Home Network Message Specification for White Goods and Its Applications

D.S. Kim*, G.Y.Cho*, W.H.Kwon*, Y.I.Kwan** and Y.H.Kim***
*ERC-ACI, School of Electrical Engineering, Seoul National University, Seoul, Korea
** DVN Team, DM Lab., LG Electronics Inc., Korea
***PLC Team, Korea Electric Research Institute

Abstract

In this paper, a home network message specification for white goods based on power line communication is proposed. It is designed for white goods such as air conditioners, refrigerators, washing machines, etc. The proposed home network message specification is composed of a virtual device service and a device-specific attribute. For practical implementation, an application software and two types of power line modules are used.

Keywords: Home network message specification, power line communication, white goods

1 Introduction

Easy installation and low cost are important factors in designing home network systems. Home appliances will be networked together, through some medium, for a home network system. Many home appliances, including white goods, are already connected to the power line. This makes the power line convenient for use as communication media for home appliances. As electrical outlets and the power lines are available throughout the house, power line communication (PLC) reduces cabling cost, as it requires little or no re-wiring, and is easy to use. In particular, in the case of white goods in the kitchen area, low cost and easy use are the most important factors in designing a home network system, since the manufacturer wants competitive prices, and various end-users need ease of use. Thus, PLC is a powerful tool that meets the requirements of a home network system for white goods.

In general, different vendors manufacture white goods used in the kitchen area. Therefore, a home-networking environment for white goods has a more heterogeneous platform. For white goods to be properly integrated into a home network system, a specific home network message specification (HNMS) for white goods is necessary for sharing a common interface and information with overall home network systems.

There are many standards that define the home network message specification [1–4]. The European Home Systems (EHS) Specification has defined a means for home devices to interact and communicate in and around the home [1]. The ECHONET (Energy Conservation & Health Care Network) specification also has defined a middleware and a full protocol stack [2]. However, in [1][2], the message specification does not cover white goods such as refrigerators, microwave ovens, and washing machines, etc.

As a message specification for white goods, the LonMarks organization has defined a message specification for refrigerators [3]. However, this message specification is designed for a display module only and does not consider the full message specification. In the case of the Consumer Electronic Bus (CEBus), the message specification contains only CEBus-compliant products such as light fittings, spring coolers, security, etc. However, it does not consider white goods. In [5–11], home network systems have been studied using the Lonworks and CEBus devices. However, these approaches did not consider the message specification for white goods. Therefore, the message specification for white goods based on PLC is necessary for a practical home network system. With this motivation, this paper proposes a simple message specification for white goods based on PLC. This paper is based on the basic ideas given in [12][13].

In this paper, the HNMS is designed for remote control and monitoring of all white goods in the home. The proposed HNMS is composed of a virtual device service and a device-specific attribute. Application software including a home server program and device emulators based on Universal Plug and Play (UPnP) middleware [14] and two types of PLC modules (high and low data rate) for white goods are implemented. Finally, its usefulness is proven through simulations and experiments.

This paper is organized as follows. In Section 2, the design consideration of the home network system based on PLC is described. Section 3 proposes an HNMS for white goods. Section 4 describes a test-bed implemented using PLC modules. Finally, the conclusions are presented in Section 5.

2 Design Considerations

Most off-the-shelf home automation modules are add-on PLC modules that sit between the power outlets and the ordinary devices to be remotely controlled. Therefore, the proposed home network system is

designed to build a network by using the existing AC power lines in the kitchen area. To realize a home network message specification for white goods based on PLC, the design should deal with the general characteristics of the home network system, as well as the specific characteristics of the PLC-based home network system. The general characteristics of the home network system are as follows:

- The HNMS should have a standardized networking structure.
- The application parts of the home network system need standardized data objects.
- For presentation, data interpretation is required.

The specific characteristics of the PLC-based home network system for white goods are as follows:

- The refrigerator is a candidate for the home server.
- Each device has different characteristics in terms of bandwidth, connectivity, security, interference, etc.
- On/off status of an add-on module may not be consistent with that of the device that plugs into it. When the device is broken, physically switched off, or unplugged, it is no longer controllable by the remote user.
- Energy management must work with the connected home appliances using PLC.

The home network system must be able to keep track of all changes in the entire network in order to support reliable operations, irrespective of the different characteristics. The add-on module should check the status of the devices in order to maintain consistency. For security in the home network system, the system uses a digital signature and encryption for a canonical solution. However, in this paper, the security problem and the noises of various white goods connected to the network were not considered.

3 Home Network Message Specification of White Goods

The proposed home network architecture considers the ability to fully integrate and manage all white goods within a home. It specifies a set of Application Program Interfaces (APIs), allowing the developer to develop the application for the home network system. The architecture of the home network system using the power line brings the power of client-server computing to home network tools.

The HNMS provides network services, it maintains the network database, and enables and coordinates multiple points of access to its services and data. HNMS services are grouped into two categories: confirmed and unconfirmed services. Confirmed services require confirmation when in services. A server to report a significant event to the client can invoke unconfirmed services.

Figure 1 shows the structure of the proposed home system based on PLC. The home server (refrigerator) has a Virtual Device Service (VDS) that is provided by a controlled device. The home server has all the VDSs supported by each device connected in the home network. By using the information in the related VDS, the home server can respond to the external requests of each device. The Device-Specific Attribute (DSA) provides device-specific functional aspects for white goods vendors.

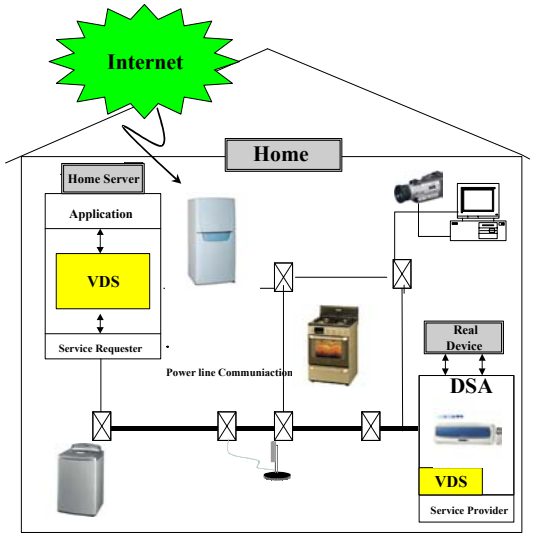


Fig.1 Structure of Home Network System based on PLC

The definitions of VDS and DSA are as follows:

- Virtual Device Service (VDS): an abstract representation of a specific set of resources and functionality at a real home device
- Device Specific Attribute (DSA): an attribute mapping of the VDS to the physical and device-specific functional aspects of the real home device.

3.1 Virtual Device Service (VDS)

The structure of the VDS model is shown in Fig. 2. Each module has several objects, including the related variables and functions. The function of each module is described as follows:

- **Information Object**: one or more data elements that refer to the device information elements that are referenced by a single name or description
- **Variable Access (VA) Object**: an abstract object that represents control variables of each

physical device

- **Database (DB) Object**: an abstract object that represents a subset of the capabilities of a VDS.
- **Action Service (AS) Object**: an abstract object representing a dynamic element that most closely corresponds to an execution thread
- **Event Object**: an abstract object that keep track of the status of the physical device

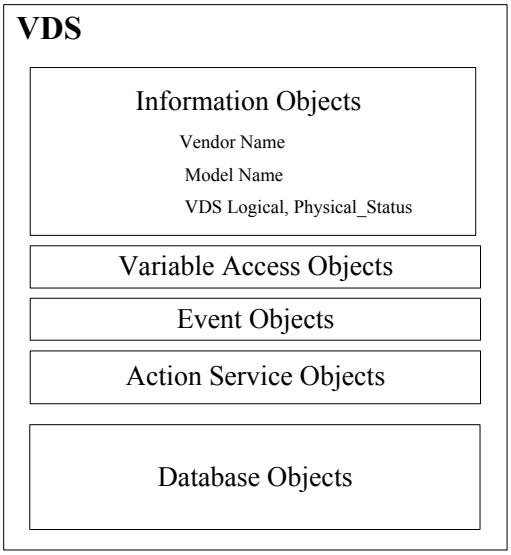


Fig. 2 HNMS VDS Model

The syntax notation of a VDS Information object is defined as follows. It is composed of the product information and the physical and logical status of the installed device.

```
typedef struct  VDS_info {
      Visible_string          vendor_name;
      Visible_string          model_name;
      Logical_status          vds_logical_status;
      Physical_status         vds_physical_status;
      Resource_list           list_of_action_service;
      Resource_list           list_of_data_base;
      Resource_list           list_of_variable_access_lists;
 }VDS_info;

typedef enum Logical_status{

      LS_State_changes_allowed,
      LS_No_state_changes_allowed,
      LS_Limited_services_permit,
   } Logical_status;
   typedef enum Physical_status{
     PS_Operational,
      PS_Partially_operational,
      PS_Inoperable,
      PS_Needs_PowerOn;
  } Physical_status;
```

For an example, the vendor and model name are a character string that identifies the vendor of the system that supports VDS. This information may be used by the client application to identify the remote controller.

3.2 Message Specification of White Goods

The VA objects for each appliance are described in Tables 1–3. For the function profiles of white goods, the products of seven major companies were investigated. Table 1 shows the variable access objects of an air conditioner. It is composed of variable name, data type and type description.

Table 1 VA Objects of Air Conditioner

Name	Type	Type Description	R/W
AC_Status	Enum	enum {idle, cold_air, clean_air, hot_air}	R
AC_Room Temperature	Int		R
AC_Target TemperatureSet	Int		W
AC_Target Temperature	Int		R
AC_AirFource	Enum	enum{low, medium, high}	R
AC_AirFourceSet	Enum	enum{low, medium, high}	W
AC_AirUDDirection	Int	int(-90~+90)	R
AC_AirUDDirection Set	Int	int(-90~+90)	W
AC_AirLFDirection	struct	struct { bool auto; int Lfdegree; }	R
AC_AirLFDirection Set	struct	struct { bool auto; int Lfdegree; }	W
AC_Reservation TimeSet	Time		W
.....	.....	.....	….

The VA object includes control variables of each device detailed by the developer. It defines services and functions for monitoring and operating the home appliances. Table 2 shows the VA object of a microwave oven. The VA object may be used to represent each variable of the microwave oven for monitoring and control. Table 3 describes the VA object of a washing machine. As with the microwave oven, the variable object may be used to control and monitor the washing machine. The VA object also supports function variables for the end user.

The Event Object provides services using three sub-objects. The objects are defined as the Event Condition object, Event Action object, and Event Enrollment Object. Each of these objects models a specific aspect of the state information with the management of events.

Table 2 VA Object of Microwave Oven

Name	Type	Type Description	R/W
MO_Power	Enum	On/Off	R/W
MO_Status	Enum	Run, Stop	R/W
MO_Door	Enum	open, close	R/W
MO_Temperature	Float		R
MO_TemperatureSet	Float		W
MO_Start	Boolean	On	W
MO_Protection	Enum	On/Off	R/W
MO_Warming	Enum	On/Off	R/W
MO_WarmingTemperature	Float	On/Off	R/W
MO_EndingAlarm	Enum	On/Off	R/W
MO_EconomicPower	Enum	On/Off	R/W
MO_Menu_Select	Enum	warm, cook, grill	W
.....	.....	.....	.....

The DB and AS objects for the appliances are composed of structure names and elements, and have a similar syntax notation for all appliances. The following syntax notations are examples of the DB and AS objects for a microwave oven:

```
Typedef struct  DB_info {
   Visible_string          data_base_name;
    DB_State               state;

}DB_info;
 typedef struct MO_DB_Data {
      kindof_MO_Select                Select;
      unsigned int           Temperature;
      int                      Humidity;
      time              Time;
}MO_DB_Data;


typedef enum kindof_MO_ {
                           Warming,
                           Cook,
                           grill
                } kindof_MO_Select;
```

The syntax notation of the AS object is described as follows:

```
typedef struct  AS_info {
   Visible_string          action_service_name;
} AS_info
```

The following syntax notations are examples of DB objects for a washing machine. All white goods have a similar structure and syntax notation in their DB objects. Therefore, these objects for other white goods are omitted in this paper.

```
Typedef struct WM_Method_DB_Data
kindof_WM_Material   Material;
float                                Detergent;
float                                Bleach;
float                                FabricSoftener;
int                                  WashingTime;
int                                  RisingNumber;
time                                 DryingTime;
float                                WaterHeight;
float                                WaterTemperature;

 WM_Method_DBData;

typedef enum kindof_WM_Material
                           very_soft,
                           soft,
                           hard,
                           very_hard
                kindof_WM_Material ;
```

Table 3 VA Object of Washing Machine

Name	Type	Type Description	R/W
WM_State	Enum	Enum {idle, washing, rinsing, drying, pumping}	R
WM_MaterialSet	Enum	Enum {very_soft, soft, hard, very_hard}	W
WM_Material	Enum	Enum {very_soft, soft, hard, very_hard}	R
WM_WaterSupplySet	Struct	Struct {bool Hot_Water; //on, off Bool Cold_Water; //on, off}	W
WM_WaterSupply	Struct	Struct {bool Hot_Water; //on, off Bool Cold_Water; //on, off}	R
WM_WaterHeight	Float		R
WM_WashingMethodSet	Enum	Enum {auto, manual}	W
WM_Reservation	Struct	Struct {bool Reservation; time Time; WM_method method;}	R
.....	.....	.....	.....

4 Implementation

In this section, the home server program, emulators of white goods, and applied PLC modules are described. In the case of the PLC module, commercial chip sets and a modem are used for testing the developed message specification and its application. In this paper, the noises of the various household devices and office equipment connected to the network were not considered.

4.1 Home Server using UPnP Middleware

In this paper, Extensible Markup Language (XML)-based home network middleware UPnP is applied to the developed test-bed using HNMS function profiles. UPnP enables the browser to be extended to control devices, and because UPnP devices are controlled with explicit protocols, the browser must somehow learn to talk to the UPnP devices. This learning process is driven entirely from the device itself and is accomplished by uploading an XML document that fully describes the capabilities of the device.

Fig. 3 shows the main home browser developed for the home network system. The XML-based browsing scheme in the simulation consists of five steps. The procedure for the browsing scheme in the developed home network system is shown in Fig. 4.

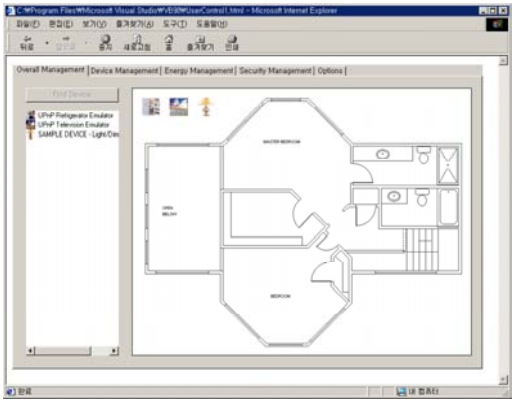
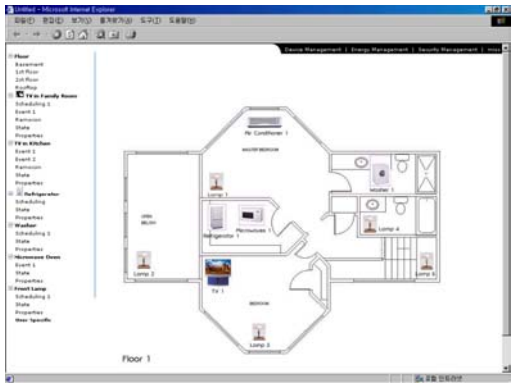


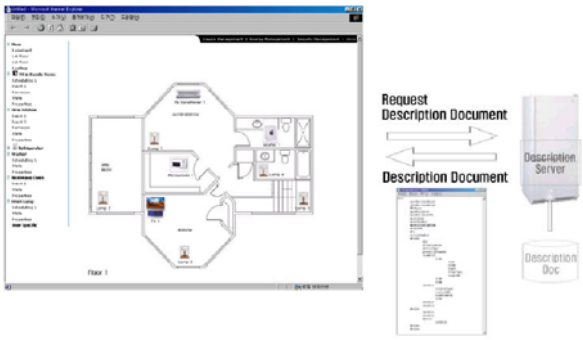
Fig. 3 Main Home Server

Each device can transfer bulk data (e.g., files) or control data streams from any device on the network to any other device on the network. The Control Point sends a discovery request to a newly plugged-in device and the device responds in Step 1. When a device is plugged into the network, the device must automatically configure itself. The device then announces its presence to the other devices already on the network using a simple discovery protocol based on the Internet Hyper Text Transfer Protocol (HTTP), and is immediately ready to share its services with any device that requests them.

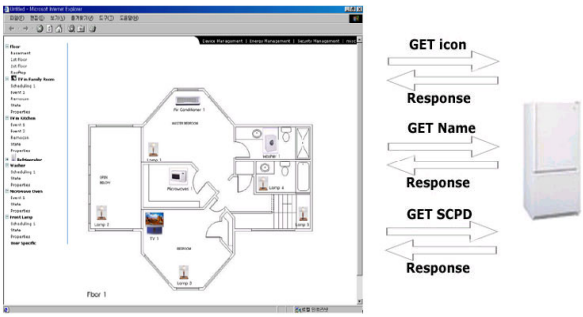
The User Control Point (UCP) requests a description document and the device sends it in Step 2. The UCP can retrieve a description document by issuing an HTTP GET on a description Uniform Resource Location (URL). This URL is returned in the location header of a Simple Service Description Protocol (SSDP) announcement. An HTTP GET is used to retrieve sub-elements of a description document that are expressed as URLs. The UCP requests additional information, such as device icons, the name in Step 3. When the user selects a device icon, the presentation server of the controlled device sends the device user interface in Step 4. In Step 5, the user can control and monitor the device with the User Interface (UI).



a) Step 1: Discovery and Identification

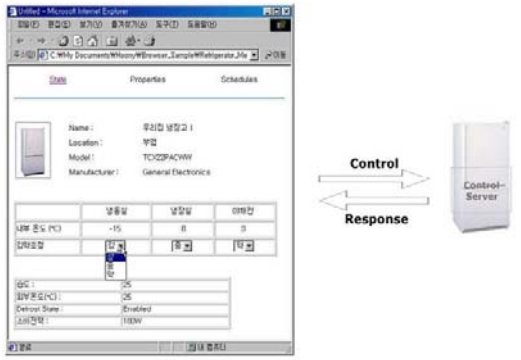


b) Step 2: Send and Request of Device Description



c) Step 3: Additional Information Exchange

c) Step 4: Send Device User Interface



d) Step 5: Device Control

Fig. 4: XML-based Browsing Scheme in Test-bed

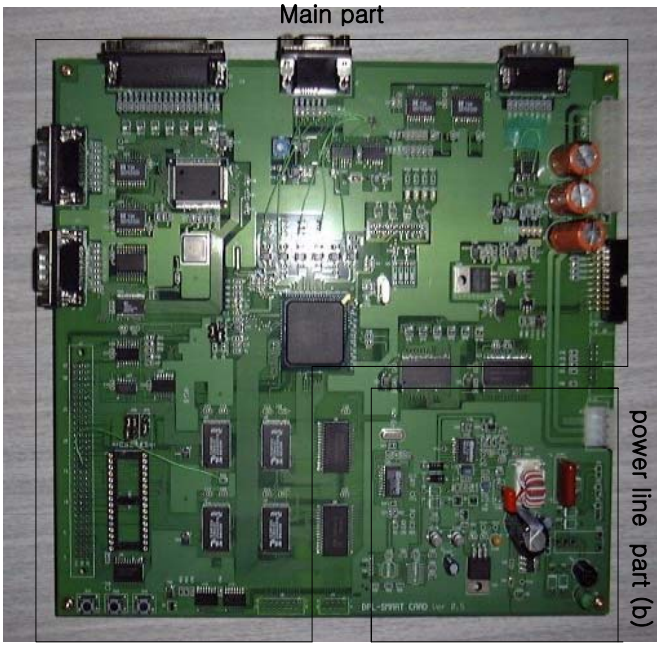


Fig .5: Prototype of Low Rate PLC Module

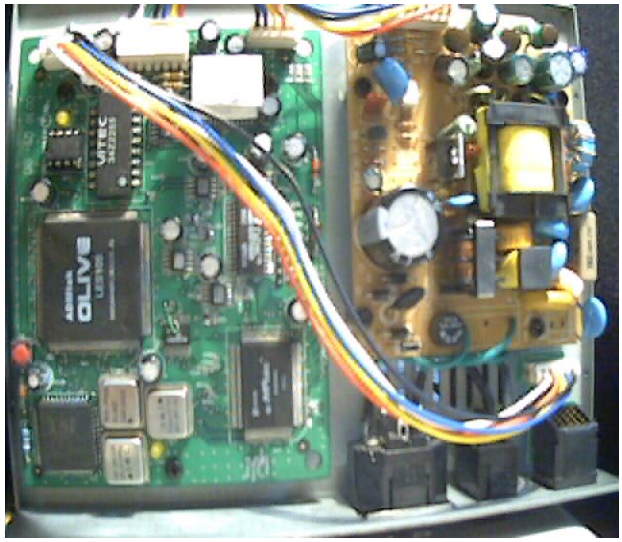
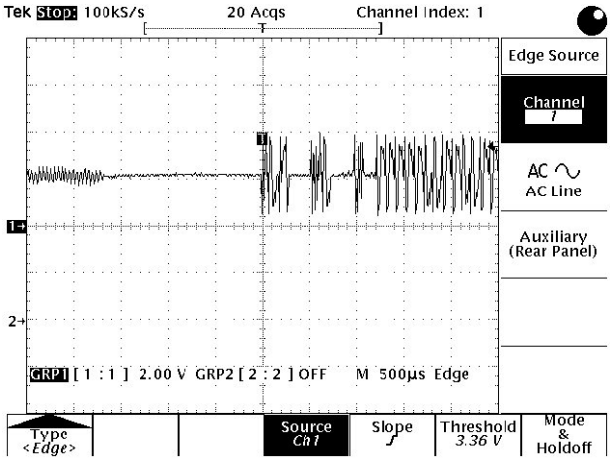


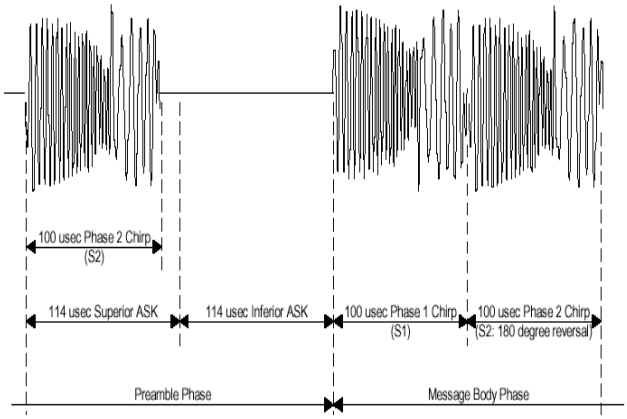
Fig. 6: Ethernet-based High Rate PLC Module

4.2 PLC Modules in the Test-bed

For the simulation, two types of PLC modules are used. One is used for low data rate (9600 bps) applications, such as an air conditioner, using an 8-bit microprocessor. For this module, commercial low rate PLC chipset is used. The other module (Fig. 6) is used for home appliances such as a refrigerator used as a kitchen server, which demand a medium data rate in data communication. For the web-connected application, the 32-bit microprocessor and low rate PLC chipset are used. For high data rate devices (1 Mbps), a PLC modem (Fig. 6) is used as a server and an emulator's test in the home network system.



(a) Measurement of data signal in transmission



(b) Signal Pattern using the PRK method

Fig. 7: Measurement of the transmission signal low rate PLC module

In Fig. 7, the measured signals of power line communication are shown. In Fig. 7 (a), the data signal of the low date rate PLC module is measured in transmission. Fig. 7 (b) shows the preamble signal and main data signal using phase reverse keying (PRK).

4.3 Implementation of Emulators using HNMS

In Fig. 8, the function block of the refrigerator is described. The refrigerator emulator is composed of sub-modules, e.g., a freezer, a vegetable compartment, and a dispenser. In Fig. 9, the refrigerator emulator based on the proposed function block is shown. This emulator can be operated on a Linux platform. Fig. 9 represents the status of the refrigerator. It displays the opening status of the vegetable room door. The air conditioner emulator is shown in Fig. 10. This emulator can be operated on a WinCE platform.

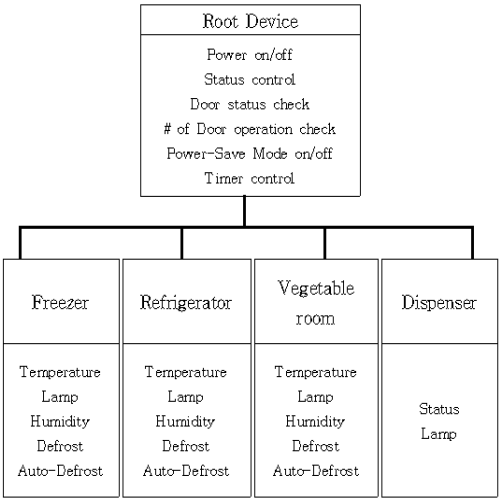


Fig. 8: Function Block of Refrigerator Emulator

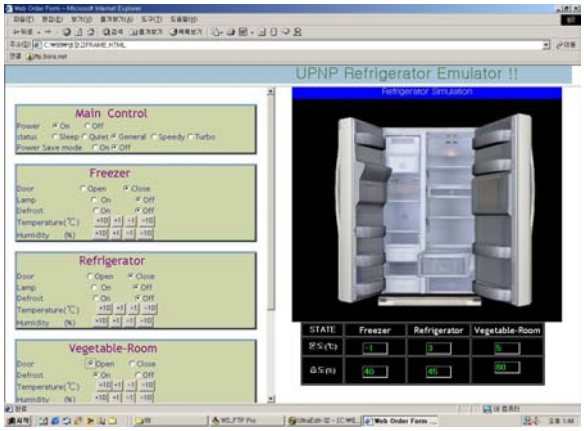


Fig. 9: Display of Refrigerator Emulator

Fig. 10: Display of Air Conditioner Emulator

5 Conclusions

In this paper, an HNMS for white goods based on PLC is proposed. The proposed message specification is designed for remote control and monitoring of white goods in the kitchen area. The proposed home network message specification is composed of a virtual device service and a device-specific attribute. It specifies a set of APIs allowing manufacturers to develop applications for the home network system of white goods. As examples, the specifications for a refrigerator, a washing machine, and a microwave oven are described.

The developed home network system includes emulators of each appliance and a home server program. The usefulness of the proposed method is proven through simulations and experiments using the developed PLC modules.

The results of this paper will be useful for basic work for the home network system with white goods. By using an HNMS of white goods, a home network system can be introduced with a low cost and it is simple to implement in an existing home. The implementation results of this paper can be a design guideline for the home network system with white goods.

In the future, the HNMS will be upgraded for reliable home network systems, including a simple recovery scheme and home security.

6 References

[1] Home System specification Version 1.3 Reference, European Home Specification (EHS), March, 1997

[2] ECHONET specification Version 1.0, ECHONET forum, February, 2000

[3] Lonmark 1.0, LonMark Association, February, 2000

[4] EIA Home Automation System (CEBus) Standard IS-60. "Common Application Language Specifications," EIA, Part 8, Jun. 29th 1992.

[5] S. Koutroubinas, T. Antonakopoulos, V. Makios, "A New Efficient Access Protocol for Integrating Multimedia Services in the Home Environment," IEEE Transactions on Consumer Electronics, Vol. 45, No. 3, pp.481 -487, Aug. 1999

[6] A. Chandra, V. Gummalla, J.O. Limb, "An Access Protocol for a Wireless Home Network," International Conference on Wireless Communications and Networking, Vol. 3, pp.1392-1396, 1999

[7] D.L. Waring, K.J. Kerpez and S.G. Ungar, "A Newly Emerging Customer Premises Paradigm for Delivery of Network-Based Services," International Journal of Computer Networks, vol.31, No.4, pp.411-424, 1999

[8] J. Newbury, W. Miller, "Potential Communication Services Using Power Line Carriers and Broadband Integrated Services Digital Network," IEEE Transactions on Power Delivery, Vol.14, No.4, pp.1197-1201, 1999

[9] T. Pfeifer, "Internet-Intranet-Infranet: a Modular Integrating Architecture," IEEE Workshop on Future Trends of Distributed Computing Systems, pp.81-86, 1999

[10] S. Ramanathan, R. Gusella, "Home Network Controller for Providing Broadband Access to Residential Subscribers," IEEE Transactions on Consumer Electronics, Vol.41, No.3, pp.859-868, Aug. 1995

[11] P.M. Cocoran, F. Papai and A. Zoldi, "User Interface Technologies for Home Appliances and Networks," IEEE Transactions on Consumer Electronics, Vol.44, No.3, pp.679-685, Aug. 1998

[12] D.S. Kim, W.H. Kwon, Y.H. Kim, H.J. Shin, Y.I. Kwan, "Development of Home Network System Based on Power Line Communication", pp.709-711, Proceedings of IKEE Conference, 2000

[13] D.S. Kim, K.J. Myung, W.H. Kwon, H.J. Shin, "Power Line Based Home Network Message Specification and Its Application", IEEE International Symposium on Consumer Electronics, pp.111-117, 2000

[14] Universal Plug and Play Device Architecture Reference Specification Version 1.0, Microsoft Corporation, June 2000

[15] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi and P. Samarati, "Design and Implementation of an Access Control Processor for XML Documents," Computer Networks, Volume 33, Issues 1-6, June 2000, pp.59-75.

Dong-sung Kim: Dong-sung Kim was born in Korea on January 25, 1969. He received the B.S. and M.S. degrees in Electronic engineering from Hanyang University, Seoul, Korea, in 1992 and 1994, respectively. Since 1998, he pursues the Ph.D. degree from Seoul National University, Seoul City, Korea, in 1975. From 1994 to 1998, he worked as a full-time researcher in ERC-ACI at Seoul National University. His main research interests are currently home automation system, networked control system, and industrial networks. Since 1998, he is IEEE/ACM student member.

Gyeyoun Cho: Gyeyoun Cho was born in Korea on September 24, 1970. He received the B.S. and M.S. degrees in industrial engineering from Seoul National University, Seoul, Korea, in 1993 and 1995, respectively. He worked as full-time researcher in Engineering Research Center for Advanced Control

and Instrumentation, Seoul National University in 1995 - 2000. His main research interests are traffic modeling and performance analysis of telecommunication system, industrial & local area networks and home automation system. He is currently a Ph.D candidate in Seoul National University.

Wook Hyun Kwon : Wook Hyun Kwon was born in Korea on January 19, 1943. He received the B.S. and M.S.degrees in electrical engineering from Seoul National University, Seoul, Korea, in 1966 and 1972, respectively. He received the Ph.D. degree from Brown University, Providence, RI, in 1975. From 1976 to 1977, he was an adjunct assistant professor at University of Iowa, Iowa City. Since 1977, he has been with the School of Electrical Engineering, Seoul National University. From 1981 to 1982, he was a visiting assistant professor at Stanford University, Stanford, CA. Since 1991, he has been the Director of the Engineering Research Center for Advanced Control and Instrumentation. His main research interests are currently multivariable robust and predictive controls, statistical signal processing, discrete event systems, and industrial networks.

Yuh In Kwan: Yuh In Kwan was born in Korea on April 16, 1967. He received the B.S. and M.S.degrees in Electronic engineering from Korea University, Seoul, Korea, in 1988 and 1999, respectively. Since 1989, he is research engineer in LG Electronics Digital Media Research Lab.. He is currently senior research engineer in LG Electronics Digital Media Research Lab. His main research interests are currently a home gateway system.

Yohee Kim: Yohee Kim was born in Korea on April 16, 1944. He received M.S. degrees in electrical engineering from Yonsei University and the Ph.D. degree from Kyunghee university ,Seoul, Korea in 1992 and was with KEPCO (Korea Electric Power Corporation) 15 years. He has been with KERI (Korea Electro-technology Research Institute) Changwon city , Korea, From 1981 to 2001. Now he is executive director of KERI, Seoul, Korea. and Chairman of the PLC-forum Korea. From 2000 to 2001. His main research interests are currently optical communication and optical fiber system.