

Remote management of LonWorks based control system for air conditioner

Eun-Cheol Shin, Byoung-Wook Choi
unchol@empal.com, bwchoi@sunmoon.ac.kr

Abstract: In this paper, we describe implementation and test result for a LonWorks based control system in which air conditioner is handled by LonWorks node. The LonWorks based control system for handling air conditioner remotely is integration of control nodes because LonWorks is a communication protocol which is adopted as a standard protocol in building automation systems. For the remote management of distributed control system using Lonworks, we develop protocol converter based on SoC, where there is a Neuron Chip and Embedded Linux is running. The experiment result show that protocol converter using Embedded Linux is a flexible and effective way to build a distributed control system. And the LonWorks based control system for air conditioner is useful way to give interoperability.

Keywords: Remote monitoring and control, Distributed control, system, Embedded Linux, Protocol converter, LonWorks network

1. Introduction

In this paper, we describe implementation and test results for a LonWorks based control system in which air conditioner is handled by LonWorks node. The developed LonWorks control system consists of 6 nodes such a thermometer, 7-segment, pump controller, luxmeter, LEDs, and air conditioner. Originally, each manufacturer built their own system and provided all the intelligent devices within the system. Therefore, it has been historically difficult to interconnect digital controllers from different manufactures. And standardization requires an open control network. A market standard for communication would allow interoperability that would empower the market to increase in size and efficiency, that brought about the introduction of LonWorks protocol. The advantage of LonWorks based control system for handling air conditioner remotely is integration of control nodes because LonWorks is a communication protocol which is adopted as a standard protocol in building automation systems.[1] Therefore, the interoperability between air conditioners made in different manufacturers is achieved so that it is possible to handle build control system in a unified way.

For the remote management of distributed control system using LonWorks, we develop protocol converter based on SoC, where there is a Neuron Chip and Embedded Linux is running. It is a tendency which uses an Embedded Linux instead of Real-Time operating system and commercial embedded operating system. Linux is utilizes the network environment and therefore requires and appropriate operating system for handling protocol and an advanced development environment. The open source licensing, reliability, and broad hardware support are key reason for use of Embedded Linux in

embedded industry. The protocol converter integrates control networks of LonWorks devices to distributed control system through TCP/IP protocol. This implementation is a Web-based monitoring and control. For this purpose, embedded web server porting to Embedded Linux is performed. A Java Applet is designed in which Java socket program and GUI for users are performed. [2,3,4]

3. Development of Protocol Converter

For useful management for control network, a protocol converter is implemented. It consists of two CPUs. One is S3C4530A based on ARM7TDMI core that is used for Embedded Linux and controlling peripherals. The other is a Neuron chip supporting LonWorks. The first part of ARM-based SoC is designed to interface TCP/IP, RS232C and LonWorks protocols. The LonWorks suggested by Echelon is protocol supporting OSI 7-layer. The LonWorks is proper to control and monitor network devices just like LAN. The network data from LonWorks is collected in a Neuron chip embedded in the protocol converter. The data should be transferred to ARM CPU that is a main processor. Therefore, DPRAM of CY7C53150 chip of eth Cypress is also implemented to communicate between the Neuron chip and S3C4530A. And we ported Embedded Linux.[5]

3. LonWorks Control Devices

LonWorks is a communication protocol which is adopted as a standard protocol in building automation systems and a control network using Neuron chip. Neuron chip is a VLSI integrated with microprocessor from MAC layer of this LonTalk protocols until application layer. In this paper we implement node devices of LonWokrs using Neuron chip as distributed control network. The node

controller is developed with Lonworks hardware and interface circuit in order to implement various control objects. Table 1 shows specifications of LonWorks devices. In Fig 1, there is interface circuit handling peripherals with memory-mapped I/O or general purpose I/O. Therefore, the control nodes of Fig 2 are implement by using LonWorks control module while adding interface logics. In LonWorks, lots of transceivers are supported such as twisted pair, power line, and RF, etc. The control node is implemented with twisted pair of FTT-10. The communication rate is 76.8Kbps.

Table 1. Hardware specification of LonWorks devices

CPUs		CY7C53150
Memory	FLASH	32KB
	SRAM	32KB
Peripherals		11 GPIO
		Memory Mapped I/O

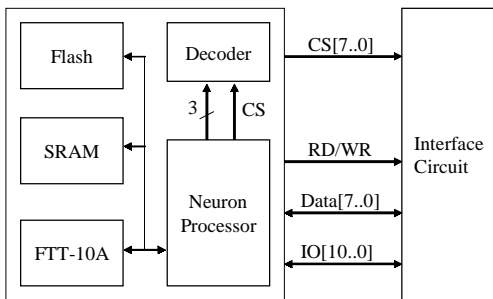


Fig 1. Block diagram of LonWorks devices

Fig 2. shows a developed control network of LonWorks, in which the control nodes are air conditioner control(AIRC), pump control(PUMP), temperature sensing(TEMP), luminous intensity sensor(LUX), and display nodes such as 7-segment(SEG), and Leds(LED).

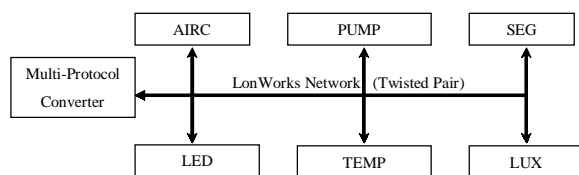


Fig 2. LonWorks network of control devices.

4. Software Structure of LonWorks Network

The program of LonWorks is done with Neuron C based on an event-driven program. The data type of network and I/O port are available. In order to communicate with LonWorks devices in the network, the network variable should be connected to each other and this is done by network binding. Binding is defined as logical connecting variable placed at distributed control nodes so that the communication between nodes is accomplished. In this paper, we use network variables and both variables declared at input and output nodes should be of the same data type.

The binding example is described in Fig 3.

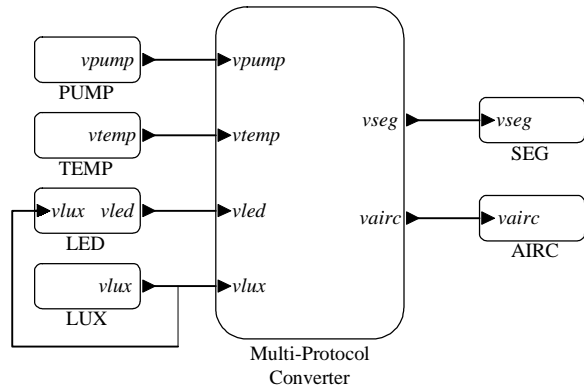


Fig 3. The structure of binding for network variables

The output variables in the protocol converter are defined for segment data and air-conditioner control. The output variable of vairc is a command to control air conditioner such as power on/off, target temperature, and the direction of wind blind. The other output variable of vseg is sued to send segment data to SEG node. The input network variables in protocol converter are vpump, vtemp, vlux, and vled to receive sensing or status data of nodes. The node referred to as TEMP and LUX are designed to measure the temperature and illumination with A/D converter. The data of LUX node is transferred to LED node to display the status, and it is sent to protocol converter.

5. Mechanism of Data Transmission

The communication between Java Applet and protocol converter is done through TCP/IP socket programming so that the command from Java Applet is transferred to LonWorks control network device or data gathered from LonWorks control devices is transferred to Java Applet. Therefore, the program in protocol converter is designed to gather data received from each protocol and convert it to socket data to utilize TCP/IP. The software of Server implemented at protocol converter is programmed specifically as Lon_driver.

Lon_driver is used to communicate LonWorks device, however, it is directly handling LonWorks devices but storing data or getting data to/from DPRAM. In Server function module, the decoding of data received form Java Applet through TCP/IP is achieved and the decoded data is sent to appropriate node. The operational order of Server program is as follows:

- creating socket server and waiting for the connection of Java Applet
- creating new thread and starting to communicate with Java Applet through socket when connection from Java Applet
- analyzing the data received from Java Applet and returns with corresponding response

- terminating thread and waiting for the connection of Java Applet again

We examine TCP/IP packet on how to transfer command to LonWorks network. The packet in Server program is defined as Table 2. The data field of addr is to describe the protocol type of control and the termination of the program. The cmd is used for command and the additional command information is included in dataL and dataH. In the case of sending packet, pump, temp, lux and led have the status of each devices.

Table 2. The packet format of TCP/IP

recv packet					
'\$'	addr	cmd	dataL	dataH	'#'

send packet					
'\$'	pump	temp	lux	led	'#'

The data transfer between two CPUs is perform in DPRAM with interrupt signal as shown in 4. In order to trigger interrupt signal, intention signal is written at 0x7f2 after writing data such as cmd, addr, dataL and dataH onto DPRAM. Then CY7C53150 chip read DPRAM according to interrupt signal. The received data is sent to the appropriate node through LonTalk protocol.

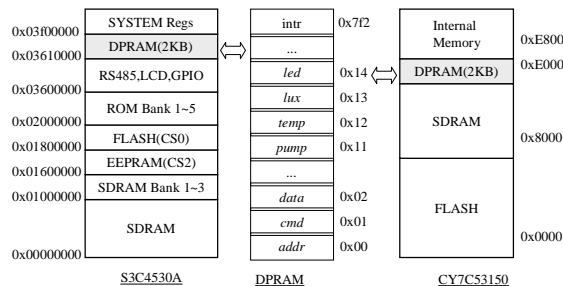


Fig 4. The data transfer mechanism between two CPUs

6. Implementation System

For remote management of LonWorks based control system for air conditioner, we used Java Applet and embedded Web-server on Embedded Linux.

To our knowledge, little architectural work has been done to show the suitability of Embedded Linux for remote management of LonWorks based control system in terms of protocol converter. In the protocol converter, Boa Web Server to be sued for Web service and Network Server for socket communication between Java Applet and Lon Device Server for LonWorks communication are performed. Java Applet is initially placed at the protocol converter and it is uploaded to client after connection Boa Web server from Web browser and it is running in client. The objective of Java Applet is to build GUI and perform socket communication between Java Applet and Network Server running in protocol

converter. Fig 6. shows a implemented GUI.[6]

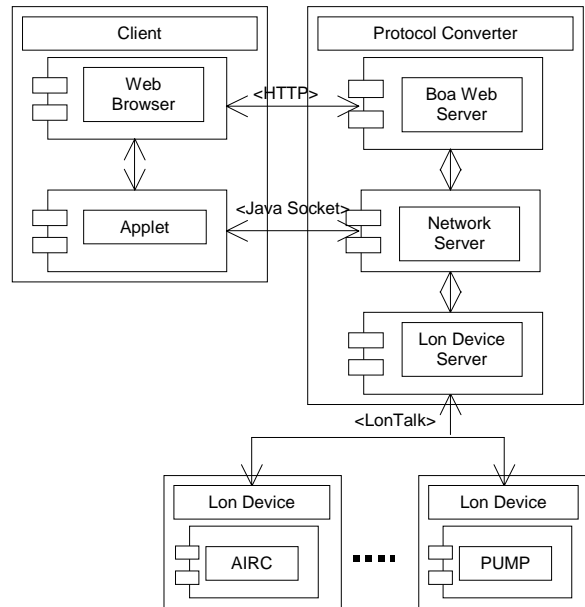


Fig 5. The structure of distributed control system with protocol converter



Fig 6. GUI implemented with the air conditioner in Java Applet

7. Implementation System

In this paper, we designed a remote management of LonWorks based control system for air conditioner. A protocol converter and the LonWorks devices are implemented.

The experiment result shows that protocol converter using Embedded Linux is a flexible and effective way to build a distributed control system. And the LonWorks based control system for air conditioner is useful way to give interoperability

REFERENCES

- [1] <http://www.echelon.com/products/lonworks/default.html>, "Introduction to the LonWorks system"
- [2] M.Furuya, H.Kato and T.Sekozawa, "Secure Web-Based Monitoring and Control System", IEEE 2000, pp.2443-2448.
- [3] H. Hiraishi, H. Ohwada, and F. Mizoguchi, "Web-Based communication and control for multiagent robots", Proc.of the 1998 IEEE/RSJ, Victoria, B.C., Canada, pp. 120-125, October, 1998
- [4] L.Q.Kong, J. Malee and T. Korte, "A simple Architecture for Real-Time Web Based Devices Control and Monitoring", ESCC 2001, Chicago, no.230, 2001
- [5] S3C4530A User's guide, Samsung 2001.
- [6] O.Storm, K. Svarstad, and E. J. AAS, "On the Utilization of Java Technology in Embedded Systems", Design Automation for Embedded Systems, Kluwer Academic Publishers, 8, pp. 87-106, 2003