# An Internet Accessible Remote Controlled Home Automation System

Mave T. Houston, Richard O. Chapman, and Loretta A. Moore

Department of Computer Science & Engineering
Auburn University, Alabama 36849
+1 334 844 6314
{maveh, chapman, moore} @ eng.auburn.edu

## Abstract

Home automation systems have been in development since the late seventies. However, the home automation market has not fully reached its potential. This can be attributed to a number of factors. Remote home automation systems, however, are quickly becoming more and more popular. Users like the ability to change their home environment while not at home. The remote home automation systems in existence, however, use the telephone as a medium for communication. These telephone user interfaces, unfortunately, are crude at best and often lead to a frustrating attempt to access the system. A better system would be one that could offer a graphical user interface, such as a homepage on the internet. Thus, if the user can access a web browser, the user can access the home automation system. In this paper, we discuss the design and implementation of such a system, an Internet Accessible Remote-control Home Automation System.

## 1 Introduction

Recent advances in technology have enabled computers to extend beyond the research and commercial arena and into the homes of the average consumer. This is realized in the form of personal computers, advanced home stereo and theater systems, and home security systems. Moreover, with the advent of numerous new, advanced appliances, consumers are increasingly becoming more interested in the easy management of many of their appliances and equipment. Systems that make it easy for users to control their appliances, computer(s), entertainment and security equipment, are referred to as home automation systems.

When designing home automation systems, there are a myriad of issues that should be addressed, not the least of which is how to integrate incompatible hardware devices from different manufacturers. To ease this problem, hardware/software codesign techniques can be applied [1]. Thus the hardware/software codesign concept that involves using a "software-glue" interfacing to hold the hardware and software components together, much like the way a builder uses mortar to hold together a house of bricks and rocks, has tremendous potential.

The home automation system proposed in this work would not only employ the idea of "software-glue" mentioned above, but would also add convenience by being accessible remotely. This system is a remote controlled home automation system that is accessible via the internet. Java will serve as the "software-glue" that holds this system together, since the user-interface of the system as well as the interfaces to the hardware devices will be written in Java. The development of this system will explore the architecture-neutral feature of Java, analyzing the capability of Java to interact with any architecture, making it a suitable choice for being the "software-glue". In essence, the system will be a step in a new direction for home automation, and hardware/software codesign.

## 2 Design Issues

As was stated before, the overabundance of home automation bus standards and electric devices that cannot communicate with each other are two major problems of home automation. This lends itself to the idea that integrating various devices with various architectures can be a difficult and daunting task. Simply designing a home automation system with (specific) respect to hardware or software exclusively can cause system integration problems in later stages of development. Hardware/software (HW/SW) codesign concentrates on solving problems encountered in the design of hardware/software systems and, for embedded systems, solving the problem of having application-specific embedded system architectures with numerous programmable processors that must be integrated so hardware and software can communicate successfully [3].

In the case of a home automation system, with various architectures and the potential to interface with numerous hardware devices and appliances, a HW/SW codesign method for embedded systems will be used. This method is used to develop an interface for a heterogeneous system. The interface can be hardware or software, but must have the ability to communicate efficiently with all components of the heterogeneous system [4]. The idea of this "glue" will be extended to the home automation problem of integrating all software and hardware components with a "software glue". That is, software will serve as the interface to all hardware and software, satisfying the need for a strong interface to all of the system components. This is a must and can be acquired by using the "glue" logic method proposed by Lin [4].

Java was chosen as the software interfacing for many reasons. Because Java was created for heterogeneous network environments, one of its initial goals was to be used in embedded systems (like home automation systems) that require minimal memory, and use a minimal amount of hardware and software resources [6]. Thus java's contribution to the proposed system encompasses many areas. First, Java will serve as a software interfacing for all software and hardware components. Secondly, Java will provide a graphical user interface using the Abstract Window Toolkit (AWT).

Numerous home automation systems have been introduced on the market that employ many different protocols and busses. Some of these include: SmartHouse, CEBus, LONworks and PowerHouse. Some automation systems also allow for remote control access, but only through direct telephone control. No current home automation systems allow for control through a remote site on the Internet, as this proposed system will. Secondly, although numerous Home Automation Systems use microcontrollers as the central control for the manipulation of home devices, no one has used microcontrollers coupled with Field Programmable Gate Arrays. The use of a microcontroller will free up the personal computer from undue traffic accrued by the user commands coming from the remote site. Additionally, with the use of Java, exploration of its architecture-neutral feature will answer the question of whether Java can actually interface properly with hardware and software in an embedded system and meet real-time constraints. This system also uses methodologies from Hardware/Software Codesign, specifically the method proposed that employs the notion of using a "glue" of sorts to hold the components of a heterogeneous embedded system together. On a less technical note, the system will also provide convenience features such as remote VCR programming, thermostat control, answering machine message access and control of AC appliances such as lights, television, and entertainment equipment [5].

## 3 Algorithmic View of the System

The Internet Accessible Remote Control Home Automation System (IARHAS) consists of several components. One component is the user interface. This user interface is a hypertext interface, i.e., it is presented as a homepage. The user will basically enter the system by accessing a link to this homepage, and enter appropriate information as prompted by the interface. More specifically, Java applets will capture this information and generate java source code modules to carry out user requests. Another component of the system is a personal computer (at the home) that serves two purposes. The first is to emulate a web server that will receive the java source code modules that were generated by the user's session at the remote site. Secondly, the personal computer will also communicate with the microcontroller and house the answering machine software. The microcontroller has the task of communicating with the pc and controlling the hardware that controls the devices of interest. The last component consists of custom hardware interfaces to the various controlled devices. This will be implemented by using Field Programmable Gate Arrays. These various components are depicted in figure 4.1.

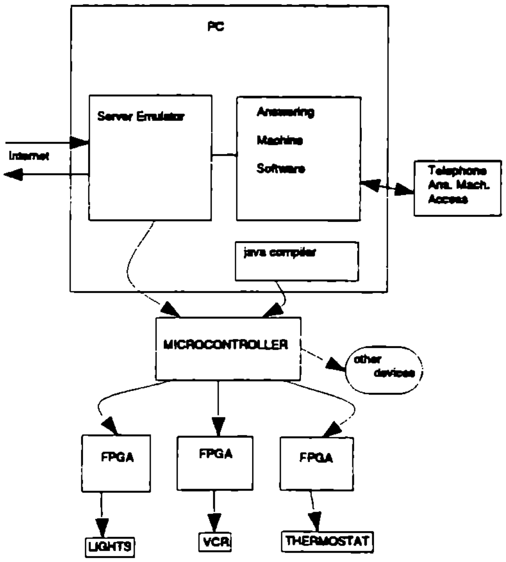

figure 4.1, system architecture

The control of the IARHAS begins from the IARHAS Main Homepage, which is reached through a hyperlink from the user's homepage. All attempts to enter the system from the Internet via going to a particular page in the system will be met with a user authentication check and then control will be routed to the IARHAS main homepage. That is, every attempt to access a page other than the main page (i.e., going to a specific html file of the system) will result in a user authentication check, unless the user has already been verified. Upon user authentication, the user will be routed to the IARHAS main page. Otherwise, the user will be given a warning message and returned to the last link accessed before the failed attempt, and the attempt will be logged.

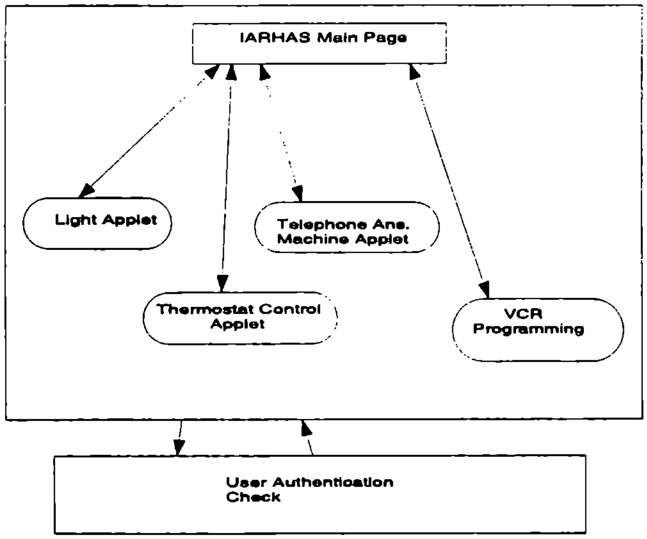
IARHAS Main Page
Light Applet
Telephone Ans. Machine Applet
Thermostat Control Applet
VCR Programming
User Authentication Check

figure 4.2, Hypertext Flow of the IARHAS

After the user has been authenticated and before the main IARHAS page is displayed, a request is sent to the server emulator of the home pc for the answering machine information. The answering machine software resides on the personal computer, so to access the number of messages and the messages themselves is as simple as copying the files with that information and sending them back to the user who is accessing the IARHAS. Then, after the system request (of message files) is fulfilled, the user is routed to the main IARHAS page, where the user session can begin.

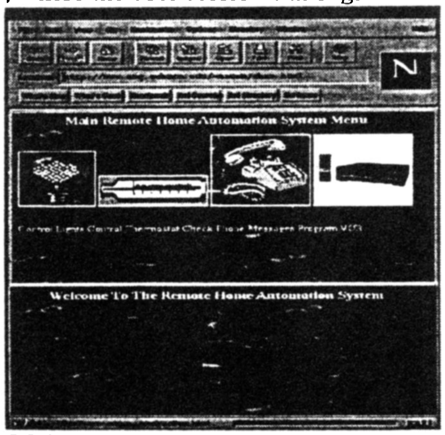
N
Main Remote Home Automation System Menu
Control Lights Control Thermostat Check Phone Messages Program VCR
Welcome To The Remote Home Automation System

figure 4.3, Main IARHAS page

The main IARHAS page is split into two frames, as depicted in figure 4.3, with the upper frame displaying the choices to the user. The initial lower frame welcomes the user and explains usage details, etc. When the user clicks on one of the choices, the corresponding applet is displayed in the lower frame. The user then has a choice of four applet links: control lights, control thermostat, access answering machine, or vcr programming.

Each applet has its own set of actions and reactions to events. It's purpose is to gather information from the user and generate a Java module to implement the user's request. After the user completes the session, the session should be submitted. This will actually direct the system to send the complete set of generated Java code modules to the server emulator running on the home PC.

Once the packet has been received by the home personal computer, it is downloaded into the microcontroller. This microcontroller is running a Java bytecode interpreter which interprets the Java code and produces an executable that will control the Field Programmable Gate Arrays (FPGA's) connected to the microcontroller. The FPGA's will control the lights, vcr programming, and temperature control.

## 4 Summary

The notion of a truly architecture-neutral language is one that if of great interest to many programmers. Java has made the claim that it can be implemented on numerous processors, and that it could be the solution to many integration problems. The purpose of the development of the IARHAS system is to implement the codesign methodology that calls for a software interfacing, or "glue", using java. The implementation of the IARHAS system can be accomplished in four primary steps. Step 1 is to implement the hypertext user interface. Step 2 is to design and implement the custom hardware components. Step 3 involves the integration of the software and hardware, using the software interfacing written in java. At present time, the user interface is nearing completion, and the design of the hardware has been initiated. We expect completion of this system by early April, 1997.

## References

[1] Buchenrieder, Klaus, Rozenblit, Jerzy W., "Codesign: An Overview", Codesign, IEEE Press, 1995.

[2]Cornell, G. Horstmann, C. S., Core Java, The Sunsoft Press, Mountain View, CA, (1996).

[3]Ernst, Rolf, Henkel, Jorg, Benner, Thomas, "Hardware-Software Cosynthesis for Microcontrollers." IEEE Design and Test of Computers, IEEE 1993.

[4]Lin, B., Vercauteren, S., DeMan, H., "Embedded Architecture Co-Synthesis and System Integration.", IEEE Proceedings Fourth International Workshop on Hardware/Software Codesign, March 18-20, 1996.

[5]Strassberg, Dan, "Home Automation Buses: Protocols Really Hit Home", EDN, April 13, 1995.

[6]Yellin, Frank, "Low Level Security in Java", Fourth International World Wide Web Conference, Boston, MA, Dec. 1995. World Wide Web Consortium.