SubDevice Implementation Guide

Document: ETG.2200 G D V3.2.0

SECTION I – EtherCAT SubDevice introduction and implementation procedure

SECTION II – ESC overview and EtherCAT development products

SECTION III – EtherCAT P introduction and implementation

SECTION IV – Safety over EtherCAT introduction and implementation

Created by:	EtherCAT Technology Group
Contact:	techinfo@ethercat.org
Date:	05.02.2025
Version	V3.2.0

EtherCAT Technology Group

LEGAL NOTICE

Trademarks and Patents

EtherCAT®, Safety over EtherCAT® and EtherCAT P® are registered trademark and patented technology, licensed by Beckhoff Automation GmbH & Co. KG, Germany. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Disclaimer

The documentation has been prepared with care. The technology described is, however, constantly under development. For that reason, the documentation is not in every case checked for consistency with performance data, standards or other characteristics. In the event that it contains technical or editorial errors, we retain the right to make alterations at any time and without warning. No claims for the modification of products that have already been supplied may be made based on the data, diagrams and descriptions in this documentation.

Copyright

EtherCAT® Technology Group

DOCUMENT ORGANIZATION

This document provides help for implementing an EtherCAT© SubDevice from a generic and a practical point of view. It answers the following questions:

- How is the EtherCAT SubDevice architecture?
- What steps are helpful to implement an EtherCAT SubDevice?
- Which documents are available?
- What kinds of EtherCAT development components are available? What are the differences?
- Is EtherCAT training and implementation support available?
- Why attend an EtherCAT Plug Fest?
- How to obtain conformance for EtherCAT devices?

There are many possibilities for how an EtherCAT SubDevice implementation can be realized. However, the way described in this document has proved to lead to a fast EtherCAT SubDevice implementation. The document is organized in four sections:

SECTION I – EtherCAT SubDevice introduction and implementation procedure provides principal aspects of an EtherCAT SubDevice implementation and provides further information including a list of useful tools and available trainings.

SECTION II – ESC overview and EtherCAT development products provides device specific descriptions for further implementation aspects and an overview of available evaluation boards and EtherCAT SubDevice Controllers (ESCs).

SECTION III – EtherCAT P introduction and implementation provides implementation topics as well as testing conditions for the EtherCAT enhancement “EtherCAT P”.

SECTION IV – Safety over EtherCAT introduction and implementation provides detailed information about implementing Safety over EtherCAT, references to related Safety over EtherCAT specifications and documents, as well as licensing and conformance testing.

LEGAL NOTICE 0-2
DOCUMENT ORGANIZATION 0-3
ABBREVIATIONS 0-7

SECTION I – EtherCAT SubDevice introduction and implementation procedure I-11
1 Introduction I-6
1.1 Documents for detailed Information and further reading I-6
2 EtherCAT system architecture I-8
2.1 Configuration tool I-8
2.2 MainDevice I-8
2.3 SubDevice I-8
3 EtherCAT technology overview I-12
3.1 General I-12
3.2 Frame processing order I-12
3.3 SubDevice Information Interface (SII) I-13
3.4 Fieldbus Memory Management Unit (FMMU) I-14
3.5 SyncManager (SM) I-15
3.6 Distributed Clocks (DC) I-16
3.7 Data structure and communication protocols I-16
3.8 EtherCAT State Machine I-17
4 EtherCAT SubDevices implementation aspects I-20
4.1 General procedure – step by step I-20
4.2 General procedure – step by step I-20
4.3 Administrative organization I-21
4.3.1 Development time I-21
4.3.2 ETG membership and Vendor ID I-21
4.3.3 EtherCAT Conformance Test Tool license I-21
4.4 EtherCAT SubDevice design I-22
4.4.1 Bus interface to EtherCAT network I-22
4.4.2 EtherCAT SubDevice Controller (ESC) and PDI I-22
4.4.3 SII I-24
4.4.4 Application controller (host controller, µC) I-24
4.4.5 Application Layer communication protocols I-24
4.4.6 Device profiles I-27
4.4.7 Synchronization among SubDevices and the MainDevice I-29
4.4.8 Firmware update I-30
4.5 Tools for EtherCAT SubDevice development I-30
4.5.1 XML editor for generating ESI files I-31
4.5.2 EtherCAT network configurator and MainDevice software I-32
4.5.3 Monitoring communication and network diagnosis using Wireshark I-33
4.5.4 EtherCAT Conformance Test Tool I-34
4.6 EtherCAT product labels and LEDs I-35
5 EtherCAT Conformance Testing I-37
5.1 EtherCAT Conformance Test Tool I-37
5.2 Official EtherCAT Conformance Test at an EtherCAT Test Center (ETC) I-37

6 EtherCAT Development Support I-38
6.1 EtherCAT training and workshop I-38
6.2 EtherCAT development support tips I-39
7 EtherCAT Technology Group – events and support I-40
7.1 Basic information about the ETG I-40
7.2 EtherCAT Plug Fests I-41
7.3 Official EtherCAT Conformance Test certificate I-41
7.4 ETG Technical Committee I-42
7.5 Information and support I-42
7.5.1 EtherCAT Compendium I-42
7.5.2 Download area on the web site I-42
7.5.3 Knowledge Base I-42
7.5.4 Developers Forum I-43
7.5.5 Search the EtherCAT web site I-43
7.5.6 Technical support I-43

SECTION II – ESC overview and EtherCAT development products II-44
1 Introduction II-5
2 EtherCAT SubDevice evaluation boards II-6
3 EtherCAT SubDevice communication modules II-9
3.1 EtherCAT communication modules overview II-9
3.2 Beckhoff FB1111 ..... II-10
3.3 Hilscher comX II-11
4 EtherCAT SubDevice Controller overview II-12
5 Missing your device? II-16

SECTION III – EtherCAT P introduction and implementation III-1
1 EtherCAT P introduction III-5
2 EtherCAT P technology III-6
2.1 EtherCAT P connectors and cables III-6
2.2 PHY selection III-7
2.3 EtherCAT P use cases III-8
2.4 EtherCAT P device structure III-8
2.4.1 EtherCAT P in the ESI file III-10
2.4.2 EtherCAT P device types III-11
2.4.3 EtherCAT P device categories III-11
3 EtherCAT P specification and documents III-13
4 EtherCAT P conformance testing III-15
4.1 General III-15
4.2 Evaluate your current EtherCAT SubDevice for EtherCAT P III-15
4.3 Contact for EtherCAT P conformance testing III-15
5 EtherCAT P licensing III-16
5.1 General III-16
5.2 License agreement III-16
6 EtherCAT P implementation aspects III-17

6.1 EtherCAT P and EtherCAT configuration tool III-17
7 EtherCAT P development support III-19
7.1 EtherCAT and EtherCAT P training III-19
7.2 Technical support III-19

SECTION IV – Safety over EtherCAT introduction and implementation IV-20
1 Introduction IV-24
2 Safety over EtherCAT technology IV-25
2.1 Overview IV-25
2.2 Documents for detailed information and further reading IV-26
3 Technology users IV-28
3.1.1 Machine builders IV-28
3.1.2 Standard EtherCAT MainDevice manufacturer IV-28
3.1.3 FSoE device manufacturer IV-29
4 Safety over EtherCAT implementation aspects IV-30
4.1 FSoE device structure IV-30
4.2 Hardware architecture IV-30
4.3 Software architecture IV-30
4.4 Safety manual IV-31
5 Safety over EtherCAT Licensing IV-32
5.1 General IV-32
5.2 License Agreement IV-32
6 Safety over EtherCAT conformance testing IV-33
6.1 FSoE Test Cases IV-33
6.2 FSoE Conformance Test Tool for FSoE devices IV-33
6.3 FSoE Conformance Test IV-34
6.3.1 FSoE SubInstance Conformance Test IV-34
6.3.2 FSoE MainInstance Conformance Test IV-35
7 Safety over EtherCAT development support IV-37
7.1 EtherCAT and Safety over EtherCAT training IV-37
7.2 Technical support IV-37
7.3 Step by step implementation for an FSoE device manufacturer IV-37
8 Frequently asked questions IV-38
DOCUMENT HISTORY IV-40

ABBREVIATIONS

Abbreviations	Description
µC / MCU	A **m**icro**c**ontroller (MCU for microcontroller unit) is a small computer on a single integrated circuit. A microcontroller contains one or more CPUs (processor cores) along with memory and programmable input/output peripherals. Program memory in the form of ferroelectric RAM, NOR flash or OTP ROM is also often included on chip, as well as a small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications consisting of various discrete chips. (www.wikipedia.org)
ADS	The **A**utomation **D**evice **S**pecification describes a device- and fieldbus-independent interface. This interface got designed by Beckhoff, and is - including the protocol - in detail documented. The ADS components get installed together with TwinCAT 3. For integration into own applications and tools there are ADS components (C/C++, .NET) available from Beckhoff. (Beckhoff Information System)
AL	The **A**pplication **L**ayer describes the highest layer of the EtherCAT SubDevice stack which includes the EtherCAT State Machine, error handling, mailbox protocol handling, SubDevice application
AoE	**A**DS **o**ver **E**therCAT (AoE) is a standard, client-server mailbox application protocol defined by the EtherCAT specification.
API	In computer programming, an **A**plication **P**rogramming **I**nterface is a set of subroutine definitions, communication protocols, and tools for building software. In general terms, it is a set of clearly defined methods of communication among various components. (www.wikipedia.org)
ASIC	An **A**pplication-**S**pecific **I**ntegrated **C**ircuit is an integrated circuit (IC) customized for a particular use, rather than intended for general-purpose use. (www.wikipedia.org)
CAN	A **C**ontroller **A**rea **N**etwork (CAN bus) is a robust vehicle bus standard designed to allow microcontrollers and devices to communicate with each other in applications without a host computer. It is a message-based protocol, designed originally for multiplex electrical wiring within automobiles to save on copper, but is also be used in many other contexts. (www.wikipedia.org)
CiA	**C**AN **i**n **A**utomation is the international users' and manufacturers' organization that develops and supports CAN-based higher-layer protocols. (www.wikipedia.org)
CoE	With the **C**AN® application protocol **o**ver **E**therCAT protocol, EtherCAT provides the same communication mechanisms as in CANopen®-Standard EN 50325-4: Object Dictionary, PDO Mapping (Process Data Objects) and SDO (Service Data Objects) – even the network management is similar. This makes it possible to implement EtherCAT with minimal effort in devices that were previously outfitted with CANopen, and large portions of the CANopen Firmware are even reusable. (www.ethercat.org)
CPU	A **C**entral **P**rocessing **U**nit, also called a central processor or main processor, is the electronic circuitry within a computer that carries out the instructions of a computer program by performing the basic arithmetic, logic, controlling, and input/output (I/O) operations specified by the instructions. (www.wikipedia.org)
DC	Distributed Clocks
DLL	The **D**ata **L**ink **L**ayer is the second layer of the seven-layer OSI model of computer networking. This layer is the protocol layer that transfers data between adjacent network nodes in a wide area network (WAN) or between nodes on the same local area network (LAN) segment. The data link layer provides the functional and procedural means to transfer data between network entities and might provide the means to detect and possibly correct errors that may occur in the physical layer. (www.wikipedia.org)
DPRAM	Dual Ported Random Access Memory
DuT	A **D**evice **u**nder **T**est is a manufactured product undergoing testing, either at first manufacture or later during its life cycle as part of ongoing functional testing and calibration checks. This can include a test after repair to establish that the product is performing in accordance with the original product specification.
EEPROM	An **E**lectrically **E**rasable **P**rogrammable **R**ead-Only **M**emory is a type of non-volatile memory used in computers, integrated in microcontrollers for smart cards and remote keyless systems, and other electronic devices to store relatively small amounts of data but allowing individual bytes to be erased and reprogrammed. (www.wikipedia.de)
ENI	The **E**therCAT **N**etwork **I**nformation represents the standardized, XML-based description of an EtherCAT network. It provides a manufacturer-independent way for configuration tools to generate and provide the network configuration to MainDevices. (www.ethercat.org)
EoE	**E**thernet **o**ver **E**therCAT allows one to use a Standard Ethernet device like a printer, camera or PC within an EtherCAT network. There is no restriction on the type of Ethernet device that can be connected. The frames are tunneled by the EtherCAT MainDevice via the EtherCAT protocol. The EtherCAT networks is fully transparent for the Ethernet device, and the real-time characteristics are not impaired. (www.ethercat.org)

Abbreviations	Description
EPU	The **E**therCAT **P**rocessing **U**nit is the logical core of an EtherCAT SubDevice Controller. It contains registers, memories and data processing elements. A frame always comes from port A before passing through the EtherCAT Processing Unit. It receives, analyzes and processes the EtherCAT data stream. (Beckhoff Infosys)
ESC	The **E**therCAT **S**ubDevice **C**ontroller processes the EtherCAT frames on the fly in hardware. It's implementation can be as an ASIC device, as IP Core for FPGAs, as system on a chip (SoC) or integrated as native EtherCAT interface on an microcontroller or CPU. There is a long list of ESCs of different types and vendors. (Knowledge Base)
ESI	The **E**therCAT **S**ubDevice **I**nformation file is a XML based file that comes with an EtherCAT SubDevice and contains the complete description of its network accessible properties, such as manufacturer and product information, Process Data, their mapping options, supported Mailbox application protocols including optional features, as well as the supported modes of Synchronization. (www.ethercat.org)
ESM	The state of the EtherCAT SubDevice is controlled via the **E**therCAT **S**tate **M**achine. Depending upon the state, different functions are accessible or executable in the EtherCAT SubDevice. Specific commands must be sent by the EtherCAT MainDevice to the device in each state, particularly during the bootup of the SubDevice. A distinction is made between the following states: init, pre-operational, safe-operational and operational, boot. The regular state of each EtherCAT SubDevice after bootup is the operational state (Beckhoff Infosys/Anpassung: CH)
ETC	The official **E**therCAT **T**est **C**enter in Eurodpe, Asia and North America are accredited by the ETG and perform the official EtherCAT Conformance Test. (ETG Brochure 11/2018)
ETG	The **E**therCAT **T**echnology **G**roup is a global organization in which OEM, End Users and Technology Providers join forces to support and promote the further technology development. (www.ethercat.org)
EtherCAT	The **Ether**net for **C**ontrol **A**utomation **T**echnology is an Ethernet-based fieldbus system, invented by Beckhoff Automation. The protocol is standardized in IEC 61158 and is suitable for both hard and soft real-time computing requirements in automation technology. (www.wikipedia.org)
EtherCAT Device	Device using EtherCAT communication
FCS	A **F**rame **C**heck **S**equence refers to an error-detecting code added to a frame in a communications protocol. (www.wikipedia.org)
FMMU	The **F**ieldbus **M**emory **M**anagement **U**nit belongs to the DLL and can be found in each I/O terminal. FMMUs are used to map logical addresses bitwise or bytewise to physical addresses of the EtherCAT **S**ubDevice Controller. (Beckhoff Infosys)
FoE	**F**ile **A**ccess over **E**therCAT is a mailbox application protocol generally intended to transfer file data on an EtherCAT network in both directions, and as such it can be used in any state where the mailbox communication is active (PREOP, SAFEOP, OP). (www.ethercat.org)
FPGA	A **F**eld-**P**rogrammable **G**ate **A**rray is an integrated circuit designed to be configured by a customer or a designer after manufacturing. (www.wikipedia.org)
FSoE	EtherCAT utilizes the protocol Safety over EtherCAT (**F**ail**S**afe **o**ver **E**therCAT) to transfer safety-critical control data through the same medium as the control data themselves. (www.ethercat.org)
FSoE Device	Device using EtherCAT communication with FSoE feature
GPIO	A **G**eneral-**P**urpose **I**nput/**O**utput is an uncommitted digital signal pin on an integrated circuit or electronic circuit board whose behavior—including whether it acts as input or output—is controllable by the user at run time. (www.wikipedia.org)
HAL	A **H**ardware **A**bstraction **L**ayer is an abstraction layer, implemented in software, between the physical hardware of a computer and the software that runs on that computer. Its function is to hide differences in hardware from most of the operating system kernel, so that most of the kernel-mode code does not need to be changed to run on systems with different hardware. (www.wikipedia.org) r
I^2C	**I**nter-**I**ntegrated **C**ircuit is a synchronous, multi-master, multi-slave, packet switched, single-ended, serial computer bus invented in 1982 by Philips Semiconductor (now NXP Semiconductors). It is widely used for attaching lower-speed peripheral ICs to processors and microcontrollers in short-distance, intra-board communication. (www.wikipedia.org)
IEC	The **I**nterglobal **E**lectrotechnical **C**ommission is a Swiss association that acts as an interglobal standards organization that prepares and publishes interglobal standards for all electrical, electronic and related technologies – collectively known as "electrotechnology". (www.wikipedia.org)
ISO	The **I**nternational **O**rganization for **S**tandardization is an international standard-setting body composed of representatives from various national standards organizations. (www.wikipedia.org)
LED	A **L**ight-**E**mitting **D**iode is a semiconductor light source that emits light when current flows through it. (www.wikipedia.org)

Abbreviations	Description
LVDS	**L**ow-**V**oltage **D**ifferential **S**ignaling, also known as TIA/EIA-644, is a technical standard that specifies electrical characteristics of a differential, serial communication protocol. LVDS operates at low power and can run at very high speeds using inexpensive twisted-pair copper cables. LVDS is a physical layer specification only; many data communication standards and applications use it and add a data link layer as defined in the OSI model on top of it. (www.wikipedia.org)
MCI	Micro Controller Interface
MDP	The **M**odular **D**evice **P**rofile defines a modeling of structures within in a device. Mainly the object dictionary structure and corresponding behavior of the entries is defined by the MDP. The intention is to provide an easy way for MainDevice and configuration tools to handle the devices. (Knowledge Base)
MII	The **M**edia-**I**ndependent **I**nterface was originally defined as a standard interface to connect a Fast Ethernet (i.e., 100 Mbit/s) media access control (MAC) block to a PHY chip. The MII is standardized by IEEE 802.3u and connects different types of PHYs to MACs. Being media independent means that different types of PHY devices for connecting to different media (i.e. twisted pair, fiber optic, etc.) can be used without redesigning or replacing the MAC hardware. Thus any MAC may be used with any PHY, independent of the network signal transmission media. (www.wikipedia.org)
NIC	A **N**etwork **I**nterface **c**ard (also known as a network interface controller, network adapter, LAN adapter or physical network interface) is a computer hardware component that connects a computer to a computer network. (www.wikipedia.org)
NW	NetWork
OEM	An **O**riginal **E**quipment **M**anufacturer is a company that produces parts and equipment that may be marketed by another manufacturer. (www.wikipedia.org)
PD	Power Device
PDI	The **P**hysical **D**evice **I**nterface is an interface that allows access to the ESC from the process side. (Beckhoff Infosys)
PDO	The **P**rocess **D**ata **O**bject protocol is used to process real time data among various nodes. (www.wikipedia.de)
PELV	IEC 61140 defines a **P**rotective **E**xtra-**L**ow **V**oltage system as "an electrical system in which the voltage cannot exceed ELV under normal conditions, and under single-fault conditions, except earth faults in other circuits". (www.wikipedia.org)
PhL	**Ph**ysical **L**ayer
PIC	Programmable Integrated Circuit
PLC	A **P**rogrammable **L**ogic **C**ontroller or programmable controller is an industrial digital computer which has been ruggedized and adapted for the control of manufacturing processes, such as assembly lines, or robotic devices, or any activity that requires high reliability control and ease of programming and process fault diagnosis. (www.wikipedia.org)
PSD	**P**ower **S**ourcing **D**evice, s. PSE
PSE	**P**ower **S**ourcing **E**quipment are devices that provide (source) power on the Ethernet cable. This device may be a network switch, commonly called an endspan (IEEE 802.3af refers to it as endpoint), or an intermediary device between a non-PoE-capable switch and a PoE device, an external PoE injector, called a midspan device. (www.wikipedia.org)
RMII	**R**educed **M**edia-**I**ndependent **I**nterface is a standard which was developed to reduce the number of signals required to connect a PHY to a MAC. Reducing pin count reduces cost and complexity for network hardware especially in the context of microcontrollers with built-in MAC, FPGAs, multiport switches or repeaters, and PC motherboard chipsets. Four things were changed compared to the MII standard to achieve this. These changes mean that RMII uses about half the number of signals compared to MII. - The two clocks TXCLK and RXCLK are replaced by a single clock. This clock is an input to the PHY rather than an output, which allows the clock signal to be shared among all PHYs in a multiport device, such as a switch. - The clock frequency is doubled from 25 MHz to 50 MHz, while the data paths are narrowed from 4 bits to 2 bits. - RXDV and CRS signals are multiplexed into one signal. - The COL signal is removed. (www.wikipedia.org)
SDO	**S**ervice **D**ata **O**bjects is a technology that allows heterogeneous data to be accessed in a uniform way. The SDO specification was originally developed in 2004 as a joint collaboration between Oracle (BEA) and IBM and approved by the Java Community Process in JSR 235. Version 2.0 of the specification was introduced in November 2005 as a key part of the Service Component Architecture. (www.wikipedia.org)

Abbreviations	Description
SELV	IEC defines a Separated (or **S**afety) **E**xtra-**L**ow **V**oltage system as "an electrical system in which the voltage cannot exceed ELV under normal conditions, and under single-fault conditions, including earth faults in other circuits". It is generally accepted that the acronym: SELV stands for separated extra-low voltage (separated from earth) as defined in installation standards (e.g., BS 7671), though BS EN 60335 refers to it as safety extra-low voltage. (www.wikipedia.de)
SII	The **S**ubDevice **I**nformation **I**nterface represents the EEPROM wherein the ESC configuration data is stored.
SIL	**S**afety **I**ntegrity **l**evel is defined as a relative level of risk-reduction provided by a safety function, or to specify a target level of risk reduction. (www.wikipedia.de)
SM	The **S**ync**M**anager provides a mechanism in an ESC to protect DPRAM from simultaneous access by the MainDevice and SubDevice application to guarantee data consistency.
SoE	**S**ervo drive profile **o**ver **E**therCAT according to IEC 61800 7 204
SPI	The **S**erial **P**eripheral **I**nterface is a synchronous serial communication interface specification used for short-distance communication, primarily in embedded systems. The interface was developed by Motorola in the mid-1980s and has become a de facto standard. (www.wikipedia.de)
SSC	The EtherCAT **S**ubDevice **S**tack **C**ode is an example source code in ANSI C supporting both the µC and the SPI interface. The code serves as a development base for implementation of EtherCAT in devices with own processor. (www.ethercat.org)
TC	**T**echnical **C**ommittee of the EtherCAT Technology Group
TCP/IP	The Internet protocol suite is the conceptual model and set of communications protocols used in the Internet and similar computer networks. It is commonly known as TCP/IP because the foundational protocols in the suite are the **T**ransmission **C**ontrol **P**rotocol (TCP) and the **I**nternet **P**rotocol (IP). (www.wikipedia.org)
USB	**U**niversal **S**erial **B**us is an industry standard that establishes specifications for cables and connectors and protocols for connection, communication and power supply between computers, peripheral devices and other computers. (www.wikipedia.org)
XML	E**x**tensible **M**arkup **L**anguage is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. (www.wikipedia.org)

NOTE: EtherCAT Medium Access Control and Terminology

The EtherCAT medium access control method follows the master/slave principle: only the main device sends frames, the subordinate devices process them. While it is considered ethically acceptable for one electronic device to impose communication behavior on another electronic device, there are people and institutions that have concerns about the use of these terms in technical descriptions and specifications. Since this document is not intended to offend any sensibilities, the term MainDevice (abbreviated MDevice) replaces “master” and SubordinateDevice (abbreviated SubDevice) replaces “slave”.

EtherCAT® EtherCAT® P Safety over EtherCAT®

SubDevice Implementation Guide

SECTION I – EtherCAT SubDevice introduction and implementation procedure

Technology overview, network architecture and functionality, SubDevice implementation procedure, exemplary implementation, support and training, EtherCAT Technology Group

EtherCAT® Technology Group

CONTENTS

1 Introduction I-6
1.1 Documents for detailed Information and further reading I-6
2 EtherCAT system architecture I-8
2.1 Configuration tool I-8
2.2 MainDevice I-8
2.3 SubDevice I-8
3 EtherCAT technology overview I-12
3.1 General I-12
3.2 Frame processing order I-12
3.3 SubDevice Information Interface (SII) I-13
3.4 Fieldbus Memory Management Unit (FMMU) I-14
3.5 SyncManager (SM) I-15
3.6 Distributed Clocks (DC) I-16
3.7 Data structure and communication protocols I-16
3.8 EtherCAT State Machine I-17
4 EtherCAT SubDevices implementation aspects I-20
4.1 General procedure – step by step I-20
4.2 General procedure – step by step I-20
4.3 Administrative organization I-21
4.3.1 Development time I-21
4.3.2 ETG membership and Vendor ID I-21
4.3.3 EtherCAT Conformance Test Tool license I-21
4.4 EtherCAT SubDevice design I-22
4.4.1 Bus interface to EtherCAT network I-22
4.4.2 EtherCAT SubDevice Controller (ESC) and PDI I-22
4.4.3 SII I-24
4.4.4 Application controller (host controller, µC) I-24
4.4.5 Application Layer communication protocols I-24
4.4.6 Device profiles I-27
4.4.7 Synchronization among SubDevices and the MainDevice I-29
4.4.8 Firmware update I-30
4.5 Tools for EtherCAT SubDevice development I-30
4.5.1 XML editor for generating ESI files I-31
4.5.2 EtherCAT network configurator and MainDevice software I-32
4.5.3 Monitoring communication and network diagnosis using Wireshark I-33
4.5.4 EtherCAT Conformance Test Tool I-34
4.6 EtherCAT product labels and LEDs I-35
5 EtherCAT Conformance Testing I-37
5.1 EtherCAT Conformance Test Tool I-37
5.2 Official EtherCAT Conformance Test at an EtherCAT Test Center (ETC) I-37
6 EtherCAT Development Support I-38
6.1 EtherCAT training and workshop I-38
6.2 EtherCAT development support tips I-39
7 EtherCAT Technology Group – events and support I-40
7.1 Basic information about the ETG I-40
7.2 EtherCAT Plug Fests I-41
7.3 Official EtherCAT Conformance Test certificate I-41
7.4 ETG Technical Committee I-42

7.5 Information and support I-42
7.5.1 EtherCAT Compendium I-42
7.5.2 Download area on the web site I-42
7.5.3 Knowledge Base I-42
7.5.4 Developers Forum I-43
7.5.5 Search the EtherCAT web site I-43
7.5.6 Technical support I-43

TABLES

Table 1: EtherCAT information, standards and references I-6
Table 2: EtherCAT State Machine description I-18
Table 3: EtherCAT State Machine transitions for network initialization I-19
Table 4: Components to develop/configure for EtherCAT devices I-21
Table 5: DPRAM size calculation example I-23
Table 6: FMMU configuration I-24
Table 7: The Modular Device Profile Object Dictionary I-27
Table 8: Tools I-30
Table 9: RUN and ERR LED indications I-36
Table 10: Port and L/A LED Label Requirements I-36
Table 11: EtherCAT training and workshops I-38

FIGURES

Figure 1: EtherCAT Network Architecture I-8
Figure 2: EtherCAT SubDevice architecture I-9
Figure 3: FPGA implementations of an EtherCAT SubDevice I-11
Figure 4: ESC with 4 ports, 3 open ports and frame processing order I-12
Figure 5: EtherCAT frame structure I-13
Figure 6: SII layout I-14
Figure 7: Mapping Example of Process Data with FMMU I-15
Figure 8: SyncManager in mailbox mode I-15
Figure 9: SyncManager 3-buffer-mode I-16
Figure 10: EtherCAT State Machine I-18
Figure 11: EtherCAT network initialization I-18
Figure 12: EtherCAT device development procedure I-20
Figure 13: ESC structure for CAN application profile applications I-25
Figure 14: SubDevice Stack Code overview I-26
Figure 15: SubDevice control stack I-26
Figure 16: product filter SSC I-27
Figure 17: MDP schema for modular devices I-28
Figure 18: ESI structure (EtherCATInfo.xsd) I-31
Figure 19: ESI file editing using CTT I-32
Figure 20: ESI file generation using a graphical editor (Altova XML Spy®) I-32
Figure 21: EtherCAT network configurator I-33
Figure 22: TwinCAT device scan, box scan and adapter settings I-33
Figure 23: Wireshark Screenshot I-34
Figure 24: Testing with the Conformance Test Tool I-35
Figure 25: EtherCAT product branding logos I-36
Figure 26: EtherCAT Conformance Tested logos I-41
Figure 27: ETG download section - filter options I-42
Figure 28: ETG webpage search I-43

1 Introduction

This chapter presents a brief overview to the EtherCAT architecture and technology. Since EtherCAT technology covers more details than presented here, a list of documents which provide deeper understanding of the technology is given first. Corresponding text passages in this guide refer to these documents. In the following chapters the basic system architecture and the system functionality of an EtherCAT network is described. Since this is a SubDevice implementation guide, it focuses on the SubDevice.

1.1 Documents for detailed Information and further reading

It is recommended to consider the following information before proceeding to develop an EtherCAT device. Some of the information below is provided in the member area[1] of the website of the EtherCAT Technology Group (ETG). ETG membership is free of charge and is required to access the wide range of EtherCAT related documents, specifications, guidelines, and includes technical support from the ETG. See chapter 4.3.2 for how to become a member and to get an account.

The complete list of all available EtherCAT documentation can be found at the download section of the ETG website (www.ethercat.org/download). Table 1 lists documents related to SubDevice implementation and general EtherCAT technology overview.

Table 1: EtherCAT information, standards and references

	Subject	Documents, description and access
Introduction	Brochures and presentations	EtherCAT is introduced in several brochures, published in different languages: → English | Japanese | Chinese | German | Korean | Italian | Spanish This description of EtherCAT technology basics is an introduction in → English | Japanese | Chinese | German | French | Italian | Portuguese An introduction to Safety over EtherCAT is available in → English | German
	Articles	EtherCAT has been introduced in several articles. A selection of them is given here. → Elektronik 23/03 (German) → AUTlook 2-3/05 (German)
	Videos (YouTube)	https://www.youtube.com/user/EtherCATGroup → EtherCAT Technology Group → EtherCAT in 20 Minutes → EtherCAT Functional Principle → Safety over EtherCAT → EtherCAT Communication Profiles
Detailed Reading	EtherCAT Compendium	This is **the** EtherCAT read - from getting started to understanding the functionalities themselves, their purpose, and the models behind. It describes the protocol as a whole and puts it into context in a very comprehensive and easy-to-read style. Those are the big add-ons compared to the very precise specification. → http://www.ethercat.org/compendium[1]
	Knowledge Base	An online information system containing FAQs and EtherCAT feature descriptions. → www.ethercat.org/kb[1]
	Technology description	Section I of the Beckhoff EtherCAT SubDevice Controller Datasheet ET1100 contains a comprehensive description of EtherCAT functionality. Sections II (ESC register description) and section III (hardware specification) provide more detailed information. → beckhoff.com > Products > I/O > EtherCAT development products

1 ETG membership sign-in required.

	Subject	Documents, description and access
	Proceedings of ETG events	Minutes of the Technical Committee meetings hold actual technology development topics: → www.ethercat.org > Downloads > Select Filter: Proceedings and Papers > Technical Committee Meeting
Development	EtherCAT Communications	The communication slides provide a broad description of EtherCAT mechanisms for developers. → English[1] | Japanese[1]
	PHY Selection Guide	The PHY Selection Guide contains information for physical level connection components of several vendors that are available for EtherCAT communication. → PHY Selection Guide (Beckhoff)
	Individual topics of ITW	The ITW world series - a week of webinars has become a standing event in ETG's event calendar. Webinar slides can be downloaded: → www.ethercat.org/downloads → Test Filter = itw
	Focus topics Technical Committee meeting	The focus topic presented during the Technical Committee meetings provide answers to many typical topics of interest. They are included in the meeting proceedings. → www.ethercat.org/downloads → Test Filter = tc
Specifications	Communication specification	EtherCAT is specified by the EtherCAT communication specification ETG.1000 parts 2 to 6. → www.ethercat.org/etg1000[1] Note ETG.1000 represents the IEC 61158 - Type 12 (EtherCAT).
	EtherCAT SubDevice Information (ESI)	The ETG.2000 specification defines the EtherCAT SubDevice Information (ESI) for the EtherCAT device description in XML format. Device description example files can also be found here. The ETG.2001 ESI Annotation specification includes sample files for ESI file development. → www.ethercat.org/etg2000[1]
	Safety over EtherCAT	Safety over EtherCAT specifies a protocol layer for safe data exchange. ETG.5100 contains the safety protocol and ETG.6100 specifies a safety drive profile. → www.ethercat.org/etg5100[1] → www.ethercat.org/etg6100[1] Note ETG.5100 represents the IEC 61784 international standard.
	Drives	The implementation directive for the CiA402 Drive Profile is specified in the ETG.6010 specification. → www.ethercat.org/etg6010[1] Note ETG.6010 is based on the IEC 61800-7-201 (CiA402 drive profile).
	Conformance	Conformance test rules are defined in the EtherCAT Conformance Test Policy ETG.9000. The conformance guide describes how developers can obtain conformance (ETG.7000). Additionally, a test record and the test request form are available here. → www.ethercat.org/etg7000[1]
	Firmware update	ETG.5003.0002 Firmware Update specification This specification is mandatory only for devices supporting the profile number 5003 (Semi Device Profile), however, it is a good guideline for a firmware update implementation on any EtherCAT SubDevice → www.ethercat.org/etg5003
	Trademark, logo and labelling rules	Marking rules, trademark, logo and labelling usage for products and documentations applying EtherCAT technology or referring to it are defined in the ETG.1300 and the ETG.9001 specifications: → www.ethercat.org/etg1300[1] → www.ethercat.org/etg9001[1]

2 EtherCAT system architecture

The basic EtherCAT system configuration is shown in Figure 1. The EtherCAT MainDevice uses a standard Ethernet port and network configuration information stored in the EtherCAT Network Information (ENI) file. The ENI is created based on EtherCAT SubDevice Information (ESI) files which are provided by the vendors for each device. SubDevices are connected via Ethernet, any topology type is possible for EtherCAT networks.

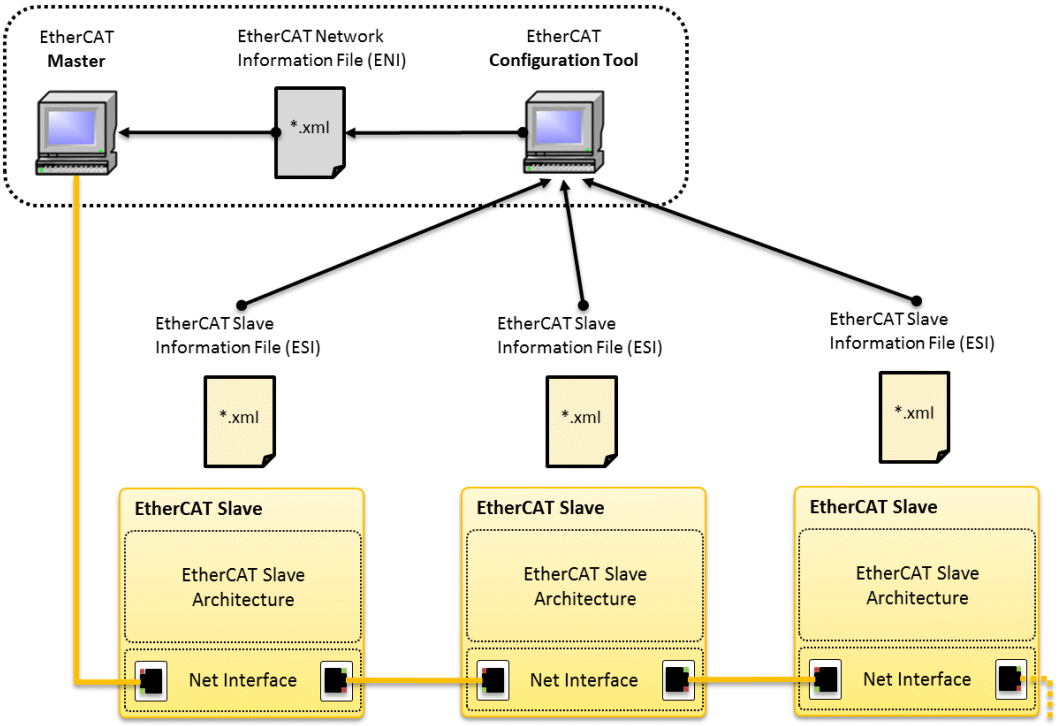


Figure 1: EtherCAT Network Architecture

2.1 Configuration tool

- EtherCAT Configuration Tool
  The EtherCAT Configuration Tool is used to generate network description, the so called EtherCAT Network Information file (XML file based on a pre-defined file schema). This information is based on the information provided by the EtherCAT SubDevice Information files (device description in XML format, see chapter 2.3) and/or the online information provided by the SubDevices in their SII and their object dictionaries.
- EtherCAT Network Information (ENI) file
  The ENI file describes the network topology, the initialization commands for each device and the commands which have to be sent cyclically. The ENI file is provided to the MainDevice, which sends commands according to this file. For more information see ETG.2100 EtherCAT Network Information specification.

2.2 MainDevice

- Hardware: The only hardware requirement for an EtherCAT MainDevice is a standard Network Interface Controller (NIC, 100 Mbit/s full duplex).
- Software: A real time runtime environment drives the SubDevices in the network. Since this guide focuses on the SubDevice, it won't get into detail to MainDevice software. Further information is available at the ETG website's product section.

2.3 SubDevice

Figure 2 shows the EtherCAT network with focus on the SubDevice architecture. Basically, the SubDevice is structured in three main components:

- Physical Layer (PhL): Network interface
- Data Link Layer (DLL): EtherCAT SubDevice Controller (ESC, communication module) and SII (EEPROM)

- Application Layer (AL): Application controller (also called host controller, microcontroller, µC)

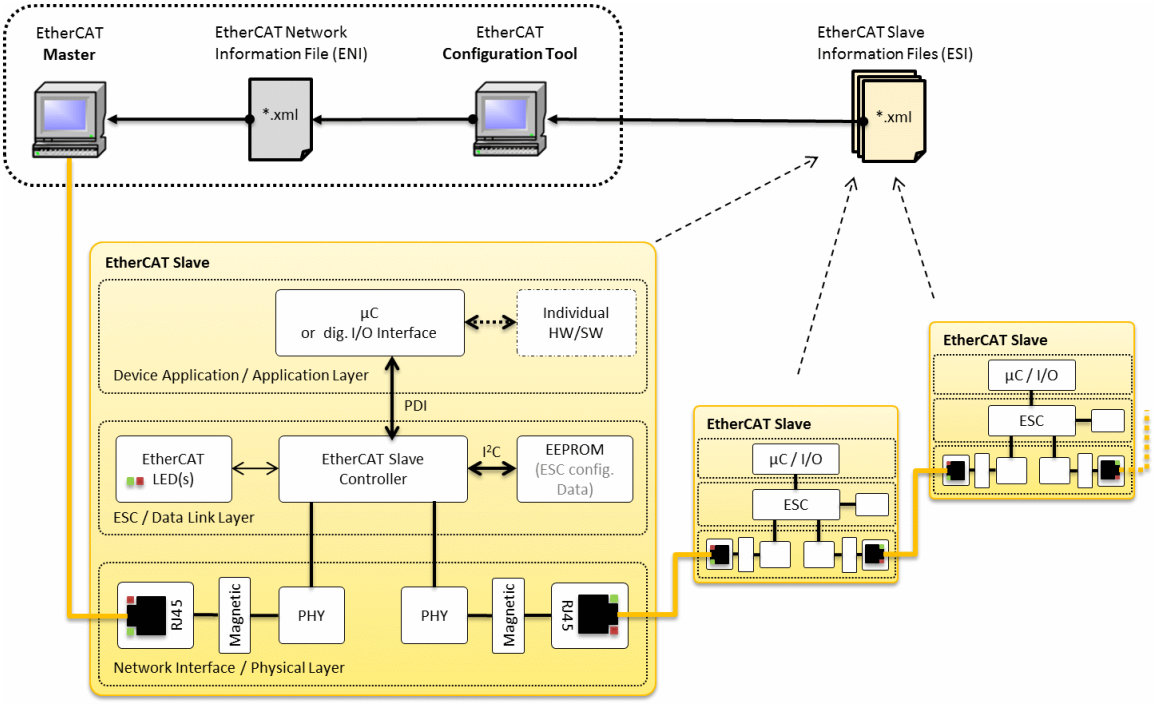


Figure 2: EtherCAT SubDevice architecture

In detail, the SubDevice consists of the following components. Criteria for these components concerning the device design and development are discussed in chapter 4.4.

- **Network interface: Standard Ethernet physical layer components**

The network interface contains the physical layer (PHY) components to process fieldbus signals. It forwards network data to the ESC and applies signals from the ESC to the network. The physical layer is based on the standards defined by standard Ethernet (IEEE 802.3).

1 Plugs:
Ethernet cable connectors. Typically, RJ45 connectors (recommended) or M12 D-code connectors for EtherCAT or M8 P-coded connectors for EtherCAT P. As EtherCAT cables, shielded twisted pair enhanced category 5 (Cat 5e STP) or better is recommended. Select an appropriate cable for the environment where the machine is installed.
2 Magnetics:
Pulse transformers for galvanic isolation.
3 Standard PHYs:
A chip that implements the hardware functions for sending and receiving Ethernet frames. It interfaces to the line modulation at one end and binary packet signaling at the other. Refer to the PHY selection guide for details.

- **EtherCAT SubDevice Controller (ESC)**

The ESC is a chip for EtherCAT communication. The ESC handles the EtherCAT protocol in real-time by processing the EtherCAT frames on the fly and providing the process data interface (PDI) for data exchange between EtherCAT MainDevice and the SubDevice's local application controller via registers and a Dual-Port-RAM (DPRAM).

The ESC can either be implemented as Field Programmable Gate Array (FPGA) or as Application Specific Integrated Circuit (ASIC). The EtherCAT frame is completely processed by the ESC, and hence, the processing speed is basically the same for any EtherCAT SubDevice. It does not depend on the performance of the application controller implementing the application. At the same time, the application on the application controller does not need to process the Ethernet frame forwarding, and hence, the application controller resources are free to be used by the SubDevice's application. The processing frequency of the application controller is defined by the host application.

The ESC processes EtherCAT frames on the fly and provides data for a local application controller or digital I/Os via the Process Data Interface (PDI). Different ESCs might support different PDIs, common PDIs are

- digital I/O interface with 32 bit I/O pins
- Serial Peripheral Interface (SPI)
- 8/16-bit synchronous/asynchronous application controller interface
- for ESC IP-cores: native FPGA on-board-buses to connect a soft-core (e.g., Avalon on Intel/Altera FPGAs, OPB on Xilinx FPGAs)

Process data and parameters are exchanged via a DPRAM in the ESC, between EtherCAT and the application controller. Data consistency is ensured by the ESC "SyncManagers"(SM), chapter 3.5.

Figure 3 shows two possible FPGA implementations of the ESC

In case of an FPGA implementation, the ESC is realized as IP. The ESC features are configurable regarding number of Fieldbus Memory Management Units (FMMUs) and SMs, Distributed Clocks and PDI type (chapter 4.4).

- **SII: EEPROM (ESC configuration data and application specific data)**

The SubDevice Information Interface (SII) is typically stored on an EEPROM device and contains hardware configuration information for the ESC which is loaded to the ESC's registers during power-up. The ESC activates the defined PDI so that the DPRAM can be accessed from the local application controller.

The SII content can be written via a configuration tool (via EtherCAT) based on the ESI file. The application controller can also access the EEPROM if access rights are assigned. However, the EEPROM is always physically accessed via the ESC, which in turn interfaces to it via Inter-Integrated Circuit (I^2C) data bus.

- **Application Layer (AL) / application controller**

Application layer services, i.e. communication software and device-specific software, is implemented on an application controller, which handles the following:

1 EtherCAT State Machine (ESM) in the SubDevice (chapter 3.8)
2 Process data exchange with the SubDevice application (e.g., application and configuration parameters, object dictionary, chapter 4.4.6)
3 Mailbox-based protocols for acyclic data exchange (CoE, EoE, FoE, chapter 3.7)
4 Optional TCP/IP stack if the device supports EoE

The application controller-performance affects solely the device application, not the performance of the EtherCAT communication. In many cases an 8-bit application controller / PIC is sufficient.

- **EtherCAT SubDevice Information File (ESI)**

Every EtherCAT SubDevice must be delivered with an ESI file, a device description file in XML format. Information about device functionality and settings is provided by the ESI. ESI files are used by the configuration tool to compile an EtherCAT network information (ENI) in offline mode (incl. process data structures, initialization commands, cyclic commands).

Refer to ETG.2000 EtherCAT SubDevice Information specification for the description details of the ESI file. See also related description in chapter 4.5.1.

- **Application-specific hardware (HW)**

Additional hardware may be required for the SubDevice-specific functionality e.g., optics/optoelectronics in sensors, plugs in gateways, displays. This hardware is connected to the application controller and is not understood as part of the EtherCAT communication functionality, here.

FPGA implementations allow the two different Implementation models shown in Figure 3

One way is integrating ESC and a soft core application controller on the FPGA. the FPGA on-board bus can then be used as PDI. Another option is using the FPGA solely for the ESC functionality and connecting an external application controller via application controller/SPI.

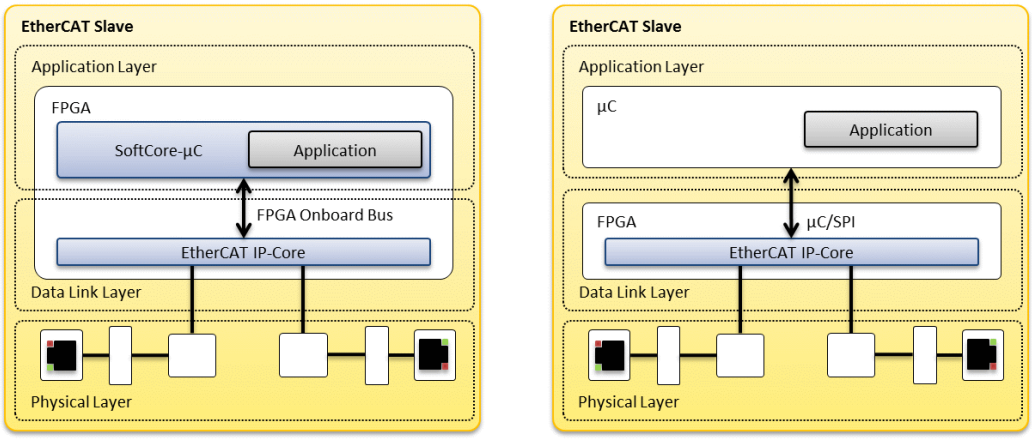


Figure 3: FPGA implementations of an EtherCAT SubDevice

A plug-in for Altera or Xilinx development environments is available to configure the IP core. The IP core is provided by Beckhoff Automation and different license models are offered for available FPGA devices.

3 EtherCAT technology overview

3.1 General

In this chapter, basic EtherCAT SubDevice features and functionalities are explained in a short. Refer to referenced material in chapter 1.1 for more details.

In general, the EtherCAT Compendium provides a very good introduction.

3.2 Frame processing order

The ESC provides up to 4 ports at maximum. Port 0 is defined as the IN port. SubDevices should provide at least two EtherCAT ports. In case the SubDevice has two ports, ports 0 and 1 should be used (e.g., in modular devices).

Any physical EtherCAT network topology always forms a logical ring since the frame processing in a SubDevice works like a roundabout, see Figure 4. The ESCs are connected to upstream (MainDevice) always via port 0 and to downstream (following SubDevices) via ports 1 to 3. The frame processing is done only once per ESC in the EtherCAT Processing Unit (EPU) which is located after port 0. Thus, returning frames will not be processed again but are only passed to the next port (as shown on port 1) or returned to port 0 (as shown on port 2).

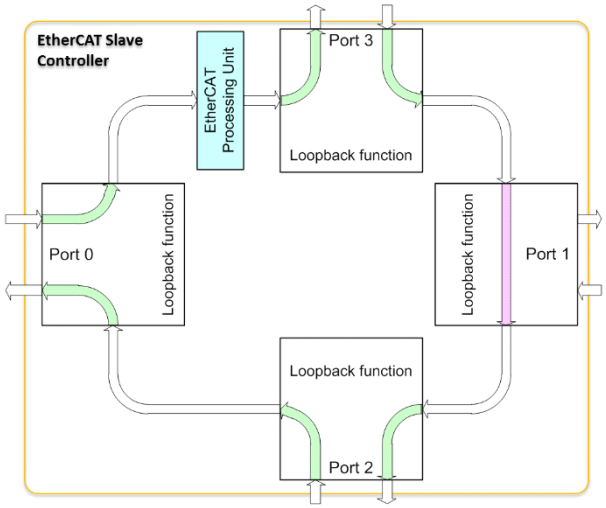


Figure 4: ESC with 4 ports, 3 open ports and frame processing order

EtherCAT frames (Ethernet frames with EtherType 0x88A4, see Figure 5) are processed by the ESC on the fly[1]. Processing of EtherCAT datagrams is started before the complete frame has been received. Thereby Figure 5 describes the EtherCAT frame(1) as used for real-time communication and 5 shows, that EtherCAT frames (2) can also be routed via IP and hence, sent via sockets. However, since the IP software stack introduces jitter, it is not used for real-time communication but might be for testing purposes. In case the frame has been corrupted, the Frame Check Sequence (FCS) does not match, and the ESC does not copy the received data to the DPRAM for the local application (e.g., PLC application).

1 For visualization, watch https://www.youtube.com/watch?v=z2OagcHG-UU

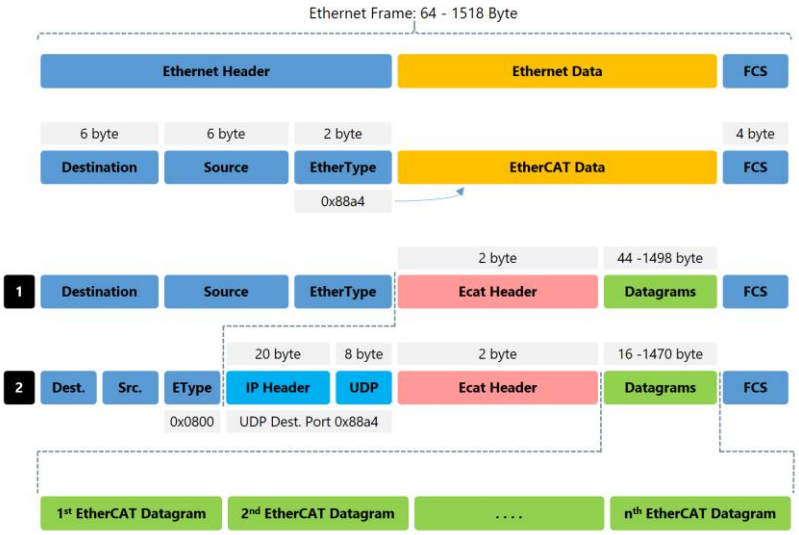


Figure 5: EtherCAT frame structure

3.3 SubDevice Information Interface (SII)

Since the DPRAM in the ESC is volatile RAM, it is connected to an EEPROM (non-volatile memory, also called SubDevice Information Interface, SII). The SII stores SubDevice identity information and information about the SubDevice's functionality corresponding to the ESI file, see Figure 6. The content of the EEPROM must be configured by the vendor during development of the SubDevice. Details about the structure of the EEPROM information can be derived from the ESI file. For the SII specification, refer to ETG.2010 and ETG.1000.6.

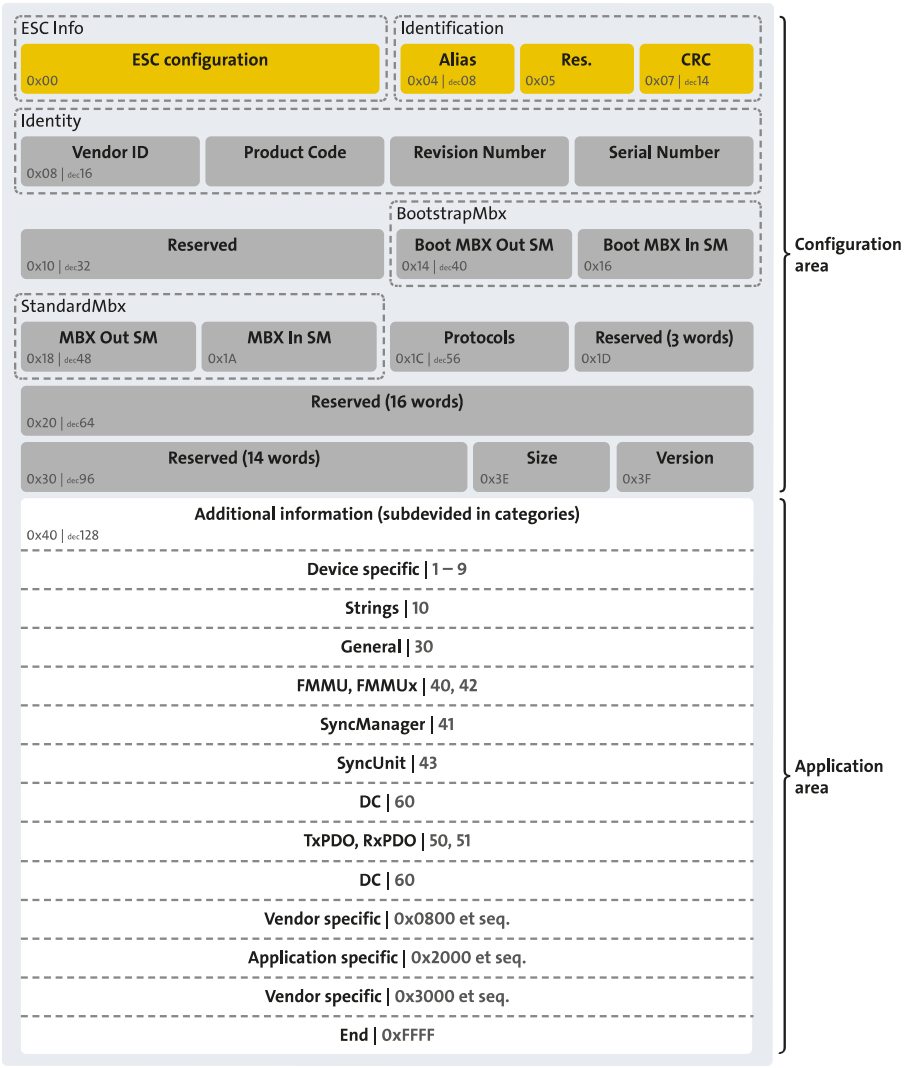


Figure 6: SII layout

3.4 Fieldbus Memory Management Unit (FMMU)

Fieldbus Memory Management Units are used to map data from the (logical) process data image in the MainDevice to the physical (local) memory in the SubDevices (see Figure 7). Process data in the MainDevice's image is arranged by tasks. Related to this, the MainDevice configures via the FMMUs which EtherCAT SubDevices map data into the same EtherCAT datagram to automatically group process data. The FMMUs thereby reduce a significant amount of CPU time in the MainDevice and save bandwidth in the network.

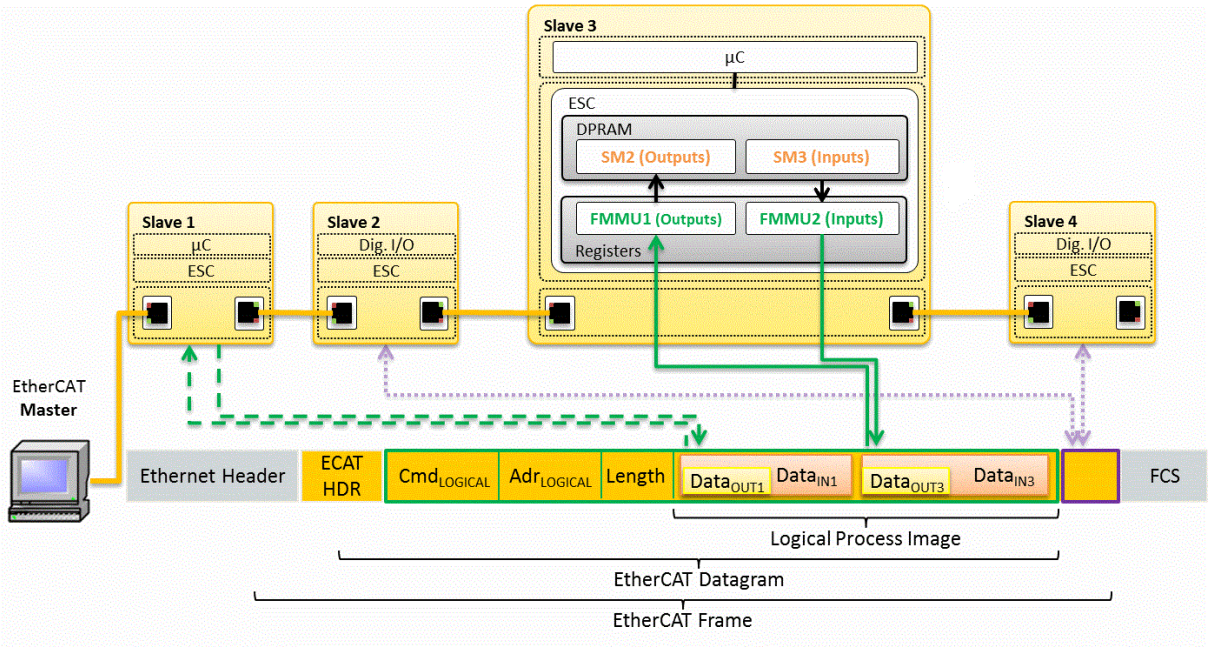


Figure 7: Mapping Example of Process Data with FMMU

3.5 SyncManager (SM)

Since both the EtherCAT network (MainDevice) and the PDI (local application controller) access the DPRAM in the ESC, the DPRAM access needs to ensure data consistency. The SyncManager is a mechanism to protect data in the DPRAM from being accessed simultaneously. If the SubDevice uses FMMUs, the SMs for the corresponding data blocks are located between the DPRAM and the FMMU. EtherCAT SMs can operate in two modes, mailbox mode and buffered mode.

Mailbox mode

The mailbox mode (Figure 8) implements a handshake mechanism for data exchange. EtherCAT MainDevice and application controller application only get access to the buffer after the other one has finished its access. When the sender writes the buffer, the buffer is locked for writing until the receiver has read it out. The mailbox mode is typically used for application layer protocols and exchange of acyclic data (e.g., parameter settings).

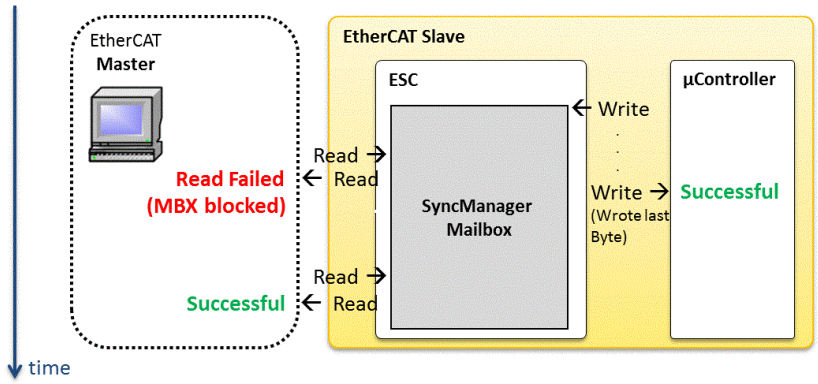


Figure 8: SyncManager in mailbox mode

Buffered mode

The buffered mode (Figure 9) is typically used for cyclic data exchange, i.e. process data since the buffered mode allows access to the communication buffer at any time for both sides, EtherCAT MainDevice and SubDevice. The sender can always update the content of the buffer. If the buffer is written faster than it is read out by the receiver, old data is dropped. Thus, the receiver always gets the latest consistent buffer content which was written by the sender.

Note, SyncManagers running in buffered mode need three times the process data size allocated in the DPRAM.

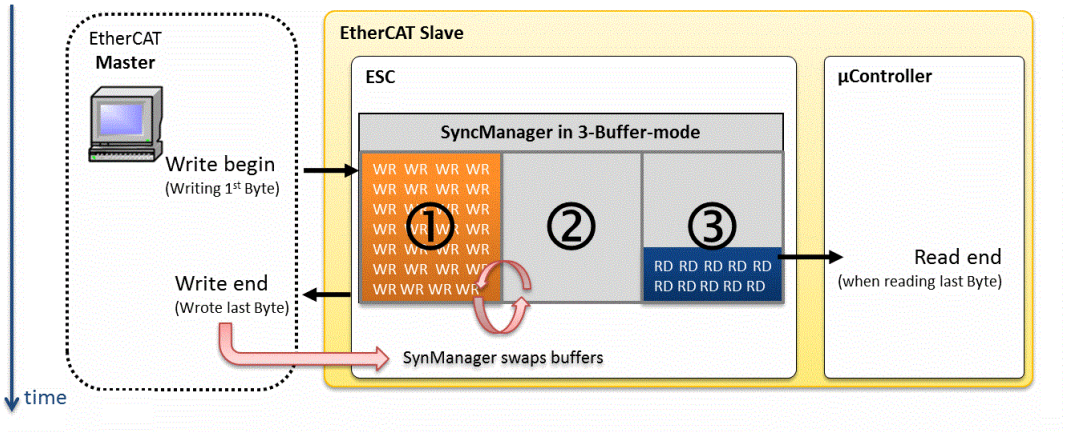


Figure 9: SyncManager 3-buffer-mode

3.6 Distributed Clocks (DC)

The method of Distributed Clocks provides high precise synchronization between SubDevices in an EtherCAT network. Since DC refers to the ESC-internal clocks, synchronization time between SubDevices can be guaranteed to much better than 1µs.

The requirement of DC depends on the necessity of synchronization precision of the developing SubDevice. For instance, in machines in which multiple servo drives are functionally coupled, the axes need to be precisely synchronized to perform coherent movement. For this reason, many SubDevices for servo drive adopt DC in order to achieve high precise synchronization with other SubDevices. The DC functionality should be implemented in cases of servo drive systems or I/O SubDevices being synchronized with servo drives.

3.7 Data structure and communication protocols

Data is exchanged cyclically or acyclically and data sizes can be fixed or configurable. For acyclic data exchange, EtherCAT provides mailbox communication protocols (CoE, SoE, EoE, FoE, AoE). Cyclic data is exchanged in Process Data Objects (PDOs) with fixed or configurable PDO sizes. In the following, the mailbox application protocols are described.

CoE: CAN application protocol over EtherCAT

This is the most used EtherCAT mailbox application protocol. CoE also provides mechanisms to configure PDOs for cyclic data exchange.

Several device profiles can be applied for EtherCAT devices by using CoE. For example the drive profile CiA402 (IEC61800-7-201) is mapped to EtherCAT this way and described in more detail in the ETG.6010 Implementation Directive for the CiA402 Drive Profile.

For all other devices, the ETG.5001 Modular Device Profile Specification defines a standardized structure for the object dictionary provided by CoE. In particular, for gateways or bus couplers, these structures are enhanced by helpful configuration mechanisms.

ETG.5003 Semiconductor Device Profile series describes a wide range of process-oriented sensors and actuators used in semiconductor manufacturing equipment, including those for mass flow controllers, temperature controllers, vacuum pumps, and much more. Since EtherCAT has become de facto standard in the semi industry, the device profiles have a very high acceptance, too.

SoE: Servo drive profile over EtherCAT

SERCOS interface[1] is a communication interface, particularly for motion control applications. The SERCOS profile for servo drives is specified by the IEC 61800-7 standard. The mapping of this profile to EtherCAT is specified in ETG.1000.3.

1 SERCOS interface is a trademark of the SERCOS International e.V.

The service channel, and therefore access to all parameters and functions residing in the drive, is based on the EtherCAT mailbox. Here too, the focus is on compatibility with the existing protocol (access to value, attribute, name, units etc.) and expandability with regard to data length limitation. The SERCOS process data is transferred using EtherCAT SubDevice controller mechanisms.

EoE: Ethernet over EtherCAT

The EtherCAT technology is not only fully Ethernet-compatible, but the protocol tolerates other Ethernet-based services and protocols on the same physical network. The Ethernet frames are tunneled via the EtherCAT protocol, which is the standard approach for internet applications (similar to VPN, PPPoE (DSL) etc.). The EtherCAT network is fully transparent for the Ethernet device, and the real-time characteristics are not impaired.

EtherCAT devices can additionally provide other Ethernet protocols and thus act like a standard Ethernet device. The MainDevice 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, e-mail, FTP transfer etc.

FoE: File access over EtherCAT

FoE (File Access over EtherCAT) is a mailbox application protocol generally intended to transfer file data on an EtherCAT network in both directions, and as such it can be used in any state where the mailbox communication is active (PreOP, SafeOP, OP). The most common use case for FoE is to download a new firmware for update when the SubDevice is in Bootstrap state. The EtherCAT SubDevice can run in the optional Bootstrap state to support a firmware download using FoE to the application controller via the EtherCAT network. A standardized firmware download to EtherCAT devices is therefore possible, even without the support of TCP/IP.

AoE: ADS over EtherCAT

ADS over EtherCAT (AoE) is a standard, client-server Mailbox application protocol defined by the EtherCAT specification.

AoE is routable and supports the contemporary handling of parallel services, and therefore is particularly suitable for the access to sub-networks via gateway devices (MainDevices which are required to operate gateway SubDevices should therefore support AoE). Due to its routability, AoE is also defined as standard protocol for the communication between an EAP (EtherCAT Automation Protocol) network and single EtherCAT fieldbus segments.

Both MainDevice and SubDevice stacks exist which support basic AoE services, enabling a light and fast implementation of this mailbox application protocol in EtherCAT devices.

3.8 EtherCAT State Machine

The SubDevice runs a state machine to indicate which functionalities are currently available. This state machine is called EtherCAT State Machine (ESM) and is shown in Figure 10.

ESM requests are written by the MainDevice to the SubDevice’s AL Control register in the ESC. If the configuration for the requested state is valid, the SubDevice acknowledges the state by updating the AL Status register. If not, the SubDevice sets the error flag in the AL Status register and writes an error code to the AL Status Code register.

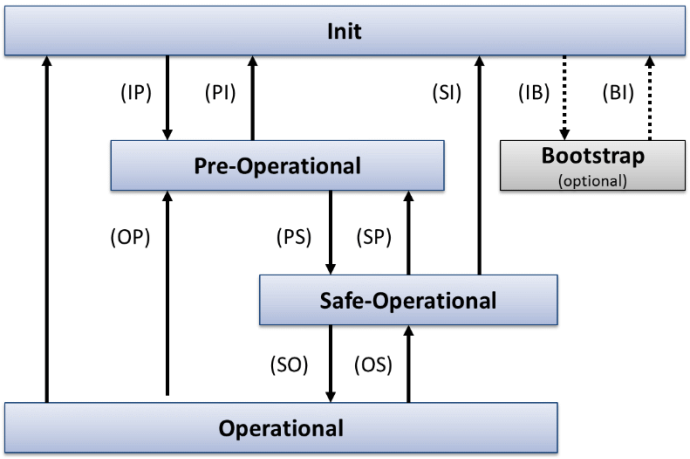


Figure 10: EtherCAT State Machine

The states are described in Table 2. For further information, refer to ETG.1000.6.

Table 2: EtherCAT State Machine description

ESM state	Available functionalities
Init (INIT)	Init state. No communication on the application layer is available. The MainDevice has access only to the DL-information registers.
Pre-Operational (PREOP)	Pre-Operational state. Mailbox communication on the application layer available, but no process data communication available.
Safe-Operational (SAFEOP)	Safe-Operational state. Mailbox communication on the application layer, process (input) data communication available. In Safe-Operational only inputs are evaluated; outputs are kept in 'safe' state.
Operational (OP)	Operational state. Process data inputs and outputs are valid.
Bootstrap (BOOT)	Bootstrap state. Optional but recommended if firmware updates necessary No process data communication. Communication only via mailbox on Application Layer available. Special mailbox configuration is possible, e.g., larger mailbox size. In this state usually the FoE protocol is used for firmware download.

The initialization information for every EtherCAT state transition is based on the ESI, a network configurator saves it to the ENI. Each SubDevice receives its required initialization commands for each state transition. The EtherCAT MainDevice maintains independent state machines per EtherCAT SubDevice in the network. The state transition control sequences are shown in Figure 11.

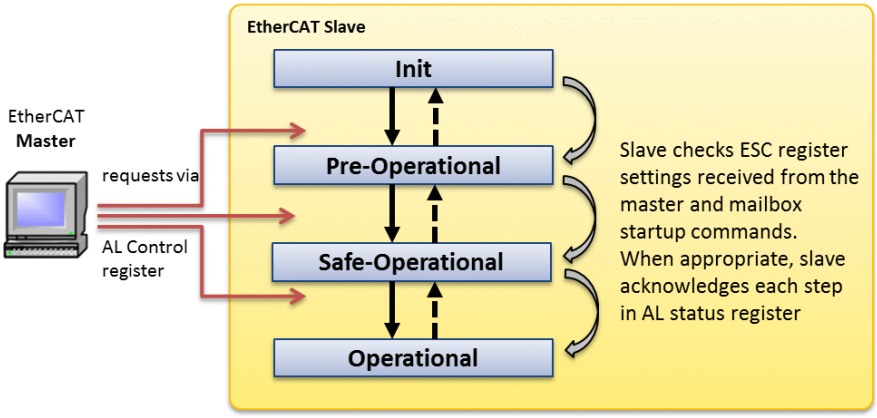


Figure 11: EtherCAT network initialization

For the development of (complex, i.e. SubDevices using an application controller) EtherCAT SubDevices, the handling of the state transition commands is mandatory. The prerequisite for the

state machine functionality is the successful reception and acknowledgement of the state transition requests in the EtherCAT SubDevice (reading/writing AL Control / AL Status registers). When the MainDevice sends a state request, the acknowledgement must not be given before the register and application configuration corresponding to the requested state is validated by the local application controller. An excerpt of EtherCAT state machine transitions for network initialization is presented in Table 3. Full data exchange with the MainDevice is enabled when the SubDevice switches to the operational state. The state machine handling is subject to tests in the EtherCAT Conformance Test Tool.

Table 3: EtherCAT State Machine transitions for network initialization[1]

Transition	MainDevice to SubDevice settings description
Init to Pre-Operational	MainDevice reads VendorID, ProductCode and RevisionNumber from EEPROM, and configures DL control registers (register 0x0100:0x0103) SyncManager registers (registers 0x800+) for mailbox communication, initialization for DC clock synchronization (if supported). MainDevice requests Pre-Operational state by writing the AL Control register (register 0x120) and waits for status confirmation via the AL Status register (register 0x130).
Pre-Operational to Safe-Operational	MainDevice configures parameters using mailbox communication, i.e.: Process data mapping if flexible, registers for process data SyncManagers, FMMU registers (0x600 and following). MainDevice requests Safe-Operational state (AL Control register 0x0120 = 0x04) and waits for confirmation via AL Status register.
Safe-Operational to Operational	MainDevice sends valid outputs and requests Operational state (AL Control register 0x0120 = 0x08, confirmation in AL Status register)
Error to Init Error to Pre-Operational Error to Safe-Operational	Incorrect ESC register configuration (DC, FMMU, SM, etc.). The AL Status Code register (register 0x134) indicates error reasons.

1 Detailed description is available in the ETG.1000 EtherCAT Communication specification (Part 6, table 103).

4 EtherCAT SubDevices implementation aspects

4.1 General procedure – step by step

This chapter shows the procedure for a typical EtherCAT SubDevice implementation process. The overview to the steps is given in chapter 4.2. The steps are described in more detail in the denoted chapters. Chapter 4.3 contains details for administrative organization. Chapters 4.4 are a detailed descriptions of the development steps. Herein, some application notes are given as well. Chapters 6 and 7 describe support provided by the ETG.

4.2 General procedure – step by step

A well proven approach to an EtherCAT SubDevice implementation is given in the following Figure 12.

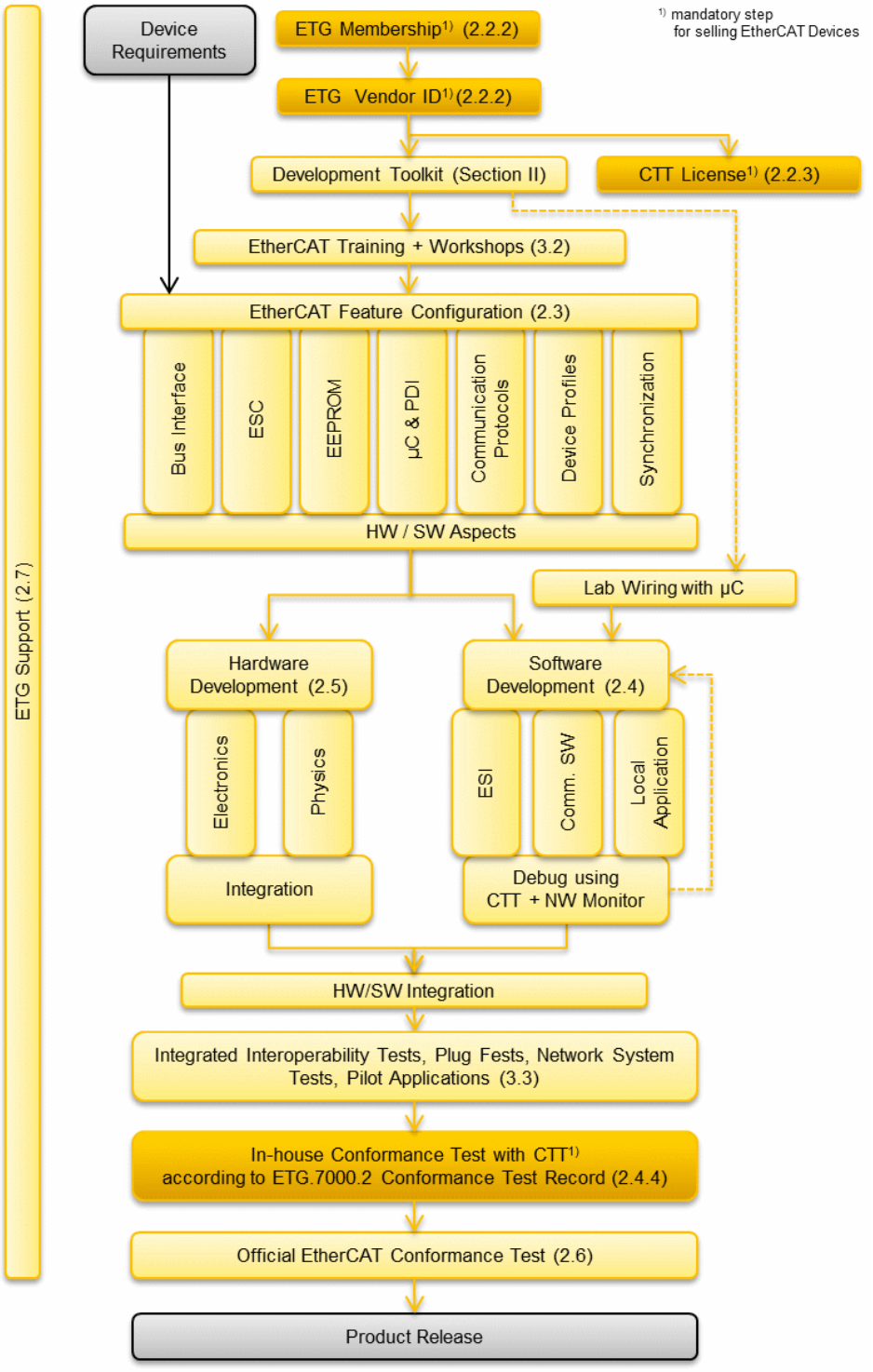


Figure 12: EtherCAT device development procedure

4.3 Administrative organization

4.3.1 Development time

To develop a new running SubDevice system, operated by a standard EtherCAT MainDevice, about 6-8 weeks are feasible to get a working solution. Herein, parts of the own application development are already included.

The hardware design of the device depends on device type (complex device with application controller or simple device without application controller) and the amount and type of ports (MII). Table 4 shows the components needed for a SubDevice.

Table 4: Components to develop/configure for EtherCAT devices

	Category	Simple device (no application controller, dig. I/O)	Complex device (with application controller)
Hardware	Application controller	--	microcontroller Programmable Memory (RUN/ERR LEDs)
	ESC	ESC (ASIC/IP Core) EEPROM	
	Port connection	MII: Plug, Magnetics, PHY, R/C Link/Activity LEDs	
	Device casing	Coverage design, or additional individual hardware etc.	
Software	Host application	--	Local application/Firmware (FW) EtherCAT communication
	Device description	ESI file EEPPROM configuration	
	Documentation	EtherCAT SubDevice documentation	

4.3.2 ETG membership and Vendor ID

Each EtherCAT compliant device must carry a worldwide unique Vendor ID assigned by the EtherCAT Technology Group (chapter 7), which requires ETG membership as well.

ETG membership is free of charge and covered by the ETG Membership By-Laws. For application send your membership request in an email to info@ethercat.org.

The Vendor ID usage is covered by the ETG.9002. The application for the Vendor ID can be done online (membership login data is required). The Vendor ID is free of charge as well. The EtherCAT Vendor ID is mandatory to meet the EtherCAT Conformance Test requirements.

4.3.3 EtherCAT Conformance Test Tool license

There are two reasons why to buy an EtherCAT Conformance Test Tool (CTT) license.

- The CTT assists EtherCAT device development by checking protocol compliance in-house and supports preparation for the official EtherCAT Conformance Test (chapter 7.3). It also delivers a good deal of development and testing supported by many built-in features, including a remote control interface to run it by a script.
- The application of the CTT for in-house tests is mandatory when selling the device to the market.

The tests performed by the CTT are specified by the ETG Working Group Conformance. The CTT software is provided by Beckhoff Automation GmbH & Co. KG.

***Important to know*:** To guarantee long-time availability of the CTT, i.e. ensuring maintenance of the software such as adding support for new operating systems, the ETG membership assembly 2008 decided unanimously for the following model: The CTT comes on a subscription basis extending itself automatically every year. A new license file is automatically being provided to the vendors. Each

EtherCAT SubDevice manufacturer who offers EtherCAT SubDevices to the market or builds own SubDevices to integrate them into their machines, shall obtain and maintain a valid subscription.

Before canceling the subscription, checking if this would violate the policies (especially the Conformance Test Policy, ETG.9003) is obligatory. In case of a cancellation, usually a standard lead-time of 3 months before the renewal of the license applies.

Support is provided by ETG (conformance@ethercat.org).

4.4 EtherCAT SubDevice design

EtherCAT features are to be selected according to the device requirements. Thus, to develop an EtherCAT SubDevice, the developer should be conscious about the requirements of the device to decide which characteristic is to be chosen for every EtherCAT feature.

In the following, an overview to the design criteria is given of which the ESC is the most important EtherCAT characteristic. The configuration of these criteria is finally stored in the ESI file and the SII.

4.4.1 Bus interface to EtherCAT network

Support of the desired bus interface(s) must be regarded in the selection of the ESC. It is one of the main criteria for ESC types.

For stand-alone devices which are connected to the network via 100BaseTX or 100BaseFX, Media Independent Interface (MII) is used.

Application note: A stand-alone device should support at least two MII ports (RJ45 or M12 D-Code connectors for EtherCAT or M8 P-coded for EtherCAT P) to provide line connection. The logical port for connection is determined based on the number of ports being used. For standard 2 port usage, port0 and port1 are used. The PHYs should be selected according to the PHY Selection Guide.

4.4.2 EtherCAT SubDevice Controller (ESC) and PDI

The ESC is the controller which provides the communication interface between the EtherCAT network and the application controller or the digital I/O (if no application controller is used).

Basically, the ESC can be implemented as ASIC or as FPGA with IP core. The EtherCAT functionality is the same for both types, so the choice which type to use is up to the vendor. If preferring an ASIC, an additional EEPROM is necessary and the DPRAM may be limited to less than 64 kB (depending on the ESC).

If know-how of FPGA programming is available and intellectual property (IP core) is already at hand, the choice for an FPGA implementation is obvious and the IP core only needs to be adapted to the EtherCAT communication. An FPGA may also be an option if hardware space for both an ASIC and an EEPROM is not available.

An overview of available ASICS and FPGAs is given by the ETG in chapter 4 of section II or in the ESC Overview. In the following, the ESC selection criteria are discussed in more detail.

- **Number and type of EtherCAT ports (MII)**

Basically, most EtherCAT devices have two ports so that they can be connected in a line topology. The number of ports and port type are a key selection criteria of ESCs.

- **Interface for process data exchange (PDI)**

Simple devices directly provide the digital I/O as PDI.

The PDI in Complex devices operates as a Serial Peripheral Interface (SPI) or as a 8/16 bit parallel parallel interface in synchronous or asynchronous mode.

If using an EtherCAT IP core, the FPGA specific on-board-bus is applied as PDI since ESC, SII (EEPROM) and application controller are integrated in the IP core. On Altera devices Avalon is used resp. OPB on Xilinx devices.

- **DPRAM size and number of SyncManagers (SMs)**

The DPRAM is used for exchange of cyclic and acyclic data via the EtherCAT network. SyncManagers (SMs) ensure data consistency within the DPRAM. Each ESC has 4kB of registers (addresses 0x0000 to 0x0FFF) which are reserved for EtherCAT and PDI communication configuration settings.

Mailbox and process data is exchanged via additional DPRAM (also called user memory). EtherCAT allows addressing of user memory of up to 60 kB. ASICs provide between 1kB and 8kB of DPRAM, IP cores can be configured to provide the full 60 kB of user memory.

Application note: The standard SM configuration is

- 1 SM per acyclic data output (mailbox out to SubDevice application, MainDevice to SubDevice)
- 1 SM for acyclic data input (mailbox in from SubDevice application, SubDevice to MainDevice)
- 1 SM for cyclic data output (process data out, MainDevice to SubDevice)
- 1 SM for cyclic data input (process data in, SubDevice to MainDevice)

For process data, SM running in 3-buffer-mode need three times the length of actual process data for physical memory. The following Table 5 shows a schema of how to allocate the length for the 4 SM.

Table 5: DPRAM size calculation example

		Buffer Count	Length [Byte]	Total length [Byte]	
SM0	Output Mailbox	1	L_MbxOut		1*L_MbxOut
SM1	Input Mailbox	1	L_MbxIn	+	1*L_MbxIn
SM2	Outputs	3	L_Out (TxPDO)	+	3*L_Out
SM3	Inputs	3	L_In (RxPDO)	+	3*L_In
				∑	DPRAM size

SyncManagers are enabled by the following settings of the MainDevice during network initialization.

- Physical address of ESC
- Data length
- SyncManager control input:
  - Operation mode (mailbox-mode/3-buffer-mode)
  - Access direction (read direction/write direction)
  - Interrupt settings (valid/invalid)
  - SyncManager watchdog setting (valid/invalid)
  - SyncManager setting (valid/invalid)

The default values are set in the ESI (chapter 4.5.1); the MainDevice initializes the SM using the values from the ESI.

- **Number of Fieldbus Memory Management Units (FMMU)**

In an EtherCAT network, the memory of all SubDevices can be compiled in the MainDevice to a logical memory. This logical memory is managed by FMMUs to map logical addresses to physical addresses in the SubDevices. For the FMMU configuration in a device, each consistent output and each consistent input block needs one FMMU and an additional FMMU for mailbox status response is necessary.

Application note: The standard configuration is one FMMU per each, cyclic output and cyclic input data block, optionally an additional one for mapping the mailbox response availability flag into process data (thus, no polling of mailboxes is necessary). If the outputs and inputs are grouped e.g., like in Table 5, 3 FMMUs are configured, see Table 6.

Table 6: FMMU configuration

FMMU	Assigned SyncManager	Name	Length [Byte]
1	SM2	Outputs	L_Out (TxPDO)
2	SM3	Inputs	L_In (RxPDO)
3	SM0 & SM1	Mbx-SM Status flags	Mbx In/Out Length

- **Distributed Clocks (DCs) for synchronization with other SubDevices**

Evaluate if the device should support high precise synchronization with other SubDevices. If so, DCs should be supported by the selected ESC. Distributed Clocks refer to the DC function for EtherCAT SubDevices (chapter 3.6). The times held by SubDevices are adjusted with this mechanism and thus enable precise synchronization of the nodes in the EtherCAT network.

4.4.3 SII

The SII data is typically stored on an EEPROM device, which is mounted outside the ESC and connected via I^2C with point-to-point link. According to the size of the EEPROM the EEPROM_SIZE signal should be set. For more details, refer to the Knowledge Base, chapter “EEPROM".

For SII (EEPROM) Enhanced Link Detection setting, refer to documentation of the ESC vendor.

4.4.4 Application controller (host controller, µC)

If a local software application provides the device functionality, any 8- or 16-bit synchronous or asynchronous microcontroller can be connected to the ESC. The application controller communicates with the ESC via the Process Data Interfaces (PDI). Such devices are complex EtherCAT devices.

To adapt the application software on the application controller to the ESC, sample software stacks are available for communication implementation, e.g., the SubDevice Stack Code (SSC). If the device is a 32-bit digital I/O interface, it is called simple EtherCAT device and no application controller or additional communication software is necessary.

In most cases, manufacturers can use a familiar microcontroller type as application controller in the EtherCAT device. If application software already exists, e.g., for a different fieldbus, it can be used for the EtherCAT device as well.

The source code for communications software on the application controller allocates about 70-kB. The following features are a typical configuration (referring to the SubDevice Stack Code):

- EtherCAT State Machine (ESM), including error handling
- Device diagnosis
- MainDevice-SubDevice data synchronization with SyncManager event (no DCs)
- Mailbox CoE
- Object Dictionary (20 objects) for process data objects
- CoE SDO services, including SDO Information services, no segmented transfer

4.4.5 Application Layer communication protocols

In EtherCAT, several protocols are available (see chapter 3.7) for the application layer to implement the required specification of the product development. When to apply them is described here.

- **CAN application protocol over EtherCAT (CoE)**
  To provide acyclic data exchange as well as mechanisms to configure PDOs for cyclic data exchange in a structured way, CoE (with SDO Information support) should be implemented.
- **Servo drive profile over EtherCAT (SoE)**
  SoE is an alternative drive profile to the CiA402 drive profile. It is often used by drive manufacturers which are familiar with the SERCOS interface.
- **Ethernet over EtherCAT (EoE)**
  EoE is usually used to provide webserver interfaces via EtherCAT. It is also used for devices providing decentral standard Ethernet ports.

- **File Access over EtherCAT (FoE)**
  If the device should support firmware download via EtherCAT, FoE should be supported. FoE is based on TFTP. It provides fast file transfer and small protocol implementation.
- **ADS over EtherCAT (AoE)**
  AoE is specified to be used for use cases as they appear e.g., for fieldbus gateways: fieldbus SubDevices behind the gateway, including their type of object dictionary can be accessed using AoE. It is routable and allows parallel requests to fieldbus SubDevices behind the gateway. AoE does not provide a semantic concept (data types, structure, etc.) as CoE does.

Application note: An exemplary CoE implementation is shown below in Figure 13.

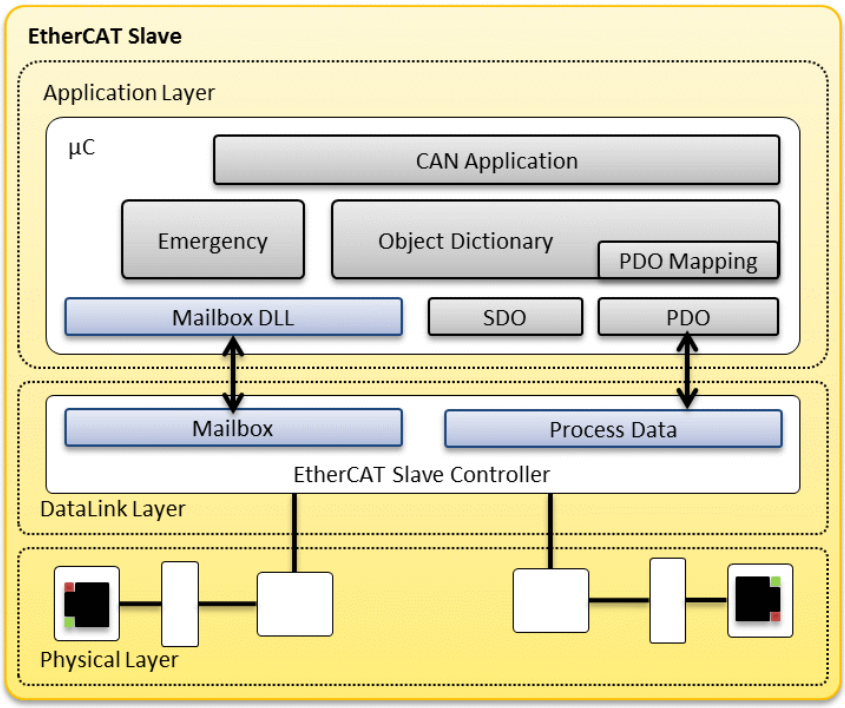


Figure 13: ESC structure for CAN application profile applications

The user application runs the device-specific software on the application controller to implement device features. SubDevice source codes offered by EtherCAT stack vendors can be found in the EtherCAT Product section (text filter: *SubDevice Stack*) and used to develop this application or to adapt existing software to EtherCAT.

Application note: EtherCAT SubDevice Stack Code (SSC).

The SSC is a free sample code from Beckhoff which provides an interface to the ESC. For hardware independent software development, the SSC runs on several evaluation kits and can be customized for implementation in accordance with the product specification. Figure 14 shows the SSC structure with the interfaces to the-user specific device application and the ESC.

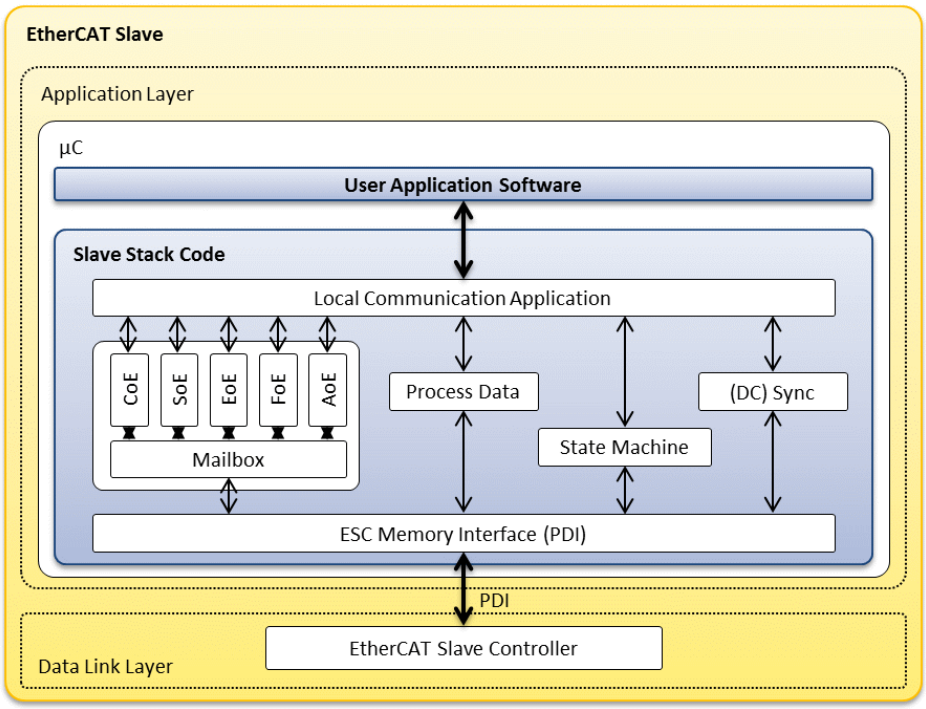


Figure 14: SubDevice Stack Code overview

Application note: EtherCAT SubDevice protocol stack.

Hilscher or HMS offer a SubDevice control stack based on its netX or AnyBus hardware with Dual-Port Memory interface (DPM) and it is available for the user application with an API. Figure 15 shows the protocol stack architecture with interfaces to the ESC and the user application.

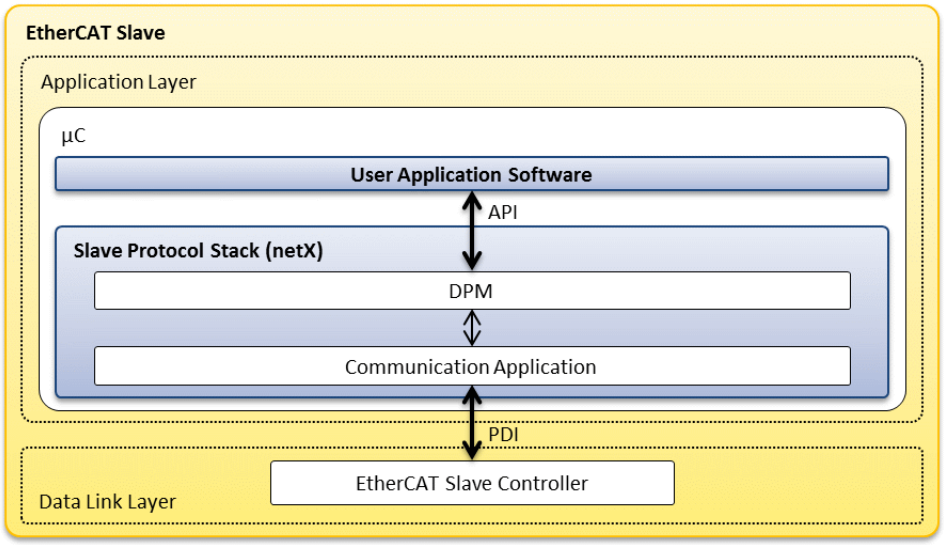


Figure 15: SubDevice control stack

A list of other available sample stacks can be obtained on the official EtherCAT Product Section of the ETG website with the filter option as shown in Figure 16.

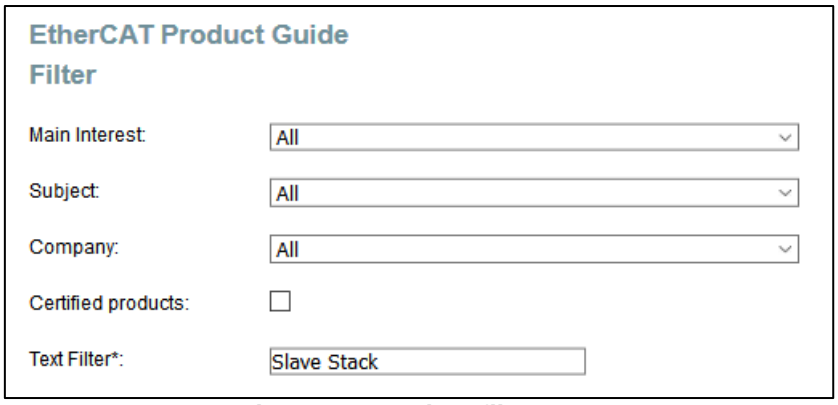


Figure 16: product filter SSC

4.4.6 Device profiles

Device profiles define a common application interface for specific devices. Drive profiles define the identifier (CoE index) and data type for the status word and control word, for set point and current value, and standard parameters. The MainDevice application such as a PLC program, can then use the same data structure for drives of different vendors.

The EtherCAT specifications define different profiles. Some examples are:

- Drive profile
  The drive profiles are according to IEC 61800-7 and include the mapping of the CiA406 drive profile to EtherCAT and the Sercos™ Drive Profile to EtherCAT. The ETG.6010 Implementation Directive describes more details and helps understanding the CiA402 drive profile on EtherCAT.
- Modular Device Profile (MDP)
  The MDP provides general rules for the structuring of the CoE object dictionary. Device specific profiles are available, too, such as for fieldbus gateways. It is specified in ETG.5001.
- Specific Device Profile (SDP)
  It defines a series of profiles for different devices used in the semiconductor manufacturing industry, mainly focusing on the process part, such as pumps, valves, mass flow controllers, temperature controllers. It is specified in ETG.5003 and its basic structure is very close to ETG.5001.

The object dictionary can be described as a two dimensional list. Each list entry is identified by an index (0x0000 – 0xFFFF) which represents an object. Each object can contain up to 255 subindices, also called object entries. The object list is structured in different areas, see Table 7.

Table 7: The Modular Device Profile Object Dictionary

Object index range	Reserved for		Comment
0x0000 – 0x0FFF	Data type area		Protected registers for ESC configuration
0x1000 – 0x1FFF	Communication area		Communication parameters, settings, etc.
0x2000 – 0x5FFF	Manufacturer-specific area		
0x6000 – 0x6FFF	Profile-Specific area	Input area	Process data input objects (mapped to TxPDOs)
0x7000 – 0x7FFF		Output area	Process data output objects (mapped to RxPDOs)
0x8000 – 0x8FFF		Configuration area	Process data configuration and settings objects
0x9000 – 0x9FFF		Information area	Scanned information from modules
0xA000 – 0xAFFF		Diagnosis area	Diagnostic, status, statistic or other information
0xB000 – 0xBFFF		Service Transfer area	Service objects

Object index range	Reserved for		Comment
0xC000 – 0xEFFF		Reserved area	
0xF000 – 0xFFFF		Device area	Parameters belonging to the device

The idea of the MDP is to provide a basic structure for MainDevices and configuration tools to handle SubDevices with complex (modular) structure easily. The user has the advantage, that if the SubDevice's variables are sorted in an MDP style, so he can find the different data types by identical patterns.

The MDP can be applied to various types of devices. It is applicable to multiple axis servo drive system of various functionality groups, such as positioning, torque and velocity control. It is further applicable to gateways between different fieldbuses, i.e., Profibus, DeviceNet. Modular devices are driven by two aspects:

- **Comprise physically connectable modules and plurality of functionalities.**

The MDP imagines SubDevices which consist of one or several modules. A module can be hardware which is connected/disconnected to a SubDevice. Examples are gateways between EtherCAT and e.g., CANopen or a bus coupler between EtherCAT and a proprietary backbone bus.

- **Comprise plurality of channels directly being connected to the EtherCAT network.**

A module can also be a logical module which describes data sets, e.g., a drive which supports a velocity controlled mode and a position controlled mode – the MDP would describe the data as two modules, one for each mode.

No matter what kind of module is described it needs more or less the same information categories, which are organized in the profile specific index range (Table 7).

Application note: Modular Device Profile structure.

Consider an MDP for a line of SubDevice modules which are connected together on a backbone layer via a coupler with MII. Figure 17 shows a schema how to define device profiles such that a modular profile dictionary is set up for the SubDevice line.

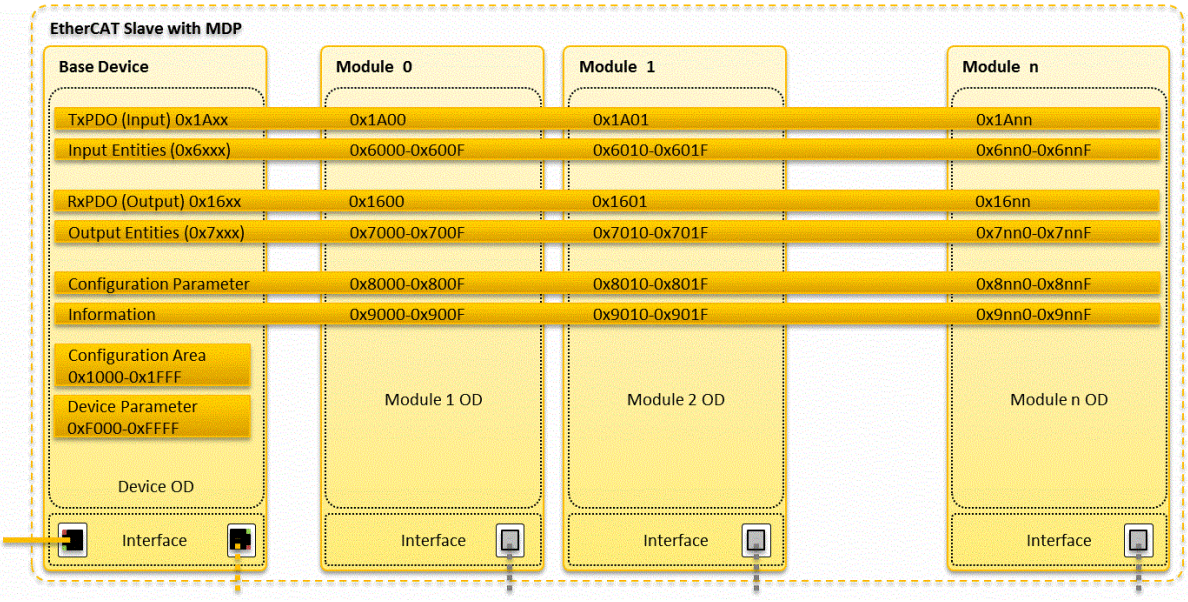


Figure 17: MDP schema for modular devices

4.4.7 Synchronization among SubDevices and the MainDevice

EtherCAT provides various synchronization options. There are three different types of synchronization methods available.

- **Freerun**

The SubDevice application runs independently of the EtherCAT cycle and is triggered by a local timer in the ESC.

- **Synchronous with frame reception (synchronization with SM event)**

The SubDevice application is triggered when new process data is received. The synchronization accuracy depends on the jitter of the message reception and the delay between the other network nodes.

- **Distributed Clocks (DC, synchronization with SYNC0/SYNC1 event)**

The ESCs contain a nanosecond based timer (DC timer) to provide high precise synchronization and time stamping. The SubDevice application is triggered with an additional interrupt signal, which is based on the DC time and is produced by the ESC. Every DC timer in the network is aligned to a reference DC clock.

Application note: The ESC system time is specified as a 64 bit value. This data size allows representation of more than 500 years. The latter 32 bits represent approximately 4.2 seconds. Refer to the datasheet of the specific ESC for details since some ESC implement only 32 bit length.

Initial value: 00:00:00 January 1, 2001
Unit: 1ns

Definition of a reference clock

One EtherCAT SubDevice (which usually is the first SubDevice that uses DC) is determined as the reference clock and becomes the clock base for the MainDevice as well as for other DC SubDevices. The reference clock is periodically provided to other SubDevices. The reference clock is adjustable by an external "global reference clock".

Function and operation of DC

The SubDevice synchronization is established during initialization of the ENI in the MainDevice. With EtherCAT, the 3 DC time synchronization functions enable high precise synchronization.

- **Measurement/calculation of the propagation delay time**

  During initialization procedure of the network, the MainDevice calculates the propagation delay, including the delay caused by cables and ESC, and sets the delay as SubDevice delay. The delay calculation algorithm is basically defined the ETG.1000.4 and further described e.g., in the ET1100 Datasheet (section I, chapter 9.1.2). After establishment of the SubDevice DC, the EtherCAT MainDevice periodically sends Auto increment Read Multiple Write (ARMW) commands to read the time from the reference clock and write it to all other DC SubDevices.

- **Drift compensation**

  The MainDevice periodically reads out the time from the reference clock using the ARMW and writes the time to the other DC SubDevices. The deviation of time data held by the SubDevice is thus minimized.

- **Offset compensation**

  Offset compensation refers to function of adjusting the system time (e.g., the calendar time) held by the EtherCAT MainDevice and the time held by SubDevice. The SubDevice can be synchronized by the EtherCAT MainDevice by writing into the SubDevice the deviation of time between the system time of the MainDevice and the reference clock.

Interrupt signal

After establishment of DC by the MainDevice, the ESC generates fixed time interrupt signals to the PDI, i.e. the application controller. Thus, the SubDevice is able to create a constant period. There are the following 3 types of generation of interrupt signals.

- SYNC/LATCH0
- SYNC/LATCH1

- IRQ (interrupt occurs by generation of SYNC0/SYNC1and mask register setting)

Note that the SYNC0/SYNC1 interrupt signals cannot be used when using the ESC LATCH0/LATCH1 function. This restriction is due to SYNC/LATCH signal lines being a shared pin.

The latch function is a function which maintains time stamp in response to latch signal input on the ESC, activate/deactivate timing edges can be set.

4.4.8 Firmware update

The EtherCAT specifications defines the FoE mailbox application protocol for firmware update in Bootstrap mode of the ESM. In addition, it defines FoE error codes and AL Status Codes which can be used to report certain errors which may occur during a firmware update procedure. However, there is no more specific description of the firmware update process as in the Semi Device Profile Specification of ETG.5003 part 2, “Firmware Update Specification”. EtherCAT devices supporting the “Semi Device Profile” (CoE Object 0x1000 = 5003dec) must also support the firmware update mechanisms defined in part 2. Even though it is not mandatory for any other EtherCAT SubDevice to do so, this part provides a good guideline for a firmware update implementation on any EtherCAT SubDevice. Some details covered by this description are:

- SubDevice accessibility in case of failed FW update
- ESC reset behavior
- Device documentation
- SII update
- FW version and functionality verification

Therefore, it is recommended to use ETG.5003 as a guideline for firmware update implementation.

4.5 Tools for EtherCAT SubDevice development

Table 8 lists tools that may be useful for EtherCAT device development. Some tools are described in more detail with their application purpose in the following subsections.

Note the Conformance Test Tool is mandatory for SubDevice vendors.

Table 8: Tools

	Tool	Description and access
Network configuration	EtherCAT configurator	Configurator for loading XML device descriptions (ESI) and for generating XML network configuration descriptions (ENI). Several EtherCAT MainDevices already include an EtherCAT configuration tool. Visit the official EtherCAT Product Section of the ETG website for the variety of configuration tools. Main Interest: Development Systems, Tools Subject: Configuration Tools For development purposes, an EtherCAT configuration tool with MainDevice (e.g., TwinCAT can be downloaded as free 7-day trail version from the beckhoff website)
Development	XML editor	Used to edit or view EtherCAT SubDevice Information (ESI) files. Any browser or text editor can be used, as well as the CTT. Further tools: Altova XML Spy (extensive xml editor, license fee required) XML Notepad (freeware)
	Hex file editor	Used to convert bitmap images (vendor or device logos) to a hex value which is needed in the ESI. Any hex editor is fine, here are two examples: HxD (freeware) Mirkes TinyHexer (freeware)
Diagnosis	Network Monitor	Wireshark (former Ethereal) can be used to monitor frame communication of EtherCAT networks. Wireshark is freeware and has already included a parser for comfortable EtherCAT frame analysis.

	Tool	Description and access
		Available for Linux and Windows
	EtherCAT Conformance Test Tool (CTT)	The CTT is used to check EtherCAT protocol compliance in-house. The test tool is provided by Beckhoff Automation GmbH & Co. KG. Contact ctt@beckhoff.com
	Further Tools	Also consult the official EtherCAT Product Section of the ETG website for a continuative list of tools.

4.5.1 XML editor for generating ESI files

The vendor needs to deliver the device with an ESI file. During the Design of an EtherCAT network, the user needs to generate a ENI file using a configuration tool based on the ESI files of the SubDevices in the network. SubDevice specific information (manufacturer, product information, profile, object, process data, sync or non-sync, SM setting) is registered in the ESI file in XML format. A single ESI file may include the information of multiple EtherCAT SubDevices.

The ESI file is defined with the ETG.2000 EtherCAT SubDevice Information specification. The structure of an ESI file is defined in the EtherCATInfo.xsd XML schema document, see Figure 18. By applying the XML schema to an XML editor, syntax checks can be made on the ESI description to avoid basic errors. The XML schema as well as a sample ESI file are available from ETG.2001 EtherCAT SubDevice Information Annotations.

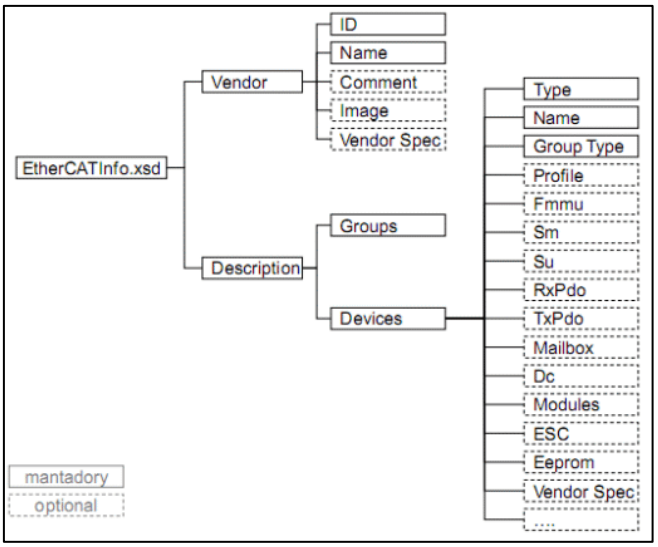


Figure 18: ESI structure (EtherCATInfo.xsd)

A text editor or (graphical) XML editor software may be used to edit the ESI file. The CTT also provides an editing environment for ESI files as shown below in Figure 19.

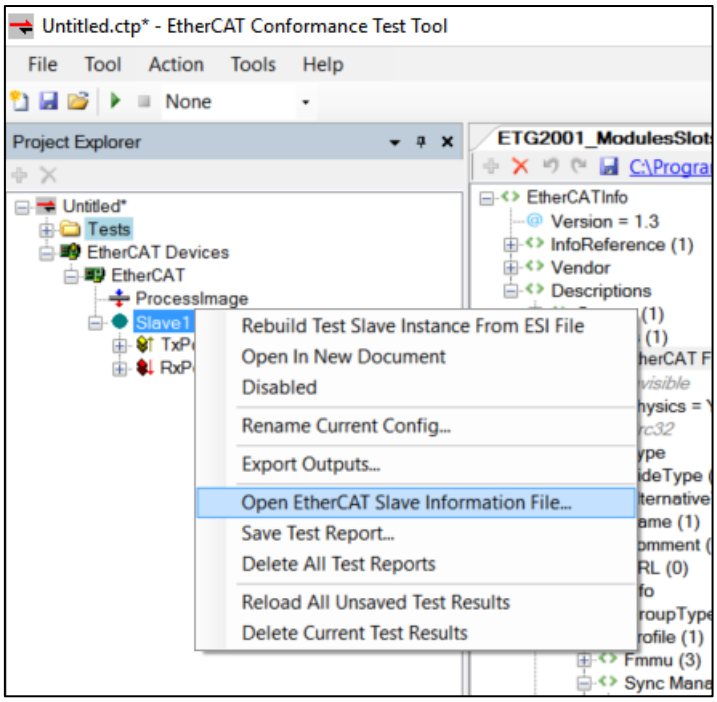


Figure 19: ESI file editing using CTT

Any other popular editor software can also be used for XML editing, e.g., Altova XML Spy (Figure 20).

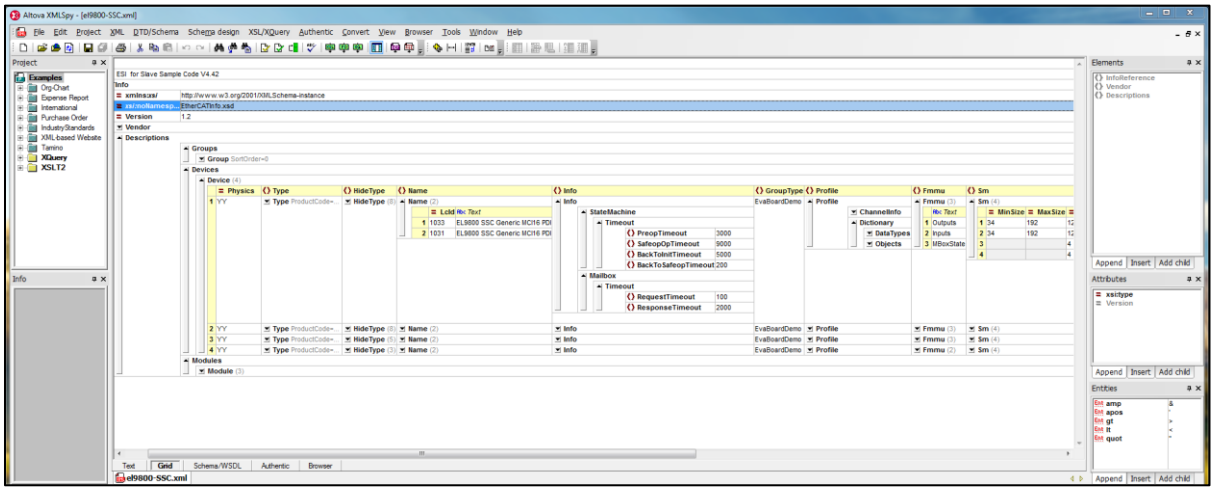


Figure 20: ESI file generation using a graphical editor (Altova XML Spy®)

4.5.2 EtherCAT network configurator and MainDevice software

For EtherCAT network configuration, an EtherCAT network configurator is necessary which loads ESI files and generates an ENI file. Available software can be found on the EtherCAT Product section of the ETG website. For example, TwinCAT has a built-in network configurator and is also available as free 7-day trial version (Figure 21).

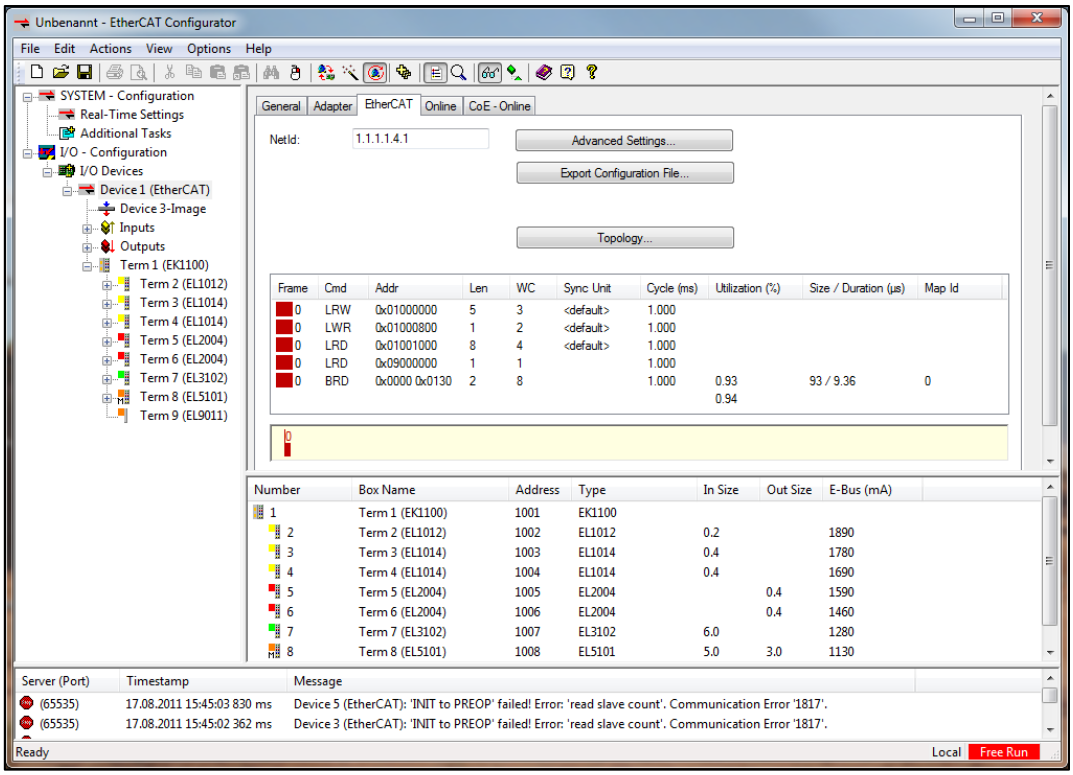


Figure 21: EtherCAT network configurator

Software for a MainDevice becomes necessary when running an EtherCAT network or debugging a SubDevice. The ESI file of the developing SubDevice needs to be stored in the MainDevices EtherCAT device repository. To set up a small EtherCAT network with a MainDevice and a SubDevice, refer to chapter 2.

A list of available MainDevices can be found on the EtherCAT Product section (text filter: MainDevice) of the ETG website. For example, TwinCAT from the Beckhoff Automation is available as free 7-day trial version. In TwinCAT System Manager, right click on I/O Device, Scan Devices and further Scan Boxes (see Figure 22). Refer to the TwinCAT manual for the subsequent steps to assemble an EtherCAT network.

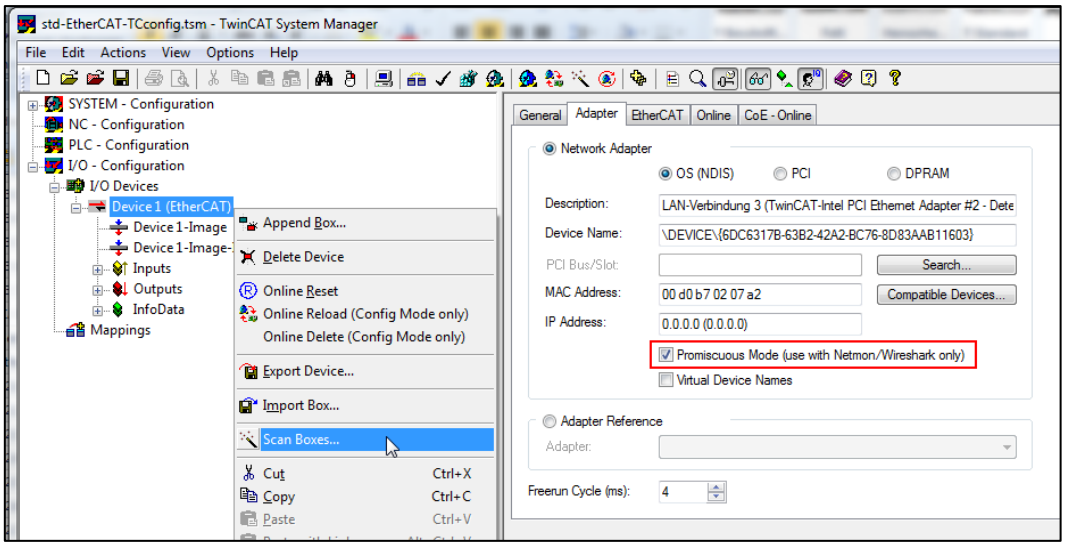


Figure 22: TwinCAT device scan, box scan and adapter settings

4.5.3 Monitoring communication and network diagnosis using Wireshark

In order to verify EtherCAT communication data, EtherCAT frames can be decrypted by a frame analyzing software such as Wireshark. Wireshark traces can be taken either on the EtherCAT MainDevice or via a real-time Ethernet probe. The promiscuous mode needs to be activated (see

Figure 22) to record EtherCAT frames on a TwinCAT MainDevice. The content of EtherCAT frames is displayed by Wireshark as shown below in Figure 23.

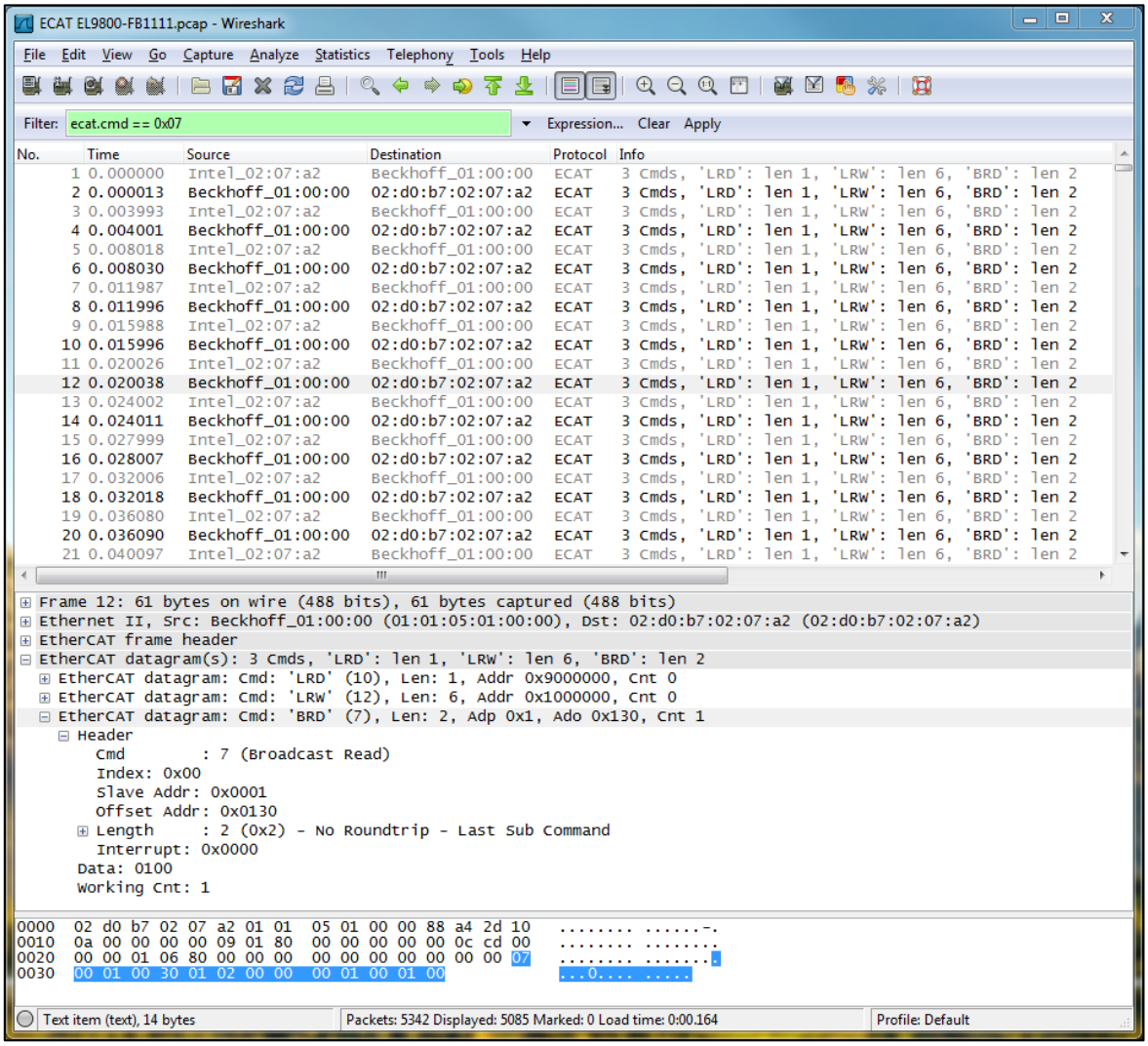


Figure 23: Wireshark Screenshot

The EtherCAT dissector is already included in Wireshark.

Further detailed information about capturing, filtering and handling of Wireshark scans can be found within the EtherCAT Knowledge Base.

4.5.4 EtherCAT Conformance Test Tool

Besides of basic software and hardware debugging, in-house EtherCAT conformance testing is mandatory to verify that the device meets the EtherCAT communication requirements. Meeting this requirement is a minimum condition to sell the product as an EtherCAT compatible product. In-house EtherCAT conformance testing is done with the EtherCAT Conformance Test Tool (CTT).

Application note: To build a conformance test environment, the following items should be prepared.

- Windows PC and network card (100 Mb, full duplex and auto negotiation must be supported)
- In case the CU2508 is used a 1 GB/s network card is required
- CTT
- Download software via www.ethercat.org/cttdownload
- Get license subscription from Beckhoff (product name is ET9400) (see chapter 4.3.3)
- NOTE: Download and install the latest CTT version. The CTT is updated periodically
- Device under Test (DuT)
- EtherCAT SubDevice Information (ESI) file
- Packet analyzing software (e.g., Wireshark)
  A network probe might be useful when DuT support DCs

- Real-time hardware extension
- The CTT runs on the Windows OS, which provides very limited real-time capabilities. To make real-time testing possible (e.g., for DuTs supporting DC) the hardware CU2508 has to be used.

The ETG.7000.2 Conformance Test Record is a guideline for testing. Basically, proceed as follows.

- Install the CTT on the Windows PC
- Copy the ESI to the device descriptions folder in the local installation folder of the CTT
- Link the device to the Windows PC, start CTT and scan for the device to load it into the CTT
- Perform the tests provided by the CTT
- Update firmware, ESI, SII and everything else until all errors are gone. The CTT test logs help to understand where updates are necessary; see Figure 24 and the CTT documentation (help file).

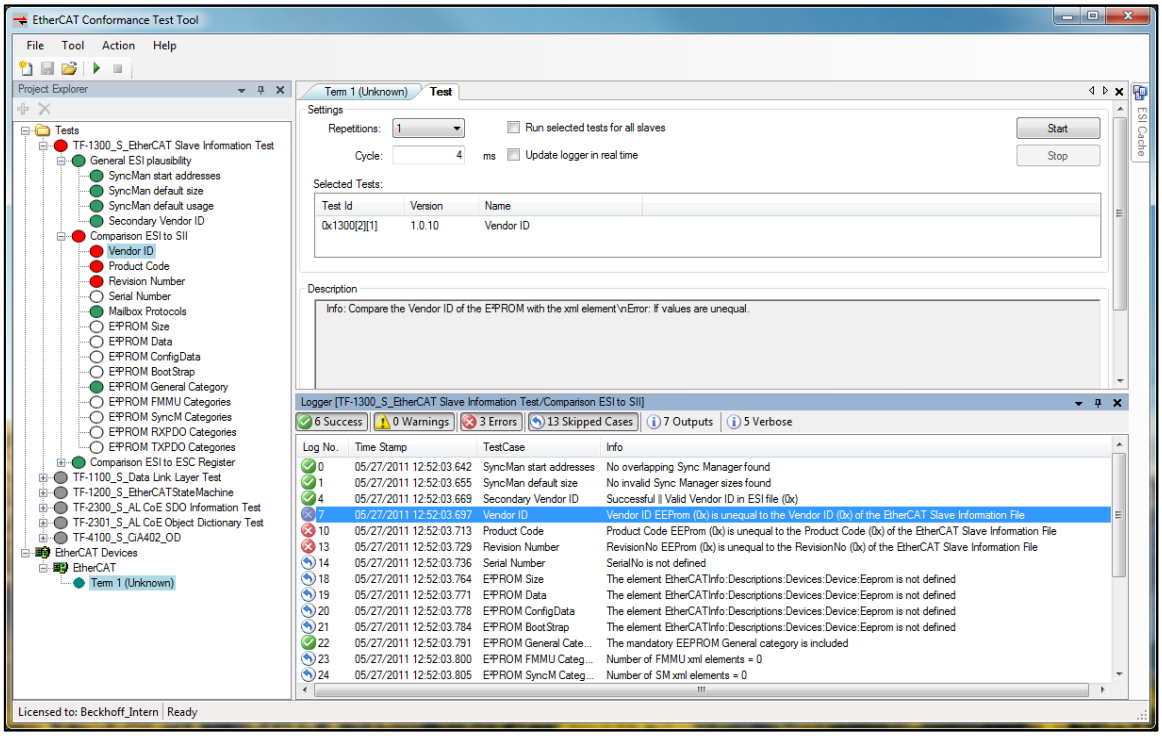


Figure 24: Testing with the Conformance Test Tool

Conformance and interoperability are very important factors for the success of a communication technology. Conformance of the technology implementation with the specifications is the pre-requisite of interoperability, which means that devices of different manufacturers co-operate in the same networked application.

The conformance testing rules and policies according to the Vendor ID agreement are covered by the ETG.7000 Conformance Test Policy, available on the ETG website.

4.6 EtherCAT product labels and LEDs

It is recommended to consider the LEDs, device identification and labeling (e.g., of the ports) during the devices hardware design. This is subject of the ETG.9001 Marking Rules and the ETG.1300 Indicator and Labelling Specification.

EtherCAT obligates various elements for indication. Such indication should be made as markings on the surface of an EtherCAT SubDevice box. The marking requirements are also the subject elements of the EtherCAT Conformance Test (ETG.7000.2 Conformance Test Record).

Activity of EtherCAT devices is indicated by LEDs, which indicates:

- Current state of the state machine: Init, Pre-Operational, Safe-Operational, Operational, (RUN LED)
- Error code (ERR LED)
- Link/Activity of all ports (L/A LED)

Application note: Referring to the ETG.1300 Indicator and Labelling Specification, the LEDs must work as shown in the following Table 9.

Table 9: RUN and ERR LED indications

RUN LED	EtherCAT state	ERR LED	EtherCAT state
Off	Init	Off	No Error
Blinking	Pre-Operational	Blinking	Invalid configuration
Single Flash	Safe-Operational	Single Flash	Unsolicited state change
		Double Flash	Application watchdog timeout
Flashes	none (Initialisation) or Bootstrap	Flickering	Booting error
On	Operational	On	PDI watchdog timeout

Application note: EtherCAT branding. At least one of the following EtherCAT logos (Figure 25) should show on the product or instruction manual:

Figure 25: EtherCAT product branding logos

The following English declaration of the EtherCAT trademark must appear in the instruction manual:

"EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH & Co. KG, Germany."

Application note: Requirements for port labels and L/A LED indication derived from the ETG.1300 Indicator and Labelling Specification (Table 10).

Table 10: Port and L/A LED Label Requirements

Label type	Requirements
IN port label	Must be placed near the port. The label should be clearly allocated to the subject port. The characters on the label should be one of the following "IN" or "ECAT IN" (Capitals and small characters both permitted).
OUT port label	Must be placed near the port. The label should be clearly allocated to the subject port. The characters on the label should be one of the following "OUT" or "ECAT Out" (Capitals and small characters both permitted).
L/A LED label	Preferably the print characters should be placed directly next to the network interface but is not compulsory. The mark can be placed on other location or can be omitted. The print characters, if not omitted, should show one of the following phrases. " ", "Link/Act" or "Link/Activity" (Capitals and small characters both permitted). Label is required on removable connectors.

5 EtherCAT Conformance Testing

5.1 EtherCAT Conformance Test Tool

The in-house test with the EtherCAT Conformance Test Tool (CTT) is mandatory. For the CTT description see chapter 4.5.4.

For the CTT subscription and licensing see chapter 4.3.3.

5.2 Official EtherCAT Conformance Test at an EtherCAT Test Center (ETC)

The procedure is described in detail in the ETG.7010. Following is an overview to the procedure of an official EtherCAT Conformance Test according to the ETG.9003.

1 Fill out the ETG.7030 Conformance Test Request form. Send the request form to conformance@ethercat.org

- When the test request is received, the ETC denoted in the test request starts arranging the test schedule and sends the test contract.
- Return the signed contract by e-mail or fax. The test fee invoice will not be issued unless the test contract is submitted.
- Send out referenced test material. A device check list assists a reference for material which is to send to the ETC a week before the test.
- Preparation of components to deliver. Ensure that all equipment is delivered to the ETC before the test date. It is NOT possible to deliver any missing items afterwards.

2 Test execution according to the ETG.7000.2 Conformance Test Record.
If preferred to attend the test in-person, ensure to have a meeting arranged with the ETC.

3 By successfully passing the EtherCAT Conformance Test, a pass notice is issued by the ETG Headquarters. The “EtherCAT Conformance Tested” certificate will then be issued and sent to the device vendor.

6 EtherCAT Development Support

6.1 EtherCAT training and workshop

Different EtherCAT trainings are available. A good reference for trainings can be found at http://www.ethercat.org/events > Training/Workshop. Some of the trainings which have been offered for many years are listed in Table 11.

Table 11: EtherCAT training and workshops

Training/ Workshop	Description	Reference
EtherCAT Technology Basics	This is the training to get started with EtherCAT, developer of a SubDevice, MainDevice, or configuration tool logic, but also as advanced EtherCAT machine application developer or service engineer. During this training the EtherCAT protocol is explained in detail, starting at a system point of view, and then explaining topology, Ethernet hardware incl. PHYS, all functionality processed by an ESC (EtherCAT SubDevice Controller) on the Data Link Layer and the Application Layer protocol and services incl. the EtherCAT State Machine, mailbox application protocols such as CoE, FoE, EoE and synchronization modes based on Distributed Clocks (DC). Conformance testing, where to find references and support and more is explained. This training is about one day in total, either in-person or online as two half days.	www.ethercat.org → Events (Beckhoff, TR8110)
FSoE Technology Basics	The Safety over EtherCAT Seminar gives you a comprehensive overview about today's requirements for safety machine architectures with the focus on safety communication with the Safety over EtherCAT (FSoE; Fail-Safe over EtherCAT) protocol. The FSoE highlights and technical features are shown and implementation aspects are provided. Decision makers, product manager as well as R&D engineers from ETG members who are involved in their companies safety product strategy are invited to this seminar. This training is about half a day.	www.ethercat.org → Events
EtherCAT Configuration and MainDevice Basics	Content: • Reference to ETG.1500 "EtherCAT Master Classes" • Requirements and interfaces to MainDevices and configuration tools • Offline/online network configuration: ESI, ENI and SII • Topology detection and monitoring • Distributed Clocks: configuration and monitoring • Handling and monitoring of the EtherCAT State Machine (ESM) • Network initialization: Init Commands and CoE start-up commands • Sending and receiving cyclic data: process image, logical addressing • Process Data configuration: CoE objects, start-up commands, ESI flags • Mailbox communication: protocols and monitoring mechanisms • Dependencies between Object Dictionary, ESI and SII • Diagnostic functionalities and their implementation • One day, as two half day sessions.	www.ethercat.org → Events (Beckhoff, TR8210)
Workshop: EtherCAT Evaluation Kit and SubDevice Stack Code (SSC)	The online training is aimed at developers of EtherCAT SubDevices using the EtherCAT SubDevice Evaluation Kit (EL98xx) and SubDevice Stack Code (SSC) from Beckhoff Automation. In addition to theoretical content, they also include practical exercises. Basic EtherCAT knowledge is assumed. The workshop is led by developers and held in manageable groups so that individual interests can be addressed.	www.ethercat.org → Events (Beckhoff, TR8100)
Workshop: EtherCAT MainDevice Sample Code	This training addresses developers and EtherCAT users alike. A deep understanding of the EtherCAT configuration and network operation is provided. Basic EtherCAT knowledge is assumed. The workshop is led by developers and held in manageable groups so that individual interests can be addressed. Offered on request	(Beckhoff, TR8200)

Both, workshops and training classes have proven to put the developer in a good starting position with a well-established understanding of the EtherCAT protocol, tools, development hardware and software including the SubDevice Sample Code as a basis to build the vendor-specific application on top.

6.2 EtherCAT development support tips

When having questions or problems with EtherCAT device development, feel free to engage individual support provided by the EtherCAT Technology Group (for contact, see chapter 7.1).

To optimize support processes, the following instructions lead to faster response time and improve support quality. Basically, explain the issue as detailed as necessary but as simple as possible.

- Which system architecture are you using?
- Hardware components: ESC, application controller, etc.
- Software components (& versions): SubDevice stack, MainDevice solution, etc.
- Infrastructure: topology, (self-made) cables, etc.
- Problem report:
  - What: Can you shortly describe the behavior?
  - When: Is the problem reproducible?
  - Where: Can you locate the problem?
  - What was already tested?
- Additional information:
  - MainDevice configuration file, in suitable format
    - E.g., *.tsm file (TwinCAT 2) or solution (TwinCAT 3)
    - ESI files of involved device(s)
  - Anything else that helps to process the issue:
    - screenshots
    - log files
    - In case of conformance testing, the Conformance Test Tool project file (*.ctp) with saved results
    - Wireshark scan (*.pcap or *.pcapng format) capturing the problem.
      To focus the scan on necessary content, follow these instructions:
      - Connect the smallest number of devices enabling to reproduce the problem
      - Capture the network start-up phase, either in the same or a separate capture
      - When using an Ethernet probe, report where exactly the probe was connected

7 EtherCAT Technology Group – events and support

The EtherCAT Technology Group (ETG) is the forum where key user companies from various industries and leading automation suppliers join forces to support, promote and advance the EtherCAT technology.

7.1 Basic information about the ETG

Goals

EtherCAT is an open technology. The ETG stands for this approach and ensures that every interested company may implement and use EtherCAT.

At the same time the ETG aims to ensure the compatibility of EtherCAT implementations by defining functional requirements, conformance tests as well as certification procedures.

The ETGs goal is to ensure that EtherCAT technology meets and exceeds the requirements of the widest possible application range. To accomplish this goal, the group combines leading control and application experts from machine builders, system integrators, end users and automation suppliers to provide both qualified feedback about application of the existing technology and proposals for future extensions of the specification.

The ETG organizes user and vendor meetings in which the latest EtherCAT developments are reviewed and discussed in regular periodical sessions.

Benefits for ETG members

ETG members get preferred access to specifications, specification drafts, white papers, prototype evaluation products and initial batch products and thus have a head start in evaluating, using, or implementing EtherCAT technology.

The members are eligible to participate in ETG Technical Working Groups (TWG) and thus have influence on future enhancements of the EtherCAT technology specifications, like safety, conformance, and much more. A closer look to all available TWGs is provided in the working group area on the ETG website.

The member companies may use the EtherCAT and the ETG logos to show their support for this technology.

How to join the ETG

If you are interested in becoming a member of the ETG, contact the ETG Headquarters for further information regarding membership request (see contact section below).

Membership costs

The membership is free of charge, thus there are no annual membership fees. According the ETG by-laws a membership fee can only be introduced if the membership assembly decides so.

Technical support

Technical support throughout the development process is provided by the ETG predominately by the headquarters in Germany, but also by the various ETG offices worldwide (depending on local capacity). If you need direct contact, address your specific question to the ETG.

Before contacting ETG for support, we expect reading the mentioned documentation above as well as the recently listed information above (see chapter 6). We strongly recommend visiting one of the EtherCAT workshops and/or seminars for developers when starting an EtherCAT implementation.

Also a good opportunity to ask for technical experience with EtherCAT and for technical questions is provided by the EtherCAT Forum and the EtherCAT Knowledge Base within the member section of the EtherCAT website.

Contact

ETG Headquarters
Email: info@ethercat.org
URL: www.ethercat.org

ETG Office North America
Email: info.na@ethercat.org

ETG Office China
Email: info@ethercat.org.cn

ETG Office Japan
Email: info.jp@ethercat.org

ETG Office Korea
Email: info.kr@ethercat.org

7.2 EtherCAT Plug Fests

Depending on the demand of ETG companies, EtherCAT Plug Fests are held several times a year at venues all over the globe. Every ETG member developing devices or tools with at least a functional prototype are allowed to attend. In practical tests interoperability and the latest features of the devices are tested and the CTT is applied. Qualified feedback of EtherCAT specialists is provided.

Dates are published on the Event Section of the ETG website. An additional invitation email is automatically sent to the ETG representatives of the ETG member companies.

Participation at EtherCAT Plug Fests is free of charge. Attendees are not entitled to publish or communicate test results of other participating companies.

7.3 Official EtherCAT Conformance Test certificate

An official EtherCAT Conformance Test is an option after successful in-house testing. With passing the EtherCAT Conformance Test successfully a “Conformance Tested” certificate is issued and thus, the vendor may label his device with the official conformance test mark (Figure 26) and use the term for advertisement for the certified device exclusively.

EtherCAT Conformance tested EtherCAT Conformance tested

Figure 26: EtherCAT Conformance Tested logos

To apply for the EtherCAT Conformance Test at any EtherCAT Test Centre (ETC) send an e-mail to conformance@ethercat.org to ask for further information and the request form. On return of the request form to the ETG the requested ETC will contact you for further steps (see chapter 5.1).

The Conformance Guide explains the most important details on the topic and gives advice for preparation of the conformance test.

There are two officially accredited test centers, one in Nuremberg, Germany, and one at ASTEM in Kyoto, Japan. The ETCs do not only perform the official conformance test, but also provide qualified feedback and implementation support for ETG members.

The official test performed by an ETC is referred to as EtherCAT Conformance Test which is regarded as higher-level test above all other tests performed individually by the users (with the CTT) since interoperability and physical layer tests are covered as well.

When successfully passed the EtherCAT Conformance Test at an ETC, a notice is given to the ETG Headquarter. An EtherCAT Conformance Tested certificate is then issued free of charge and sent to the device vendor.

The test fees are ETC-business. ETCs will provide an over, usually upon receiving the test request (ETG.703x) from ETG.

7.4 ETG Technical Committee

The Technical Committee (TC) serves as central technical board. It establishes working groups, task forces and receives their reports. Other duties of the TC are to inform about enhancements of the EtherCAT technology, progress on standardization and to discuss current technical issues with the attending ETG members.

Dates are published on the Event Section of the ETG website. An additional invitation email is automatically sent to the ETG representatives of the ETG member companies.

Participation at the TCs is free of charge.

7.5 Information and support

7.5.1 EtherCAT Compendium

EtherCAT Compendium This is the EtherCAT read - from getting started to understanding the functionalities themselves, their purpose, and the models behind. It describes the protocol as a whole and puts it into context in a very comprehensive and easy-to-read style. Those are the big add-ons compared to the very precise specification.

→ http://www.ethercat.org/compendium

7.5.2 Download area on the web site

There are heaps of information available within the download area of the EtherCAT web site at www.ethercat.org/downloads. Take advantage of the filter options, too, like shown within Figure 27:

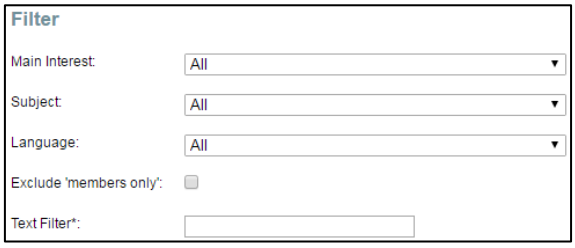


Figure 27: ETG download section - filter options

Furthermore, the filter can be set by using URL-parameter "*?tf=*" directly, for example:

- http://www.ethercat.org/en/downloads.html?tf=diagnosis
- http://www.ethercat.org/en/downloads.html?tf=safety
- http://www.ethercat.org/en/downloads.html?tf=conformance

7.5.3 Knowledge Base

As one of the main sources to complement the EtherCAT specifications the Knowledge Base (www.ethercat.org/kb) provides:

- Glossary:
  description of EtherCAT terms including references to other related readings
- Hands-on how-to descriptions:
  description of how to e.g., make a network scan, test CoE communication and many other things
- Detailed descriptions:
  elaborating the specifications where necessary
- FAQs:
  answers to frequently asked questions

The Knowledge Base is continuously extended based on the questions we receive at the ETG team – so make it a habit to check on the Knowledge Base first. Also, if you are missing any information, help us with your input on possible new entries on the Knowledge Base and send it to info@ethercat.org.

7.5.4 Developers Forum

On the EtherCAT Developers Forum, every ETG member is invited to discuss the EtherCAT technology and to post own questions. Many practical questions are already answered in the following forum topics:

- EtherCAT Specifications
  - Proposals
- Implementing EtherCAT
  - MainDevice and SubDevice Device
  - Evaluation Kit Hardware and Software
- EtherCAT SubDevice Conformance Test
  - Test Cases
  - SubDevice Conformance Test Tool
- Safety over EtherCAT (FSoE)
  - FSoE implementation
- EtherCAT Technology Group
  - ETG Services
  - New Downloads
- EtherCAT.org Website
  - Suggestions for improvements and comments
  - Knowledge Base

7.5.5 Search the EtherCAT web site

A search field is always accessible when surfing the EtherCAT web site in the upper left corner or via www.ethercat.org/search, like shown within Figure 28.

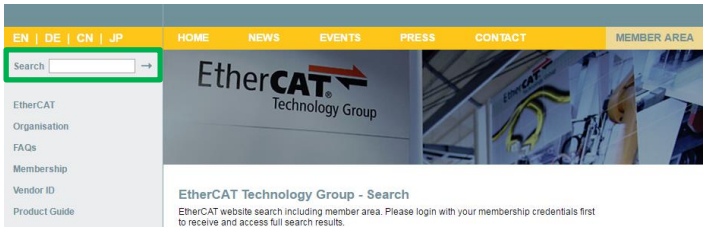


Figure 28: ETG webpage search

7.5.6 Technical support

When having questions during your EtherCAT device development and you just cannot find the right answer on any of the above-mentioned sources, engage with ETG's technology experts directly. For support tips, see chapter 6.

EtherCAT EtherCAT P Safety over EtherCAT

SubDevice Implementation Guide

SECTION II – ESC overview and EtherCAT development products

EtherCAT SubDevice Controllers, EtherCAT development products, evaluation kits, communication modules, implementation specifics

EtherCAT Technology Group

CONTENTS

1 Introduction II-5
2 EtherCAT SubDevice evaluation boards II-6
3 EtherCAT SubDevice communication modules II-9
3.1 EtherCAT communication modules overview II-9
3.2 Beckhoff FB1111 ..... II-10
3.3 Hilscher comX II-11
4 EtherCAT SubDevice Controller overview II-12
5 Missing your device? II-16

TABLES

Table 1: EtherCAT evaluation kits II-7
Table 2: EtherCAT communication modules II-9
Table 3: FB1111 options II-10

FIGURES

Figure 1: Product Guide filter for evaluation boards II-6
Figure 2: Beckhoff FB1111 ..... II-10
Figure 3: Hilscher comX module II-11

1 Introduction

There is a wide range of EtherCAT SubDevice implementation possibilities available, accounting for the different types of devices, as well as for the different types of development approaches and needs.

One outstanding feature of EtherCAT is the number of ESC vendors, ranging from different ASICs to multi-protocol-solutions, SoCs combining an ESC and microcontroller or even CPU on a single silicon and communication modules with an API. Some already come with an EtherCAT SubDevice stack, other support the integration of widely used state-of-the-art stacks.

Vendors provide development kits including documentation and for some specific trainings are offered.

This section lists those devices and products. The Lists might not be complete, however, they already provide an extensive overview. ETG encourages vendors to add their product to it.

2 EtherCAT SubDevice evaluation boards

Evaluation boards offer a good starting point for EtherCAT SubDevice development. They provide the hardware for a “sample EtherCAT SubDevice”, so that software development can be started. The product-specific PCB design can be initiated in parallel.

This chapter provides an overview of EtherCAT SubDevice development boards (Table 1) while it cannot claim to be complete. For latest information visit the EtherCAT Product Guide on the website with the filter settings as shown in Figure 1 and in a next step, the vendors directly.

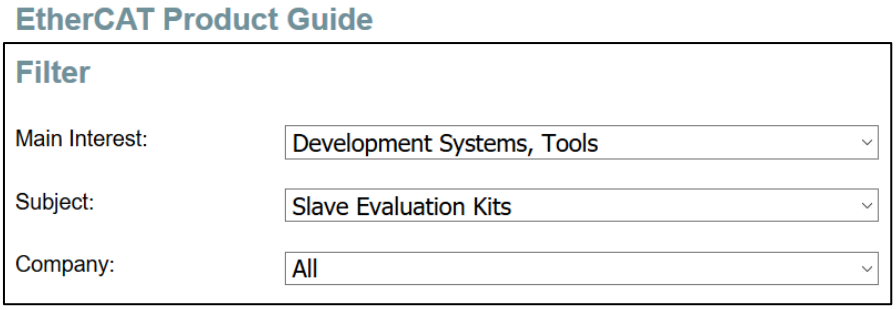
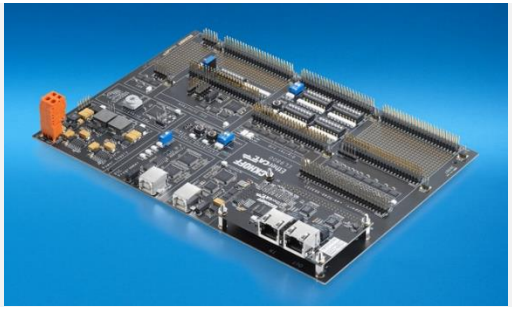
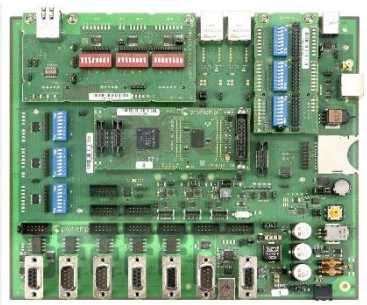
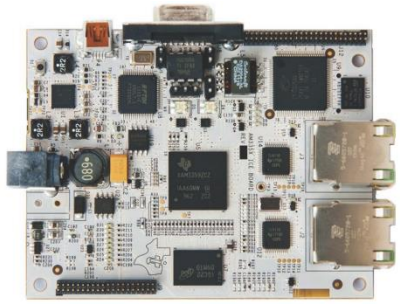


Figure 1: Product Guide filter for evaluation boards

Table 1: EtherCAT evaluation kits

Beckhoff EL9820	Hilscher NXHX 500	Profichip ANTAIOS
		
For the evaluation kit (base board EL9800 with EtherCAT piggyback controller board), a one-day hands-on workshop and a preceding one day training class explaining the EtherCAT protocol are offered optionally(section I, chapter 6.1). Base board with: • Socket for FB1111 EtherCAT piggyback controller board with EtherCAT SubDevice Controller • Several PDI (32-bit digital I/O, 8/16-bit µC, SPI) to connect hardware • On-board PIC connected via SPI to ESC with pre-installed SSC • Debugger interface for MPLAB® • Power supply (24V) • Cables, documentation • EtherCAT SubDevice Stack Code (SSC) • Handling of synchronous and asynchronous data exchange via DPRAM • Support of mailbox application protocols (CoE incl. object dictionary, EoE, FoE, AoE) • Support of synchronized application using Distributed Clocks • SubDevice controller board, equipped with ET1100	The netX network controller with its 32-bit / 200 MHz ARM CPU provides a high degree of computing performance and comprehensive peripheral functions for single chip solutions in price-sensitive applications. Here the network protocols and the application program together use the resources of the netX and are carried out together in a real-time operating system. • I/O, parallel host interface, UART, USB, JTAG • DIP-switches and LEDs for I/O, SD card slot, fieldbus interface (optional), multi-protocol support • EtherCAT SubDevice Hardware Abstraction Layer (HAL) available on demand • Integrated debug interface supplied with the HiTOP development environment from Hitex.	The new ANTAIOS multi-fieldbus controller from Profichip enables higher integration in a smaller package to fulfill the requirements of the industrial automation market. The latest development by Profichip combines a powerful processor for advanced user applications with an effective and very flexible communication technology for today's industrial applications demanding sophisticated real-time capabilities. Additionally a unique feature of this System-on-Chip (SoC) is the high-speed backplane communication master for Profichip's SliceBus® technology. • ARM® Cortex®-A5 (288MHz, 32kB+32kB cache, 64Bit FPU) • 2-port real-time Ethernet switching unit (Profinet IRT, EtherCAT, etc.) • 2 integrated Ethernet PHYs (copper + fiber) • 1 Gb Ethernet MAC • Also, non-Ethernet based fieldbus protocols are supported • DDR2 external memory interface • QuadSPI interface (e.g., NOR flash for firmware) • SD/MMC, NAND, USB2 device, SRAM master/slave, SPI master/slave
https://www.beckhoff.de/english.asp?ethercat/el9820_el9821_el9830_el9840_el9803.htm	https://www.hilscher.com/products/product-groups/network-controller/development-boards/?	http://www.profichip.com/products/tools/evaluation-kits/antaios-evaluation-kit/

TI AM3359 Industrial Communications Engine (ICE)	TESSERA Technology R-IN32M3-EC	
		
The ICE provides development of industrial communication applications, i.e. communication modules, I/O devices, sensors and other similar applications. It includes the essential peripherals for the EtherCAT communication and further industrial communication standards. The SDK includes a SYS/BIOS™ based real-time kernel with application-level communication stack and device drivers. The development and debug tool chain are also included with this platform. • Sitara AM3359 ARM Cortex-A8 MPU • RJ-45 connected to TLK110 Ethernet PHY • 8x digital in, 8x digital out • 8 MB serial SPI flash • 256 MB DDR2 (opt.) • 8 kB dual-ported RAM • Micro-SD slot • CAN, SPI, GPIO and UART • Parallel I/O to dual-ported RAM • JTAG via USB port (optionally 20 pin JTAG header) • Debug UART via USB port	TS-R-IN32M3-EC is the platform to develop software and system for Renesas Electronics industrial Ethernet communication LSI, R-IN32M3-EC. EtherCAT and other industrial protocols can be evaluated. Sample software can be downloaded from the Renesas Electronics website or the protocol stack vendor website. • Renesas Electronics “R-IN32M3-EC” • ARM Cortex™-M3 with HW-RTOS • 1.3MByte embedded SRAM (100MHz) • 10Base-T/100Base-TX Ethernet PHY (2 ch) • Flash memory (serial) : 32 Mb × dual-flash memory (parallel) : 2 Mb × 16 bit • EEPROM : 16Kbit • UART (USB), CSI, I2C, CAN, RJ45 (Ethernet/EtherCAT) • Extension connector, GPIO (DIP, LED, pin header), CC-Link • 20-pin half-pitch connector (trace supported) • IAR systems : I-jet/JTAGjet	
http://www.ti.com/tool/TMDSICE3359?keyMatch=TMDSICE3359&tisearch=Search-EN-Everything	www.tessera.co.jp/eng/products/r-in32m3-ec-e.html	

3 EtherCAT SubDevice communication modules

An overview of a selection of communication modules is given first. Several of the communication modules are then listed in alphabetical order with detailed description (see Table 2).

3.1 EtherCAT communication modules overview

Table 2: EtherCAT communication modules

	COMX	ANYBUS-S	ANYBUS-CC	FB1111-0140 | 0141 | 0142	UMD2
					
Hardware supplier	Hilscher	HMS		Beckhoff	OKI
Size (mm)	70 x 30 x 18	54 x 86 x 16,6	51,8 x 50,1 x 22,3	55 x 85,5 x 14	20 x 30 x 4
ESC*	NetX 100	FPGA with IP core + ASIC		ET1100	SH2A + ET1100
application controller interface	DPM (8/16-bit)	DPM (8-bit)		8/16-bit BUS SPI 32-bit digital I/O	
LEDs (RUN/ERR/LINK)	All	All		All	RUN, ERR
No. of ports	2 x RJ45	2 x RJ45		2 x RJ45	2 x (MII/EBUS)
Power supply	3.3 V	5 V	3.3 V	5 V	5V
Power consumption	700 mA	450 mA	500 mA	700 mA	200mA
Further information	Chapter 3.3	https://www.anybus.com/products/embedded-index/embedded-networks/ethercat		Chapter 3.2	

*see chapter 4 for detailed information about available EtherCAT features which are depending on the applied ESC.

3.2 Beckhoff FB1111

The Beckhoff FB1111 EtherCAT piggyback controller boards (Table 3) offer complete EtherCAT connection based on the ET1100 EtherCAT ASIC (Figure 2). All FB1111 have the same form factor and can be used with the EL9800 EtherCAT evaluation kit. They can be integrated as EtherCAT interfaces in devices.

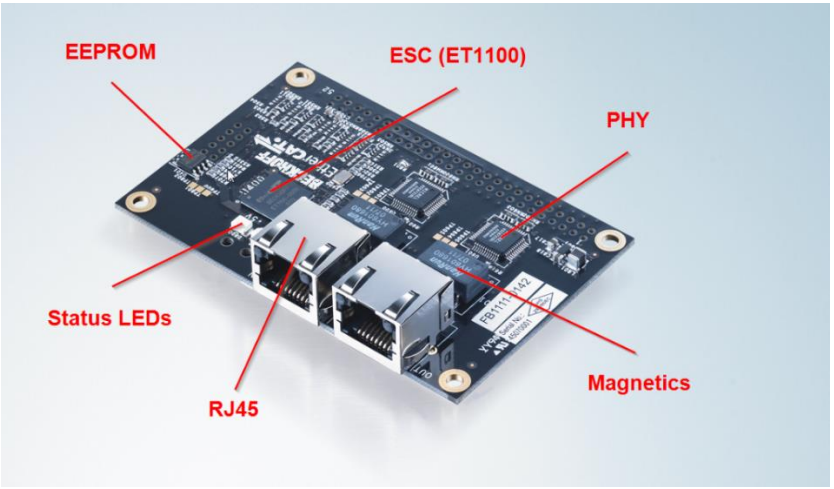


Figure 2: Beckhoff FB1111

Table 3: FB1111 options

Part	Description
FB1111-0140	EtherCAT piggyback controller board with ET1100 (ASIC) and µC interface; can be integrated as EtherCAT interface in devices.
FB1111-0141	EtherCAT piggyback controller board with ET1100 (ASIC) and SPI interface; can be integrated as EtherCAT interface in devices.
FB1111-0142	EtherCAT piggyback controller board with ET1100 (ASIC) and digital I/O interface; can be integrated as EtherCAT interface in devices; included in the EL982x evaluation kit and together with the delivered adapter card EL9803 all interfaces (µC, SPI, digital I/O) can be applied. This is the most flexible solution for starting an EtherCAT implementation.

Further information: https://www.beckhoff.com/FB1111

3.3 Hilscher comX

- Interfaces: host processor over dual-ported memory (parallel)
- Ports: 2 (100BASE-TX)

All stacks are implemented as SubDevice protocols and are executed on the comX-module (Figure 3). Data exchange with the host application is carried out via the Dual-Port Memory Interface. The process data images are available directly via memory read and write functions. The comX module features two RJ45 connectors for Ethernet. netX based comX modules gets its identity by loading an appropriate firmware file.

- All real-time Ethernet systems use netX network controller
- Available as MainDevice and SubDevice
- Two Ethernet ports
- System/status/link/activity LEDs
- 8- or 16-Bit host application interface
- USB & UART diagnostic interface
- Direct process data access
- Same dimensions and pin-compatible like the well-known COM-C module
- SYCON.net as configurator based on FDT/DTM
- Short time-to-market

Figure 3: Hilscher comX module

Further information: https://www.hilscher.com/products/product-groups/embedded-modules/communication-module/

4 EtherCAT SubDevice Controller overview

The following pages give an overview about several ESC variants. Concerning to a continuing development of products by ETG-Members we recommend also to have a look on the online version of the EtherCAT SubDevice Controller (ESC) Overview.

Name	AX58100	AX58200	AX58400	TMC8462	ET1100	ET1810/ET1811/ET1812	ET1851/ET1816/ET1817	Anybus NP40	netX 500
Type	ASIC	ARM MPU	Dual-Core ARM MPU	ASIC	ASIC	Intel (Altera) FPGA + IP Core	Xilinx FPGA + IP Core	ARM MPU	ASIC
Supplier	ASIX	ASIX	ASIX	ADI Trinamic	BECKHOFF	BECKHOFF	BECKHOFF	HMS	hilscher empowering communication
Package	80-pin LQFP 0.4 mm pitch	144-pin HSFBGA 0.8 mm pitch	225LD EHS-TFBGA 0.8 mm pitch	BGA121 0.75 mm pitch	BGA128 0.8 mm pitch	FPGA dependent	FPGA dependent	BGA VF400 0.8 mm pitch	BGA345 1 mm pitch
Size	12 x 12 mm	10 x 10 mm	13 x 13 mm	9x9 mm	10 x 10 mm	FPGA dependent	FPGA dependent	17 x 17 mm	22 x 22 mm
µC Interface	SPI/parallel (8/16-bit, asynchronous)	uC bus (Internal, AHB)	uC bus (Internal, AHB)	serial or standalone	serial/parallel (8/16bit, sync/async)*	serial/parallel (8- /16-bit, async) AVALON®*	serial/parallel (8- /16-bit, async) OPB®* and PLB®*	Anybus interface (8- / 16-bit 30 ns parallel, 20 MHz SPI, Shift register, UART)	µC bus (internal, 32bit)
Digital I/O	32	-	20	0..16*	8-32*	8-32*	8-32*	256 / 256 (Shift register mode)	-
General Purpose I/O	32	up to 76*	up to 97*	0..24*	0-32*	0-128*	0-128*	-	16
DPRAM	9 kByte	160 kByte	1 Mbyte	16 kByte	8 kByte	0...60 kByte*	0...60 kByte*	12 kByte	256/512 Byte (Mailbox/Process Data)
SyncManager Entities	8	8	8	8	8	0...8*	0...8*	4	4
FMMU Entities	8	8	8	8	8	0...8*	0...8*	4	3
Distributed Clock Support	yes (64-bit)	yes (64-bit)	yes (64-bit)	yes	yes	yes*	yes*	yes	yes
No. of Ports	2 (100BaseTX) + Opt. 1 (MII)	2 (100BaseTX) + Opt. 1 (MII)	2 (100BaseTX) + Opt. 1 (MII)	2 (100BaseTX)	2-4 (MII/E-BUS)*	1-3 (MII/max. 2 RMII)	1-3 (MII/max. 2 RMII)	2 (MII)	2 (100BaseTX)
Specials	100BASE-FX support 2 integrated PHYs 3-ch PWM and S/D I/F ABZ and Hall encoder I/F SPI master I/F	2 integrated Ethernet PHYs, USB 2.0 HS OTG, 10/100Mbps Ethernet MAC with RMII and hardware cryptography accelerator 6xLPUARTs, 3xISO-7816-3, 1xQuad-SPI, 3xI2C, 1xI2S, 2xUSCI, 2xCAN, 1xSPI Flash I/F, 2xSDHC, 1x16-ch/12-bit ADC, 2x12-bit DAC, 2xAnalog Comparators, 2xOperational Amplifiers , 4x32-bit timers, 24x16-bit PWM counters, 2xQEI, 1xECAP, supports Real-Time Clock (RTC) , Built-in Die Temperature Sensor (DTS)	Dual-Core 480MHz ARM Cortex-M7 & 240MHz Cortex-M4 MCU, 2 Mbytes embedded Flash memory, 2 ESC Integrated PHY, USB HS OTG, Additional RMII/MII Ethernet MAC with IEEE 1588 for multiprotocol support, TFT-LCD display controller, Security and Cryptographic Accelerator , 96-bit UID, Watchdog, RTC /SysTick timers, Rich communication/control interfaces such as SPI/UART/I2C /I2S/SAI/CAN/SDMMC/ADC/DAC/HDMI-CEC/PWM/DFSDM, etc.	Wide supply range (up to 35V), 2x Integrated DC/DC regulators, 8x Direct High Voltage I/Os, Multi-function I/O block, Integrated PHYs, BGA routable with standard PCB	BGA routable with standard PCB	Various license models and OpenCore Plus are available. A wide range of Intel (former: Altera) FPGAs are supported	Various license models and evaluation Version are available. A wide range of Xilinx FPGAs are supported	Multi-protocol support, ESC Frame forwarding delay: 114 ns, MDP, possible to implement several device profiles	Multi-protocol support, Integrated PHYs, Integrated µC (ARM9-200MHz)

Name	netX 51	netX 52	netX 90	netX 100	XMC4300	XMC4800	LAN9252	LAN9253
Type	ASIC	ASIC	ASIC	ASIC	ARM MPU	ARM MPU	ASIC	ASIC
Supplier	hilscher empowering communication	hilscher empowering communication	hilscher empowering communication	hilscher empowering communication	Infineon	Infineon	Microchip	Microchip
Package	PBGA324 1 mm pitch	PBGA244 1 mm pitch	LFBGA144 0.8 mm pitch	BGA345 1 mm pitch	100 LQFP (0.5 mm)	100 LQFP (0.5 mm)	64 pin QFN (0.5 mm pitch) 64 pin TQFP-EP (0.5 mm pitch)	64 pin QFN (0.5 mm pitch)
Size	19 x 19 mm	15 x 15 mm	10 x 10 mm	22 x 22 mm	16 x 16 mm	20 x 20 mm 16 x 16 mm 12 x 12 mm	9 x9 mm 12 x 12 mm	9 x9 mm
µC Interface	µC bus (internal, 32bit)	µC bus (internal, 32bit)	µC bus (internal, 32bit)	µC bus (internal, 32bit)	µC bus (internal, AHB)	µC bus (internal, AHB)	Host Bus/SPI/SQI	8/16-bit Host Bus/SPI/SQI
Digital I/O	-	-	-	-	-	-	0-16*	0-16*
General Purpose I/O	32	24	16	16	0 - 46	0 - 123	0-16*	0-16*
DPRAM	6 kByte	6 kByte	6 kByte	256/512 Byte (Mailbox/Process Data)	8 kByte	8 kByte	4 kByte	8 kByte
SyncManager Entities	8	8	8	4	8	8	4	8
FMMU Entities	8	8	8	3	8	8	3	8
Distributed Clock Support	yes	yes	yes	yes	yes (64 Bit)	yes (64 Bit)	yes	yes
No. of Ports	2 (100BaseTX)	2 (100BaseTX)	2 (100BaseTX)	2 (100BaseTX)	2 (MII)	2 (MII)	2 (100BaseTX) + opt. 1 (MII)	2 (100BaseTX) + opt. 1 (MII)
Specials	Multi-protocol support, Integrated PHYs, Integrated µC (ARM9-100MHz)	Multi-protocol support, Integrated PHYs, Integrated µC (ARM9-100MHz)	Multi-protocol support, Integrated PHYs, Integrated µC, OnChip Flash 1,5 Mbytes, OnChip DC-DC Converter, (ARM Cortex M4-100MHz) Additional integrated Application Controller (ARM Cortex M4 - 100 MHz)	Multi-protocol support, Integrated PHYs, Integrated µC (ARM9-200MHz)	EtherCAT® node on an ARM® Cortex®-M4 processor with up to 256kB on-chip flash, 128kB on-chip RAM and analog/mixed signal capabilities. Qualified for up to 125°C ambient temperature.	EtherCAT® node on an ARM® Cortex®-M4 processor with up to 2MB on-chip flash, 352kB on-chip RAM and analog/mixed signal capabilities. Qualified for up to 125°C ambient temperature.	Cable Diagnostics, 100FX support, 2 integrated PHYs, integrated 1.2V regulator	Host EEPROM Emulation support Supports for low-cost 25MHz crystal Cable Diagnostics, Wake on LAN, 2 integrated PHYs, Single Supply operation (3.3V) Integrated 1.2V regulator

Name	LAN9254	LAN9255	EC-1	RZ/T1	RZ/N2L	RZ/T2M	R-IN32M3-EC	RX72M	ANTAIOS
Type	ASIC	ARM MCU	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU
Supplier	MICROCHIP	MICROCHIP	RENESAS	RENESAS	RENESAS	RENESAS	RENESAS	RENESAS	YASKAWA
Package	80 pin TQFP-EP (0.5 mm pitch)	128 pin TQFP (0.4 mm pitch)	196 pin BGA (0.8 mm)	FBGA320 0.8 mm pitch	FBGA225, 0.8mm pitch FBGA121, 0.8mm pitch	FBGA320, 0.8mm pitch FBGA225, 0.8mm pitch	BGA324 1 mm pitch	LFBGA224, 0.8mm pitch LFBGA176, 0.8mm pitch LFQFP176, 0.5mm pitch LFQFP144, 0.5mm pitch LFQFP100, 0.5mm pitch	TFBGA-380 (0.65 mm pitch) TFBGA-385 (0.8 mm pitch)
Size	12x12 mm	14x14 mm	12 x 12 mm	17 x 17 mm	13 x 13 mm 10 x 10 mm	17 x 17 mm 13 x 13 mm	19 x 19 mm	LFBGA224: 13 x 13 mm LFBGA176: 13 x 13 mm LFQFP176: 24 x 24 mm LFQFP144: 20 x 20 mm LFQFP100: 14 x 14 mm	15 mm x 15 mm 19 mm x 19 mm
µC Interface	8/16-bit Host Bus/SPI/SQI	SPI/SQI up to 60MHz	USB Host/Function, CAN, SCIFA, I2C RSPI, Flash	16/32-bit parallel and various serial (SPI/I2C/UART)	USB Host/Function, CAN-FD, SPI, SCI, DSMIF, I2C, xSPI, External BUS IF(SRAM/SDRAM/Host IF)	USB Host/Function, CAN-FD, SPI, SCI, DSMIF, I2C, xSPI, External BUS IF(SRAM/SDRAM)	16/32-bit parallel (master/slave) and serial (SPI/I2C/UART)	USB, CAN, UART, SPI, I2C, SCI, QSPI	SPI / QSPI / 16 Bit asynchronous interface
Digital I/O	0-32*	0-32*	-	-	-	-	-	44	26Bits Input, 20Bits Output
General Purpose I/O	0-32*	0-32*	115* GPIOs / 8 Input (port multiplexed, partial 5V-tolerant, open drain, input pull-up)	0-209*	0-134*	0-193*	0-96*	0-182*	up to 32
DPRAM	8 kByte	8 kByte	512 KB (ATCM) with ECC 32 KB (BTCM) with ECC	8 kByte	8 kByte	8 kByte	8 kByte	8 kByte	up to 64 kByte
SyncManager Entities	8	8	8	8	8	8	8	8	8
FMMU Entities	8	8	8	8	8	8	8	8	8
Distributed Clock Support	yes	yes	yes (64 bit)	yes	yes	yes	yes	yes (64 bit)	yes (64 bit)
No. of Ports	2 (100BaseTX) + opt. 1 (MII)	2 (100BaseTX) + opt. 1 (MII)	2 (MII)	2 (RMII/MII)	3 (RGMII/RMII/MII)	3 (RGMII/RMII/MII)	2 (100BaseTX)	2 (100BaseTX/MII/RMII)	2 (100BaseTX) or 2 (MII)
Specials	Host EEPROM Emulation support Supports for low-cost 25MHz crystal Cable Diagnostics, Wake on LAN, 2 integrated PHYs, Single Supply operation (3.3V) Integrated 1.2V regulator	Integrated SAME53J ARM Cortex-M4F MCU 1MB Progammable Flash 256KB Main Memory SRAM Extended Industrial Temperature rated (-40 to +105C) Cable Diagnostics, Wake on LAN, 2 integrated PHYs, Single Supply operation (3.3V) Integrated 1.2V regulator	Safety Functions, Multi-Function Pin Controller	Additional Ethernet port (RMII/MII), 2-axis high-speed motion control support, digital encoder interfaces (EnDat, BiSS, others), Multi-protocol support, security option, functional safety support, Cortex-R4F (450/600MHz), Cortex-M3 (150MHz) cores	Multi-protocol support (EtherCAT, etc), Optimized for network companion chip (parallel bus slave, xSPI slave interface to connect external application CPU), One chip solution for various applications, Functional safety support, Cortex-R52 (400MHz) core	Multi-protocol support (EtherCAT, etc), Optimized for motor control (2-axis high-speed motion control support, digital encoder interfaces (EnDat, BiSS, others, etc.), Functional safety support, Cortex-R52 Dual (800MHz) cores	Multi-protocol support, SPI, I2C, UART, 1.3 Mbyte int. RAM, <1W typical incl. 2 PHYs	Multi-protocol support (EtherCAT, etc.), Security option, Encryption option, 105 °C operating temperature support, Functional safety support	Multi fieldbus protocol support, 2 x integrated PHYs, 1 x integrated GBit Ethernet MAC, Integrated ARM® Cortex®-A5 (288MHz), Backplane communication: SliceBus master for profichip's SNAP+ ASIC, Integrated technology module (2xSII / 4xPWM / 4xCounter), QuadSPI interface (e.g. NOR Flash for firmware), DDR2 external memory interface, Other external interfaces: SD/MMC, NAND, USB2 device, SRAM master/slave, SPI master/slave

Name	TRITON	C2000™ (TMS320F28388D/S)	Sitara AMIC110 SoC	Sitara AM3357/9	Sitara AM4377/9	Sitara AM571xE	Sitara AM572xE	Sitara AM65x SoC	Sitara AM64x SoC
Type	ARM MPU	TI C28x subsystem(s) with ARM Connectivity Manager	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU	ARM MPU
Supplier	YASKAWA	Texas Instruments	Texas Instruments	Texas Instruments	Texas Instruments	Texas Instruments	Texas Instruments	Texas Instruments	Texas Instruments
Package	FCBGA-784 (0.8 mm pitch)	337 BGA 0.8mm pitch 176 QFP 0.5mm pitch	324-pin NFBGA 0.8mm pitch	324-pin NFBGA 0.8 mm pitch	491-pin NFBGA, 0.65mm pitch (0.8 mm effective routing)	760-pin FCBGA 0.8 mm pitch	760-pin FCBGA 0.8 mm pitch	784-pin S-PBGA 0.8mm pitch	441-pin FCBGA 0.8mm pitch
Size	23 mm x 23 mm	16 x16 mm 26 x 26 mm	15x15mm	15 x 15 mm	17 x 17 mm	23 x 23 mm	23 x 23 mm	23mmx23mm	17.2mmx17.2mm
µC Interface	SPI / QSPI / 16 Bit asynchronous interface	16-bit async PDI interface	200MHz interconnect (internal, 32bit)	200 MHz interconnect (internal, 32bit)	200 MHz interconnect (internal, 32bit)	200 MHz interconnect (internal, 32bit)	200 MHz interconnect (internal, 32bit)	250MHz interconnect (internal, 256bit)	250MHz interconnect (internal, 128bit)
Digital I/O	32Bits Input, 22Bits Output	N/A	8	8	8	8	8	8	8
General Purpose I/O	up to 32	32	>32	> 32	> 32	> 32	> 32	>32	>32
DPRAM	up to 64 kByte	16 kByte	8 kByte	8 kByte	28 kByte	28 kByte	28 kByte	60 kByte	60 kByte
SyncManager Entities	8	8	8	8	8	8	8	8	8
FMMU Entities	8	8	8	8	8	8	8	8	8
Distributed Clock Support	yes (64 bit)	yes	yes	yes	yes	yes	yes	yes	yes
No. of Ports	4 Gbit Ethernet port	2 (MII)	2 (MII)	2 (MII)	2 (MII)	2 (MII)	2 (MII)	2 (MII)	2 (MII)
Specials	Multi fieldbus protocol support, 2 port Real-Time Ethernet switch with integrated PHYs, 2x integrated GBit Ethernet MAC,3 Integrated ARM® Cortex®-A17 (1.26GHz), Secure Core, Backplane communication: SliceBus master for profichip's SNAP+ ASIC, Integrated technology module (2xSII / 4xPWM / 4xCounter), QuadSPI interface (e.g. NOR Flash for firmware), DDR4 external memory interface, 3 PCI Express® Controller, Other external interfaces: 3 SD/MMC, NAND, USB2 device, SRAM master/slave, SPI master/slave	EtherCAT slave enabled real-time controller. Up to 925 MIPS. Single or dual C28x + CLA control subsystems for real-time control loops. Arm based Connectivity Manager for communictions and host control. On-chip flash, RAM, 4x 16-bit ADC, SDFM, 32-ch PWM, analog comparator subsystem, multiple communications ports, configurable logic block for CPLD/FPGA replacement and absolue encoder support.	Entire EtherCAT slave controller can be implemented on internal memory (no external DDR needed), Industrial Communications Subsystem (PRU-ICSS) for multi-protocol support, CAN	Industrial Communications Subsystem (PRU-ICSS) for multi-protocol support, Gigabit Switch, CAN, display, ARM Cortex-A8 (275MHz-1000MHz)	Multi-protocol support, Second PRU-ICSS for Motor control (EnDat, sigma delta filtering etc), Gigabit Switch, CAN, Display subsystem, 2D/3D graphics, Camera I/F, Optional secure boot, ARM Cortex-A9 (upto 1 GHz)	Dual Industrial Communications Subsystem (PRU-ICSS) for multi-protocol support (2 EtherCAT slave instances or EtherCAT slave to protocol gateway), Motor control (EnDat, sigma delta filtering), 2D/3D Graphics, Display subsystem, Video acceleration, PCIe, SATA, Optional secure boot, ARM Cortex-A15 (upto 1.5GHz), 2x M4 cores, 1x C66x DSP core	Dual Industrial Communications Subsystem (PRU-ICSS) for multi-protocol support (2 EtherCAT slave instances or EtherCAT slave to protocol gateway), 2D/3D Graphics, Display subsystem, Video acceleration, PCIe, SATA, Optional secure boot, 2x ARM Cortex-A15 (upto 1.5 GHz), 2x M4 cores, 2x C66x DSP cores	Entire ESC can be implemented on internal memory (no external DDR needed), 3x Gigabit Industrial Communications Subsystem (PRU_ICSSG) for multi-protocol support (up to 3 EtherCAT slave instances), PRU_ICSSG also supports Motor Control functionality (Encoder feedback such as Hiperface-DSL and EnDat and Sigma Delta filtering), up to 4x Arm Cortex-A53 cores at 1.1GHz, 2x Cortex-R5F core at 400MHz with optional lock-step for functional safety or other purposes, 2MB on-chip SRAM	Entire ESC can be implemented on internal memory (no external DDR needed), Dual Gigabit Industrial Communications Subsystem (PRU_ICSSG) for multi-protocol support (2 EtherCAT slave instances or EtherCAT slave to protocol gateway), PRU_ICSSG also supports Motor Control functionality (Encoder feedback such as Hiperface-DSL and EnDat and Sigma Delta filtering), up to 4x Arm Cortex-R5F cores at 800MHz, up to 2x Cortex-A53 cores at 1GHz, 1x Cortex-M4 core at 400MHz for functional safety or other purposes, 2MB on-chip SRAM

5 Missing your device?

Section II generally contains a snapshot of the spectrum of available products for a SubDevice implementation. ETG members that offer EtherCAT development products, EtherCAT implementation services and EtherCAT workshops are invited to contribute information to the ETG for this guide. A range of products can be found at www.ethercat.org/products.

If you are missing your device here, you found an error or you have a suggestion for SubDevice implementation support, feel free to contact the ETG and help to improve this document.

EtherCAT EtherCAT P Safety over EtherCAT

SubDevice Implementation Guide

SECTION III – EtherCAT P introduction and implementation

EtherCAT P technology introduction, EtherCAT P specification and documents, licensing, conformance testing, implementation

EtherCAT Technology Group

CONTENTS

1 EtherCAT P introduction III-5
2 EtherCAT P technology III-6
2.1 EtherCAT P connectors and cables III-6
2.2 PHY selection III-7
2.3 EtherCAT P use cases III-8
2.4 EtherCAT P device structure III-8
2.4.1 EtherCAT P in the ESI file III-10
2.4.2 EtherCAT P device types III-11
2.4.3 EtherCAT P device categories III-11
3 EtherCAT P specification and documents III-13
4 EtherCAT P conformance testing III-15
4.1 General III-15
4.2 Evaluate your current EtherCAT SubDevice for EtherCAT P III-15
4.3 Contact for EtherCAT P conformance testing III-15
5 EtherCAT P licensing III-16
5.1 General III-16
5.2 License agreement III-16
6 EtherCAT P implementation aspects III-17
6.1 EtherCAT P and EtherCAT configuration tool III-17
7 EtherCAT P development support III-19
7.1 EtherCAT and EtherCAT P training III-19

TABLES

Table 1: DC power and communication on the same four wires III-6
Table 2: EtherCAT P type, category and description III-12
Table 3: EtherCAT P information, standards and references III-13

FIGURES

Figure 1: EtherCAT P – power combined with EtherCAT on the same four wires III-6
Figure 2: M8 P-coded connector and cable III-7
Figure 3: Trapezoidal EtherCAT P connector for hybrid cables III-7
Figure 4: EtherCAT P is suitable for any kind of devices III-8
Figure 5: EtherCAT P features specified on PhL III-8
Figure 6: A typical EtherCAT device structure III-9
Figure 7: A typical EtherCAT P device structure III-9
Figure 8: Basic EtherCAT P block diagram for circuitry on the IN port III-10
Figure 9: Description of EtherCAT P power consumption characteristics in the ESI file III-11
Figure 10: Mixed EtherCAT / EtherCAT P network with all EtherCAT P device categories III-11
Figure 11: Baseline wander testing with a 75m cable III-15
Figure 12: EtherCAT P logo III-17
Figure 13: EtherCAT P planning tool integrated into a network configuration tool (Beckhoff) III-18
Figure 14: Showing if power-supply is sufficient or not III-18

1 EtherCAT P introduction

The following chapter describes the EtherCAT P technology and its benefit in brief. It provides an overview; however, it does not mean to replace reading the EtherCAT and EtherCAT P specifications and documents.

2 EtherCAT P technology

EtherCAT P is an enhancement to EtherCAT: it combines power (2 x 24V/3A) and the EtherCAT data transmission on the same four wires (Figure 1).

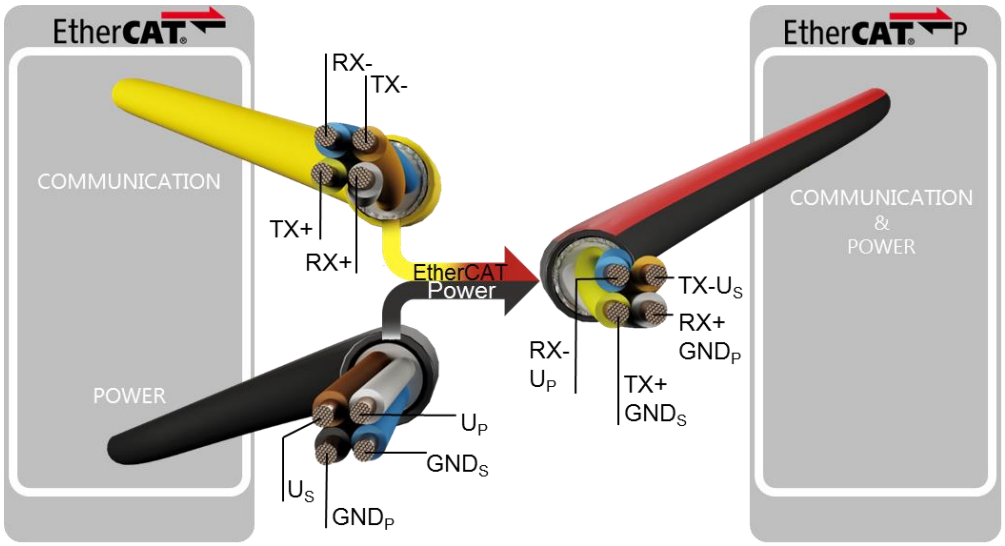


Figure 1: EtherCAT P – power combined with EtherCAT on the same four wires

As shown in Table 1 the four wires utilized for EtherCAT, and the four wires used for powering U_S (for logic) and U_P (for the output peripherals) are combined on the same four wires by EtherCAT P.

Table 1: DC power and communication on the same four wires

(typical) wire color	Yellow	Orange	White	Blue
EtherCAT	TX+	TX-	RX+	RX-
Power	GNDS	US	GNDP	UP

The Ethernet signal used for EtherCAT is combined with the DC currents for U_S and U_P and provides a technology that comprises the following main features:

- Dual power supply
- U_S for system and sensors, 24 V DC/3 A
- U_S for peripheral voltage for actuators, 24 V DC/3 A
- Power forwarding through EtherCAT P devices within each network topology (e.g., daisy-chain, line, etc.)
- 100 % EtherCAT-compatible
- 100 Mbit/s full duplex, processing on the fly, Distributed Clocks, etc.
- Cascadable in all topologies (star, line, tree)

2.1 EtherCAT P connectors and cables

EtherCAT P connectors for 24V/3A are M8 P-coded connectors. This connector provides a unique mechanical keying. This prevents from accidentally connecting EtherCAT devices to an EtherCAT P device. As a result of this simple mechanical concept no smart chips for power-sensing inside the EtherCAT devices are required. Figure 2 shows the M8 P-coded connector. EtherCAT P cable colors are specified to be black and red.

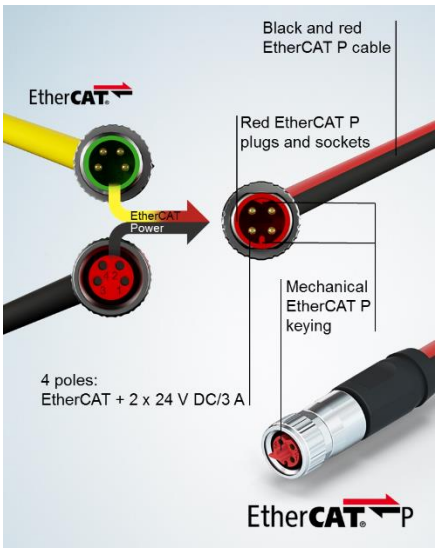


Figure 2: M8 P-coded connector and cable

In combination with the M8 P-coded connectors certain cables are specified (e.g., AWG22/7 and AWG24/7).

When an EtherCAT P cable is combined with an additional power-cable in a hybrid cable, a trapezoidal EtherCAT P connector is used. This allows for a high-density packaging within the hybrid cable as shown in Figure 3.

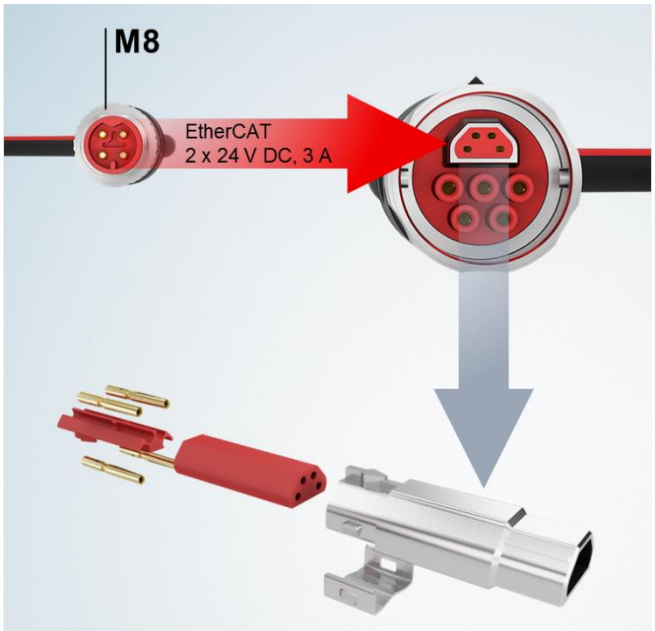


Figure 3: Trapezoidal EtherCAT P connector for hybrid cables

The hybrid cables allow additional energy transmission to supply complete machines, cabinets, robots with one cable including power and EtherCAT for example.

EtherCAT P hybrid cabling and connector technology is under development and will be added to the ETG specifications at a later stage.

2.2 PHY selection

A list of recommended PHYs is provided by the PHY Selection Guide.

Due to the internal interconnection, EtherCAT P places an increased requirement on the SubDevice's analog circuitry design, including its PHYs.

An initial assessment of the SubDevice, specifically the PHY's behavior already used in an existing SubDevice implementation can be done as described in chapter 4 of this section.

2.3 EtherCAT P use cases

EtherCAT P combines all beneficial EtherCAT features – such as line/tree/star topology, unlimited number of devices in the network, Distributed Clocks, diagnosis features, fast EtherCAT performance, and more - with power on the same cable and connector.

It's suitable for all different kinds of devices as shown in Figure 4.

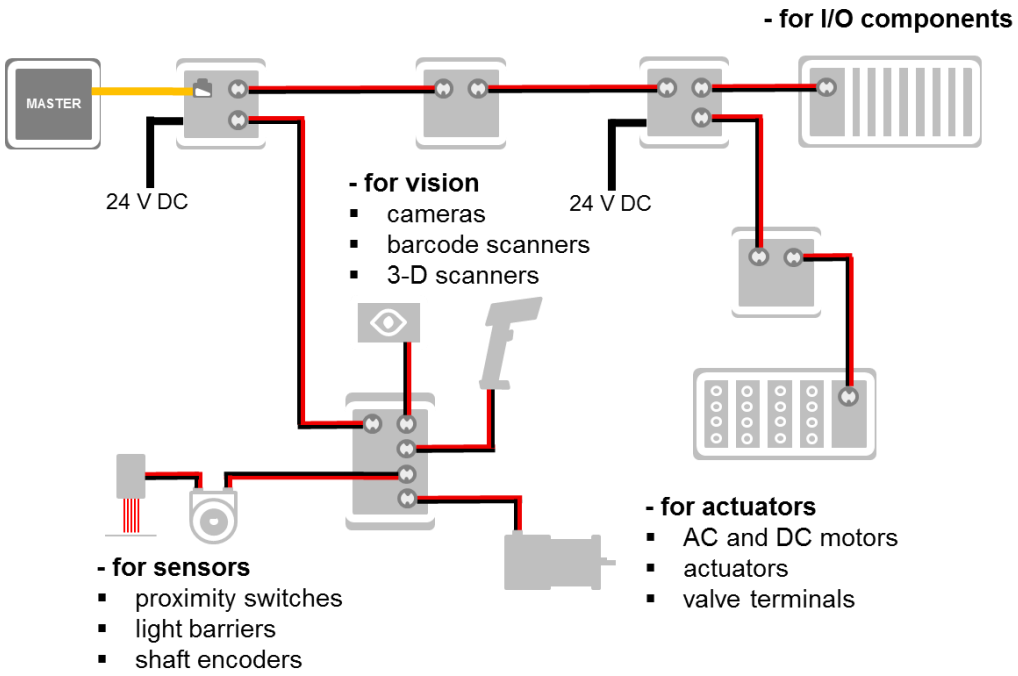


Figure 4: EtherCAT P is suitable for any kind of devices

2.4 EtherCAT P device structure

The ISO/OSI layer model structures communication stacks and specification in the way shown in Figure 5. Taking a reference to EtherCAT, the EtherCAT P functionality and specification is included by the physical layer and its specification parts.

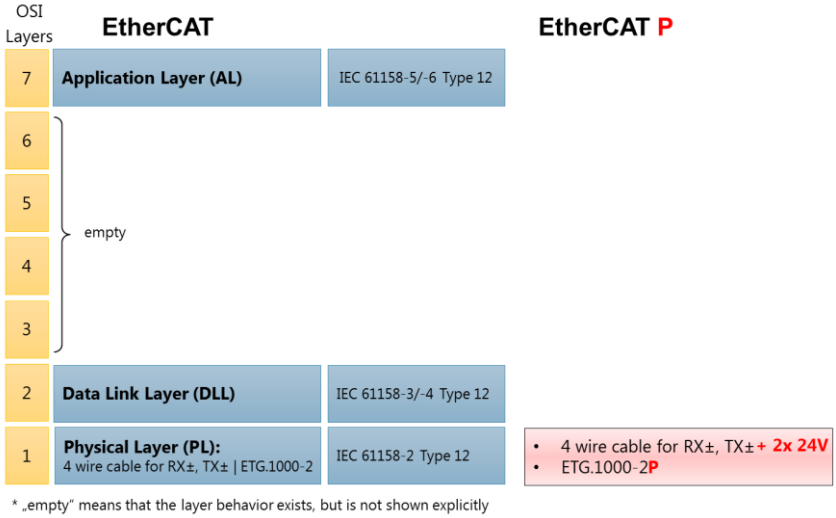


Figure 5: EtherCAT P features specified on PhL

EtherCAT P ports can be on the IN port and on one or several OUT ports of an EtherCAT SubDevice.

Figure 6 shows the EtherCAT device structure itself (in green) within the EtherCAT network, as well as the relation with the EtherCAT configuration tool.

The EtherCAT SubDevice uses a standard Ethernet physical layer layout to interface to the EtherCAT network. The ESI file describes the EtherCAT features in an XML file. This is provided to the EtherCAT network configuration tool. The configuration tool is used to configure the network layout including a

description of the network initialization commands and the cyclic commands. This description is provided to the EtherCAT MainDevice using the ENI file.

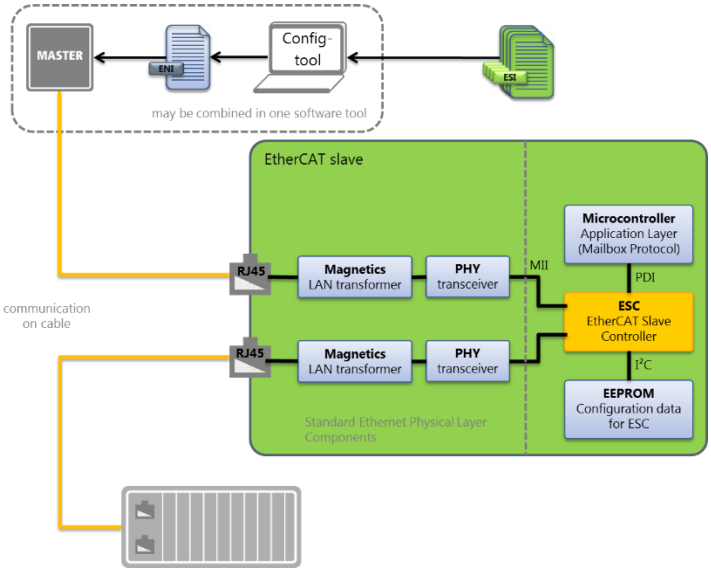


Figure 6: A typical EtherCAT device structure

The add-ons that make an EtherCAT device to be an EtherCAT P device are shown in Figure 7.

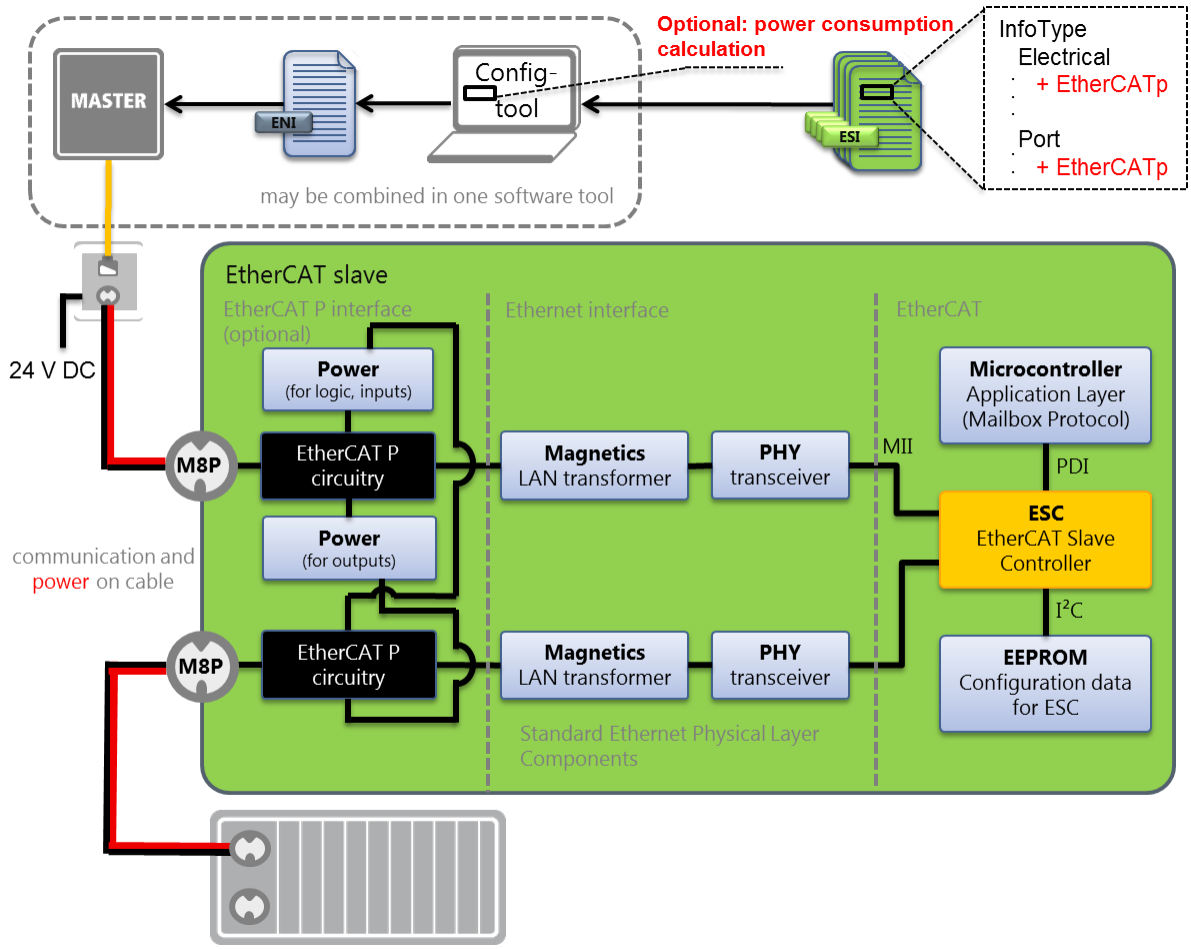


Figure 7: A typical EtherCAT P device structure

The EtherCAT P features on SubDevice side are:

- EtherCAT P interface with the EtherCAT P circuitry and EtherCAT P connector M8 P-coded
- ESI file enhancements to describe the power consumption of the EtherCAT P SubDevice and the power supplied to external sensors/actuators

EtherCAT P networks can be configured with existing configurations tools already – no change on them is required. However, to simplify planning, the configuration tools may be enhanced to calculate and assess the power consumptions in the EtherCAT P segments.

There are no EtherCAT P requirements on the MainDevice to make the system work, or, in other words: Any existing MainDevice can be used to control an EtherCAT P network.

Figure 8 shows a basic EtherCAT P block diagram for circuitry on the IN port:

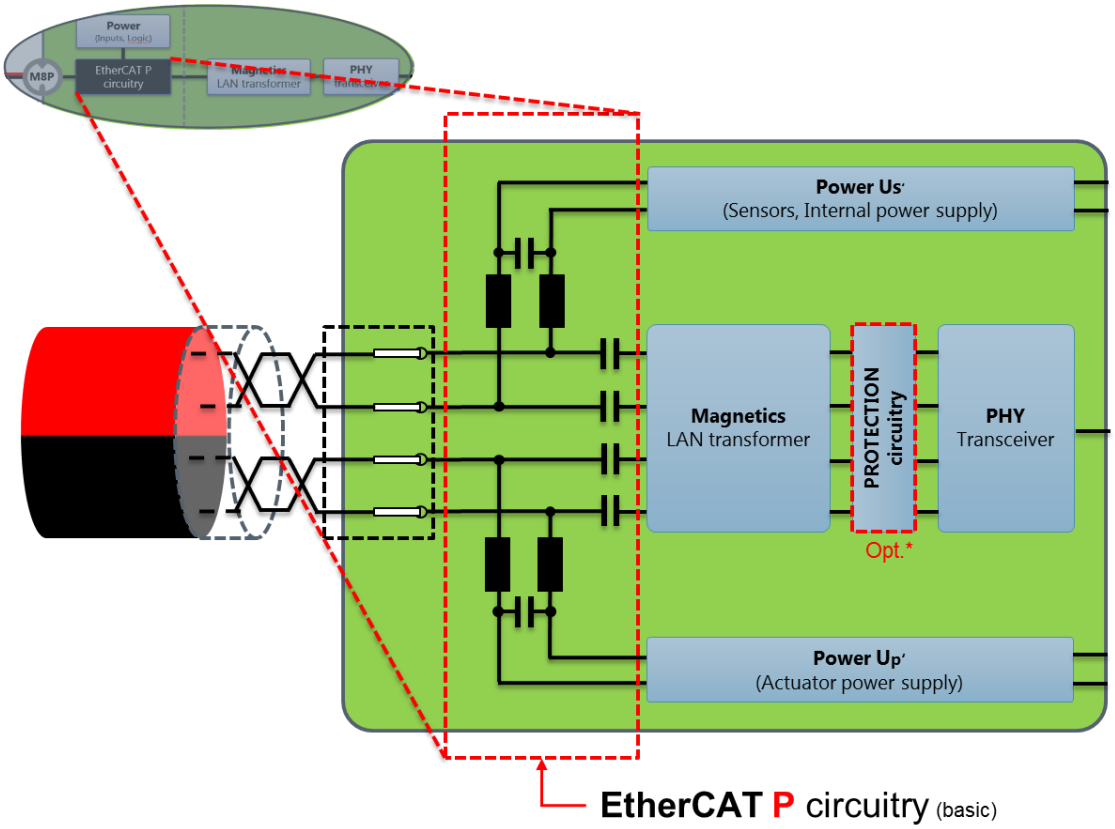


Figure 8: Basic EtherCAT P block diagram for circuitry on the IN port

The four wires are RX± and TX±, which also carry U_S24V/GND and U_P24V/GND. They are connected to the IN port.

The capacitors between the EtherCAT P connector and the magnetics describe a **high pass filter:** They are transparent for the high frequencies but block the DC currents of U_S and U_P.

The LC combination describes a **low pass filter:**
It pass-through the DC currents of U_S/U_P with 24V each but block the high frequencies of the communication signals.

2.4.1 EtherCAT P in the ESI file

The ESI file describes (Figure 9) that the EtherCAT SubDevice is an EtherCAT P SubDevice (either as power sourcing device or powered device), and the power-consumption characteristics.

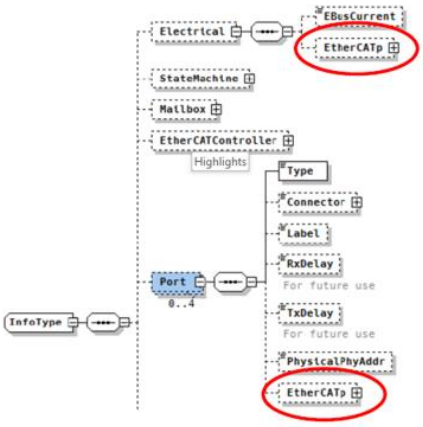


Figure 9: Description of EtherCAT P power consumption characteristics in the ESI file

2.4.2 EtherCAT P device types

EtherCAT P distinguishes the following three EtherCAT P device types to describe if they are consuming power or supplying power to the EtherCAT P system.

- Powered Device (PD):
  Uses the power supplied on its IN port
- Power Sourcing Device (PSD):
  The electronics of the device itself and the power supplied to all OUT ports is taken from an external power supplied to the PSD. Power supplied via the IN port is not forwarded to the OUT ports nor used by the PSD itself.
- Passive Device:
  The power is decoupled by the device without using it. The EtherCAT communication can still be used for connecting further devices.

2.4.3 EtherCAT P device categories

Figure 10 shows a mixed EtherCAT / EtherCAT P network with the different EtherCAT P device categories.

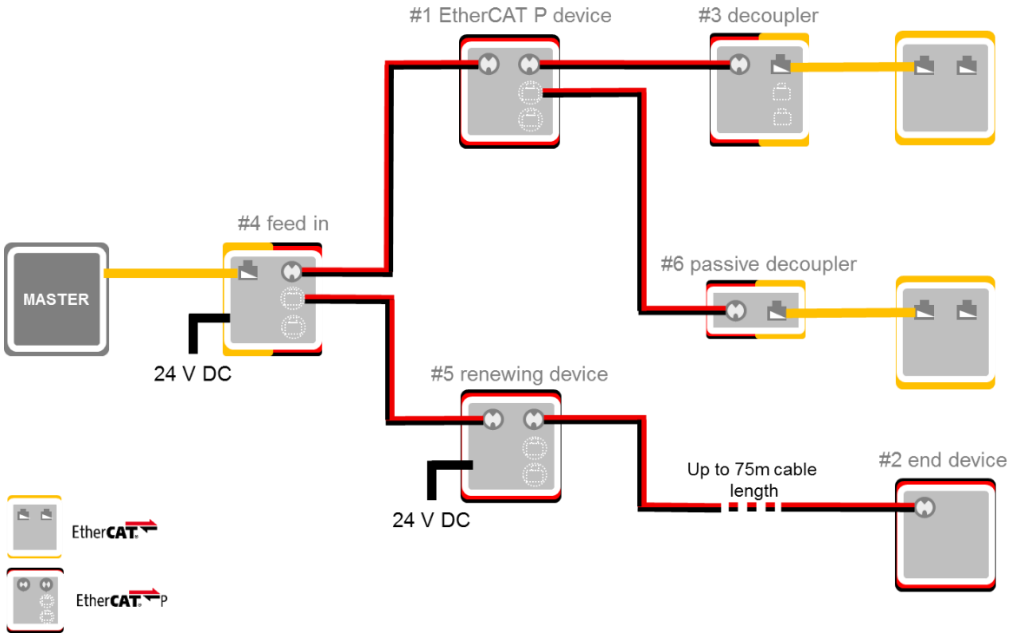


Figure 10: Mixed EtherCAT / EtherCAT P network with all EtherCAT P device categories

Table 2 describes the different EtherCAT P device categories.

Table 2: EtherCAT P type, category and description

#	Type	Category	Description
1	PD	device	“Standard” EtherCAT P device with an EtherCAT P IN port and at least one EtherCAT P OUT port. Other OUT ports may also be EtherCAT.
2	PD	end device	Only an EtherCAT P IN port. Suits ideally for very small EtherCAT P devices (e.g., proximity sensors)
3	PD	decoupler	Decouples power and EtherCAT on the EtherCAT P IN port and has EtherCAT OUT ports. This device has an ESC inside.
4	PSD	feed in	Builds the start of the EtherCAT P segment on the OUT port, while it combines the EtherCAT signal from the IN port and U_S/U_P (taken from external power supply)
5	PSD	renewing device	EtherCAT P device using external power supply to refresh U_S and U_P on the OUT ports. The power supplied on the EtherCAT P IN port is not used any more.
6	Passive	passive decoupler	A decoupler just without ESC (this is the only EtherCAT P device without ESC)

3 EtherCAT P specification and documents

EtherCAT P has been included into the EtherCAT standards. EtherCAT has been described by means of the ISO/OSI layer models. Hence, the EtherCAT P specifications can be navigated in the same manner.

Table 3 provides an overview of EtherCAT P-related specifications and documents. This also includes the application note serving as a practical EtherCAT P implementation guide on a doing level – also marked as “top reading” below.

Table 3: EtherCAT P information, standards and references

	Subject	Documents, Description and Access
Introduction	EtherCAT Compendium	Section II: Technology Details, chapter 3 Data link layer (DLL) describes the basic concept of EtherCAT P, and how it relates to 100BASE-TX → http://www.ethercat.org/compendium
	Articles	EtherCAT P has been introduced in several articles. A selection of them is given here. → PC Control (English): 01/2016 | (German): 01/2016 (stronger technical footprint) → PC Control (English): 01/2016 | (German): 01/2016 (system view)
	Proceedings of ETG events	Minutes of the Technical Committee Meetings give additional background information. EtherCAT P was introduced on the spring meeting 2016. Meeting minutes from then on are of specific EtherCAT P interest. → www.ethercat.org → Downloads → Select Filter: Proceedings and Papers → Technical Committee Meeting
Specifications	Communication slides	The communication slides provide a broad description of EtherCAT mechanisms for developers. It also describes some basics on the physical layer, which, naturally, include some EtherCAT P basics, too. → English
	Application note TOP READING	While the EtherCAT specifications describe the EtherCAT P technology in a more formal context, the application note aims to give very practical guidance on a doing-level. It includes details for both, PSD and PD. This includes EtherCAT P schematic details, electronic components, layout recommendations for grounding/ EMI/ EMC/ layout and examples. → Application Note: EtherCAT P Implementation Guide www.ethercat.org/ethercatp
Specifications	EtherCAT P specification	The main EtherCAT P specification document. It describes: Voltages, system architecture, device types, powered devices, power sourcing devices, passive components, device categories, physical layer extension, cables, connectors. It refers to related specifications → ETG.1030 www.ethercat.org/ethercatp
	EtherCAT P connector	Specification of M8 P-coded connector. Any cable manufacturer can produce and sell such a connector. The M8 P-coded connector has also been submitted for IEC standardization. → ETG.1030.1 www.ethercat.org/ethercatp
	EtherCAT P physical layer extension	EtherCAT P physical layer specifics. → ETG.1000.2 P www.ethercat.org/ethercatp
	EtherCAT SubDevice Information (ESI)	Description of EtherCAT P-specific details in the ESI file, such as EtherCAT P device type (PD, PSE), min/max voltages and load types. → ETG.2000 (EtherCAT P): www.ethercat.org/ethercatp The related schema file is also available for download → EtherCATInfo.xsd (and related xsd files): www.ethercat.org/ethercatp

	Subject	Documents, Description and Access
	ETG.9001 Marking Rules	As with EtherCAT and Safety over EtherCAT, for EtherCAT P logo and trademark are defined. This and their usage are specified in ETG.9001. → ETG.9001 (EtherCAT P): www.ethercat.org/ethercatp

4 EtherCAT P conformance testing

4.1 General

Since EtherCAT P combines power and data on the same cable, a faulty implementation might influence the whole system. Furthermore, wrong power consumption entries in the ESI file may lead to network configurations that do not work reliably, since the planning tool relies on this data.

Therefore, for EtherCAT P devices the physical layer test is mandatory. EtherCAT P enhancement in the ESI file is tested with the default test set included in the CTT.

In the introductory phase the EtherCAT P physical layer test is available in Germany and Japan – **free of charge** – retesting as well!

4.2 Evaluate your current EtherCAT SubDevice for EtherCAT P

To evaluate your current SubDevice's EtherCAT interface to be used as basis for an EtherCAT P interface PCB, the EtherCAT Test Centre (ETC) in Germany provides a set of test adaptors. It is sufficient to make a first pragmatic test on specific operating situation of the PHY: The CTT test file executed in the set-up shown in Figure 11 allows to exclude a communication drawback caused by the baseline wander (BW) effect. It will also check for communication issues induced by the power supplies.

The test requires a CU2508, 2 EtherCAT P test adapters and the CTT. The test file TF-1000 checks the functionality and is used with three different cable lengths. The Figure 11 shows the testing setup with a 75 m long cable.

To borrow such a set including TF-1000 contact ETC Germany (etc@beckhoff.com).

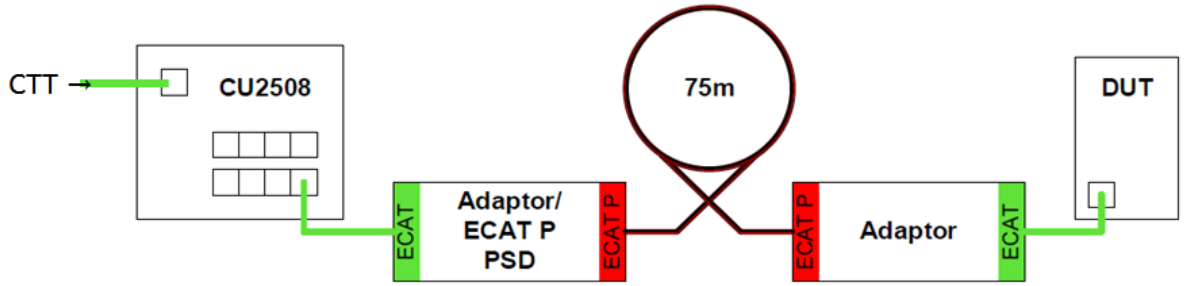


Figure 11: Baseline wander testing with a 75m cable

4.3 Contact for EtherCAT P conformance testing

As with EtherCAT and Safety over EtherCAT, contact conformance@ethercat.org for EtherCAT P conformance testing.

5 EtherCAT P licensing

5.1 General

Like EtherCAT, EtherCAT P is a protected technology – this helps to ensure compatibility and interoperability. This concept has proven to be very successful with EtherCAT and therefore is also applied to EtherCAT P: Implementing EtherCAT P in products requires a license. Again, as with EtherCAT itself, Beckhoff as inventor of EtherCAT P is supporting and encouraging the widespread adoption of EtherCAT P. Therefore, the license for EtherCAT P is free of charge. For interoperability reasons, EtherCAT P may only be used with specified connectors.

The hybrid connectors will be licensed separately (to connector makers). Users and device vendors do not need an additional license for using them.

5.2 License agreement

EtherCAT P licensing is particularly simple if you have already signed an EtherCAT Technology Family License Agreement with Beckhoff - Beckhoff then provides a side letter. Newly issued license agreements already include EtherCAT P.

Contact licensing@beckhoff.com regarding EtherCAT P.

6 EtherCAT P implementation aspects

The previous chapters have provided all the basic insight and references to implement an EtherCAT P device. Of course, all EtherCAT implementation related steps remain. Sections I and Section II provide comprehensive information on it.

Regarding the EtherCAT P specific part, no matter if started with a new device from scratch or enabling an existing EtherCAT device with EtherCAT P, the implementation of the EtherCAT P specific part goes along the following few steps:

ETG membership

5 License agreement
6 Study application note (Section III, chapter 3) and EtherCAT P specifications
7 Design EtherCAT P specific PCB along application note
8 Use already available EtherCAT P devices for pragmatic functionality testing
9 To configure the test network, use an EtherCAT configuration tool supporting the configuration of EtherCAT P networks
10 Update/use label and trademark term (Figure 12)
11 Contact conformance@ethercat.org for EtherCAT P conformance testing (optionally, and recommended, also for EtherCAT conformance testing

Figure 12: EtherCAT P logo

6.1 EtherCAT P and EtherCAT configuration tool

The EtherCAT configuration tools task is to generate a network description, standardized as EtherCAT Network Information (ENI) within ETG. It describes the topology, all EtherCAT SubDevices with their assigned EtherCAT address, the initialization commands for each SubDevice and the cyclic commands to exchange cyclic input and output data between MainDevice and SubDevices. All this remains unchanged. In fact, no change at all is necessary on the EtherCAT configuration tool to run EtherCAT P SubDevices in a network. Power-supply must be guaranteed, as with any other fieldbus SubDevice.

As mentioned earlier, the configuration tools may include functionality to calculate and assess the power consumptions in the EtherCAT P segments to simplify planning. The configuration tool can verify if the daisy-chained power is sufficient for each individual EtherCAT P SubDevice and its connected loads.

Once the EtherCAT power consumption calculations have been finalized, the actual EtherCAT network configuration as described above can be done and the MainDevice can run the EtherCAT / EtherCAT P network without even knowing of EtherCAT P details.

Figure 13 shows how an EtherCAT P planning tool is integrated into the EtherCAT network configuration tool to verify the power consumption of the EtherCAT P segments.

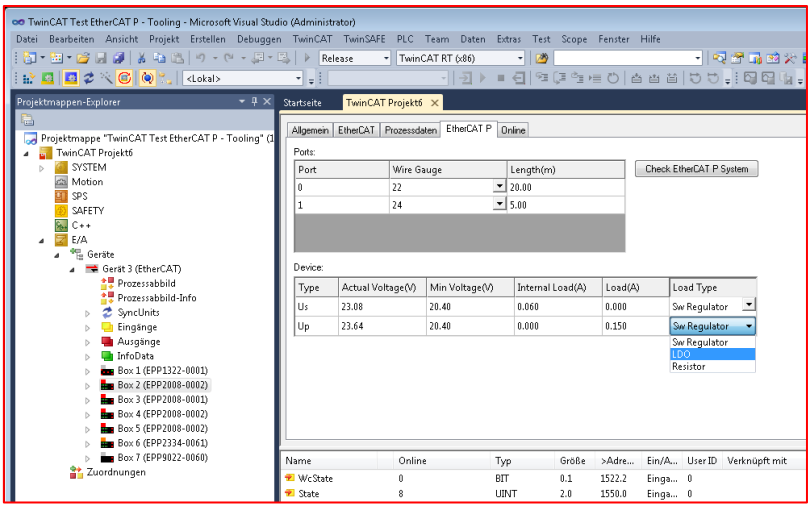


Figure 13: EtherCAT P planning tool integrated into a network configuration tool (Beckhoff)

A table shows if the power supply meets the power consumption of each individual EtherCAT P SubDevice (Figure 14). Also, the load and load types of each EtherCAT P SubDevice can be configured.

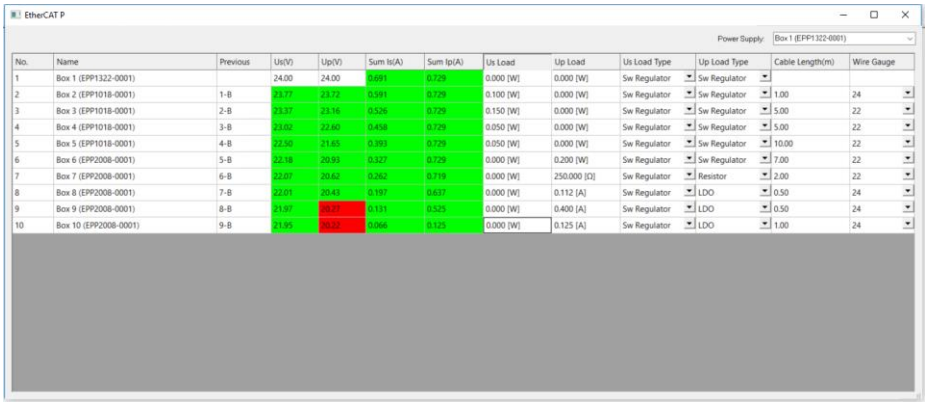


Figure 14: Showing if power-supply is sufficient or not

7 EtherCAT P development support

7.1 EtherCAT and EtherCAT P training

For the complete list of trainings and workshops, see Section I, chapter 6.1. A dedicated training for EtherCAT P is not listed/available. Questions can always be addressed to techinfo@ethercat.org.

7.2 Technical support

Technical support throughout the development process is provided by the EtherCAT Technology Group predominantly by the headquarters in Germany, but also by the various ETG offices worldwide (depending on local capacity). If you need direct contact, address your specific question to ETG (techinfo@ethercat.org).

EtherCAT EtherCAT P Safety over EtherCAT

SubDevice Implementation Guide

SECTION IV – Safety over EtherCAT introduction and implementation

Safety over EtherCAT technology introduction, Safety over EtherCAT specifications and documents, licensing, conformance testing, implementation

EtherCAT Technology Group

CONTENTS

1 Introduction IV-24
2 Safety over EtherCAT technology IV-25
2.1 Overview IV-25
2.2 Documents for detailed information and further reading IV-26
3 Technology users IV-28
3.1.1 Machine builders IV-28
3.1.2 Standard EtherCAT MainDevice manufacturer IV-28
3.1.3 FSoE device manufacturer IV-29
4 Safety over EtherCAT implementation aspects IV-30
4.1 FSoE device structure IV-30
4.2 Hardware architecture IV-30
4.3 Software architecture IV-30
4.4 Safety manual IV-31
5 Safety over EtherCAT Licensing IV-32
5.1 General IV-32
5.2 License Agreement IV-32
6 Safety over EtherCAT conformance testing IV-33
6.1 FSoE Test Cases IV-33
6.2 FSoE Conformance Test Tool for FSoE devices IV-33
6.3 FSoE Conformance Test IV-34
6.3.1 FSoE SubInstance Conformance Test IV-34
6.3.2 FSoE MainInstance Conformance Test IV-35
7 Safety over EtherCAT development support IV-37
7.1 EtherCAT and Safety over EtherCAT training IV-37
7.2 Technical support IV-37
7.3 Step by step implementation for an FSoE device manufacturer IV-37
8 Frequently asked questions IV-38

TABLES

Table 1: Standards and References IV-26
Table 2: Test executions depending on the Sub-/MainInstance type IV-34
Table 3: Normative FSoE connections necessary for the complete coverage IV-35

FIGURES

Figure 1: FSoE system architecture IV-25
Figure 2: Decentralized safety logic approach with standard PLC IV-28
Figure 3: Devices with FSoE interface IV-29
Figure 4: Hardware architecture IV-30
Figure 5: Software architecture IV-31
Figure 6: FSoE device assessment and approval IV-33
Figure 7: FSoE Conformance Test Tool IV-34
Figure 8: Safe Inputs and Safe Outputs relative to the FSoE SubInstance IV-35
Figure 9: FSoE MainInstance (MainDevice) connected to FSoE CTT with an EtherCAT bridge IV-36

1 Introduction

This document describes from a very practical point of view which topics have to be kept in mind for successful usage and/or implementation of the Safety over EtherCAT Technology. It considers the following questions:

- What are the requirements for a machine builder, EtherCAT MainDevice manufacturer or Safety device manufacturer?
- What kind of information and documentation is available?
- How to start with an implementation?
- Where can I get technical support?
- Is a conformance test available?

The EtherCAT Technology Group will not assume any responsibility or liability if a manufacturer of a Safety over EtherCAT device is infringing safety standards or regulations.

All responsibilities for the proper application of Safety over EtherCAT Technology, i.e. the development, the creation and certification of safe products in whole or in part including the safety risk and hazard analysis and classification, remains with the device manufacturer.

2 Safety over EtherCAT technology

2.1 Overview

Safety over EtherCAT (FSoE) describes a protocol for transferring safety data up to SIL3 between FSoE devices. FSoE frames are cyclically transferred via a subordinate fieldbus that is not included in the safety considerations, since the subordinated fieldbus can be regarded as a black channel. The FSoE frames exchanged between two communication partners are regarded as process data by the subordinated fieldbus.

FSoE uses a unique MainInstance/SubInstance relationship between the FSoE MainInstance and a FSoE SubInstance; it is called FSoE connection (Figure 1). In the FSoE connection, each device only returns its own new message once a new message has been received from the partner device. The complete transfer path between FSoE MainInstance and FSoE SubInstance is monitored by a separate watchdog timer on both devices, and in each FSoE cycle.

The FSoE MainInstance can handle more than one FSoE connection to support several FSoE SubInstances.

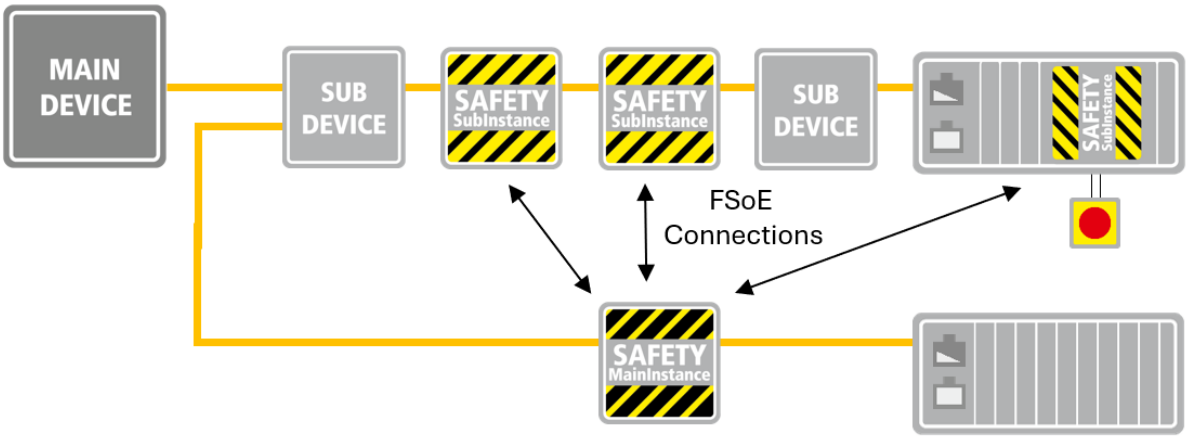


Figure 1: FSoE system architecture

The integrity of the safety data transfers is ensured as follows:

- Session number for detecting buffering of a complete startup sequence
- Sequence number for detecting interchange, repetition, insertion or loss of whole messages
- Unique connection identification for safely detecting misrouted messages via a unique address relationship
- Watchdog monitoring for safely detecting delays not allowed on the communication path
- Cyclic redundancy checking for data integrity for detecting message corruption from source to sink

State transitions are initiated by the FSoE MainInstance and acknowledged by the FSoE SubInstance. The FSoE state machine also involves exchange and checking of parameters for the communication relation.

The FSoE state machine is a separate state machine and runs on top of the EtherCAT State Machine (ESM).

Black channel approach

FSoE protocol is implemented using a black channel approach; there is no safety related dependency to the standard communication interface. The communication interface including controllers, ASICs, links, couplers, etc. remains standard.

The communication path is arbitrary; it can be a fieldbus system, Ethernet or other networking technologies based on fiber optics, copper wires or even wireless transmission. There are no restrictions or requirements on bus coupler or other devices in the communication path.

2.2 Documents for detailed information and further reading

Table 1 lists the relevant documents for the Safety over EtherCAT technology.

Table 1: Standards and References

Document	Description	Reference
ETG.5100	**Safety over EtherCAT Specification** FSoE protocol specification approved by TÜV.	Available per email send request to ETG (info@ethercat.org)
IEC 61784-3	**IEC specification of FSoE protocol** IEC 61784-3: Industrial communication networks - Profiles – Part 3: Functional safety fieldbuses, defines general requirements for functional safety fieldbuses. Functional Safety Communication Protocol FSCP 12/1 defines the Safety over EtherCAT technology. This part has the same content as ETG.5100.	www.iec.ch
ETG.5120	**Safety over EtherCAT Specification Enhancements** This specification contains enhancements of the Safety over EtherCAT protocol. These enhancements are part of the Safety over EtherCAT specification and shall be considered for device implementation.	www.ethercat.org/etg5120
FSoE license	**Safety over EtherCAT license** Safety over EtherCAT is a registered trademark and patented technology licensed by Beckhoff Automation GmbH & Co. KG. Beckhoff has assured that it is willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. **The license is available free of charge**. Beckhoff offers a license agreement.	Send request to Beckhoff (licensing@beckhoff.com)
Safety over EtherCAT Conformance Test		
ETG.9100	**Safety over EtherCAT Policy** Rules and requirements for using and implementing Safety over EtherCAT technology. The objective of this specification is to maintain the integrity of both EtherCAT and Safety over EtherCAT (FSoE). All requirements defined in the ETG.9100 that are applicable for a device shall be fully met.	www.ethercat.org/etg9100
ETG.7100 series	**Safety over EtherCAT Conformance Test Specification** The ETG.7100 series consists of following parts:	www.ethercat.org/etg7100
ETG.7100.1	**ETG.7100.1: General Requirements** defines the FSoE test in which the conformance of the FSoE device under test with the FSoE specification is tested	www.ethercat.org/etg7100
ETG.7100.2-2	**ETG.7100.2-2: SubInstance Test Record** A set of test instructions for the performance of the FSoE SubInstance Conformance Test and documentation of it at the same time. The document includes an informative test execution guide.	www.ethercat.org/etg7100
ETG.7100.2-3	**ETG.7100.2-2: MainInstance Test Record** A set of test instructions for the performance of the FSoE MainInstance Conformance Test and documentation of it at the same time. An informative execution guide is available in a separate document.	www.ethercat.org/etg7100
ETG.7100.3	**ETG.7100.3: FSoE test cases specification** Comprehensive test list for FSoE MainInstance and FSoE SubInstances (Excel sheet) Approved by TÜV	Comes with FSoE Conformance Test Tool (ET9402, ET9403)
ET9402	**Safety over EtherCAT SubInstance Conformance Test Tool** • Automatic test tool for FSoE SubInstance devices • Mandatory for approval of FSoE SubInstances. (The tool is offered by Beckhoff. Test cases are defined in ETG TWG Safety)	Send request to Beckhoff (your local representative)
ET9403	**Safety over EtherCAT MainInstance Conformance Test Tool** • Automatic test tool for FSoE MainInstance devices • Mandatory for approval of FSoE MainInstances. (The tool is offered by Beckhoff. Test cases are defined in ETG TWG Safety)	Send request to Beckhoff (your local representative)

Document	Description	Reference
Safety over EtherCAT profile specifications		
ETG.5001.4	**Modular Device Specification – Part 4: MDP Safety Module Specification** Standardized module profiles for FSoE digital I/O devices, FSoE drives and FSoE MainInstance devices	www.ethercat.org/etg5001
ETG.6100	**Safety over EtherCAT Drive Profile** Profile for adjustable speed electrical Power Drive Systems (PDS) that are suitable for use in Safety-Related (SR) application with Safety over EtherCAT protocol	www.ethercat.org/etg6100
Safety over EtherCAT training		
FSoE_Seminar.pdf	**Safety over EtherCAT seminar presentation** • Basic of safety networks and international standards • Safety over EtherCAT technology • Technical implementation aspects • Safety drive profile • Benefits for the user	http://www.ethercat.org/download/safety_seminar/default.asp
Important standard EtherCAT specifications, further standards: www.ethercat.org → Downloads		
ETG.1000	**EtherCAT Specification** EtherCAT Data link layer and application layer specification	www.ethercat.org/etg1000
ETG.2000	**EtherCAT SubDevice Information (ESI) Schema and Specification** Describes the structure of the EtherCAT SubDevice description in XML format. FSoE related Parts are included.	www.ethercat.org/etg2000
ETG.2100	**EtherCAT Network Information (ENI) Schema and Specification** Describes the structure of the EtherCAT network information description in XML format. Parts for Copy Information (SubDevice-to-SubDevice communication) are included	www.ethercat.org/etg2100
ETG.2200	**EtherCAT Implementation Guide (this document)** Describes from a very practical point of view which topics have to be kept in mind for a successful EtherCAT implementation	www.ethercat.org/etg2200

3 Technology users

According to different use cases different users of the FSoE technology can be distinguished:

- Machine builder:
  builds a machine with COTS devices including FSoE devices
- EtherCAT MainDevice manufacturer:
  vendor of non-safety-related control systems (MainDevice and/or IO devices).
  Integration of COTS FSoE devices in the control architecture is required.
- FSoE device manufacturer:
  vendor of safety-related devices with FSoE interface

3.1.1 Machine builders

A machine builder or system designer who uses devices with the Safety over EtherCAT technology has the responsibility to perform a safety risk and hazard analysis and classification for his machine and to ensure a continuous safety-chain.

All devices connected to a safety communication system shall fulfill the separated (or Safety) Extra-Low Voltage / Protective Extra-Low Voltage system requirements, which are specified in the relevant IEC standards, such as IEC 60204-1.

The resulting safety-function response time must fit to the application.

3.1.2 Standard EtherCAT MainDevice manufacturer

A vendor of a non-safety-related control system (e.g., standard PLC) with an EtherCAT interface (EtherCAT MainDevice) can support the usage of FSoE devices within the EtherCAT network. The MainDevice operates the bus; the FSoE Logic is integrated in an FSoE MainInstance device that is an EtherCAT SubDevice, as shown in Figure 2.

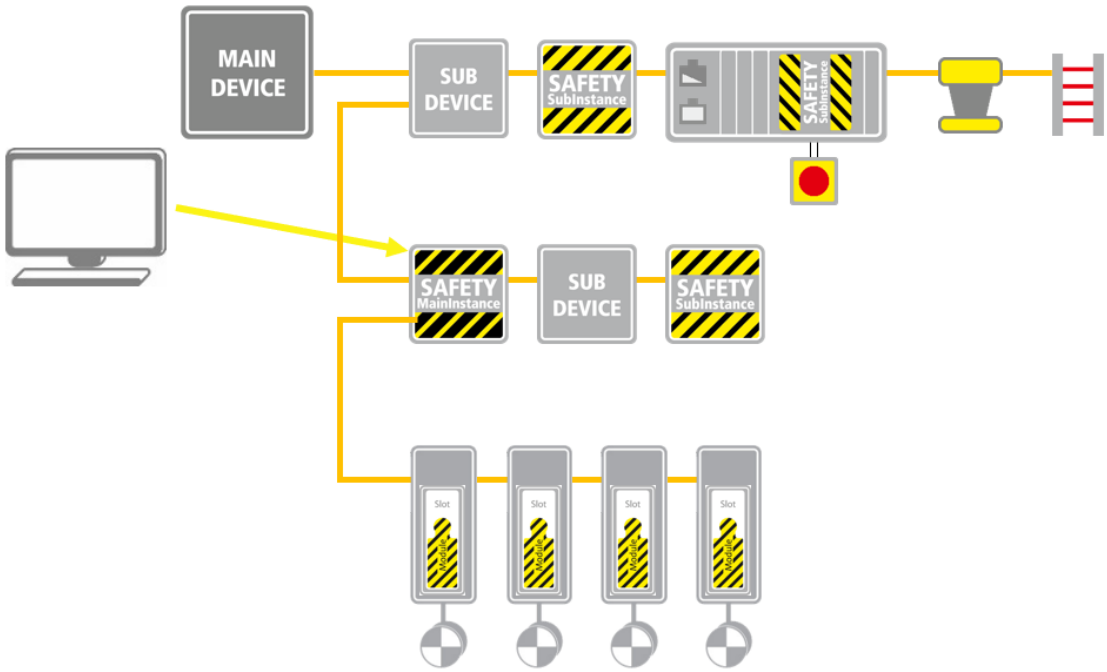


Figure 2: Decentralized safety logic approach with standard PLC

Requirements for the EtherCAT MainDevice:

- Support SubDevice-to-SubDevice communication
  Copy the safety frames from the FSoE MainInstance to the FSoE SubInstances, and vice versa.
  The copy information is part of the ENI (ETG.2100) file.
- Support an interface for the configuration tool of the FSoE logic device.

3.1.3 FSoE device manufacturer

The device manufacturer shall implement the Safety over EtherCAT protocol and the safety application according to the related safety standards (Figure 3). It is mandatory that the implementation is approved by a notified body.

The Safety over EtherCAT policy ETG.9100 defines rules and requirements for using and implementing the Safety over EtherCAT technology.

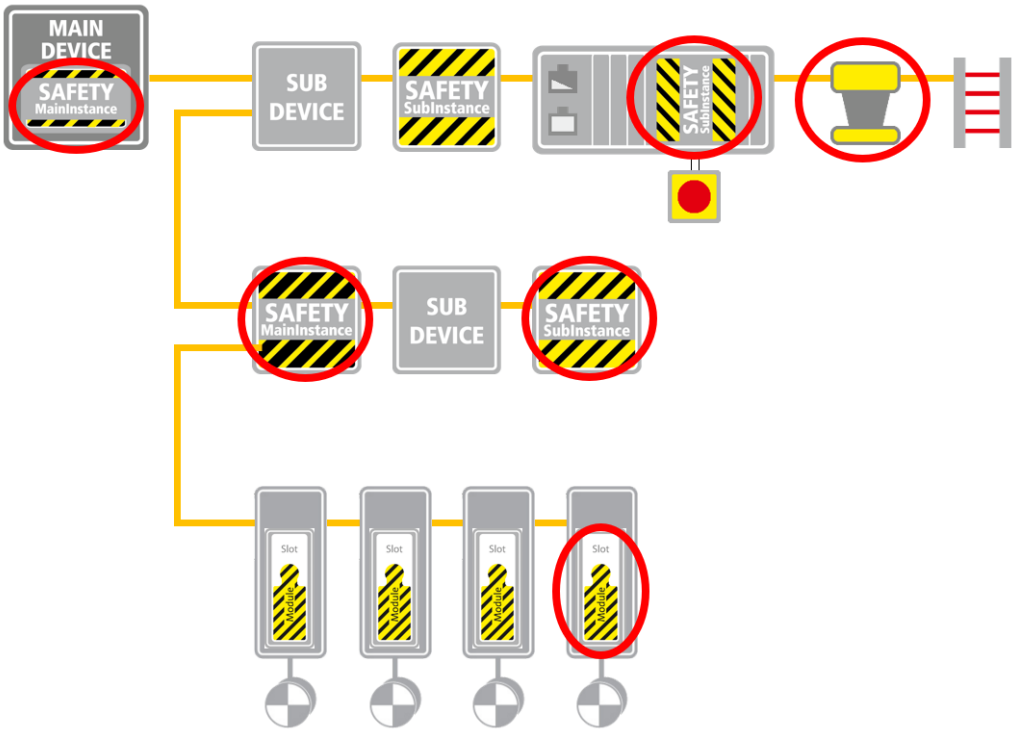


Figure 3: Devices with FSoE interface

The implementation of FSoE devices requires an FSoE license, described in chapter 5.

See next chapter 4 for implementation details.

4 Safety over EtherCAT implementation aspects

4.1 FSoE device structure

The ETG.5100 Safety over EtherCAT specification comprises a protocol specification for a safety-related data transfer up to SIL3. It *does not* define a particular hardware architecture or software design.

The report of the protocol approval demands an implementation that fulfills the following requirements:

- Complete fulfillment of IEC 61508 and IEC 61784-3
- Complete fulfillment of the FSoE protocol specification (ETG.5100)
- Implementation must fulfill the requirements of the claimed safety level and corresponding product-specific requirements.

The ETG.9100 FSoE policy defines further rules and requirements for using and implementing the Safety over EtherCAT technology. All requirements defined in the ETG.9100 that are applicable for a device shall be fully met.

4.2 Hardware architecture

According to the black channel approach the communication hardware in a device can remain single channel, i.e. the standard EtherCAT SubDevice Controller (ESC) for the EtherCAT interface can be used.

EtherCAT or any other communication interface like an internal backbone can be used.

For the processing of the FSoE protocol *usually* redundant microcontroller architecture is needed (Figure 4). Each microcontroller calculates the Safety over EtherCAT protocol; the results are cross-checked.

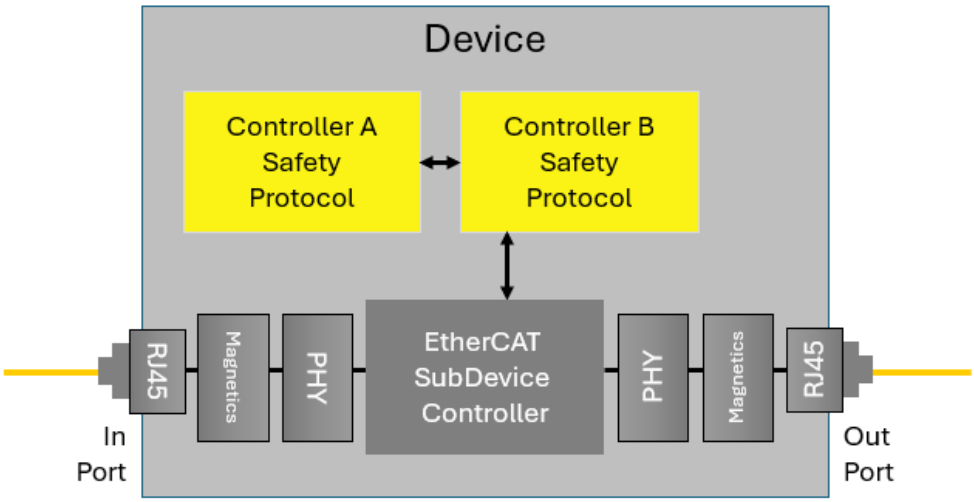


Figure 4: Hardware architecture

4.3 Software architecture

The FSoE protocol is processed upon the application layer of the communication interface (Figure 5).

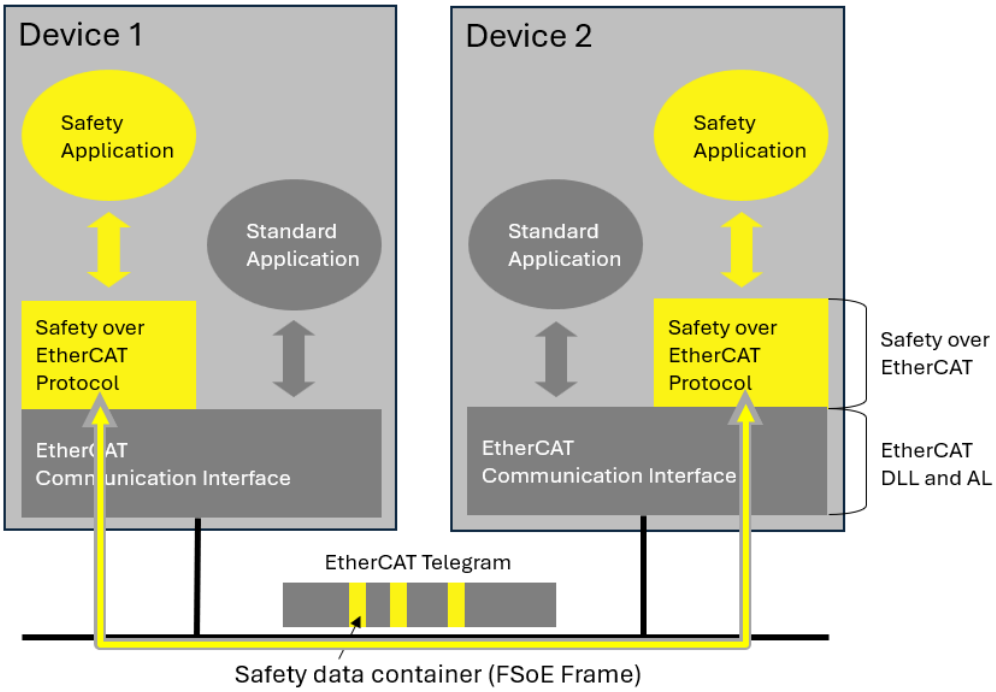


Figure 5: Software architecture

For a safety-related software environment several self-test functions (e.g., memory tests, controller tests and peripheral tests) must be performed to detect dangerous errors. These requirements are outside the scope of the FSoE protocol – see IEC 61508 or appropriate product specific standards.

4.4 Safety manual

Implementers shall supply a safety manual, but meeting the following points at a minimum:

- The safety manual shall inform the users of constraints for calculation of system characteristics.
- The safety manual shall inform the users of their responsibilities of proper parameterization of the device.

In addition to the requirements of this clause the safety manual shall follow all requirements in the FSoE policy and IEC 61508.

5 Safety over EtherCAT Licensing

5.1 General

Like EtherCAT, Safety over EtherCAT is a protected technology – this helps to ensure compatibility and interoperability. This concept has proven to be very successful with EtherCAT and therefore is also applied to Safety over EtherCAT: Implementing Safety over EtherCAT in products requires a license. Again, as with EtherCAT itself, Beckhoff as inventor of Safety over EtherCAT is supporting and encouraging the widespread adoption of Safety over EtherCAT. Therefore, the license for Safety over EtherCAT is ***free of charge***.

5.2 License Agreement

Safety over EtherCAT licensing is simple if you have already signed an “EtherCAT Technology Family License Agreement” with Beckhoff - Beckhoff then provides a side letter. Newly issued License Agreements already include Safety over EtherCAT.

Contact licensing@beckhoff.com regarding Safety over EtherCAT.

6 Safety over EtherCAT conformance testing

The implementation of the FSoE protocol in a device must meet the Safety over EtherCAT specification requirements. For the device approval the procedure and requirements described in the Safety over EtherCAT policy ETG.9100 and in the FSoE Conformance Test specification ETG.7100 shall be fulfilled.

The FSoE policy defines the overall assessment and approval procedure of FSoE devices according to Figure 6.

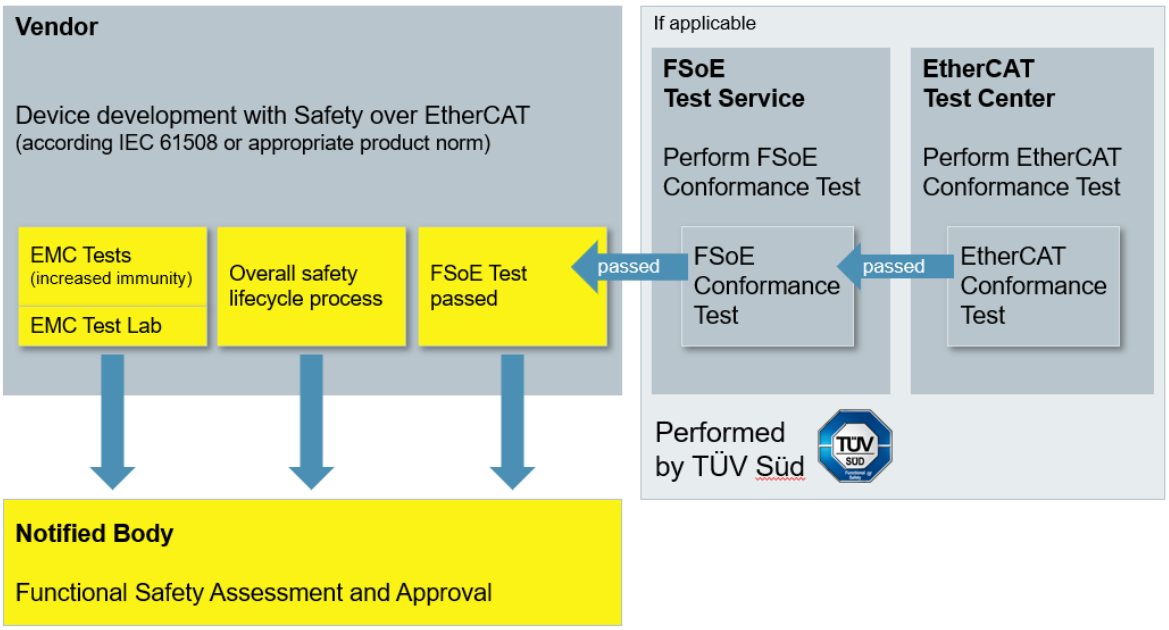


Figure 6: FSoE device assessment and approval

The approval of a FSoE device is done with an in-house test and within an FSoE Test Center.

6.1 FSoE Test Cases

The ETG.7100.3 defines a comprehensive and exhaustive list of test cases for FSoE MainInstance and FSoE SubInstance devices. The vendor is responsible to integrate those tests in its overall test plan and shall perform and pass those tests for the FSoE device release.

The test cases are approved by TÜV.

6.2 FSoE Conformance Test Tool for FSoE devices

The FSoE Conformance Test Tool (FSoE CTT) allows checking the protocol compliance of an FSoE device to the FSoE specification. FSoE CTT is to be used during validation of devices supporting an FSoE interface. The FSoE CTT shall be used for in-house testing in the device manufacturer's test lab and is used for official FSoE Conformance Test at an FSoE Test Center.

Figure 7 shows how the Conformance Test Tool for FSoE tests works. The FSoE CTT is approved by TÜV.

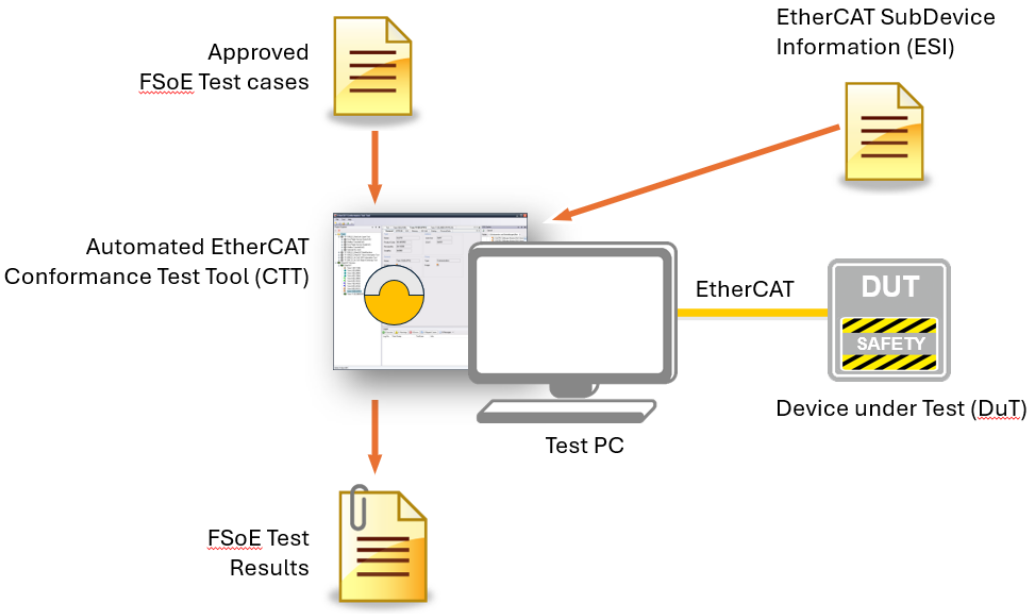


Figure 7: FSoE Conformance Test Tool

Table 2 shows possible test executions, which can be used depending on the MainInstance and SubInstance type.

Table 2: Test executions depending on the Sub-/MainInstance type

	FSoE MainInstance	FSoE SubInstance
EtherCAT MainDevice	FSoE CTT for FSoE MainInstances using an EtherCAT bridge device	ETG.7100.3 SubDevice tests incorporated in vendors' test environment
EtherCAT SubDevice	FSoE CTT for FSoE MainInstances	FSoE CTT for FSoE SubInstances
Non-EtherCAT device	ETG.7100.3 MainInstance tests incorporated in vendors' test environment	ETG.7100.3 SubDevice tests incorporated in vendors' test environment

If the DUT has any other communication interface, a connection via a gateway might be possible. This option should be used to run the automated tests with the CTT, if appropriate.

6.3 FSoE Conformance Test

A conformance test of the safety protocol implementation is available for FSoE MainInstance and FSoE SubInstance.

6.3.1 FSoE SubInstance Conformance Test

For FSoE SubInstances that are EtherCAT SubDevices the corresponding test cases are available within the Conformance Test Tool (CTT). These devices shall additionally pass a test in an official EtherCAT Test Center (ETC) including:

- EtherCAT Conformance Test as a prerequisite
- FSoE Conformance Test

The device vendors shall use the FSoE SubInstance conformance test record ETG.7100 Part 2-2 for validation of conformance. The FSoE test record is a set of test instructions for the performance of the FSoE conformance test and documentation of it at the same time. It is also used in a FSoE test center.

6.3.2 FSoE MainInstance Conformance Test

For FSoE MainInstances that are EtherCAT MainDevices or EtherCAT SubDevices the corresponding test cases are available within the Conformance Test Tool (CTT). The implementation is approved by TÜV. These devices shall additionally pass a test in an official EtherCAT Test Center (ETC) including:

- EtherCAT Conformance Test*
- FSoE Conformance Test

*The EtherCAT Conformance Test is not applicable for FSoE MainInstances that are EtherCAT MainDevice.

The device vendors shall use the FSoE MainInstance conformance test record ETG.7100 Part 2-3 for validation of conformance. The FSoE test record is a set of test instructions for the performance of the FSoE conformance test and documentation of it at the same time. It is also used in a FSoE test center.

An overview of the 7 normative FSoE connection configurations that defined for the MainInstance Conformance Test are shown in Table 3.

Table 3: Normative FSoE connections necessary for the complete coverage

No	SafeInput* [Byte]	SafeOutput* [Byte]	Safe Application Parameter [Byte]	Rationale
1	2	1	0	Code coverage
2	2	4	0	Code coverage
3	1 (minimum)	1 (minimum)	0	Minimum length Safe-In/Out
4	252 (maximum supported)	252 (maximum supported)	0	Maximum length Safe-In/Out
5	2	4	9 (not word aligned)	Check content of FSoE connection parameter set
6	1	1	256 (maximum supported)	Maximum length of FSoE connection parameter set
7	1	1	SRA CRC SRA Parameter	Check SRA CRC calculation of FSoE MainInstance configuration tool

*The lengths of Safe Inputs/Safe Outputs are understood relative to the SubInstance operated by the MainInstance (see Figure 8).

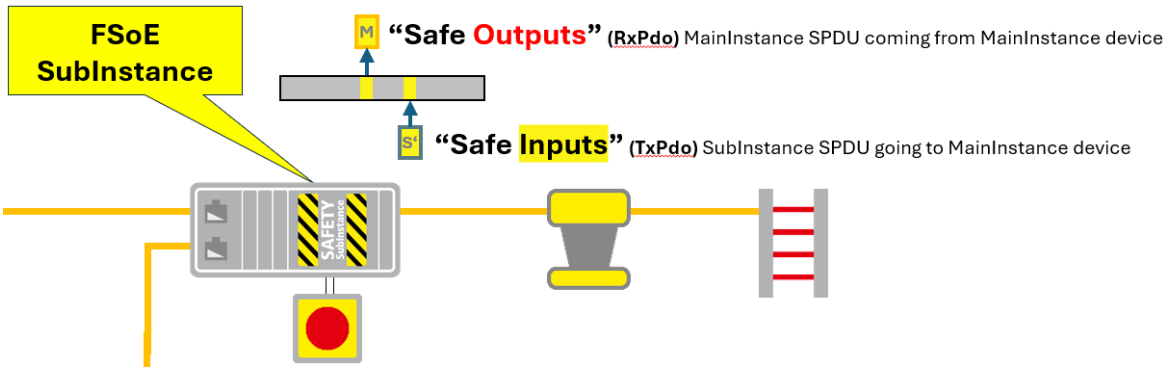


Figure 8: Safe Inputs and Safe Outputs relative to the FSoE SubInstance

For the test execution the following normative ESI files are provided:

- ETG7100_FSoE_Modules.xml
  Module file with normative FSoE connection configurations
- ETG7100_FSoE_SubInstance.xml
  FSoE SubInstance ESI template file
- ETG7100_FSoE_ECAT-Bridge_Template.xml
  ESI template file for EtherCAT bridge device EL6695

If a connection is not supported by the MainInstance implementation contact ETG via conformance@ethercat.org. ETG will provide a tailored version of the ESI file that meets the DuT requirements. When contacting the ETG provide the following information:

- Minimum length Safe Input*
- Minimum length Safe Output*
- Maximum length Safe Input*
- Maximum length Safe Output*
- Maximum length FSoE Connection Parameter Set
- Support of SRA parameters? Yes/No
- Other restrictions?

State either the length of the “raw SafeData” without FSoE Cmd, CRCs and FSoE ConnID OR the length of the complete safety container.

The FSoE MainInstances that are an EtherCAT MainDevice are connected to the FSoE CTT by using an EtherCAT bridge device EL6695 from Beckhoff Automation (see Figure 9).

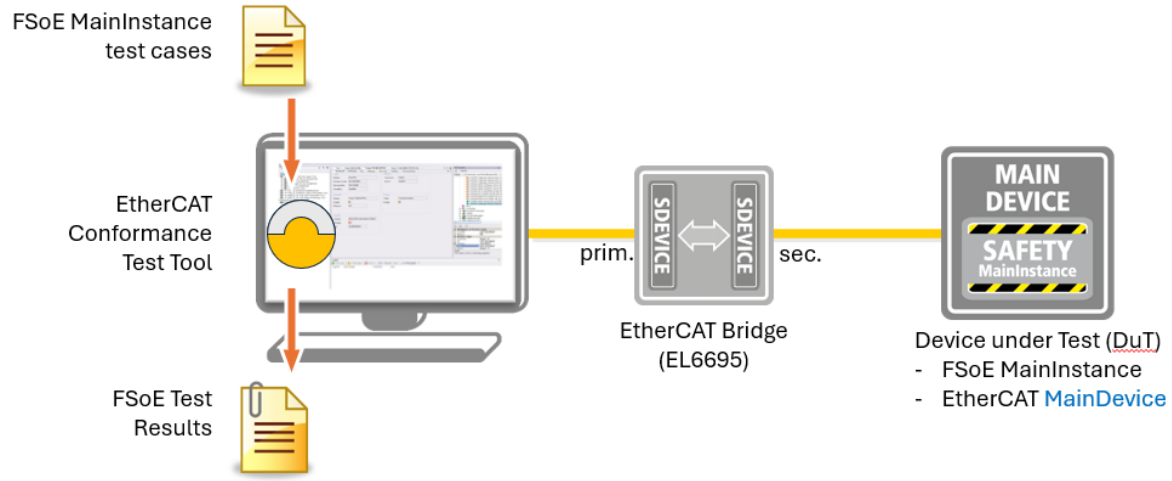


Figure 9: FSoE MainInstance (MainDevice) connected to FSoE CTT with an EtherCAT bridge

The EL6695 must be programmed with the normative FSoE connection configurations. The programming is done by the ETG. Contact conformance@ethercat.org to arrange the programming of your EL6695 with the normative FSoE connections.

7 Safety over EtherCAT development support

7.1 EtherCAT and Safety over EtherCAT training

For the complete list of trainings and workshops, including Safety over EtherCAT see Section I, chapter 6.1.

7.2 Technical support

Technical support throughout the development process is provided by the EtherCAT Technology Group predominantly by the headquarters in Germany, but also by the various ETG offices worldwide (depending on local capacity). If you need direct contact, address your specific question to ETG (techinfo@ethercat.org).

7.3 Step by step implementation for an FSoE device manufacturer

The following approach of implementing FSoE for an existing device might look like:

- Get an overview of the Safety over EtherCAT technology
  www.ethercat.org/safety
- Attend the Safety over EtherCAT seminar
  (for dates see www.ethercat.org > Events)
- Download all relevant documentation (see Table 1)
- In addition, take care at least of the following safety standards:
  - IEC 61508 and IEC 61784-3
- Get a free of charge Safety over EtherCAT license (send email to info@ethercat.org)
- Use FSoE Conformance Test cases for the conformance test and FSoE CTT for FSoE SubInstances to test your device with the latest FSoE features implemented.
- System test, interoperability test (e.g., at an EtherCAT Plug Fest)
- FSoE SubInstances shall be tested in a FSoE test center
- Approve your integration by a notified body (see chapter 6)

8 Frequently asked questions

1 Do I need a redundant EtherCAT interface within my Safety over EtherCAT device?

No.
The Safety over EtherCAT protocol is implemented using a black channel approach. There is no safety-related dependency to the standard communication interface. The communication interfaces such as controllers, ASICs, links, couplers, etc. remain unmodified.

2 Do I need redundant controller architecture for my Safety over EtherCAT device?

Usually yes.
Usually means, that common solutions use two microcontrollers. In fact, this is not demanded by the Safety over EtherCAT specification. A protocol implementation must fulfill following requirements:

- Complete fulfilment of IEC 61508 and IEC 61784-3
- Complete fulfilment of the FSoE protocol specification
- Complete fulfilment of the claimed safety level and corresponding product-specific requirements.

3 Can I use Safety over EtherCAT via other communication systems than EtherCAT?

Yes.
Since the beginning in 2005 Safety over EtherCAT was open and independent of the underlying bus system. The communication path is arbitrary. The communication path is arbitrary; it can be a fieldbus system, Ethernet or other networking technologies based on fiber optics, copper wires or even wireless transmission. There are no restrictions or requirements on bus couplers or other devices in the communication path.

4 Is there a certified Safety over EtherCAT stack available?

Yes.
Within the ETG there are service providers available offering pre-certified FSoE protocol stacks and safety development services.

ETG does not offer such kind of stack, because the Safety over EtherCAT specification is quite lean and the protocol state machine is well defined. Experience shows that an implementation can be done in very short time – often shorter than to adapt a certified stack that is not changeable in existing software architectures.

5 Is a Safety over EtherCAT conformance test available?

Yes.
For FSoE devices a Safety over EtherCAT test case specification exists and is approved by TÜV. For FSoE SubInstance devices that are an EtherCAT SubDevice and for FSoE MainInstances that are either an EtherCAT SubDevice or an EtherCAT MainDevice the test cases are integrated in the FSoE Conformance Test Tool (CTT) so that an automated test can be performed.

The Safety over EtherCAT policy ETG.9100 includes the overall test procedure for a device approval.

6 Do I need an approval by a notified body (e.g., TÜV, BGIA) for my Safety over EtherCAT device?

Yes.
The development of a device using the Safety over EtherCAT technology shall be assessed. The device approval includes a passed EMC report, the Safety over EtherCAT conformance approval and the overall safety lifecycle process approval according to IEC 61508 or appropriate product standards. The assessment shall be done by a notified body.

7 Do I need to perform an official test at an FSoE test center for my device release?

Yes, for FSoE SubInstance devices that are an EtherCAT SubDevice and for FSoE MainInstances that are either an EtherCAT SubDevice or an EtherCAT MainDevice.
For EtherCAT SubDevices the FSoE device approval shall further include a passed test in an official EtherCAT Test Center. Precondition for the FSoE Conformance Test is a valid EtherCAT Conformance Tested certificate for the FSoE device.

All tests performed by the FSoE test center are available for preparation in-house.

8 Why do I need a license to use the Safety over EtherCAT protocol within my device?

Safety over EtherCAT is a technology that is used by many device manufacturers. For such a technology the most important issue is compatibility. This ensures the safety integrity according to the approved Safety over EtherCAT specification but also – and this is of same importance – interoperability in the field. With the license the device manufacturer gets the right to implement the technology – but he has to do this compatible to the specification. This rule is part of the license agreement.

Machine builders and control system providers who use off-the-shelf Safety over EtherCAT devices do not need a license.

9 How can I get and use the Safety over EtherCAT logo?

The Safety over EtherCAT logo can be obtained from the ETG download website. The Safety over EtherCAT logo shall only be used in accordance with the EtherCAT marking rules as published by the ETG.9001.

10 I'm an EtherCAT MainDevice vendor. How can I support Safety over EtherCAT devices?

If you just want to support off-the-shelf Safety over EtherCAT devices in the EtherCAT segment you do not need any safety-related implementation in the MainDevice. Safety over EtherCAT MainInstances with an EtherCAT SubDevice interface are available and can be used as safety logic devices.

Only SubDevice-to-SubDevice communication must be supported by the EtherCAT MainDevice to route the safety frames from the Safety over EtherCAT MainInstance to the Safety over EtherCAT SubInstances and vice versa.

11 I'm a machine builder. Do I need a license to use Safety over EtherCAT devices?

No.
You can use off-the-shelf Safety over EtherCAT devices in the machine without a license.

You have to take care of the resulting Safety Integrity Level (SIL) or Performance Level (PL). Relevant standards (IEC 62061, ISO 13849) or product standards as well as compliance to other relevant standards, like national and international legal requirements (e.g., directive of machinery, OSHA, UL etc.) must be fulfilled, of course.

DOCUMENT HISTORY

Version	Comment
1.0.0	Official release
1.1.0	Document revised Editorial changes ESC variants updated More implementation products added New documentation links
1.1.1	Editorial changes
1.1.2	Use of marking rules (ETG.9001) and indicator specification (ETG.1300) added
1.1.3	Editorial changes
1.1.4	Editorial changes
1.1.5	Editorial changes ETG.9003 Conformance Test Policy added ETG.9002 Vendor ID Policy added Minor changes in step by step implementation
1.1.6	Editorial changes
2.0.0	Document revised New document structure Enhanced general procedure - step by step Major content enhancements
2.0.1	Editorial changes Policies added Download links updated/fixed Minor content enhancements
2.0.3	Editorial changes
2.1.0	Contact email address is conformance@ethercat.org
2.1.1	Development products updated
2.1.2	Development products (Renesas, TESSERA) updated
2.1.4	Correction of trade mark term acc. To ETG.9001
2.1.5	Editorial changes
2.1.6	Added reference to ETG.5003.2
2.1.7	Editorial Changes, updates of hyperlinks and ESC overview Add and update chapters on Knowledge Base/Search/Download, support
3.0.0	Add EtherCAT P section
3.0.1	Development products (Profichip) updated
3.0.2	Add description on CTT license
3.0.3	Update of EtherCAT P specifications status (available as release)
3.0.4	Update of contact graphics Update of colors of some graphics
3.1.0	Add “Evaluate your current EtherCAT SubDevice for EtherCAT P” Add products Update abbreviations, links, references, layout
3.1.1	Update links, references Major content enhancements Add Section IV Safety over EtherCAT introduction and implementation Add section information in footer
3.2.0	Update EtherCAT terms to consider inclusive language New Abbreviations table Update section IV Safety over EtherCAT introduction and implementation and add FSoE MainInstance Conformance Test