

## DEVELOPMENT OF A SYSTEM FOR “HOME CONTROL ON FINGERTIPS”

**Grisha Spasov, Mitko Shopov, Ivan Stankov, Pavel Zaikov, Nikola Hadzhinikolov**

Department Computer Systems and Technologies, Technical University of Sofia – branch Plovdiv, Tzanko Djustabanov N° 25, 4000 Plovdiv, Bulgaria, phone: +359 32 659 576, e-mail: gvs@tu-plovdiv.bg

*Nowadays, advanced technology devices are all around us. We take them for granted and we are using them in all spheres of social life. We decided to combine the fast development of the technology and daily human habits in a centralized system that we called “Home Control on Fingertips” (HOCFIT). The system is designed in hierarchical three-layer model: master, sub-masters, and end-devices. Each of them has their features and suitable interfaces to communicate with each other. Master accepts the commands from the authorized user and transmits it to the definite sub-master and the sub-master manipulates the end-device. The communication software is based on the TCP/IP protocol stack.*

*The advantage of this system lies in the universal design, which allows easy and fast future development when connecting new devices for control or data acquisition as well expansion of the system's functionality. The extendibility and flexibility of the system are assured using popular technology like Bluetooth, integrated in GSMs, PDAs, Notebooks, etc., and IrDA, used in every remote control, almost every new domestic appliances have today.*

**Keywords:** home automation, distributed embedded systems, home networking.

### 1. INTRODUCTION

Nowadays house is equipped with different technical facilities like domestic appliances, TVs, Audio and Video systems, Air-conditioner, security home systems, etc. For their optimal exploitation, control, monitoring and reaction in emergency events the user should possess specific knowledge and skills, which is not so easy for all people, especially for people with different kind of physical disabilities. The main reason for development of Networked Embedded Systems ‘home automation’ is to facilitate and optimize the everyday usage of all these appliances [1]. We decided to combine the fast development of the technology and daily human habits in a centralized system that we called “Home Control on Fingertips” (HOCFIT).

We have used real-life approach to justify and establish the benefits of our system considering the following prerequisites and possibilities:

- Widely spread of computers and advanced technology in human lives in general;
- Automation of essential daily tasks, allows people doing more important and pleasant things;
- Easily control and work for people with physical disabilities in home environment.
- Monitoring status of your home, when you are at work or vacation. HOCFIT will notify you for emergencies;
- The heart of our system will be developed on widely used communication protocols and standards. In this way, it is very easy to be enhanced and customized.

All what you need to use the HOCFIT is just having a mobile phone, PDA, Laptop or workstation and they to be connected to the Global Network – Internet or just be in appropriate distance for wireless Bluetooth communication. HOCFIT is very flexible system – to access it, doesn't matter what hardware is using and what operating system is running. The only requirement at user side is integrated TCP/IP stack, some primitive web browser and network connection.

## 2. SYSTEM OVERVIEW

The functional diagram of the HOCFIT system is shown on figure 1.

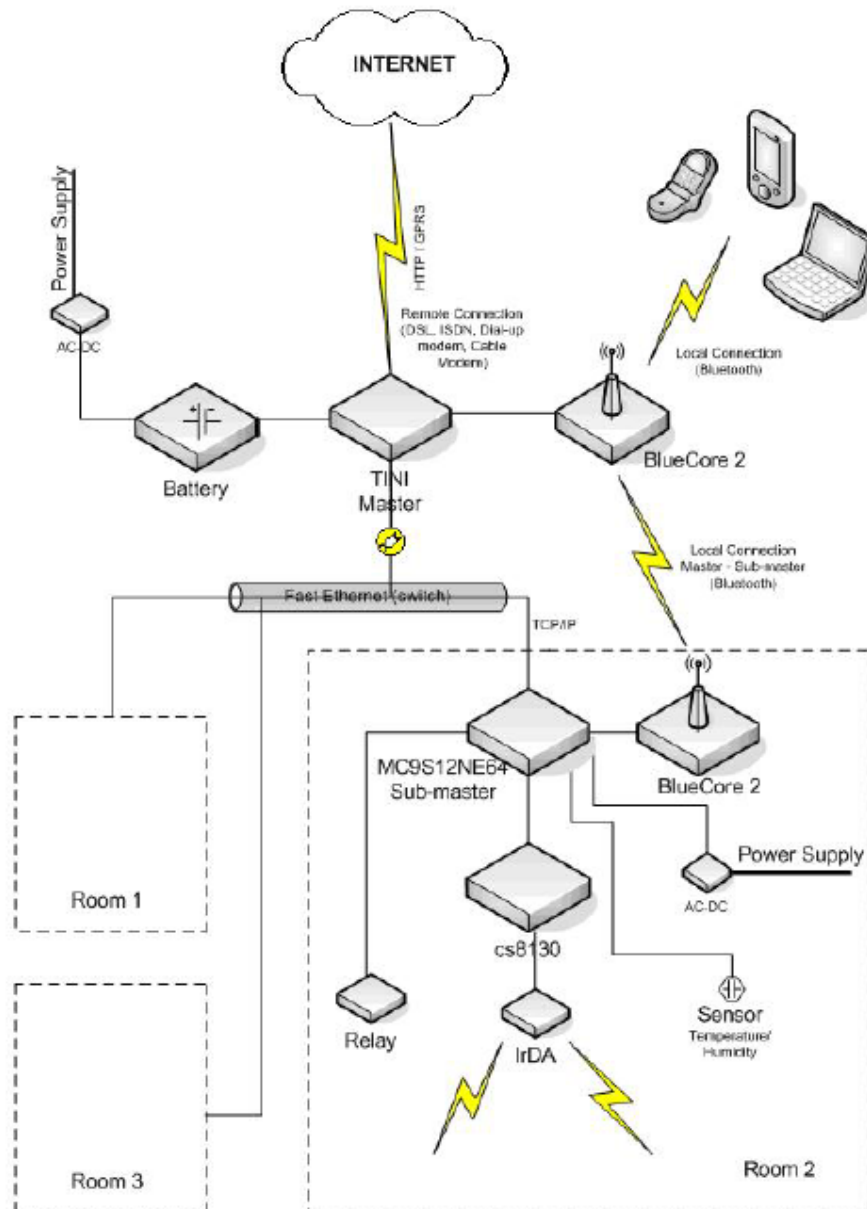


Figure 1. The HOCFIT functional diagram.

We have designed a system, which can be accessed from a central (master) unit to any part of the home. We choose three-layers architecture. One master controller on top level will provide the external access to the entire system. It will support various connections to the Internet for remote access, and a Bluetooth and Fast Ethernet modules for local connections. Next system layer consist of many sub-master microcontrollers depending on the particular needs, with different interfaces – SPI, RS232, 1-wire, etc. The lower levels of system architecture are end-devices such as different types of sensors and controls – switches, IrDA transmitters and so on. The advantages of this design are:

- One sub-master device for each room controlling a variety of end devices;
- Universal method for communication between master and sub-masters at physical layer of TCP/IP protocol stack. Employing Ethernet for main connection and Wireless (Bluetooth) for back-up connection;
- Reduced need for cables, because one sub-master will represent many devices in the system;
- External connections to the master are provided by means of different connections to the Internet, because it depends on the ISP service. For example, most often the Internet connection is accomplished by PSTN using telephone line and PPP protocol. Another alternative is cable modems. Using USB or RS 232 interfaces on the master, there are no problems to connect any types of modems. The third possibility is usage of GSM communication – some providers give dial-up Internet via GSM;
- Redundant topology increases reliability of the system;
- Administration of the system is performed with Remote Login accomplished by PC or GSM (GPRS capable).

The system hardware is based on different microcontrollers – for master: DSTINIm400 (Dallas Semiconductors) and sub-masters: MC9S12NE64 (Freescale – launched by Motorola).

### **2.1 Overview of the Master device system**

We chose for master controller DSTINIm400, which is based on processor DS80c400 with working frequency up to 75MHz [3]. It supports many different interfaces: RS232, Ethernet – 10/100 Mbps, CAN, I2C – connection to co-processor for USB interface – Cypress SL811HS and 1-Wire™ net for future expansion. We use operating system – TINI™ RTOS 1.15 version and Slush (small command shell intended to provide a UNIX® interface to runtime environment). It supports a large variety of network protocols like PPP, IPv4, IPv6, TCP, UDP, IGMP, ICMP, DAD, SMTP, DHCP, FTP, HTTP, and TELNET. Software applications can be written in Java™, C, or 8051 assembly. Our choice is Java™ – base on 'Java™ 2 Platform, Standard Edition' (J2SE) package. Java files are compiling with BlueJ – simple compiler, to class files and convert to \*.tini files using TINI converter. All applications and codes are written in multithread concepts, because operating system provides parallel working up to 32 threads, software synchronised by semaphores. So for each task or complex calculation we create new thread and after it finished we stop the thread and release system resources. For example we create a new thread to

listen for user requests on specific port or file access. Because master controller has more processor power and memory than sub-masters – standard it is 1MB flash and 1MB SRAM, we decided to use it for storing incoming information of sub-masters and controlling all communication. We use the memory for creating three different applications: DHCP server; local connection; outgoing connection to Web interface.

## 2.2 Dynamic content HTML pages

We use web interface as a single interactive communication to the user. It is based on HTTP 1.1 and working on application layer protocol of TCP/IP stack [2]. Through that interface user can make different choices – mark/unmark options, command end-devices and execute tasks on the sub-masters. Assigning names to the sub-masters allow us easier control over the system – for example: the room's name where microcontroller is situated. Via Web interface the user requests are accepted and the responsible actions are generated. We use one of methods for taking dynamic content of web page for this purpose. CGI and Java applets are used for establishing a connection to the Web server. The open-source Web server for Java applets (servlets) developed for TINI is employed (Tynamo™ Web server – written in Java). It takes request (most often POST) and mapped to correct running application. Our web interface is separate in one application, which is listening on port 80 and waiting users to connect.

Figure 2.1 – Log In screen for Web interface

Figure 2.2 – main menu screen for web interface /High speed with Administrator rights/

The Web interface is organised in hierarchical structure – different HTML pages and capabilities that they give to user of HOCFIT system. After successful log in –

figure 2.1, a main page is displayed with different sub-menus – figure 2.2. Content of this page depends on username (Administrator or Guest account) and connection speed you have chosen. At the bottom of web page short information is carried out about date of last log in and source IP address. Sub-menu items are as follows:

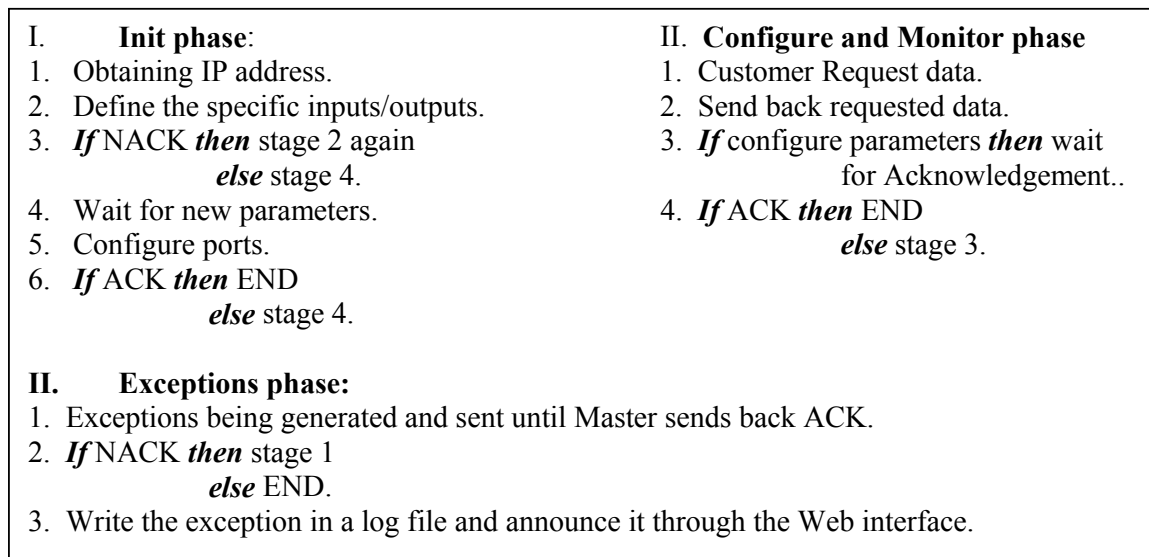
- Configure – a table of currently active sub-master devices is displayed. Each controller could start up to 10 different tasks. Each task begins in predefined time (Date, Hour, Minute) and stops on specific condition – reaching temperature or humidity value or again in specific date, hour, minute. Each task has different living conditions – Active, Waiting, None (so user is free to add new);
- Monitor – the same as Configure sub-menu, but user can only watch started tasks and current configuration in table on specified sub-master controller. This menu is accessible in Guest mode;
- Logs & Exceptions – Exceptions that could be generated are divided in two levels. First level (top level) are these kinds of errors that have to be immediately processed like message from home alarm system – alarm on event, alarm from temperature or pushed ‘panic’ button. Reaction of the system is sending a SMS message. Second level messages are for information – last system log in, when defined sub-master is switch on/off to local network or user defined event. All these logs (level two exceptions) are written to file and could be viewed from Web interface;
- System Configuration (Init Phase) – after a new sub-master is added to the local connection, it is waiting for custom configuration – define sub-master pins – digital inputs, digital outputs and devices connected to via Infrared;

### **3. APPLICATION COMMUNICATION PROTOCOL**

Application communication protocol that we have designed in our system for connection between Master and sub-masters is based on HTTP 1.1. The data transmission is realised by connection oriented protocol with ACK/NACK messages [4]. The communication protocol is divided in three phases: Init phase, Configure and Monitor phase, Exceptions phase.

- Init phase – when the new connected sub-masters are switched on in the HOCFIT. Each sub-master gets own dynamic IP address by DHCP protocol. Then it transmits data for configuration of the ports and their status to the Master device.
- Configure and Monitor phase – the main phase of communication protocol in HOCFIT system, where the current data for end-devices are transmitted from sub-masters to the master device on request or in the other case to reconfigure the sub-master ports.
- Exceptions phase – it is activated in emergency situations, when some of the controlled parameters is out of predetermined range, or alarm situations go on (fire alarm, home security alarm, power supply break down, panic button, etc.)

The algorithms supporting the phases of communication protocol are shown on figure 3.



*Figure 3. Algorithms supporting communication protocol.*

## 5. CONCLUSIONS

We have designed a system, which will help people to be in touch with their home any time and anywhere. It will be helpful for people with physical disabilities and to their relatives. Let's look at real situation when our system can be useful. For example when someone is going to work or vacation and forget switch on domestic appliance. HOCFIT can inform him/her in time before fire or technical damage is happen. User will switch off it remotely via pleasant Web interface.

This system is very useful with capability to be enhanced with a large number of sub-master controllers and end devices. It will work without any problems in small homes, but in large houses, hotels and hospitals it is necessary to add new master devices (as distributed systems) and to modify the communication protocol.

The HOCFIT system is designed in the frame of the IEEE Computer Science International Design Competition 2005 and have passed the evaluation procedures to the final stage.

## 5. REFERENCES

- [1] Estrin, D., G. Borriello, R. Colwell, J. Fiddler, M. Horowitz, W. Kaiser, N. Leveson, B. Liskov, P. Lucas, D. Maher, P. Mankiewich, R. Taylor, J. Waldo, *Embedded Everywhere. A Research Agenda for Networked Systems of Embedded Computers*, NAP Washinton, D.C. 2001, ISBN 0-309-07568-8.
- [2] Jan Axelson, *Embedded Ethernet and Internet Complete*, Lakeview Research LLC, 2003, ISBN 1-931448-00-0.
- [3] Dallas *DSTINIm400 homepage* – <http://www.dalsemi.com/>.
- [4] Sridhar, T., *Designing Embedded Communications Software*, CMP Books © 2003, ISBN: 157820125-X.