

REMOTE EXPERIMENTATION: INTEGRATING RESEARCH, EDUCATION, AND INDUSTRIAL APPLICATION

Juarez Bento da Silva¹, João Bosco da Mota Alves², Gustavo Ribeiro Alves³

¹ Southern University of Santa Catarina, Araranguá, SC, Brazil, juarez@unisul.br

² Federal University of Santa Catarina, Florianópolis, SC, Brazil, jbosco@inf.ufsc.br

³ Polytechnic Institute of Porto, School of Engineering, Porto, Portugal, gca@isep.ipp.pt

Abstract: This paper presents a low-cost scaled model of a silo for drying and airing cereal grains. It allows the control and monitor of several parameters associated to the silo's operation, through a remote accessible infrastructure. The scaled model consists of a 2.50 m wide x 2.10 m long plant with all control and monitor capacities provided by micro-Web servers. An application running on the micro-Web servers enables storing all parameters in a data basis for later analysis. The implemented model aims to support a remote experimentation facility for technological education, research-oriented tutorials, and industrial applications. Given the low-cost requirement, this remote facility can be easily replicated in other institutions to support a network of remote labs, which encompasses the concurrent access of several users (e.g. students). *Copyright © 2007 IFAC*

Keywords: Remote Experimentation, embedded web servers, Micro WEB-Server, Remote Monitoring and controlling.

1. INTRODUCTION

Technology is driving newer and more innovative ideas within informatics and computer systems. Recent examples of online monitoring and control support this opinion. In fact, online monitoring and control is being widely studied and applied in many different areas such as: traffic control, Earth phenomena (e.g. earthquakes), weather, industry, and many others. In parallel and motivated by the growing popularity of the Internet, many companies working with embedded systems are now offering micro Web servers that allow monitor and control operations via the Internet, at low cost. Printers, routers, and webcams are some examples of devices that support web-based configuration and status display (Bento, 2002). Devices incorporating micro Web servers carry many advantages, e.g. the ability to be remotely accessed through the Internet, using a simple Web browser. The remote access supports several new functionalities, such as: failure detection and diagnosis, firmware update and remote maintenance, among others [Ford, 2003]. These new functionalities may even provide a considerable reduction on maintenance costs, if the system under maintenance is built with such devices.

In the following sections we describe an example of such a system built with devices incorporating micro Web servers. The system in question corresponds to a low-cost scaled model of a silo for drying and airing cereal grains. It may be used for technology education, research, or industry-oriented vocational training. One main advantage is its low-cost that allows an easy replication by other Institutions or Organizations.

2. SYSTEM COMPONENTS

A remote system may either support failure diagnosis, device control, firmware updates, and other similar functional aspects that usually imply a reduction on pre-emptive and corrective maintenance costs. Using the Internet as the remote access media has also enabled a considerable reduction on such costs, due to its widespread acceptance and support. Remote experimentation has therefore emerged on that line of reasoning. As a system, a remote experiment is usually composed of one or more core components and one or more additional components that depend on the subject or object under experimentation. The following sections will thus describe the components that form our remote experiment, in particular.

2.1. Micro Web Server (hardware)

The core component of our remote experiment is a Micro Web Server (MWS) built in-house at the RexLab (a joint venture of the Federal University of Santa Catarina and the University of Southern Santa Catarina, both at Brazil). A MWS may be described as an autonomous component able to connect electrical / electronic devices to an Ethernet-based network (such as the Internet). It is built around a low-cost, low-power Atmel AT90S8515 microcontroller, which supports the TCP/IP protocol (ATMEL 2001; Celso 2000; Intel Corporation 1989). The microcontroller in itself runs at 8 MHz and contains three internal memories: (1) static RAM, (2) programmable Flash memory, and (3) EEPROM. The block diagram illustrated in figure 1 depicts the microcontroller and the surrounding components that form the MWS. The right part of the block diagram illustrates the interconnections with the outside world.

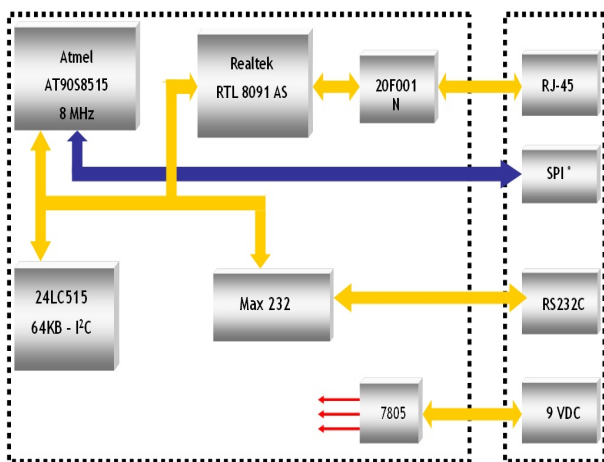


Fig. 1: Block diagram of the RExLab MWS

A serial I²C EEPROM, with 64 kbytes, is used for storing the code and the file system of the MWS. The physical interface with the Ethernet is done through a RealTek RTL8019AS device, which supports the Ethernet 802.3, full duplex communication mode. The default interface is done via a 10BaseT (RJ45) port, although it also supports an AUI port (optical link). Finally, the MAX 232 is a driver/transceiver for serial, RS-232C, communication.

2.2. Micro Web Server (software)

The software developed for our MWS includes some firmware (e.g. I/O functions) and the application code. The firmware is mainly based on a simple kernel, which structure is illustrated by figure 2. The RealTek device implements the two bottom layers, leaving the remaining ones for the microcontroller. The layer on the top is related to the web http application server, while the remaining ones (network and transport) contain the TCP/IP stack and the network adapter drivers.

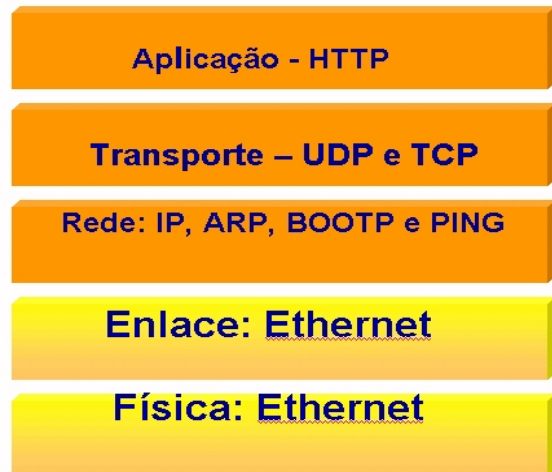


Fig. 2: MWS firmware (layers from top to bottom: application, transport, network, logical, physical)

The Address Resolution Protocol (ARP) is also handled within the MWS, which means associating an Ethernet address to an IP address. A unique Ethernet address is allocated to each MWS. The IP address may be obtained in a static or dynamic form. In a static form it is determined in advance and then stored inside the microcontroller flash memory. In a dynamic form it uses the BOOTP protocol, to repeatedly ask for a valid IP address. The MWS also responds to ICMP (Internet Control Message Protocol) Echo Requests (normally associated with the “ping” command), which allows an easy way to measure the total RTT (Round Trip Time).

The MWS also supports UDP-type (User Datagram Protocol) packets, normally used by applications such as SNMP (Simple Network Management Protocol) and DNS (Domain Name System). At the TCP/IP level, our MWS responds to HTTP GET requests (normally issued by the web browsers), which are addressed to the TCP80 port. As any normal web server, our MWS responds to such requests by sending HTML (HyperText Markup Language) documents, text, or images. In all cases, the system kernel handles all the Internet protocols listed above. One restriction identified so far is the total length of one Ethernet packet (1400 bytes) for answering an HTTP GET request, in order to save memory resources. There are however several HTML coding techniques that allow working within this restriction, without affecting the quality of the service, namely HTML frames and multiple JPEG or GIF images. The MWS functionality is also easily upgradable, which allows for the integration of JavaScripts (either into the HTML code or as separate modules) and Java applets, or CGI (Common Gateway Interfaces) routines (Schmidt 2003; RABBIT Semiconductor 2003).

The main factor for keeping the system cost low is to use low-priced devices. An additional factor that influences the system cost is the use of open source tools for developing and debugging the application run by each device. These two factors were carefully observed in our

development process. The price of our MWS is around US\$ 30 and its size is 82 mm x 94 mm, as seen on figure 3. The total consumption (electrical current) is around 40 mA. These characteristics allow using the MWS with several objects, to provide remote control and monitoring (due to its reduced size and low cost). Given the past experience acquired with this MWS (e.g. remote temperature sensing, as described in Bento, 2006), in small applications, we decided to build up a larger and more complex system, and made it totally controllable and observable through the Internet, by using our MWS. Our choice went for a scaled model (total dimension of 2.5 x 2.1 meters) of a grain storage tower (i.e. a silo), which is described in more detail in the following subsection.

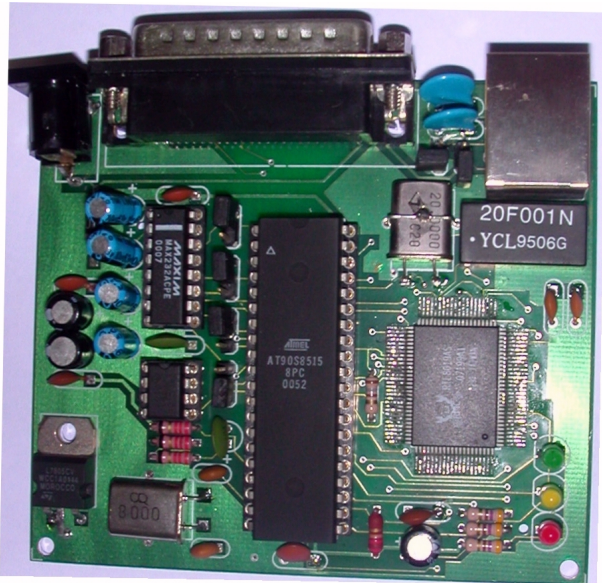


Fig. 3: Micro Web Server - MWS

2.3. The silos model

A general view of the developed model is illustrated in figure 4. As mentioned, the access to the silos parameters (both in terms of monitor and control parameters) is to be made through the Internet, using one MWS (i.e. typical client/server architecture). On the server side, the silos model contains one MWS that receives the temperature and humidity readings from a 10-bit, 8-channels, analogue/digital converter (ADC), connected to a series of sensors. The ADC is connected to the MWS through a serial RS-232C port. The client side is supposed to be any web browser supporting Java. The silos web interface comprises HTML with Java applets that allow a direct communication link with the MWS for the purpose of reading and controlling the models parameters. On the server side, there are CGI scripts running at the MWS that interpret all requests made by the user, when accessing the silos model through the web. Before presenting the details of the model operation, we describe in the following paragraph the structure and functionally of a conventional silo.

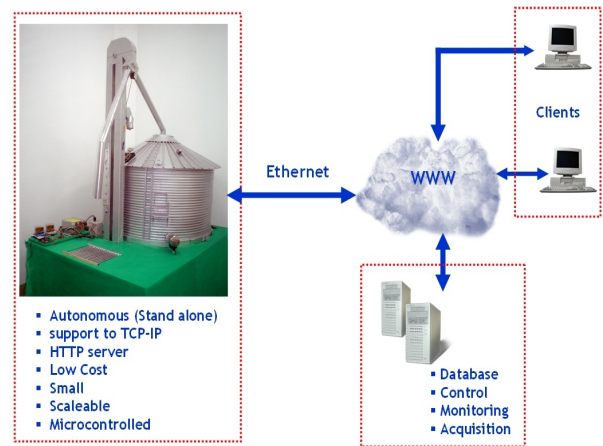


Fig. 4: General view of the model

A conventional silo is a metallic structure built on the top of a concrete basis. The silo is built on such a way that an interior air chamber is formed. One or more heaters and air pumps allow injecting hot air into the silos inside. The air trespasses the grains stored inside the silo, which impairs them from rotting. In Brazil, the most used silos are 11 metres diameter with a total storage capacity of 430 tons of grain. A detailed description of the silos operations is available from (Nisshin Engineering Inc., 2006). The model was thus implemented with the same functional characteristics of a normal silo for airing cereal grains, as our intention was to make available a remote experiment environment as close as possible to a real industrial environment.

3. MODELS OPERATION

Referring to the implemented model, the grains are dropped off into the respective silos compartment and then loaded into the internal containers by a bucket elevator standing in parallel with the silos body structure, which runs vertically (see figure 5-1 for a detailed view of the bucket elevator). The elevator drops off the grains into a tube that leads into the container. In our model, the grain is removed from the silos interior following a similar process, i.e. by controlling the direction of a servo motor (see figure 5-4 for a detailed view of its placement) it is possible to put grains into or out of the silo. Once the grain is stored inside the silo, internal sensors allow monitoring both the internal temperature and humidity. The readings may be remotely accessed using the functionalities provided by the MWS. The Parameters Control System (PCS) receives this information from the MWS in time intervals fixed by the systems administrator. In every new reading, the PCS compares the actual values with the ones defined as trigger levels. If for instance the actual temperature is higher than the limit set by the administrator, the PCS automatically turns on the aeration system. Once the temperature drops below that setpoint the PCS shuts off

the aeration system (see figure 5-3 for a detailed view of its placement).

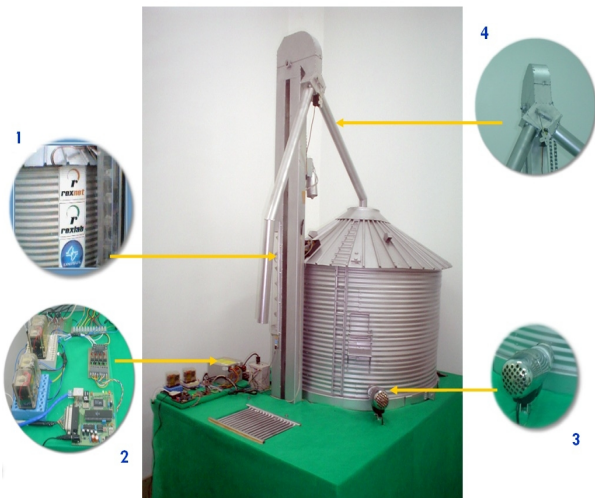


Fig. 5: Some details of the silos model

The model contains three temperature sensors (two located inside the silo and one outside) and two humidity sensors (again, one inside and one outside the silo). These sensors are connected to the ADC, which in turn is connected to the MWS that reads and displays the values. The readings are also stored by the MWS for later analysis, if needed. As depicted in figure 6, the MWS controls the aeration system, the bucket elevator and the grain flow direction through an external motors power board, clearly visible on figure 5-2. Figure 6 also depicts the grain level sensor that allows determining the used storage capacity (from 0 to 100%, in steps of 10%). This sensor is connected to one of the eight channels available at the ADC.

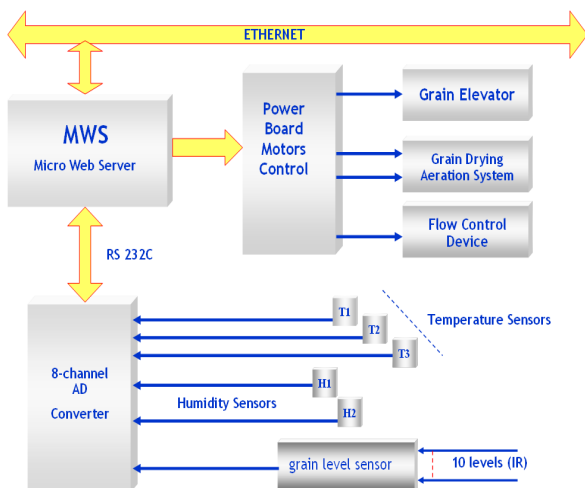


Fig. 6: Block diagram of the silos model control / observation mechanisms.

As all parameters are controlled and monitored by the MWS, the possibility of remotely accessing those

parameters lays on the creation of a simple webpage able to exchange information with the MWS. This possibility is described in the following section.

4. PRESENT DEVELOPMENT

The silos parameters can be monitored and controlled through a webpage located at the following URL <http://computacao.unisul.br/projetos/silo/>. This same page includes a visual feedback from the silos operation, provided by one of two existing webcams. Figure 7 corresponds to a print screen of a client presently accessing the system. The top left quarter contains an image of the silos interior, the tubes and the bucket elevator. The used storage capacity is also indicated (40%, on figure 7) by the coloured area. The image will also show movement if the bucket elevator is working. The top right quarter contains the image returned by the webcam. The current readings (internal temperature sensor 1 and 2, external temperature sensor, internal and external humidity sensors, and level sensor) are displayed on the bottom right quarter. The control parameters (temperature setpoint, aeration system, bucket elevator, and flow direction) are displayed on the bottom left quarter.



Fig. 7: Web page for controlling and monitoring the silos model.

Notice that there are two webcams transmitting live from the RExLab. The current image source can be selected from the webpage (webcam 1 – webcam 2). It should again be stressed the fact that all variables are not only observable but also controllable. In the present implementation there is a small animation that illustrates what is happening inside the silo, thus facilitating the comprehension of the silos operation. Although these graphical elements improve the overall aspect of the remote experiment interface, they adversely affect the access by disabled people, as indicated by the Web Accessibility Initiative, according to (The World Wide Web Consortium, 2006)

5. AN INTEGRATED APPROACH

There are many several examples of remote experiments available in the Internet and supported by the scientific and educational community^{1,2}. In addition, there are also examples of remote experiments aimed for industrial applications. The goal of integrating these three perspectives into one single remote experiment places new considerations for its specification. Although we can not claim that the remote experiment described in this paper was clearly developed with those three perspectives in mind, there is a set of characteristics that allow its general use. The next subsections (or paragraphs, to be more accurate) will therefore analyse the use of this remote experiment under each perspective.

5.1. Technological Education

The silo model can be used in local labs within technological courses (high school level), engineering courses (undergraduate level) or vocational training. In the first type we can consider courses for technicians working in maintenance services, or very simply as remote operators in industrial areas.

At the Higher Education level, the system may be used for lab classes in electronic, electrical, agronomy, computation, or automation courses. The system in itself, or its components, are useful for supporting theoretical contents on any of these courses.

5.2. Scientific Research & Development

The silos prototype is also adequate for supporting the test of new products used for conserving the cereal grains, as well as products used against fungus or bacteria that may attack the cereal grains. The reduced size and cost allows an easy replication that increases the number of products that can be tested, on site, in parallel.

5.3. Industrial application

In an industrial application environment, the control / observation mechanisms based on our MWS allow an easy and seamless path to remote operation. Not accounting for the modules that would be required for connecting the MWS to more powerful actuators and motors, the overall architecture is applicable to an industrial size silo. Moreover, we also think that new silos could be built in proportional sizes according to the quantity of grain stored / produced by small and medium enterprises. This would allow a more decentralized operation, with significant savings on transport costs. Again a control system based on our MWS could be used for such small / medium sized silos, with the immediate advantage of allowing Internet or Intranet

control / observation. Although a bit distant in time, we foresee that there will undoubtedly be an increase on the need for individuals trained in remote engineering as a result of a widespread use of MWS.

6. CONCLUSION

The implemented silos model illustrates the possibility of remotely controlling and monitoring complex (automated) systems, using a low cost approach based on MWS. Besides the cost aspect, the system is also accessible by any remote client using a simple web browser, which allows an easy and flexible use by any student or teacher. Given these two aspects, the implemented silos model can be easily replicated in other institutions to support a network of remote labs, which encompasses the concurrent access of several users for technological education, research-oriented tutorials, or even industry-oriented training.

ACKNOWLEDGEMENTS

The authors acknowledge the sponsorship of the European Commission, EuropeAid, Cooperation Office, Programme América Latina – Formación Académica (ALFA), through Project ALFA-II-465-A.

REFERENCES

- J. Bento S., 2002, "Monitoramento, aquisição e controle de sinais elétricos, via Web, utilizando microcontroladores.", Florianópolis-Sc, 110 p, Dissertation (MSc.)
- R. Ford, 2003 "Embedded System Monitoring Architectures and Costs", Department of Computer Science, IEEE, University of Kansas, pp. 237-244.
- J. Bento S., 2006, Remote experiments hosted by UNISUL, <http://computacao.unisul.br/projetos/>
- G. Schmidt, "Beginner's introduction to AVR assembly language". Available at http://www.dg4fac.de/avr/avr_em/>. Access in 2003, August.
- RABBIT Semiconductor. "Rabbit2000 TCP/IP Development Kit". Available at http://rabbitsemiconductor.com/products/rab20_tcpip/rab20_tcpip_dev_kit.html>. Accessed in 2003, July.
- Nisshin Engineering Inc., Grain silos system, at <http://www.nisshineng.com/eng/products/pdf/E31.pdf>>. Accessed in 2006, July.
- ATMEL 8-bit "AVR Microcontroller with 8K Bytes In-System Programmable Flash (AT90S8515)", Atmel Corporation, Updated September 2001, <http://www.atmel.com/atmel/acrobat/doc0841.pdf>.
- Celso Renato G., PROVIDELO, Estudo Básico da Arquitetura AVR. <http://july.sel.eesc.usp.br/documents/uisp>, 20 September 2000.

¹ <http://www.isepp.ipp.pt/swier06/>, accessed 2006, July.

² <http://lab.artec.uni-bremen.de/deriveSERVER/index.php>, accessed 2006, July.

Intel Corporation, 8-bit Embedded Controller Handbook, Santa Clara, 1989.

The World Wide Web Consortium (2006). Web Accessibility Initiative (WAI) <http://www.w3.org/WAI/>