The Online Temperature Control

Florin Ravigan, Niculae Boteanu and Laurențiu Alboteanu

University of Craiova/Department of Electromechanical, Environmental and Industrial Informatics,
107 Decebal Blvd., Craiova, Romania, florin.ravigan@ie.ucv.ro

Abstract – **In this paper is presented the hardware and software design of a remote controlled system which can have multiple destinations. In particularly, this system is used to create an online thermostat using a microcontroller with ARM Cortex M4 architecture and an application designed for Android operating system. The system was designed by authors from the ground with the objective to create a framework for wide range of applications in electromechanical domain with network communication capabilities. The main hardware element is the processing unit for which was chosen the STM32F407VGT6 microcontroller. This is based on a 32-bit architecture working at 168MHz frequency which allow very fast running several tasks: scanning of digital sensors and a touchscreen, achieve SPI communication with an Ethernet driver, run an algorithm for temperature control. All software routines for microcontroller were built to be easy to port in another applications being developed in C language and structured in reusable functions. The application for mobile devices was written in Java language and was compiled with latest Android Studio. The system is functional and can be used in multiple applications offering the advantages provided by network connection. All presented stuff can be used to create a lot of network connected applications.**

Keywords: *microcontroller; ARM; Android; temperature; sensor; software; network, touch screen.*

I. INTRODUCTION

In the present paper is described the components and steps to achieve a temperature monitoring system with the possibility of local and remote display. These are possible locally via a LCD and remotely via a client as an Android application. The system's structure is presented by Figure 1.

The main component consists in a development system with a microcontroller from STMicroelectronics. This microcontroller is STM32F407VGT6 and has a Cortex-M4 architecture with a 32-bit bus and a working frequency of 168MHz which gives a high processing capacity, which can be used for running some complex code sequences written in C language or assembler.

These microcontrollers have a hardware architecture that allows development of command and control systems with high performance in processing power and the stability.

It can scan, considering the limited hardware resources, a number of sensors, both digital and analog, and can activate a series of analog or digital outputs, depending on needs.

The information is displayed locally on a color LCD or are available upon request via an Ethernet interface, to a remote client.

This system is also equipped with a resistive touch screen, which makes the interaction more user friendly.

For remote interaction, the microcontroller runs a routine that when receives a request, serves an HTML code that incorporates the package of information.
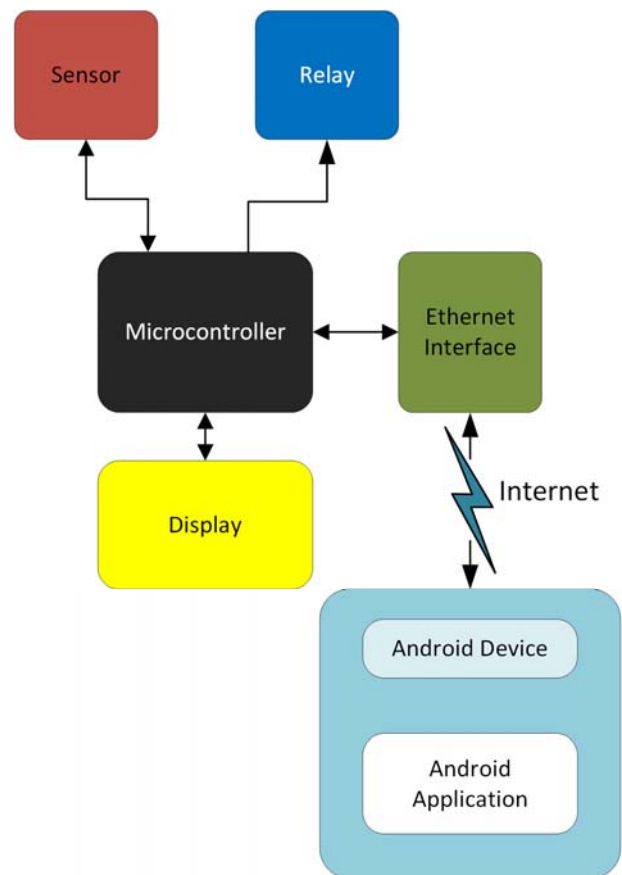

Fig. 1 The system's structure.

The application of this system is an online thermostat which will generate the command according to the control algorithm of temperature for a central heating unit.

A. The Microcontroller

The used system is a compact development platform developed around a microprocessor with ARM Cortex M4 architecture STM32F407VGT6 family produced by ST Microelectronics. This electronic board is produced by MikroElektronika [2].

This system also has integrated:

• a resistive touch screen with a resolution of 320x240 pixels;

• accelerometer that can be used for detecting vibrations in applications;

• a slot for a micro-SD card useful to store log files or media resources;

• USB interface for programming or serial communication with another device;

• 8-bit flash memory to save data which can be restored at reboot;

• Mp3 decoder that can be used for playing various acoustic warning signals;
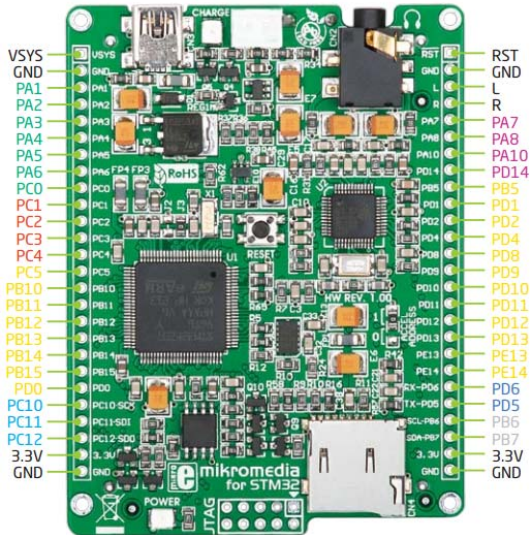
• Two connectors with 26 pin.

Fig. 2 The microcontroller board pin outs.

The STM32F407VGT6 model is a microcontroller produced by ST Microelectronics and has ARM Cortex-M4 architecture [5].

This microcontroller dispose of:
• 32-bit kernel capable of 1.25 million instructions per megahertz;
• 196 kB RAM;
• 83-pin input / output;
• SPI interface, I2C, CAN, USB, Ethernet;
• USART, UART;
• Timers 16 and 32 bits up to 168MHz;
• 16MHz internal oscillator, 32 kHz, PLL;
ADC, DAC, etc.

B. The tactile display

The ILI9341 display is a touch screen having a diagonal of 3 inches, 320x240 pixels resolution and a color depth of 262,000 colors [11][12]. Over LCD is a resistive layer that enables detection of coordinates relative of touches which are used for user interaction.

C. The Temperature sensor

The one-wire temperature sensor DS18B20 is produced by Dallas Semiconductors and is a sensor which do not requires any other components to measure the temperature (except a pull-up resistor between data terminal and Vcc) and it can measure values between -55 °C and +125°C with an error of ± 0.5%[4].

It is developed both TO-92 and μSOP capsule.

The pin description:
• GND - ground
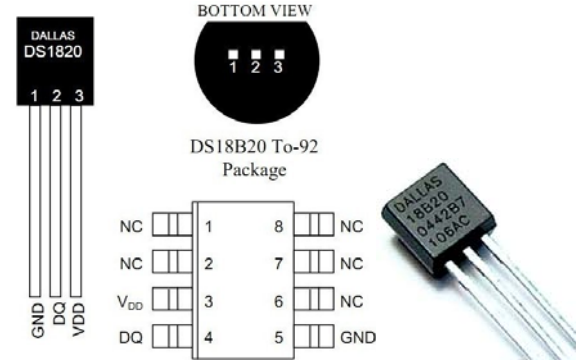• DQ - Data In/Out
• VDD – power supply
• NC – not connected

Fig. 3 The temperature sensor.

The resolution of this sensor can be selected by software between 9 and 12 bits. The conversion of temperature to digital can be done in maximum 750ms at 12 bits resolution.

To function, the sensor needs a 4k7 resistor connected between VDD pin and DQ. Data terminal is connected to the microcontroller pin PB10.

D. Ethernet Interface

To provide the remote control for that system, it was designed an Ethernet interface. It features ENC28J60, a 28-pin, 10BASE-T stand-alone Ethernet Controller with an on-board MAC & PHY, 8K Bytes of Buffer RAM and SPI serial interface [1].
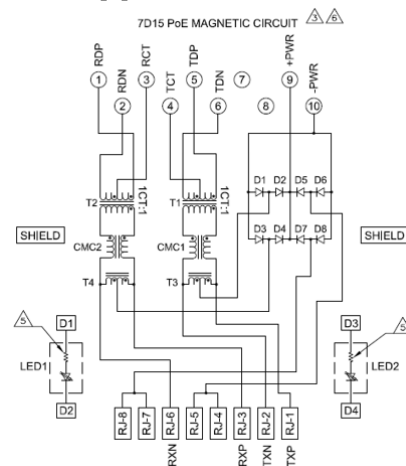

Fig. 4 The RJ-45 connector.

Chip supports programmable automatic retransmit on collision and automatic rejection of erroneous packets. Board contains standard RJ-45 connector (which has built-in the magnetic coupling elements and the protection diodes), transmit, receive and power LEDs [8]. The usage of an on-board 25 MHz crystal oscillator ensures a very stable operation.

The schematic of interface is presented below.

The connection with microcontroller board is made using the following table, the communication being carried out by SPI serial interface.

On the left side are the Ethernet interface pins and in the right column is the microcontroller board.

The main disadvantage of ENC28J60 chip is that it does not support DHCP address allocation, the IP address must be provided at initialization sequence [6][9].
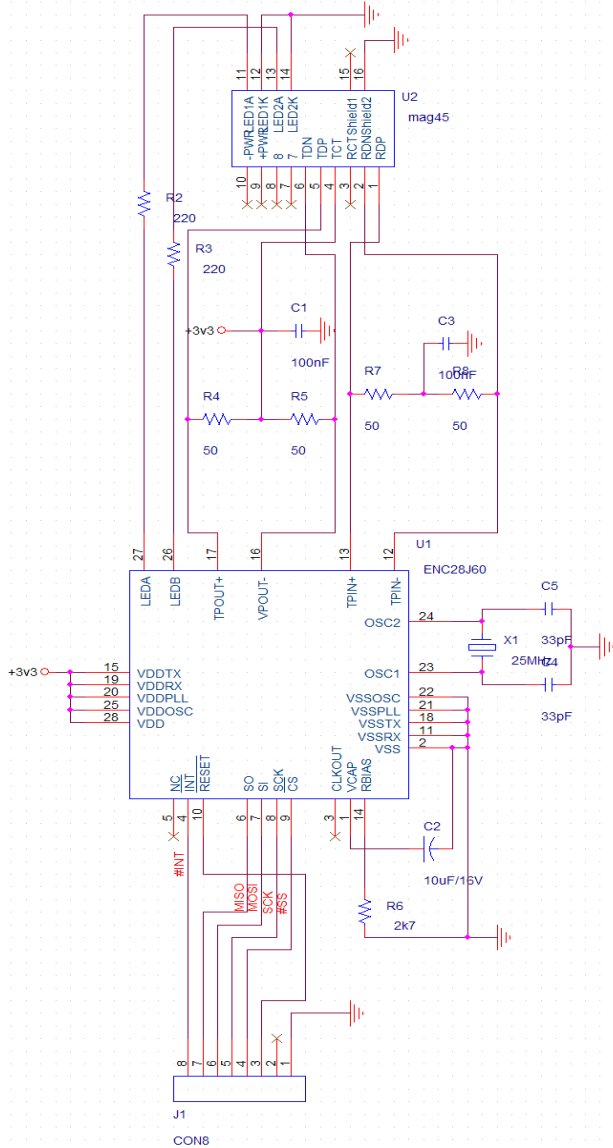
TABLE I.
THE CONNECTIONS BETWEEN ETHERNET INTERFACE AND MICROCONTROLLER

Ethernet Interface	Microcontroller
/RST	PE13
/CS	PE14
SCK	PC10
MISO	PC11
MOSI	PC12

This network interface can be used with any microcontroller with SPI communication port.

E. The relay

The relay was designed using a MOSFET transistor(IRL530N). The command of this is made using the pin PD1 directly connected to gate terminal.

This solution eliminates the classic relay and it makes the system usable in explosives environments.
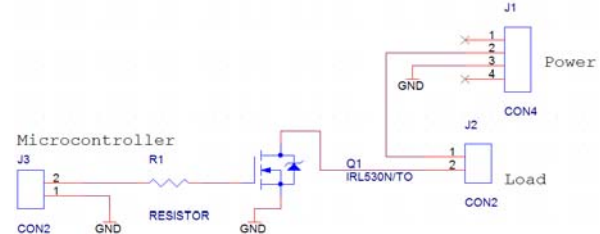

Fig. 7 The Ethernet interface schematic.

II. THE SOFTWARE PACKAGES

The entire application contents multiple software packages:

- for the microcontroller are developed two packages: one that draws on the touch screen graphical interface and one that runs the whole algorithm;
- the application for Android device.

A. Microcontroller's Software

The entire program for the microcontroller was programmed using C language and must fulfill the following tasks[9]:

- It initializes the hardware resources used in the application;

- creates the graphical user interface running on the touch screen;

- scans and extract information about temperature from digital sensor;

- it sets up and maintaining Internet communications network and serves the data packages (this part includes elements of HTML and JavaScript languages)[10].

To achieve these tasks were developed separate software routines. The one that makes the graphical user interface consists of three screens.

The first screen is primarily designed as a menu through which the user can set the operation of the system and view the screen with information about the application(Fig.8).

Fig. 5 The Ethernet interface schematic.

Fig. 6 The Ethernet board.

Fig. 8 The main and About screens.

In this window, the user can choose to see the screen with information about application (the author and build version) or he can move to temperature screen. Pressing the button "Temperature" the screen will change with another window which will display the value of temperature provided by DS18B20 sensor and the status of relay.

The graphical interface was maintained very simple to boost the performance of whole software package, that part being the largest consumer of processor time.
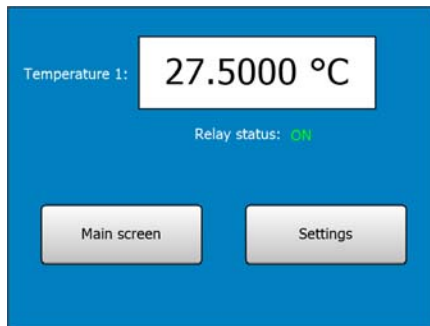

Fig. 9 The temperature screen.

Tactile surface is scanned as an interruption. Every interruption is evaluated and if the coordinated coincide with the declared areas a handler is called. This routine will load a simple function which will call another graphical display or change a variable in the main program.

It should be noted that the touchscreen calibration software was implemented in initialization routine without the need to be done every restart as normally required.

The button "Settings" will display the screen from figure 9. Here, the user can preset the desired value of temperature.
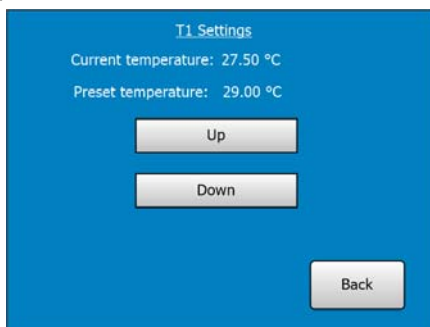

Fig. 10 The settings screen.

The next task of microcontroller is to scan the sensor. This part of program respects the indications provided by DS18B20 sensor's datasheet. The sensor is initialized, is interrogated and the provided information is converted to a readable form.

The next part of software is dedicated to network

communication. Here are the steps algorithm to achieve that task:

- microcontroller configure SPI communication;
- sets the configuration registers of the circuit ENC28J60(addresses, mask, gateway, port)[9]; the default port 80 can be replaced by security reasons with another.
- it is expected to receive a network packet;
- Package type is detected;
- If the protocol is IP header is extracted it;
- If the type is TCP packet handler is calling the package.

Web page served by the microcontroller is designed as a constant strings containing HTML and JavaScript calling variables from program code.
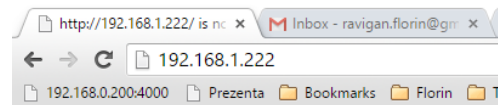
To serve and display the temperature, the JavaScript code is something like this:

$<script>document. write(T\_) </script>$

where "T_" is a variable from program.

The main function repeats in a while loop:

- it generates the graphical user interface and reads the user interaction;
- it scans the temperature sensor and makes conversions of temperature's value to float and to string;
- listen interrupts generated by SPI interface used for communication over the network and serves the required information.

Fig. 11 The temperature information in a web browser.

The information provided by microcontroller can be visualized also in a web browser typing in address bar http://ip_address_of_microcontroller. That interface has the possibility to set the desired temperature and is self-refreshable at every 5 seconds.

Also, the microcontroller permits the view of system's status adding to address the parameter "/s" (Fig. 10).
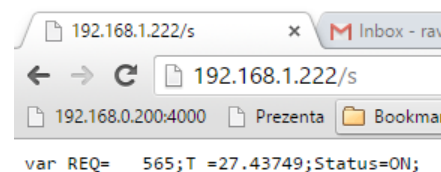

Fig. 12 The system's status in browser.

The microcontroller's port D is configured as outputs. All bits can be toggled using after address the parameters "/t0"…"t7", except the PD1 where is connected the relay. PD1 will be maintained in the status required by the control algorithm.

By the software code the command can be generated by an "and" logical operation between PD1 and another port D pin, which will make possible to manual override the status of the system.

B. Android Application

The Android application was developed using Android Studio IDE. The entire code is written in Java and XML. It is build using Android 5.1.1 API, but it can be run on any device having Android 4.0 and newest. Also, the application is resolution's device independent because it was built using relative layouts [3].

The application needs a single declared permission: full network access.

This application consists of two activities which are drawn two screens:
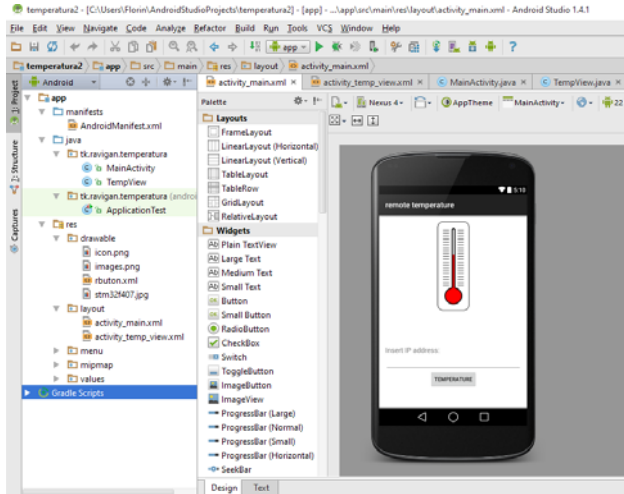

Fig. 13 The Android Studio Project.

- First activity is the one in which the user enters the IP address where is the microcontroller. This is sent to second activity to be processed.
- The second is the one in which are displayed information about temperature provided by the microcontroller and the status of command.

The second activity of the application is based on a WebView element from android.webkit.WebView library [7].

The application has for a built-in status notification system. This notification will appear in status bar every time the command turns on.

In Figure 12 the notification is the red icon. When scroll down the notification drawer the user can view the details and with swipe action can dismiss it (Fig. 15). If a notification is displayed it will be not multiplied.

After turn on the power, the system has a one second delay to initialize. The tests of viability were made in a network with 100Mbps bandwidth loaded over 90% with controlled traffic using a low cost routing equipment.
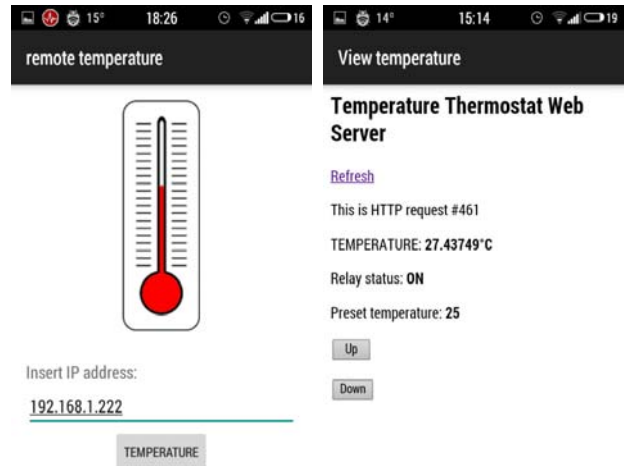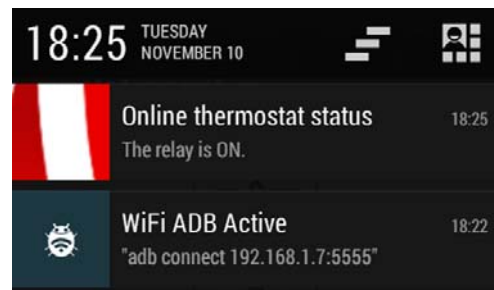

Fig. 14 The screens of Android application.

Fig. 15 The notification from mobile application.

III. TESTS

Because the transmitted packages are very small (under 1 Kbytes) the communication was very stable with 0% loss.
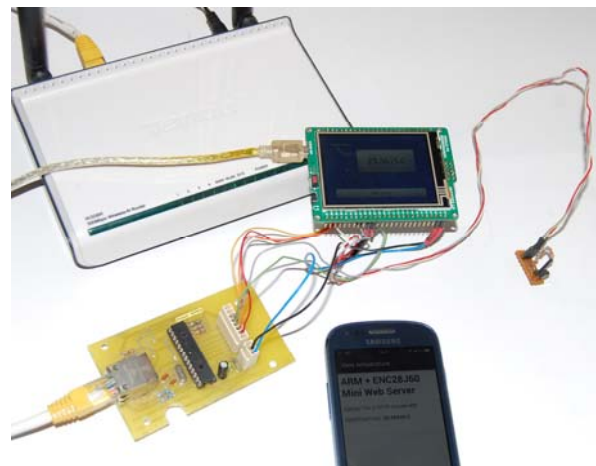

Fig. 16 The whole system.

The system used for application development can be seen in the image below.

Also, the information provided by microcontroller can be viewed in a web browser on any operating system, just typing the IP address.

IV. CONCLUSIONS

The presented system offers an effective solution, at a low-cost, robust in operation. Following tests were carried out revealed the following conclusions:

- The system is relatively inexpensive offering outstanding performance;
- the graphical interface is very responsive;
- Written routines are optimized for driving speed and can be easily ported to other microcontrollers in the family STM32F4xx and not only.
- The system can be used to control and various electromechanical systems using a feedback from a sensorial system.

The application can be developed as needed by making the following additions:

- it can achieve an interface through which you can specify the alarm temperature;
- it can implement a routine to store a history of temperature's values;
- EEPROM memory usage of the system microcontroller to save various settings;
- Using micro-SD slot, the system can save log files for later analysis.
- All packages can be encrypted and signed to avoid corrupted information.
- The network card can be replaced by o GSM module which can be connected easier with microcontroller by serial port.

In this case, the system connects to Internet using 3G or 4G technologies offered by GSM networks, becoming in this way a wireless mobile device.

REFERENCES

[1] F. Ravigan, N. Boteanu, A.M. Drighiciu, "Ethernet interface for a system of automation with microcontroller, " *Annals of the University of Craiova, Electrical Engineering series*, No. 35, 2011; ISSN 1842-4805.

[2] F. Ravigan, N. Boteanu and L. Alboteanu, "Remote control of a positioning system," *2nd International Conference on Systems, Control and Informatics (SCI 2014) EUROPMENT CONFERENCES International Athena*, 2014.

[3] L.Darcey, S. Conder, "Teach Yourself Android application development in twenty-four hours", 2012.

[4] *** - DS18B20 Programmable Resolution 1-Wire Digital Thermometer Datasheet.

[5] *** - http://www.mikroe.com/mikromedia/stm32-m4/

[6] *** - http://www.microchip.com

[7] *** - http://developer.android.com

[8] *** - http://www.digikey.com/catalog/en/partgroup/mag45-series/

[9] *** - http://www.mikroe.com/download/eng/ documents/ compilers/ mikroc/pro/pic/help/spi_ethernet_library.htm

[10] *** - http://blog.tkjelectronics.dk/2012/08/ethernet-on-stm32f4 discovery-using-external-phy/comment-page-1/

[11] *** - https://www.adafruit.com/datasheets/ILI9341.pdf

[12] *** - http://stm32f4-discovery.com/2014/04/ library-08-ili9341 - lcd-on-stm32f429-discovery-board/