

A Simple PC based Hardware for Process Control through the Internet

Dr. Mohamed Abdullah Hussain, Dr. Mumtaz Mohamed Ali, Dr. Wrya Mohamed Ali

Computer Science Institue, Suleimani-Iraq, blndo2001@yahoo.com

Information Engineering Faculty, AL-Nahrain University, mumtaz_almukhtar@yahoo.com

Communications S.A. in 92158 Suresnes Franceand, wryamam@yahoo.fr

Abstract

Internet based process had widespread in the industry field in the recent years. Little documentation can be found on this due to company polices to protect their projects. In our region the case is different and it is rare to find internet based process control. Simplicity in any system is an asset to have. Especially, in our region in which we lack hardware and software company dealers. It is best to depend on free software as far as is possible in addition to on simple easy repairable systems.

In this research a hardware interface circuit and a software system has been designed to control the temperature and level of a liquid tank (water has been considered in our case study). The main advantage of the designed interface circuit is its simplicity and low cost. The same can be true for the software system in which we used Java servlet to accomplish the communication task. Java servlets have many advantages over the CGI based methods used previously that has common problems of speed and data persistence.

In security aspects a password has been added to all control action commands to secure our process control tasks. The designed hardware and software systems could be used for remote or local control with no amendment and for both manual and automatic control.

Keywords: Remote Instruments Control, Network based Control, Network based Instruments Control, Remote Process Control, Internet based Process Control

1. INTRODUCTION

In this work the field of remote control has been explored. The characteristic of our work environment have been considered from the start which are:

- Lack of hardware and software company dealers, which results in their non ability to provide any after sale services.
- The big amount of routines and convincement efforts for purchasing any new expensive system.

- Non availability of suitable environment for installing dedicated complex systems in most cases.

Hence, simplicity of the designed system as whole was one of the goals in my mind. In addition to the use free software and the less expensive hardware as far as possible.

2. INTRODUCTION TO PC BASED PROCESS CONTROL

Instrumentation and process control can be traced back many millennia. Some of the early examples are the process of making fire and instruments using the sun and stars, such as Stonehenge [1]. The evolution of instrumentation and process control has undergone several industrial revolutions leading to the complexities of modern day microprocessor-controlled processing. Today's technological evolution has made it possible to measure parameters deemed impossible only a few years ago. Improvements in accuracy, tighter control, and waste reduction have also been achieved.

With the increased role of PC based control and the widespread use of networking systems and the Internet, the idea of interchanging control data between the computers emerged. In many process control systems we can have several PCs controlling several process variables and data could be exchanged between these computers and a supervisor computer or a server.

One of the most common used applications for remote control systems involves client-server architectures. The general structure of a client-serve based remotely controlled laboratory is shown in figure 1.

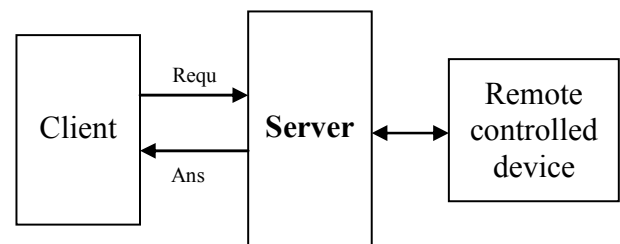


Figure 1: General structure of a remotely controlled device using client-server architecture

Information exchange between process plants and Internet-based clients allows the clients to remotely monitor and adjust the behavior of the process.

3. LITERATURE REVIEW

Most of the work in the area of remote control lies in the field of laboratories and instrumentation control, below is a survey on what has been done in this field:

In [2] a system was developed in which students could perform experiments by remote control via the Internet. At the client site a version of the Netscape browser is required that can accept JavaScript. At the server there is an HTTP server. CGI is used to extract information from the client sent data.

The server controls the experiment itself over a local network, such as over the parallel measurement bus IEEE 488 and RS232 to the laboratories machines and instruments. Using the graphical programming language G and the Internet toolkit from LabVIEW, the engineer is able to design particular measurement and control program. The main experiment is to get ultrasonic measurement from industrial ultrasonic equipment USD15D and two cameras.

In [3] systematic design methods for the design of remote monitoring and control over the Internet is developed. Five essential design issues that arise from Web-related features of Internet-based remote control have been investigated: communication latency, network communication architecture, optimal control structure, web-based user interface, and security aspects.

In [4] an illustration for remote control of hardware in the laboratories for educational purposes is made. The goal is to eliminate extensive programming using high level languages, such as Java to achieve remote connectivity, and focus in the primary task of implementing control algorithms. The use of MS NetMeeting for Internet-based control is described. Control experiments were carried out by a remote user using Matlab-based real-time tools and MS NetMeeting. The plant has its own electronics (amplifier, filter, etc.), provided in the I/O Electronics box. The controllers are designed using Matlab R12 with Simulink.

The article in [5] describes a new way of teaching adopted at an educational institution that uses dynamic and interactive simulations in a stand-alone or web-based environment. The article focuses on how this new stand-alone experimentation environment maintains a clear separation between the graphical experimentation interface, developed in Java, and the math and simulation engine. By constructing the environment in this fashion, the engine can be replaced with a different one or with a real plant, or can even be ported to a remote server.

In [6] a virtual Lab for controlling real experiments is presented. The aim is to provide students access to various experiments in control engineering. The Virtual Lab is based on a distance education concept due to the fact that certain students may be interested in studying even at places which are far away from campus eliminating the necessity to be there in person. In the virtual Lab they are able to gain some practice in control

theory at their convenience thereby saving travel time and cost. An approach based on a client/server architecture written in Java is proposed. CGI is used to extract parameters from commands send by the client to the server. One of the experiments is to design a speed controller for an educational vehicle with wheels. On the client side a student can use the Virtual Lab with a web browser and Java run time environment.

[7] is an article that describes the different methods could be used to achieve remote control. There is a mentioning of Web browsers to achieve instruments control. The use of HTTP protocol has been mentioned to exchange the information between the client and server, but on the server side they were using common gate interface (CGI) to extract information from the HTTP. Although remotely controlling applications and distributing control via the Web has countless benefits related to operator convenience, as well as company time and cost saving, operators should also be cognizant of possible drawbacks which is basically high amounts of traffic on the network that could lead to slow update or data transfers.

The common CGI drawbacks ranging from speed problem of to data persistence and development is shown in [8].

In [9] the architecture of a synchronous remote accessing control laboratories is presented. Various communication protocols were tested to access the virtual laboratory on the Web. However, the time delay decreased interaction between the process and the user. The paper presents real manipulation of real devices used in an instrumentation and control laboratories on the Internet. Control variable could be level, flow or pressure. A programmable logic controller (PLC) communicates directly with the sensors to receive signals and to activate the actuators. Communication between the PLC and the computer is assured by LabVIEW (a National Instruments graphical software). The user interface is also designed using LabVIEW

Ref [10] shows that Internet World Wide Web provides a best way to materialize remote access, reducing the requirements in the educational laboratories to personal computers (PC). In the work most of the measurement instruments used in test and characterization processes can be remotely programmed from an external controller by using a communication interface IEEE-488. In the work GBIP (General Purpose Interface Bus) has been used with HP 9145B semiconductor parameter analyzer implemented in their laboratories. The virtual instrument (HP 9145B) could be managed by the user from any web browser.

In [11] with the aid of the LabView and VEE PRO software, http-based client-server systems and gateway systems for the remote control of experiments were developed, built and tested. One example involves a complex experiment using ultrasonic, eddy-current and optical sensors. If Java is used as a software platform, all access to the hardware components must be made using programming methods. The online visualization

of remote-controlled experiments and the interactive control possibilities is one of the most important challenges that have to be solved.

In [12] a double client-server architecture for an online experiment is presented. The first client-server is between the client browser and the web server associated with the online lab management system. The user connect to the online laboratory using a web browser. The second client-server is between the online lab management system and the scattered resources that are wrapped as Web Services.

In [13] the objective, strategy and implementation details of a new undergraduate course, Internet-based Instrumentation and Control, are presented. It is offered to senior-level undergraduate engineering students who are interested in sensing, instrumentation, control and web. Visual basic is used to write software components for specific tasks. In the experiment the student will get familiar with programming environment. They will be able to create a new project in VB.NET and write simple Visual Basic program. The basic structure used in this work is to use a web browser by the client. The server is an ASP.Net based communicate with the client. The ASP.Net server communicate through a serial link with a Basic Stamp 1 controller.

3.1 CONCLUDING REMARKS FROM THE REVIEW

In most of the works mentioned in the literature we can see the use of RS-232, RS 485 and IEEE-488 and HP-GBIB hardware [10],[12],[14] for controlling Labs or instruments. We did not come across any use of regular computer parallel printer port (PPP) for Internet based remote control. Using the PPP will reduce the hardware cost and increase portability. Some documents mentions that there is some timing problem related to the PPP, as written in [14], but for our case the process that has been selected is a slow one and this timing problem and if it may emerge with a high speed process it will not with our selected process. In the work an actual real process has been created and controlled, a slow one has been selected to minimize the chance of instability. Also, we did not come across of using Java servlet to extract HTTP request parameters. The use of CGI has been mentioned.

4. THE CURRENT PROJECT

In this work the following systems and programs have been designed:

1. A water tank process.
2. A hardware interface circuit.
3. The interface hardware communication program.
4. A program to enable process control over networks and Internet.

4.1 THE WATER TANK PROCESS

When we have started the project we were seeking for an appropriate instrument or process to start our remote control operations with it. Due to unavailability of an appropriate instrument we have designed a water tanking system and as shown in figure 2. Our major goal has been achieved with it, which are monitoring (by sensing several parameters) and control (through generating several control signals).

It is basically consisting from two tanks the main one and a sub tank. The main tank has a supply and drain for water flow in and out, respectively. There is a heater inside it, a temperature sensor and two level sensors. The sub tank which works as a reservoir has also a supply and drain lines. The sub tank is much bigger than the main tank to hold all the fluid of the main tank without overflowing.

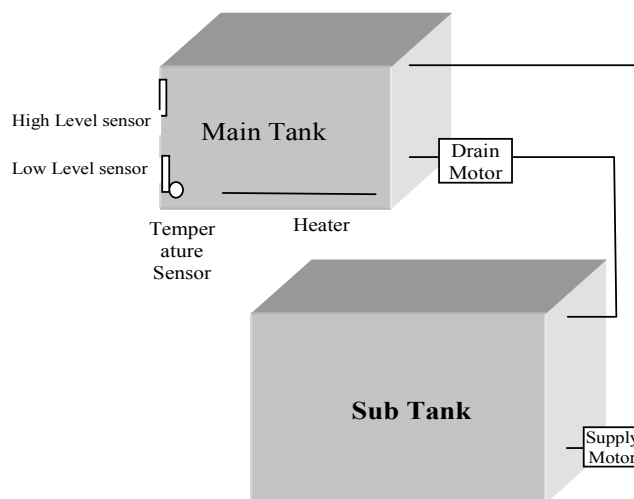


Figure 2: The Water Tank Process

In the work we have used water but, any other fluid could be used and in that case the level sensors have to be changed. Appropriate sensors capable of sensing the fluid level should be used. In this case impedance based level sensors have been used, when there is water between the sensors points the impedance will be low and this will generate a logic one signal. When there is no water the impedance will be high and this will generate logic zero signal.

4.2 THE HARDWARE INTERFACE CIRCUIT

The hardware interface circuit has been designed to work with the parallel printer port (PPP). The parallel port consists of a connector with 17 signal lines and 8 ground lines. The signal lines are divided into three groups, Control (4 lines), Status (5 lines), and Data (8 lines).

The Control lines are used as interface control and handshaking signals from the PC to the printer. The Status lines are used for handshake signals and as status

indicators for paper empty, busy indication and interface or peripheral errors. The data lines are used to provide data from the PC to the printer, in that direction only. Later implementations of the parallel port allowed for data to be driven from the peripheral to the PC.

Three output pins D1 to D3 are used to drive the heater, supply and drain motors. Ack, Busy, Paper Empty, Select and Error of PPP have been used to input parameters to the computer. Busy, Paper Empty, and Select are used to read the tank temperature through a simultaneous type A/D converter. The other two pins are used to read the level of the water tank. The hardware block diagram is shown in figure 3.

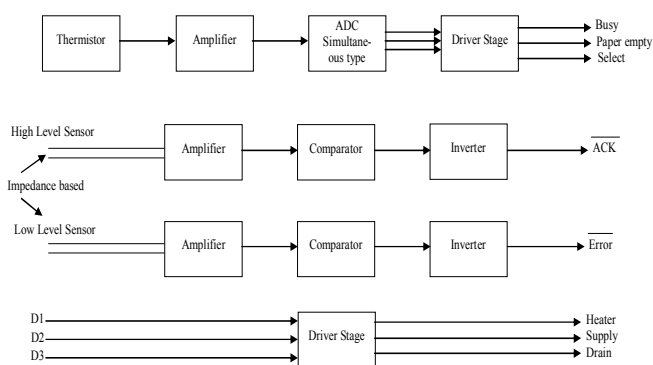


Figure 3: A block diagram of the hardware circuit used to interface the tank to the PC

The hardware has been designed mainly using comparators and TTL ICs. Using the PPP gave the chance to engage the hardware with any available PC or Laptop (if it is equipped with PPP) with no need to install any special hardware inside the PC (or the Laptop). At the client side, sensors and control parameters could be read and given respectively to the process (plant under control), through and ordinary web page designed for that purpose.

4.3 THE INTERFACING HARDWARE SOFTWARE

A C based interfacing program have been designed to work with the PPP. The program consists of codes that read the PPP status port and write data to the PPP data port. In the reading process a filtration process have been introduced to tackle some unstable states we did encounter in the testing phase. The filtered status port data are stored in a binary file with name INPUT.

In writing to the output data port the content of a binary output file with name OUTPUT is sent to the port.

A sample of the C program debug screen window is shown in figure 4 and the program flowchart is shown in figure 5.



Figure 4: In Port Reading Debug Window

As it is shown in the flowchart the program first get the input port status and save it in variable In, but it will not save it to INPUT file. It will do more input port readings before deciding to save the input in the INPUT file. Two more reading are required and if the other two reading matches the first one it will save the input to the INPUT file.

By this way we will eliminate the chance of storing transient non stable input in the INPUT file. It is true that this filtration will increase the delay of getting the input port status but it will enhance accuracy of the reading. Just like any other hardware filtration it will do an averaging for the input signal.

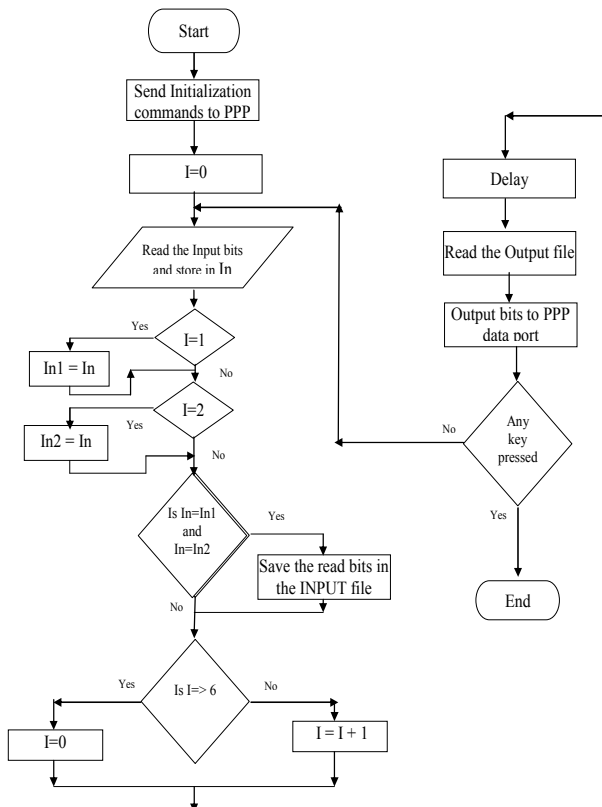


Figure 5: Flowchart of the Hardware Interfacing Program with Filtration

If the next two consecutive reading matches the first one the program will assume that those readings are stable and use them for four more times in the comparison process. Of course the amount of time to depend on the first three readings can be adjusted by varying the conditional statement “Is $I \geq 6$ ”. Choosing lower values will decrease this dependability. This value of 6 has been chosen based on practical observation of the input port readings

4.4 SERVLETS BASED PROCESS CONTROL PROGRAMS

Java servlets have been used to design the communication programs. Servlets were designed to solve the problems of CGI and create robust server-side environments for Web developers. Similar to CGI, Servlets allow a request to be processed by a program and let the same program produce a dynamic response. Servlets additionally defined an efficient life cycle that included the possibility of using a single process for managing all requests. This eliminated the multi-process overhead of CGI and allowed for the main process to share resources between multiple servlets and multiple requests.

In the work two kinds of servlets programs have been designed. One is for Manual control and the second is for Automatic control. Both kind of servlets are activated by a submit form from the client side. Two main forms have been designed one for Manual control mode (figure 6) and the other for Automatic control mode (figures 7).



Figure 7: Automatic Mode Submit Form

When the main HTML page is appearing (which is a simple selection HTML page), the client have two options, one for manual mode and the other for automatic mode by which he is directed to one of the forms shown in figures 6 and 7..

In the Manual control mode the Servlets program will send the control parameters to the OUTPUT binary files and reads parameters from the INPUT file. Then it will encapsulate the current temperature and level as an HTTP response and send it back to the client. The client can send control commands at any time and he will get status back as soon as the servlet program reads the parameter from the INPUT file. It is required that user is aware of the password that the system administrator at the server side should give. Otherwise he can not do any control action.

The Automatic mode program pseudo code is shown in figure 8.



Figure 6: Manual Control Mode Form

```

Temperature Control, Automatic Mode
{
Extract parameters from the HTTP Post request;
Extract the password from the HTTP Post request;
Retrieve the password from pass.txt file;
If the retrieved password equals the user given password
{
Store the requested temperature in variable setTemp0;
Transform the set point temperature to the nearest in band one and store it in
setTemp;

Initialize variable count to one:
Initialize variable lcount to one:
Do While (Count is less than 10000 or lcount<>3)
{
Send Temperature and Level commands to the
Output Binary File (OUTPUT);
Get temperature and level from the
Input file (INPUT) and store in sensTemp1;
IF (setTemp-sensTemp)>5 Then {
If lcount<>1 then send Switch on the Heater to
OUTPUT;
Send response to the client to signal the current
temperature and level;
lcount=1;
}
IF (setTemp-sensTemp)<5 Then {
If lcount<>2 then send Switch off the Heater to
OUTPUT;
Send response to the client to signal the current
temperature and level;
lcount=2;
}
IF absolute of (setTemp-sensTemp)<=5)
Then {
If lcount<>2 then send Switch off the Heater to
OUTPUT;
Send response to the client to signal the current
temperature and level (set point is reached);
lcount=3;
}
Increase count, lcount;
} Else {
Notify the user that he is authorized to access the server;
}
}
}

```

Figure 8 : Automatic Mode Servlet Pseudo Code

In the Automatic mode the client writes the desired temperature and the access password. Then the Servlets automatic mode program will go in a loop polling the hardware till the hardware reaches the desired temperature. During this process the client will get continuously the level and the temperature of the tank.

5. RESULTS

The system operation has been tested on Internet in both manual and automatic mode. Response time for actuators (heater, supply and drain motors) switching on or off were always less than two second. This response time characteristic is very acceptable for our selected process. The designed tank process needs about 40 minutes for rising the temperature from 0 to about

100°C (based on practical calibration). Hence having this delay is negligible for the current process.

For the Internet system, two ISPs were used; one for the server and the next is for the client. The reason for using two providers is to make an actual real test for process control and monitoring. Two providers usually mean that the client and the server are in different countries, cities or sites.

6. FEATURES AND APPLICATION FIELDS

The designed system could be applied to mobile system if they have Internet or network access. A client that is sitting in a mobile vehicle equipped with network access can control the temperature or level of any of the above mentioned processes. Also, a mobile process like a water tank or a patrol tank can be controlled or monitored from fixed or a mobile client.

The designed hardware is simple and could be plugged to any PC or Laptop that have a PPP port. This eliminates opening the PC for installing a card, driver installation and compatibility problems, and also allows easy system troubleshooting in case of PC or interfacing hardware malfunction. A market survey made on 19/4/2007 for checking the availability of PPP on modern PC's motherboards (MB) gave us that it is still embedded in modern MBs. Two types have been checked, Gigabyte Intel 865 chipset and ASRock P4I65G, both types were equipped with PPP.

The C language hardware interfacing software and Java based communication programs can be obtained for free from Borland and Sun Microsystems. This will reduce the total system cost.

There is no need to have a dedicated client PC, any available PC could be used by the client. The same could be said for the server side PC. In the work we have used a Celeron based 2.1Ghz computer to control the tank in addition to running to our regularly needed programs on that computer, e.g. MS-Office, MSN messenger, Yahoo messenger and Skype. There was no need for a dedicated server.

In addition to the basic use of the system for remote process control and monitoring, it could be used for local or direct control. No changes are required; it is effective for direct control as it is for remote control.

The system could be used at Electrical, Computer, Control and Automation engineering department as an educational system for:

- Use of computers to control devices and instruments (direct).
- Remote control of devices and instruments.
- Write programs to do different control actions.
- Understand the role of process characteristics on the design of control systems.

The processes that this work could be applied to (close in characteristic to our worked on process):

- City water supply tanks.
- Boilers of buildings climate control systems.
- Patrol station's tank level control, after changing the sensors to a proper type.
- Water Dam level control.
- Automated houses water tanking systems.

7. CONCLUSION

In this work a novel approach was used to implement Internet based remote process control. The characteristics of the designed system are:

- Effective, have low cost and simple.
- Could be used for direct (local), network, and Internet based process control.
- Could be used for teaching purposes.
- Modular, the components engaged together could be replaced without affecting other components. For example, in some modern laptops we can not see the PPP port. In those cases a hardware circuit that could use the USB or a Data Acquisition Card could be used without affecting other parts operation.

REFERENCES

- [1] William C. Dunn, *Fundamental of Industrial Instrumentation and process Control*, McGraw-Hill, 2005.
- [2] Page,G.F.,Ewald, H., "Performing Experiments by Remote Control using the Internet", *Global J. of Engineering Education*, Vol 4, No.3, pp.286-292, 2000.
- [3] Hong-Yan Li, "Web-based remote monitoring and control for process plants", in *Proceeding of the Fourth Internatioal Conference on Machine Learning and Cybernetics*,18-21 August, pp. 936-941, 2005.
- [4] Nitin Swany, Ogenjen Kuljaca and Frank L.Lewis, "Internet-Based Educational Control systems Lab Using NetMeeting", *IEEE Transaction on Education*, Vol 45, No.2, pp. 145-151, May 2002.
- [5] J.Sanchez, F.Morilla and S.Dormido, J.Aranda, and P.Ruiperez, "Virtual and Remote Control Labs using Java:A Qualitative Approach", *IEEE Control systems Magazine*, pp.8-20, April 2002.
- [6] Christof Rohrg and Andreas Jochheim, "The Virtual Lab for Controlling Real Experiments via Internet", *Proceedings of the 1999 IEEE, USA*, August 22-27, pp.279-284, 1999.
- [7] Kris Fuller, "Control at a distance", *Spie's oemagazine*, pp.20-23, October 2002.
- [8] Larne Pekowsky, *Java Server Pages*, Addison-Wesley, 2004.
- [9] Maarouf Saad, Hamadou Saliah-Hassane, "A Synchronous Remote Accessing Control Laboratory On the Internet", *ICEE*, August 6-10, pp.8DI-30-8DI33, Oslo 2001.
- [10] R.Cabello, M.Dieguez, and I.Gonzalez, "Remote Control of Electronic Equipment An Application to the Practical formation of University Students", *Ist Technical Workshop of the Computer Engineering Departament- Universidad Aut□noma of Madrid, Cantoblanco-Campus (Madrid, Spain)*, March 2000.
- [11] Page,G.F.,Ewald, H., "Client-Server and Gateway Systems for Remote Control in Engineering Education", *Global J. of Engineering Education*, Vol.7, No.2, pp.201-208, 2003.
- [12] Yuhong Yan, Yong Liang and Xinge Du, "Controlling Remote Instruments Using Web Services for Online Experiment Systems", in *Proceedings of the IEEE International Conference on Web Services (ICWS'05)*, 0-7695-2409-5/2005.
- [13] Hanqi Zhuang, and Sal Morgera, "An Undergraduate Course-Internet-based Instrumentation and Control", *34th ASEE/IEEE Frontiers in Education Conference*, pp.TIH-12-TIH-17, October 20-23, Savannah, 2004.
- [14] Zafir Hassan and Ala Hasan, *PC Interfacing Black Book*, Spot for Education and Science, 2003.
- [15] Gregory K. McMillan and Douglas M. Considine, *Process/Industrial Instruments and Controls handbook*, McGRAW-HILL, 1999.
- [16] Elliotte Rusty Horold, *Java Netwrok Programming*, O'Reilly & Association, Inc., 2000.
- [17] JOHN R.HUBBARD, Ph.D., *Theory and Problems of Programming with Java*, Schaum's Outline Series, McGraw- Hill, 2004.