

1 Overview

End users and/ or System integrators expect a defined minimum functionality and interoperability when selecting an EtherCAT Master device.

The ETG (EtherCAT Technology Group) has defined EtherCAT Master Classes (ETG.1500) with a well defined set of Master functionalities. In order to keep things simple only 2 Master Classes are defined:

- Class A: Standard EtherCAT Master Device
- Class B: Minimum EtherCAT Master Device

The principle idea is that each implementation should aim to meet Class A requirements. Only if resources prohibit, e.g. on embedded systems, at least Class B shall be met.

Additional Functionality, which can be considered to be optional, is described by Feature Packs. The Feature Pack describes all mandatory master functionality for a specific feature, e.g. Redundancy.



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

2 Core License

Feature ID: Unique identification used in ETG.1500 EtherCAT Master Classes

FP: Available as Feature Pack

*1: According to ETG.1500 Master Classes not mandatory for Class A

*2: According to ETG.1500 Master Classes not mandatory for Class B

Feature name	Short description	EC-Master Class A	EC-Master Class B	Feature ID
Basic Features				
Service Commands	Support of all commands	✓	✓	101
IRQ field in datagram	Use IRQ information from Slave in datagram header	✓	✓	102
Slaves with Device Emulation	Support Slaves with and without application controller	✓	✓	103
EtherCAT State Machine	Support of ESM special behavior	✓	✓	104
Error Handling	Checking of network or slave errors, e.g. Working Counter	✓	✓	105
VLAN	Support VLAN Tagging	✓	-- (*2)	106
EtherCAT Frame Types	Support EtherCAT Frames	✓	✓	107
UDP Frame Types	Support UDP Frames	-- (*1)	-- (*2)	108
Process Data Exchange				
Cyclic PDO	Cyclic process data exchange	✓	✓	201
Multiple Tasks	Different cycle tasks Multiple update rates for PDO	✓	✓	202
Frame repetition	Send cyclic frames multiple times to increase immunity	-- (*1)	-- (*2)	203
Network Configuration				
Online scanning	Network configuration functionality included in EtherCAT Master	✓	✓	301
Reading ENI	Network Configuration taken from ENI file			
Compare Network configuration	Compare configured and existing network configuration during boot-up	✓	✓	302
Explicit Device identification	Identification used for Hot Connect and prevention against cable swapping	✓	✓	303
Station Alias Addressing	Support configured station alias in slave, i.e. enable 2nd Address and use it	✓	✓	304
Access to EEPROM	Support routines to access EEPROM via ESC register	✓	✓	305
Mailbox Support				
Support Mailbox	Main functionality for mailbox transfer	✓	✓	401
Mailbox Resilient Layer	Support underlying resilient layer	✓	✓	402
Multiple Mailbox channels		✓	✓	403
Mailbox polling	Polling Mailbox state in slaves	✓	✓	404

Feature name	Short description	EC-Master Class A	EC-Master Class B	Feature ID
CAN application layer over EtherCAT (CoE)				
SDO Up/Download	Normal and expedited transfer	✓	✓	501
Segmented Transfer	Segmented transfer	✓	✓	502
Complete Access	Transfer the entire object (with all sub-indices) at once	✓	✓	503
SDO Info service	Services to read object dictionary	✓	✓	504
Emergency Message	Receive Emergency messages	✓	✓	505
PDO in CoE	PDO services transmitted via CoE	-- (*1)	-- (*2)	506
EoE				
EoE protocol	Services for tunneling Ethernet frames. includes all specified EoE services	✓	✓	601
Virtual Switch	Virtual Switch functionality	✓	✓	602
EoE Endpoint to Operation Systems	Interface to the Operation System on top of the EoE layer	FP (*1)	FP (*2)	603
FoE				
FoE Protocol	Support FoE Protocol	✓	-- (*2)	701
Firmware Up-/Download	Password, FileName should be given by the application	✓	-- (*2)	702
Boot State	Support Boot-State for Firmware Up/Download	✓	-- (*2)	703
SoE				
SoE Services	Support SoE Services	✓	✓	801
AoE				
AoE Protocol	Support AoE Protocol	✓	-- (*2)	901
VoE				
VoE Protocol	External Connectivity supported	✓	-- (*2)	1001
Synchronization with Distributed Clock (DC)				
DC support	Support of Distributed Clock	✓	-- (*2)	1101
Continuous Propagation Delay compensation	Continuous Calculation of the propagation delay	✓	-- (*2)	1102
Sync window monitoring	Continuous monitoring of the Synchronization difference in the slaves	✓	-- (*2)	1103
Slave-to-Slave Communication				
via Master	Information is given in ENI file or can be part of any other network configuration Copying of the data can be handled by master stack or master's application	✓	✓	1201
Master information				
Master Object Dictionary	Support of Master Object Dictionary (ETG.5001 MDP sub profile 1100)	FP (*1)	FP (*2)	1301

3 Feature Packs

3.1 FP Cable Redundancy with/without Distributed Clocks

3.1.1 Basic Functions

In case of cable break all types of EtherCAT communications (process data and mailbox protocols) shall be supported without any restrictions.

Handling of the following use cases:

- Normal operation
- Stay operational in case of cable break between two slaves
- Stay operational in case of cable break between primary port and first slave
- Stay operational in case of cable break between secondary port and last slave
- Stay operational in case of cable fixed
- Start/Stop (State change) in case of cable break
- Adjustment of Auto Increment address in case of cable break
- Frame loss in case of cable break (partner frame was not received)

3.1.2 Diagnosis Functions

Localization of cable break shall be possible (Number of slaves on each port)

Support function to check link status of primary port and of secondary port.

3.1.3 Redundancy with Hot Connect

Combination of feature packs Hot Connect and Cable Redundancy

3.2 FP Hot Connect with/without Distributed Clocks

3.2.1 Basic Functions

Handling of the following use cases:

- Differentiation between mandatory and “hot connect” (optional) slaves. Hot connect slaves are identified by using the configured IdentifyCmd.
- Bus can be transferred to OPERATIONAL state if “hot connect” slaves are missing.
- Add or remove additional slave devices during bus communication is in progress. Newly added slaves are automatically transferred into the OPERATIONAL state.
- Bus may remain operational if an hot connect slave fails.
- No wrong slaves must be connected (e.g. wrong address). If a wrong slave is connected, the bus stub must be cut-off at this point by the master application.
- The network information file (ENI) contains the parameters and process data for all slaves
- The network has to be configured in a way that all possible slaves are connected simultaneously, even if not all slaves can be connected at once.

3.2.2 Diagnosis Functions

The application is informed which slaves are currently connected. In case of removing or adding of slaves the application is informed about changes.

3.3 FP Master Object Dictionary

The Master Object dictionary contains information about the network configuration and EtherCAT slave diagnosis data. The details are defined in the document ETG.5001 MDP sub profile 1100.

3.4 FP TCP-Server and Remote API

The TCP-Server together with the Remote API library allows accessing EC-Master from a remote system. For example, access EC-Master which is running in VxWorks control system from a Notebook that is running Windows (e.g. read master or slave status).

3.5 FP EoE Endpoint

The interface allows sending and receiving Ethernet frames to/from the EtherCAT network. This may be used to implement a network driver on top of EC-Master.

4 Order information

4.1 Master Cores

Description	Order No.
EC-Master Class A, Core SDK, Windows CE, x86, 32 Bit	100-103-1-1
EC-Master Class A, Core SDK, Windows CE, ARM, 32 Bit	100-103-2-1
EC-Master Class A, Core SDK, RTX8-RTX2011, x86, 32 Bit	100-153-1-1
EC-Master Class A, Core SDK, RTX2012, x86, 32 Bit	100-157-1-1
EC-Master Class A, Core SDK, RTX64, x86, 64 Bit	100-158-1-2
EC-Master Class A, Core SDK, VxWorks, x86, 32 Bit	100-203-1-1
EC-Master Class A, Core SDK, VxWorks, ARM, 32 Bit	100-203-2-1
EC-Master Class A, Core SDK, VxWorks, PowerPC, 32 Bit	100-203-3-1
EC-Master Class A, Core SDK, Linux, x86, 32 Bit	100-253-1-1
EC-Master Class A, Core SDK, Linux, x86, 64 Bit	100-253-1-2
EC-Master Class A, Core SDK, Linux, ARM, 32 Bit	100-253-2-1
EC-Master Class A, Core SDK, Linux, PowerPC, 32 Bit	100-253-3-1
EC-Master Class A, Core SDK, Windows, x86, 32 Bit	100-303-1-1
EC-Master Class A, Core SDK, Windows, x86, 64 Bit	100-303-1-2
EC-Master Class A, Core SDK, INTime, x86, 32 Bit	100-353-1-1
EC-Master Class A, Core SDK, RTOS-32, x86, 32 Bit	100-403-1-1
EC-Master Class A, Core SDK, QNX, x86, 32 Bit	100-453-1-1
EC-Master Class A, Core SDK, T-Kernel, x86, 32 Bit	100-503-1-1
EC-Master Class A, Core SDK, TI Starterware, ARM, 32 Bit	100-703-2-1
EC-Master Class A, Core SDK, Xenomai, x86, 32 Bit	100-723-1-1
EC-Master Class A, Core SDK, NIOS EDS, 32 Bit	100-763-1-1

4.2 Feature Packs

Description	Order No.
EC-Master Class A, FP Hot Connect SDK	100-801-0-0
EC-Master Class A, FP Cable Redundancy SDK	100-811-0-0
EC-Master Class A, FP Remote Access API SDK	100-821-0-0
EC-Master Class A, FP EC-Engineer SDK	100-831-0-0
EC-Master Class A, FP EoE Endpoint SDK	100-851-0-0
EC-Master Class A, FP Master Object Dictionary SDK	100-861-0-0
EC-Master Class A, FP SuperSet ENI SDK	100-871-0-0

5 References

ETG Standards

- [1] ETG.1000.2: Physical Layer service definition and protocol specification
- [2] ETG.1000.3: Data Link Layer service definition
- [3] ETG.1000.4: Data Link Layer protocol specification
- [4] ETG.1000.5: Application Layer service definition
- [5] ETG.1000.6: Application Layer protocol specification
- [6] ETG.1005: EtherCAT Automation Protocol
- [7] ETG.1020: EtherCAT Guidelines and Protocol Enhancements
- [8] ETG.1500: EtherCAT Master Classes
- [9] ETG.2000: EtherCAT Slave Information
- [10] ETG.2100: EtherCAT Network Information
- [11] ETG.1400: EtherCAT Technology Description
- [12] ETG.5001: EtherCAT Modular Device Profiles
- [13] ETG.6010: EtherCAT Implementation Guideline for CiA402 Drive Profile
- [14] ETG.8000: EtherCAT Frequently asked Questions