You Make the Call:
Industrial Ethernet Solutions EtherCAT and Ethernet Powerlink

Mario Mitchell
Product Manager-Electronics Business Unit
Parker Electromechanical Division- North America

Fieldbusses have become an integrated component of automation technology. They have been tried and tested and are now widely established. It was fieldbus technology that enabled the wide-scale application of PC-based control systems. While the performance of controller CPUs - particularly for IPCs - is increasing rapidly, conventional fieldbus systems tend to represent "bottlenecks" that limit the performance control systems can achieve. An additional factor is the layered control architecture consisting of several subordinate (usually cyclic) systems: the actual control task, the fieldbus system and perhaps local expansion busses within the I/O system or simply the local firmware cycle in the peripheral device. Reaction times are typically 3-5 times higher than the controller cycle time - an unsatisfactory solution (See Figure 1 below).

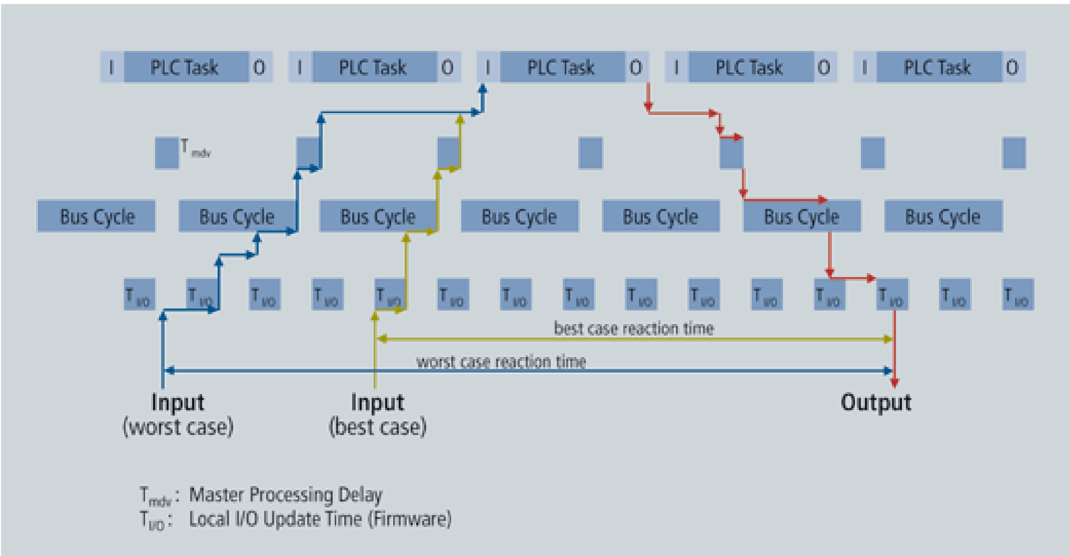

Figure 1: Response times of conventional fieldbus systems

During the last session of the Manufacturing Track of the Automation Forum of 2012, the co-founder and editor-in-chief of *Automation World*, Gary Mintchell, assembled key representatives of the six leading industrial Ethernet protocols. Among those protocols represented were CC-Link, EtherCAT Technology Group, Ethernet/IP, Ethernet Powerlink, and Profinet. Each technology has a strong user base and a compelling reason why it would be preferred in one application over another. A common thread is that Ethernet is the equalizer in the “field-bus wars” over the years. We now see Ethernet in variations such as the protocols listed above. The focus of this paper is to take a look at two of the protocols Parker has employed within our electronics products: Ethernet Powerlink and EtherCAT [1].

EtherCAT: The Need for Speed

EtherCAT technology overcomes the inherent limitations of other Ethernet solutions: the Ethernet packet is no longer received, then interpreted and process data is then copied at every device. The EtherCAT slave devices read the data addressed to them while the frame passes through the node. Similarly, input data is inserted while the telegram passes through (see Fig. 2). The frames are only delayed by a few nanoseconds.

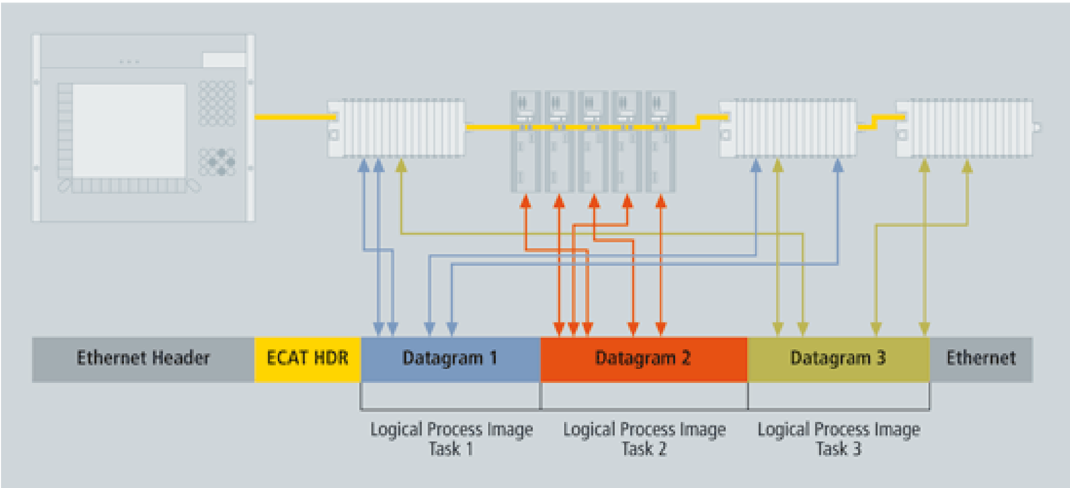

Figure 2: Process data is inserted in telegrams

The EtherCAT protocol is optimized for process data and is transported directly within the Ethernet frame thanks to a special Ethertype. It may consist of several EtherCAT telegrams, each serving a particular memory area of the logical process images that can be up to 4 gigabytes in size. The data sequence is independent of the physical order of the Ethernet terminals in the network; addressing can be in any order. Broadcast, Multicast and communication between slaves are possible. Direct Ethernet frame transfer is used in cases where maximum performance is required and the EtherCAT components are operated in the same subnet as the controller.

However, EtherCAT applications are not limited to single a subnet: EtherCAT User Datagram Protocol (UDP) packages the EtherCAT protocol into UDP/IP datagrams (see Fig. 3). This enables any control with an Ethernet protocol stack to address EtherCAT systems. Even communication across routers into other subnets is possible. In this variant, system performance obviously depends on the real-time characteristics of the control and its Ethernet protocol implementation. The response times of the EtherCAT network itself are hardly restricted at all: the UDP datagram only has to be unpacked in the first station.

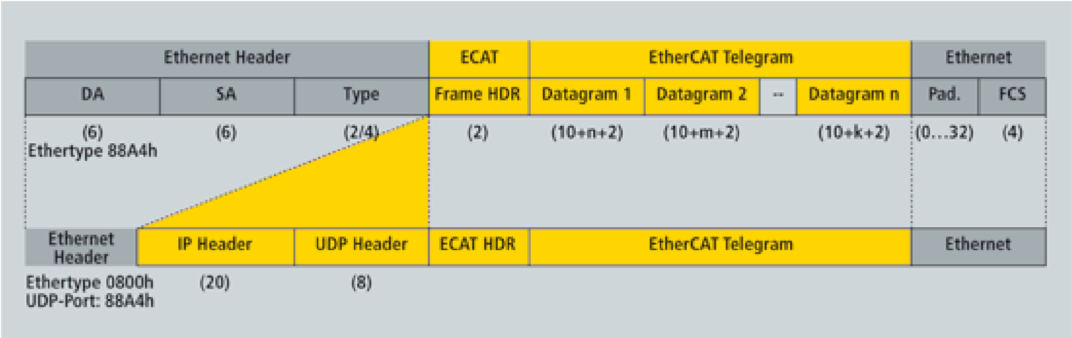

Figure 3: EtherCAT: Standard Frames according to IEEE 802.3

The Topology of EtherCAT

Line, tree or star: EtherCAT supports almost any topology (see Fig. 4). The bus or line structure known from the fieldbusses thus also becomes available for Ethernet, without the quantity limitations implied by cascaded switches or hubs.

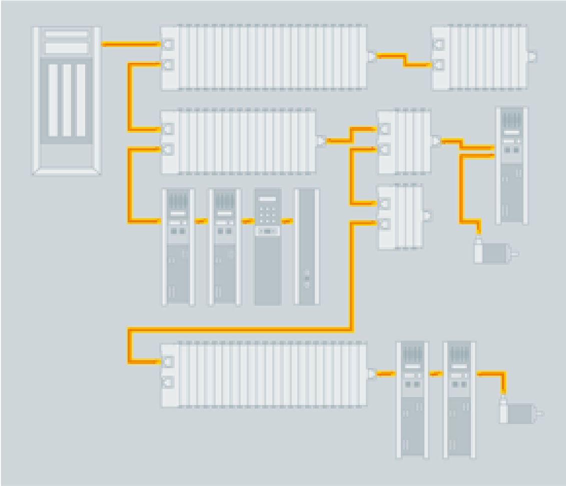

Figure 4: Flexible Topology: Line, Tree or Star

Particularly useful for system wiring is the combination of line and branches or stubs: the required interfaces exist on many devices (e.g. on I/O modules); no additional switches are required. Naturally, the classic switch-based Ethernet star topology can also be used.

EtherCAT is characterized by outstanding performance, very simple wiring and openness for other protocols. EtherCAT sets new standards where conventional fieldbus systems reach their limits: 1000 I/Os in 30 µs, optionally twisted pair cable or optical fiber and, thanks to Ethernet and Internet technologies, optimum vertical integration. With EtherCAT, the costly Ethernet star topology can be replaced with a simple line structure – no expensive infrastructure components are required. Optionally, EtherCAT may also be wired in the classic way, using switches, in order to integrate other Ethernet devices. Where other real-time-Ethernet approaches require special connections in the controller, for EtherCAT, very cost-effective standard Ethernet cards (NIC) are sufficient.

EtherCAT is versatile: Master-to-slave, slave-to-slave and master-to-master communications are supported (see Fig. 5). Safety over EtherCAT is available. EtherCAT makes Ethernet down to the I/O level technically feasible and economically sensible. Full Ethernet compatibility, Internet technologies even in very simple devices, maximum utilization of the large bandwidth offered by Ethernet, outstanding real-time characteristics at low costs are outstanding features of this network [2].

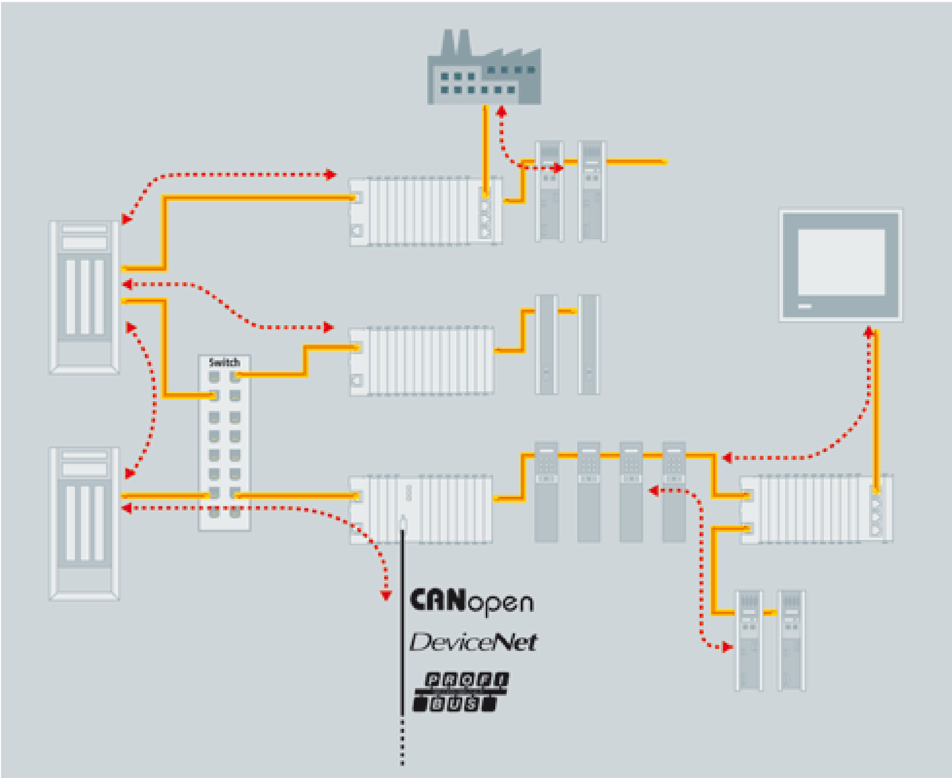

Figure 5: Flexible Topology: Line, Tree or Star

Ethernet Powerlink: Time is of the Essence

Ethernet Powerlink (EPL) is a real-time, open protocol for standard Ethernet. It was introduced by the Austrian automation company B&R in 2001. Ethernet Powerlink was standardized by the Ethernet Powerlink Standardization Group (EPSG), which was founded in 2003 as an independent association. EPL uses a mixed-polling and time-slicing mechanism using isochronic and asynchronous phases:

- It guarantees the transfer of time-critical data in short isochronic cycles with configurable response times
- It employs time-synchronization of all nodes with very high precision (in the sub microsecond time frame). For example, current implementations using EPL can reach a cycle time of under 200 microseconds with a jitter of less than 1 microsecond. The original specification of the EPL physical

layer of the Open Systems Interconnection (OSI) model, was specified at 10Base-X Fast Ethernet. Since the end of 2006, EPL with Gigabit Ethernet has supported a transmission rate ten times higher using better Cat6 cabling. In short, cabling technology has greatly benefitted EPL.

- The transmission of less time-critical data is reserved for the asynchronous phase. As a result of splitting the data delivery into isochronous and asynchronous phases, the standard data link layer of the OSI model is extended.

EPL Basic Cycle

After system start-up is finished, the Real-Time domain is operating under Real-Time conditions. The scheduling of the basic cycle is controlled by the Managing Node (MN). The overall cycle time depends on the amount of isochronous data, asynchronous data and the number of nodes to be polled during each cycle.

The basic cycle consists of the following phases:

- **Start Phase**: The Managing Node is sending out a synchronization message to all nodes. The frame is called SoC - Start of Cycle.
- **Isochronous Phase**: The Managing Node calls each node to transfer time-critical data for process or motion control by sending the Preq - Poll Request - frame. The addressed node answers with the Pres - Poll Response - frame. Since all other nodes are listening to all data during this phase, the communication system provides a producer-consumer relationship.

The time frame that includes Preq-n and Pres-n is called time slot for the addressed node.

- **Asynchronous Phase**: The Managing Node grants the right to one particular node for sending ad-hoc data by sending out the SoA - Start of Asynchronous - frame. The addressed node will answer with ASnd. Standard IP-based protocols and addressing can be used during this phase.

The quality of the Real-Time behavior depends on the precision of the overall basic cycle time. The length of individual phases can vary as long as the total of all phases remain within the basic cycle time boundaries. Adherence to the basic cycle time is monitored by the Managing Node. The duration of the isochronous and the asynchronous phase can be configured.

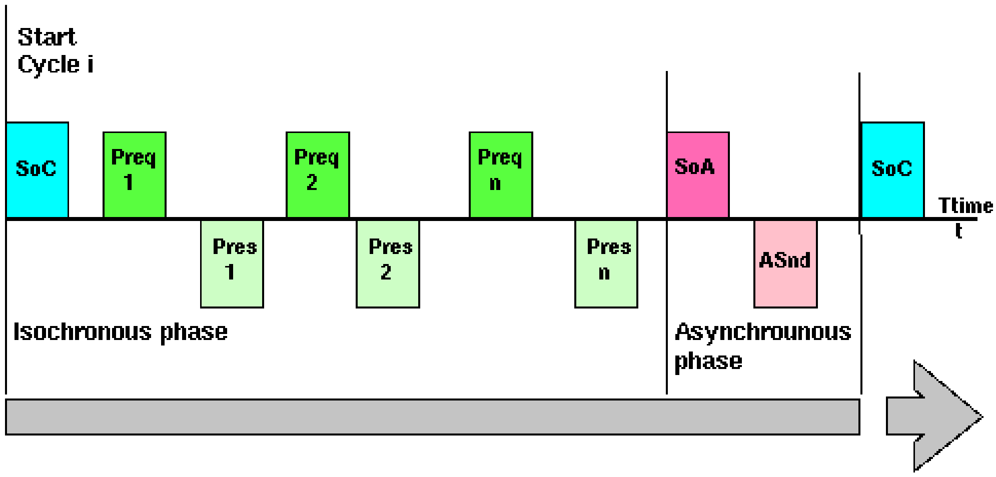

Figure 6: Frames above the time line are sent by the MN, below the time line by different Controlled Nodes (CN).

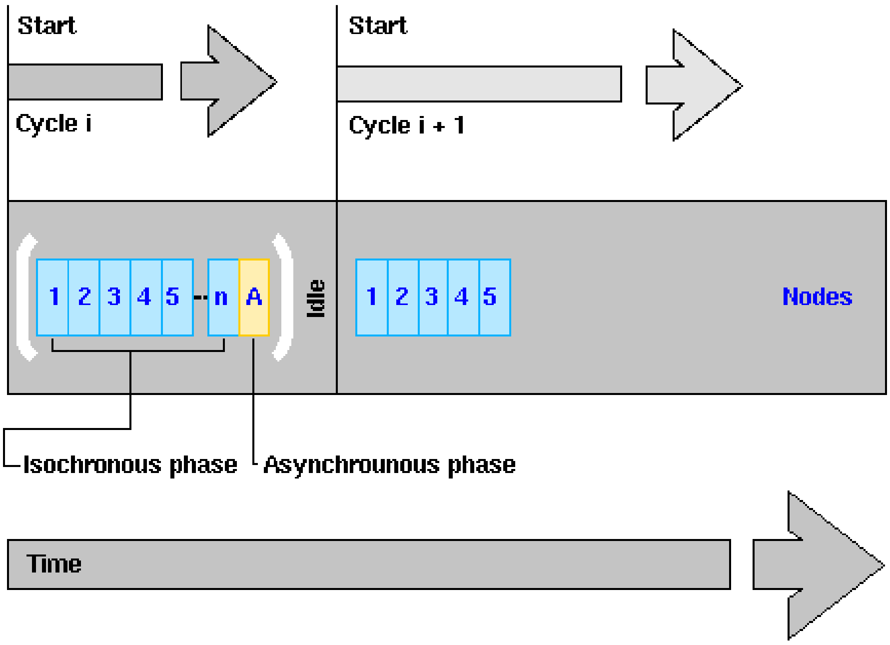

Figure 7: Time slots for nodes and the asynchronous time slot

Multiplex for Bandwidth Optimization

In addition to transferring isochronous data during each basic cycle, some nodes are also able to share transfer slots for better bandwidth utilization. For that reason, the isochronous phase can distinguish between transfer slots dedicated to particular nodes, which have to send their data in every basic cycle, and slots shared by nodes to transfer their data one after the other in different cycles. Therefore, less important yet still time-critical data can be transferred in longer cycles than the basic cycle. Assigning the slots during each cycle is at the discretion of the Managing Node [3].

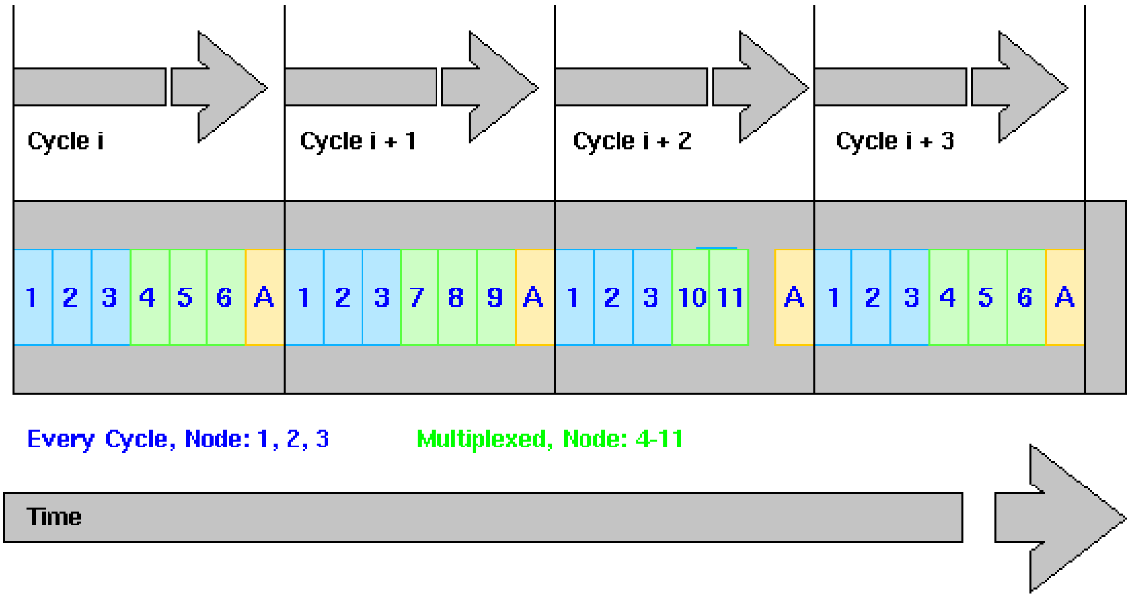

Figure 8: Time slots in EPL multiplexed mode.

Most applications target a cycle time of 1 ms. In some applications, 500 µs or 250 µs are required, but these generally have a reduced number of nodes. The EPL cycle time is determined by the total of all communications on the bus, which is easy to determine by totaling all request/response packets. Since every individual node has to perform specific tasks like SoC processing, asynchronous traffic and other application tasks, there are also node-internal processes like network management and the EPL cycle state machine that must be considered. When considering EPL, it is important to calculate cycle times that take into account all the message events on the EPL bus.

Although the EPL protocol software can be implemented on virtually any platform making use of any standard Ethernet controller, the performance impact of industrial real-time Ethernet protocols like EPL on the application CPU has to be evaluated carefully. Today, it is widely accepted to introduce an exclusive communication resource in addition to the application microcontroller into automation devices in order to provide a dedicated communication performance independent from the application [4].

Summary

In summary, both EtherCAT and EPL are flexible with respect to network topology. EtherCAT certainly allows users to get the required information to the right node at great speeds, while EPL delivers the data needed on time. The industrial Ethernet bus solution is constantly evolving. Whether it’s cabling or other hardware that allows for faster signal processing, the shift is strong toward this type of bus solution. Users must decide through application examples, research, and product availability within their specific market segment which solution would suit their needs.

Parker products are available with a wide variety of communication options, including ETHERNET Powerlink and EtherCAT. For more information, visit www.parkermotion.com.

References

[1] http://www.automationworld.com/information-management/industrial-ethernet-panel- discusses-protocol-differences
[2] http://www.ethercat.org
[3] en.wikipedia.org/wiki/Ethernet_Powerlink
[4] http://www.rtcmagazine.com/articles/view/100821