Purpose of this document

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

This slide set intends to provide an overview over the diagnostic capabilities provided by EtherCAT.

It contains a description of the basic diagnosis functionalities and the most typical error scenarios within an EtherCAT network.

It is primarily intended for end users, as well as for machine builders and system integrators.

The knowledge of EtherCAT basics is taken for granted.

For additional information about EtherCAT diagnostics - including more detailed error scenarios – which could be of interest for EtherCAT master and slave manufacturers, please refer to slide set “EtherCAT Diagnosis For Developers”.

For comments regarding the slides please contact info@ethercat.org

Nuremberg, November 2018,

EtherCAT Technology Group

EtherCAT®

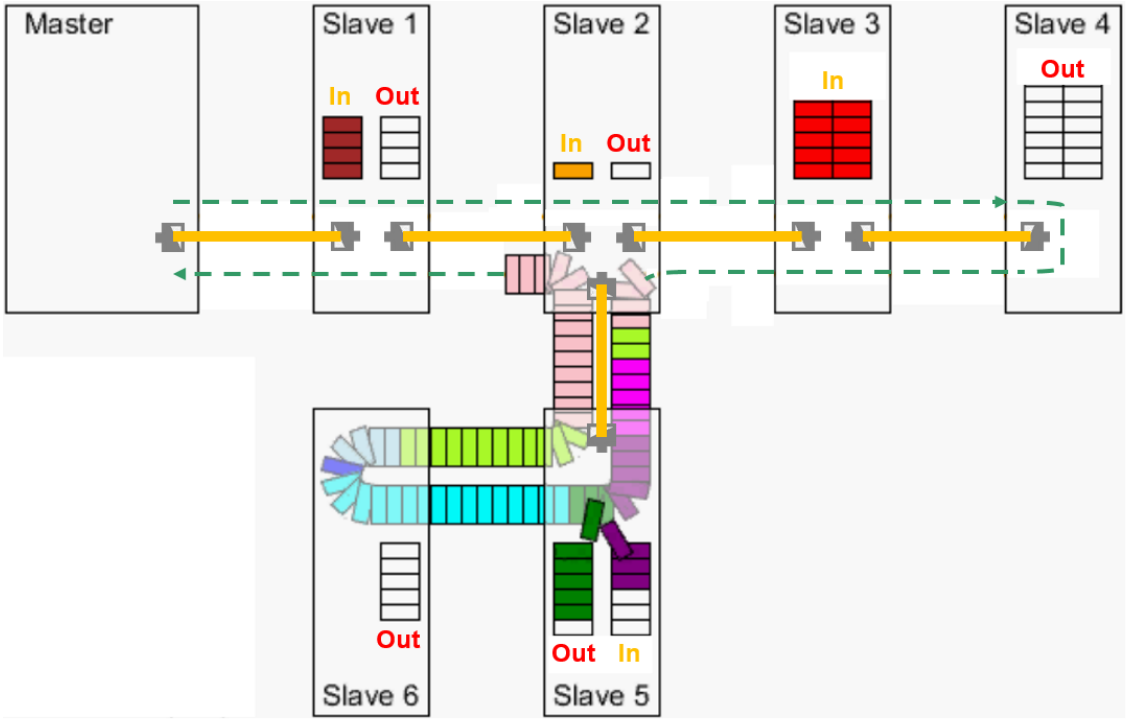
Diagnostic Features Overview

EtherCAT functional principle

In an EtherCAT network, information is exchanged by means of Ethernet frames, each one consisting of one or more datagrams.

Regardless of the hardware topology (line, daisy-chain, star, …), frames are always sent by the master, go through all slaves and return to the master after completing the „loop“.

Data carried by frames are processed by slaves „on-the-fly“.



Network Error Types

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

Errors which can affect an EtherCAT (like any other fieldbus) network can be grouped in **two categories**:

1. **Hardware errors**

   a. The physical medium is interrupted or the network topology is unexpectedly changed, and frames do not reach all the network slaves or do not return to the master at all (e.g. damaged cables, loose contacts, slave reset during operation).

   b. All slaves are reached by frames, but the correct bit sequence is corrupted (e.g. EMC disturbances, faulty devices).

2. **Software errors**

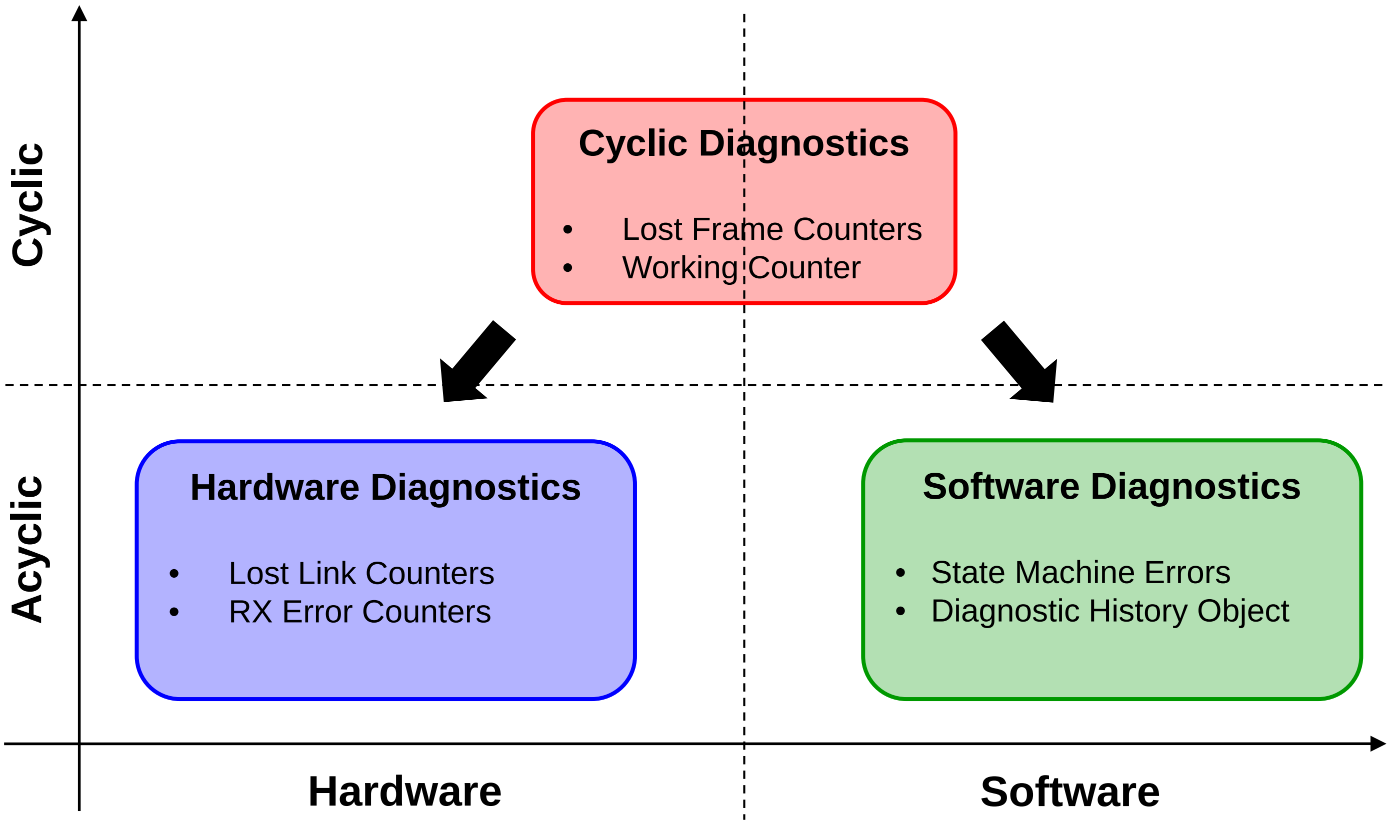
   a. The parameters sent by the master during the start-up phase are wrong or do not match the slave expectations (e.g. wrong process data size/configuration, unsupported cycle time).

   b. A slave previously working error-free detects an error during operation (e.g. synchronization loss, watchdog expiration).

EtherCAT diagnostic information overview

EtherCAT Diagnostic

EtherCAT provides extensive **diagnostic information** both at hardware and at software level. For the sake of simplicity, this diagnostic information can be classified according to the following scheme:



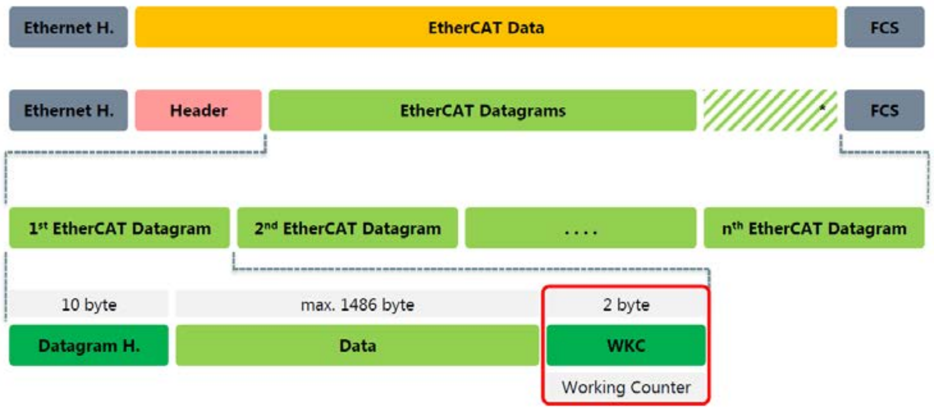
Cyclic Diagnostic

EtherCAT Technology Group

Working Counter

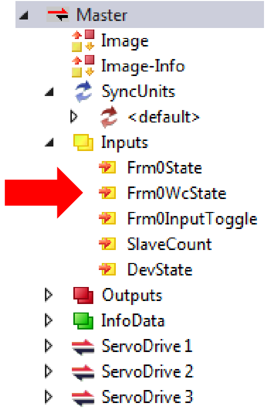
EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example



Each datagram in an EtherCAT frame ends with a 16-bit **Working Counter** (WKC), which is incremented by each slave addressed by the datagram itself. In case a datagram returns to the master with an invalid (= unexpected) WKC, the input data carried by that datagram are discarded by the master.

Master devices can optionally inform the control application (PLC, NC, …) about the Working Counter state (at least for datagrams carrying cyclic process data) by means of some cyclic variable in the network process image.

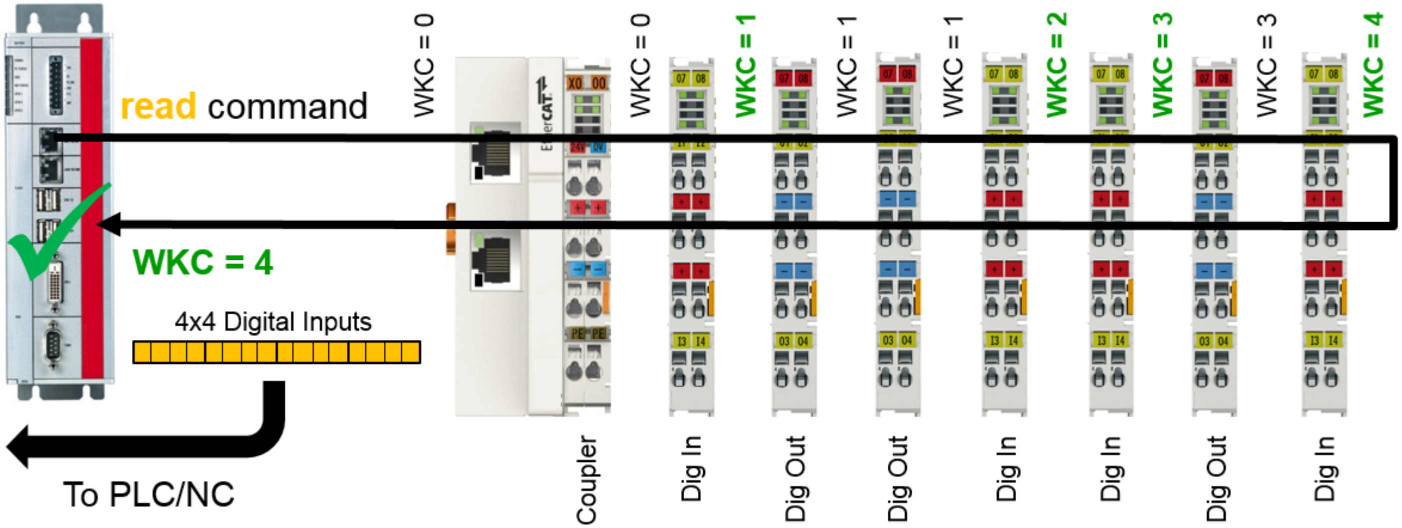


Working Counter – Example 1

All addressed slaves (Digital Inputs, in the example below) successfully process the datagram.

WKC value returning to the master = expected value → **WKC valid**

- ➢ input data in the datagram forwarded by the master to the control application (PLC, NC, …)

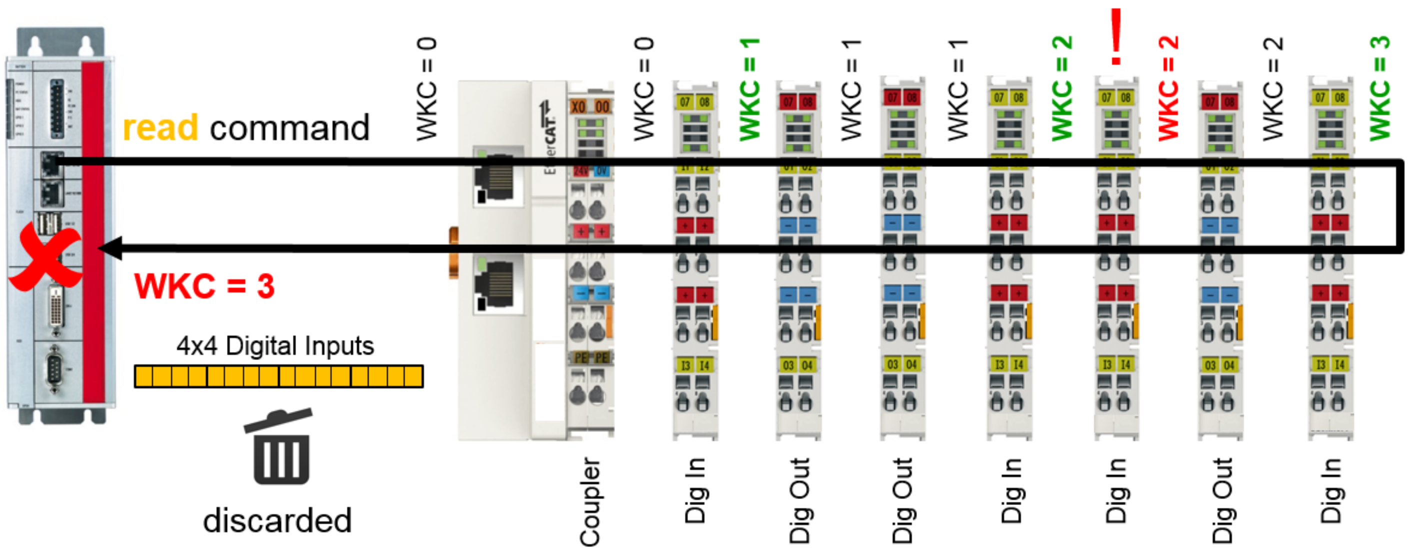


Working Counter – Example 2

One addressed slave (Digital Input, in the example below) fails to process the datagram.

WKC value returning to the master ≠ expected value → **WKC invalid**

- ➢ input data in the datagram are discarded by the master (PLC/NC uses old data)



Working Counter Summary

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

The Working Counter is always received by the master together with the corresponding datagram, and enables therefore an immediate reaction in case of invalid or inconsistent data.

The information concerning the Working Counter is basically a digital information ("WKC correct" vs. "WKC invalid"), and therefore does not distinguish among different error causes. An invalid WKC can result from several different situations:

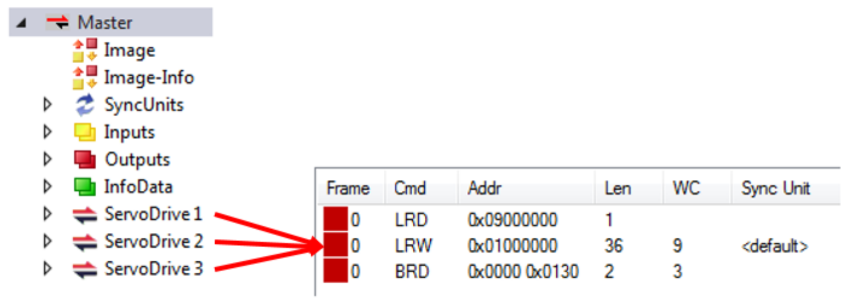
- One or more slaves are not physically connected to the network, or they are not reached by the frames.
- One or more slaves have been reset
- One or more slaves are not in Operational state

Whenever Working Counter errors occur, the problem should be investigated deeper by means of further Hardware Diagnostic and Software Diagnostic functionalities.

Sync Units

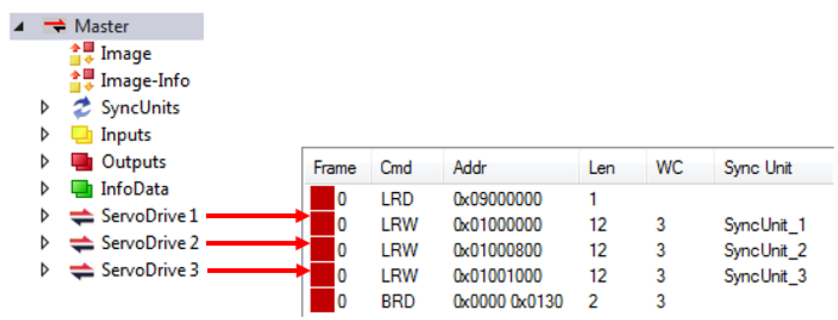
Masters can optionally enable to group network slaves into disjoint subsets called **Sync Units**. Slaves belonging to different Sync Units are served by separate datagrams, and therefore are also independent from the point of view of the Working Counter diagnostics.

- One (default) Sync Unit: if one drive fails incrementing the WKC, the input data of all three drives are discarded by the master:



Frame	Cmd	Addr	Len	WC	Sync Unit
0	LRD	0x09000000	1		
0	LRW	0x01000000	36	9	<default>
0	BRD	0x0000 0x0130	2	3	

- Separate Sync Units: if one drive fails incrementing the WKC, only the input data of that slave are discarded:



Frame	Cmd	Addr	Len	WC	Sync Unit
0	LRD	0x09000000	1		
0	LRW	0x01000000	12	3	SyncUnit_1
0	LRW	0x01000800	12	3	SyncUnit_2
0	LRW	0x01001000	12	3	SyncUnit_3
0	BRD	0x0000 0x0130	2	3	

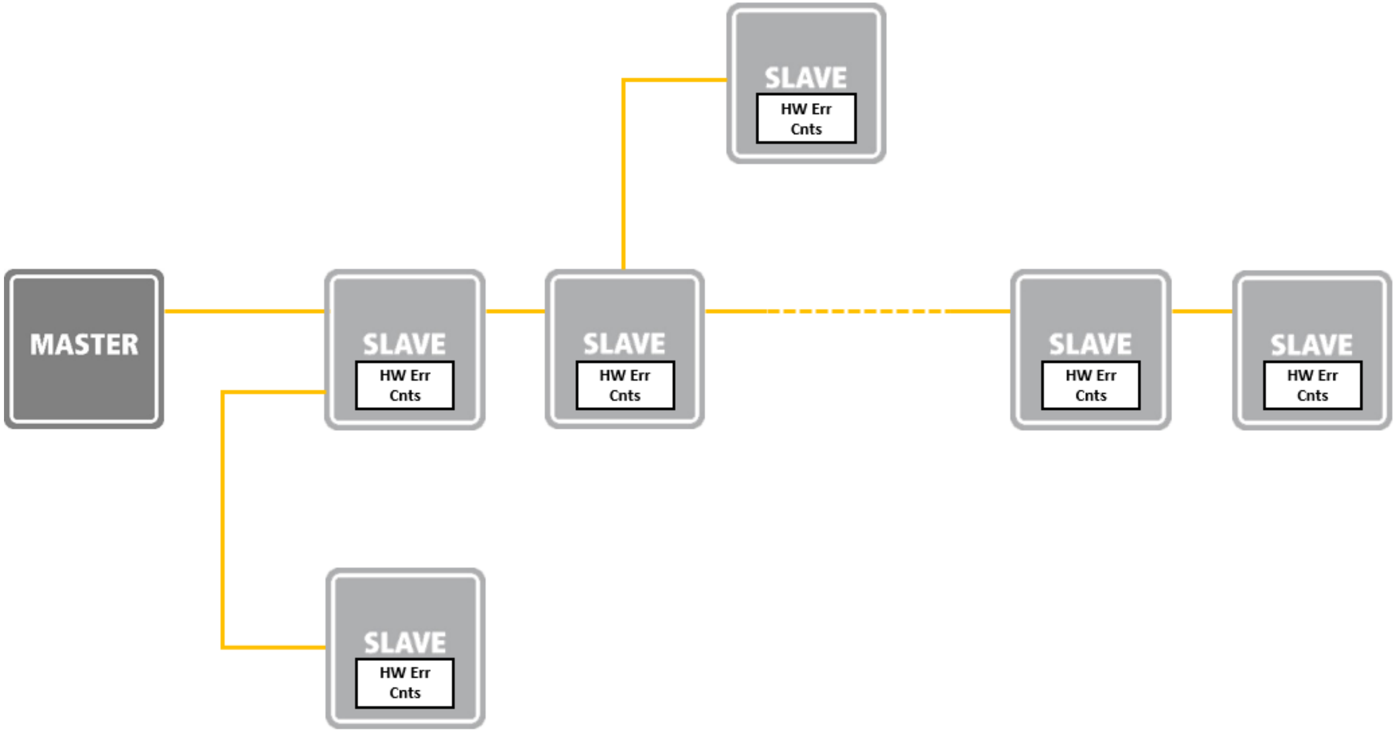
Hardware Diagnostic

Hardware Diagnostic

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

The basic diagnostic information at hardware level consists of error counters provided by slave devices at standard memory addresses.



These memory addresses can be accessed by the master device and be provided to the control application (for example by means of dedicated variables, or via function blocks in the PLC program).

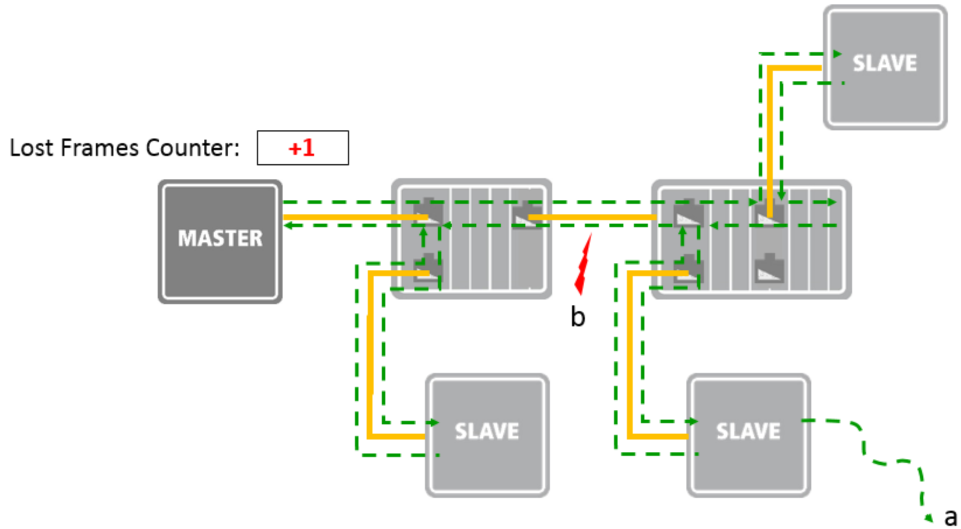
Master Lost Frames Counter

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

A frame shall be considered as „lost“ by the master either if it does not return to the master at all (a), or it is corrupted and therefore the information contained in it is meaningless (b).

Both situations can be monitored by the master by checking suitable fields of the incoming frames, and reported to the user by means of a corresponding **Lost Frame Counter**.



The master Lost Frame Counter can be considered as the first indicator of communication issues at hardware level in an EtherCAT network: an increment should trigger a deeper investigation by reading and interpreting Hardware Error Counters of slave devices.

Hardware Error Counters

- **Lost Link Counters** (optional): incremented when physical link is interrupted

Register	Length	Meaning	
0x0310	1 byte	Lost Link Counter port 0	
0x0311	1 byte	Lost Link Counter port 1	
0x0312	1 byte	Lost Link Counter port 2	
0x0313	1 byte	Lost Link Counter port 3	

- **RX Error Counters** (mandatory): incremented in case of signaling error:

Register	Length	Meaning	
0x0300	1 byte	Frame Error Counter port 0	Rx Error Counter port 0
0x0301	1 byte	Physical Layer Error Counter port 0	
0x0302	1 byte	Frame Error Counter port 1	Rx Error Counter port 1
0x0303	1 byte	Physical Layer Error Counter port 1	
0x0304	1 byte	Frame Error Counter port 2	Rx Error Counter port 2
0x0305	1 byte	Physical Layer Error Counter port 2	
0x0306	1 byte	Frame Error Counter port 3	Rx Error Counter port 3
0x0307	1 byte	Physical Layer Error Counter port 3	

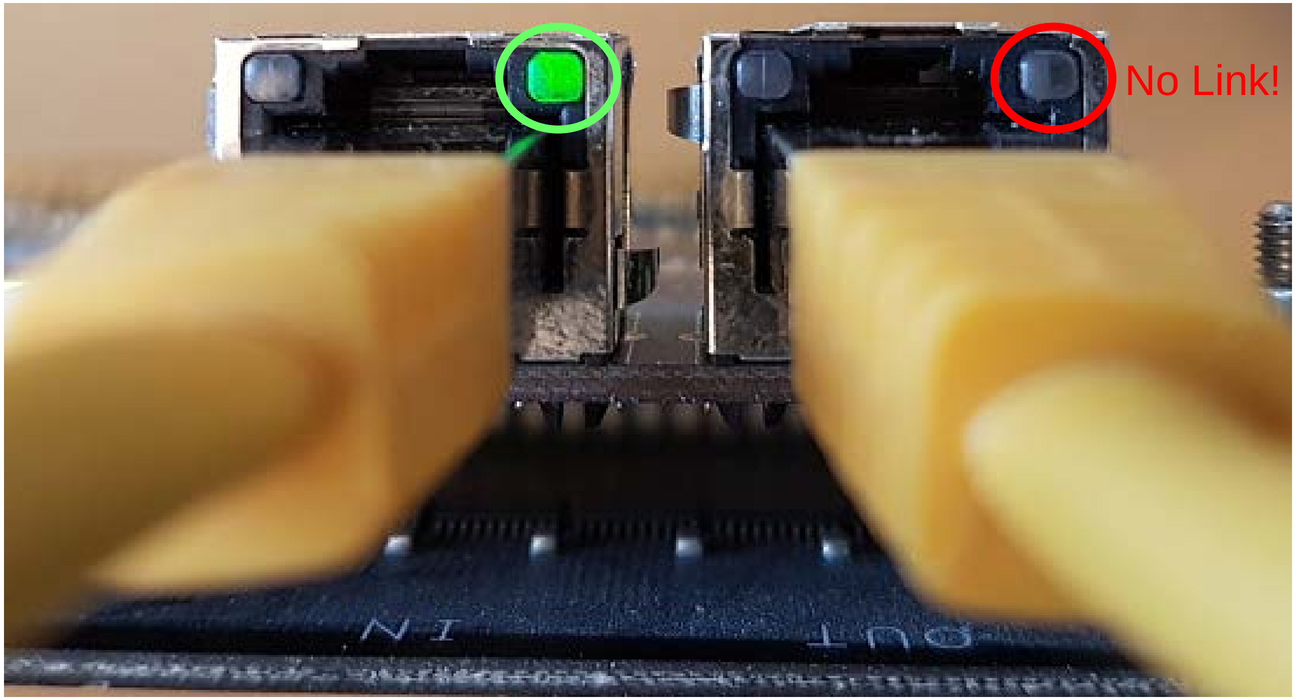
Link/Activity LEDs

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

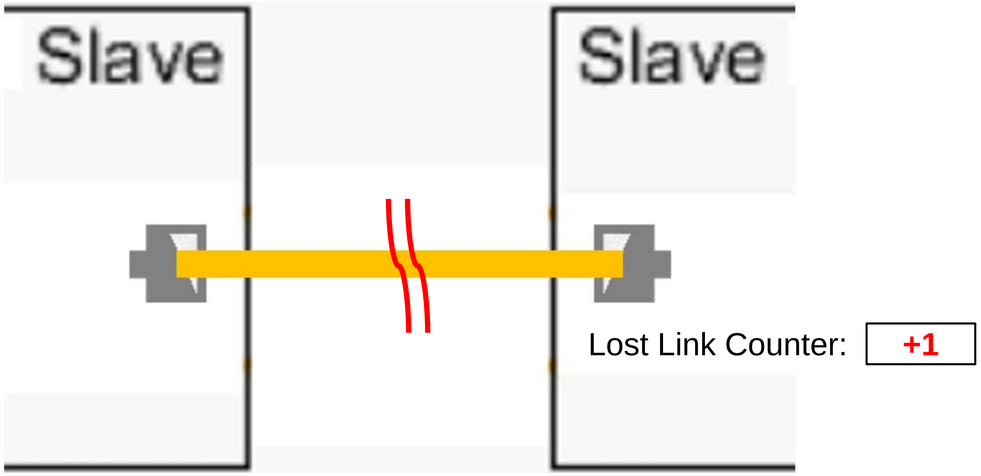
EtherCAT slave devices mandatorily support a Link/Activity LED for each port with removable connector.

Before checking Link Lost Counters (or for slaves which do not support Link Lost Counters at all), a visual inspection of Link/Activity LEDs can therefore easily enable to detect permanent interruptions of the physical link: in this case, the LED will be permanently off.



Lost Link Counters

An increment in a Lost Link Counter indicates an interruption in the hardware communication channel – during link down frames are not send to the neighboring device:



Most likely reasons for link loss are:

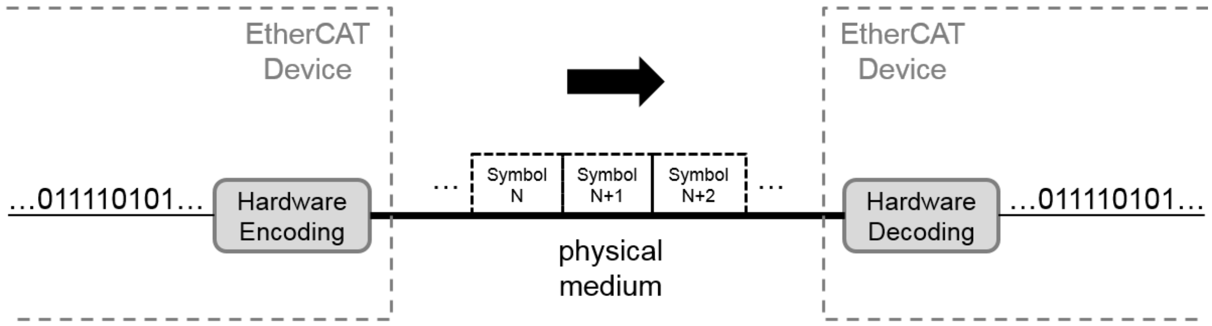
- Temporary or permanent device power-supply loss, or device reset.
- Damaged cables or connectors or poor/oxidized contacts
- EMC disturbances

Hardware Coding of Information

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

In order to be transmitted on a physical medium, digital information needs to be encoded (on transmitter side) and decoded (on receiver side) into specific current/voltage „symbols“.

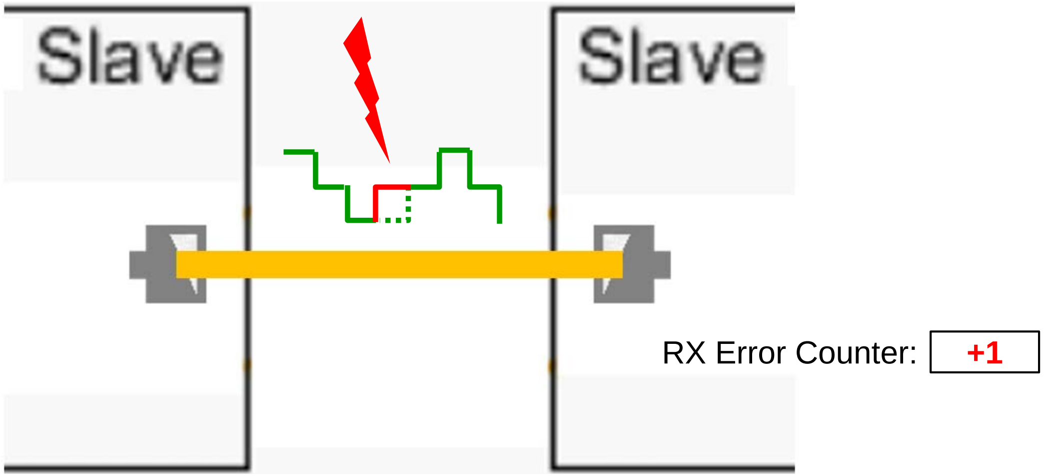


Coding results are dependent from the state of the link:

- The hardware coding defines **valid** and **invalid** symbols.
- Symbols are transmitted on the physical medium both **within** and **outside** frames (in order to enable the receiver to detect link losses).

RX Error Counters

A change of RX Error Counters indicates that the hardware signal received was corrupted and that the carried data will be discarded:



Most likely reasons for signal corruption are:

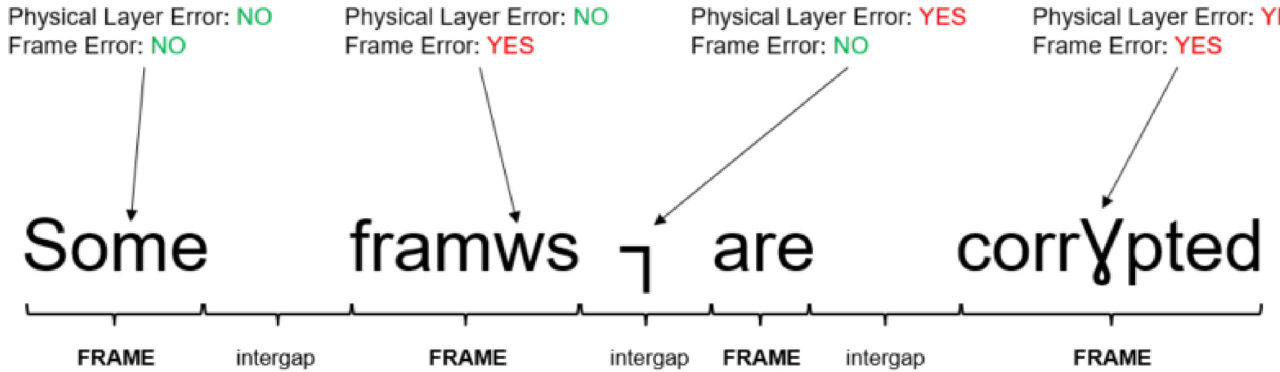
- External EMC disturbances (usually sporadic counter increment)
- Damaged devices or interconnections (usually fast and systematic counter increment)

Physical Layer and Frame Errors

Invalid Frame Counters report the following compound information:

- **Physical Layer Errors** (counted by Physical Layer Error Counters):
  - Correspond to individual invalid symbols
  - Can occur both within and outside frames (when occurring within frames, they represent usually also Frame Errors)
- **Frame Errors** (counted by Frame Error Counters):
  - Correspond to frames whose overall bit sequence was corrupted
  - Can occur only within frames

The difference between the two error types can be explained taking a written language as comparison:

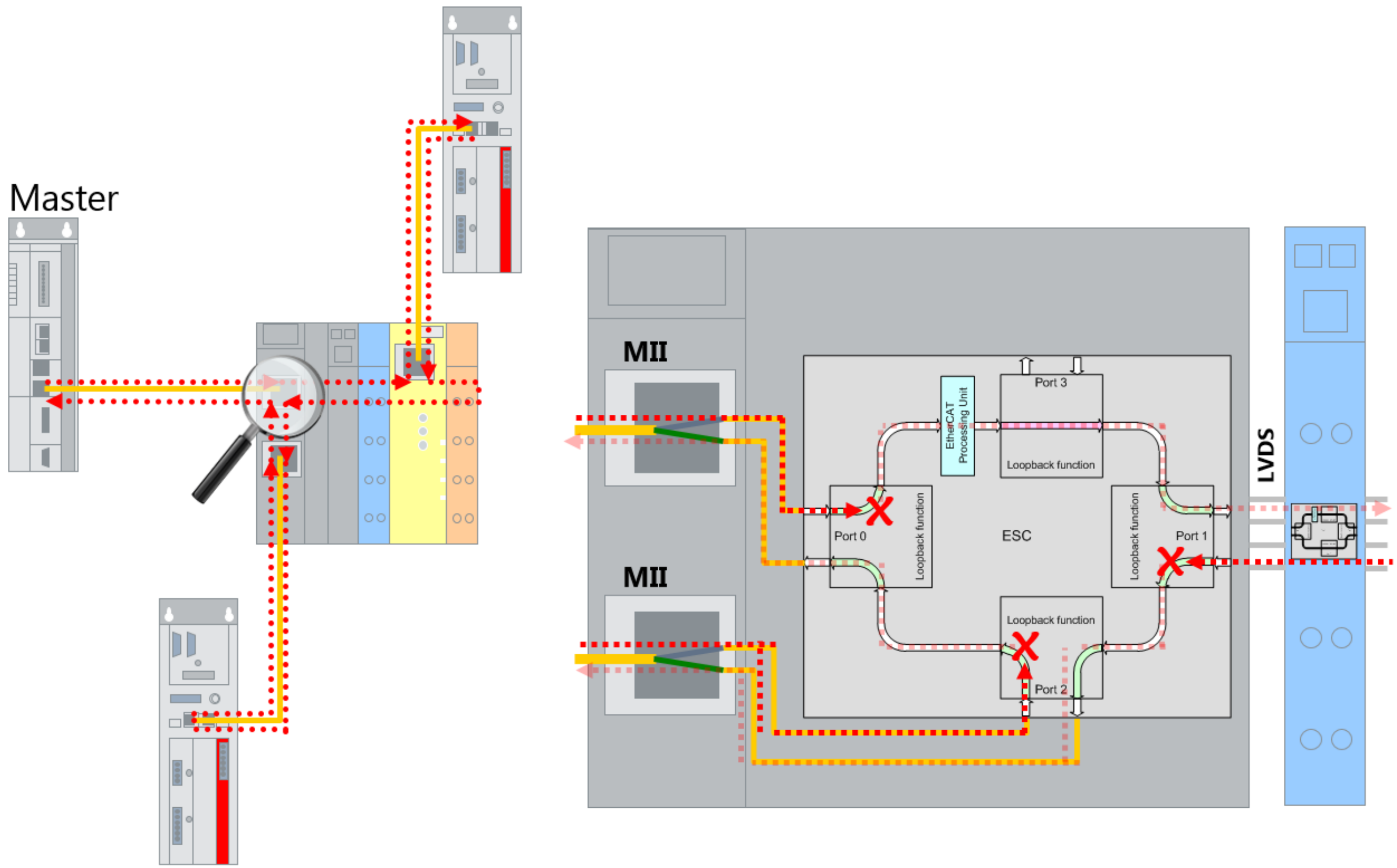


Frame Error Detection

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

In particular, Frame Errors are checked by each slave port (which in case increments the corresponding Frame Error Counter) when frames reach the port from the outside (x).



Comments on Hardware Errors

Some additional comments about hardware errors:

- Physical Layer Errors (and occasionally also Frame Frame) can be detected by a device immediately after the device itself was powered-on, or immediately after a neighbouring device was powered-off. Only hardware errors occurring during operation should be considered as a actual or potential problem, and investigated.

- No communication interface is totally error-free. Typically communication interfaces ensure a Bit Error Rate of 10^{-12} (one corrupted bit every thousand billion bits transmitted), which would mean a sporadic change of hardware error counters (in a timeframe of days or weeks) even if no critical situation is present. Only burst or often occurring (in a timeframe of seconds or minutes) hardware errors should be considered as a actual or potential problem, and investigated.

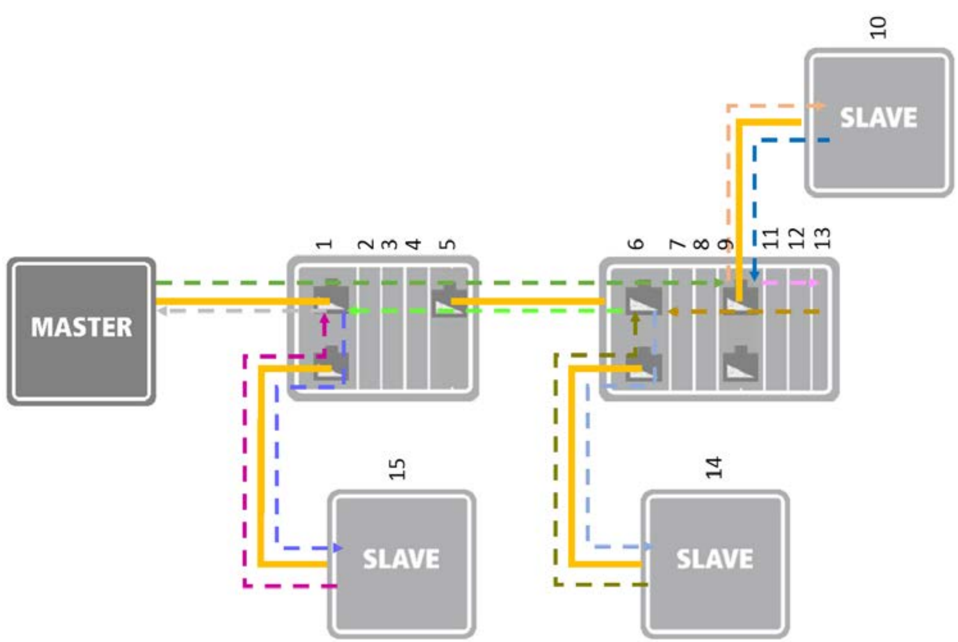
- Errors occurring outside frames, when occurring often and during operation, are also a symptom of hardware problems. Yet, the main attention should be focused on the Frame Errors as these indicate a corruption of the frame content and therefore of the information itself. Frame Error Counters should be interpreted in the following way.

Diagnostic Procedure

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

1. Follow the frame path through the network and determine in which sequence the CRC check is performed (according to [Frame Error detection]) by each port).



CRC checked by	
slave 1	port 0
slave 2	port 0
slave 3	port 0
slave 4	port 0
slave 5	port 0
slave 6	port 0
slave 7	port 0
slave 8	port 0
slave 9	port 0
slave 10	port 0
slave 9	port 3
slave 11	port 0
slave 12	port 0
slave 13	port 0
slave 12	port 1
slave 11	port 1
slave 9	port 1
slave 8	port 1
slave 7	port 1
slave 6	port 1
slave 14	port 0
slave 6	port 2
slave 5	port 1
slave 4	port 1
slave 3	port 1
slave 2	port 1
slave 1	port 1
slave 15	port 0
slave 1	port 2

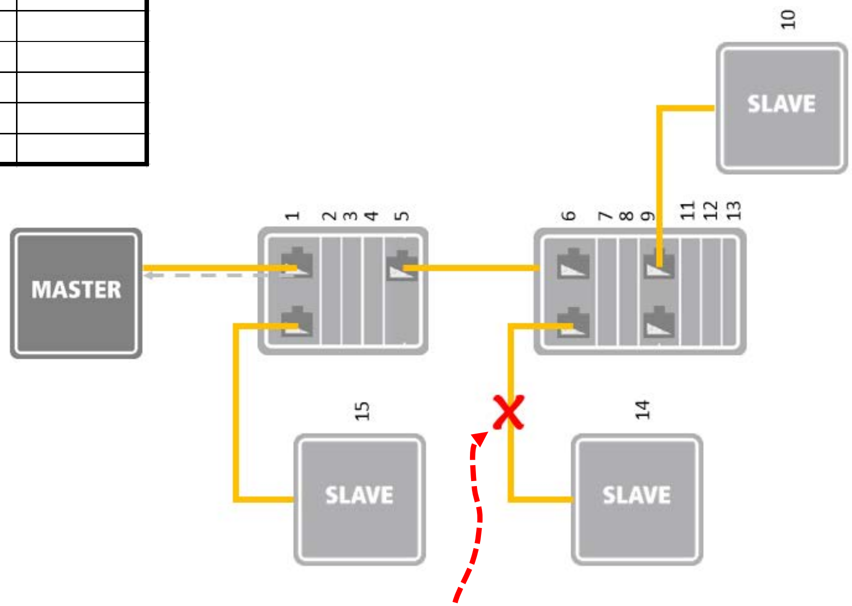
Diagnostic Procedure

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

2. Detect the first port reporting a Frame Error Counter ≠ 0 according to this sequence:

	CRC port 0	CRC port 1	CRC port 2	CRC port 3
slave 1	0x00	0x00	0x00	
slave 2	0x00	0x00		
slave 3	0x00	0x13		
slave 4	0x00	0x00		
slave 5	0x00	0x13	0x13	
slave 6	0x00	0x00		
slave 7	0x00	0x00		
slave 8	0x00	0x00		
slave 9	0x00	0x00		0x00
slave 10	0x00			
slave 11	0x00	0x00		
slave 12	0x00	0x00		
slave 13	0x00			
slave 14	0x0A			
slave 15	0x13			



First port reporting Frame Error Counter ≠ 0 → most likely problem location.

Hardware Diagnostic Procedure

3. Check the following hardware aspects:
   - Check cable between detected and previous slave:
     - EtherCAT cable is routed near to power cables or noise sources
     - Self-made cable connectors have been badly implemented
     - Cable is not properly shielded
   - Check detected and previous device:
     - Not suitable power-supply (for example, low LVDS current)
     - Devices don´t share the same ground potential
   - Try to replace/swap devices at two ends of the detected location, in order to check if errors are related to a specific device part.

As external EMC disturbances are asynchronous with the communication, both Pgysical Layer and Frame Errors should be counted in this case (even if their ratio can vary). Completely unbalanced counter values (many Physical Layer Errors with no Frame Errors, or many Frame Errors with no Physical Layer Errors) could instead indicate an internal device issue: replace the devices could be therefore the first suggested step in this case.

Installation Guideline

A careful planning and implementation of the network infrastructure is the first and most important requisite in order to obtain a stable and error-free transmission.

For this purpose, the **ETG.1600** “EtherCAT Installation Guidelines” is available for download (not only for ETG members!) on the ETG website:

EtherCAT Installation Guideline

Guideline for Planning, Assembling and Commissioning of EtherCAT Networks

Document: ETG.1600 G (R) V1.0.1

Software Diagnostic

EtherCAT State Machine

The operation of every EtherCAT slave device is governed by the EtherCAT state machine.

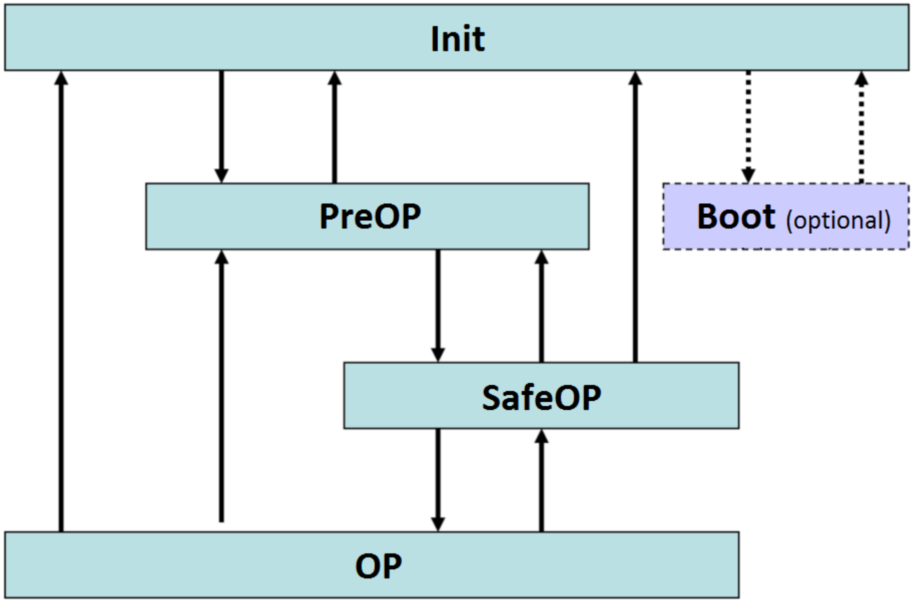
Init: neither acyclic (Mailbox) nor cyclic (Process Data) communication is possible

PreOP: acyclic, but not cyclic data exchange is possible

SafeOP: both acyclic and cyclic data exchange are possible, yet cyclic outputs remain in a predefined state.

OP: both acyclic and cyclic exchange possible without limitations.

Boot: optional state for firmware update, only file transfer over Mailbox enabled.



- Each slave reports its current state, as well as the flag of an error condition in the state machine, in **AL Status** register 0x0130.
- The master requests a new state to a slave by writing **AL Control** register 0x0120 of the slave itself. Spontaneous (backward) transitions can be performed by a slave without master request only in case an error in the state machine occurs.

Run LED

The EtherCAT state machine provides the basic diagnostic information at software level.

Slaves with removable connectors support a Run LED indicator reporting the current state of the slave device in the state machine:

- **Init**: off
- **PreOP**: blinking slowly
- **SafeOP**: single flash with longer pause
- **OP**: on
- **Boot**: flickering fast or off

Each slave leaving OP state during operation without an explicit request from the controller should require a diagnostic investigation.

Error/Status LED and AL Status Code

Slaves with removable connectors can optionally support an Error LED indicator reporting the main State Machine error categories:

- **No error**: off
- **Blinking**: configuration error
- **Single Flash**: generic runtime error
- **Double Flash**: process data watchdog expired
- **…**

Run and Error LEDs can also be combined in a two-coloured Status LED:

Whenever a slave cannot be in the last state requested by the master, an error is reported in AL Status register and a corresponding error code is written in **AL Status Code** register 0x0134. The AL Status Code can be read by the master and reports the diagnostic information provided by the state machine, completing the visual information provided by the Error/Status LED (if one of these LEDs is supported).

AL Status Code

State Machine errors (and corresponding AL Status Codes) can be grouped into the following two categories:

- **Initialization errors** (slave does not reach OP state during start-up): the master requests a state transition, but the slave refuses it because one or more necessary conditions to enter the new state are not satisfied.

  Typical initialization errors:

  - 0x0003 : Invalid Device Setup
  - 0x001D : Invalid Output Configuration
  - 0x001E : Invalid Input Configuration
  - 0x0035 : Invalid Sync Cycle Time

- **Runtime errors** (slave autonomously steps back from OP to a lower state): the slave detects an error during operation and spontaneously performs a backward-transition without master request.

  Typical runtime errors:

  - 0x001A : Synchronization error
  - 0x001B : Sync manager watchdog
  - 0x002C : Fatal SYNC error

AL Status Code – Initialization Errors

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

The information needed by the master to properly configure a slave is derived from the ESI file (typical) or from the slave EEPROM content.

If a slave does not reach the OP state during start-up:

1. Check if slave default settings were changed, and in case delete and append/scan the slave again (default settings will be restored).

2. (In case network configuration is based on ESI) Check if the ESI file containing the slave description is correctly provided to the master configuration tool.

3. (In case of modular slaves) Check if the configured module list corresponds to the physically connected hardware modules.

4. (In case of DC-Synchronous devices) Check if the master jitter could prevent from a proper slave synchronization.

AL Status Code – Runtime Errors

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

Once a slave reached OP state successfully, it should never leave this state without an explicit master request.

If a slave suddenly leaves the OP state:

1. Check if hardware errors (like link loss or frame corruption - see hardware diagnostic features) occur, as such errors could indirectly cause a watchdog reaction or a loss of synchronization.

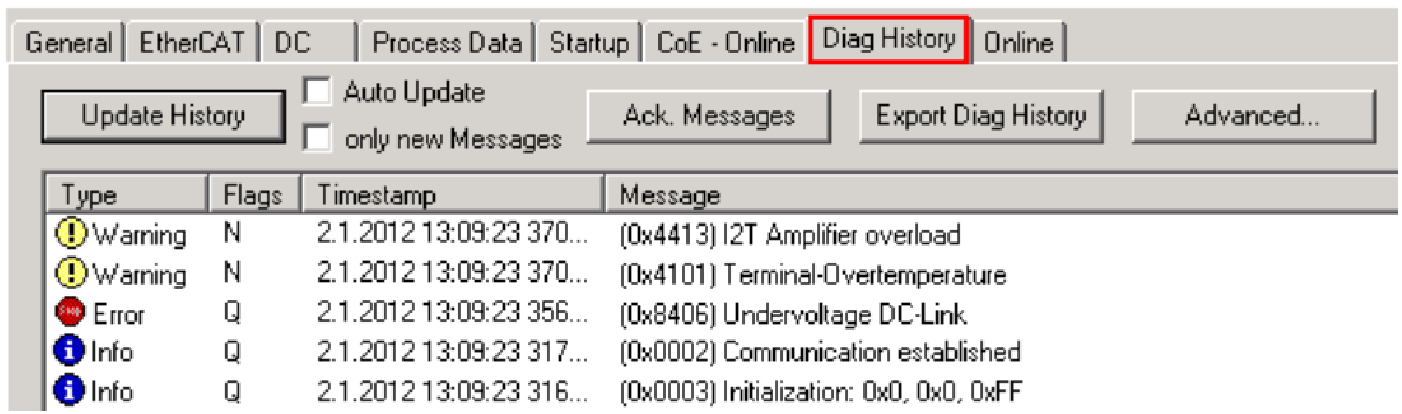
2. (In case of process data watchdog errors) Check if the master application (PLC, NC, …) is running.

3. (In case of synchronization errors) Check if the master jitter performances could justify a synchronization loss (synchronization errors can easily occur if maximum jitter > 20÷30% of the communication cycle time).

Diagnosis History Object

In order to report application-specific errors, slave devices can optionally support CoE Diagnosis History Object **0x10F3,** which can be read by the master via standard SDO services.

Configuration tools can support a graphical interface for the Diagnosis History Object:



Diagnostic Steps on Machine or Plant

Diagnostic Steps on Machine or Plant

Sometimes diagnostic registers are not directly accessible to machine operators, therefore the suggested steps for hardware and software diagnostics cannot be immediately applied: in this case, some preliminary steps can help to locate, and often solve the problem (especially if this is at hardware level).

If these steps do not help to troubleshoot the issue, deeper Hardware and/or Software Diagnostic should be performed with the help of the operating interface (if diagnostic information is available) or of the machine builder.

Whenever communication issues on the EtherCAT network occur:

	Check	Failed when…	If failed…
1	Check Link/Activity LEDs of slave ports connected to the network for each link	LED is stable OFF	Check that devices at both ends of the link are on
			Check that cable connectors are properly inserted
			Check that cable is not mechanically interrupted or damaged along its path
			Check pin-to-pin continuity between end connectors for each wire by means of a tester
			Try to replace the cable.

Diagnostic Steps on Machine or Plant

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

	Check	Failed when…	If failed…
2	Check time elapsed between cable insertion (or device power-on) and Link/Activity LED goes ON (or flickering) for each link	Delay > 6÷7 seconds	Check that devices at both link ends are grounded to the same potential
			Check that connectors have been properly manufactured (only in case of self-assembled cables)
			Check maximum cable length according to cable section (should be ≤ 100 m for AWG 22, cables with smaller sections like AWG 24 or 26 have more strict limitations)
			Check end-to-end cable resistance (should be ≤ 57,5 Ω/km for AWG 22 cables)
3	Check Run LED for each slave device	LED is not stable ON	Check that Link/Activity LED is flickering (confirming that data are received by slave)
			Check blinking code shown by Error/Status LED (if supported)
			Check slave-specific diagnostic information (if supported)
4	In all cases when the available information enables to identify a precise location in the network where communication issues start to appear (only one part of the machine stops working, the operator interface reports errors coming from a precise subset of slaves, …)		Check cables like at points 1 and 2, starting from the network segment(s) affected by the issue.
			Replace cables, starting from the network segment(s) affected by the issue.
			One at a time, replace the devices at two ends of segment(s) affected by the issue.
5	In the case when communication issues affect the whole network		Check cable between master and first slave like at points 1 and 2.
			Restart the master
			Replace the master

EtherCAT Diagnostic

- Diagnostic Features Overview
- Cyclic Synchronous Diagnostic
- Hardware Diagnostic
- Software Diagnostic
- Diagnostic Procedure Example

Please visit

www.ethercat.org

for more information

EtherCAT Technology Group
ETG Headquarters
Ostendstr. 196
90482 Nuremberg, Germany
Phone: +49 911 54056 20
info@ethercat.org