REMOTE CONTROL USING FUZZY LOGIC

EMRE GEYLANİ

OCTOBER 2005

REMOTE CONTROL USING FUZZY LOGIC

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

OF

CANKAYA UNIVERSITY

BY

EMRE GEYLANİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

IN

DEPARTMENT OF COMPUTER ENGINEERING

OCTOBER 2005

Approval of the Graduate School of Çankaya University

Prof. Dr. Yurdahan Güler
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Mehmet Tolun
Head of the Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Ali Sebetci
Supervisor

Examining Committee Members :

Assist. Prof. Dr. Murat Erten

Assist. Prof. Dr. Ali Sebetci

Assist. Prof. Dr. Reza Hassanpour

ABSTRACT

REMOTE CONTROL USING FUZZY LOGIC

Geylani, Emre

Master of Science, Deparment of Computer Engineering

Supervisor: Assist. Prof. Dr. Ali Sebetci

October 2005, 46 pages

Nowadays the Internet is playing a very important role in different domains. During the previous years a lot of research has been done for trying to develop applications, which make it possible to supervise and control industrial processes using the World Wide Web.

This paper presents a remote control technique with the advantages of the internet and platform independent technology. A web based control & monitor program has been developed at the PLC (Programmable Logic Controller) Server side to be able to control and monitor the real-time environments. Program is executable from all client computers in the network area only using java runtime installed web browsers.

The PLC in the study gathers analog data from the environment, processes them and logs the events in the registers. Client computers may log on to the

PLC and access data using the software.

Keywords: web based control, internet-based monitoring, process control, remote control

ÖZ

FUZZY MANTIK KULLANARAK UZAKTAN KONTROL

Geylani, Emre

Yüksek Lisans, Mühendislik - Mimarlık Fakültesi

Danışman: Yrd. Doç. Dr. Ali Sebetci

Ekim 2005, 46 sayfa

Günümüzde internet bir çok farklı alanda önemli rol oynamaktadır. Geçtiğimiz yıllarda bir çok araştırmacı internet ağını kullanarak kontrol yapabilen programlar üzerinde çalışmışlardır.

Bu çalışmada, internet ve platform bağımsız teknoloji avantajlarını kullanan bir uzaktan kontrol tekniği üzerinde çalışılmıştır. PLC (Programlanabilir Mantık Denetleyici) Sunucusu tarafında web tabanlı kontrol ve izleme yazılımı çevresel değişkenleri konrol ve izleme amacıyla geliştirilmiştir. Programı çalıştırmak için, network içindeki java sanal makinası kurulu bir bilgisayarda web tarayıcıyı kullanmak yeterlidir.

PLC analog bilgiyi sahadan alır, gerekli mantıksal işlemleri gerçekleştirir ve olayları hafıza alanlarında saklar. İstemci bilgisayarlar programı kullanarak

PLC'ye bağlanıp bu bilgilere ulaşabilirler.

Anahtar Kelimeler: web tabanlı kontrol, internet tabanlı veri izleme, proses control, uzaktan kontrol.

To My Father and Mother . . .

ACKNOWLEDGEMENTS

I express my thanks to my thesis supervisor Assist. Prof. Dr. Ali Sebetci, Assist. Prof. Dr. Reza Hassanpour, Prof. Dr. Turhan Alper and Prof. Dr. Mehmet Tolun for their support during this work.

I am grateful to the people at Meko Company, especially i express my eternal gratitude toward Rıza Aycan informing me about modern control techniques and letting me to use the company resources. I am also sincerely thankful to Adnan Öztürk, Kemal Ersoy and Yalçın Araşan at Meko Company for their precious support.

I would like to sincerely thank to Gökhan Kaptan for his guidance throughout the PLC programming stage.

I am also thankful to Mehmet Geylani for his inspirations and supports in this study.

Another thanks go to my precious friends Atilla Güneş, R.Rajesh and Manoj Rapaka who were supporting me to do my best in this thesis.

Finally, my greatest and eternal gratitude goes to my beloved family for their everlasting support and love.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLES

CHAPTER 1

INTRODUCTION

If an organization has equipment located over a large geographic area, it is difficult and expensive to manually monitor and control the status of the equipment and processes that are vital to safe and efficient operations, it is wise to implement a remote control system [1].

The World Wide Web has made it possible to send lots of data from one side of the world to the other side in almost no time. The use of the Internet for real time interaction of the remote controlling and monitoring of the plants would give us many advantages. This technology is not only be used in the industry, but also in the field of medicine, education and etc..

The Web Based Control and Data Acquisition (WBCADA) can be described as the whole of operations performed to control or monitor a system situated in a closed network. WBCADA is based on existing technologies, programming languages as Perl and Java and specifications as HTTP, TCP and UDP [2].

Operation of many industrial, facilities, natural resources, and utilities depend upon equipment and processes that are remote. Remote monitoring offers these organizations tools to quickly identify problems and remotely manage and control equipment and processes. Fortunately, low-cost, high-speed computer and communications technology enables any organization, regardless of size and budget, to automate and centralize the management of the operation.

Since most industrial equipment has monitoring instrumentation, it is straight-forward to install an electronic device such as PLCs that collects this information and forwards it over a wireless or wired communications network to a central location [1].

A person or computer can analyze the information, decide what action to take, and dispatch a person to the location. With added intelligence, the remote electronic device or central computer can automate the process by analyzing the information it collects, selecting the appropriate action, and correcting the problem by sending control signals that change the status of the equipment.

Monitoring control systems offer tangible, real savings in traditional hardwired and leased-line applications as well as new automation opportunities where hardwired systems are technically and/or economically impractical.

Modern control applications like SCADAs (Supervisory Control and Data Acquisition) support the transfer of data over conventional and cellular wireless, dial-up and leased-line telephone circuits. These systems are used to display the remote location schematics, alarms, charts(trends) and generating reports and etc.

The contact or communication between the central and the remote site can be done in many ways, depending on the availability of the infrastructure and the user preference, The following solutions are possible [3]:

- Public telephone switched network (PTSN)
- Leased line or dedicated wireline
- Conventional FM radio system
- Trunked radio system

- Microwave communication network
- Fiber optics direct link or network
- Satellite communication network
- Available Data communication LAN's

Various applications such as serving the gas and oil industry, electric and water utilities, communications, public security and etc. are general areas for these applications.

SCADAs can be integrated with web technology. Any client who is in the network can access the Web SCADA through Web server using the HTML browsers. A browser can view the SCADA information on any OS (Operating System) or machine platform. This is known as platform-independent technology. For security reasons, Web SCADAs generally have strong login policies.

CHAPTER 2

PREVIOUS STUDIES AND HISTORY

As a basis for the possible next generation of control systems, the concept of the Internet based process control has been introduced in recent years. To date, most research work on the Internet based process control has resulted in small-scale demonstrations like Sun Microsystems and Cyberonix, Foxboro, and Valmet. Most of them were developed in Java [4].

In 1995 an online controlled robot arm has been implemented by Dr. Ken Taylor. Users were logging on to a local computer to control the robot arm and see the positional status. This project was one of the first online control and data acquisition experiments [5].

Some companies try to produce Internet control systems as a control device. Some researchers in this area, from higher education institutions, focus on developing web-based virtual control laboratories for distance learning purposes. A remotely located user has possibility to conduct experiments in the laboratory via the Internet. The students can continuously access their hypothetical experiments setup. The major advantage of this virtual laboratory is the minimal cost needed to set up a laboratory, as it only requires a robust communication network [6].

The International Federation of Control (IFAC) has held the first workshop on Internet Based Control Education in Spain in 2002. The ScadaOnWeb system funded by the European Council targets Internet based protocols enabling the monitoring and optimization of the process via the web. It is hoped that the specific web based approach towards the development of an online framework will eventually result in the adoption of ScadaOnWeb as an industry standard for transporting large volume of process data online [7].

The master thesis titled as Remote Control of Heating, Ventilating and Air Conditioning System has been studied by Yu-Loong Liev in 2003 as a home automation system. In this study Liev has focused on controlling daily house variables using internet. Program has developed in a graphical programming language called LabView. Software controlling and gathering datas from devices were performed via X-10 protocol that uses only the power line as transmission medium. LabView has an add-on software which is capable of implementing a web server. Authorized users may have logged on to local computer to control the system [8].

Another academic study titled as Web Based Monitoring & Control of Industrial Processes has been authored by Egwin Warnier, Leena Yliniemi and Pasi Joensuu in 2003 at University of Oulu. This report was a brief overview about the design methods, architectures and the problems concerning security aspects of web based control systems. In the report the internet based process control has been examined and defined only as an extra control level added into the existing process control hierarchy. The topics such as user interface, universally platform independent programming languages, minimized communication loads, and security measures have been defined as important issues in the report [2].

CHAPTER 3

REMOTE CONTROL THEORY

In this chapter the backgrounds of the remote control and the PLCs have been discussed to clarify the terminolgy and to help the understanding of the topics in the following chapters.

3.1. BASICS OF REMOTE CONTROL

Remote Control is the ability to remotely monitor and control the network, collect information, and provide information in a useful manner to the end user. Also it can be defined as a software or device that monitors inputs, makes decisions based on its setup values, and controls outputs to automate a process or machine.

3.1.1. Remote Control Terminology

The following terms have been defined to help readers for a better understanding of the material covered in the thesis.

3.1.1.1. Accuracy

The term accuracy describes the total of all deviations between a measured value and the actual value. Accuracy is usually expressed as the sum of non-linearity, repeatability and hysteresis. Accuracy may be expressed as the percent of a full-scale range or output, or in engineering units [9].

3.1.1.2. Address

An address is a unique numeric or alphanumeric data (point) identifier [9].

3.1.1.3. Analog/Modulating/Continuous

These synonymous terms are used to describe data that has a value that is continuous between set limits represented by a range or span of voltage, current or resistance. The value is non-integer (real) with a resolution (number of significant digits) limited only by the measurement and analog-to-digital signal conversion technology. In typical systems, analog data from an input device is converted into a value for processing within the controller. Likewise, values are converted into analog output signals for use by a controlled device, such as an actuator [9].

3.1.1.4. Digital/Binary/Discrete

These synonymous terms are used to describe data that has a value representing one state or another. Typical values are on or off, alarm or normal, 0 or 1, high or low, etc. In the hardware side, these values most commonly relate to the state of a set of switch or relay contacts (open or closed) [9].

3.1.1.5. External Point

Data that is received by a controller from an external source, or sent by a controller to an external source, is an external point. The terms hardware, input or output may be used to describe an external point [9].

3.1.1.6. Global Point

Global points originate from a controller within a network that is broadcast via the network to other controllers [9].

3.1.1.7. Input

The term input is used to define data flow into a controller or control function [9].

3.1.1.8. Internal Point

An internal point is one that resides within a digital controller that does not directly originate from input or output points. Internal points can be constants such as fixed set points created by a programmer's or operator's assignment. Internal points may also be created as defined by the programmer/operator by applying logic and mathematics to other virtual, input or output points or combinations of points. The terms virtual, numeric or data may be used to describe an internal point [9].

3.1.1.9. Output

Output defines the data flow out of a controller or control function [9].

3.1.1.10. Process Medium

A process medium is a material in any phase (solid, liquid or gas) that is being used in a process. The most common types of process mediums used in commercial and industrial heating ventilating and air conditioning systems are liquid mediums (i.e., chilled water for cooling) or gaseous mediums (i.e., airflow in a duct) [9].

3.1.1.11. Sensor

A sensor is a device in primary contact with a process medium. It measures particular properties of the process medium (i.e., temperature, pressure, etc.) and relates those properties to electrical signals such as voltage, current, resistance or capacitance [9].

3.1.1.12. Transducer

Transducers accept an input of one character and produce an output of a different character. (Examples: voltage to current, voltage to pneumatic (pressure) and resistance to current) [9].

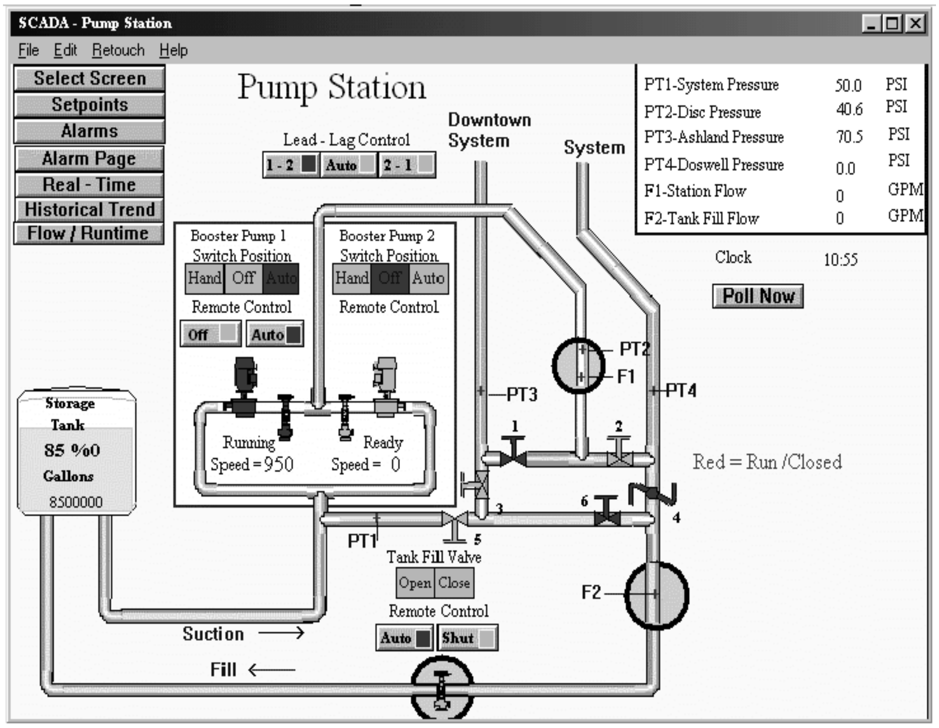
3.1.1.13. Transmitter

A transmitter is a transducer that is paired with a sensor to produce a higher-level signal (typically) than is available directly from the sensor. These sensors may be integral or remote and may include digital or analog signal processing. (Examples: temperature transmitter employing a temperature sensor. The temperature sensor varies the resistance with temperature change and the transmitter outputs a related 4-20 mA current output for use by a controller) [9].

3.1.2. SCADA

A SCADA system automates and centralizes the monitoring, control, and alarming of the remote system. Remote Terminal Units (RTUs), located at multiple sites, collect data from electronic devices, and forward the data to a central control center. The Control and Monitoring Program displays and analyzes data from reporting sites, issues alarms, reports on abnormal events, and initiates corrective actions as directed by an operator or automated instructions.

The ability of a SCADA system to not only monitor the status of equipment, but to control the remote equipment from a central location, allows more efficient and cost-effective management of remote sites. Routine adjustments to levels, power, pressure, temperature, etc. improve system performance while reducing maintenance costs. Response time to emergencies is significantly reduced and personnel productivity is therefore improved. SCADA is an operational tool that maximizes efficiencies and minimizes costs. An example to the SCADA applications has been given in Figure 3.1 [10].



PT1-System Pressure	50.0	PSI
PT2-Disc Pressure	40.6	PSI
PT3-Ashland Pressure	70.5	PSI
PT4-Doswell Pressure	0.0	PSI
F1-Station Flow	0	GPM
F2-Tank Fill Flow	0	GPM

Clock 10:55

Figure 3.1 – A View from a SCADA program

3.1.3. Alarm Processors

By interfacing with sensors that monitor conditions, actuators that can adjust processes and intelligent equipment such as PLCs. Alarm Processors send alert messages to key personnel upon detection of an alarm condition. Messages can be text pages or voice messages sent by radio or telephone.

For operations that require timely action, Alarm Processors are an ideal way to alert maintenance personnel [1].

3.1.4. Remote Terminal Units (RTUs)

RTUs located at multiple remote sites to collect information from sensors and status and alarm switches. The RTUs transfer the data to the System Controller. RTUs can also output commands to control the process.

Depending upon unit specifications common functions that may be performed by RTU’s include:

1. Continuous Monitoring & Alarm Notification
2. Data Logging & Control.

Continuous Monitoring is the most common use of RTU technology. This function provides personnel with the ability to contact, interrogate and interact with unattended equipment at all times without having to visit the site. Types of information that are continuously monitored include performance levels, output parameters, predictive and preventative maintenance schedules and other environmental conditions. When combined with Alarm Notification technology, RTU’s can detect and report pre-programmed alarm conditions such as system or equipment failures, out-of-range performance levels, maintenance alerts or security breaches. When an alarm condition occurs, the system automatically contacts predetermined destinations to notify personnel of the location and nature of the alarm. Most systems will continue contacting destinations until an authorized individual acknowledges the alarm condition via a remote device.

Data logging systems allow for remote data collection from a variety of process points. These systems sample designated input values at scheduled intervals and then send that data to a predetermined destination in formatted reports. Depending upon the model, units can be polled at any time and will report the data to a remote location [1].

3.1.5. System Controller

The System Controller, as a front-end processor device, manages the flow of data between the remote RTUs and the Control Program in the host management system [1].

3.1.6. Java as a Programming Language in WEB

Java is the first programming language designed from the ground up with networking in mind. As the global Internet continues to grow, Java is uniquely suited to build the next generation of network applications. It provides solutions to a number of problems which are difficult to address in other programming languages. The level of safety with Java applets is greater than obtained with the other software. Java makes writing networking programs easy. It is relatively straightforward for Java applications and applet to send and receive data and to communicate across the Internet, limited only by security restrains. Java is portable and platform-independent. In the past, software developers had to work with a specific native instruction set, which locked them into a specific hardware and operating environment. Unlike other programming languages, Java executes in a run time environment called a virtual machine. The Java virtual machine executes byte-code (platform independent code) that a Java compiler generates and it can be incorporated or embedded in Web browsers, or the kernel of the operating system. Only a Java-capable browser is required which will download the applets from the web server for running them on the client system. No specific software is required to be installed at the remote site [2].

3.2. PLCs

PLCs are the control hubs for a wide variety of automated systems and processes. They contain multiple inputs and outputs that use transistors and other circuitry to simulate switches and relays to control equipment. They are programmable via software interfaced via standard computer interfaces and proprietary languages and network options. A typical PLC has been shown in Figure 3.2

Programmable logic controllers I/O channel specifications include total number of points, number of inputs and outputs, ability to expand, and maximum number of channels. Number of points is the sum of the inputs and the outputs. PLCs may be specified by any possible combination of these values. Expandable units may be stacked or linked together to increase total control capacity. Maximum number of channels refers to the maximum total number of input and output channels in an expanded system. PLC system specifications to consider include scan time, number of instructions, data memory, and program memory. Scan time is the time required by the PLC to check the states of its inputs and outputs. Instructions are standard operations (such as math functions) available to PLC software. Data memory is the capacity for data storage. Program memory is the capacity for control software.

Available inputs for programmable logic controllers include DC, AC, analog, thermocouple, RTD, frequency or pulse, transistor, and interrupt inputs. Outputs for PLCs include DC, AC, relay, analog, frequency or pulse, transistor, and triac. Programming options for PLCs include front panel, hand held, and computer.

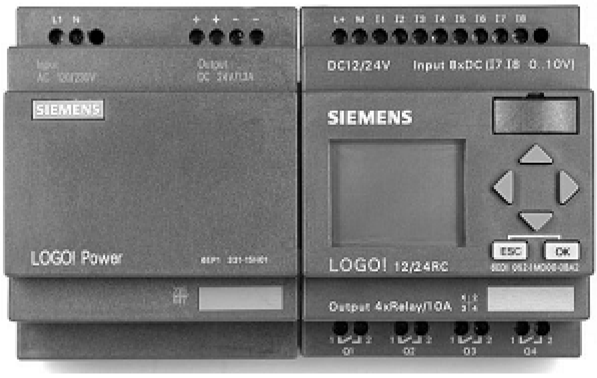


Figure 3.2 – SIEMENS Logo series PLC

Programmable logic controllers use a variety of software programming languages for control. These include IEC 61131-3, sequential function chart (SFC), function block diagram (FBD), ladder diagram (LD), structured text (ST), instruction list (IL), relay ladder logic (RLL), flow chart, C. The IEC 61131-3 programming environment provides support for five languages specified by the global standard: Sequential Function Chart, Function Block Diagram, Ladder Diagram, Structured Text, and Instruction List. This allows for multi-vendor compatibility and multi-language programming. SFC is a graphical language that provides coordination of program sequences, supporting alternative sequence selections and parallel sequences. FBD uses a broad function library to build complex procedures in a graphical format. Standard math and logic functions may be coordinated with customizable communication and interface functions. LD is a graphic language for discrete control and interlocking logic. It is completely compatible with FBD for discrete function control. ST is a text language used for complex mathematical procedures and calculations less well suited to graphical languages. IL is a low-level language similar to assembly code. It is used in relatively simple logic instructions. Relay Ladder Logic (RLL), or ladder diagrams, is the primary programming language for programmable logic controllers (PLCs). Ladder logic programming is a graphical representation of the program designed to look like relay logic. Flow Chart is a graphical language that describes sequential operations in a controller sequence or application. It is used to build modular, reusable function libraries. C is a high level programming language suited to handle the most complex computation, sequential, and data logging tasks. It is typically developed and debugged on a PC [11].

Programmable logic controllers can also be specified with a number of computer interface options, network specifications and features.

CHAPTER 4

THE WEB BASED CONTROL & DATA ACQUISITION SYSTEM

4.1 SOFTWARE

4.1.1. Introduction to Web Based Control & Data Acquisition

The rapid growth of the Internet provides tremendous opportunities for Internet based Automation. Household electronic devices such as lights, appliances, climate-control systems, and surveillance cameras are linked to the Internet through wire or wireless networks [12].

The most logical way for accessing controlling and monitoring systems in a plant via the Internet is to use a browser like Netscape or Internet Explorer. The user should be able to rapidly see what is happening in plants. It should be born in mind that media available in the Internet environment outside the central control room is very much limited compared to those in the central control room. There should also be made a difference between process operation functions and monitor operations functions. The first type needs flowcharts indicating current process status and historical trend displays while the other one asks for controller window displays.

In the control panel, the GUI displays the industrial process and makes it possible for the user to adjust the settings. Most important thing is the monitor panel will give user a dynamic image that consists of graphic information providing the essential information of the current system status. Unlike a normal web page image, the dynamic image is regularly generated by the server according to the system status, sent to the clients, and is automatically refreshed after a certain period of time. This provides clients real-time information about the system. In order to achieve the above functions, the server push mechanism has been used. The basic principle of the server push mechanism is that the information sending action is based upon information changes, monitored by the server, rather than on the client request. This not only speeds up client information updating, but also reduces the server loading. Choosing the best media for different interface tasks and minimizing the amount of irrelevant information in the interface may not be forgotten because the irrelevant information may obscure important information by attracting the attention of a user.

4.1.2. WBCADA Software in the Thesis

This software has been developed to briefly exemplify the technologies and capabilities of web based control and monitoring technology. Java programming language has been selected because of its benefits and ease in developing web applications. The program resides in the PLC Web Server as a java applet. Applets are client side programs that await to be run from remote locations. Java 2 Runtime Environment installed computers are able to connect the servers by using web browsers such as Internet Explorer 6.0 or Netscape Navigator. Platform independent technolgy of java makes it possible to log the different kind of operating systems on to a system. For example Windows, Linux and Sun OS users may connect to a system at the same time without any problem. The software is designed as a multi user, industrial remote control and monitor program. The administrator of the system may define new users with different level of authorities, delete users who have no authorities any more or change the authority levels of the jobs. Administrator also has the abilitiy to reset the user reports and data reports that has been logged in the PLC. The system is protected by a strong log in policy. All users even guests have to be registered by the administrator.

Modicon Premium PLC which is used in the present study supports Modbus Protocol over TCP/IP. This communication protocol used for transmitting and receiving data over network environment which has been briefly discussed in Section 4.1.3.1.

There are 4 different sections which are allowed to the different level of users:

1. Graphical Representation Section
2. Data Reports & User Event Reports
3. Settings Section
4. Administration Section

While guest level users only have the authority to access the graphical representations, administrator has full access to the system. There are 4 different type of job levels.

1. Guest : Minimum authority level.
2. Operator : Medium authority level
3. Engineer : High authority level.
4. Administrator : Full authority.

There are no limits for the number of jobs and users in the system except for the administrator level job. The level of the jobs can be modified by the administrator in the administration section.

4.1.2.1. Graphical Representation

This section mainly focuses on monitoring the real time environments using graphical tools and general information about the system (Figure 4.1). By default all users have access rights to this section because of its output based structure. Although the program is running in the web browser, there is no need to refreshes the screen for updates. Software continues communication with the PLC and refresh its data and graphical parts when needed. At the left most side there is a bar graphic and right top corner there is a speedo meter graphic. These graphs represent the real time values of system analog variables, such as Water Level and Temperature. At the bottom side there is a chart graph. This graphical unit can represent more than one value at the same time as a function of time.

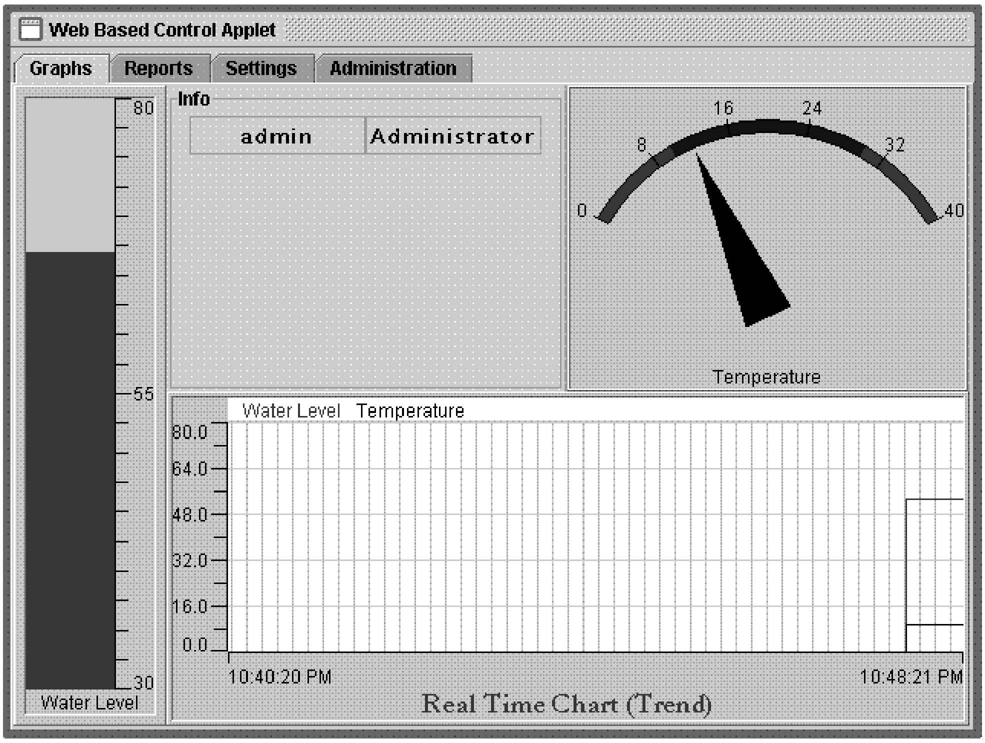


Figure 4.1 – WBCADA Applet Graph Section

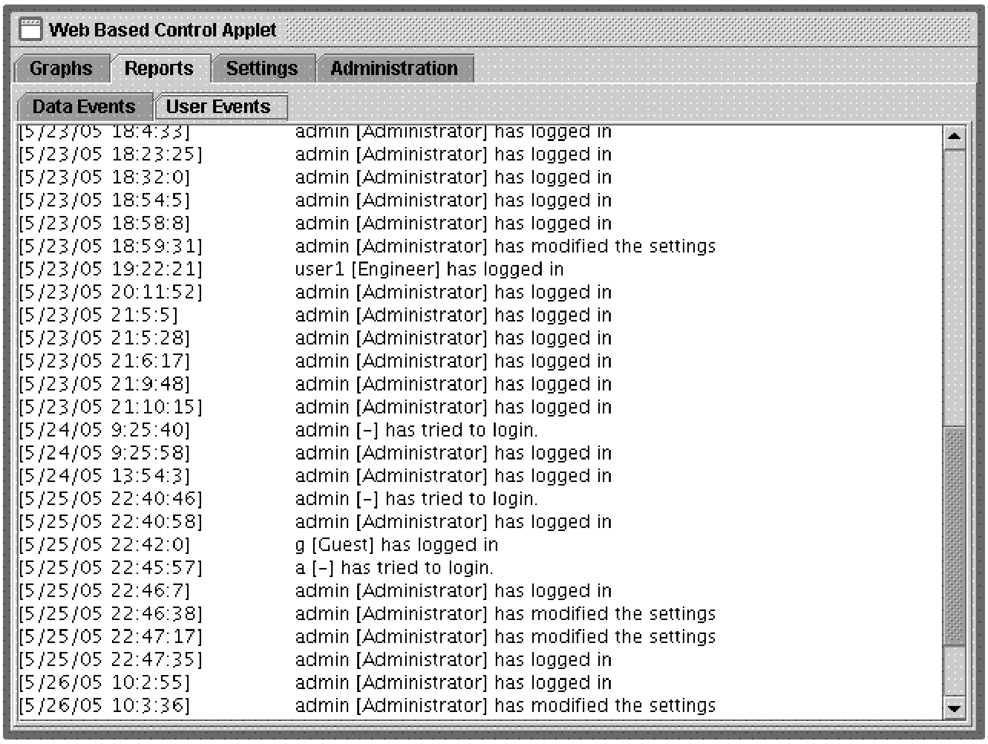
4.1.2.2. Reports Section

The system has a strong report logging structure. These reports can be achieved by the authorized users (by default) whenever they desire. Reports logged can be classified in to two main categories. (Figure 4.2)

1. Data Event Reports
2. User Event Reports

PLC gathers and compares the environment variables with the critical minimum and critical maximum set values. If there are undesired conditions, PLC logs these events with value and time occurred in registers. Users may access these informations in Data Event section.

User Event section is used to log the user activity. Users who have logged in to system, user names who couldn't log in to system and users who has modified the settings are reported here with the complete date and time. Data Reports recorded at the PLC register area and maintained using Modbus protocol while User Reports recorded at the PLC Server as an ASCII file, and maintained using FTP protocol.



```
[5/23/05 18:4:33]      admin [Administrator] has logged in
[5/23/05 18:23:25]     admin [Administrator] has logged in
[5/23/05 18:32:0]      admin [Administrator] has logged in
[5/23/05 18:54:5]      admin [Administrator] has logged in
[5/23/05 18:58:8]      admin [Administrator] has logged in
[5/23/05 18:59:31]     admin [Administrator] has modified the settings
[5/23/05 19:22:21]     user1 [Engineer] has logged in
[5/23/05 20:11:52]     admin [Administrator] has logged in
[5/23/05 21:5:5]       admin [Administrator] has logged in
[5/23/05 21:5:28]      admin [Administrator] has logged in
[5/23/05 21:6:17]      admin [Administrator] has logged in
[5/23/05 21:9:48]      admin [Administrator] has logged in
[5/23/05 21:10:15]     admin [Administrator] has logged in
[5/24/05 9:25:40]      admin [-] has tried to login.
[5/24/05 9:25:58]      admin [Administrator] has logged in
[5/24/05 13:54:3]      admin [Administrator] has logged in
[5/25/05 22:40:46]     admin [-] has tried to login.
[5/25/05 22:40:58]     admin [Administrator] has logged in
[5/25/05 22:42:0]      g [Guest] has logged in
[5/25/05 22:45:57]     a [-] has tried to login.
[5/25/05 22:46:7]      admin [Administrator] has logged in
[5/25/05 22:46:38]     admin [Administrator] has modified the settings
[5/25/05 22:47:17]     admin [Administrator] has modified the settings
[5/25/05 22:47:35]     admin [Administrator] has logged in
[5/26/05 10:2:55]      admin [Administrator] has logged in
[5/26/05 10:3:36]      admin [Administrator] has modified the settings
```

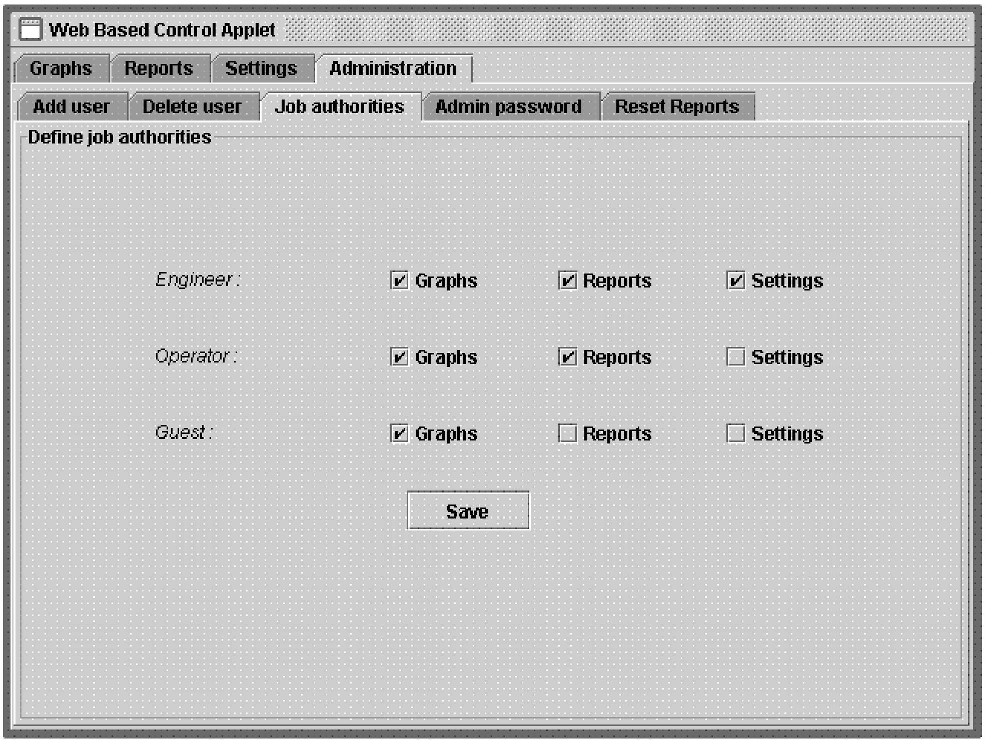
Figure 4.2 – WBCADA Applet Report Section

4.1.2.3. Settings Section

Only users who have high level of authority can access this section. Data ranges, critical minimum and maximum values and names of the environment variables are controlled in this section. It is very easy to cause a fatal control error by entering wrong values. So it is wise to register only a few employees for this section.

4.1.2.4. Administration Section

This is the most private section of the software. Administrator may define new users, delete old users and reset User Event reports and Data Event reports. Job Authorities section enables administrator to define different level of authorities due to needs. (Figure 4.3)



Define job authorities

Engineer : ☑ Graphs ☑ Reports ☑ Settings

Operator : ☑ Graphs ☑ Reports □ Settings

Guest : ☑ Graphs □ Reports □ Settings

Figure 4.3 – Job Authorities of Administration Section

4.1.2.5. Login Section

Web security is a primary concern when dealing with the Internet. Since we plan to have a PLC server that is connected to the Internet or through a dial-in line, we need to look at remote access security. Only administratively registered users can log in to particular sections. Remote access security revolves around the authentication, authorization, and accounting model.

Authentication looks at who the user or entity is. Common authentication implementation is with a username/password scheme.

Authorization determines what each user can do on the remote machine. This is very important to ensure that not everybody can change settings or see critical reports.

Accounting records what the user has been doing. This allows the administrator to look at how users are accessing the system [13].

4.1.2.6. PLC Program

PLC has its own program memory and loops it continuously to control the environment locally. This is a specific graph based logic program known as "ladder program" which compares inputs with the set values provided by WBCADA Software and sets related outputs due to control process. Program also has the responsibility to log the data events such as critical low and critical high real time environment variables.

4.1.3. Communication with the PLC

The common language used in this study to communicate via PLC is the Modbus protocol. This protocol defines a message structure that controllers will recognize and use, regardless of the type of networks over which they communicate. It describes the process a controller uses to request access to another device, how it will respond to requests from the other devices, and how errors will be detected and reported. It establishes a common format for the layout and contents of message fields.

4.1.3.1. The Modbus Protocol

The Modbus protocol provides the internal standard that the controllers use for parsing messages. During communications on a Modbus network, the protocol determines how each controller will know its device address, recognize a message addressed to it, determine the kind of action to be taken, and extract any data or other information contained in the message. If a reply is required, the controller will construct the reply message and send it using Modbus protocol. It is an application layer messaging protocol, positioned at level 7 of the OSI model that provides client/server communication between devices connected on different types of buses or networks (Figure 4.4). The industry's serial de facto standard since 1979, Modbus continues to enable millions of automation devices to communicate. Today, support for the simple and elegant structure of Modbus continues to grow. The Internet community can access Modbus at a reserved system port 502 on the TCP/IP stack.

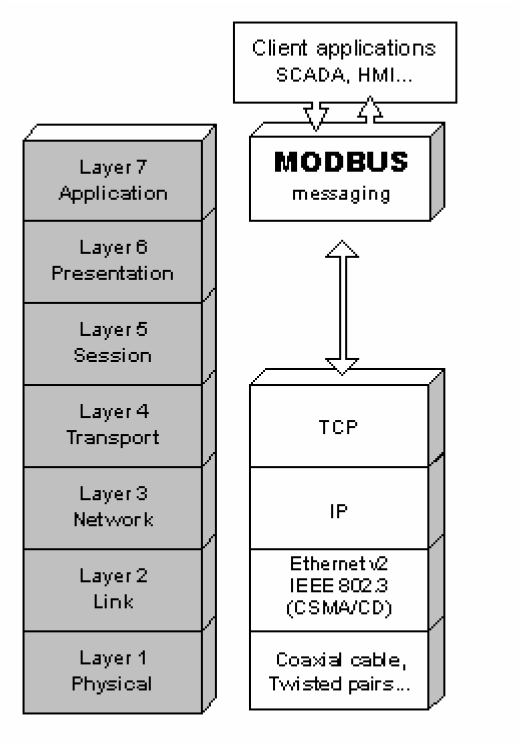


Figure 4.4 – Modbus architecture

Modbus is a request/reply protocol and offers services specified by function codes and an application layer messaging protocol for client/server communication between devices connected on different types of buses or networks. It is currently implemented using:

- TCP/IP over Ethernet.
- Asynchronous serial transmission over a variety of media (wire : EIA/TIA-232-E, EIA- 422, EIA/TIA-485-A; fiber, radio, etc.)
- Modbus Plus, a high speed token passing network.

The Transfer Control Protocol (TCP) guarantees us that all packets sent by the server, and vice versa, will be delivered to the client implying that the lost packets will be retransmitted. TCP also has a built-in mechanism designed to probe and to adapt the sending rate of the packets to the available bandwith. These mechanisms make it a robust and reliable protocol, well suited to transfer bulk data However, from the real-time application point of view, this protocol has a drawback of having unpredictable arrival time of data [14].

4.1.3.2 Protocol description

The Modbus protocol defines a simple PDU (Protocol Data Unit) independent of the underlying communication layers. The mapping of Modbus protocol on specific buses or network can introduce some additional fields on the ADU (Application Data Unit)

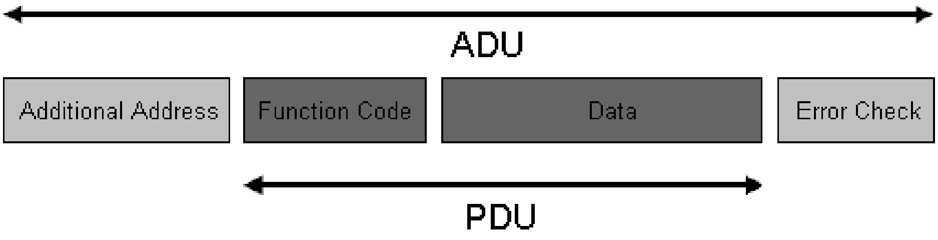


Figure 4.5 – General Modbus frame

The Modbus application data unit is built by the client that initiates a Modbus transaction. The function indicates to the server what kind of action to perform. The Modbus application protocol establishes the format of a request initiated by a client (Figure 4.5). The function code field of a Modbus data unit is coded in one byte. Valid codes are in the range of 1 ... 255 decimal (128 – 255 reserved for exception responses). When a message is sent from a Client to a Server device the function code field tells the server what kind of action to perform. Function code "0" is not valid. Sub-function codes are added to some function codes to define multiple actions. The data field of messages sent from a client to server devices contains additional information that the server uses to take the action defined by the function code. This can include items like discrete and register addresses, the quantity of items to be handled, and the count of actual data bytes in the field. The data field may be nonexistent (of zero length) in certain kinds of requests, in this case the server does not require any additional information. The function code alone specifies the action. If no error occurs related to the Modbus function requested in a properly received Modbus ADU the data field of a response from a server to a client contains the data requested. If an error related to the Modbus function requested occurs, the field contains an exception code that the server application can use to determine the next action to be taken. For example a client can read the ON / OFF states of a group of discrete outputs or inputs or it can read/write the data contents of a group of registers. When the server responds to the client, it uses the function code field to indicate either a normal (error-free, Figure 4.6) response or that some kind of error occurred (called an exception response, Figure 4.7). For a normal response, the server simply echoes to the request the original function code.

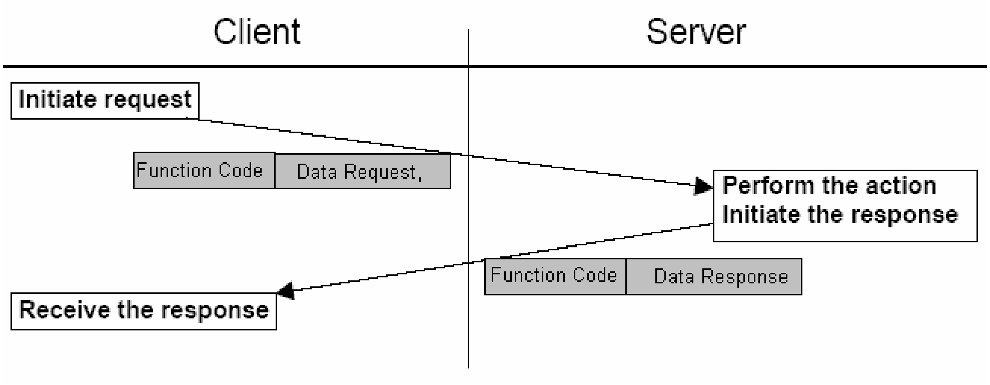


Figure 4.6 – Modbus transaction (error free)

For an exception response, the server returns a code that is equivalent to the original function code from the request PDU with its most significant bit set to logic 1 [14].

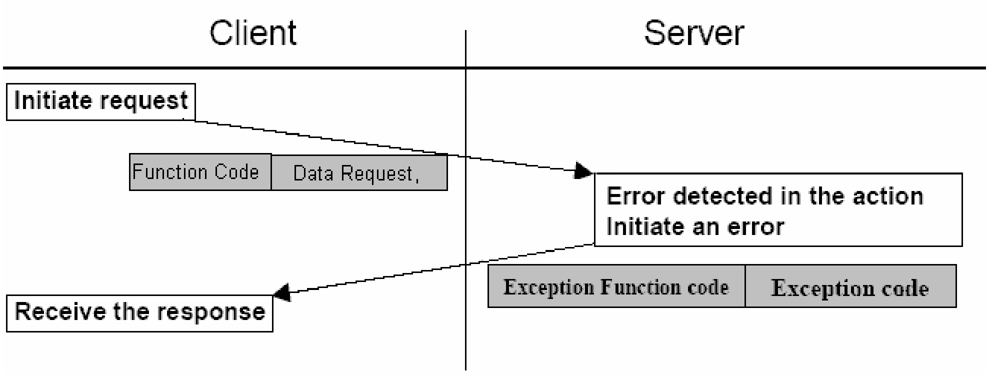


Figure 4.7 – Modbus transaction (exception response)

4.1.3.3. Data Encoding

Modbus uses a 'big-Endian' representation for addresses and data items. This means that when a numerical quantity larger than a single byte is transmitted, the most significant byte is sent first. So for example [14];

Register size	Value	
16 – bits	0x1234	the first byte sent is 0x12 then 0x34

4.1.3.4. Modbus Data Model

Modbus bases its data model on a series of tables that have distinguishing characteristics. The four primary tables are shown in Table 4.1.

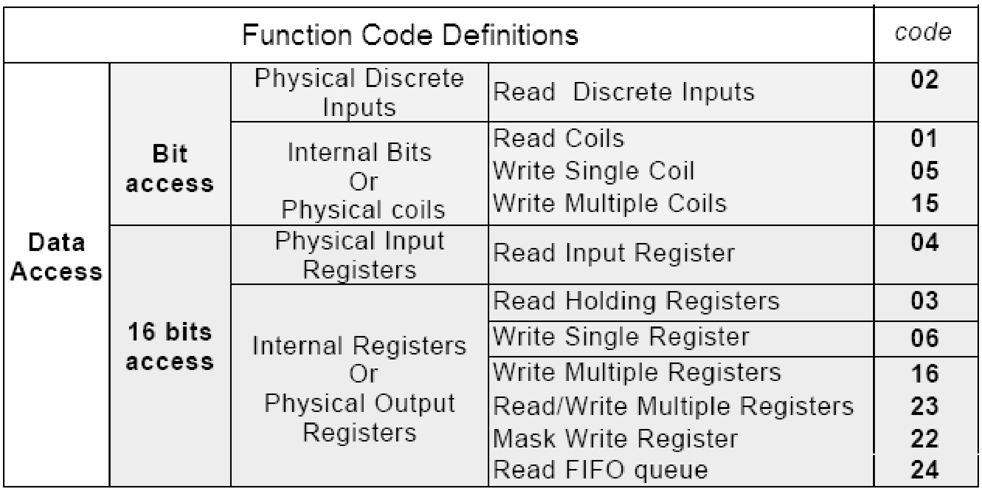
The distinctions between inputs and outputs, and between bit-addressable and word-addressable data items, do not imply any application behavior. It is perfectly acceptable, and very common, to regard all four tables as overlaying one another, if this is the most natural interpretation on the target machine in question

Table 4.1 – Modbus Data Model table

Primary Tables	Object Type	Type of	Comments
Discrete Input	Single bit	Read-Only	This type of data can be provided by an I/O system
Coils	Single bit	Read-Write	This type of data can be alterable by an application
Input Registers	16-bit word	Read-Only	This type of data can be provided by an I/O system
Holding Registers	16-bit word	Read-Write	This type of data can be alterable by an application

For each of the primary tables, the protocol allows individual selection of 65536 data items, and the operations of read or write of those items are designed to span multiple consecutive data items up to a data size limit which is dependent on the transaction function code (Table 4.2). It's obvious that all the data handled via Modbus (bits, registers) must be located in device application memory. The only requirement is to link data reference with physical address. Modbus logical reference numbers, which are used in Modbus functions, are unsigned integer indices starting at zero [14].

Table 4.2 – Modbus Function Code Definitions



Function Code Definitions				code
Data Access	Bit access	Physical Discrete Inputs	Read Discrete Inputs	02
		Internal Bits Or Physical coils	Read Coils Write Single Coil Write Multiple Coils	01 05 15
	16 bits access	Physical Input Registers	Read Input Register	04
		Internal Registers Or Physical Output Registers	Read Holding Registers	03
			Write Single Register	06
			Write Multiple Registers Read/Write Multiple Registers Mask Write Register Read FIFO queue	16 23 22 24

4.1.3.5. Definition of Modbus Transaction

Once the request has been processed by a server, a Modbus response using the adequate Modbus server transaction is built. Depending on the result of the processing a positive Modbus response or an exception Modbus response are built [14].

The Figure 4.8 describes the generic processing of a Modbus transaction in server side.

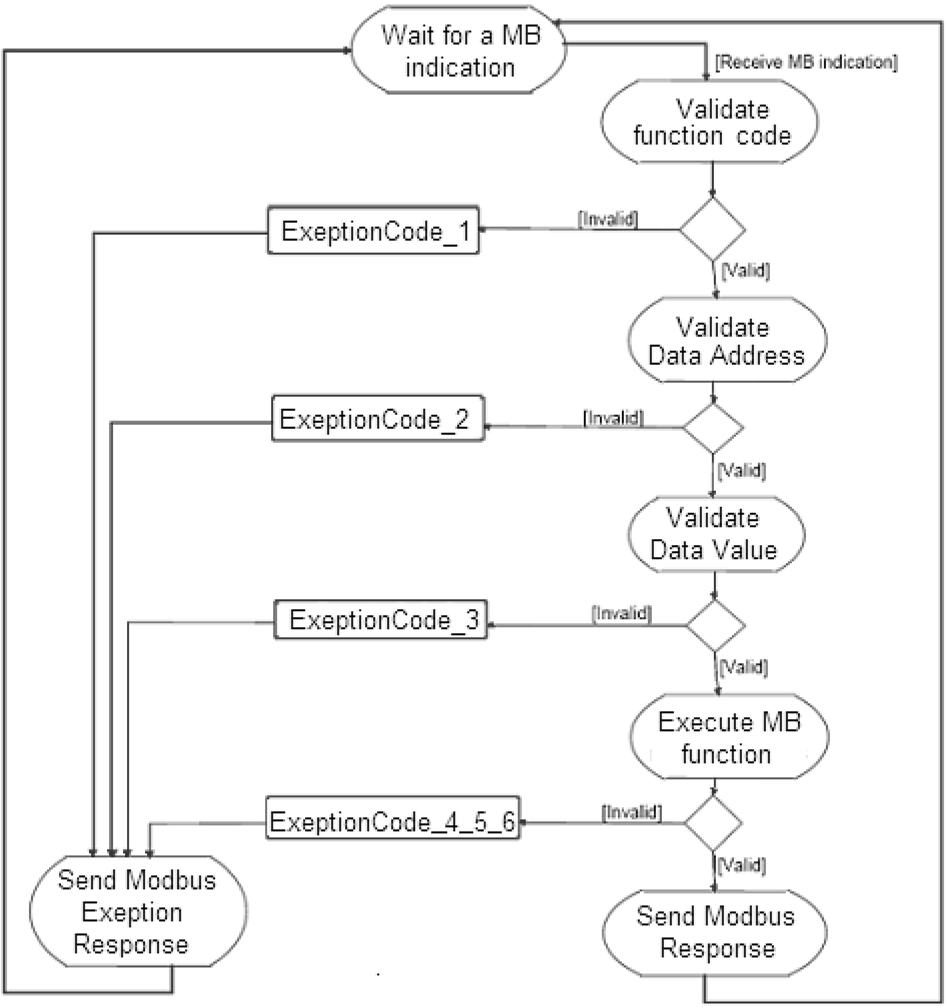


Figure 4.8 - Modbus Transaction state diagram.

4.1.3.6. Finalization

After completing every message preparation according to Modbus protocol, These include Address message, Function message, Data message(s) and CRC message, Master device is ready to send all messages through the network line. And this procedure continues as Slave device responses.

4.2. HARDWARE

4.2.1 Modicon Premium PLC

The Modicon Premium PLC was selected for the study, mainly because of the extendable rack structure which includes a specific ethernet module (web server) [15].

Processor manages a complete PLC station which is made up of:

- discrete input/output modules
- analog input/output modules
- application-specific modules

(i.e. counting, axis control, step by step control, communication, etc.), which can be distributed over one or more racks connected to the Bus as seen at the Figure 4.9.

Technical specifications of the Modicon Premium PLC:

- 4 to 16 extendable racks
- 512 to 2048 discrete I/O
- 24 to 256 analogue I/O
- 8 to 64 application-specific channels. Each application-specific module (counter, motion control, communication or weighing) comprises n application-specific channels
- A protected internal RAM memory (32 to 96 Kwords) which can

receive the entire application and can be extended by a RAM or Flash EPROM PCMCIA memory card (32 to 512 Kwords in program memory).

- A real-time clock
- Various communication modes
  - ModBus
  - FipWay
  - AS-I Bus
  - InterBus
- The application is designed and installed using PL7 Junior/Pro software under Windows 95/98, Windows 2000 or Windows NT 4.x which offer the following:
  - Four programming languages:
    - Ladder language (LD),
    - Grafcet (SFC)
    - Structured Text language (ST)
    - Instruction List language (IL).
  - A multitask software structure: master task, fast task, event processing using event tasks.

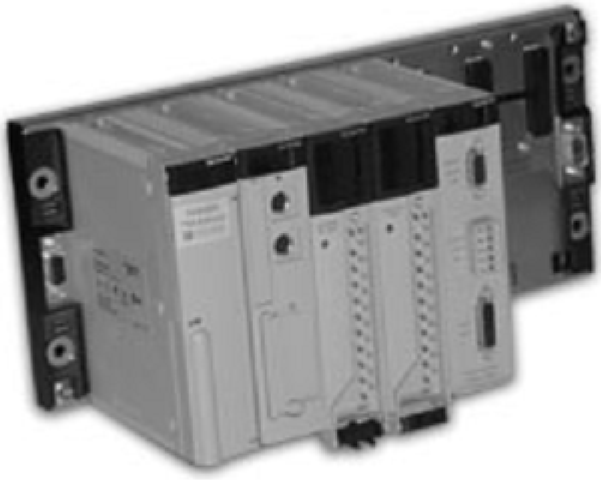


Figure 4.9 – Modicon Premium PLC with its rack

The TSX ETY 110 communication module is used for communication in an Ethernet architecture. These are single format modules which are installed in a rack slot on Modicon Premium PLC station. Module has an embedded web sever of 8 Megabytes and has the ability of TCP/IP Modbus message handling. This server is where the java program resides in the PLC. (Figure 4.10)

Transmission speeds of TSX ETY are 10 Base-T (RJ 45) or 10 Base-5 (AUI): 10 Mbps; User Web pages (1,4 Mo); Client/Server requests: 128 bytes in synchronous mode and 1 Kbytes in asynchronous mode.

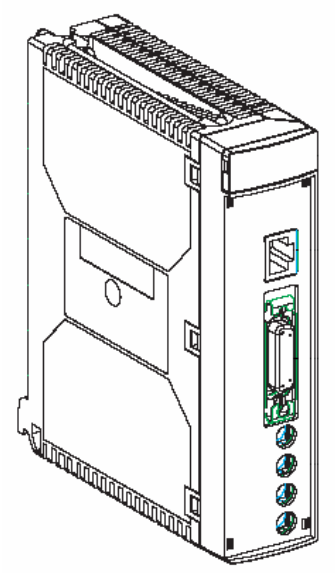


Figure 4.10 – Ethernet Module

CHAPTER 5

EXPERIMENTAL RESULTS & CONCLUSIONS

WBCADA is a relatively young topic. With the improvement of PLC and computer technologies, platform independently connection to these PLCs like web server has became available. This grants users in to a world wide remote control clients. Authorized users, may access to server with the proper IP address and get/set the newest information remotely. Benefits of the WBCADA can be listed as the following:

- Can be login systems from any where in the world
- Browser software support, No need installation of client software
- Platform independent clients
- Only can be viewed by anyone who is authorized.
- Easier and faster to make changes, using the advantages of scalability features of System Controller and RTUs have.
- Personel safety via remote control
- Provides immediate status of the region to Monitor System
- Take up less space
- Limit Loss avoided since computer based implementation handles lots of different tasks, there is no need to pay for lots of laborer anymore.

Although all this looks promisingly two main problems should be faced before the web based control and data acquisition can be implemented. The first one is the aspect of time delay, which can lead to irregular data transmission and data loss. If the Internet is heavily loaded, the responses may be delayed and the operator corrections may take too long time for correcting the system. In the worst-case this can make the whole system unstable. The other one is the problem of security. When malicious hackers can grant access to a system the consequences can be catastrophic. Other problems concern the distance or logistics. If something goes wrong with the system, a lot of time and preparation can be needed before somebody can intervene to requirement specifications and system implementation.

In this study, aspect of time delay caused by the internet load tried to be solved using local PLC program. The PLC system is able to run itself without dependence to any user and has the ability to control the environment variables with logging critical data activities due to preset values which have been set by the users. This architecture prevents critical data loss and control delay over long distance TCP/IP networks like internet.

Security concern also approached as an important issue here. Login policy protects the control system from unauthorized users and attackers. Different levels of jobs developed to maintain and manage security in the system. Whenever a user logs in to the system or changes settings, user name, job and the complete date will be stored in the server. Unregistered login events also being stored in the server with username and complete date to be aware of the system security.

REFERENCES

[1] http://www.zetron.com/

[2] Warnier E., Yliniemi L., Joensuu P.: "Web Based Monitoring and Control of Industrial Processes", Report A No 22, University of Oulu, September 2003.

[3] Stadtwerke E. S.: "The SCADA and Energy Management System", Municipal Utilities Company, March 2001.

[4] Yang S.H., Chen X., Alty J.L.: "Design Issues and Implementation of Internet-Based Process Control Systems", Control Engineering Practice, Volume 11, June 2003.

[5] Tan L.: *Mobile Scada with Thin Clients,* Department Of Engineering Australian National University, 2003.

[6] Ramakrishna V., Zhuang Y., Hu S.Y., Chen J.P., Ko C.C., Chen B.M., Tan K.C.: "Development of a Web-Based Control Experiment for a Coupled Tank Apparatus.", American Control Conference, Volume 6. 2000.

[7] Salzmann C., Gillet D.: "Real-Time Control Over the Internet", 15th IFAC World Congress, Barcelona, Spain, July 2001.

[8] Yu Loong L.: *Remote Control of Heating, Ventilating and Air Conditioning System with LabView*. Department Of Electrical & Computer Engineering, Missisipy State University. December 2003.

[9] http://www.ddc-online.org/

[10] http://www.racoman.com/

[11] http://programmable-logic-controllers.globalspec.com/

[12] Zhuang H., Morgera S.: "Internet Based Instrumentation and Control", Second LACCEI International Latin American and Caribbean

Conference for Engineering and Technology, 2004.

[13] Furuya M., Kato H., Sekozawa T.: "Secure Web-Based Monitoring and Control System", Industrial Electronics Society, 26th Annual Conference of the IEEE. Vol. 4, 2000.

[14] http://www.modbus.org/

[15] http://www.schneiderelectric.com

[16] Motorola Inc., *Introduction to Moscad and Scada,* June 2003.

[17] Telemecanique, *Modicon Automation Platform Catalogue,* April 2004.

[18] Lee P., Myung L.: "Transmission Modeling and Simulation for Internet-Based Control.", Industrial Electronics Society, 27th Annual Conference of the IEEE, Volume 1, 2001.

[19] Daneels A., Salter W.: "Selection and Evaluation of Commercial SCADA Systems for the Controls of the CERN LHC Experiments" International Conference on Accelerator and Large Experimental Physics Control Systems, Trieste, 1999.

[20] Qiu B., Gooi H.B.: "Web-Based SCADA Display Systems (WSDS) for Access via Internet", Power Systems, IEEE Transactions , Volume15, May 2000.

[21] Yang S.H., Alty J.L.: "Development of a Distributed Simulator for Control Experiments Through the Internet.", Future Generation Computer Systems, Volume 18, September 2001.

[22] Yang S.H., Tan L.S., Chen, X.: "Requirements Specification and Architecture Design for Internet-Based Control Systems.", Computer Software and Applications Conference, 26th Annual International, August 2002.

[23] Lee P., Myung L.: "Transmission Modeling and Simulation for Internet-Based Control." Industrial Electronics Society, 27th Annual Conference of the IEEE, Volume 1, 2001.

[24] Yeung K., Huang J.: "Development of the Internet Based Control Experiment.", Decision and Control, Proceedings of the 40th IEEE Conference, Volume 3, 2001.

[25] Yliniemi L, Lindfors J., Leiviskä K.: *Transfer of Hypermedia Material Through Computer Networks,* University of Oulu, May 1996.