PHP + APACHE + TESTPOINT = AN ORIGINAL WAY FOR HAVING REMOTE CONTROL OVER ANY TYPE OF AUTOMATION

Fl. MANEA[1], C. CEPIŞCĂ[2]

Lucrarea prezintă rezultatele activităţii desfăşurate pentru crearea unei soluţii ieftine necesară transmiterii web a informaţiilor. Sunt studiate problemele teoretice şi practice privind tehnicile de control şi sunt indicate elemente de proiectare. Ideea de bază a conceptului propus constă în utilizarea combinată a elementelor software pentru controlul şi monitorizarea oricărui proces de automatizare şi construcţia unei interfeţe web pentru orice tip de instrument de măsurare.

The paper contains a short report on a two-year work aiming to create a cheap and new type of web transmitting data. Control software and design problems are shortly presented. The main idea of this concept is the use of those combined softwares gives us a very flexible and powerful way to control and monitoring an automation process from anywhere and to build an web interface for any type of instruments.

Keywords: virtual instrumentation, remote digital control systems, web interface.

1. Introduction

Networked I/O is certainly not a new concept. Companies have relied on the capability to perform remote I/O over networks for years. Typically, communications between the remote I/O system and the host computer have been accomplished through a serial standard such as RS-232 or RS-422/485. These have generally taken the form of some type of sensor bus or device bus, such as CAN, Device Net and LON works [1]. These standards have advantages and disadvantages compared to Ethernet. Ethernet and Transmission Control Protocol/Internet Protocol (TCP/IP) offer performance and cost advantages over RS-232, RS-422, and GPIB interface technologies for distributed data acquisition applications. This is particularly true for large enterprises that must share measurement data across several departments. These protocols also make it

[1] Software Eng., Luxten Lighting Company SA.
[2] Prof., Dept. of Electrical Measurements, Electrical Apparatus and Static Converters, University "Politehnica" of Bucharest, Romania

possible to access data acquisition data over the Internet. Even though data acquisition hardware now is available with Ethernet connectivity, there still are many data acquisition cards, external data acquisition systems, and instrument racks set up the way they have always been—close to corresponding computer systems and the sensors and signal sources being monitored and communicating by older methods. Hereafter, we will update these old instruments, make them web compatible!

2. Traditional Approaches for Remote I/O

2.1 RS-232

RS-232 is by far the simplest and least expensive. Every PC shipped today can communicate over an RS-232 line. Only a compatible cable and a terminal program are required. RS-232 is limited to about 50 feet [2], but it can operate over huge distances with a little help from a modem. Only one device can be connected to a single RS-232 port. While virtually all computers have an RS-232 port, few have more than two. Most PC operating systems cannot handle more than four without special software.

RS-232 also is typically not electrically isolated from the host computer. For example, if a power line were inadvertently dropped across the RS-232 line, the host computer most likely would be destroyed. As a result, RS-232 may be appropriate for situations in which very few remote I/O systems are needed and where the likelihood of electrical transients is low.

2.2 RS-422/485

RS-485 allows multiple devices (up to 32) to communicate at half-duplex on a single pair of wires, plus a ground wire (more on that later), at distances up to 1200 meters (4000 feet). Both the length of the network and the number of nodes can easily be extended using a variety of repeater products on the market.

How does the hardware work? Data is transmitted differentially on two wires twisted together, referred to as a "twisted pair". The properties of differential signals provide high noise immunity and long distance capabilities. A 485 network can be configured two ways, "two-wire" or "four-wire" [3].

In a two-wire network the transmitter and receiver of each device are connected to a twisted pair. Four-wire networks have one master port with the transmitter connected to each of the slave receivers on one twisted pair. The slave transmitters are all connected to the master receiver on a second twisted pair. In either configuration, devices are addressable, allowing each node to be communicated to independently. Only one device can drive the line at a time, so drivers must be put into a high-impedance mode (tri-state) when they are not in

use. Some RS-485 hardware handles this automatically. In other cases, the 485 device software must use a control line to handle the driver. (If your 485 device is controlled through an RS-232 serial port, this is typically done with the RTS handshake line.)

Two-wire or four-wire? Two-wire 485 networks have the advantage of lower wiring costs and the ability for nodes to talk amongst themselves. On the downside, two-wire mode is limited to half-duplex and requires attention to turn-around delay. Four-wire networks allow full-duplex operation, but are limited to master-slave situations (i.e. a master node requests information from individual slave nodes). Slave nodes cannot communicate with each other. Remember when ordering your cable, two-wire is really two wires + ground, and four-wire is really four wires + ground.

Connecting a multidrop 485 network. The EIA RS-485 Specification labels the data wires A and B, but many manufacturers label their wires + and -. In our experience, the - wire should be connected to the A line, and the + wire to the B line. Reversing the polarity will not damage a 485 device, but it will not communicate. This said, the rest is easy: always connect A to A and B to B.

Signal ground, don't forget it. While a differential signal does not require a signal ground to communicate, the ground wire serves an important purpose. Over a distance of hundreds or thousands of feet there can be very significant differences in the voltage level of ground. RS-485 networks can typically maintain correct data with a difference of -7 to +12 Volts. If the grounds differ more than that amount, data will be lost and often the port itself will be damaged. The function of the signal ground wire is to tie the signal ground of each of the nodes to one common ground. However, if the differences in signal grounds are too great, further attention is necessary. Optical isolation is the cure for this problem.

Many RS-422/485 protocols were developed in the 1980s for such applications. Profibus, Interbus and CAN were a few. These were designed with proprietary interests in mind, and each subsequently received backing from several vendors in an attempt to develop an open international standard with widespread support.

However, competing interests have prevented any of these buses to emerge as a clear leader. Since interoperability is such an important aspect to consider when designing a system, it would be wise to consider a truly open protocol that can deliver this capability.

2.3 Ethernet

Ethernet has been around for more than 10 years and has become a commodity in modern business environments. Ethernet also has many characteristics that make it suitable for industrial networked and remote-sensor

I/O applications. It combines the low cost of RS-232 with the multidrop capability of RS-422/485 and provides a clear standard for communications [4].

The same economic forces that drove the PC into the industrial marketplace are now working to drive Ethernet there. While these networks have traditionally been used in office environments, industrial users cannot afford to ignore the cost and performance benefits of Ethernet in their factories. Ethernet has become the medium of choice to communicate management data throughout the enterprise.

One of the best parts of using Web technology software, there is no user software to distribute, no new software for a different connected platform, and no tracking of users. The HTML graphics and data displayed on the remote browsers are stored in the server [5].

Internet appliance and sent to the user via the network, on demand. The user provides the remote browser and computer. The HTML graphics can be displayed on any commercial browser. The embedded Web server will provide some security (usually logon/password) and the corporation's firewall will keep outsiders away. Probably all of your customers are familiar with the use of a Web browser so training of end users is limited to educating them about the tester.

Ethernet is an open standard worldwide. Originally developed by Xerox, it was standardized in the mid-1980s by IEEE. Since then, it has proliferated so that most new buildings are prewired for Ethernet.

TCP/IP is a suite of protocols that has virtually universal support. It comes as part of all Windows 95 and NT operating systems, it is used almost universally in UNIX environments, and it is the protocol used on the Internet. By using TCP/IP, you can be confident that you will have little or no trouble with interoperability [6].

2.4 Instruments with web interface

In the past two years, a new type of instruments was developed. This kind of instruments has no front panel, no buttons, sliders and so on. They have only the connectors for wiring signals and one Ethernet connector.

The front panel is on the pc. And the great thing is the pc could be any pc from the network!. An example of this kind is WebDaq100, produced by Capital Equipment Corporation [7]. Is a complete data acquisition system with built-in web-server, graphing, email, FTP and alarms.

No front panel mean less money for the production line and smaller time for the assembly line.

The emergence of web interfaces as an ubiquitous standard, network interfaces built into many computers, and available technology for embedding full

web servers into devices make it possible to create easy-to-use devices. Building a flexible user interface into the box eliminates drivers, software, and hardware installation hassles while allowing both local use as a computer peripheral and remote access [8]. Both general-purpose data acquisition and conventional test and measurement instruments can benefit from becoming web instruments.

3. How to transform our old instrument based RS232 or RS485 into a web based instrument

One of the most elegant and partially free method to do that is to build two software applications, one for the interface with our instrument via Test Point, Lab View or any other software designed for industrial automations and the second application for transmitting the acquired data to web. The second application will be developed using the script language PHP.

PHP and Apache server are the keys for the second application. These are the most used softwares for online applications.

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for modern operating systems including UNIX and Windows NT . The goal of this project is to provide a secure, efficient and extensible server that provides HTTP services in sync with the current HTTP standards. Apache has been the most popular web server on the Internet since April of 1996. The February 2005 Netcraft Web Server Survey found that more than 68% of the web sites on the Internet are using Apache, thus making it more widely used than all other web servers combined. The Apache HTTP Server is a project of the Apache Software Foundation [9].

PHP is a widely used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML [10].

3.1 How to put data on the web

Putting data on the web can be done in two ways (Fig.1). The first method send acquired data to a database server. The simplest approach defines an HTTP object method in the database that allows each table in the database to interpret and respond to HTTP messages. The database can then act directly upon HTTP messages and provide real time responses to web clients with dynamically created pages. Online database applications could include dedicated HTTP-based web servers, WAP servers, NNTP news servers, remote boot servers, and even SMTP mail generation and handling.

The second method use SOCKS object for establishing a direct connection between web clients and 1st application. [11].

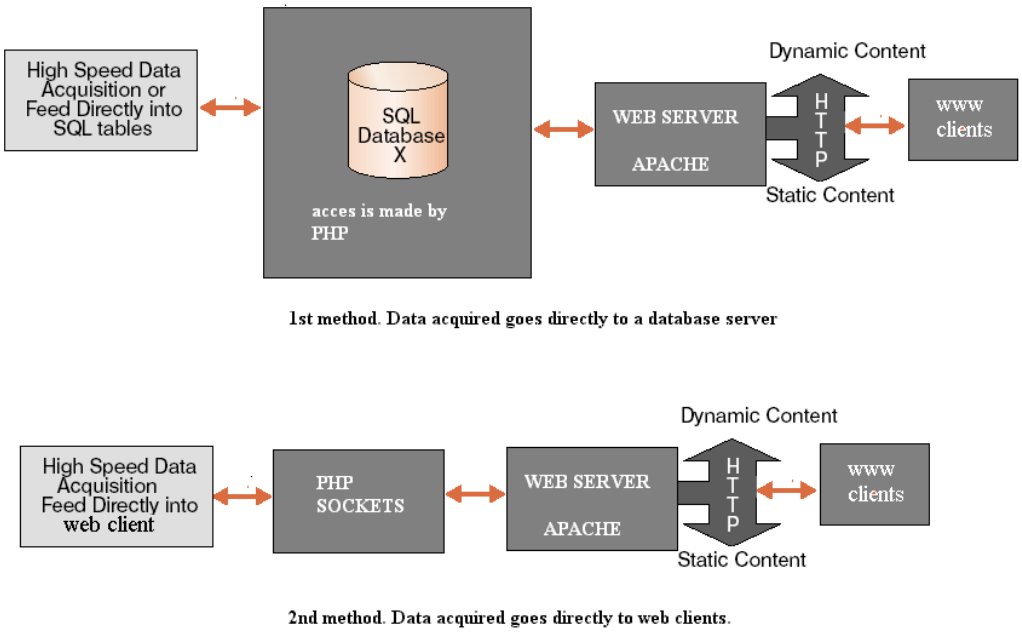

Fig. 1 Two methods for transmitting web data

A socket is a software endpoint that establishes bidirectional communication between a server program and one or more client programs. The socket associates the server program with a specific hardware port on the machine where it runs so any client program anywhere in the network with a socket associated with that same port could communicate with the server program (Fig 2).

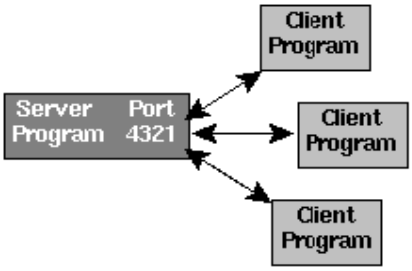

Fig 2. Socket concept

A server program typically provides resources to a network of client programs. Client programs send requests to the server program, and the server program responds to the request.

One way to handle requests from more than one client is to make the server program multi-threaded. A multi-threaded server creates a thread for each communication it accepts from a client. A thread is a sequence of instructions that run independently of the program and of any other threads. Using threads, a multi-threaded server program can accept a connection from a client, start a thread for that communication, and continue listening for requests from other clients.

.

4. The experiment

In laboratory, using very reliable position transducers, and heavy-duty hydraulic cylinders, two electro hydraulic servomechanisms were designed and tested in order to be used for the control of the wicket gates and turbine runner openings and for transmitting data over the Internet. All the computations were performed in order to add a web interface for the hydro mechanical speed and power governor of the hydropower unit from the RAMNICU VALCEA Hydropower Station, located on the Olt river (2 units of 23.5 MW) [12].

For Internet communication, APACHE SERVER with PHP 4.0 support was installed. The main instruments who was updated with web interface was an ADWIN PRO from Keithley Inc, USA

The main task of this industrial instrument is the generation of the reference voltage for the guide vanes servomechanism, and the reference voltage for the runner blades servomechanism, in order to obtain maximum overall efficiency. This instrument basically contains a powerful SHARK DSP, running at 40 MHz. The control software was written in ADBASIC, which is a simple and direct programming environment. As an interface computer for programming and supervising the control process, another industrial computer was chosen: a Pentium III heavy-duty panel computer running under WIN 98 SE. As data acquisition and dialog software, TEST POINT 4 from CAPITAL EQUIPMENT CORPORATION (S.U.A.) was chosen.

The interface computer stores all the process information with the same sample rate on a static HDD.

For transmitting data on web, we have chosen the second method, which allow us a real time control.

All the informations are available on the local net page built under PHP4.0, at 4 seconds refresh for page [13].

On the main page we have informations regarding the speed governor, oil pressure and few other parameters (Fig. 3).

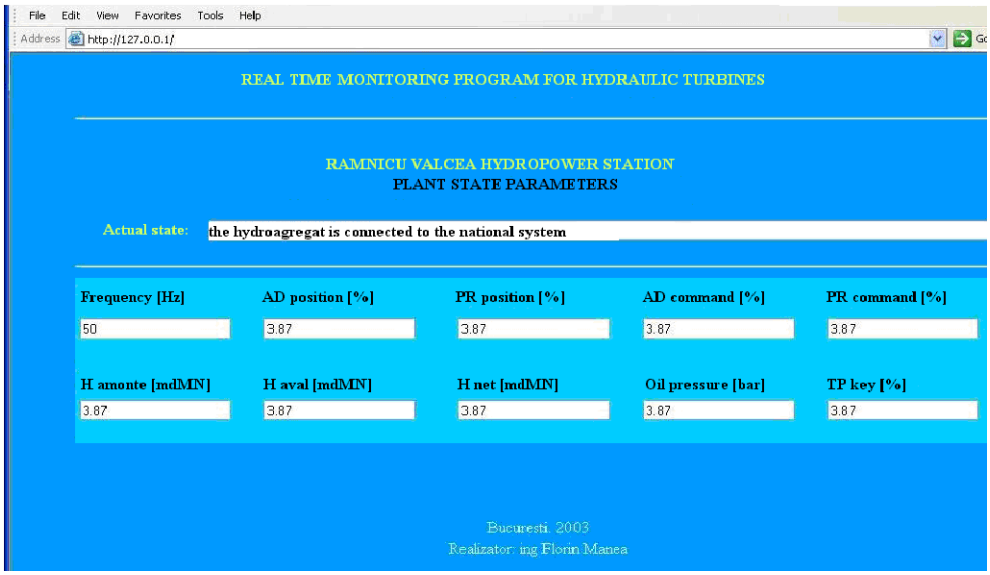

Fig. 3 The web panel interface

5. Conclusions

With a small effort, we demonstrated how can be updated an instrument with web interface even if it has no Ethernet connector. Only a PC with network interface is required. In this manner, we can update all kind of old instrumentation. If previously, only small teams of scientists or engineers had access to the data, now our web instruments provides everyone potential access to the data. And this is a good thing.

R E F E R E N C E S

[1] *Syd Mayer,* Creating web based data acquisition appliances, Evaluation Engineering Magazine. www.evaluationengineering.com , 2005.
[2] *Qi Wang*. How to make distributes measurements using Ethernet. ww.evaluationengineering.com, 2003.
[3] *BB Electronic,* The RS232 Standard, www.bb.com , 2005.
[4] *BB Electronic,* The RS485 Standard, www.bb.com , 2005.
[5] *Vance VanDoren, Mario Bernadini,* Ethernet Explained, B&B electronics, www.bb.com. 2005.
[6] *Robert Winkler*, Data acquisition embraces Ethernet, Evaluation Engineering Magazine, www.evaluationengineering.com, 2005.
[7] *Jeanine Katzel, Ethernet in automation*, www.manufacturing.net, 2005.
[8] *CEC, Network instrumentss,* www.cec488.com, 2005.
[9] *ApacheTeam,* The Apache Server, www.apache.org, 2005
[10] *PhpTeam,* PHP HyperTextPreprocessor, www.php.net, 2005
[11] Socket Communications, http://java.sun.com. 2005
[12] *Nicolae Vasiliu, C. Calinoiu, Florin Manea.*, A new type of electro hydraulic digital speed governor for hydraulic turbines, International Conference for on computational methods in fluid power technology, Melbourne, 2003.
[13] *Florin Manea,*. Masurari electrice distribuite sub mediul Internet, Electronica Aplicata nr.15, 2004.