N94-21335

DISTRIBUTED INTELLIGENT CONTROL AND STATUS NETWORKING

Andre Fortin
Manoj Patel
NASA/Goddard Space Flight Center
Network Control Systems Branch
Greenbelt, MD 20771

Summary

Over the past two years, the Network Control Systems Branch (Code 532) has been investigating control and status networking technologies. These emerging technologies use distributed processing over a network to accomplish a particular custom task. These networks consist of small intelligent "nodes" that perform simple tasks. Containing simple, inexpensive hardware and software, these nodes can be easily developed and maintained. Once networked, the nodes can perform a complex operation, without a central host. This type of system provides an alternative to more complex control and status systems which require a central computer. This paper will provide some background and discuss some applications of this technology. It will also demonstrate the suitability of one particular technology for the Space Network (SN) and discuss the prototyping activities of Code 532 utilizing this technology.

Introduction

Over the last two years, under a research project, the Network Control Systems Branch (Code 532) at Goddard Space Flight Center has been investigating distributed intelligent control and status networking technologies. These emerging technologies use distributed processing over a network of smart devices or "nodes" to accomplish a particular custom task, like home automation, without a central host. These networks are designed to be simple and inexpensive for an engineer to develop and implement a custom task. This paper will provide some background and discuss some applications of this technology. It will also demonstrate the suitability of one particular technology for the Space Network (SN) and discuss the prototyping activities of Code 532 utilizing this technology.

Background

A continuing trend to convert from centralized to distributed intelligent control and status systems seems to have become widespread. A wide spectrum of systems designed for industrial automation, building controls, automobiles, and consumer products are following this trend. The reasons for this trend are numerous.

Centralized control and status systems consist of remote sensors that provide feedback to a central microcontroller, which in turn sends signals to monitors and actuators. Each centralized control system is unique with its own input/output and processing requirements. It consists of large computational engines that process the inputs and outputs of a whole suite of sensors and equipment and relay the changes and actions to the appropriate targets. These systems are expensive to develop and install, and very difficult to expand and maintain on an ongoing basis.

Intelligent distributed control and status systems consist of nodes with embedded intelligence where each node performs a simple task. Nodes may be devices such as proximity sensors, switches, motion detectors, and relays. While individual nodes in this system perform a simple task, the entire system performs a complex operation such as automation of a building, factory or a house, without the need for a central host.

Intelligent distributed control systems have a number of significant benefits over centralized control systems. These systems have lower starting costs by using simple and common methods of communicating between all nodes of the system, like using one protocol. The networking nature of these systems facilitates graceful growth by easing expansion and reconfiguration. These systems cut installation costs by sharing common wiring among nodes. Finally, the flexibility of these systems allows many diverse products and applications to interoperate.

Existing networking technologies such as local area networks (LANs) could be used to network these distributed processing systems. However, the Ethernet protocol used to operate a LAN would be an overkill for the requirements of a control and status network. Ethernet is designed to move huge amounts of user documents, data bases, and graphics files among computers, file servers, and printers. A control/status network involves moving relatively short data frames carrying command and status information that trigger actions within devices. Ethernet is much more complex than what is necessary for a control/status network. A control/status scenario could be implemented with a LAN. Such a system, however, would be a large, expensive development effort with complex hardware and software for each node. New technologies have recently been introduced that provide a full package of hardware and software development tools, a microprocessor or microcontroller, I/O interfaces for interfacing to sensors and actuators, a network operating system, and a communications protocol to develop these distributed intelligent control and status networks.

Examples of Uses

Targeted uses of distributed intelligent control and status networks include industrial, home, and office automation. Particularly custom uses are known to exist for cars and mobile homes. These environments have special needs. The nodes must be small, inexpensive, and sometimes use existing wiring. Some of theses networking technologies have multiple physical media available to allow for node to node communication, such as existing power lines, RF, and twisted pair cable.

A good example frequently given by distributed intelligent control and status networking vendors is a system that controls and monitors common equipment found in the home or office. Consider a node in every device, such as the power outlets, the lights, the light switches, the smoke detector, the major appliances, and the security system. Such a system, in the event of a fire, could disable power going to the outlets, flash the overhead lights near emergency exits, turn off major appliances, like a furnace, and perhaps turn on the security system (if there is no fire alarm, like in a home). This would all be accomplished without a central host. The smoke alarm node would detect the smoke and send the appropriate messages to the applicable nodes. All of these nodes would be networked via the AC power.

Another good example is a system to simplify the wiring in a vehicle. In the dash board and steering column of a vehicle, large amount of cables run from the passenger compartment to the engine department to control various devices around the car. The lights (rear brake lights, parking lights, and headlamps) are just one subsystem within a vehicle that needs controlling from the operator. Presently these devices are controlled by many devoted wires running from the steering column/dashboard to the individual lamps. Consider one wire running between everything, the headlamps, the brake lights, the parking lights, and the switches/controls. Each lamp would contain a node that would control it and each switch/control in the passenger compartment would also have a node. These nodes would signal each other based on operator input. The headlamp node would turn on the high beam if it received a signal from a switch node located near the operator. Because of the networking nature of this technology, the parking light node would ignore the message to activate the high beam since this message is not for that node.

From the previous examples, it can be seen that this networking technology provides for simplification of wiring and providing for automation. Other features include simplicity of engineering and maintenance. Nodes, like in the previous example, that are designed to be light nodes can be generic. Specific features of a light node can be determined at installation time and not at manufacturing time. A building engineer or home owner would not have to stock various light nodes for repair, special features like blinking during a fire alarm (assuming the node is the closest light to an escape exit) can be programmed at installation time. Ease of engineering is also a good trait of these technologies, due to the nature of the simple hardware and processors used in the nodes.

There are several vendors that offer these technologies in which some are very focused on a particular application. General Motors has a system called CAN, targeted for vehicle automation and control. CEBus, a home automation standard, is available free for any manufacturer to design nodes for. The Local Operating Network (LON), designed by Echelon Corporation, is a flexible technology that can be used in many applications and environments. It may be well suited for systems here at NASA. All of these proprietary distributed intelligent control and status systems can provide homes, offices, industrial plants, and large systems with simple methods to meet their control and status needs.

Local Operating Networks (LON)

The current control/status systems in the Space Network are centralized and card-caged. Some of these systems involved a considerable amount of design and development effort and are very difficult to expand and maintain. Additionally, the equipment utilizes a number of different interfaces such as GPIB, MS 1553, RS232, and RS422. The control systems for this scenario have dedicated, point-to-point connections to each equipment. Therefore, interoperability between the equipment is a major problem that demands a large amount of software overhead and processing requirements from the centralized computer system. LON offers a single chip, networking solution to these problems that is inexpensive, modular, interoperable, flexible, and easy to develop. Other control/status networking technologies that were considered were focused on particular applications and did not provide the flexibility demanded by the Space Network systems. This technology has the most potential for successfully replacing or complementing a variety of centralized data processing, data distribution, and fault isolation and monitoring systems within the Space Network.

The technology centers around a single VLSI chip called the Neuron. This inexpensive, multiprocessor chip (less than $10) implements a multitasking network operating system, a networking protocol that conforms to the 7 layer OSI model, and a flexible input/output interface through built-in device drivers. Neuron chips provide all the necessary functionality to intelligently process a node's inputs and outputs and relay that information to other nodes over a local network. One Neuron is required for every node on the distributed control and status network. A variety of nodes can be designed each serving different functions. Once these nodes are created and communicating among themselves, a distributed processing control and status network called a LON is created.

The integrated hardware/software development system of LON allows the engineer to easily design and test a small, simple node with custom hardware tied to the Neuron's generic I/O interface and custom application software that defines the functionality of the node on the network. This software is written in a unique, network programming language called Neuron C. This proprietary language is an extended version of ANSI C that provides support for new data types, statements, and function libraries. These enhancements optimize the chip for hardware interfacing to I/O devices and real-time, node-to-node communications on a network.

Neuron C shields the programmer from the hardware details of input/output processing, the protocol details of the physical medium being used, and the details of packet generation, addressing, and transmission. This makes the application software short and simple and spares the programmer from learning a new protocol. The development system allows the engineer to thoroughly debug the node's hardware with this application code through a process of single-stepping through hardware breakpoints. The development system's capability for network management and monitoring allows the engineer to test the nodes with other nodes on a prototype network on this system and manage the network once all the nodes are installed and operational. Thus, LON's integrated development environment makes node design quick and easy.

The protocol in which these Neuron-chip based nodes talk to each other is called LONTALK. This 7-layer protocol has been optimized for movement of relatively short data frames carrying control/status information. It provides support for multiple transmission media which include twisted pair (which runs from 4.9Kbps to 1.25Mbps), radio frequency (4.9Kbps to 625Kbps), powerline (10Kbps), and even custom media to suit the designer's specific needs. The protocol is actually transparent to the medium used, allowing a variety custom distributed control and status networks to be easily created.

LON technology simplifies the development and maintenance of distributed, intelligent control and status networks by providing an inexpensive communications and control processor, an integrated hardware/software development system, and a control/status networking protocol in one package. Thus, these distributed nodes are designed, developed, tested, and installed with minimum effort and time. Adding a new node to the system is simply a matter of connecting it to the network and making other nodes on the network aware of the new node. Thus, expansion is simplified. The LONTALK protocol solves the problem of interoperability among different nodes. Thus, Local Operating Networks resolves the problems of increased node development time, increased installation costs, difficult expansion, and a lack of interoperability currently experienced in the design of centralized control and status systems.

Prototyping Activities of Code 532

After researching into several distributed intelligent control and status networking technologies, Code 532 decided to evaluate the LON technology. Under a research project, Code 532 procured the necessary hardware and software to evaluate and prototype LON. A technical paper was written that describes in detail what was accomplished in the research and evaluation phase of the LON. Technical information can be found in this paper located in the Code 532 branch library.

After the research and evaluation phase, a prototype system using LON was placed in an operational environment in the Network Control Center (NCC). This system, called the Block Rate Monitor (BRM), consists of a block counter, a gateway, and a host computer. The block counter is a LON node that passively monitors four NASCOM lines. This node counts the number of NASCOM blocks per second and checks for clock presence. The statistical data is then passed over a LON dedicated twisted pair cable to the gateway. The gateway is also a LON node that converts data sent over the LON to RS-232. This protocol converter allows a workstation to interface to the LON. The workstation receives the statistical data and provides the operator with bargraphs (see figure 1) and stripcharts (see figure 3) to evaluate existing traffic. The operator can also graph logged data for a historical perspective of line traffic (see figure 2). The workstation logs 20 days worth of traffic statistics for four NASCOM lines.

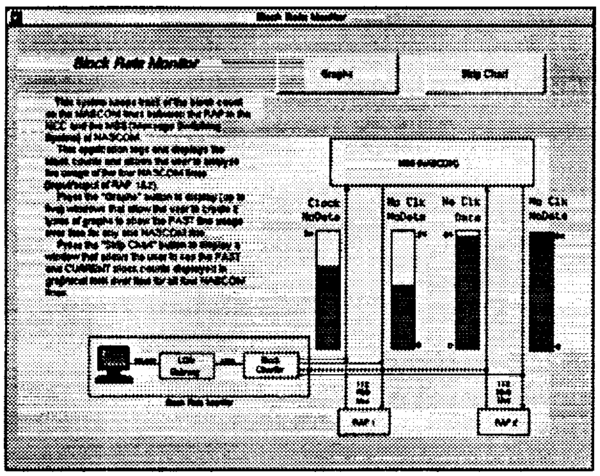

Figure 1. Example main window with bargraphs.

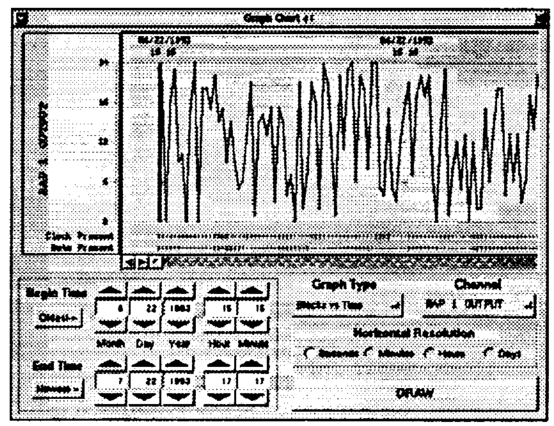

Figure 2. Example of historical data

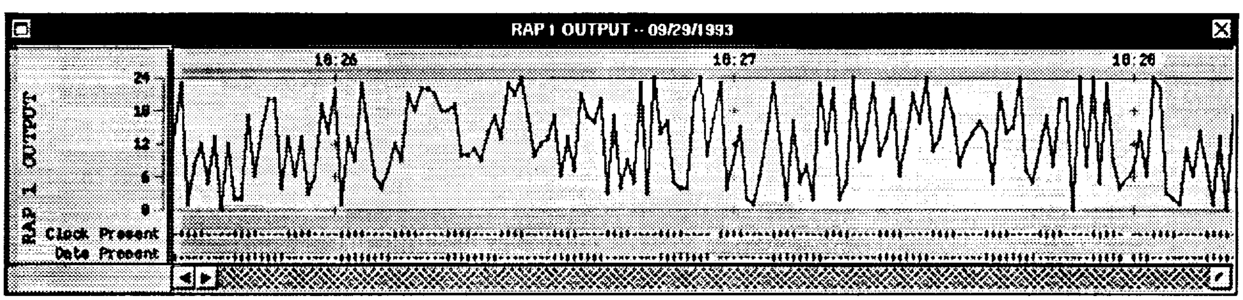

Figure 3. Example strip chart output from BRM application (driven with random data).

With the Block Rate Monitor, the operator can conveniently monitor, in real time, the line activity from a console. The operation of a transmitting system can be deduced as well. If it is not generating any blocks, the system could be down, or there could be a bad cable or patch. This system has already been used to determine if line rate increases were needed (from 112Kb/s to 224Kb/s), but these monitored lines do not have heavy traffic loads and therefore an increase in the line rate was not approved. This system is a very small LON. It only consists of two LON nodes, a block counter, and a gateway. The next phase of our prototyping a control and status network in an SN environment will greatly expand the LON and the concept of remote traffic monitoring.

The second prototype system is called the Data Integrity Monitoring System (DIMS). This system will consist of 40 LON nodes that each monitor a NASCOM line. Each node, or Data Integrity Monitor (DIM), accumulates statistical data of one NASCOM line and is physically the size of 2 cigarette packs. The cost of each node will be around $350. These nodes can count blocks, evaluate the CRC, count bad blocks, determine the clock rate, and detect inverted data. With each node monitoring different NASCOM lines, a LON of nodes is created that relays line activity to the operator. Just as in the first prototype, a gateway receives the status information from each DIM and sends it to a workstation.

This workstation will have a new application that will allow the operator to use this system for performance monitoring, fault isolation, and traffic monitoring. With DIMs located on the inputs and outputs of a system, performance monitoring would be achieved. By comparing the graphs of the traffic on the input and the outputs, a time difference should be observed and indicate the time it took for a block to be processed. Since the workstation application has detailed physical diagrams of the site interconnections along with traffic gauges on the lines, fault isolation would be achieved by an operator noting that a line showing no activity should, in fact, have some activity. This would direct the operator to step back in the diagram of the system, to find the break in the system. The operator would have to determine by this application, which patch panel has been mispatched, which cable has gone bad, or which system has failed.

Our prototypes, utilizing the LON technology, are concepts similar to the Small Network Management Protocol (SNMP). SNMP is designed for Ethernet and works with Ethernet host computers. The DIMS, on the other hand, can incorporate any communication line protocol monitor and works as a layer on top of the existing communications lines. The advantages for a DIMS approach include passiveness, non-interference with operational communication lines/systems, and a multi-protocol line/system monitoring. The advantage for an SNMP approach would be a software only system that monitors Ethernet systems on existing Ethernet host machines. SNMP is more inexpensive and simple, but is limited to Ethernet.

Future plans in these prototyping activities include building nodes that monitor other communication lines (like MIL-STD-1553, GPIB, and Ethernet) and expanding the system to monitor every communication line in the NCC. Future nodes will be much smaller and cost under $150.

Conclusion

Distributed intelligent control and status networking offers a new and unique solution to the traditional methods of performing control and status. The problems of increased development time, lack of flexibility, lack of interoperability, difficult expansion, and difficult maintenance that we face with centralized control and status vanish with this new method of networked control and status.

Local Operating Networks appears to be on the forefront of this new trend by offering an inexpensive, flexible solution that resolves all of the problems experienced in the design of centralized control and status systems. Our prototyping efforts will determine this technology's feasibility to the Space Network systems.

References

Kleiner, Art. "The Ultimate Control Chip: The Programmable World." Popular Science, May 1992, pp. 90 - 95.

LonWorks Products 1992. USA: Echelon Corporation, 1992.