SOFTWARE IMPLEMENTATION FOR DISTRIBUTED MONITORING CONTROL SYSTEMS BASED INDUSTRIAL AUTOMATION USING VISUAL STUDIO. NET

Dr. Hla Myo Tun, U Win Khine Moe, Dr. Zaw Min Naing
Department of Electronic Engineering, Yangon Technological University
No. (3/29), Nawaday Garden, Hlaing Thar Yar Township, Yangon, Myanmar
E-mail: hmyotun@myanmar.com

ABSTRACT
Distributed Control System (DCS) is more popular than any other control systems in the modern industrial processes. This research aims to develop “Real Time Graphical User Interface Monitoring System” for Industrial Automation. The current research project is based on the construction of DCS based Vehicle Spare Parts Manufacturing Plant. The Industrial Local Area Network (LAN) is built between the server and the clients. The client-server model is used for this research because the datagram client program and datagram server program have been created by JAVA programming. This research focuses on cheap and flexible distributed control system. In this research, the simulation of the whole process is conducted in real-time condition by using Visual Basic.Net programming. The parallel ports of Client computers and the controllers have been communicated with parallel port interface VB.Net program.

Keywords: Software implementation, Distributed Monitoring Control Systems, Industrial Automation, Visual Studio, Simulation.

1. INTRODUCTION
Traditionally a DCS is a system of controllers linked by a data network, as a single system. Functionality, physical location, or both separate these controllers. The DCS is typically used in complex process applications where large amounts of I/O and data are required, such as chemical plants or oil refineries. They are well suited to batch processes and have an ability to handle complex interlocks and timing between operations. DCS are multitasking systems able to handle large common databases. A DCS allows for various control loops, using graphical representation of function blocks, and is easier to program than ladder logic-based PLCs. Scan rates can be more predictable than the PLCs. PLCs have scan rates dependent on the amount of I/O (Renold and Krutz, 2006). A DCS has safety features, redundancy, and even diagnostics built into its control philosophy to be more robust with less down time (Renold and Krutz, 2006). The present system has the integrated system of hardware and software reduces engineering time, process simulation and advanced control applications are available and it can be designed for ease of future expansions.

2. DCS BASED INDUSTRIAL AUTOMATION
In modern industries, automation control can be applied to small or large processing plant such as chemical plants, waste water purifying plants, power plants etc (Jenkins, 2005). Vehicle Spare Parts Manufacturing Plant is one of the manufacturing processes. The objective of this system is to produce qualified spare parts of automobile. Other processes are ignored. The production system is shown in Figure 1 with sample block diagram.

Figure 1: Block Diagram of DCS Based Vehicle Spare Parts Manufacturing Plant

2.1 Software Components of DCS
The same types of software components should be used for all the different DCS partitions. They are the following:
- A distributed SCADA system
- Control applications using the SCADA tools
- Interfaces to the front-end
- Interfaces to the external systems
- Non-SCADA front-end control applications

In the vehicle spare parts manufacturing plant, the overall process is used to control by Supervisory Control and Data Acquisition system and Front End System such as Distributed Control System (DCS). It is shown in Figure 2. It can control in various control system. But most of modern industries use DCS system because of
security and reliability of the system. DCS system includes master station including server and client computers.

3. INTERFACING FOR PARALLEL PORT
The parallel communication port is used to communicate, control and collect data with Operation mode linking with RTU. In this research the two Multiplexers are used. It used library files “.dll” to control visual basic.net for input/output from actual process. Windows have other options for driving device, including DLLs. A visual basic program can call a DLL directly to access a DLL. A DLL is a set of procedures that windows applications can call. When an application runs, it links to the DLLs declared in its program code, and the corresponding DLLs load into memory. Multiple applications can access the same DLL. The application system can call DLL procedures much like any other subroutine or function.

Many programming languages are enabling to write and compile DLLs. Creating a DLL can be as simple as writing the code and choosing to compile it as a DLL rather than as an executable (.exe) file. Visual Basic programs can call any DLL, whether it was originally written in Basic or another language. The parallel port in the original IBM PC, and any port that emulates the original port’s design, is sometimes called the SPP, for Serial Port. The parallel port is used in Basic or another language. The parallel port in the original IBM PC, and any port that emulates the original port had no written standard beyond the schematic diagrams and documentation for the IBM PC. Microsoft’s visual basic has been the most popular choice for basic programmers developing window programs. It can be added the Input and Output to the language in a dynamic linked library (DLL). A DLL contains code that any windows program can access, including the program written in visual basic (MSDN, 2005; Dave, 2002; Cornell and Morrison, 2002; PO, 2002). This research includes DLLs for port access: input32.dll, for use with 32-bits VB program (Wakefield et al., 2001; Burchart et al., 1999).

Module modinout
Public outS As Integer
Public outselect As Integer
Public outStart As Integer
Public inRobotGripper As Boolean
Public inConveyor As Boolean
Public inRobotBase As Boolean
Public inRobotArm As Boolean
Public inError As Boolean
Public inWorkPiece As Boolean
Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal Port Address As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" Alias "Out32".

4. EXPERIMENTAL PROCEDURE
The experimental procedures for designing DCS software are discussed later.

4.1 Writing VB.net Software from VB.net IDE
The Project Designer provides a distributed location for managing project properties, settings, and resources. The Project Designer appears as a single window in the Visual Studio IDE, much the same as other designers such as the Form or Class designers. It contains a number of pages that are accessed through tabs on the left-hand side. Information entered into the Project Designer persists when it switch from one page to another, when it build the project, or when it close the designer; an Undo command is available on the Edit menu to roll back changes. It can be accessed the Project Designer using the Properties command on the Project menu.

Visual Basic.Net from IDE is used for monitoring of the DCS system and uses window application form, classes, modules and etc. Properties boxes and Tools box are used to design components as real devices. And the monitoring and interfacing visual basic programs are written in the design vb forms.

4.2 Design Procedure for Main Page
It can be designed a window form by using the software application data and used data by depending on the process from the properties of items. The control devices are used by using the toolbox of the form. And the code program for each component is written to the code viewer by using visual basic. Timer tools are used for timing of data interfacing and input/output process data. A Timer is used to raise an event at user-defined intervals. The window timer is designed by a single-threaded environment where UI threads are used to perform the processing. It requires that the user code have a UI message pump available and always operate from the same thread, or marshal call onto the another thread.
4.3 Design Procedure for Monitoring of Simulation Page
This page uses the simulation of components on the form (frmSimulation). The signal of simulation for each is get to display on the page from timer (timIn_Tick) of main page.

4.4 Implementation Program for Sensor Page
This page is linked with the simulation page to see the condition of sensors including in the system. It uses that by pressing the sensor button (btSensor) from the simulation. This uses back button to back the simulation page. The displaying of image for the running sensors is two colors of design for sensor. It uses timer tool (timer1_Tick) to get signal from input system.

5. IMPLEMENTATIONS OF DISPLAYING RESULT ON COMPUTER.
This section describes design requirements of monitoring results on the personal computer. The real time monitoring software was processed using VB.net and the results were exported to PC parallel port. From then on, the data was sent to display unit. In order to do it, 16F877A microcontroller was used to interface parallel port and computer. These implementations were written in VB.net programming language.

The command provided by VB.net was exported to microcontroller via parallel port (Axelson, 1994; Mustafa, 1994; Hall, 1994). If the signal from the remote terminal unit receive by master terminal unit, VB.net will sent a command to microcontroller (Gomez, 2000; Tun, 2008) to instruct the process of this manufacturing plant. Depending on instruction, MTU’s signal would be sent to the microcontroller. The results are shown in the following figures.

The command provided by VB.net was exported to microcontroller via parallel port (Axelson, 1994; Mustafa, 1994; Hall, 1994). If the signal from the remote terminal unit receive by master terminal unit, VB.net will sent a command to microcontroller (Gomez, 2000; Tun, 2008) to instruct the process of this manufacturing plant. Depending on instruction, MTU’s signal would be sent to the microcontroller. The results are shown in the following figures.

Figure 3: Simulation Result of the Main Page
Figure 4: Simulation Result of Hardware Window
Figure 5: Simulation Result of Animation Window
Figure 6: Simulation Result of Data Window
Figure 7: Simulation Result of the Sensor Window
Figure 8: Simulation result of the Exit Window

6. CONCLUSIONS
This system aims to develop the controlled devices by monitoring the whole system. This monitoring system can be accessed with the other process. This system is developed to control the automation processes such as electrical distributed system, water waste system, and etc. Using this monitoring control system, it is intended to become security and reliability from the processes.

In this research (Juniper, 2009); design and implementation of industrial automation that used Visual Basic.Net programming for the monitoring of DCS system have been studied. Communication system uses parallel port and interfacing circuit. Hardware Units for this process have been designed with the small sample model and tested by connecting hardware and computer, and it can start/stop with component using monitoring system and can see all of state running for the this processes.

REFERENCES


Jan Axelson, 1994. Parallel Port Complete, PO Box 16262, Irvine.


Juniper Networks, Inc. 2009. Architecture for Secure SCADA and Distributed Control System Networks, USA.

Dr. Hla Myo Tun received his Bachelor Degree in Electronic from Mandalay Technological University, Mandalay in 2002, Master Degree in Electronic from Yangon Technological University, Yangon in 2003 and Doctoral Degree in Electronic from Mandalay Technological University, Mandalay in 2008. His professional experiences are as follows: Department Head (2008-up to present) in Technological University (Kalay), University Lecturer (2005-2008) in Technological University (Pathein), Assistant Lecturer (2002-2005), and Demonstrator (1999-2002) in Technological University (Insein). His teaching experiences in Technological University (Pathein) are Control Engineering, Automatic Control Systems, Digital Signal Processing, Digital Control System, SCADA (Supervisory Control and Data Acquisition), Electronic Engineering Circuit Analysis Using MATLAB. His current research interests are Real Time Distributed Control System in Industrial Automation, Missile Control Using MATLAB. Dr. Hla Myo Tun is a member in Myanmar Engineering Society (MES). He received three papers from IEEE publications in Thailand in 2008.