

Benefiting from Industrial Ethernet at the Device Level with Smart Remote I/O and Peer-to-Peer Technology

Written by: Peishan Juan, Advantech Corporation, Industrial Automation Group

There can be too much of a good thing, or so goes the old saying. Fortunately, new technology can come to the rescue, at least in certain cases. For example, an industrial enterprise can grow too big for efficient automation control using traditional technology. However, advances in industrial Ethernet can provide a solution, if such techniques as smart remote I/O and peer-to-peer technology are implemented. The result could be better and more flexible control, along with a change in the way Ethernet is used.



For one quick example of the benefits possible, consider a site with widely scattered oil tanks. Monitoring the level in those tanks is best done with remote Ethernet-based smart I/O modules. Polling the tanks one by one to get their level works but the amount of time needed goes up as the number of tanks increases. With peer-to-peer capability, smart remote modules can send a message to a control room if the level in a tank is lower – or higher – than a set point. The effect is a real-time update, done in a cost-effective way that scales easily with increasing site size.

This one example shows the possibilities of this new technology. In this white paper, the shift to Ethernet at the device level will be explored and some of the most popular industrial Ethernet implementations described. Peer-to-peer and graphic control languages will be examined. That will be followed by a listing of some of the add-on benefits of industrial Ethernet.

At the Device Level

Having conquered the factory floor from its office stronghold, Ethernet is now migrating down the automation network hierarchy. A study published by the ARC Advisory Group in early 2008 noted that the worldwide market for Ethernet-based automation devices and I/O was a little over one million nodes in 2007. The analysts predicted a steady expansion, with the market growing at a compound annual growth rate of 27.5 percent. By 2012 the number of nodes will top three and be on the way to four million.

Behind this explosion lie several reasons. One is that Ethernet is now well established at the control level, making it logical to think of extending it downward. Secondly, overall improvements in reliability have made it possible to envision using Ethernet in the most demanding motion control applications. A third factor is the advent of automation-applicable standards and intelligent implementation strategies.

One key benefit of this downward push is network commonality throughout the factory floor and beyond. For end users, having Ethernet everywhere means the first two layers of the standard seven layer communication model are the same throughout the enterprise. Having the physical and data link layers the same everywhere eases the task of network integration. It also helps in network configuration and reconfiguration.

This commonality means that a single skill set can handle the entire network, reducing or eliminating the need to maintain specialized technical expertise for other networking technologies. Another advantage is that Ethernet is both available from and supported by major IT and automation vendors, providing a boost to network management. A somewhat hidden benefit is that

an Ethernet-based approach can leverage ongoing technology advances. For example, connections were originally limited to speeds of 10 megabits per second or less. Versions of the standard now offer 10 gigabit speeds, a thousand fold improvement.

The commonality of Ethernet means an easier exchange of data, since at least some layers are common. The same network can potentially route the traffic needed for everything from process optimization to asset management.

There are also signs that the emphasis in this push into automation is on commonality and not necessarily openness. For proof, consider that single vendor applications are increasingly adopting industrial Ethernet. In such cases, end users will not be swapping out one vendor's wares for those of another. Analysts have found the same trend to be the case in embedded applications, which typically run proprietary protocols.

On the other hand, this push toward commonality does not mean that higher level protocols are merging. Indeed, that is where much of the value added claims from vendors arise.

Popular Protocols

This fact is evident in the number of different industrial Ethernet protocols that exist. These all share the same physical and link layer but those above that differ. The leading industrial Ethernet-based networks, according to ARC analysts, are

- * EtherCAT
- * EtherNet/IP
- * Ethernet PowerLink
- * PROFINet
- * MODBUS TCP/IP
- * SynqNet

(Listed alphabetically, not according to market share)

EtherCAT is related to the CANopen serial protocol and is governed by a number of IEC standards, including IEC 61158. The protocol is optimized for short cycle times, with modifications to the handling of the Ethernet packet frame that minimize the delay at each node.

EtherNET/IP is associated with DeviceNET and ControlNET in that it uses the same application layer protocol. EtherNET/IP employs the Internet standard TCP protocols for transport. Another popular protocol used by some industrial Ethernet implementations is UDP, which can be faster and more efficient than TCP at the risk of perhaps lesser reliability.

Despite its name, Ethernet PowerLink has nothing to do with power distribution via Ethernet cabling. Instead, it adds mixed polling and time slicing mechanisms to standard Ethernet to transform it into a deterministic, real-time protocol suitable for machine and device control. As might be gathered from its name, PROFINet is the industrial Ethernet standard controlled by the same organization that manages the serial Profibus standard. Like others, it's found in IEC standards. The protocol has both real time and isochronous implementations, with the former differing from the latter chiefly in the degree of determinism. In the protocol, changes to the priority with which data are treated help enable real-time operation.

Then there's SynqNet. It was developed originally to meet the machine control needs in the semiconductor and electronic assembly markets, as well as elsewhere. Like the others, it was built on the Ethernet physical and link layers.

MODBUS TCP/IP is an Ethernet version of the long-available Modbus serial communications protocol. One difference between the two is that the serial protocol requires a checksum calculation as a way of verifying data while the Ethernet version does not.

While there are arguments to be made for each of these industrial Ethernet implementations, Advantech has settled on Modbus TCP/IP because it offers the needed performance and is part of an openly published, royalty-free, and very widespread communication family.



What all of these industrial Ethernet implementations offers is greater speed than is possible over traditional serial networks. The venerable RS-232 protocol, for instance, offers speeds of only up to 9.6 kilobits per second, far less than what Ethernet allows. Even RS-485, which is a significantly faster, is still at least a factor of three and as much as a factor of 30 slower than Ethernet, depending upon the implementation.

Some of this speed increase is offset by the fact that the minimum frame size of Ethernet is often much larger than the few bytes of data typical for industrial communication. Thus, the data transmission efficiency may be low and perhaps only a few percent, in some cases. Nonetheless, the higher bit rate does win out, particularly in those instances where a substantial amount of data must be transmitted.

Among Peers

There's also another significant advantage offered by industrial Ethernet and the use of smart I/O modules. The combination makes peer-to-peer configurations pay off.

In the more traditional master-client mode found in serial and proprietary networks, a controller has to read the data from input module. It then has to send data back to the output module, completing the input-output loop. Those transactions require a controller, along with wiring to and from the I/O module. This back-and-forth sequence can also take time, particularly if the controller is busy with another task.

Finally, this arrangement can be difficult to scale for a variety of reasons. It may involve a proprietary network, which may only be able to cover a limited distance. If the span is too great, achieving coverage could present a problem and thereby make scaling difficult. Likewise, there could be scaling problems if the number of connections to the controller overwhelms the device or proprietary protocol.

In contrast, peer-to-peer connections run from specific input channels on one module to specific output channels on another. The data automatically transfers from one to the other, simplifying the entire I/O process. One obvious advantage is that no controller is needed, saving system hardware costs. A less obvious benefit is that the wiring can be simplified, since it need only run from one module to the next. If the connections are done using industrial Ethernet, then the wiring options become very flexible.

Speed and scalability can be other pluses with peer-to-peer. The first arises because the controller is eliminated. This change rids the system of any delays associated with communication to and from the controller, as well as any processing time of the controller itself. Scalability results from the nature of the connection and the communication. As far as each module is concerned, it only

connects to its peer, no matter how big the installation is. Communication is initiated by the device when there is data to transfer, again without regard to the number of nodes on the network.

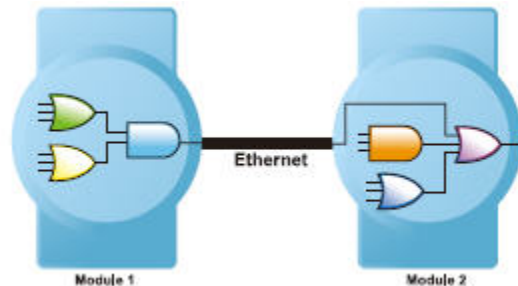
For an example of how peer-to-peer technology can help solve some real world problems, consider the case of a company with three branches situated in separate countries. At the central headquarters, personnel in a control room monitor the gates leading to the different sites over an Internet connection. Given the desire to actively control the gates, one solution would be to run a separate communication network. With peer-to-peer modules, however, that control can be done over the existing intra- and Internet by simply placing a module at the control room and each gate. The control room module can act as a controller for gate through the remote module. There's not need to run special wires or construct another communications channel.

Of course, there are different ways to implement peer-to-peer connections. One is to simply map the channel of one module to another, ensuring the security and pairing of the connection by only allowing a specific IP or MAC address control authority. The other is a more advanced case, with multiple modules mapping to one on either the input or output side. Advantech's ADAM series of controllers offer such a capability. They thus enable flexible channel mapping, doing so while offering a response time of less than 1.2 milliseconds for wired modules and less than 30 milliseconds when in an ad hoc wireless mode.

A Picture is Worth ...

Those modules also offer Advantech's Graphic Condition Logic or GCL. This is a graphical configuration utility that exploits the power of smart Ethernet I/O to make logic control easy. In this approach, the input, logic, execution and output of a logic rule are represented by icons that are configured for a desired outcome.

For example, the input may consist of two digital and one analog signal. These could be combined in a logical AND so that both digital signals would have to be high and the analog input greater than five volts before the output would be generated. That determination would be done according to a clock in a given execution time. As for the output, that could be a digital and analog signal, along with a message to the appropriate computer.



This sounds complicated, even though it actually is one of the simpler control situations likely to be encountered. In a text-based system, such a configuration would have required a substantial amount of what could be cryptic code.

Grasping the function of the module at a glance would be difficult.



In contrast, a graphical representation is easier to initially implement and later decipher, as well as being easier to debug. Going graphical requires two things, though. One is enough intelligence in the module and system to handle the chore of making the complicated appear simple. The other is a utility that gives the human engineer access to the right graphical elements. This utility

need only be available in the development environment. The final configuration can be downloaded to the module and so there's no need for the module to be complex enough to host the development environment.

However, the utility and module requirements are made a bit steeper when Ethernet connectivity is brought into the picture. Then the system has to handle local and remote outputs. In the latter, the two modules may be connected over the Ethernet and actually be physically very remote from one another. The system should also handle distributed cascade logic and feedback functions, which will allow for more complex logic to be implemented. Finally, there's a need to handle a wide variety of analog and digital inputs, including temperature, voltage, current and others on the analog side and contact or counters on the digital side. The same range is needed on the analog and digital outputs.

Advantech's ADAM modules are one example of how such a graphical language can be implemented. They offer other attributes that help process engineers, such as the ability to scale an analog input so that it can be converted from a voltage or current reading to a temperature or pressure, for example. The logic condition can then be a test of that scaled value, an important help when the test is being set up and also of assistance when someone else has to make changes to the system.

It is also important that all of this behind-the-scene processing be done quickly enough. It does no good to have an easy to develop, debug and maintain graphical representation if the hardware doing the actual work is too slow. Thus, it's important to check the processing time to ensure the response is fast enough. For instance, Advantech's GCL modules have a processing time of less than a millisecond when executing locally. Even when the input and output channels are remote and not on the same module, the processing time is less than three milliseconds, with the increase due to communication delays.

Additional Benefits

Finally, there are some add on benefits in going with industrial Ethernet. Since standard Internet protocols like TCP and UDP are used as the transport layer, standard tools can also be used. For example, Web-enabled modules can be implemented. These can be a productivity enhancer, allowing easy access using nothing more than an appropriate web browser. With such a setup, development and collaboration can be done either locally or remotely, without having to learn or maintain new software.

What's more, a module-based Web server can present data to anyone who's interested. In this way information can flow from the factory floor to the front office, satisfying diverse needs with a seamless stream of data.

These are just some ways in which technology advances provide solutions for control engineers and others involved in industrial processing. For such situations, Ethernet improvements, smart remote I/O and peer-to-peer technology can lead to better and more flexible control.

###