

SINGLE-BOARD SERVER AND SOFTWARE APPLICATION USED IN WEATHER STATION

Kamen Krastev¹, Tsvetomir Gotsov², Anelia Manukova³

¹University of Ruse, Faculty of EEA, Department of Electronocs, 8 Studentska Street, 7017 Ruse, Bulgaria

²University of Ruse, Faculty of EEA, Department of Electronocs, 8 Studentska Street, 7017 Ruse, Bulgaria

³University of Ruse, Faculty of EEA, Department of Electronocs, 8 Studentska Street, 7017 Ruse, Bulgaria

Abstract

The purpose of this article is to represent and summarize information about the use of LAMP (LINUX, Apache, MySQL, PHP) in non-standard systems. The aspects explored are the “Open Source” and “Open Source Hardware” platforms and their application to achieve more reliable electronic systems for collecting and processing data with low cost. Modern high-tech systems are designed with specific objectives - to be flexible and easily adaptable to the rapidly changing environment. The proposed version of a web server, located on a single-board computer, is mobile. It is equipped with a number of standard communication interfaces. It can be applied either as a field device situated in the environment under measurement, or as a laboratory unit located close to the user. The software, implemented in the server, is based on high-level scripting languages. This allows processing of data with various mathematical, statistical and probabilistic models.

Keywords: web server, single-board computer, data analysis, statistical methods, user experience

1. INTRODUCTION

With the development of ARM processor single-board computers are becoming more affordable in terms of price. Their productivity is steadily increasing, but it has not reached that of the established platforms such as Intel or AMD. ARM platforms are not the typical embedded systems. Sample hardware parameters of such devices are: 1GHz - operating frequency of the processor, 512 MB RAM volatile working memory, 4GB NAND Flash, etc. Despite their lower parameters, they are suitable for many applications, fairly well replacing conventional computers. Most of them do not require forced cooling, which makes them more reliable, quieter and energy-saving. The average consumption of such a device is about 5W.

1.1 Structure and Design

The output parameters of the system are mostly a part of the generally accepted weather components, together with a group of specific elements, closely related to the particular application of the station. The standard weather elements, monitored by the system, are air temperature, air pressure, relative humidity, wind direction, wind speed, rainfall, etc. The specific weather elements are: soil temperature measured at 5 depths, volumetric water content, etc. The interval between records in the data base is one minute under the standard D6176-97(2003)^[4]. The designed system is based on a single-board computer Olinuxino A13 with an open source hardware.

1.2 Hardware

The proposed system is flexible, which allows its adaptation to an advanced PCB for a corresponding processor.

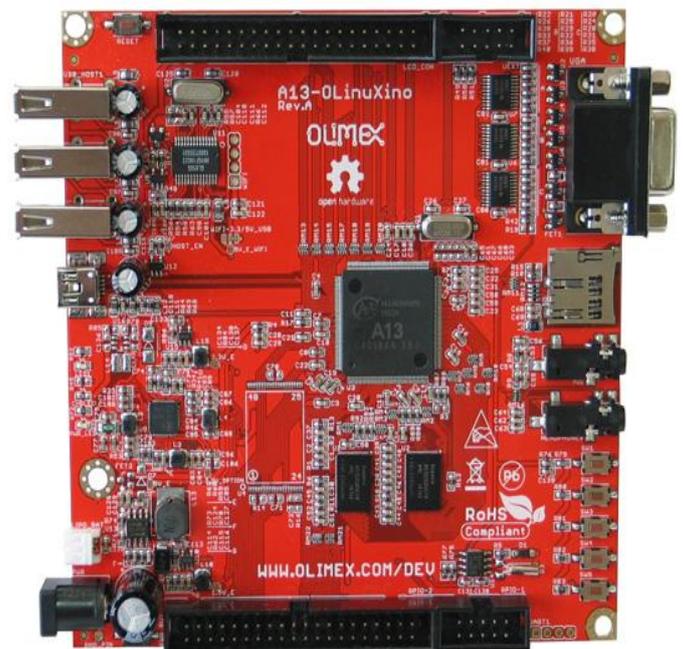


Fig. 1. Hardware view of Olinuxino A13 – WiFi

Figure 1 illustrates the exterior of the hardware manufactured by Olimex. Olinuxino-WiFi is a single-board computer, competitive to the famous Raspberry Pi and BeagleBone. It contains standard interfaces, typical to personal computers, such as VGA and USB. It is also fitted with UART and GPIO which are commonly found in microcontrollers. The board has UEXT (Universal EXTension connector) interface, which is specified by Olimex Ltd and contains the three communication interfaces most common to microcontrollers. These are: I2C, and asynchronous serial communication SPI.

1.3 Software

The software is based on the popular platform for web applications LAMP. LAMP is a set of software products designed to provide everything necessary for the operation of a web application.

The word is formed as an acronym for:

L – Linux: Operating System;

A – Apache: Web server;

M – MySQL: Relational database;

P – PHP: Scripting language.

Figure 2 illustrates the structure and principle of operation of the software.

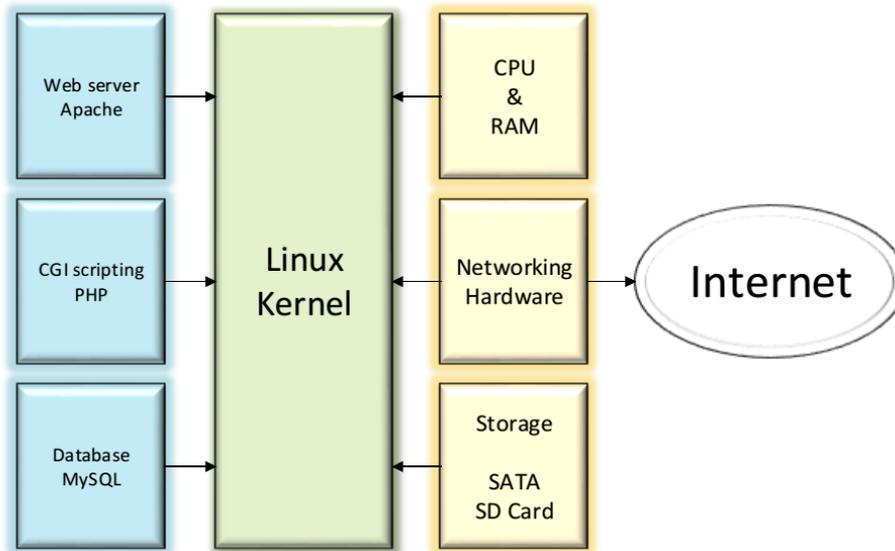


Fig 2 Structure of the single-board server

1.4 Special Application for Data Acquisition

The application which runs on the hardware and software platform, is linked to the data collecting device via standard internet protocols (TCP, UDP, FTP, etc.). Several devices connected as a distributed system are used to collect data. Thereby, each measurement is topologically performed in order to cover a geographic area. Data is stored at a central point. In this case, the application utilizes a TCP protocol and opens a new socket for each measuring device connected to the server. In practice, the communication channel includes drives which support TCP / IP STACK.

2. SOFTWARE ENGINEERING

The structure of the application is designed to meet the requirements for data collection on weather elements. One of its primary tasks is to draw graphs of the indicators relevant to the user. The application can also present data in tabular form. It is able to export data so that data becomes accessible by other specialized software products.

2.1 Architecture

At the base of the web application is the MVC (model view controller). Figure 3 represents the working principle of MVC.

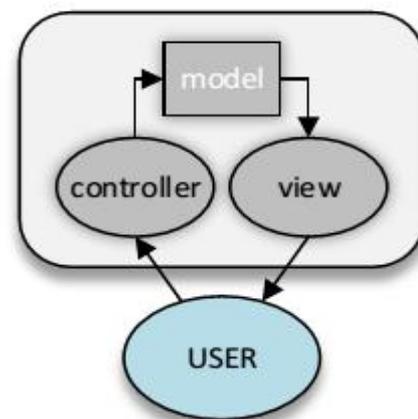


Fig. 2. block diagram of MVC

Model View Controller (MVC) is a software architectural pattern that separates the representation of information from the user interaction with it.

The model consists of data, business rules, logic and functions. A view can be any representation of data to the user, such as a table or chart. Multiple views of the same data are possible either as a bar graph or as a table. These can be suited to the needs and requests of the user. The controller is the mediator between the user and the application, since it translates the user’s request on the input into a valid model or view on the output.

Although it was originally developed for personal computers, nowadays MVC is widely accepted as the architecture of web applications in all major programming languages. In this particular case, the script for the logical part of MVC, the controller and the model is written in the programming language PHP. The view is created in HTML and JS.

The MVC organizes and distributes tasks between the 3 basic components of the application:

- The controller (login.php) sends commands to the model (login - function) in order to update its state. Moreover, it can send commands to the view to change the way information is presented (login_form.php).
- When there is a change in its state, the model notifies the view (login_form.php – username and password fields) and the controller (login.php – to execute a query to the database for a user). This notification allows the view to send an output stream to the controller. The controller updates the output and all available commands.
- The view collects information from the model in order to create the HMI. It highlights the corresponding text fields in green for a successfully registered user.

2.2 Mathematical and Statistical Modeling

One of the most crucial aspects of software applications is that mathematical and statistical methods of analysis can easily be implemented into them. The software environment and programming languages are not specialized for mathematical modeling, but are flexible and easily customizable.

In probability theory standard deviation measures the amount of variation or dispersion from the average. The following formula is used to implement standard deviation:

$$\sigma = \sqrt{\frac{\sum t \ln X(t-\mu)^2}{N}}, \quad (1)$$

- The standard deviation function is described in the programming language PHP and is presented in Fig. 5.

```
function standard_deviation($X, $bSample = false)
{
    $fMean = array_sum($X) / count($X);
    $fTotal = 0.0;
    foreach ($X as $i)
    {
        $fTotal += pow($i - $fMean, 2);
    }
    $fTotal /= ( $bSample ? count($X) - 1 : count($X) );
    return (float) sqrt($fTotal);
}
```

Fig. 5. Standard deviation script in PHP

Random processes and events frequently occur when monitoring technological processes. In probability theory a stochastic process or random process is the result of a deterministic process (or deterministic system). Instead of working with only one possibility of how the process might evolve over time (such as in the case of solutions to the ordinary differential equation), in a stochastic or random process there is an uncertainty about its future development (evolution) described by probability distributions. This means that even if the initial condition (or starting point) is known, there are many possibilities for how the process might evolve, but some results may be more likely than others.

In order to implement such type of algorithms in the software, it is necessary to use a pseudo-random number generator. In high-level programming languages these functions are used as a predefined block rand ().

The need for additional data channels and transfer of information to other devices makes FTP data transfer particularly necessary.

```
$ftp_username = 'user';
$ftp_userpass = '123456';
$ftp_server = '192.168.0.10';

$ftp_conn = ftp_connect($ftp_server) or die("Could not connect to $ftp_server");
$login = ftp_login($ftp_conn, $ftp_username, $ftp_userpass);
ftp_pasv($ftp_conn, true);

$local_file = 'data1.csv';
$server_file = 'data_one.csv';
// download server file
if (ftp_get($ftp_conn, $local_file, $server_file, FTP_ASCII)){
    echo "Successfully written to $local_file.";
}
else{
    echo "Error downloading $server_file.";
}
ftp_close($ftp_conn);
```

Fig. 6. FTP connection script in PHP

The function shown on Fig. 6 establishes a connection to the FTP server. In this case, the software application is a "client" to another data server. The parameters, providing access to the server, are indicated at the beginning. The client application tries to connect the user to the server after acquiring user authorization. The relationship mode is passive. After data is downloaded files are automatically renamed to ensure accurate and flawless further use. If the connection fails the system attempts to reconnect after a specified period of time.

2.3 User Experience

A web application is developed to improve the interaction between human and machine. The authorization system allows personalized access and settings for different users. (Figure 7) HMI is consistent with modern views for aesthetics and simplicity.

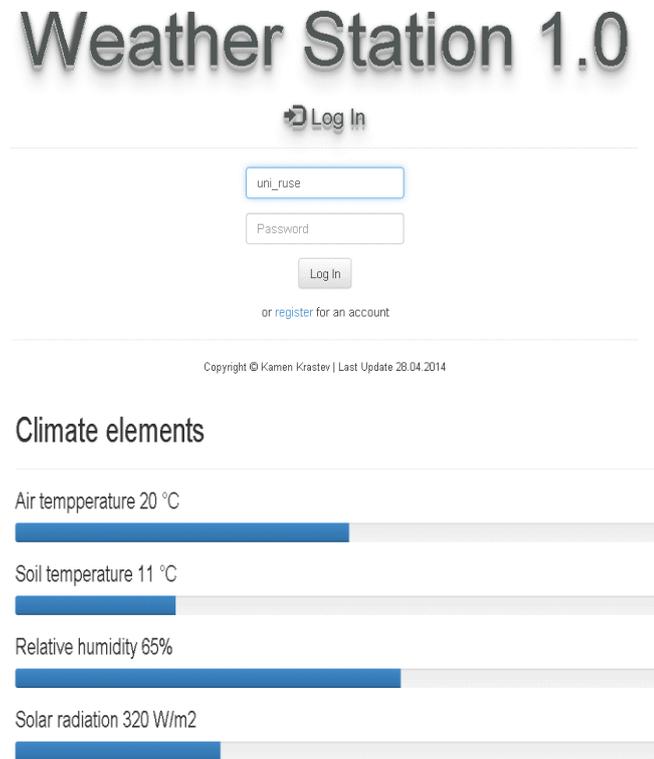


Fig. 7. User interface

The diagram on the right side of Figure 7 shows in an analog way some basic weather elements. As a general rule human perceptions respond better to graphics and diagrams than to numbers. The software templates are functional. They help the user to experience intuitive perception.

3. CONCLUSION

The software application is designed according to the needs of weather stations. Two mathematical and statistical models are implemented, enabling data storage and analysis. The application makes use of an advanced approach to display data. Data is stored in standard formats and can easily be exported to other media or systems. A single-board computer with an ARM processor is responsible for executing the software. The main advantage of the proposed system is that it uses a development board with open source hardware. The universal PCB is subject to modifications so that it meets the requirements for robustness and electromagnetic compatibility. The single-board computer has low energy consumption which makes it suitable for powering a variety of autonomous systems - photovoltaic, electrochemical, etc.

ACKNOWLEDGEMENTS

The present document has been produced with the financial assistance of the European Social Fund under Operational Programme “Human Resources Development”. The contents of this document are the sole responsibility of “Angel Kanchev” University of Ruse and can under no circumstances be regarded as reflecting the position of the European Union or the Ministry of Education and Science of Republic of Bulgaria.

Project № BG051PO001-3.3.06-0008 “Supporting Academic Development of Scientific Personnel in Engineering and Information Science and Technologies”

REFERENCES

- [1] Van der Geer, J., Hanraads, J. A. J., & Lupton, R. A. (2000). The art of writing a scientific article. *Journal of Scientific Communications*, 163, 51 - 59.
- [2] Strunk, W., Jr., & White, E. B. (1979). *The elements of style* (3rd ed.). New York: Macmillan.
- [3] Mettam, G. R., & Adams, L. B. (1994). How to prepare an electronic version of your article. In B. S. Jones, & R. Z. Smith (Eds.), *Introduction to the electronic age* (pp. 281-304). New York: E-Publishing Inc.
- [4] ASTM Standard D6176-97(2003). "Standard practice for measuring surface atmospheric temperature with electrical resistance temperature sensors"; prepared by Committee D22.11 on Meteorology, ASTM International, West Conshohocken, PA. 2003.
- [5] Krastev K., Gotsov Ts., Manukova-Marinova A. (2014). Embedded System for Measurements and Data Acquisition with Web Interface Based on ARM Processors, *Annual journal of ELECTRONICS*, Technical University of Sofia, 2014, pp. 204-207
- [6] Levski D., Gotsov Ts., Krastev K., Manukova A., Dimov V., (2011). Embedded Automatic Weather Station. *Elektronika Magazine*. Poland, 10.2011. pp. 159-163
- [7] World Meteorological Organization. (2008). *Guide to Meteorological Instruments and Methods of Observation*.
- [8] J. C. Jensen, E. A. Lee, and S. A. Seshia, “Teaching Embedded Systems the Berkeley Way,” in *Workshop on Embedded Systems Education* (in conjunction with ESWeek), Tampere, Finland, October 2012.
- [9] P. Mitros, K. Afridi, G. Sussman, C. Terman, J. White, L. Fischer, and A. Agarwal, “Teaching Electronic Circuits Online: Lessons from MITx’s 6.002x on edX,” in *International Symposium on Circuits and Systems (ISCAS)*. Beijing, China: IEEE, May 2013.
- [10] Dan Eisenreich, Brian DeMuth. *Designing Embedded Internet Device*. 2003, Newness.