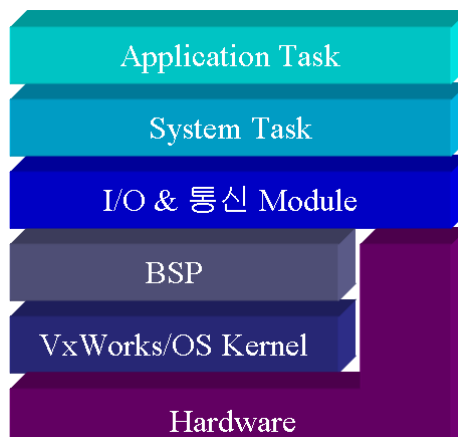
(4) Ethernet Control 부

WTB Analyzer는 UI software interface용으로 10/100Base-T로 TCP/IP Socket통신을 통하여 PC로 WTB 통신 Data를 전송한다.

(5) Software

WTB Analyzer Software는 [그림12]와 같이 Ethernet control 및 WTB 통신 Control을 위한 BSP를 기반으로 WTB 통신 Data 처리를 위한 application 및 UI Interface를 위한 TCP/IP application으로 이루어진다.

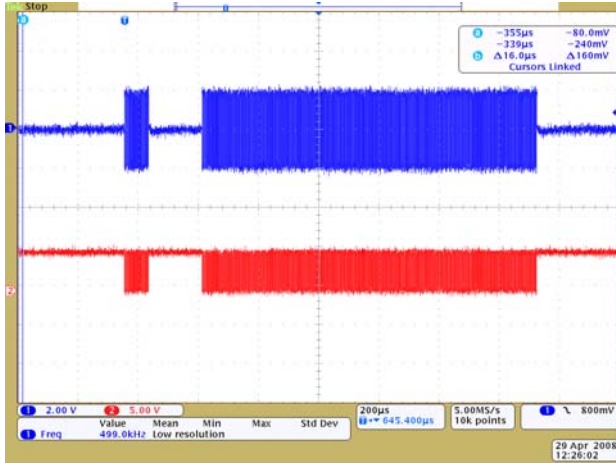
[그림12]는 WTB Analyzer의 프로토콜 스택 구성을 보여준다.



[그림12] WTB Analyzer 프로토콜 스택

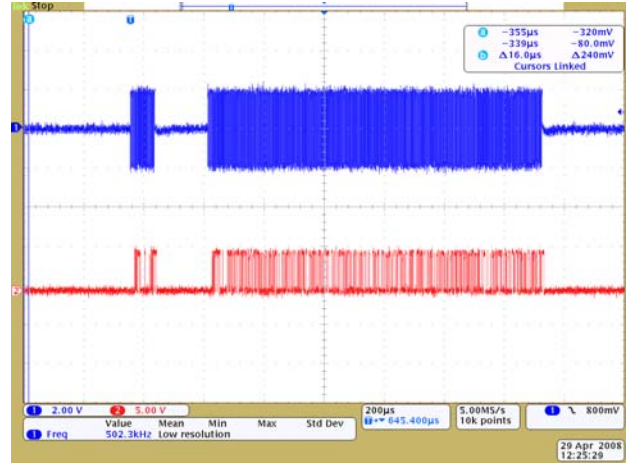
3.3 WTB 통신 Signal

아래의 [그림13]은 WTB Analyzer에서 입력된 WTB 통신 Data를 오실로스코프를 이용하여 capture한 image이다.



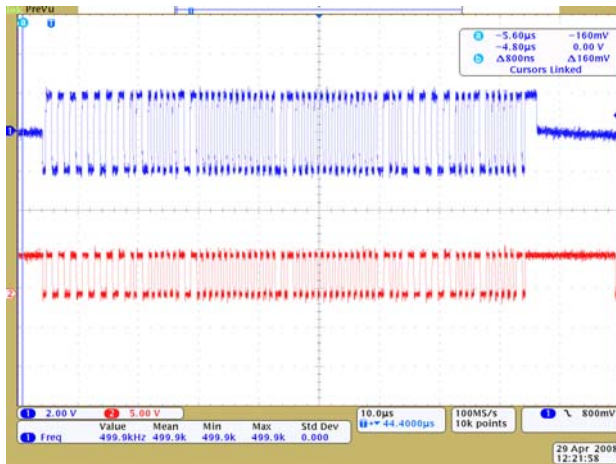
Process Data Telegram

- (a) 위 : WTB 통신 signal (RS-485 level)
- (b) 아래 : Receiver로 입력된 data signal
(TTL Level, Manchester code)



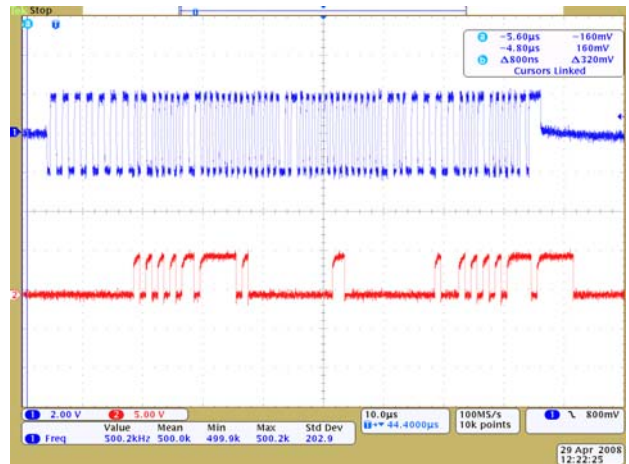
Process Data Telegram

- (a) 위 : WTB 통신 signal (RS-485 level)
- (b) 아래 : CPU로 입력되는 data signal
(TTL Level, NRZ code)



Master Frame

- (a) 위 : WTB 통신 signal (RS-485 level)
- (b) 아래 : Receiver로 입력된 data signal
(TTL Level, Manchester code)



Master Frame

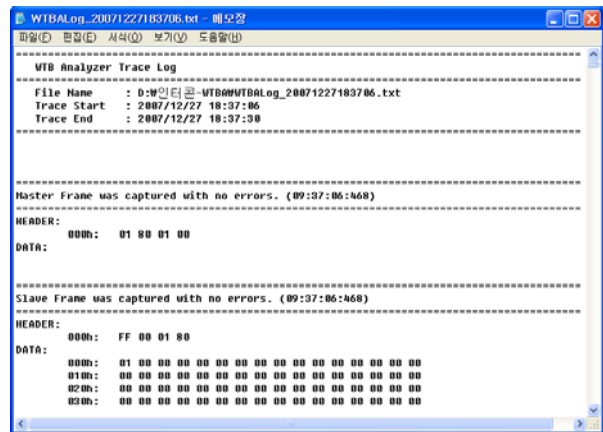
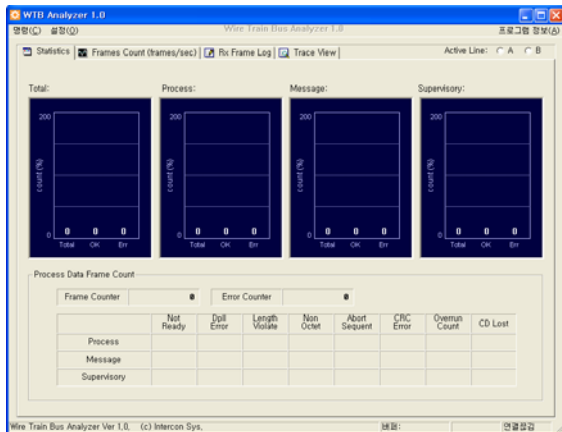
- (a) 위 : WTB 통신 signal (RS-485 level)
- (b) 아래 : CPU로 입력되는 data signal
(TTL Level, NRZ code)

[그림13] WTB Analyzer Data capture image

3.4 모니터링 프로그램.

모니터링 프로그램은 WTB Analyzer로부터 실시간으로 데이터를 받아 Total Frame, 정상 Frame, Error Frame을 나타내며, 실시간 Data를 확인할 수 있도록 Trace View 기능을 제공하며, 또한 Data Logging 기능을 통하여 통신 Data의 흐름을 파악할 수 있도록 한다.

이 모니터링 프로그램은 visual c++ 6.0을 이용하여 구현하였다.



[그림14] WTB Analyzer 모니터링 프로그램과 Logging된 Data

(1) Statistics window

Total Frame에 대한 정상 Frame과 Error Frame의 수, Process Data 통신에 대한 정상 Frame과 Error Frame의 수, Message Data 통신에 대한 정상 Frame과 Error Frame의 수, Supervisory Data 통신에 대한 정상 Frame과 Error Frame의 수를 나타내며 각 Error에 대한 분석을 위해 세부적인 Error의 내용 및 Error Frame count 기능을 구현하였다. 또한 현재 WTB 통신 Line의 상태를 나타낸다.

(2) Frame Count window

Process Data, Message Data, Supervisory Data가 전송되는 양을 파악하여 1초마다 각 Data의 수를 graph를 통하여 현시하도록 기능을 구현하였다..

(3) Log window

WTB 통신 Data의 내용을 단위시간(1초 ~)으로 설정하여 파일로 저장할 수 있도록 하는 기능을 구현하였다.

(4) Trace view window

WTB 통신 Data를 실시간으로 분석하고 감시하기 위한 창으로써 각 node와의 통신 상태를 확인하며 각 node에서 slave frame이 정상적으로 전송되고 있는지 확인하기 위한 기능을 제공한다. 또한 전체 node에 대한 통신 확인뿐만 아니라 선택된 node에 대해서만 통신 상태를 확인할 수 있는 기능을 구현하였다.

4. 결론

국내에서는 G7 고속전철기술개발사업을 통하여 한국형고속열차(HSR350x) 시제차 7량에 적용되기 시작한 TCN은 고속전철 및 전동차를 위한 전용의 네트워크로써 차량의 고속화와 안전성 및 유연성을 위하여 채택되는 분산제어 방식을 위한 가장 적합한 국제표준의 통신방식이다. 따라서 국내에서만 아니라 국제적으로도 점차 그 수요가 확대되고 있는 추세이나 국제적으로도 이러한 TCN을 검증하거나 시험할 수 있는 장비가 부족하며 수입을 원해도 구매할 수 없는 상황이다.

본 논문에서는 이러한 TCN 통신 중 WTB통신에 대한 Analyzer를 개발함으로써 점차 확대될 국내의 TCN 통신을 사용하는 열차에 대한 검수 및 시험을 원활하게 진행함으로써 열차의 장치들이 제어장치들과 원활히 통신이 가능하게 되어 보다 안정적인 차량 운행이 될 것으로 기대한다.

후 기

본 연구는 국토해양부의 차세대 고속철도기술개발사업의 일환으로 수행되었으며, 연구에 도움을 주신 관계자 여러분들의 지원에 감사드립니다.

[참고문헌]

1. IEC 61375-1 Standard, Train Communication Network. first edition 1999-09: Part (1) General Architecture (2) Real-time Protocol (3) Multifunction Vehicle Bus (4) Wire Train Bus (5) Train Network Management (6) Train Communication Conformance Testing, 1999
2. 최명호 외 2명, '전동차량용 필드버스 기술 동향', 제어계측 2003.
3. Communication-based train control systems, <http://www.tsd.org>
4. Kirmann, H., Zuber, P.A., "The IEC/IEEE train communication network", IEEE Micro, Volume: 21 Issue:2, March-April 2001
5. UIC 556 Standard, Information Transportation on the Train Bus 1999.05.01 v2.0
6. W.Richard Stevens, TCP/IP Illustrated, Volume 1, Addison-wesley publishing company, Inc. 1997
7. 한국형고속열차 연차보고서, 2001