
LabView를 이용한 PC기반의 교육용 전력선망 통신 제어시스템 구현

박태진* · 이운선** · 이상희*** · 이종희****

Implementation of PC-based education PLC(Power Line-network Communication) control system using Labview

Tae-Jin Park* · Un-Sun Lee** · Sang-Hee Lee*** · Jong-Hee Lee****

요 약

국내 에너지의 30% 이상이 조명부문에서 소비되고 있어서 그 소비량과 CO₂배출량은 점차적으로 늘어날 수밖에 없다. 이와 같은 상황에서 스마트그리드 기반의 전등 제어시스템 구축에는 중요한 의미가 있다. 결과적으로 오늘날 산업현장에서 사용되는 전력선망 통신모뎀과 응용 단말기들을 활용, 본 논문은 PC기반의 전력선망 통신 제어시스템을 구현해봄과 동시에 다양한 통신 테스트 장치 및 도구로 활용하는데 유용할 것이다. 실험을 위해서 사용되는 모뎀은 ZIC 프로토콜 기반의 플래넷 제품을 활용하고 응용프로그램은 랩뷰(LabView)를 사용한다. 그 결과는 주로 저속 전력선 통신에 의한 장치 제어에 초점이 맞추어져 있으며, 전력선 통신 실험을 위한 전력선 통신 테스트 장치로 각종 실험을 수행할 수 있다.

ABSTRACT

More than 30% of energy in Korea was spent in lighting part and then the consumption level and CO₂ emission will be increase from now. In this like situation, control-system construction for street-lamp of a smart grid-based that it has important meaning. Nowadays as a result, as an trying to utilize using Power-line-network communication modem and application a terminal in industrial fields, in this paper, we are want to implement the Power-line-network communication control system of PC-based. Therefore it will be use varied communication test-equipment and tools through the system. For experimentation, using the MODEM was able to use Planet production of based on the ZIC protocol and then the application program use the LabView. The result is primarily focused on equipments control by the PLC, and that we are able to perform each test with PLC equipment for experiment it.

키워드

PLC, PLC 모뎀, ZIC 프로토콜, 스마트 그리드, 전등제어시스템, 랩뷰

Key word

Power Line Communication, PLC Modem, ZIC protocol, Smart Grid, Lighting control system, LabView

* 정회원 : 신라대학교(주저자, csptj2@naver.com)
** 정회원 : 한국폴리텍IV대학 대전캠퍼스
*** 정회원 : 한국폴리텍VII대학 부산캠퍼스
**** 증신회원 : 신라대학교(교신저자)

접수일자 : 2012. 10. 25
심사완료일자 : 2012. 12. 09

Open Access <http://dx.doi.org/10.6109/jkiice.2013.17.3.730>

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

국내 에너지의 30%이상이 조명부문에 소비되고 있어서 그 소비량과 CO₂배출량은 점차적으로 늘어날 수밖에 없다. 이와 같은 상황에서 전력선망(PLC: Power Line-network communication)을 중심으로 하는 스마트 그리드 기반의 전등제어시스템 구축에는 중요한 의미가 있다. 이와 같은 전력선망 통신은 주로 저속 및 중속의 전기기기 제어에 활용되며, 어디에나 존재하는 것으로 차세대 유비쿼터스의 핵심기술이 될 것이다. 결과적으로 오늘날 산업현장에서 사용되는 전력선망통신 모델과 응용 단말기들을 활용함으로써 PC기반의 전력선망 통신제어시스템을 구현해 볼 수 있으며, 다양한 통신 테스트 장치 및 도구로 활용하는데 유용할 것이다[3,5].

본 논문에서 사용되는 모델은 ZIC 프로토콜 기반의 플래넷 제품을 활용하고 응용프로그램은 랩뷰(LabView)를 사용한다[6,7]. 랩뷰에서는 ZIC 프로토콜 주소의 구조와 할당에 따라 패킷별 기능을 정의하고 동작할 수 있도록 스위치와 릴레이를 초기화 한다. 또한, 시리얼 통신을 위해 지원하고 있는 인터페이스를 설정하고 각 송수신 장치별 제어 및 검증절차를 수행한다. 결과적으로는 송수신 데이터 해석에 따라 수행되는 통신 테스트 장치 및 도구로서 적용할 수 있게 된다.

이와 같은 ZIC 프로토콜은 기본적으로 양방향 통신을 위한 CSMA/CDCA 통신을 하고 있는데, 기존의 미국 및 유럽의 De-Facto 표준으로 사용되고 있는 X10 제품과도 호환을 이룰 수 있도록 만들어졌다[1]. 그 결과는 주로 저속 전력선망 통신에 의한 장치 제어에 초점이 맞추어져 있으며, 전력선망 통신 실험을 위한 테스트 장치로 각종 시험을 수행할 수 있다.

II. 관련 연구 분석 및 기술

2.1. ZIC 프로토콜

ZIC(ZBus I/O Controller)는 기존 규격 매립박스에 설치 가능하며, 백열등, 형광등, 할로젠등과 연결하여 리모콘으로 On/Off제어가 가능하다. 또한 리스폰스(Response) 패킷의 송신 여부 선택이 가능(ZIC 버튼 설정

시, ACK송신 여부 설정)하며, 최대 8대의 서로 다른 ZBus I/O의 릴레이 제어가 가능하다[3,4].

PCDC 시리얼 인터페이스는 보오율 4,800bps에서 19,200bps로 하고 데이터 비트 8비트, 스톱 비트 1비트, 패리티 비트 None, 흐름제어 None(only RX, TX GND)으로 구성한다. ZIC 프로토콜 기반의 시스템 구성도는 그림 1과 같다.

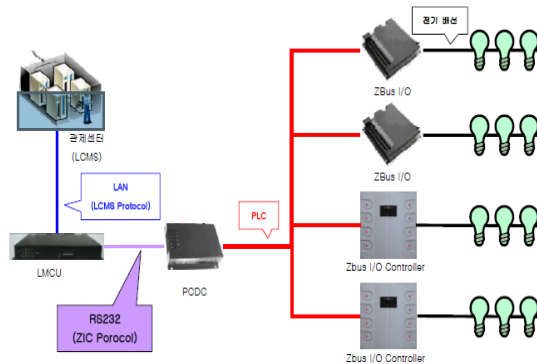


그림 1. ZIC프로토콜을 기반으로 하는 시스템 구성도
Fig. 1 ZIC protocol-based composition diagram of system

2.2. 패킷구조

패킷구조는 헤더, 메시지, 그리고 테일로 구성되며, 각 정의별 크기는 표1과 같고 패킷에 대한 코드를 설명은 표2와 같다.

표 1. ZIC 프로토콜의 패킷구조
Table. 1 Packet structure of ZIC protocol

De- fine	Header					Message			Tail	
	STX	D/A	S/A	LEN	P/T	D/T	CM D	Data	CRC 16	ETX
크기 [Byte]	1	2	2	1	1	1	1	N	2	1

Define	Code	Description	Remarks	
Header	STX	0x02	Start of packet	-
	D/A	-	Destination address	Address structure
	S/A	-	Source address	
	LEN	-	Length of STX-ETX	
	P/T	0x01	Request packet	Packet type
0x10		Response packet		
0x11		Notification packet		
Message	D/T	0x05	Device type	ZBus I/O ZIC
	CMD	-	Command	-
	Data	-	Reference of CMD	MAX 108Byte
Tail	CRC 16	-	Checksum of packet error	STX-Data
	ETX	0x03	End of packet	-

ZIC는 어드레스를 가지지 않고 기기번호를 가지며, 그 어드레스의 구조와 할당은 표2와 같고, 패킷별 명령에 대한 정의 및 설명은 표3과 같다.

표 2. ZIC 프로토콜 주소의 구조와 할당
Table. 2 Structure and assignment of ZIC protocol address

Device type	High (Device Type Address)	Low (Device Sequence Address)	Remark
LMCU	0x00	0x00	PCDC
PGW	0x01 ~ 0x02	0x01 ~ 0xFF	ZBus LCMS Protocol
SPGW	0x03 ~ 0x04	0x01 ~ 0xFF	ZBus LCMS Protocol
ZBus I/O	0x05 ~ 0x0C	0x01 ~ 0xFF	
LPCU	0x00	0x01 ~ 0x7F	ZBus LCMS Protocol

표 3. 패킷별 명령에 대한 정의 및 설명
Table. 3 Definition and description of each packet commands

CODE	Definition	Description
0x01	Request packet	PCDC(LMCU)→ZBus I/O
0x10	Response packet	PCDC(LMCU)←ZBus I/O (Request packet에 대한 응답)
0x11	Notification packet	PCDC(LMCU)←ZBus I/O (사용자 수동 제어 시 발생 이벤트)

CMD	Definition	Description
0x10	State	ZIC 또는 ZBus I/O의 상태
0x11	Relay Control	ZIC 또는 ZBus I/O의 제어
0xFA	B/S State	ZIC의 B/S(Button Setting) 상태
0xFB	Button Setting	ZIC의 B/S(Button Setting) 설정
0xFC	S/N State	ZIC의 S/N(Serial Number) 상태
0xFD	S/N Setting	ZIC의 S/N(Serial Number) 설정
0xFE	FAN State	FAN의 상태요청 및 응답
0xF5	R/S State	ZIC의 R/S(Repeater Setting) 상태
0xF6	R/S Setting	ZIC의 R/S(Repeater Setting) 설정

III. 시스템 설계 및 구현

3.1. ZIC기반의 전등제어시스템 설계

7-통신 ZIC 기반의 전등제어장치로서 시스템의 구성도는 그림 2와 같고 랩뷰에 의해서 프론트 패널이 구성된 화면은 그림 3과 같다.

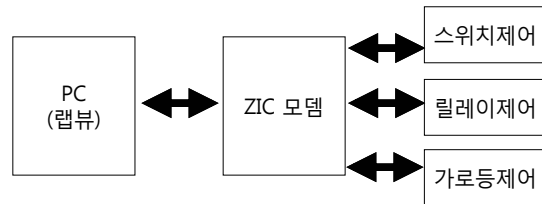


그림 2. ZIC기반의 제어시스템 구성도
Fig. 2 Composition diagram for ZIC-based control system

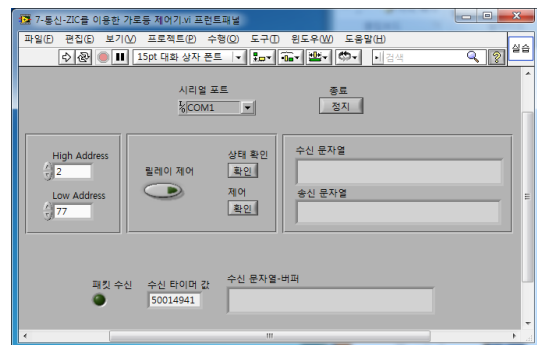
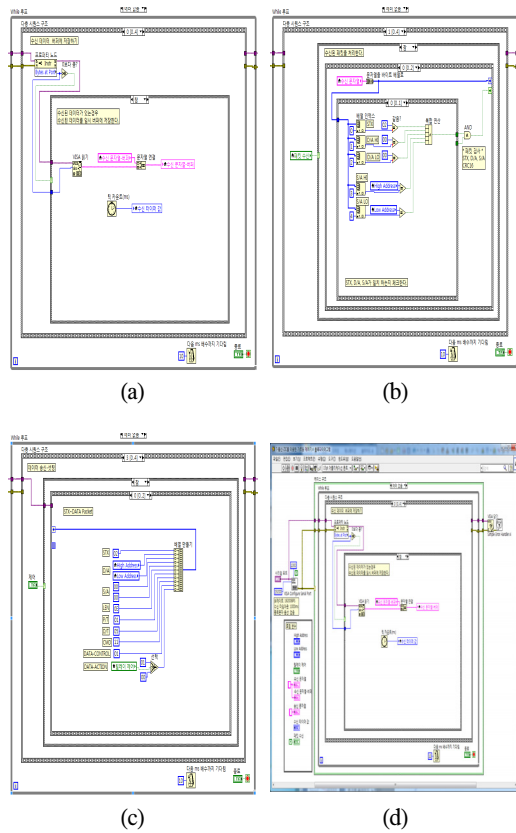


그림 3. 리퀘스트/리스폰스 데이터 필드 형식
Fig. 3 Format of data field for request/response

그림 3과 같이 프린트패널에서 컨트롤을 추가하고 컨트롤에 대한 프로퍼티 변경을 완료한 후, 블록 다이어그램에서 랩뷰의 소스코드 작업을 수행한다. 여기서 수신 문자열, 송신 문자열, 수신 문자열-버퍼의 인디케이터에는 16진수 패킷의 데이터로 나타낸다. 그림 4는 각 프레임 임별 완성된 블록 다이어그램이며, 수신 데이터의 버퍼 저장, 패킷 분석과 처리, 패킷의 전송 등 4개의 프레임은 가지는 다층 시퀀스 구조로 구성한다.



Define	Header				Message				Tail	
	STX	D/A	S/A	LEN	P/T	D/T	CMD	DATA	CRC16	ETX
크기	1 Byte	2 Byte	2 Byte	1 Byte	1 Byte	1 Byte	1 Byte	N Byte	2 Byte	1 Byte
[Byte]										

그림 4. 프레임별 완성된 블록 다이어그램
 (a) 프레임 0번 (b) 프레임 1번 (c) 프레임 2번
 (d) 프레임 3번 (e) 패킷 구조
 Fig. 4 Completed block diagram of all frames
 (a) Frame no.0 (b) Frame no.1 (c) Frame no.2
 (d) Frame no.3 (e) Packet structure

표4는 그림 4의 시각적 코드에서 텍스트 기반의 설계로 재표현 했으며, 프로세스 정의 및 코드를 나타낸다.

표 4. 텍스트 기반의 설계 : 프로세스 정의 및 코드
 Table. 4 Text-based design : Process definition and code

```

【클로별 변수 초기화 하기】
: 스위치 유닛 포트1, 스위치 유닛 포트2
: 릴레이 유닛 포트1, 릴레이 유닛 포트2
    
```

```

【시리얼 통신 프로세스 수행하기】
: 시리얼 인터페이스 초기화
: 수신 데이터를 받아 버퍼 저장 후, 패킷 분석
: 수신 데이터가 있는 경우와 없는 경우 체크
: if(데이터 수신 && 타임아웃 시간 초과) {
    패킷으로 인정하고 저장; }
: 입시 버퍼 값을 수신 데이터에 저장
: 수신 패킷 분석
: 수신 및 송신 패킷 데이터 읽기
: 수신 패킷 수신시 업데이트 상태를 열려준다.
-1(업데이트 있음)/0(없음)
: 패킷 송신
-OpMode: 0(상태 확인)/1(제어)
-UnitSel: 0(2구 스위치)/1(릴레이 제어기)
-Port: 0(오프)/1(온)
: 상태 확인
-DA 어드레스 값: 2구 스위치
-DA 어드레스 값: 릴레이 제어기
: 패킷 조립
: Generate CRC Data
    
```

```

【스위치 제어하기】
else if( OpMode==1 ){
    // DA, CMD: 2구 스위치
if( UnitSel==0 ){
    // Bit0: Relay 1 on, Bit1: Relay 2 on
CmdBuf = Port1 + (Port2 << 1); }
: DA, CMD: 릴레이 제어기
else if( UnitSel==1 ){
    // Bit0: Relay 1 on, Bit1: Relay 2 on
CmdBuf = Port1 + (Port2 << 1); }
: 패킷 조립
// Generate CRC Data
crc16 = PacketGetCrc16( PpacketTxData, 11 );
PpacketTxData[11] = crc16 >> 8; //High
PpacketTxData[12] = crc16 & 0x00FF; //Low
PpacketTxData[13] = 0x03; // ETX
PpacketTxLength = 14;// Packet Data Length
}
    
```

↓

【수신 패킷】
<pre> : 수신 패킷 분석 : 패킷 분석 후, 스위치/릴레이 상태의 업데이트 : Start of Packet: 0x02 if((PcaketRxData[0]==0x02) && (PcaketRxData[1]==0x00) && (PcaketRxData[2]==0x00)) {; } : Generate CRC Data : Received CRC Data </pre>
<pre> : CRC16의 검증 if(Crc16Cal==Crc16Rx){ : PT(Packet Type) 체크 - Packet Type: 0x10=Response, 0x11=Notification : DT(Device Type), CMD(Command) 체크 - Device Type: 0x05=ZBus I/O, ZIC - Command: 0x10=State Response, 0x11=Control Response if((PcaketRxData[7]==0x05)&& ((PcaketRxData[8]==0x10) (PcaketRxData[8]==0x11))) {; } } : SA High, SA Low 체크 if((PcaketRxData[3]==UnitSwitchIdHi) && (PcaketRxData[4]==UnitSwitchIdLo)) { </pre>
<pre> : 수신 패킷에서 제어 상태 값을 읽어온다: 2구 스위치 및 릴레이 제어기 // Bit0: Relay 1 on, Bit1: Relay 2 on UnitSwitchPortSet1=(PcaketRxData[9] & 0x01) ? 1:0; UnitSwitchPortSet2=(PcaketRxData[9] & 0x02) ? 1:0; } else if((PcaketRxData[3]==UnitRelayIdHi) && (PcaketRxData[4]==UnitRelayIdLo)){; } </pre>
<pre> : Get CRC16 while (nLen--){ nTemp = (nCrc >> 8) ^ *pbyBuf++; nCrc <<= 8; nQuick = nTemp ^ (nTemp >> 4); nCrc ^= nQuick; nQuick <<= 5; nCrc ^= nQuick; nQuick <<= 7; nCrc ^= nQuick; } </pre>

3.2. ZIC기반의 전등 제어시스템 구현
 3.2.1. 다층 시퀀스 구조 및 패킷동작
 다층 시퀀스 구조에서 시리얼 통신의 송수신 패킷을 작성하되, 4개의 프레임으로 표5와 같이 구성한다.

표 5. 다층시퀀스 구조
 Table. 5 Structure of multi-layer sequence

프레임	기능
0	수신되는 데이터를 버퍼에 저장
1	수신 패킷 분석 및 처리
2	상태 확인 버튼 클릭 시 해당 패킷 송신
3	제어 버튼 클릭 시 송신 패킷 전송
4	와이어 연결을 통한 송수신 값의 출력

제어시스템 구현을 위해서는 다층 시퀀스 구조의 0번 프레임부터 시작하되, 수신된 문자열에서 배열데이터로의 변환과 다음 프레임에서 사용이 가능하도록 연결해주는 동작이 필요하며, 프레임 2번은 프레임 1번에서 수신된 데이터를 분석하기 위해서 CRC방식의 동작이 요구된다. 또한 프레임 2,3번은 0번 프레임과 1번 프레임에서 계산된 패킷 데이터와 맨 끝의 ETX 바이트에 대한 배열을 통해서 패킷 완성한 후, 송신 되는 문자열을 디스플레이 되도록 한다.

끝으로 프레임 4번에서는 프레임의 왼쪽과 오른쪽에 와이어를 연결함으로써 값이 그대로 전달되도록 한다. 또한, 다층 시퀀스 구조의 0번 프레임에서부터 시작된 프로그램은 다층 시퀀스 구조의 3번까지 각 패킷별 헤더와 메시지, 테일에서 지정된 크기만큼 구성되고, 그 출력된 값은 복합연산 함수에 연결되어 처리된다. 상세한 다층 시퀀스 구조별 패킷동작은 표7과 같으며, 패킷 종류는 표6과 같다.

표 6. 패킷 종류
 Table. 6 Kind of packet

CODE	정의	설명
0x01	Request	PCDC(LMCU) → ZBUS I/O
0x10	Response	PCDC(LMCU) ← ZBUS I/O (Request 패킷에 대한 응답)
0x11	Notification	PCDC(LMCU) ← ZBUS I/O(사용자 수동 제어 시 발생 이벤트)

표 7. 다층 시퀀스 구조별 패킷동작
Table. 7 Packet action of multi-layer sequence structures

패킷정의	내용		
프레임 0	Header:STX	*입력된 상수값(0~4)은 '같음'함수로 비교연산 수행함 *이 출력값은 복합연산 함수에 연결되고 AND 논리연산을 수행함 *CRC를 포함한 패킷 검사	
	Header:D/A HI		
	Header:D/A LO		
	Header:S/A HI		
	Header:S/A LO		
프레임 1	Header:P/T	*CRC 계산과 CRC로 계산된 데이터의 비교와 논리함수 AND에 연결함 *배열인덱스 포트에 '6' 생성, 이 값으로 수신되는 패킷 처리	
	Message:DT	*배열인덱스로부터 7을 입력받고 0x05와 '같음'함수 연결 후, 출력값은 AND와 연결 및 검사수행	
	Message:CMD	*배열인덱스로부터 8을 입력받고 0x10와 '같음'함수 연결 후, 출력값은 AND와 연결 및 검사수행	
	Message:DATA	*배열인덱스로부터 9를 입력받고 AND와 연결 및 검사수행 *패킷의 비트 0,1에 해당된 릴레이가 할당되고, 이 비트 상태만 확인(스위치 1, 스위치 2)	
	Tail:CRC16	CRC 2Byte 계산	
프레임 2	Header:STX	0x2	배열 생성함수에 연결하고 출력은 시퀀스 로컬을 통해 다음 프레임으로 전달
	Header:D/A	High, Low	
	Header:S/A	00, 00	
	Header:LEN	0C	
	Header:P/T	01	
	Message:DT	05	
	Message:CMD	10	
	Message:DATA		
Tail:CRC16	*1번프레임에서 계산된 패킷데이터와 ETX를 배열 생성함수에 연결하여 패킷 완성. *출력 포트는 VISA 함수의 버퍼 쓰기 포트에 연결		
Tail:ETX			
프레임 3	Header:STX	0x2	배열 생성함수에 연결하고 출력은 시퀀스 로컬을 통해 다음 프레임으로 전달
	Header:D/A	High, Low	
	Header:S/A	00, 00	
	Header:LEN	0E	
	Header:P/T	01	
	Message:DT	05	
	Message:CMD	11	
	Message:Data-CONTROL	03	

MessageData-ACTION	*스위치 1,2를 불리언(0,1)에 연결 후, 출력은 곱하기 함수의 입력포트에 연결함. *다른 입력 포트에는 각각 상수를 생성하고 값을 입력 *‘곱하기’ 함수의 출력은 ‘더하기’ 함수의 입력포트에 연결함
Tail:ETX	

IV. ZIC기반 전등제어시스템의 실험

4.1. 실험장치 구성

ZIC기반의 전등제어시스템 구현에 따른 제작과 실험은 랩뷰 프로그램을 통해서 이루어지고, 실험에 사용된 전등 컨트롤러는 그림 5와 같다.

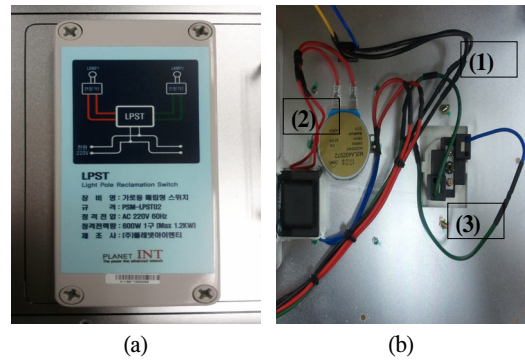


그림 5. 전등 컨트롤러 실습장치
(a) 전등 컨트롤러(앞) (b)컨트롤러와 연결된 장치 내부
Fig. 5 Equipment for an electronic lamp controller practice
(a) Street-lamp controller(Front) (b) Connected interior of equipment with controller

먼저, 모델에 전원케이블과 RS-232 케이블을 연결한 후, 전자식 스위치와 외부 전원케이블을 연결하고(좌측 2 포트가 전원입력/우측 2포트가 스위치 출력) 안정기와 할로젠램프를 연결한다. 컨트롤러에는 전압입력(1번선 2가닥) 포트와 전압출력(2번선 2가닥, 3번선 2가닥) 포트가 존재한다. 외부 AC 220V 전원과 검정색 2가닥(1)을 연결하고 녹색 2가닥(3)은 아울렛의 빨강색 2가닥(2)에 AC 모터를 연결한다. 모델과 RS-232 커넥터를 판넬에 연결하고 배선을 완성한다.

4.2. 데이터 전송과 시리얼 통신

먼저 데이터 전송 단위 테스트를 위해서 VISA 시리얼 포트 설정 함수를 사용, 시리얼 통신의 송수신 패키지가 다층 시퀀스 구조 안에서 처리되도록 하는 프로그램을 만든다. 통신환경을 위해서는 프런트패널에서 만든 시리얼 포트 컨트롤을 연결하고 보오 전송속도를 19200으로 하며, 타임아웃 포트는 1000을 입력한다. 다층시퀀스 구조의 0번에서부터 3번까지로 작성한다(그림 6-8).

4.2.1. 시퀀스 구조의 동작절차

시퀀스의 동작절차에는 다층 시퀀스 0에서부터 3까지 이루어지며, 복합연산 동작과 프로토콜 구성 및 배치 형태에 따라 전력망 통신이 구성되고 제어될 수 있다. 시퀀스별 각각의 절차와 그 구성 내용은 다음과 같다.

① 다층 시퀀스 구조#0

- 시리얼 포트에 수신된 데이터가 있는지 체크하기 위해서 그림 6과 같이 VISA 프로퍼티 노트라는 함수를 삽입하고, 해당 시리얼 포트에 수신된 데이터의 수를 반환한다. 반환 시 데이터의 개수가 '0'보다 큰지 여부를 체크할 수 있도록 하고, 데이터가 '0' 이상일 때의 함수 출력이 '참'이 되어 수신 데이터가 있음을 나타낸다.
- 데이터 패킷 수신 타임아웃 상수를 두어 종료되었는지 여부를 판단하도록 한다.

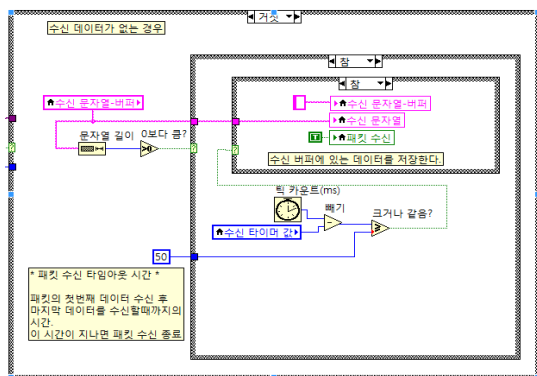
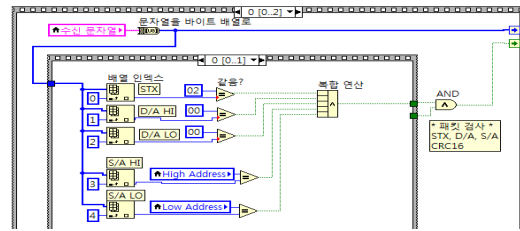


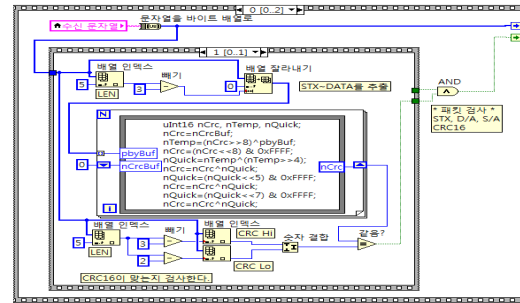
그림 6. 다층 시퀀스 구조 #0: 수신데이터 통신처리 절차
Fig. 6 Structure of multi-layer sequence #0: Communication processing procedure of received data

② 다층 시퀀스 구조#1

- 데이터 수신이 완료되어 패킷 수신 불리언이 '참' 값으로 저장된 경우, 패킷 수신 로컬 변수는 케이스 구조의 선택자 터미널에 연결이 되며, '참' 케이스에는 다층 시퀀스 로컬의 삽입과 코딩을 수행한다.

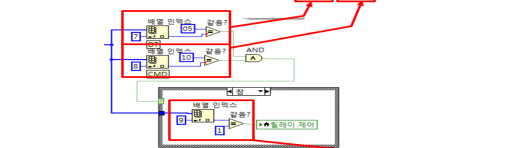


(a)



(b)

Define	STX	D/A	S/A	LEN	P/T	D/T	Message	DATA	CRC16	Tail
크기 [Byte]	1 Byte	2 Byte	2 Byte	1 Byte	1 Byte	1 Byte	1 Byte	N Byte	2 Byte	1 Byte



(c)

그림 7. 다층 시퀀스 구조 #1: 수신데이터 추출, 메시지 필드의 검사수행 및 릴레이 제어. (a) 데이터 패킷의 배열 인덱스 생성과 추출 (b) 수신데이터의 통신오류 검증을 위한 CRC탐색과 비교 (c) 메시지의 DT,CMD,DATA필드 검사수행 및 릴레이 제어

Fig. 7 Structure of multi-layer sequence #1: Extraction of received data, inspection of message field and relay packet control. (a) Creation of array index and extraction of data packet (b) CRC search and comparison for communication error inspection of received data (c) Inspection of DT,CMD and DATA field in message and relay control

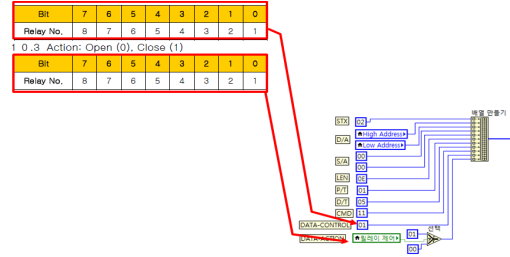
- 그림 7(a)와 같이 배열 인덱스 함수를 통해 각각 0~4까지의 상수를 입력할 수 있는데, 패킷의 STX, D/A, S/A를 ‘같음’ 함수로 비교한 후, 이 출력은 복합 연산 함수에 연결된다. 또한 수신데이터 패킷의 배열 인덱스를 생성, 옵셋에는 상수 ‘0’과 연결하여 START에서부터 DATA 필드까지를 추출한다.
- 그림 7(b)와 같이 시프트 레지스터를 추가하고, 패킷에 포함된 CRC 데이터를 찾아 ‘같음’ 함수로 계산된 CRC 데이터와 비교한 후, 배열인덱스 함수의 인덱스 포트에 상수 6을 생성, 패킷 타입의 필드를 선택한다.
- 배열 인덱스 함수의 인덱스에 각각 상수값 7, 8, 9를 입력한 후, 패킷에서 DT, CMD, DATA 필드를 선택하고 이 출력은 ‘같음’ 함수의 입력에 연결 후 출력값은 AND와 연결 및 검사 수행한다. ‘참’ 케이스의 패킷에 DATA에 해당하는 필드를 선택하여 릴레이 제어 컨트롤의 로컬 변수에 연결한다(그림7(c)). 계속해서 패킷의 DATA 필드의 비트 0에 해당하는 릴레이만 할당이 되고 이 비트 상태만을 확인한다.

③ 다층 시퀀스 구조#2

- 케이스 구조 안에는 다층 시퀀스 로컬이 삽입 되고, START에서부터 DATA 필드까지를 설정한 후, 16진수 상수 및 High Address와 Low Address의 로컬 변수는 각각 ‘배열 만들기’ 함수에 연결한다.
- 패킷의 STX에서부터 DATA 필드의 배열 데이터는 시퀀스 로컬을 통해서 0번 프레임으로부터 값이 전달이 되고 반복 루프의 왼쪽 프레임에 와이어로 연결, 입출력 포트를 추가한 후, 수식 노드에 프로그램을 입력한다.
- 패킷의 끝부분인 ETX 필드는 각각 ‘배열 만들기’ 함수에 연결하여 패킷을 완성한 후, 바이트 배열을 문자열로 함수에 연결한다. 출력 포트는 VISA 쓰기함수의 버퍼 쓰기포트에 연결함과 동시에 송신 문자열 컨트롤의 로컬 변수에 연결하여 송신되는 문자열을 디스플레이 되도록 한다.

④ 다층 시퀀스 구조#3

- 패킷 내 DATA에는 그림 8(b)와 같이 Control, Action 필드가 할당되어 있으며, 그림 8(a)의 릴레이 컨트롤 필드에는 상수 1을 입력하고 Action 필드에는 릴레이 제어 로컬 변수를 선택 함수의 s 포트에 연결한다. t 포트에는 상수 1을 입력하고 f 포트에는 상수 0을 입력한다. 선택함수의 출력은 배열 만들기 함수의 제일 하단에 입력 포트에 연결한다.



(a)

Relay Control Request

Define	Header				Message				Tail					
	STX	D/A		S/A		LEN	P/T	D/T	CMD	DATA	CRC16		ETX	
		High	Low	High	Low						B	C		
No.	0	1	2	3	4	5	6	7	8	9	A	B	C	D
CODE	0x02	-	-	0x00	0x00	0x0E	0x01	0x05	0x11	Control	Action	-	-	0x03

(b)

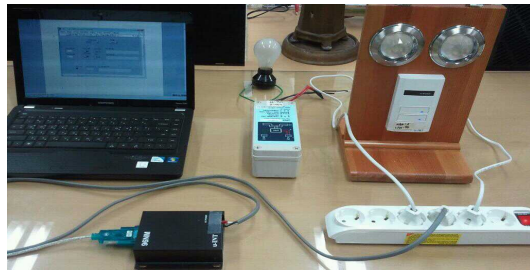
그림 8. 다층 시퀀스 구조 #3: 릴레이 제어 및 요청 패킷
(a) 릴레이 제어를 위한 포트 연결
(b) 릴레이 제어 요청 패킷

Fig. 8 Structure of multi-layer sequence #3: relay control and request packet
(a) Port connection for relay control
(b) Received packet for relay control

4.2.2. 실험결과

전력선망 기반의 2구 스위치 및 진구, 컨트롤러, 그리고 모뎀 및 RS-232 커넥터를 구성하고, 랩뷰에 의해서 설계되어진 코드에 따라 전등 스위치 및 전구를 제어하는 실험을 수행했다.

그림 9(a)는 랩뷰에서 2구스위치 제어를 위한 초기상태이고 그림 9(b)와 그림 9(c)는 좌측, 우측 램프를 제어한 상태를 보인다. 또한 그림 9(d)는 양측의 램프를 제어한 것이며, 그림 9(e)는 전구를 포함한 전등 전체를 제어하고 있음을 보였다. 즉, 각 프레임별 다층시퀀스에 구현된 프로토콜 구성 및 배치 방법, 그리고 코드절차에 의해서 스위치와 릴레이 장치가 제어된다. 이와 같은 결과를 통해서 전력선망 통신모뎀과 응용 단말기들을 활용, 다양한 통신의 기본 원리를 학습할 수 있으며, 이를 토대로 해서 스마트그리드 기반의 전력선망 통신을 직접 제어하고 수행해 볼 수 있는 테스트 장치 및 도구로서도 유용함을 알 수 있다.



(a)



(b)



(c)



(d)



(e)

그림 9. 전력선망 통신제어 실험

(a) 스위치제어(초기상태) (b) 스위치제어(On:Lamp#1)
 (c) 스위치제어(On:Lamp#2) (d) 스위치제어
 (On:Lamp#1,#2) (e) 스위치제어(On:Lamp#1,#2,#3)
 Fig. 9 Experience of PLC-based communication control
 (a) Switch control(initiation status) (b) Switch control
 (On:Lamp#1) (c) Switch control(On:Lamp#2) (d) Switch
 control(On:Lamp#1#2) (e) Switch control(On:Lamp#1#2#3)

V. 결 론

국내 에너지의 30%이상이 조명부문에서 소비되고 있어서 그 소비량과 CO₂배출량은 점차적으로 늘어날 수밖에 없다. 이와 같은 상황에서 전력선망을 중심으로 하는 스마트그리드의 기반의 전등제어시스템 구축에는 중요한 의미가 있다. 확장해서 스마트그리드 기반의 전력선 및 이기종간 통신장치에 대해서도 제어 및 통합관제가 가능하다.

따라서 본 논문에서는 오늘날 산업현장에서 사용되는 전력선망 통신모뎀과 응용 단말기들을 활용함으로써 PC기반의 전력선망 교육용 통신제어시스템을 구현하고 다양한 통신의 기본 원리를 학습할 수 있도록 하는 테스트 장치 및 도구로 활용한다. 랩뷰에서 설계되어지고 각 프레임별 코드화된 통신제어시스템으로 전등 스위치, 릴레이 제어를 수행하되, 전력선망 통신제어시스템 시퀀스의 동작절차가 다층 시퀀스에 따라 이루어지며, 복합연산 동작과 프로토콜 구성 및 배치형태에 따라 전력선망 통신이 구동되고 제어될 수 있다.

참고문헌

- [1] 박중현, "Z-Bus 기반 전력선 통신 시스템의 통신품질 개선", 충북대학교 석사학위 논문, 1-40쪽, 2005년
- [2] 박상환, "전력선 통신을 기반으로 한 통합 무선인식/무선센서노드 시스템의 설계 및 구현에 관한 연구", 창원대학교 박사학위 논문, 1-145쪽, 2010년 8월
- [3] 정경열, 김권태, 최대석, 박창대, 류길수, "PLC통신을 이용한 LED가로등 제어 및 모니터링 시스템 구현", 한국마린엔지니어링학회, 공동학술대회 논문집, 213-214쪽, 2010년 4월
- [4] 황민태, 최성수, 이원태, "전력선 홈 네트워크를 위한 신호 세기 기반의 자동 주소 할당 기", 한국멀티미디어학회논문지, 제11권, 제8호, 1072-108쪽, 2008년
- [5] 한재용, 이순흠, "초절전형 PLC2구 스위치 개발", 대한전자공학회, 전자공학회논문지-SD, 제44권, 제11호 (통권 제365호), 51-55쪽, 2007년 11월
- [6] 이규화; 김일주; 최준영; 이송근, "랩뷰를 이용한 실시간 전력계통 교육용 시뮬레이터 개발", 조명전기설비학회논문지, 제24권, 제6호, 177-182쪽, 2010년 6월
- [7] 김주은; 최남섭; 한병문, "LabVIEW를 이용한 전력전자 컨버터 교육 프로그램", 전력전자학회논문지, 제.17권, 제1호, 48-56쪽, 2012년 2월

저자소개



박태진(Tae-Jin Park)

1983년 동의대학교 물리학과 (이학사)
1995년 부경대학교 전산정보학과 (이학석사)

2008년 부경대학교 전자계산학과 (이학박사)
2000년~2003년 거제대학 전자계산학과, 초빙전임강사
2005년~2007년 마산대학 메카트로닉스과, 강의전담
2008년~현재 한국폴리텍VII대학 메카트로닉스학과, 겸임교수
2000년~현재 신라대학교 컴퓨터정보공학과 외래교수
2010년~2012년 (주)4D Zone, 기술이사
2012년~현재 (주)엠투엠코리아, 기술이사&기술고문
※관심분야: 영상처리, 신호처리, 임베디드시스템, 증강현실, USN응용시스템



이종희(Jong-Hee Lee)

1984년 경북대학교 전자공학과 전산공학 (공학석사)
1990년 경북대학교 전자공학과 전산공학전공 (공학박사)

1979년~1987년 경남정보대학 전자계산과, 교수
1988년~현재 신라대학교 컴퓨터공학과, 교수
※관심분야: 신경회로망 설계, 영상처리, 컴퓨터교육



이운선(Un-Sun Lee)

1997년 한국기술대학교 정보통신공학과 (공학사)
2008년 한국기술대학교 산업기술공학 (공학석사)

2009년 ~ 현재 부경대학교 첨단정보과학 및 정보기술 국제화 협동과정 (박사과정 수료)
2006년 ~ 현재 한국폴리텍IV대학, 전자과 조교수
※관심분야: 모바일프로그래밍, USN시스템, 자동제어시스템



이상희(Sang-Hee Lee)

2001년 진주산업대학교 기계공학과 (공학사)
2007년 동의대학교 기계공학과 (공학석사)

2000년~2007년 한국폴리텍VII 대학 기계공학과, 교수
2007년~현재 한국폴리텍VII대학 메카트로닉스학과, 교수
※관심분야: 기계제어, 시스템 자동제어