Title:	Document Version:
Deliverable D4.9 **First Prototype of the PLC-based Home Automation System**	1.1

Project Number:	Project Acronym:	Project Title:
IST-2001-37613	6POWER	IPv6, QoS & Power Line Integration

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type* - Security**:
31/03/2003	02/04/2003	P – PU

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other
** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Responsible and Editor/Author:	Organization:	Contributing WP:
Antonio F. Gómez Skarmeta	UMU	WP4

Authors (organizations):

Jordi Palet (Consulintel), Daniel Martínez Manzano (UMU), Juan José Pujante Moreno (UMU).

Abstract:

This document explains the home automation device designed and developed by the University of Murcia.

Keywords:

Home Automation, IPv6, PLC, Power Line Communications, Smart Card.

© 6POWER Consortium

Revision History

The following table describes the main changes done in this document since its creation.

Revision	Date	Description	Author (Organization)
v1.0	31/03/2003	Document creation	Daniel Martínez (UMU) Juan José Pujante (UMU)
v1.1	02/04/2003	Final Review	Jordi Palet (Consulintel)

Executive Summary

The University of Murcia has been working in the development of a home automation device that allows controlling external devices.

This document explains this device created by the University of Murcia, the components that are part of it and its functionality as a home automation element in buildings and residences.

Table of Contents

Table of Figures

1. INTRODUCTION

The University of Murcia has been working in the development of a device capable of carrying out home automation actions like door and lighting control.

This device makes use of Smart Cards, carries out secure communications and its software is completely written in Java. It is also prepared to be able to control external signals that allow it to extend its functionalities.

Actually, the device is able to perform actions like open doors, check if door is open or closed and check internal parameters. These actions can be performed from a Java application thanks to the device's Java API.

This device allows the establishment of access policies for buildings and working time control for the personnel.

2. DEVICE DESCRIPTION

All the functionalities described before can be carried out thanks to the combination of software and hardware components.

2.1 Hardware Components

The processing module of the device is based on a i486DX single board computer (SBC) running at 133 MHz and an interface board based on the PIC-16F877 micro-controller running at 20 MHz.

Figure 2-1: SBC Main Board

The board has a DiskOnChip of 16 Megabytes, a 10Base-T ethernet network interface and supports remote booting.

This main board connects the others hardware components. As it has been described, the device is able to perform actions like open doors and check some internal parameters. So, the device has a controller board (designed by UMU) to manage peripherals and actuators.

This board manages the display and keyboard that will be described later. In the same way, it can enable and disable the actuators that are connected to it.

Figure 2-2: Controller Board

The board manages the display, keyboard and actuators thanks to a simple serial port protocol. Now, this board is able to manage up to two external systems (doors, light, …) by its two actuators (relays), to read two digital inputs and one analog input (sensors) and to control the display and keyboard through its serial port. Also, it has a beeper for user feedback operations.

The device works with Smart Cards compliant with the ISO/IEC 7816. Actually, the device is working with a Smart Card reader from C3PO. The Smart Card reader is controlled through the serial port.

Figure 2-3: C3PO Smart Card Reader

The user can interact with the device by means of a 4x4 matrix keyboard. This component allows the user to introduce the personal identification number of his Smart Card or introduce others commands. The keyboard is connected to the controller board through a 10–wire bus and a serial port connector.

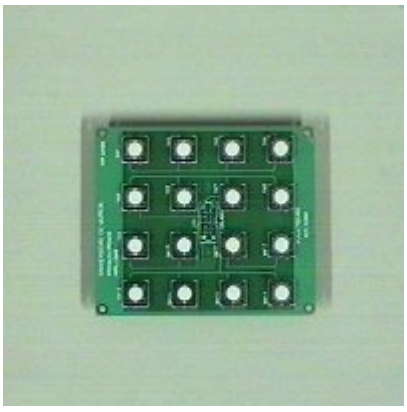

Figure 2-4: 4x4 Matrix Keyboard

For a total interaction user-device, it has a 4x20 LCD display. The applications can show messages through this display.

Figure 2-5: LCD Display

2.2 Software Components

In this chapter, all the software components of the device will be explained. The software components run over Red Hat Linux operating system. Due to the disk space limitations of the main board's Disk-On-Chip, it was necessary to reduce the size of the operating system to be able to be installed.

The current application that manages door and others actuators is written in Java, so another software component of the device is a Java Virtual Machine for running the application.

The device has a complete Java API to control the actuators and inputs and to carry out secure communications.

The device makes use of a wide range of technologies like:

- *JavaComm:* Java technology for the communication with the different hardware components through serial or parallel port [JavaComm].
- *OCF (Open Card Framework)*: this Java framework allows a structured access to smart cards. The device works with ISO/IEC 7816 smart cards and with smart cards based on JavaCard techonology [OCF].
- *SSL (Secure Socket Layer)*: thanks to this secure protocol, the device can carry out transactions where authentication, confidentiality and integrity are assured.
- *Jini*: Java technology based on RMI to create a distributed architecture. By means of this technology is possible to manage and monitor the devices of the architecture from a central point.

It is necessary to emphasize that thanks to the current operating system installed in the device, it is able to support IPv6 addressing. This is important, because it will be possible to deploy device into Ipv6 networks.

The migration of the current device's application will not be a problem either. The Java Virtual Machine version 1.4 and upper includes support for Ipv6 in TCP and UDP based applications.

3. PLC INTEGRATION

An interesting aim to achieve is the integration of PLC technology in the devices designed by UMU. This would allow deploying these devices where dedicated data networks are not available.

The current prototype of the device does not allow incorporating a PCI PLC cards due to its space and size limitations. Also, the main board has not any PCI slot where to plug a PLC board. The integration of a PLC card requires significant changes in the device's hardware that now cannot be achieved.

With this first prototype, the communication between the PLC technology and the device is done via the OSGi Framework. This framework is a services gateway where the service providers can deploy their services. [D4.8] This element represents the central point for services in the PLC remote automation scheme. The framework will run on a PLC-enabled host and it will allow to access and to use the device's available services.

In this case, the device can access to the PLC data network by means of a PLC external modem. The connection between both devices is through ethernet. The home automation device makes use of its ethernet port to gain access to the data network.

The future version of the device will be guided to have a main board with PCI connectors. With this fact, the aim to achieve is the fully integration of PLC technology in the device. It will be possible then to plug a PLC card, being able the device to be connected through the PLC network without the need of an external device like a PLC modem.

In the Figure 3-1 it is possible to watch a scheme where the device controls the main door of the residence. The device in conjunction with the OSGi framework, allow the user to manage remotely the door making use of a service installed on the framework.

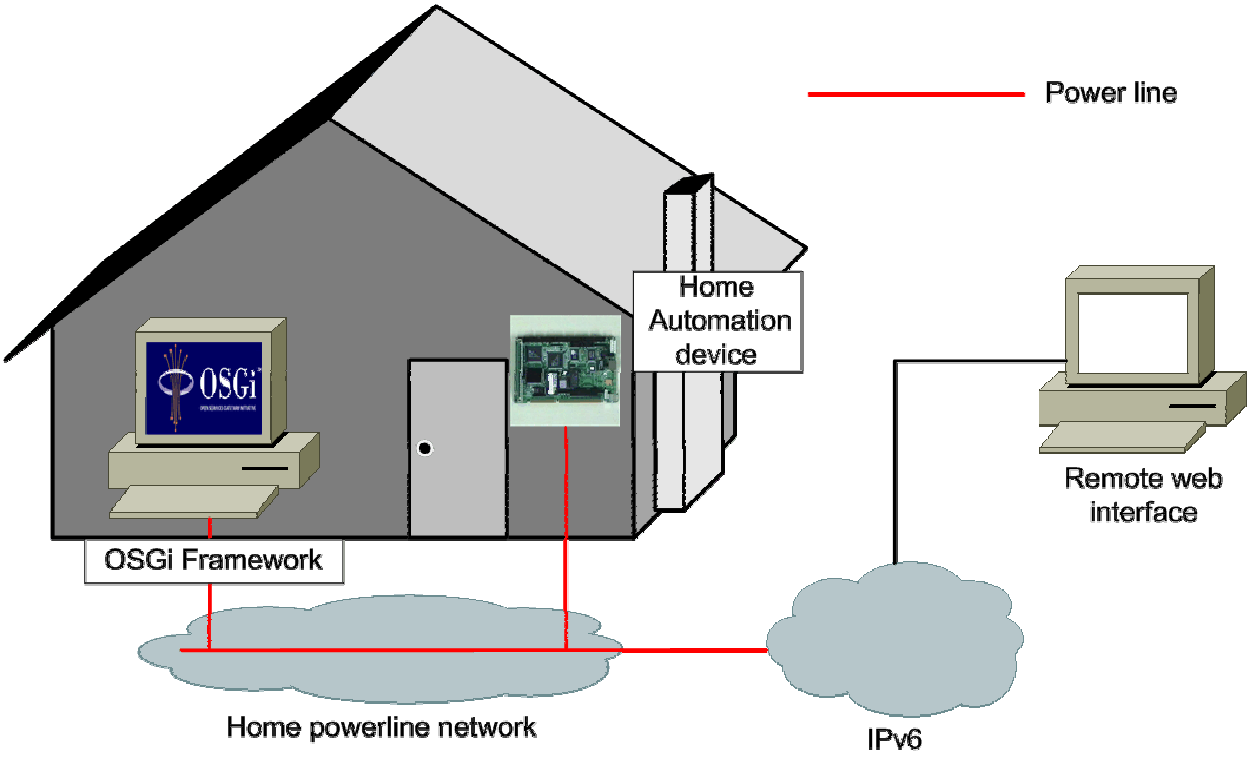

Figure 3-1: Home Automation Device in Action

4. SUMMARY AND CONCLUSIONS

As seen, the device described in this document is an important and useful element to be incorporated in a home automation system.

The fact of being able to be connected to a wide area data network allows the remote management of the device.

Its Java API allows application developers to create software able of using all the hardware features of the device to control doors and other systems.

5. REFERENCES

[D4.8] 6POWER Project. Deliverable 4.8 "Identification of IPv6-enabled device to be used in Home Automation".

[JavaComm] Java Communications API. http://java.sun.com/products/javacomm/

[OCF] Open Card Framework. http://www.opencard.org/