Domotic Embedded System

Lidia Dobrescu
"POLITEHNICA" University of Bucharest
ETTI Faculty
Bucharest, Romania
lidia.dobrescu@electronica.pub.ro

Abstract— This paper presents an original domotic embedded system for room temperature monitoring. The OpenRemote is the main software interface between the user and the system, but other software components and communication protocols are used, such as 1-Wire protocol for temperature monitoring devices, RS-232 for the central PC unit and OWFS software for remote control using Android mobile devices. The system architecture consists in hardware and software components to remote control a room temperature parameter for energy efficiency increasing.

Keywords—OpenRemote; 1-Wire; ATMEGA16 microcontroller; Android devices; embedded system, domotic system.

I. INTRODUCTION

The hardware system includes an ATmega16 microcontroller from Atmel manufacturer and a DS18S20 temperature monitoring device with a DS2480B driver as an interface circuit for temperature control.

A central PC with Linux (Ubuntu) operating system is also integrated in the system.

The mobile device can be a smart phone or a common tablet using an Android or IOS operating system.

In the original proposed system, the 1-Wire protocol is used for long distances communications, combining it with OpenRemote free software in an original system controlling the serial port of an Ubuntu server, a mobile device and a microcontroller hardware system. The OpenRemote is free source software working with Java language.

The original 1-Wire protocol was a device communications bus system designed for small networks, on short distances, providing low-speed data and signaling, typically used to communicate with small inexpensive devices such as digital sensors for room temperature or light, with many advantages regarding simplicity and costs. It is extended in this paper beyond its original designed limits.

RS-232 standard and 1-Wire communication protocols are also used in the system. The conversion from RS-232 to TTL voltage signals are realized using MAX232 electronic circuit.

All the above components are embedded using a C program controlling the microcontroller and OpenRemote Design with OWFS software. The OpenRemote program has also been originally interfaced with ATmega16 microcontroller.

The paper describes the new proposed system from the digital thermometer to the command smart phone, highlighting the strong interdependence between hardware and software design.

A real domotic system involves the home automation, in order to keep the constance of the desired monitoring parameters from a room or a building. The proposed system performs the remote temperature monitoring and it can be extended towards a real domotic system for changing the room conditions in order to establish the desired parameter with future mechanical or electrical systems such as heating, ventilation, lighting, shading house appliance.

II. THE TEMPERATURE MONITORING DEVICE

The temperature monitoring is realised using a programmed sensor network [1] and 1-Wire protocol.

The sensor is a DS18S20 integrated circuit and it is connected with DS2480B interface as shown in figure 1.

In fact the sensor is a digital thermometer. Its performance is proven by large temperature measurement limits and temperature step. It also has an alarm function for overpassing the temperature limits settled down by the user in the operating range between –55°C and +125°C. A greater accuracy of ±0.5°C is obtained in the range of –10°C to +85°C, suitable to a normal room temperature towards a real domotic system [2].
The 1-Wire bus is the data line for the sensor DS18S20 for proper communication with the central microprocessor. As a designed protocol for short connection, the 1-Wire protocol is widely used for connecting nearby devices. Soon the 1-Wire protocol has became such popular and its users have succeed to extend its networked applications beyond the common sizes. In this manner the networks using such a simple and robust protocol have became a complex arrangement of sensors, data lines, and connections.

The maximum size extension and the maximum weight of a 1-Wire based network can be hardly calculated. The insufficient drive current provided by the interface is generally a limitation factor. A suitable interface’s impedance in order to avoid signals reflections and interferations with other network slaves are also needed. The reflections, the cable delay, its own resistance conducting to signal degradation are another important limitations for desired network radius.

Another limitation is the network weight. The quite rapid cable charging and discharging can be sometimes a real problem in increasing the network radius. Usually 200m radius can be designed. Increasing the radius towards 500m implies a carefully design.

As the master device, the Atmel microprocessor identifies and addresses the network devices. Each device has a unique code. This important property enables a practically unlimited number of devices that can be accessed on the central bus.

Figure 2 shows a block diagram of the used DS18S20 microprocessor. The unique code of the sensors is stored in a ROM on 64 bits. The monitored output signal from the temperature sensor is stored in a 2-byte temperature register. Another two important registers called TH and TL are used for higher and lower threshold alarm trigger. For data storing when the device can be powered down a nonvolatile memory is also used.

The DS18S20 circuit has the ability to operate either with or without an external power supply. The DS18S20 microprocessor has a VDD power supply. Without this classical power supply it can also operate due to a parasitic power supply extremely useful in case of space constraints remote applications. In figure 2 this special solution of power supply is indicated using a power capacitor that stores power in high state of the bus.

The 1-Wire network was developed by Dallas Semiconductor as a MicroLan.

The DS2480B microprocessor is the master of 1-Wire network. Its communications lines TXD and RXD from the serial communication port are linked with the 1-Wire bus.

Fig. 3. Block diagram DS2480 [4]

Fig. 4. Pinout diagram DS2480 [4]

The inherent problems related to data rate and speed communication between the port and the 1-Wire bus are solved with configurable parameters.

The diagram in figure 3 shows timing converting circuitry. Different bytes for data, commands and calibrating the time generator are used.

All 1-Wire bus and commonly processor data lines speed communication software interfaces are controlled by timing generator.

III. RS-232, TTL CONVERSION AND THE MICROCONTROLLER

RS232 is a commonly used asynchronous serial protocol. It can be used to create a data link between a microcontroller and a standard PC. A microcontroller uses the universal standard called Universal Synchronous Asynchronous Receiver/Transmitter, USART. In 5V system it is a serial data communication. The RS232 is used by personal computers. A conversion between +/-10V of RS232 to to +/-5V of TTL is required in order to permit a proper communication between the microcontroller and the computer. The MAX232, shown in figure 4 is an integrated circuit for the desired matching in a simple manner shown in figure 5. MAX232 is a voltage converter between RS232 and TTL. As a simple converter with a charge pump for voltage doubling.

It includes an electrolytic capacitor. The complete 1-Wire circuit is shown in Figure 5.

Fig. 5. 1-Wire Circuit
The ATmega16 is a 8-bit AVR RISC, low-power CMOS microcontroller and faster than a CISC one.

Power consumption versus processing speed is carefully optimized [7].

The AVR microcontroller was connected with PC. The data is transmitted from the controller using RS232 standard and displayed on the PC using Hyper Terminal. The USART block is sometimes called the Serial Communications Interface or SCI. The program used for USART was Atmel Studio 6 shown in Figure 6.

In the main program PD6 and PD4 have been defined as the pins where two LEDs, a green and a red one were connected as output loads. A new variable press_key was defined to keep the input character. An init function for the serial link was defined. An infinite while was created for input character. The "b" character for lighting and "z" for stop. "Y" is for interrogation.

IV. OWFS PROGRAMS

OWFS consists in a set of programs that can easily access the 1-wire. A virtual file system is created using the same 1D as the directory and the individual properties of the device files [8]. OWhttpd is designed as a web server. The figure 8 shows the main screen for this server.

The temphigh or templow are temperature thresholds, and their overpassing trigger the alarm state in the alarm directory. A non-volatile memory keeps this temperature thresholds in no power state.

It is preceded by the connection of the network 1-Wire system through the serial port to the PC with the operating system Linux version Ubuntu 12.04.

A command terminal is started to descend into the root privileged mode on. After the install of OWFS (which can perform by "Ubuntu Software Center" simple) more OWFS lines are shown in figure 9.

V. OPENREMOTE APPLICATION

OpenRemote is an integrated platform. It brings together several residential automation protocols and vendor-specific device protocols and allows an automation solution building that covers a wide spectrum of devices. 1-wire can be used for identification and authentication, sensing and controlling for example lights or temperature through relays.
Implementation of 1-Wire support in OpenRemote allows read and write access to 1-Wire network. If a parameter value can be read from 1-Wire device via OWFS, it can be accessed via sensor in OpenRemote. Write commands are executed via buttons, switches or sliders. This will, for example, allow users to control external system via relays.

The software package is free to be downloaded and it is designed for LINUX operating system. Certified products are extensively tested in combination with Professional Designer and supported by OpenRemote certified Integrators.

Because OpenRemote uses owserver to communicate with physical 1-wire devices, this part must be configured first in the OWFS environment.

Most typical usage pattern of OWFS is to use it as a filesystem. This is very useful access method to 1-wire that provide easy way to test all the devices[8].

With OpenRemote a custom user interface can be created to control the sensor devices. This can be done with OpenRemote Designer and its basic functions of the UI Designer. The application is split into two main sections: the building modeler and the user interface designer.

For many Android, web-based or iOS control devices control panel can be created using user interface design view. The same device can be created in different user profiles.

On each screen several widgets together make up the interface. A widget is a single element that can be interacted with by the user. Each widget has its own functionality and appearance that can be easily modified using a few settings. First, a general overview of the use of widgets in the UI Designer is given [9].

OpenRemote's architecture can be fundamentally configured and changed by user imagination. „User interface design, installation management and configuration can be handled remotely with OpenRemote cloud-based design tools” [9].

VI. CONCLUSIONS

The proposed embedded system is designed for energy efficiency increasing and green communications and it allows a remote control of a room temperature for an end-user with a mobile device as a smart phone or a tablet. The sensor output is 1-Wire sent to a microcontroller where it is analysed.

A central PC with an installed OWFS server is used as a programmed relay between a mobile device and a microcontroller, as the core of the entire system. The OpenRemote program has been originally interfaced with ATmega16 microcontroller.

The hardware and software solutions are originally put together using different conversion solutions in a unique architecture.

This project can be the starting point towards a real domotic system as a future house automation.

OpenRemote is a suitable solution to automate modern homes and control heating, TV, lights, media centers using Android devices. It is an open source program control and panel software in a domotic space and it can be integrated with various software protocols.

REFERENCES

[1] S. Sitharama Iyengar, Nandan Parameshwaran, Vir V. Phoha, N. Balakrishnan, Chuka D. Okoye “Fundamentals of sensor Network Programming: Application and technology”, John Wiley & Sons, Inc., Hoboken, New Jersey, 2010, pp.113-130.
[2] Maxim Integrated Products, Inc., DS18S20 Datasheet (2010), Available: http://datasheets.maximintegrated.com/en/ds/DS18S20.pdf
[3] Maxim Integrated Products, Inc., Tutorial 148, Guidelines for Reliable Long Line 1-Wire Networks, (2008), Available: http://www.maximintegrated.com/en/app-notes/index.mvp/id/148.
[4] Maxim Integrated Products Inc., Application Note 192, Using the DS2480B Serial 1-Wire Line Driver (2002), Available: http://pdfserv.maximintegrated.com/en/ds/DS2480B.
[5] Extreme electronics, Best free Microcontroller Tutorials and Projects, (2014), Available: http://extremeelectronics.co.in/avr-tutorials/rs232-communication-the-basics/.
[6] Atmel Corporation, AVR8-bit Microcontroller (2002), Available: http://www.atmel.com/Images/doc2466.pdf.
[7] Lim Siong Boon, Serial Communication, RS485 Interfacing, (2009), Available:http://www.siongboon.com/projects/ 2006 – 03 -06_serial_communication/#RS48 & 2012Interfacing.
[8] Sourceforge.net, Dice Holdings, Inc. Service (2014), Available: http://owfs.sourceforge.net/owfs/1.html.
[9] OpenRemote Inc., Open remote Documentation, (2008-2012), Available:http://www.openremote.org/display/docs/OpenRemote+2.0+How+To+-+1-Wire+Sensors.