By Richard H. Caro,
Vice President,
ARC Advisory Group.

Real-Time Networks in Industrial Automation

Tomorrow's products will communicate over open COTS networks using COTS equipment running application software over COTS operating systems. Proprietary hardware will no longer exist except where the equipment touches the process itself. This article explains why.

CONTROL SYSTEM ARCHITECTURE

Since the mid-1970's, process control systems for the fluid process industries have been build around high speed data communications networks linking distributed computers. Each computer is assigned some dedicated set of functions such as closed-loop feedback and feedforward control, human-machine interface (HMI), and historical data collection and reporting. Distributed Control Systems (DCS), now use personal computers for HMI and data collection, but with the need for high reliability, controllers have typically used highly proprietary hardware and software. Networks used to link distributed elements have been highly proprietary.

Programmable Logic Controllers (PLC) and Computer Numerical Controllers (CNC) are used to operate discrete parts manufacturing processes. PLCs and CNCs have been highly proprietary devices, but are now also based on personal computers with logical processing in software. The first PLCs had the input and output real-world signal processing hardware (I/O) interfaces well integrated with the logical processing unit, but today's I/O is usually mounted close to the process or machine and connected to the PLCs logic solver with a communications network. The PLC originally had no interface for plant operators, but today the PLC is most often interfaced with a PC-based HMI software package. Likewise, CNCs originally contained all of the I/O required to control motion of the machinery for metal cutting or plastics molding, and had little or no operator interfaces. CNC today most often uses PC-based HMI to allow the operator to monitor the machine and make necessary adjustments during setup and operation.

INPUT/OUTPUT

One of the classic problems in industrial automation is the high cost of connecting sensors and actuators which are in the field or shop floor with the control system, either a DCS, PLC or CNC. Point-to-point wiring is now common between sensors and actuators and the I/O attachment point. Remote I/O is designed to reduce the length of these interconnections by locating in the field near the sensors and actuators, but still requires wiring from the field device to the I/O card. Fieldbus was developed to eliminate the need for remote I/O by allowing intelligent sensors and actuators to communicate directly with each other and with the controller. Devicebuses are used to connect remote I/O units to a controller and to each other. Sensorbuses are a different technology intended to eliminate the number of these wires by interconnecting a number of sensors and actuators. Unfortunately, market forces have confused these clear definitions, and allowed all types of field or shop floor data communications to be called "fieldbus."

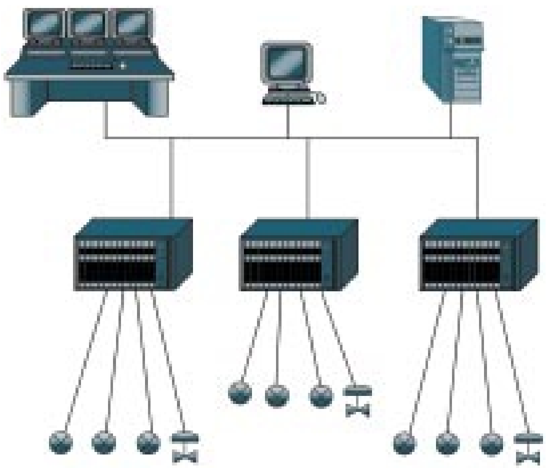

Figure 1. Distributed Control Systems. (DCS)

THE CAN STORY

While communications among intelligent field devices and controllers is being supported by Ethernet and some fieldbus technologies, significant wiring savings is possible with some of the devicebus and sensorbus technologies. CAN (Controller Area Network) was developed to be used on the automobile to save money by reduction in the complexity of the automotive wiring harness. The intent was that the CAN chip be embedded in each sub-assembly such as a light cluster, where it would control each bulb separately in response to a command from a central controller. A simple wire carrying both power and signal would connect all sub-assemblies replacing all of the complex point-to-point wiring. It was only natural for CAN to be selected for industrial automation discrete networks. Unfortunately, the CAN standard (ISO 11898) does not specify the wiring or an application layer. Four separate organizations use the CAN protocol:

- ODVA (Open DeviceNet Vendor's Association),

- SDS (Smart Distributed System),
- CiA (CAN in Automation), and
- CAN Kingdom

Each organization specifies a different wiring plan and their own application interface, and are therefore not interoperable. Many of these differences are being discussed in standards committees within the ISO (International Standards Organization), but the result is expected to be a formal documentation of each of the different choices within a common framework specification. While the CAN chip itself is very low cost due to the large volume of production, the other portions of the circuit add to the cost of an implementation.

EVOLVING REAL-TIME NETWORKS

Another popular communications protocol is Profibus, which was designed to interconnect PLCs and host computers. Profibus is a bus network based on a popular serial communications line, and a token bus protocol. It may be used for peer-to-peer communications, but is most often used in master/slave mode. Less popular than Profibus is WorldFIP, an arbitrated protocol originally invented for the French nuclear power market. Both Profibus and WorldFIP are included in the European fieldbus standard.

Foundation Fieldbus has been developed by a consortium (Fieldbus Foundation) of industrial automation suppliers and is based on the development of specifications by the IEC (International Electrotechnical Committee) fieldbus standards committee included in IEC 61158. This effort has created a network architecture specifically for all parts of industrial automation including process control in the fluid process industries (chemicals, petroleum, metals, forest products, food, beverage, cement, etc.) not well addressed by other buses.

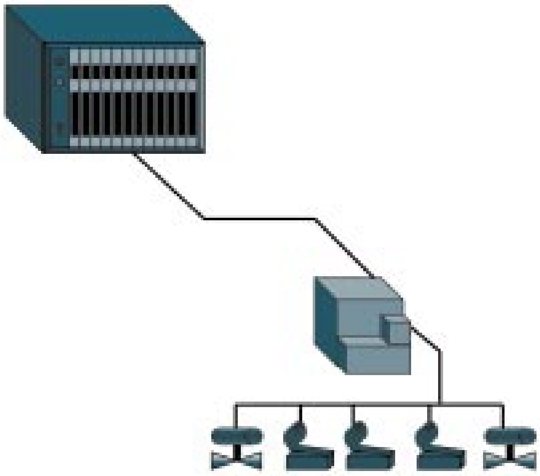

Figure 2. PLC with remote I/O.

ETHERNET TCP/IP

Today, most parts of the industrial automation industry are rapidly converging on Ethernet with TCP/IP as the common data network base technology. Ethernet provides a fast data path at very low cost, especially compared with the cost of proprietary interfaces. The economy of scale in semiconductors applies since annually there are millions of Ethernet chips produced, compared with a few thousands proprietary network interface chips or even the open/standard industrial network chips such as for Profibus, LonWorks, Foundation Fieldbus or WorldFIP. However, Ethernet and TCP/IP only define the low level protocols which move data across networks. Each of the industrial automation networks still has its own unique Application Layer protocol, which defines the nature of the message. Fortunately, all modern operating systems have been built with the understanding that communications net-

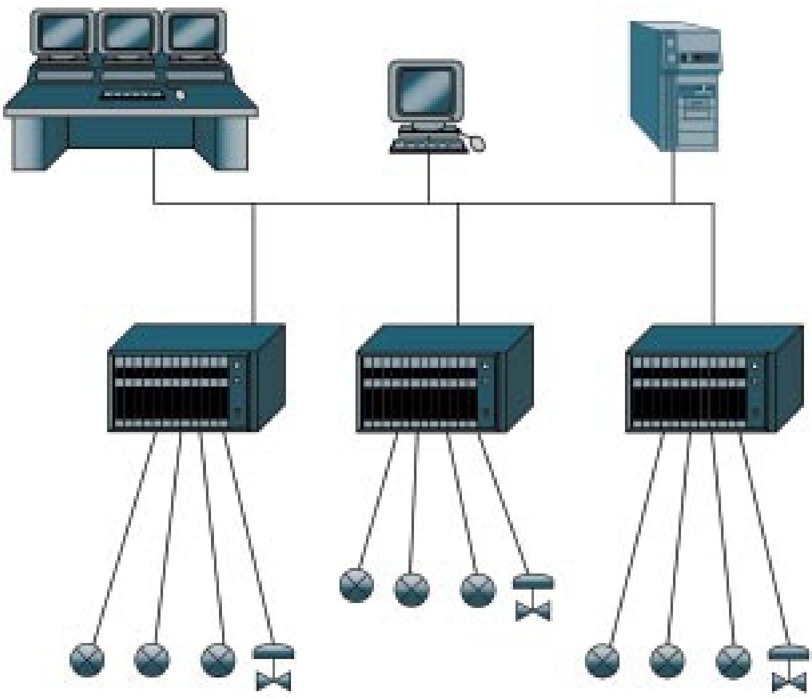

Figure 3. Smart servodrives &motors

works require a unique protocol "stack" which can be added to the communications subsystem very easily. Thus, devices responding to different protocol stacks can be connected to the same network and will all be able to communicate with the same host computer which has the capability to handle multiple communications stacks. They will also be able to interoperate with each other, if that is required, if they are each provided with multiple communications stacks.

MOTION CONTROL NETWORKS

One application in industrial automation which has been seeking network solutions is electronic motion control. The motion control suppliers formed a consortium, IGS (Interest Group Sercos), to develop a common serial protocol explicitly for motion control. Sercos has been formalized as IEC 61491. Sercos features a highly synchronous messaging structure to allow coordination of motion across many axies simultaneously. While the original Sercos chips were somewhat limited in performance, the newest chips operate up to 16 Mbps. Many suppliers of motion control equipment are now in the process of implementing Firewire (IEEE 1394) chips rather than Sercos to take advantage of the much higher speeds of Firewire with dramatically lower parts cost. Firewire also provides a protocol which enables highly synchronous operations, and is included as standard equipment on many PCs.

OBJECTS IN AUTOMATION

So far, we have explored the syntax of industrial data networking, but not its semantics - or the meaning of the data stream. At the lowest levels, each network protocol inserts a packet of data inside its application layer frame which is itself inserted into a TCP/IP structure and sent via Ethernet. In order to provide a common framework for messages, a higher order messaging standard is required to provide interoperability without resorting to a multitude of system level device drivers. The future for industrial automation network data interfaces is in the definition of industrial automation object class libraries. Objects are a suitable abstraction of industrial automation devices such as sensors, actuators and controllers. Each object may have one or more functions, which are called methods in object language. While the methods may not be changed from outside the object, applications communicate to the objects via attributes, which are the variables used by the object for its functions, and the information produced by the object as it works. Complex devices may be defined by linking lower level objects thereby inheriting all of their characteristics. If each industrial automation device has a standard object definition, then communicating with it as an object makes a unique application layer unnecessary.

Objects are an emerging technology for industrial automation and have first appeared in the work of the OPC Foundation. OPC is "OLE for Process Control" where OLE (Object Linking and Embedding) is a

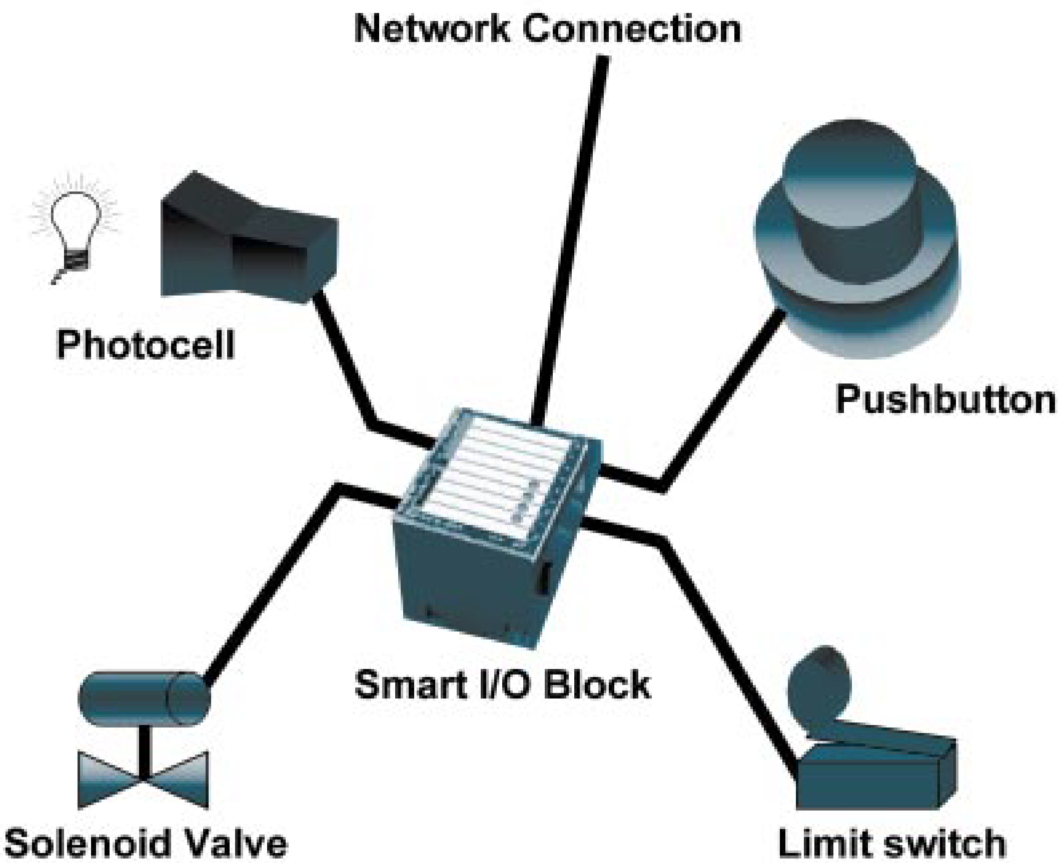

Figure 4. Four-point block I/O.

Microsoft term replace by COM (Component Object Model) and lately by SOAP (Simple Object Access Protocol). Telecommunications and the financial community have focused on CORBA (Common Object Request Broker Architecture), but this technology depends on the integrity of a centralized database (the object broker), which is not practical for industrial automation that is increasingly distributed.

Industrial automation has long been familiar with the concept of function blocks which appear in the Foundation Fieldbus specification for process control, and in the IEC standard 61131-3 Programming Languages for PLCs. These function blocks are true objects, and eventually will be supported in an emerging object framework for industrial automation. Likewise, profiles are being developed for both DeviceNet and Profibus which have all of the characteristics of objects, and will eventually be supported in the industrial automation object framework. While this early work on profiles and function blocks will be the basis for a future object definitions, it is too soon to cast this work into international standards. With a few years of experience and evolution of these definitions, casting them as international standards will make sense by perhaps 2005.

COMMERCIAL OFF-THE-SHELF

Industrial automation has been a bastion of specialized and highly proprietary designs. As a result, while most industrial automation equipment has been highly reliable or even fault-tolerant, it has also been very expensive and difficult to maintain. Suppliers frequently would compete in the basis of the excellence of their hardware. Similar to the changes in the military electronics market, industrial automation is moving away from proprietary designs to more commercial off-the-shelf (COTS) equipment and software. The most profound effects have been due to the rapid cost reduction and increasing performance of the PC. Where proprietary integrated circuits were previously required to obtain the desired performance of a controller product, COTS PCs are now capable of making this same computation with conventional software. Where expensive graphical workstations were previously required to obtain the resolution, detail and speed for HMI, COTS PCs now provide that performance.

DOWNWARD MIGRATION OF INTELLIGENCE

The most notable change in industrial automation is the migration of intelligence from centralized controllers to distributed controllers and now to the field devices themselves. The enabling technology has been the data communications network joining distributed elements. Proprietary networks led to closed systems all supplied by one vendor and a few associated partners. Costs were high per connection, and speeds were slow. Connection of a non-partner's device often required a "foreign device interface" at great cost, slow performance, and questionable reliability. Users have always been dissatisfied with closed systems since they limit choice, always cost more, and make them dependent upon the one vendor for items unrelated to their core competency.

CONCLUSIONS

Today, the paradigm has shifted to open networks joining systems supplied by many suppliers each to the level of their own core competency. Users are now more often able to choose the "best of breed" components for their automation systems and increasingly are able to integrate them without problems. The paradigm shift has not yet been completed, however. There are still suppliers "holding out" to squeeze the last dollar, yen, deutchmark, or franc out of their already sunk investment in legacy technology. The direction has been cast. So tomorrow's products will communicate over open COTS networks using COTS equipment running application software over COTS operating systems. Proprietary hardware will no longer exist except where the equipment touches the process itself. The user will benefit by having the best possible industrial automation system at the lowest possible cost ■

Richard H. Caro is Vice President at ARC Advisory Group.