Session 1559

THE PC-BASED DISTRIBUTED CONTROL SYSTEM SIMULATOR

Wayne L. Brown, P.E
DeKalb Technical Institute

ABSTRACT

This paper is intended to provide faculty, who teach control system principals, with a conceptual approach to the design of a personal computer-based distributed control system simulator employing industry-grade components in lieu of prepackaged trainers.

INTRODUCTION

Until recently, personal computers were never considered to be employable as the host computer in distributed control system (DCS) schemes. These systems relied on mini computers, loosely called mainframes, and proprietary DCS units. Therefore, the use of DCS schemes were restricted to companies that could afford the large capital outlay required. Due to this constraint, many small companies have not incorporated this type of system in their manufacturing processes. However, with the advances in PC technology and control system software or supervisory control and data acquisition (SCADA) software, distributed control system schemes have become more affordable for the small company. Since the DCS system reveals instantly accurate patterns and trends based on real-time data, advantages such as reduced waste and product consistency can be obtained. In addition, management can receive production data, recorded by the host PC, through a local area network (LAN) or a wide area network (WAN). With this, management decisions can be made using real-time data. Therefore, the small company can compete with the large company product on the quality scale.

With an industrial-grade DCS simulator that has been designed on-site by students in their laboratory classes, participating students will be better prepared to design and implement PC-based distributed control systems on the job. In addition, introducing small and medium size companies to the relatively inexpensive PC-based distributive control system can be a great opportunity for graduates.

THE SIMULATED MANUFACTURING PROCESS

A simulated manufacturing system with two continuous processes must first be selected. Since it is easily accessible, a suggested process is the conditioning of tap water with the continuous processes as temperature and pH control.

Tap water can be heated inexpensively and its pH can be altered easily with household vinegar and bleach. This process can be employed not only to teach students the electrical and mechanical design principals of control systems but it can also be used to teach principals of fluid dynamics and chemical principals such as chemical anticipation.

IMPLEMENTING THE PC-BASED DCS SCHEME

Designing a distributed control system typically means incorporating a DCS scheme into an existing process. This can be accomplished by identifying the components required to communicate with existing equipment such as smart controllers and transducers. Therefore, the basic and most important DCS components are the control system software, host computer, and communications network. Since the SCADA software establishes software and hardware platform requirements and is the heart of any distributed control system, the most efficient design procedure begins with the selection of the control software or SCADA.

There are several SCADA software packages available, such as USDATA's FactoryLink, Intellution's Fix DMACS, and WonderWare's InTouch. For efficient operation, the SCADA package selected should operate on a multitasking platform and possess the capability of communicating on a LAN or WAN. Also, the package should include a substantial symbol library. This last requirement may appear minor. However, drawing symbols in the process's functional diagram can be extremely time consuming.

The software platform requirement is important to the efficient and dynamic operation of the SCADA package. Multitasking operating systems have typically, in the past, been employed by supervisory and data acquisition software developers. In fact, this has been the primary reason for the delay in using personal computers in manufacturing control system applications. Due to the limited speed and memory capacity, the personal computer was not capable of employing a multitasking operating system. With speeds of three hundred megahertz, RAM memory of one gigabyte, and hard disk drive storage of several gigabytes, current PCs are not limited to single tasking, single user applications. In fact, multitasking platforms such as OS/2, WindowsNT, Windows95, and Linux, which are true multitasking operating systems, are in common use today.

The capability of interfacing with existing smart controllers must not be overlooked when examining SCADA software. The smart controllers, typically programmable logic controllers, usually specify the network topology and protocol to be employed. Therefore, some of these packages require additional software drivers and hardware to communicate with the smart controllers.

In order to gain one of the major advantages of the DCS scheme, which is just-in-time manufacturing, a high-speed communications network must be employed. Therefore, the existing smart controllers must be network compatible. Many network compatible smart controllers are equipped with proprietary networks, referred to as data highways. To employ an existing controller with this type of network, in a distributed control system, a hardware interface must be installed in the host computer. This piece of hardware is typically known as the network interface card (NIC) and is the gateway to the communications network. Since there are several types of NICs, careful consideration must be given to the topology and protocol employed by the controller's data highway in order to insure compatibility.

Even though performance, ease of use, and interoperability are key evaluation criteria for any SCADA software package, the following is intended to provide the manufacturing engineer with a concise list of SCADA software evaluation criteria.

1. INTEROPERABILITY.
   This refers to the interaction of all control system hardware and software components at all levels.
2. INTERCONNECTIVITY.
   This criterion is concerned with the transmission medium, which is constrained by the network topology and how efficiently the system's components communicate with each other.
3. DISASTER PROCESSING.
   This component is defined by the efficiency with which the software provides the operator with system failure information and the ease at which the operator is permitted to bring the system back to maximum operation after system failure.
4. DATABASE.
   This refers to the software's ability to maintain the system's database.
5. PROCESSES/DATA.
   This criterion is concerned with the variety of processes and data that can be controlled by the SCADA package.
6. DIAGNOSTICS.
   The SCADA package's ability to assist in the resolution of system failures is evaluated by this diagnostic utility.
7. SECURITY.
   This component is concerned with the levels of security provided by the software.
8. MONITORING/CONTROL
   Monitoring of a given process in real-time and control of that process, within preset parameters, is evaluated by this criterion.
9. ALARM MANAGEMENT/LOGGING.
   This is the category for detecting, annunciating, managing, and storing alarm conditions.
10. STATISTICAL PROCESS CONTROL.
    This is the portion of the SCADA package that evaluates the process data. Production and quality is greatly effected by this data.
12. OPERATOR INTERFACE.
    The graphical user interface (GUI) is evaluated using this criterion.
13. TRENDING.
    The software's ability to display trending plots using historical and current data is considered in this category.
14. REPORT GENERATION.
    The production of logs and reports using current real-time data and data retrieved from historical files is evaluated under this category.

CONCLUSION

Due to the low cost of PCs and advances in computer technology, the personal computer-based distributed control system scheme could become a cost efficient and valuable part of an existing process control system.

With a little computer knowledge, there are numerous automation opportunities available to the manufacturing engineer.

REFERENCES

1. Duffy, C.A., "Sara Lee Top Three To Fill Corporate Needs", PC Week, pp. 70, 1993.
2. Frem, E., "Benchmarks Provide Unexpected Learning Experience", PC Week, pp. 73, 1993.
3. Stallings, W., Local and Metropolitan Area Networks, Macmillan, 1993.
4. Waterbury, R.C., "OIs Are Operator Eyes, Ears, Nose, And Hands", InTech, pp.26-28, 1992.
5. Mueller, S., Upgrading & Repairing PCs, QUE Corporation, 1997, 8th Edition.

WAYNE L. BROWN

Wayne Brown received his Bachelor's degree from Southern Polytechnic State University and his M.S. degree from East Tennessee State University. He is a registered professional engineer and teaches engineering technology at DeKalb Technical Institute.