

How to Select the Right Industrial Ethernet Standard: EtherCAT



Thomas Mauer

Welcome back to the second installment of “[How to select the right industrial Ethernet standard](#),” a blog series for designers who have decided to leverage industrial Ethernet in their systems. This series will cover several common industrial Ethernet communication protocols to help you select the best standard for your application. You can read the previous post, about [Sercos III](#), [here](#).

In this installment, I’ll take a close look at EtherCAT, its end applications and how it works.

Beckhoff Automation invented EtherCAT. Since 2003, it has been under the umbrella of the EtherCAT Technology Group (ETG), an industrial fieldbus organization with around 2,600 member companies.

EtherCAT can be found in [factory automation](#), semiconductor tools, packaging robotics and more. One of my favorite nonindustrial examples is the World of Color show in Disneyland, where EtherCAT controls the light show.

On the technical side, EtherCAT is a master-and-slave network architecture as shown in [Figure 1](#). The EtherCAT slave has two Ethernet connectors in order to support a simple line topology. It is also possible to support a ring topology for redundancy, in which case the EtherCAT master also needs two Ethernet ports. An EtherCAT network can support up to 65,535 slave devices.

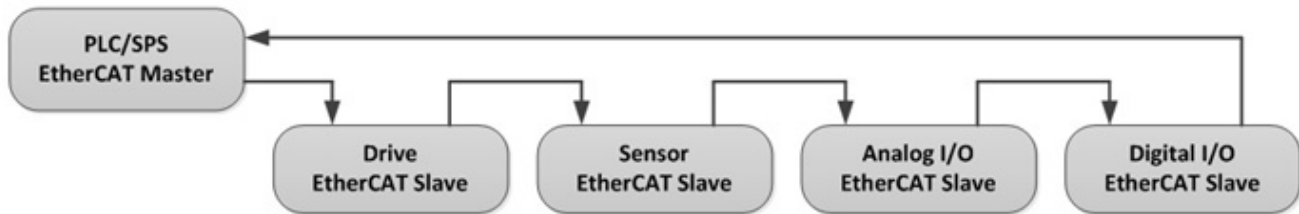


Figure 1. EtherCAT Master-and-slave Example Network with EtherCAT Frame Flow

Only the EtherCAT master generates an EtherCAT frame; all slaves receive and process this frame. The last slave loops back the EtherCAT frame, and it travels back through all slaves (without further processing) to the master. The EtherCAT frame has reserved space for process data for each slave, and none of the slaves change the frame length itself. It is a bit like a train with multiple cars and seats for passengers, where each slave has an assigned car or an assigned seat. EtherCAT allows addressing down to the “seat” level – the bit level in process data.

An EtherCAT master requires a standard Ethernet MAC peripheral only as the master has no specific need to process the EtherCAT frame in real-time. The master only needs to trigger the start of an EtherCAT frame with an accurate time base.

An EtherCAT slave requires specific Ethernet hardware (EtherCAT MAC) support – typical implementations use Application Specific Integrated Circuits (ASIC) or Field Programmable Gate Arrays (FPGA) as seen in [Figure 2](#) – as it needs to process incoming EtherCAT frames on the fly. This means that EtherCAT MAC processes the EtherCAT frame while frame reception is ongoing. Typical EtherCAT devices have a 1µs port-to-port delay between received and transmitted frames.



Figure 2. EtherCAT Slave with an ASIC/FPGA and External Processor

The EtherCAT standard does not allow other types of Ethernet frames in an EtherCAT network. This is mainly to keep the real-time Ethernet channel clear for EtherCAT frames. In case a non-EtherCAT frame (an EtherType not matching with EtherCAT) is in the EtherCAT network, the EtherCAT slaves will destroy the frame by invalidating the frame checksum (CRC). This makes the none-EtherCAT frame useless for standard Ethernet devices, and they will sooner or later stop “polluting” the EtherCAT network.

It is possible to transmit standard Ethernet frames (Transmission Control Protocol/Internet Protocol (TCP/IP), User Datagram Protocol (UDP)) over an EtherCAT network by encapsulating the Ethernet frame into an EtherCAT frame. This is done through various EtherCAT profiles that sit on top of the EtherCAT host-processor stack. An example profile is the Ethernet over EtherCAT profile.

Other features and functions of EtherCAT devices include:

- Distributed clocks – a precise time-synchronization method across the slaves and the master.
- Fast link-drop detection with loopback (requiring Ethernet PHY transceiver support) – when the Ethernet PHY, such as the DP83826E, detects a link down, it notifies the EtherCAT hardware. Within less than 10µs, the EtherCAT hardware will loop back the EtherCAT frames.

Using an ASIC or FPGA for EtherCAT adds cost and board space. An alternative solution is to use the PRU-ICSS peripheral in many TI Sitara processors. For example, TI provides PRU-ICSS firmware for the [AM335x processor](#), enabling an EtherCAT slave device without the use of an external FPGA (Figure 3). You can evaluate TI’s EtherCAT slave solution with the [TMDSICE3359 evaluation board](#).

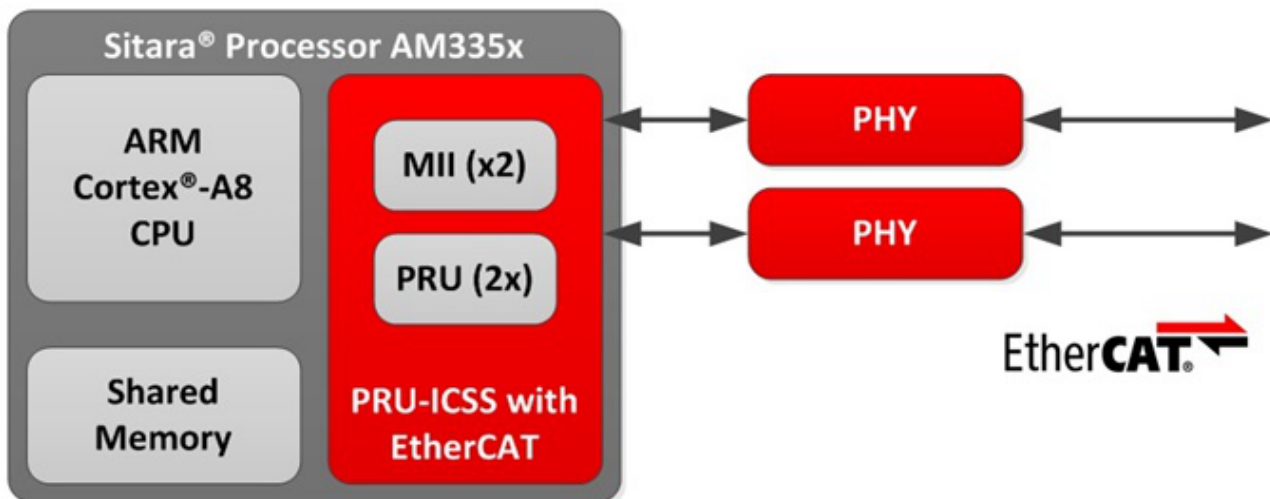


Figure 3. EtherCAT Slave Integrated with a TI Sitara Processor

I hope this post has shed some light into EtherCAT’s functionality. Make sure to read additional posts in this series to learn more about the many industrial Ethernet standards.

Additional Resources

- View the [EtherCAT Slave and Multi-Protocol Industrial Ethernet Reference Design](#)
- Read the previous post in this series, “[How to select the right industrial Ethernet standard: Sercos III.](#)”
- Read a blog post on “[How to make industrial Ethernet as simple as a standard Ethernet.](#)”

- Read the white paper, “[EtherCAT on Sitara Processors](#),” which provides an introduction to EtherCAT.
- Take [an inside look at industrial Ethernet communication protocols](#) in this white paper.
- Learn more about the EtherCAT standard on the [ETG website](#).

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](https://www.ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2023, Texas Instruments Incorporated