DNP3.0 트래픽 모니터링 시스템

송병권*, 김세벽**, 정태의***, 김건웅****
*서경대학교 정보통신공학과, **서경대학교 전자컴퓨터공학과,
***서경대학교 컴퓨터과학과, ****목포해양대학교 해양전자통신공학부,
e-mail: bksong@skuniv.ac.kr

DNP3.0 Traffic Monitoring System

Byung-Kwon Song*, Sei-byuck Kim**, Tae-Eui Jeong***, Kun-Woong Kim****
*Information Communication Engineering, SeoKyeong University
**Electronic Computer Engineering, SeoKyeong University
***Computer Science, SeoKyeong University
****Division of Marine Electronic and Communication Engineering,
Mokpo National Maritime University

요 약

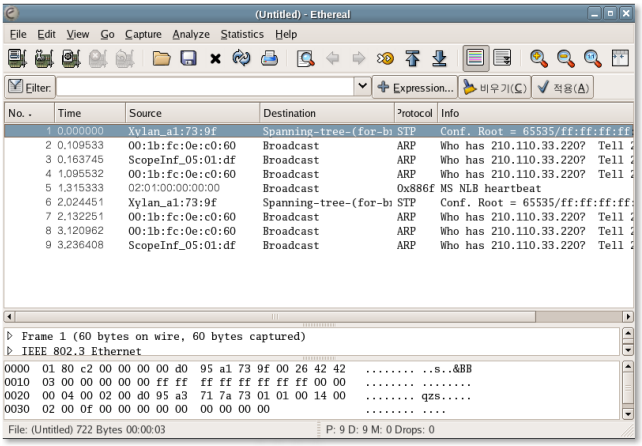
DNP3.0(Distributed Network Protocol 3.0) 프로토콜은 자동화 처리 시스템 사이에서 Master와 Slave의 개념을 적용한 프로토콜로써 현재 한전 SCADA 시스템의 표준 프로토콜로 선정되어 원방 감시 제어용으로 사용되고 있다. 이 DNP3.0 데이터를 RS-232C를 이용해서 전송 할 때, 각각의 DNP3.0 Layer인 DataLink Layer, Transport Layer 그리고 Application Layer의 분석 모듈을 설계하고 구현하였다.

1. 서론

DNP3.0(Distributed Network Protocol 3.0)은 원격 검침 및 배전 자동화에 사용되는 프로토콜로서 사실상의 표준인 Object-Oriented Protocol로써 기본적으로 Master, Slave 구조를 가지며 Event 발생 시 자발적 메시지인 Unsolicited Response를 취한다. OSI(Open System Interconnection) 7 Layer에서 변형된 EPA(Enhanced Performance Architecture)를 적용하여 Application Layer, Datalink Layer, Physical Layer로 구성되어 있다[1]. 이러한 DNP3.0 프로토콜은 현재 한전 SCADA(Supervisory Control and Data Acquisition) 시스템에 적용되어 Master Server와 RTU(Remote Terminal Unit)과의 통신에 사용되고 있다. 이와 같은 프로토콜 데이터가 전송되는 경우를 각 Layer별로 분석하고 모니터링 함으로써 현재 전송된 데이터의 상태를 알 수 있고, 이를 통해 긴급 상황 발생 시 현 상태에 대한 대처가 가능하다. 본 연구에서는 DNP3.0 프로토콜상의 Request 데이터가 전송되고 이에 대한 Response가 응답하는 상황과 Unsolicited 메시지가 발생하여 자발적인 응답을 하는 상황을 모두 모니터링 하여 해당하는 DNP3.0 메시지를 분석하고 로그파일을 남김으로써 현재 DNP3.0 데이터의 상태와 현재 DNP3.0 Master server로의 데이터 전달이 정확히 되었는가를 알아 볼 수 있다.

※ 본 논문은 서울시 산학연 지원 산업 신기술 개발 과제 연구비로 수행되었음.

2. 관련 연구

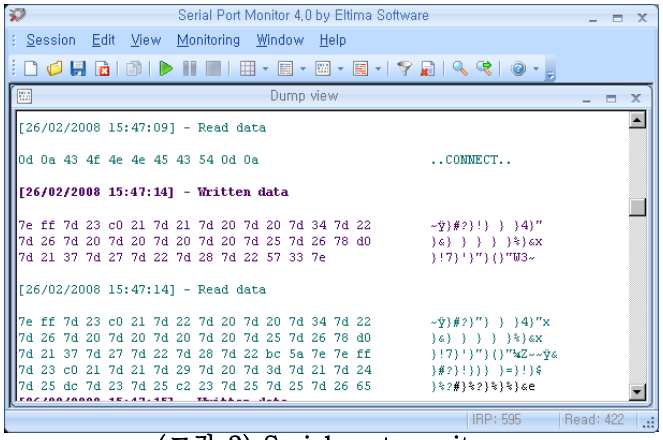


(그림 1) Ethereal - Network Protocol Analyzer

모니터링 프로그램이라고 하면 가장 먼저 떠오르는 게 (그림 1)의 Ethereal 프로그램이다. Ethereal 프로그램은 Ethernet 기반의 거의 모든 프로토콜 데이터를 캡처하고 모니터링 해주는 프로그램으로 해당 프로그램을 모티브로 삼았다.

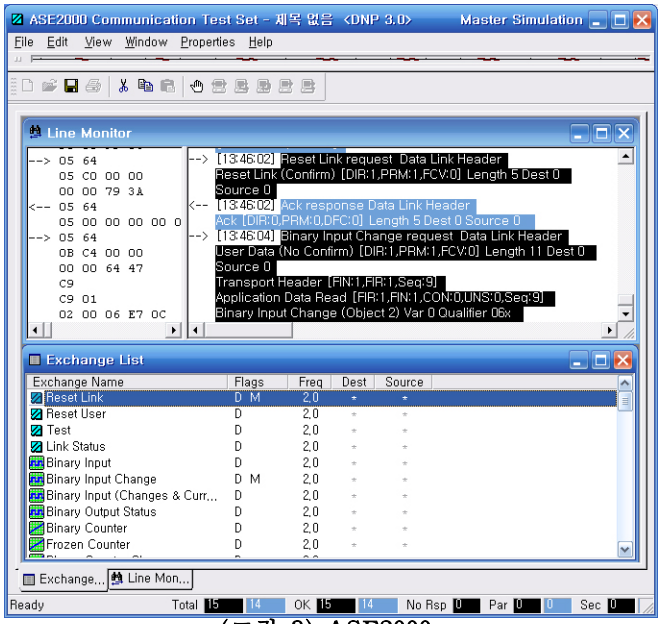
DNP3.0 데이터 전송 시 Serial Line을 통해 전송되는데이를 모니터링하기 위한 Serial Port Monitor 프로그램이 (그림 2)의 Serial Port Monitor 프로그램으로 RS-232C 상의 데이터를 캡처하여 모니터링 할 수 있다. 이 프로그램은 실제 전송되는 데이터를 ASCII 문자와 함께 모니터링 해줌으로 실제 Serial Line을 통해 전송되어져 오는 데이터를 분석하는데 사용하였다. (그림 2) 해당 Serial Line으로 Read와 Write를 했을 때의 데이터를 바이트코드와

ASCII 코드의 형태로 나타내준 그림이다.



(그림 2) Serial port monitor

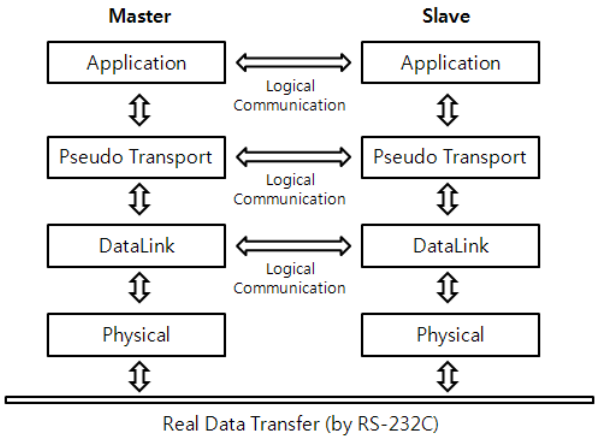
(그림 3)은 DNP 3.0 시뮬레이터인 ASE2000으로 이 프로그램은 마스터와 슬레이브로 구성된 DNP3.0 스택이 모두 구현된 시뮬레이터로서 한 대의 컴퓨터에서 마스터와 슬레이브 간 DNP3.0 데이터 송·수신을 시뮬레이션 해줌으로써 이를 바탕으로 DNP 3.0 분석 모듈을 구현하였다.



(그림 3) ASE2000

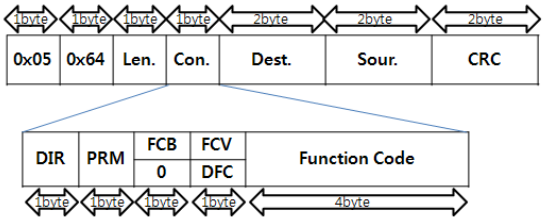
3. DNP 3.0 프로토콜

(그림 4)은 DNP 3.0은 OSI(Open System Interconnection) 7 Layer에서 변형된 EPA(Enhanced Performance Architecture)를 적용 Datalink Layer, Transport Layer, Application Layer로 나뉘어있다[5].



(그림 4) DNP3.0 Protocol Stack

3.1. DataLink Layer



(그림 5) DataLink Layer Format

DataLink layer는 DNP 메시지의 시작을 알리는 0x0564필드와 데이터의 길이를 나타내는 Length 필드, 현재 DNP 메시지 물리적 전송 방향(DIR), 현재 DNP메시지의 생성지(PRM), 간단한 신뢰 검증을 위한 FCB 비트, FCB비트를 활성화 할 것인지의 여부를 나타내는 FCV비트, 오버플로 방지를 위한 DFC 비트와 DataLink가 포함한 데이터가 어떤 의미인지를 나타내는 FC 필드로 구성된다. Destination Address는 목적지 주소를, Source Address는 원본메시지 주소를 Little Endian 방식으로 저장한다. CRC(Cyclic Redundancy Check) 필드는 Start 2byte, Length, Control, Dest. Sour.가 모두 포함된 채 계산된 값이다[1].

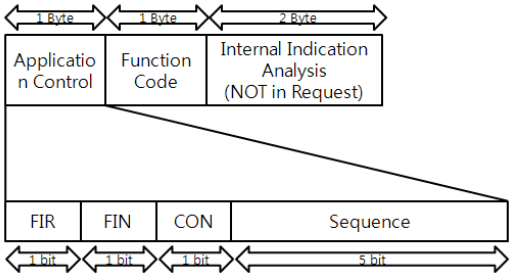
3.2. Transport Layer

(표 1) Transport Layer Format

FRAG		SEQ_NUM					
bit7	bit6	bit5	bit4	bit3	bit2	Bit1	Bit0
First	Final	0-64					

Transport Layer는 상위 Application Layer에서 249바이트 이상 전송을 원할 때 확장하기 위해 도입된 Layer이다. First와 Final 비트는 현재 프레임의 분절 상태를 알려주고 Seq_num은 순서번호를 나타낸다[1].

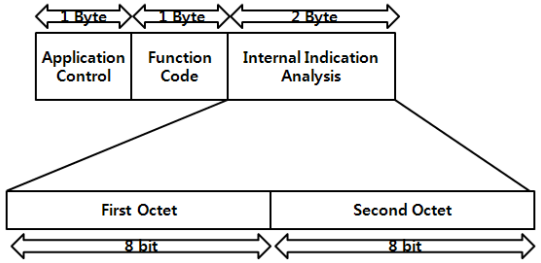
3.3. Application Layer



(그림 6)Application Layer Header

Application Layer는 실제 DNP 3.0데이터가 실려 있는 Layer로 AC(Application Control) 필드와 Function code 필드, 그리고 Response 메시지에만 포함되는 IIN(Internal Indication)필드가 있다. AC필드는 분절 데이터를 위한 First, Final과 Confirmation을 요구하는 Confirm 비트가 있고 메시지 순서번호를 위한 Sequence 필드로 구성되어 있다. DNP 프로토콜 스펙 상의 생성되는 메시지의 종류는 3가지 인대, 요청 메시지인 Request 메시지와 응답 메시지인 Response 메시지, 그리고 이벤트가 발생했을 때 자신의 이벤트 발생을 알리는 Unsolicited 메시지가 있다. Sequence 번호 중 16~31번까지가 Unsolicited 메시지를 위해 예약되어 있다[2].

FC(Function Code) 필드는 메시지가 어떤 역할을 하는지에 대한 코드 값을 나타낸다. Request일 경우와 Response일 경우 코드 값의 의미가 달라진다. Unsolicited 메시지일 경우 Response 메시지에서 FC 값을 130으로 주고 송신하게 된다.



(그림 7) Internal Indication Analysis in Response

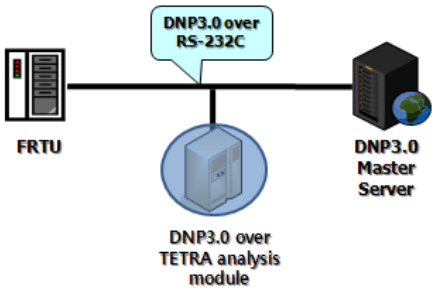
IIN 필드는 현재 request에 대한 수행을 못했거나 formatting 에러가 났거나 요구한 데이터가 사용 불가능일 경우에 그에 해당하는 비트가 설정되어 응답 발신 된다[1].

4. 분석 모듈 구현

4.1. 구현 환경

분석 모듈 구현을 위한 시스템과 네트워크 구성은 다음과 같다.

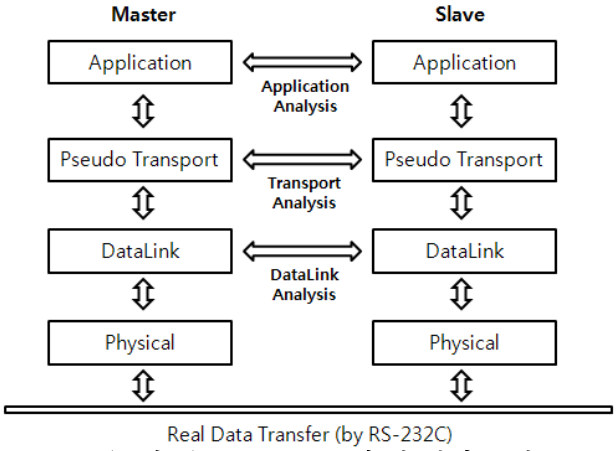
- FRTU 1대 (한전 KDN 대여)
- DNP 주장치 1대 (한전 KDN 대여)
- DNP 3.0 분석 모듈



(그림 8) 네트워크 구성도

(그림8)에서 보는 바와 같이 종단간의 통신은 DNP3.0 Master 서버와 FRTU 간에 DNP3.0 프로토콜을 RS-232C를 이용하고 중간에 RS-232C를 브릿지하여 전송되어지는 프레임을 캡처한다.

4.2. 모듈 구현



(그림 9) 프로토콜 스택 간 분석 구간

각 필드마다 고유의 의미를 분석하기 위해 각 바이트별로 비트 연산을 수행하여 의미하는 데이터를 추출해 낼 수 있다[3].

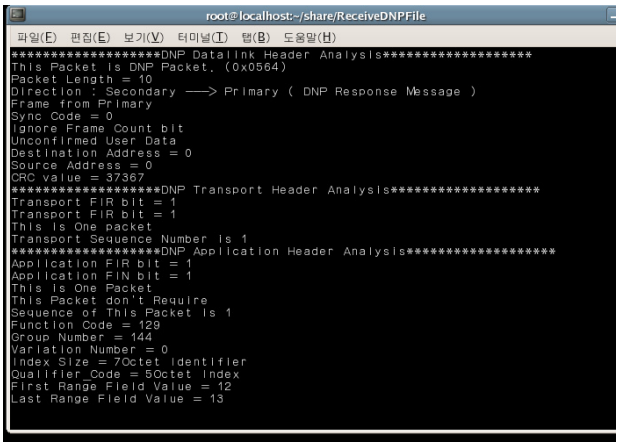
4.2.1. 모니터링 모듈 구조

분석 모듈을 C기반의 Pseudo code로 나타내었다.

```
//트래픽 모니터 메인 모듈
void DnpMonitor(){
        Create Buffer(data);
        if(read(data) == TRUE){
                analysis(data);
        }else wait for any data
}
//메시지 분석 모듈
void analysis(data){
        for( i = 0 ; i < Total_Size; i++ ){
                DataLink_Analysis(data);
```

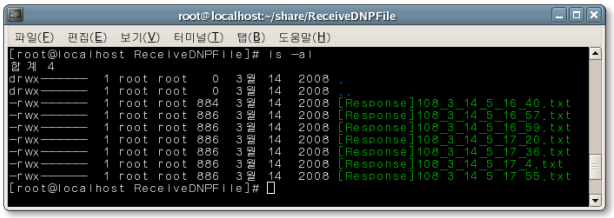
```
                Transport_Analysis(data);
                Application_Analysis(data);
        }
}
//Data Link Analysis
//Inner Protocol Header를 제거한 후 분석한다.
void DataLink_Analysis(Analysis_data){
        if(Start == 0x0564)
        //DataLink Analysis , print and log file Write
}

//Transport Layer Analysis
//DataLink Header를 제거한 Application Layer를 분석한다.
void Transport_Analysis(Analysis_data){
        //Transport Analysis , print and log file Write
}
//Application Layer Analysis
//Transport Header를 제거한 Application Layer를 분석한다.
void Application_Analysis(Analysis_data){
        //Application Layer Analysis , print and log file Write
}
```

6.3. Traffic monitoring



(그림 10) 분석 화면

(그림 10)는 실제 DNP 메시지를 SDS 전송형태로 전송 받았을 때 나오는 Analysis 화면이다. 각 Layer별로 순차적으로 분석하여 출력하였다.



(그림 11) Log 파일

마지막으로(그림 11)는 분석과 동시에 받거나 보낸 시각을 파일명으로 하여 (그림 10)의 내용을 저장하였다.

5. 결론 및 향후 과제

본 논문에서는 DNP 3.0 데이터 전송 간에 해당 데이터를 모니터링 하는 모듈을 제안, 구현하였다.

실제 SCADA 시스템에서 DNP 3.0 데이터 전송이 이루어질 주 서버와 RTU(Remote Transfer Unit)의 통신 모듈에 이 모니터링 모듈을 삽입하여 모니터링 기능을 추가하거나 임베디드형 분석기에 본 모듈을 장착, 네트워크상에 연결하여 추가 모니터링을 하면서, 이를 이용하여 에러 발생 시의 후 적극적인 조치를 할 수 있고, 현재 전송중인 DNP3.0 데이터 상태를 알 수 있다.

앞으로 DNP3.0 프로토콜만 분석하는 대에 그치지 않고, 다양한 프로토콜 분석 모듈을 삽입하여 배전자동화 시스템에서 사용되는 IEC61850, ModBus 같은 프로토콜을 모두 모니터링 할 수 있는 모듈로의 개발이 남아있고, 실제 이 모듈을 이용한 사용자의 편의성을 고려한 GUI (Graphic User Interface)로 업그레이드가 필요하다.

참고문헌

[1] DNP User Group, "Distributed Network Protocol DNP 3.0 BASIC 4 DOCUMENT SET"

[2] DNP User Group, "DNP3 Protocol Primer"

[3] DNPSec: Distributed Network Protocol Version 3 (DNP3) Security Framework