DPA(DNP3.0 Protocol Analyzer) 플랫폼 개발

송병권※, 이상훈○, 정태의※※, 김건웅※※※, 김진철※※※※, 김영억※※※※
서경대학교 정보통신공학과※, 서경대학교 전자컴퓨터공학과○
서경대학교 컴퓨터과학과※※, 목포해양대학교 해양전자통신공학부※※※
한전KDN(주) 정보통신연구그룹※※※※
{bksong, marufloor, tejeong}@skuniv.ac.kr, kgu@mmu.ac.kr {kjc, yekim}@kdn.com

Development of DPA(DNP3.0 Protocol Analyzer) Platform

Byeong-Kwon Song※, Sang-Hun Lee○, Tae-Eui Jeong※※, Gun-Woong Kim※※※,
Jin-Cheol Kim※※※※, Young-eok※※※※

Dept of Information and Communications Engineering, Seokyeong University※
Dept of Electronic and Computer Engineering, Seokyeong University○
Dept of Computer Science, Seokyeong University※※
Def of Marine Electronic and Communication Engineering,
Mokpo National Maritime University※※※
Information and Communication Research Group, Korea Electric Power Data Network Co.,Ltd. ※※※※

요 약

DNP3.0은 산업 분야에서 SCADA(Supervisory Control And Data Aquisition) system의 개방형 프로토콜로 사용되어지고 있다. 본 논문에서는 개발하거나 개발된 Module들 사이에서 송·수신되는 DNP3.0 PDU를 분석할 수 있는 기능을 제공하는 DPA(DNP3.0 Protocol Analyzer) Module을 설계 및 구현하였다. 해당 Master Station에서 Request 메시지를 생성하여 Outstation으로 전송하고 Outstation에서 수신된 Request 메시지를 분석하여 Response 메시지를 생성하여 Response한다. 또한 Master Station과 Outstation에서 DNP3.0 프로토콜을 사용하여 통신하는 중간에서 송·수신하는 메시지를 Monitoring한다.

1. 서론

현재 국내 및 해외 전력 산업에서 자동화 처리 및 하나의 Master Station이 다수의 Outstation을 제어하기 위한 종단 간 통신 프로토콜로 DNP3.0을 많이 사용한다.

DNP(Distributed Network Protocol)3.0은 종단 A와 B 지점 사이에서 Serial과 IP communications를 사용하여 Data를 전송한다. 이 프로토콜은 전력과 수력에서 주력으로 사용되고 있다[1].

한국전력공사에서는 현재 SCADA(Supervisory Control And Data Aquisition) system의 표준 프로토콜중 하나인 원방감시제어가 가능한 DNP3.0을 사용한다. 이는 DNP3.0이 SCADA Application을 위해 의도되어진 프로토콜이기 때문이다.

본 연구에서는 Master Station이 Request 메시지를 생성하여 Outstation으로 전송하고 Outstation에서 Master Station으로부터 수신한 Request에 대한 Response 메시지와 이벤트에 의해 생성된 Unsolicited 메시지를 생성하여 Master Station에게 응답하는 기능을 가지고, Master Station과 Outstation 간 DNP3.0 통신 프로토콜을 사용하고 있는 곳에서 통신 중인 송·수신 메시지를 모니터링 및 분석하여 보여주는 Monitoring 툴 기능을 가진 DNP3.0 Protocol 메시지를 생성하고 분석하는 Module을 제안한다.

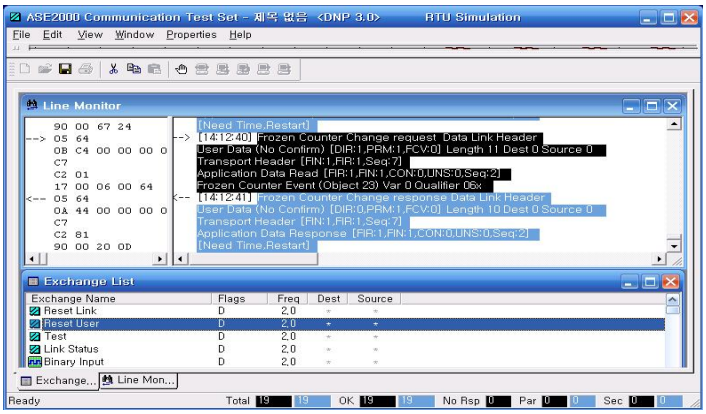
※ 본 논문은 한전KDN(주) 위탁연구 및 중기청 산학협력 연구비로 수행되었음.

2. 관련연구

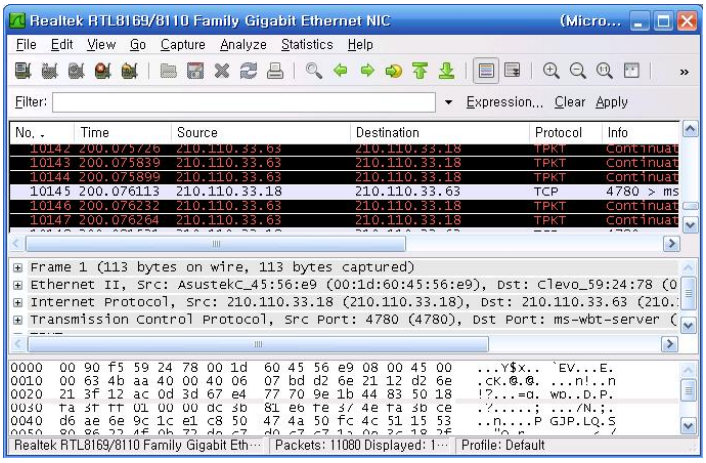
DNP3.0에 관련한 프로그램은 상용프로그램인 (그림 2)의 SUBNET Solutions社 의 ASE2000(Applied Systems Engineering)[2]와 프리웨어인 (그림 3)의 WIRESHARK[3]가 있다. ASE2000은 Master Station과 Outstation을 Simulation해주며 WIRESHARK은 종단 사이에서 모니터링을 한다.

ASE2000은 DNP3.0 메시지의 생성을 쉽게 Simulation하게 해줄 수 있는 장점이 있다. 반면에 ASE2000은 외부장치에 통신할 경우 PCMCIA to Serial PCI카드를 구입하

지 않는 이상 이 프로그램으로 통신할 수 없다. 이는 하드웨어적으로 구비되어있지 않으면 사용하는데 제한이 있는 단점을 가지고 있다. 본 논문에서는 이점을 개선하여 직접 Serial Cable을 통해서 DNP3.0 data를 전송이 되고, WIRESHARK와 같이 중간에서 Monitoring하는 역할을 가져 메시지 생성, 분석, 모니터링을 할 수 있는 프로그램을 제안한다.



(그림 2) ASE2000 실행화면



(그림 3) WIRESHARK 실행화면

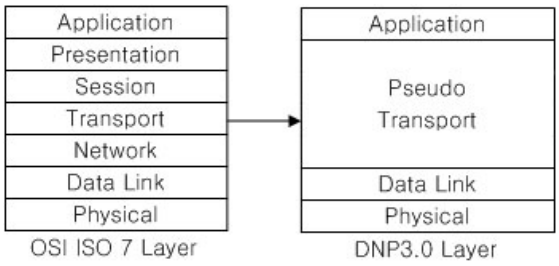
3. DNP(Distributed Network Protocol)3.0

DNP(Distributed Network Protocol)은 자동화 처리 시스템의 컴포넌트들 사이에서 Master와 Slave의 개념을 적용한 통신 프로토콜로 사용된다. 1990년 IEC875-5에 기초하여 DNP1과 DNP2가 개발되었다. 3년 뒤 1993년 DNP3 Basic 4 Document를 발표하고, DNP Users Group을 결성하였다. 현재 전기, 석유, 가스, 수력 등의 산업계의 표준으로써 개방형 프로토콜로 적용하여 사용되고 있다[4].

3.1 DNP3.0 Architecture

DNP3.0의 Protocol Model의 계층 구조는 ISO OSI 7 Layer에서 변형된 EPA(Enhanced Performance Architecture)를 적용하였다[2].

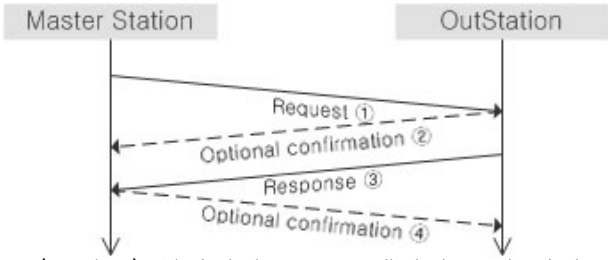
DNP3.0은 제일 아래 Physical Layer가 있으며, 차례대로 위에 Data Link Layer, Application Layer가 있다. 이 EPA구조는 Serial Port를 통해서 Data의 전송이 이루어지므로 ISO OSI 7 Layer의 Network Layer, Transport Layer, Session Layer, Presentation Layer의 3, 4, 5, 6 계층의 역할이 필요하지 않으므로 처리시간 단축 효과를 내기 위해 (그림 4)와 같이 포함되지 않았다.



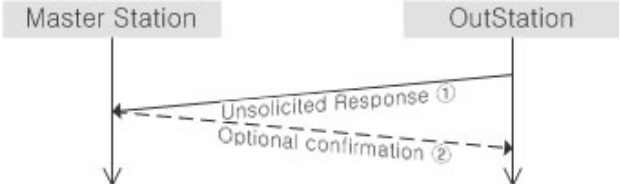
(그림 4) DNP3.0 Layer Architecture

3.2 DNP3.0 메시지 흐름

DNP3.0의 메시지의 흐름은 (그림 5)에서처럼 Master Station이 Outstation에게 Request를 보내서 Response를 받는 것이 일반적인 방법이며, Request나 Response시 Confirm bit를 1또는 0으로 설정하여 보내게 됨으로 선택적으로 확인응답인 Confirmation 메시지를 받을 수 있게 된다. 반면 Buffer Overflow와 같은 상태의 변화가 나타나게 되면 이벤트가 발생하게 되어서 Outstation은 (그림 6)과 같이 Unsolicited 메시지를 보내게 되며 Confirm bit 설정에 따라 확인응답을 받을 것에 대해서도 선택할 수 있다.



(그림 5) 일반적인 DNP3.0 메시지 교환 과정



(그림 6) 이벤트 발생 시 메시지 교환 과정

4. DNP3.0 메시지 생성 및 전달

DNP3.0 메시지의 생성은 보내고자 하는 메시지를 선택하면 Application Layer를 작성하고 아래 계층으로 보낸다. Pseudo Transport Layer에서는 상위 계층에서 받은 APDU(Application Protocol Data Unit)를 Data Link에서 한 번에 보낼 수 있을 만큼씩 Segment한 다음 TPDU(Transport Protocol Data Unit)를 만들어 아래 계층으로 보낸다. Data Link Layer는 상위 계층으로부터 받은 TPDU를 자신의 Data Link Layer Header를 Encapsulation하여 LPDU(Link Protocol Data Unit)를 전송하게 된다.

4.1 Application Layer Header

Application Layer는 DNP3.0의 데이터가 실리는 Layer는 (그림 7)처럼 Application Layer의 Header는 AC(Application Control) Field, FC(Function Code) Field 그리고 INN(Internal Indication) Field로 구성되어진다. INN Field는 Response 메시지에만 포함되는 Field이다.

AC Field는 분절 데이터를 위한 1bit의 First와 Final 그리고 Confirmation Response를 요구하는 Confirm bit가 있으며 메시지의 순서번호를 위한 5bit의 Sequence Field로 구성되어진다. 이 순서번호는 DNP3.0 메시지 종류 3가지 Request 메시지, Response 메시지, Unsolicited Response 메시지에서 Unsolicited Response 메시지가 16 ~ 31번 즉 $10000_{(2)}$ ~ $11111_{(2)}$의 비트사이의 번호로 예약되어 있다. 그리고 0 ~ 15번 즉 $00000_{(2)}$ ~ $11110_{(2)}$의 비트사이의 번호는 Master의 Request와 Slave의 Unsolicited Response를 제외한 모든 Response의 순서번호로 예약되어 사용되어진다[2].

FC Field는 메시지의 역할을 나타내며 INN Field는 Request에 대한 에러에 해당하는 bit를 세트해서 Response한다.

1 Byte		1 Byte		2 Bytes	
AC		FC		INN	
FIR	FIN	CON	SEQ	Fir OCTET	Sec OCTET
1bit	1bit	1bit	5bits	8bits	8 bits

(그림 7) Application Layer Header

4.2 Transport Layer Header

상위 Application Layer로부터 받은 APDU를 Link Layer에서 User Data를 16bytes 담을 수 있는 단점을 보안해 최대크기 249Bytes로 APDU를 Segment하여 Transport Layer Header (그림 8)와 같이 First와 Final Bit를 0 또는 1로 세트하여 세그먼트의 시작, 끝 또는 중간 세그먼트인지 알려주며 SEQ는 세그먼트의 순서번호를 나타낸다. 그리고 Transport Layer Header 뒤에 Segment된 Frame을 붙이게 된다.

1bit	1bit	6bit
First	Final	SEQ

(그림 8) Transport Layer Header

4.3 Data Link Layer Header

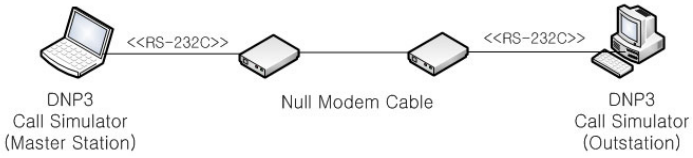
Data Link Layer는 종단간의 실질적인 통신을 나타내는 부분으로 아래 (그림 9)와 같이 구성되어있다. 0x0564의 DNP 메시지의 시작을 알리는 2Byte의 Start Field와 DNP 메시지의 길이를 나타내는 Length Field, 현재 데이터의 기능을 나타내는 1Byte의 Control Field, 2Bytes씩의 2개의 목적지 주소와 발신지 주소가 Little-Endian 방식으로 저장되는 Destination Address Field와 Source Address Field 그리고 마지막 Field는 CRC Field로 앞의 6Bytes의 CRC계산한 값이 저장된다.

1byte	1byte	1byte	1byte	2bytes	2bytes	2bytes
Start 0x05	Start 0x64	Length	CON	Dest Addr	Source Addr	CRC
DIR	PRM	FCB / 0	FCV / DFC	Function Code		
1bit	1bit	1bit	1bit	4bits		

(그림 9) Data Link Layer Header

4.3 DNP3.0 메시지 전달

생성된 DNP3.0메시지를 (그림 10)과 같이 네트워크 구성이 되어있다. Master Station에서 Serial Cable을 통해서 Request 메시지가 전달되게 된다. Outstation도 Serial Cable을 통해서 메시지를 수신하게 되고 다시 Response를 Master Station에게 보내게 된다. 가운데 Null Cable은 Rx와 Tx가 꼬여 Master Station이 Tx를 통해 데이터를 보내게 됨으로써 수신부인 Outstation은 Rx로 데이터가 수신된다.



(그림 10) 네트워크 구성도

4.3 DNP3.0 메시지 생성 모듈 구조

다음은 [표 1]에 메시지 생성 과 전송 모듈을 Pseudo code로 나타내었다.

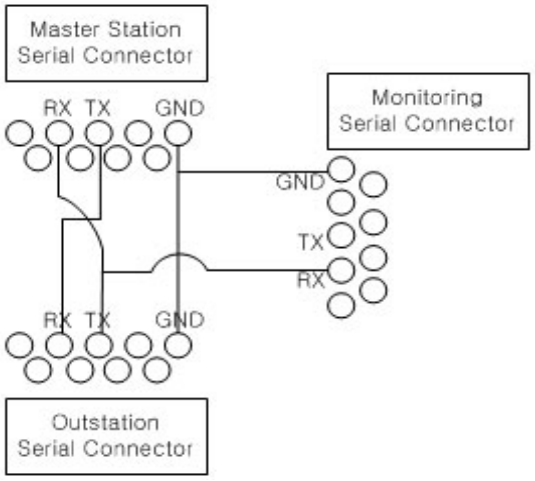
```
//Start Program
READ ObjectData
SET APDU, TPDU, LPDU
APDU = AddApplicationHeader(ObjectData)
TPDU = AddTranportHeader(APDU)
LPDU = AddDataLinkHeader(TPDU)
SendToSerial(LPDU)
//End Program

Function AddApplicationHeader(Data)
{
    SET AppHeader
    CONCATENATE APDU as AppHeader  plus Data
    RETURN APDU
}
Function AddTransportHeader(Data)
{
    SET TransHeader
    CONCATENATE TPDU as TransHeader plus Data
    RETURN TPDU
}
Function AddDataLinkHeader(Data)
{
    SET DLinkHeader
    CONCATENATE LPDU as DLinkHeader plus Data
    RETURN LPDU
}
void SendToSerial(LPDU)
{
    OPEN SerialPort
    SEND LPDU to SerialPort
}
```

[표 1] 메시지 생성 및 전송 모듈

5. DNP3.0 모니터링

종단 간 통신하는 중간에서 (그림 11)와 같이 RS232C를 Bridge하여 DNP3.0 메시지 전달과정을 모니터링 할 수 있다.

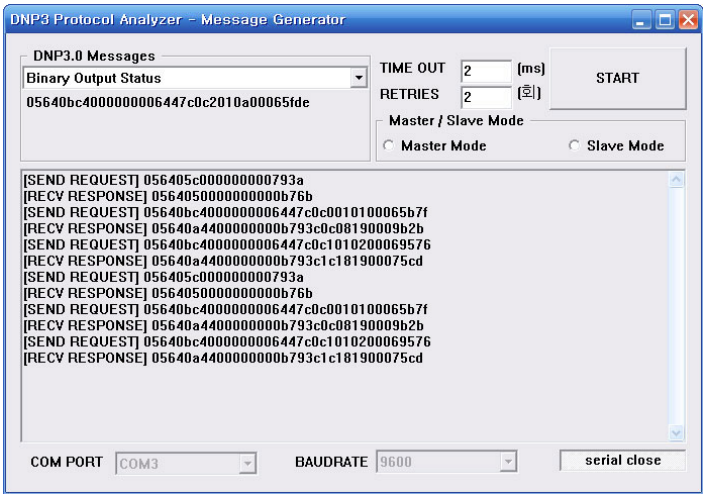


(그림 11) RS232C 결선도

Master Station의 Tx는 Outstation의 Rx와 연결이 되어있고, Master Station의 Rx는 Outstation의 Tx와 연결이 되어있고, Master Station 의 GND는 Outstation의 GND와 연결이 되어 있다. 이와 같은 연결선에서 Master Station의 Rx와 Outstation의 Tx와 연결된 선을 Monitoring Serial Connector의 Rx와 연결하면 Outstation에서 Response하는 메시지를 모니터링 할 수 있다. 이와 같은 그림은 (그림 11)과 같이 결선되어있다. 반대로 Master Station의 Tx와 Outstation의 Rx를 연결한 선에서 Monitoring Serial Connector의 Tx와 연결하면 Master Station의 Request 메시지를 모니터링 할 수 있다.

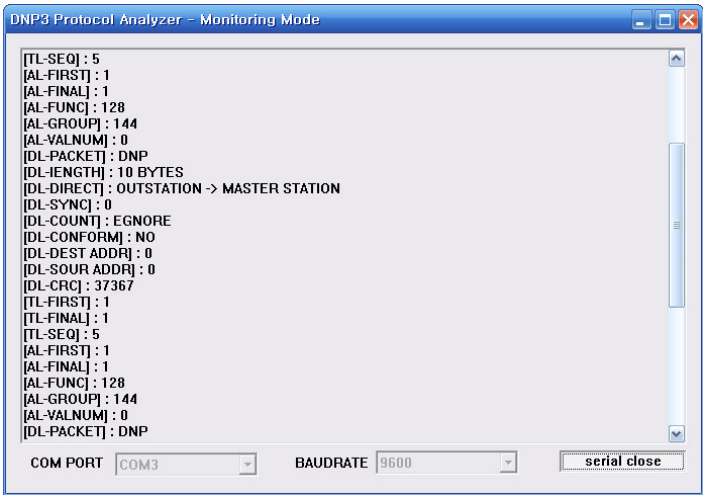
6. DPA(DNP3.0 Protocol Analyzer) 실행화면

(그림 12)는 프로그램에서 메시지 생성하여 주고받는 과정을 Capture 한 화면이다.



(그림 12) DNP3.0 메시지 생성 및 송·수신

(그림 13)은 프로그램에서 DNP3.0을 모니터링 하여 메시지를 분석한 과정을 Capture한 화면이다.



(그림 13) DNP3.0 메시지 Monitoring & Analysis

7. 결론 및 향후 과제

본 논문에서는 DNP3.0 메시지를 생성하고 통신 중인 DNP3.0 메시지를 Monitoring하고 분석하는 Module을 제안하고 그 토대로 구현하였다.

이 모듈은 DNP3.0 프로토콜을 사용하는 곳에서 메시지를 생성하여 SCADA system에서 Master와 Outstation이 없이 중간에 망을 테스트 한다거나 또는 종단 사이의 장치 개발 중에 메시지를 전달하는 장치의 역할 을 수행할 수 있다. Monitoring은 네트워크상에서 RS232C를 결선하여 DNP3.0 통신 중에 Data값을 확인하여 Data 통신 상태를 확인 할 수 있다. 또한 GUI환경으로 사용자가 손쉽게 접근하여 사용할 수 있다.

향후 연구로는 SCADA 시스템에 적용되는 다른 프로토콜 모듈을 연구·개발하는 것과 일부러 부정확한 메시지를 보내는 역할과 과부하를 줄 수 있는 기능을 추가하는 것이다.

참고문헌

[1] DNP Users Group, "DNP3 Primer", Revision A, Page 1, 20 March 2005

[2] SUBNET SOLUTIONS INC. "ASE2000 Communication Test Set Getting Started & User Guide"

[3] WIRESHARK, "http://www.wireshark.org/"

[4] DNP Users Group, "Distributed Network Protocol V3.00 Documentation"