Moving up to Industrial Ethernet: The EtherCAT protocol

EtherCAT is an Ethernet-based fieldbus system. EtherCAT handling is straightforward and similar to a fieldbus given its flexible topology and simple configuration. Moreover, since EtherCAT can be implemented cost effectively, the system enables use in applications where fieldbus networking was not previously an option.

Fieldbuses have become a tried and tested component of automation technology. It was fieldbus technology that enabled the wide scale application of PC-based control systems. While the performance of controller CPUs – particularly for Industrial PCs – is increasing rapidly, conventional fieldbus systems tend to represent bottlenecks that limit the performance which control systems can achieve.

An additional factor is the layered control architecture comprising 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 three to five times higher than the controller cycle time – an unsatisfactory solution (**Fig. 1**).

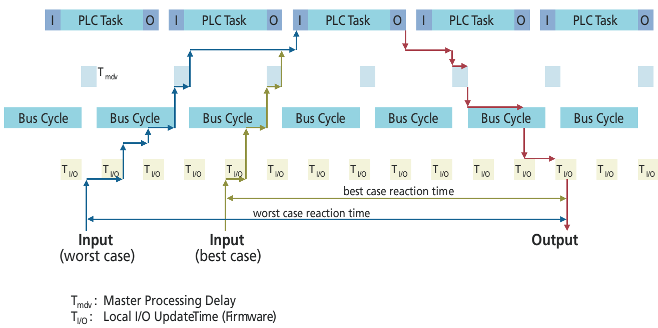

Fig. 1. Response times of conventional fieldbus systems

Above the fieldbus system level, i.e. for networking controllers, Ethernet has already been the de facto network technology for some time. However, its application at the drive or I/O level in areas that were previously dominated by fieldbus systems is relatively new. Real time capability with small data quantities dominates the requirements for this type of application provided that the performance can be delivered with cost effectiveness. EtherCAT meets these requirements and at the same time makes internet technologies available at the I/O level.

Ethernet and real time capability

There are many different approaches that try and provide real time capability for Ethernet. For example, the CSMA/CD media access procedure is disabled via higher level protocol layers and replaced by the time slice procedure or polling. Other propositions use special switches that distribute Ethernet packets in a precisely controlled timely manner. While these solutions may be able to transport data packets quickly and accurately to the connected Ethernet nodes, the times required for the redirection to the outputs or drive controllers, and the time taken to read the input data strongly depend on the implementation.

If individual Ethernet frames are used for each device, the usable data rate is very low in principle: The shortest Ethernet frame is 84 bytes long (including interpacket gap, IPG). This has bandwidth implications. If for example a drive cyclically sends 4 bytes of actual value and status information receiving 4 bytes of command value and control word information in response, a 100% bus load (i.e. with infinitely short response time of the drive) represents a maximum attainable data delivery rate of only 4/84, just 4.8%. At an average response time of 10µs, the rate drops to 1.9%.

These limitations apply to all real time Ethernet approaches that send an Ethernet frame to each device (or expect a frame from each device), irrespective of the protocols used within the Ethernet frame.

EtherCAT technology overcomes these inherent limitations since the Ethernet packet is no longer received, interpreted and process data then copied at every device. The EtherCAT slave devices read the data addressed to them while the frame passes through the node. Input data

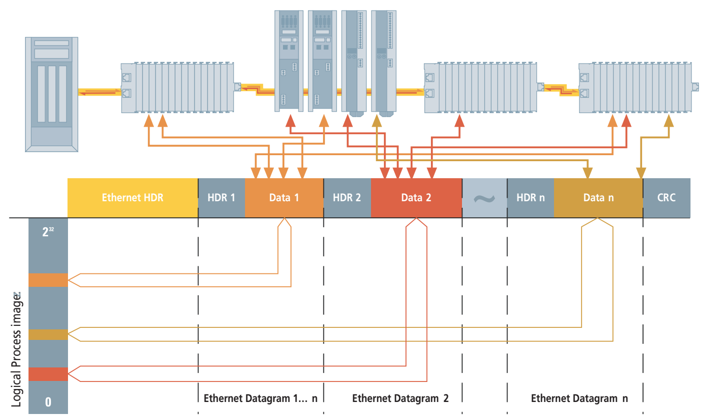

Fig. 2. Insertion of process data into the datagram under EtherCAT

is similarly inserted while the telegram passes through (**Fig. 2**) delaying the frames by only a few nanoseconds. Since an Ethernet frame comprises the data of many devices both in send and receive direction, the usable data rate increases to over 90%.

The full duplex features of 100BaseTX are fully used such that effective data rates of 100Mbps (>90% of 2 x 100Mbps) can be achieved (see **Fig. 3**). The Ethernet protocol according to IEEE 802.3 remains intact right up to the individual device and no sub-bus is required.

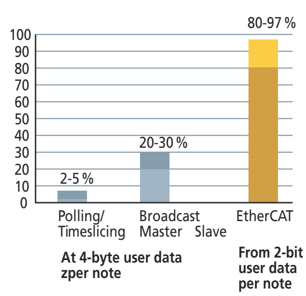

Fig. 3. Comparison of bandwidth utilisation

In order to meet the requirements of a modular device like an electronic terminal block, the physical layer in the coupling device can be converted from twisted pair or optical fibre to LVDS (alternative standard Ethernet physical layer[5]). A modular device can thus be extended efficiently. Subsequent conversion from the backplane physical layer LVDS to the 100Base TX physical layer is possible at any time – as usual with Ethernet.

The EtherCAT protocol

A special Ethertype optimised for process data is transported directly within the Ethernet frame. It may consist of several EtherCAT datagrams, each serving a particular memory area of the logical process images that can be up to four 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 UDP packages the EtherCAT protocol into UDP/IP datagrams (**Fig. 4**). This enables any controller with an Ethernet protocol stack to address EtherCAT systems.

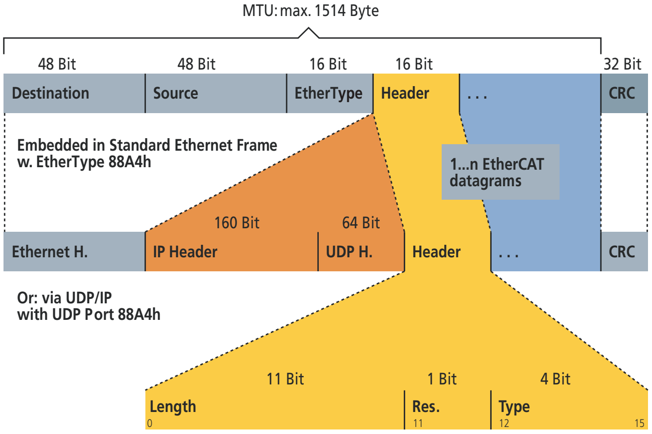

Fig. 4. EtherCATuses standard frames according to IEEE 802.3 [3]

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.

As well as data exchange using the master/ slave principle, EtherCAT is also suitable for communication between controllers (master/master). Freely addressable network variables for process data and a variety of services for parameterisation, diagnosis, programming and remote control cover a range of requirements. The data interface for master/slave and master/master communication are identical. For slave to slave communication, two mechanisms are available. Upstream devices can communicate to downstream devices within the same cycle of allowing extremely fast data exchange. Since this method is dependent on topology, it is particularly suitable for the slave to slave communication relationships frequently occurring in practical machine design, e.g. in printing or packaging applications.

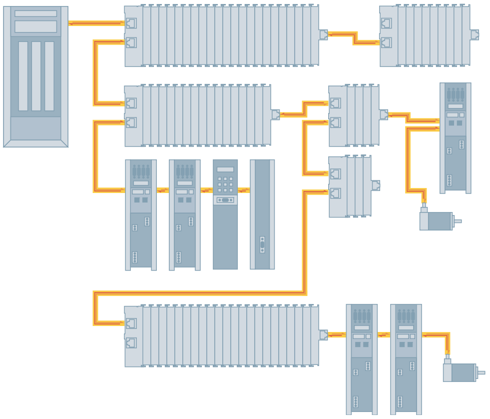

Fig. 5.Topology can be laid out as Line, tree or star

For freely configurable slave to slave communication, the second mechanism applies: the data is relayed by the master. Here two cycles are needed, but due to the inherent performance of EtherCAT, this is still fast.

EtherCAT only uses standard frames[3] – the frames are not shortened. they can thus be sent from any Ethernet MAC, and standard tools (e.g. monitor) can be used.

The protocol can support almost any topology (**Fig. 5**). The bus or line structure of the fieldbus world can be run on Ethernet without the quantitive limitations implied by cascaded switches or hubs. A combination of line and branches or stubs is particularly useful for system wiring. The required interfaces exist on many devices (such as I/O modules) and no additional switches are required. Naturally, the classic switch-based Ethernet star topology can also be used.

Wiring flexibility extends to a choice of different cables. Standard Ethernet cabling would be used while plastic optical fibres (POF) will complement the system for special applications. Different combinations of optical and copper cabling can be used in combination with switches or media converters. The Fast Ethernet physics enables a cable length of up to 100m between two devices. Since a maximum of 65535 devices can be connected, the size of the network is almost unlimited.

Distributed clocks

Accurate synchronisation is important where spatially distributed processes require simultaneous actions. This may be the case, for example, in applications where several servo axes carry out coordinated movements simultaneously. The most powerful approach for synchronisation uses accurately aligned distributed clocks as described in the IEEE1588.[6] In contrast to fully synchronous communication where synchronisation quality suffers immediately in the event of a communication fault, distributed aligned clocks have a high degree of tolerance versus possible fault related delays within the communication system.

With EtherCAT, the data exchange is based entirely on a pure hardware machine. Since the communication uses a logical (and thanks to full duplex Fast Ethernet also physical) ring structure, the master clock can determine the propagation delay offset to the individual slave clocks simply and accurately – and vice versa. The distributed clocks are adjusted based on this value, which means that a very precise network wide timebase with a jitter of significantly less then 1µs is available (**Fig. 6**). External synchronisation, e.g. across the plant, is then based on IEEE 1588. ▶

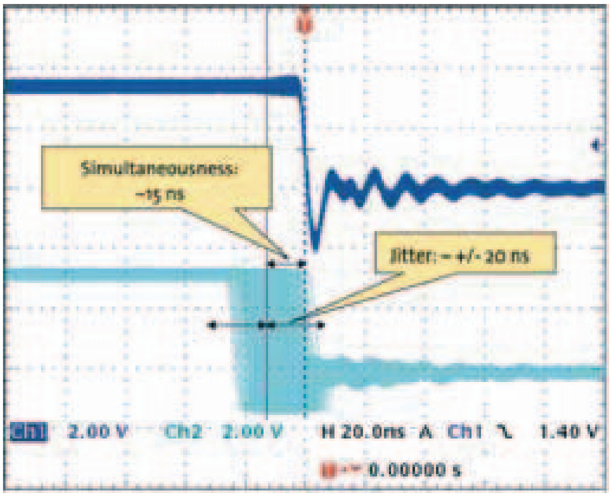

Fig. 6. Synchronism and simultaneous response illustrated in a scope view of two distributed devices with 300 nodes and 120m of cable between them

However, high resolution distributed clocks are not only used for synchronisation. They also provide accurate information about the local timing of the data acquisition. For example, a motion controller typically calculates velocities from sequentially measured positions. Particularly with very short sampling times, even a small temporal jitter in the position measurement leads to large step changes in the computed velocity. With EtherCAT, timestamp data types are introduced as a logical extension. The high resolution system time is linked to the measured value, which is made possible by the large bandwidth offered by Ethernet. The accuracy of a velocity calculation then no longer depends on the jitter of the communication system. It is orders of magnitude better than that of measuring techniques based on jitter free communication.

System performance

EtherCAT depends on hardware integration in the slave, and direct memory access to the network controller in the master. Protocol processing thus takes place completely within hardware and is thus fully independent of the run time of protocol stacks, CPU performance or software implementation. The update time for 1000 I/Os is 30µs – including I/O cycle time (**Table 1**). Up to 1486 bytes of process data can be exchanged with a single Ethernet frame – this is equivalent to almost 12,000 digital inputs and outputs. The transfer time for this data quantity is only 300µs.

Process Data	Update Time
256 distributed digital I/O	11µs = 0.01ms
1000 distributed digital I/O	30µs
200 analogue I/O (16 bit)	50µs (i.e. 20kHz sampling)
100 Servo axes, with 8 Bytes input and output data each	100µs
Fieldbus Master-Gateway (1486 Bytes Input and 1486 Bytes output data)	150µs

Table 1. EtherCAT Performance Overview

Communication with 100 servo axes is also fast: all axes are provided with command values and control data and can report their actual position and status every 100µs. The distributed clock technique enables the axes to be synchronised with a deviation of significantly less than one microsecond. And even at this pace, there is more than sufficient bandwidth for asynchronous communications such as TCP/IP, parameter download or diagnostic data upload. This level of performance enables control concepts that could not be realised with classic fieldbus systems.

The communication technology matches the superior computing capacity of modern Industrial PCs removing the bus system as a bottleneck in the control concept. Distributed I/Os are recorded faster than is possible with most local I/O interfaces. The EtherCAT technology principle is scalable and not bound to the baud rate of 10 MBaud since extension to Gbit Ethernet is also possible.

Diagnostics

Experience with fieldbus systems shows that availability and commissioning time crucially depends on the diagnostic capability. Only faults that are detected quickly and accurately and located unambiguously can be rectified quickly. Therefore, special attention was paid to diagnostic features during development of the protocol. During commissioning, the actual configuration of the nodes (e.g. drives or I/O terminals) should be checked for consistency with the specified configuration. The topology should also match the configuration. Due to the built-in topology recognition down to the individual terminals, this verification can not only take place during system start up, automatic reading in of the network is also possible (configuration up load).

Bit faults occurring during the transfer are detected through evaluation of the CRC checksum. The 32 bit CRC polynomial has a minimum hamming distance of 4. Apart from broken wire detection and localisation, the protocol, physical layer and topology of the EtherCAT system enable individual quality monitoring of each individual transmission segment. The automatic evaluation of the associated error counters enables precise localisation of critical network sections. Gradual or changing sources of error such as EMI influences, defective connectors or cable damage are detected and located, even if they do not yet overstrain the self healing capacity of the network.

Availability

Increasing demands in terms of system availability are catered for with optional cable redundancy that enables devices to be exchanged without having to shut down the network. Adding redundancy is relatively inexpensive: the only additional hardware required is another standard Ethernet port (no special card or interface) in the master device turning the single cable line topology into a ring. Switchover in case of device or cable failure only takes one cycle, so even motion control applications can survive a cable failure without problems.

EtherCAT also supports redundant masters with hot standby functionality. Since the slave controllers immediately return the frame automatically if an interruption is encountered, failure of a device does not necessarily lead to the complete network being shut down. Drag chain applications, for example, can thus be specifically configured as stubs in order to be prepared for cable break.

Safety

Conventionally, safety functions are realised separately from the automation network, either via hardware or using dedicated safety bus systems. Safety over EtherCAT enables safety related communication and control communication on the same network.

The safety protocol is based on the application layer of EtherCAT, without influencing the lower layers. It is certified according to IEC 61508 and meets the requirements of Safety Integrity Level (SIL).[4] The data length is variable, making the protocol equally suitable for safe I/O data and for safe drive technology. There are no restrictions regarding the communication medium or the transfer rate. Like other EtherCAT data, the safety data can be routed without requiring safety routers or gateways. First fully certified products featuring Safety over EtherCAT are already available.

Use with industrial PCs

With increasing miniaturisation of the PC components, the physical size of Industrial PCs is increasingly determined by the number of required slots. The bandwidth of Fast Ethernet, together with the process data width of the EtherCAT communication hardware enables new directions. Classic interfaces that are conventionally located in the IPC are transferred to intelligent EtherCAT interface terminals (**Fig. 7**). Apart from the decentralised I/Os, drives and control units, complex systems such as fieldbus masters, fast serial interfaces, gateways and other communication interfaces can be addressed. Even further Ethernet devices without restriction on protocol variants can be connected ▶

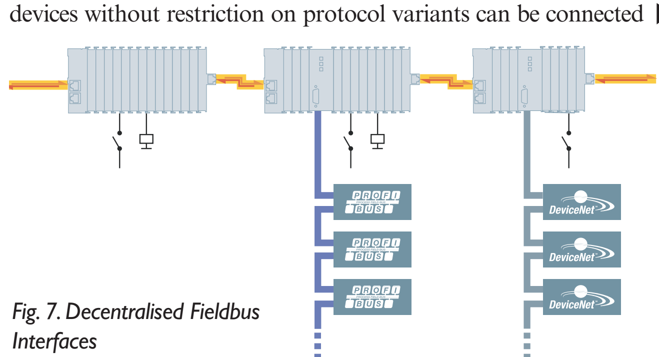

Fig. 7. Decentralised Fieldbus Interfaces

via decentralised switchport devices. The central IPC becomes smaller and therefore more cost-effective. One Ethernet interface is sufficient for the complete communication with the periphery.

Device profiles

The device profiles describe the application parameters and the functional behaviour of the devices including the device class specific state machines. For many device classes, fieldbus technology already offers reliable device profiles, for example for I/O devices, drives or valves. Users are familiar with these profiles and the associated parameters and tools. No EtherCAT-specific device profiles have therefore been developed for these device classes. Instead, simple interfaces for existing device profiles are being offered. This assists users and device manufacturers alike during the migration from the existing fieldbus to EtherCAT.

CANopen device and application profiles are available for a range of device classes and applications, ranging from I/O components, drives, encoders, proportional valves and hydraulic controllers to application profiles for plastic or textile machinery, for example. EtherCAT can provide the same communication mechanisms as the familiar CANopen[7] mechanisms: object dictionary, PDO (process data objects) and SDO (service data objects) – even the network management is comparable. EtherCAT can thus be implemented with minimal effort on devices equipped with CANopen. Large parts of the CANopen firmware can also be reused. Objects may optionally be expanded in order to account for the larger bandwidth available under Ethernet.

Sercos interface is acknowledged and appreciated worldwide as a high performance real time communication interface, particularly for motion control applications. The Sercos profile for servo drives and the communication technology are covered by the IEC 61491 standard[8]. This profile can be readily mapped to EtherCAT. The service channel, and therefore access to all parameters and functions residing in the drive, is based on the EtherCAT mailbox (**Fig. 8**). Here too, the focus is on compatibility with the existing protocol (access to value, attribute, name, units etc. of the IDNs) and expandability with regard to data length limitation.

The process data, with Sercos in the form of AT and MDT data, are transferred using EtherCAT slave controller mechanisms. The mapping is similar to the Sercos mapping. The EtherCAT slave state machine can also be mapped easily to the phases of the Sercos protocol. EtherCAT provides real time Ethernet technology for this device profile, which is particularly widespread in CNC applications – and can draw on the best of both protocols such as precise network wide synchronisation. Optionally, the command position, speed or torque can be transferred. Depending on the implementation, it is even possible to continue using the same configuration tools for the drives.

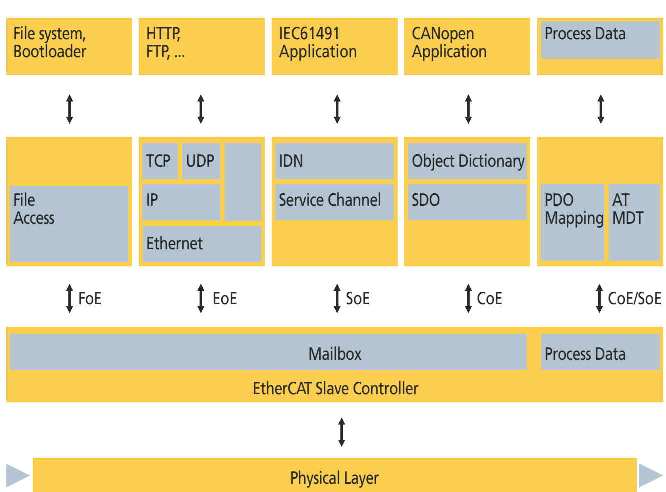

Fig. 8. Several device profiles and protocols can co-exist side by side

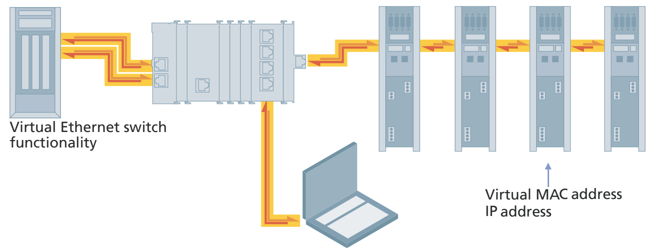

Fig. 9. EtherCAT enjoys transparency with other Ethernet protocols. Ethernet frames are tunnelled via the EtherCAT protocol

Use with standard Ethernet

EtherCAT technology is fully Ethernet compatible. The protocol tolerates other Ethernet-based services and protocols on the same physical network – usually with minimum loss of performance. There is no restriction on the type of Ethernet device that can be connected within the EtherCAT segment via a switch port. The Ethernet frames are tunnelled via the EtherCAT protocol, which is the standard approach for Internet applications (e.g. VPN, PPPoE (DSL) etc.).The EtherCAT network is fully transparent for the Ethernet device, and the real time characteristics are not impaired (**Fig. 9**).

EtherCAT devices can additionally feature other Ethernet protocols and thus act like a standard Ethernet device. The master itself acts like a layer 2 switch that redirects the frames to the respective devices according to the address information. All internet technologies can therefore also be used in the EtherCAT environment: integrated web server, email, FTP transfer etc.

File Access over EtherCAT (FoE) is a simple protocol similar to TFTP which enables access to any data structure in the device. Standardised firmware upload to devices is therefore possible, irrespective of whether or not they support TCP/IP.

Since no hubs and switches are required for EtherCAT, costs associated with these devices including power supply, installation, etc. are avoided. Standard Ethernet cables and standard low cost connectors are used if the environmental conditions permit this. For environment requiring increased protection sealed connectors according to IEC standards are specified.

Summary

EtherCAT is characterised by outstanding performance, simple wiring and openness for other protocols. It sets new standards where conventional fieldbus systems reach their limits: 1000 I/Os in 30µs, optionally twisted pair cable or optical fibre and, thanks to Ethernet and Internet technologies, optimum vertical integration. Ethernet star topology can be replaced with a simple line structure. Optionally, EtherCAT may also be wired in the classic way using switches, in order to integrate other Ethernet devices.

References

3. IEEE 802.3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications.
4. IEEE 802.3ae 2002: CSMA/CD Access Method and Physical Layer Specifications: Media Access Control (MAC) Parameters, Physical Layers, and Management Parameters for 10Gbps Operation.
5. ANSI/TIA/EIA 644 A, Electrical Characteristics of Low Voltage Differential Signalling (LVDS) Interface Circuits
6. IEEE 1588 2002: IEEE Standard for a Precision Clock Synchronisation Protocol for Networked Measurement and Control Systems
7. EN 50325 4: Industrial communications subsystem based on ISO 11898 (CAN) for controller device interfaces. Part 4: CANopen.
8. IEC 61491: Electrical equipment of industrial machines – Serial data link for real time communication between controls and drives

EtherCAT Technology Group, http://www.ethercat.org

EtherCAT FAQs – *from the EtherCAT Technology Group*

EtherCAT is faster than my application requirements. Why should I use it?

Superior fieldbus performance never harms. Even with slow controls, it improves reaction times and reduces configuration effort, since default settings will do the job. If you do not care so much about performance though, use EtherCAT for its other benefits: e.g. lower costs, more flexible topology or simply ease of use.

Why does EtherCAT provide cost advantages?

For several reasons: Inexpensive slave controllers lead to lower slave device costs. No special master card required, the on-board Ethernet controller is sufficient. No switches or hubs required, therefore lower infrastructure costs. Use of standard cabling. Simple to implement, therefore lower implementation costs. Auto-configuration is supported, no manual address setting required, therefore lower configuration costs.

Is EtherCAT limited to Master/Slave Applications?

No. Like with every real time Industrial Ethernet system, one device (the master) has to be in charge of the network management and organise the Medium Access Control. With EtherCAT, slave-to-slave communication is supported in two ways: topology dependent within one communication cycle, topology independent within two cycles. Since EtherCAT is faster than competing systems, slave-to-slave communication using two cycles is also faster.

EtherCAT specifies two different physical layers – why?

EtherCAT uses standard 100BASE-TX on standard CAT5+ cables. Since EtherCAT is also used as 'backplane bus' for modular devices, an even lower cost physical layer from IEEE802.3ae was added for such applications: LVDS (also called: E-Bus). Outside such modular devices, the physical layer is changed back to 100Base-TX.

Do I have to be an ETG member to use EtherCAT?

No. However, you may want to consider joining the ETG in order to indicate your interest in and support for this technology to your suppliers and customers. As an ETG member, you are invited to attend the ETG meetings, you get access to technology information, draft specs and can influence the direction in which the technology moves.

EtherCAT is an open technology. What does this mean?

This means that everybody may use, implement and benefit from this technology. This also means that EtherCAT implementations have to be compatible, and nobody may change the technology in a way that prevents others to use it. EtherCAT already is a publicly available specification, published by IEC (IFC/PAS 62407), and is on its way to be integrated into the international fieldbus standard IEC 61508.

How about licenses?

The license for implementing an EtherCAT master is free of charge. For slave devices EtherCAT has adopted the CAN license model (CAN is an excellent example for a standardized open technology that is protected by patents): The small license fee is 'embedded' in the Slave Controller Chip, so that device manufacturers, end users, system integrators, tool manufacturers etc. do not have to pay a license.

Will Asics replace the FPGAs?

No. FPGAs are a versatile, manufacturer independent and cost effective solution – not only when additional device functionalities are integrated. An FPGA based EtherCAT interface already undercuts legacy fieldbus interface costs – and prices are expected to drop even further. Therefore further FPGA implementations are on the roadmap.