Measuring amplifier GSV-8

EtherCat Protocol

Last edit: 05/02/2021

ME-Meßsysteme GmbH
Eduard-Maurer-Str. 9
16761 Hennigsdorf

Tel.: +49 3302 89824 60
Fax: +49 3302 89824 69

Mail: info@me-systeme.de
Web: www.me-systeme.de

Table of contents

General

Besides the default USB interface the GSV-8-EtherCAT offers an additional EtherCAT interface. The application layer is implemented according to the EtherCAT standard, using CoE (CANopen over EtherCAT). The EtherCAT protocol is defined in IEC61158.

The EtherCAT protocol uses standard Ethernet frames with Ether-type 0x88A4. An EtherCAT telegram may consist of several sub telegrams, the so called EtherCAT commands. Most of the device configurations are done via the EtherCAT commands.

EtherCAT Overview
An EtherCAT network is built of a master and one or more slaves. While the master is usually built of a powerful PC the slaves are optimized for the fast transfer of the process data from and to the several control units. The algorithms to process the data transfer are implemented on the PC.
The main features of EtherCAT are:

- A broad applicability. As an EtherCAT master every commercially available computer with a normal Ethernet controller can be used.
- Highest efficiency. The Ethernet bandwidth is nearly fully usable.
- Short cycle times. The EtherCAT cycle times are typically in the range of 10..50 µs.
- Variety of bus topologies. EtherCAT supports a huge variety of bus topologies, e. g. line, ring, star, etc., thus supporting redundancy; hot connect of segments and device exchange in an active network.

Connection of the EtherCAT Wires

EtherCAT two network connectors. The connection Port0 is used for the connection towards the EtherCAT-Master, Port1 to connect more Slaves or back to the master for redundancy purpose. The GSV-8 uses M12 connectors (IP code 67) according to IEC 61918, annex H.

Signal	Function	Pin-No. M12 connector	Pin-No. RJ45
TD+	Transmit data +	**1**	1
TD-	Transmit data -	**3**	2
RD+	Receive data +	**2**	3
RD-	Receive data -	**4**	6

EtherCAT System Architecture

On the perspective of the “normal” Ethernet topology the EtherCAT bus shows up as a

single Ethernet participant. Within this “participant” however there is no Ethernet controller with an application processor but rather several EtherCAT slaves.
The EtherCAT master uses the network configuration which is stored in the EtherCAT Network Information file (ENI). The ENI is created by the EtherCAT Configuration Tool based on the EtherCAT Slave Information (ESI), which is provided for every device by the vendor (for GSV-8: GSV8EtherCat<VerNo>rev<RevNo>.xml, available on www-me-systeme.de). The slaves are connected via standard Ethernet cables.
The EtherCAT master system just requires a standard Network Interface Controller (NIC, 100 Mbit/s Full duplex) and a real time run-time environment that drives the slaves in the network.

The slaves process the data “on the fly” while receiving them and putting appropriate process data into the stream in the same instance. I. e. the data stream is not copied first than processed and finally an answer is sent back. In fact the whole processing takes place in one step while the stream is running through a slave device, thus the whole frame is delayed by just a few bits.

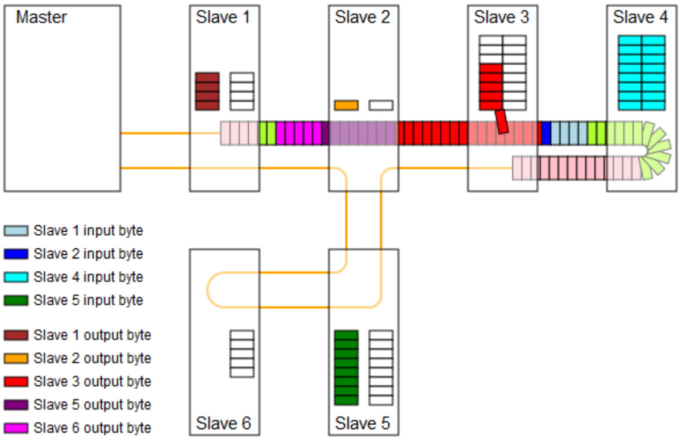


Figure : EtherCAT communication principle

Each slave device holds an addressable 64kByte RAM area that can be read or written to or even read and written in the same cycle. There may be multiple EtherCAT commands within one Ethernet frame to address or read/write individual slave devices.

EtherCAT Protocol

The EtherCat commands were transferred in the datagram area of an Ethernet frame as shown in the table . Since the EtherType is set to 0x88A4 to identify an EtherCAT frame they will not pass any router. The GSV-8 only supports the Direct Mode Addressing, i.e. the IP address and MAC of an Ethernet frame is ignored.
Each EtherCAT datagram consists of the datagram header, the data and the so called “working counter” (WKC). The working counter is incremented by every device that is adressed by an EtherCAT command.

EtherCAT protocol frame

Field	Value / Description
Length	Length of the EtherCAT datagrams (excl. FCS)
Res	Reserved, 0
Type	Protocol Type. Only EtherCAT commands (Type = 0x01) are supported by ESCs
Cmd	EtherCAT Command Type
Idx	The index is a numeric identifier used by the master for identification of duplicates / lost datagrams that shall not be changed by the slaves.

Table : EtherCAT field descriptions

EtherCAT Slave Architecture

The main components of the EtherCAT slaves are:

- Physical Layer: Network interface
- Data Link Layer: EtherCAT Slave Controller (ESC, communication module) and EEPROM
- Application Layer: Application controller or microcontroller

The ESC is a hardware module for EtherCAT communication. The ESC handles the EtherCAT protocol in real-time by processing the EtherCAT frames on the fly and providing the interface for data exchange between the EtherCAT master and the slave's local application controller via registers and a DPRAM (dual-port RAM).
The ESC processes EtherCAT frames on the fly and exchanges data with the local controller of the GSV-8, which processes the measuring data.

EEPROM EtherCAT Slave Configuration

Since the DPRAM in the ESC is a volatile RAM, it also has an EEPROM (NVRAM, also called Slave Information Interface, SII). The EEPROM stores slave identity information and information about the slave's functionality corresponding to the ESI file. The content of the EEPROM is configured by the manufacturer with necessary (default) settings. EEPROM information can be derived from the ESI file. In case of updating the device firmware, the EEPROM content of the ESI should be renewed, too, because it contains the firmware revision number.

States of an EtherCAT slave

The slave runs a state machine to indicate which functionalities are actually available.
ESM requests are written by the master to the slave's AL Control register in the ESC. If the

configuration for the requested state is valid, the slave acknowledges the state by setting the AL Status register. If not, the slave sets the error flag in the AL Status register and writes an error code to the AL Status Code register.

EtherCAT Slave State Machine

State	Available Functions
INIT	Init state. No communication on the application layer is available. The master has access only to the DL-information registers.
PREOP	Pre-Operational state. Mailbox communication on the application layer available, but no process data communication (PDO) available.
SAFEOP	Safe-Operational state. Mailbox communication on the application layer, process data communication available. In SafeOp only inputs like measuring values are evaluated; outputs are kept in 'safe' state.
OP	Operational state. Process data inputs and outputs are valid.
BOOT	Bootstrap state. In this state some devices use the FoE protocol for firmware download. Not supported by GSV-8, which uses the USB port for firmware download.

Table : EtherCAT slave states

The initialization information of every EtherCAT state transition is derived from the EtherCAT Slave Information file (ESI, available from ME-Messsysteme) by a network configurator and stored in the network information file (ENI). Each slave gets its required initialization commands for each state transition. The EtherCAT master initializes the slave(s) using this ENI, e.g. logical slave I/O mapping is done according to the network topology.

EtherCAT Commands

All supported EtherCAT low-level Command types are listed in table below. For Read-write operations, the Read operation is performed before the Write operation.

CMD	Abbr.	Name	Description
0	NOP	No operation	Slave ignores command
1	APRD	Auto Increment Read	Slave increments address. Slave puts read data into the EtherCAT datagram if received address is zero.
2	APWR	Auto Increment Write	Slave increments address. Slave writes data into memory location if received address is zero.
3	APRW	Auto Increment R/W	Slave increments address. Slave puts read data into the EtherCAT datagram and writes the data into the same memory location if received address is zero.

CMD	Abbr.	Name	Description
4	FPRD	Configured Address Read	Slave puts read data into the EtherCAT datagram if address matches with one of its configured addresses.
5	FPWR	Configured Address Write	Slave writes data into memory location if address matches with one of its configured addresses
6	FPRW	Configured Address R/W	Slave puts read data into the EtherCAT datagram and writes data into the same memory location if address matches with one of its configured addresses.
7	BRD	Broadcast Read	All slaves put logical OR of data of the memory area and data of the EtherCAT datagram into the EtherCAT datagram. All slaves increment position field.
8	BRW	Broadcast Write	All slaves write data into memory location. All slaves increment position field.
9	BRW	Broadcast R/W	All slaves put logical OR of data of the memory area and data of the EtherCAT datagram into the EtherCAT datagram, and write data into memory location. BRW is typically not used. All slaves increment position field.
10	LRD	Logical Memory Read	Slave puts read data into the EtherCAT datagram if received address matches with one of the configured FMMU areas for reading.
11	LWR	Logical Memory Write	Slaves writes data to into memory location if received address matches with one of the configured FMMU areas for writing.
12	LRW	Logical Memory R/W	Slave puts read data into the EtherCAT datagram if received address matches with one of the configured FMMU areas for reading. Slaves writes data to into memory location if received address matches with one of the configured FMMU areas for writing.
13	ARMW	Auto Increment Read Multiple Write	Slave increments address. Slave puts read data into the EtherCAT datagram if received address is zero, otherwise slave writes the data into memory location.

Table : EtherCAT Commands

GSV-8 EtherCat Implementation

The GSV-8 measuring amplifier has eight analogue inputs, which are digitized with a 24 Bit Sigma-Delta analog-to-digital converter, who converts all channels simultaneously. In

conjunction with the buffered-mode Sync-Manager of the EtherCat interface, the data of the eight channels representing the excitation of the plugged sensors belong exactly to the same point of time. The rate, with which the measuring controller updates the values, is configurable from 1 to 12000 values/second.
The GSV-8 implements the parts of the CoE CanOpen 404 device profile, which is used for measuring devices.
Most configuration parameters can be accessed through objects communicated through the Sync-Manager's mailbox mode. The objects are identified by indices, which in itself are sub-divided into sub-indices. Sub-indices 1 to 8 often refer to the corresponding analogue input channel configuration. From device software version 1.45, many objects have the additional subindices 9 and 10, which refer to the counter or frequency measuring digital input(s).

The EtherCAT communication interface is described by a standartized device description file (available from the vendor), the ESI file. It has a specialized xml format. Many EtherCAT master programs can read it and then they have information about the communication properties of the device, e.g. the SDO object dictionary and the PDO properties.

The EtherCat Slave interface is realized using a Slave controller by Beckhoff Automation GmbH, Germany.

GSV-8 Application data communication

EtherCAT devices communicate data through RAM memory, which is accessed directly by the EtherCAT interface. This memory is divided into 4 parts (managed by a "FMMU" = field memory management unit), which are assigned to particular services:

Name	SM Channel	Service	Description
Sync Manager 1 (SM1)	0	SDO out	Service data request by the master. Parameters are communicated through SDOs.
Sync Manager 2 (SM2)	1	SDO in	Service data response from the device. Parameters are communicated through SDOs.
Sync Manager 3 (SM3)	2	PDO out	Process data from master to device. Not used by GSV-8.
Sync Manager 4 (SM4)	3	PDO in	Process data from device to master. Used by GSV-8 for measuring data and sensor status. Mappable (see SDO 1A00 description).

Service-Data Objects (SDO)

Many device parameters can be accessed through SDOs. Those who are not accessible can be changed by the serial USB interface and an appropriate software (GSVmulti / BlueDAQ). SDO communication is always initiated by the master and always responded by the device. The response may or may not contain data. It has an error code that shows whether the request was successful or not. If not, the response contains an error code as shown below.

SDOs are distinguished by their Index. Many SDOs are further divided into sub-indices, which access data for particular variables, often related to input channels of the device. The particular SDOs are described below in the Object dictionary. With EtherCAT, one can access all subindices at once, using the "complete access" method.

Error Messages

These error codes can be returned by the Mailbox/SDO when accessing the Object Dictionary and an error occurs.

Error Code (hex)	Meaning
0503 0000	Toggle bit not changed
0504 0000	SDO protocol timeout (*reserved*)
0504 0001	Command Byte invalid or unknown
0601 0000	Unsupported access to an object
0601 0001	Attempt to read a write only object
0601 0002	Attempt to write a read only object
0601 0003	Entry can not be written because Subindex0 is not 0
0602 0000	Object does not exist in the object dictionary
0604 0041	Object can not be mapped to PDO
0604 0042	Mapped Object exceeds PDO
0604 0043	General parameter incompatibility reason (*reserved*)
0604 0047	Device incompatibility (*reserved*)
0606 0000	Access failed due to an hardware error (*reserved*)
0607 0010	Parameter length error
0607 0012	Length of service parameter too high (*reserved*)
0607 0013	Length of service parameter too low (*reserved*)
0609 0011	Sub-index does not exist
0609 0030	Invalid value for parameter (download only)
0609 0031	Value of parameter too high (download only)
0609 0032	Value of parameter too low (download only)

Error Code (hex)	Meaning
0800 0000	General error
0800 0020	Data cannot be transferred or stored to the application
0800 0022	Data cannot be transferred or stored to the application in the present device state.
0800 0023	Object is not in the object dictionary

Table : Mailbox/SDO Error Codes

Distributed clocks

The synchronization mode used is the telegram synchronous mode, synchronous with Sync Manager 3, which is used for the process data inputs, containing the mapped measurement values (see Obj.Descr. 1A00h and 1C33h).

The available hardware synchronization mode is the Latch 1 method.

After a measurement data frame acquisition is completed, 1.45 µs later the slave controller captures the system time of the positive edge of its Latch 1 input (ET1100 registers 0x09C0..0x09C7).

When the calculation of the physical PDO values is completed, the system time of the negative edge of the Latch 1 input is captured by the ET1100 (registers 0x09C8..0x09CF).

Object Dictionary

This chapter specifies the objects of the GSV-8-EtherCAT implementation. Mnay of them follow the CANopen device profile 404, since the GSV-8 EtherCAT is a CoE device.

Index (Hex)	Name	Type
1000	Device Type	Communication
1001	Error Register	Communication
1008	Device Name	Communication
1009	Hardware Version	Communication
100A	Software Version	Communication
1010	Store Parameters	Communication
1011	Restore Parameters	Communication
1018	Identity Object	Communication
10F0	Backup Parameter Handling	Communication
10F1	Error Settings	Communication
10F8	Timestamp Object	Communication[1]
1A00	Tx PDO 1 Mapping Parameter	Communication
1C00	Sync Manager Type	Communication
1C12	RxPDO Assign	Communication
1C13	TxPDO Assign	Communication
1C33	SyncManager Input Parameter	Communication
6000	DI Read 8 Lines	Application, Digital Input[2]
6112	AI Operating Mode	Application, Analog Input
6114	AI ADC sample rate	Application, Analog Input
611C	AI TEDS control	Application, Analog Input
6125	AI Autozero	Application, Analog Input
6126	AI Scaling Factor	Application, Analog Input
6127	AI Scaling Offset	Application, Analog Input
6130	AI Process Value Float	Application, Analog Input
6131	AI Physical Unit	Application, Analog Input
6148	AI Span Start Float	Application, Analog Input
6149	AI Span End Float	Application, Analog Input

[1] This object is present from firmware version 1.43 on

[2] This object is present from firmware version 1.45 on

Index (Hex)	Name	Type
6150	AI Status	Application, Analog Input
6160	AI Control byte	Application, Analog Input
61A0	AI Filter Type	Application, Analog Input
61A1	AI Filter constant	Application, Analog Input
9100	AI Field Value	Application, Analog Input

Table : Object Dictionary

Index 1000h: Device Type

This object describes the device type and which profile the device is conforming to.

Sub-Index	Data Type	Access	Description	Default value
0	Unsigned32	ro	Device Type/Profile	0x800B0194

Table : Index 1000h

This object is read-only and only implements the sub-index 0. Any other access will result in an error.
The value 0x00220194 contains the following two fields:

Device Profile Number	0194h = 404d
Additional Information	0Bh = 00001011b

The device is claiming conformity to the CiA404 Measuring Device Profile and that the following function blocks are present:

- Bit 16: =1: Digital input present
- Bit 17: Analogue input block (always present in GSV-8)
- Bit 19: Analogue output block
- Bit 31: Device-specific PDO-Mapping (always set)

Index 1001h: Error Register

This object describes the device Error state.

Sub-Index	Data Type	Access	Description	Default value
0	Unsigned8	ro	Error flags	0x00

Table : Index 1001h

This object is read-only and only implements the sub-index 0. Any other access will result in an error.
Of these flags the following are implemented:

- Bit 0: Generic Error. This flag will be 1 on any error and 0 if there is none.
- Bit 2: Voltage Error. If =1, it means that there's a problem with an analogue bridge sensor at the input, e.g. excitation voltage fault.

Index 1008h: Device Name

This object contains the device name.

Sub-Index	Data Type	Access	Description	Default Value
0	String(5)	ro	Device Name	"GSV-8"

Table : Index 1008h

This object is read-only and only implements the sub-index 0. Any other access will result in an error.

Index 1009h: Hardware Version

This object contains the hardware version string.

Sub-Index	Data Type	Access	Description	Default Value
0	String(2)	ro	Hardware version	"04"

Table : Index 1009h

This object is read-only and only implements the sub-index 0. Any other access will result in an error.

Index 100Ah: Software Version

This object contains the software version string.

Sub-Index	Data Type	Access	Description	Default Value
0	String(5)	ro	Software version	"01.47" (example)

Table : Index 100Ah

This object is read-only and only implements the sub-index 0. Any other access will result in an error.

Index 1010h: Store Parameters

By writing to this object, device parameters can be stored to non-volatile memory. By reading, information on storage behaviour can be retrieved.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	10[3]
1	Unsigned32	rw	Store EtherCAT application Objects	2

[3] From device software version 1.47. Before, only Subindex 0 and 1 was supported (Value was =1).

Sub-Index	Data Type	Access	Description	Default Value
2	Unsigned32	rw	Store EtherCAT Communication Objects (PDO Mapping 1A00) only	1[4]
3	Unsigned32	rw	Store EtherCAT application objects (6148h, 6149h) only	1 [4]
4	Unsigned32	rw	Auto-Store Last power-on values	2
5 to 10d	Unsigned32	rw	Parameter sets for manual store/restore	1

By reading the object, it gives Information on parameter storage capabilities:

Value	Meaning
0	At this Parameter set =Subindex no values are stored (not used)
1	At this Parameter set =Subindex, parameters are stored by command only
2	At this Parameter set =Subindex, the device stores parameters autonomously

By writing the Signature "save", at some subindices, parameters can be stored as follows:

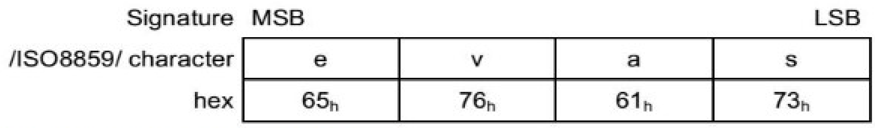
Subindex 1: EtherCAT application objects can be stored. For Firmware version until 1.47, this applies to the Objects 0x6148 and 0x6149 only. This Subindex is also present in firmware versions prior to 1.47, with the same functionality. When saving, a checksum value is written to Object 0x10F0. From firmware version 1.48, the PDO mapping as defined by SDO 1A00h will be stored, too, along with the Objects 0x6148 and 0x6149.

Subindex 2: EtherCAT Communication Objects can be stored. This is the PDO mapping defined by SDO 1A00h. After storing, the mapping will automatically be restored on the next power-on cycle (if valid).

Subindex 3: EtherCAT application objects can be stored. These are the SDOs 6148h and 6149h, since all other application objects are stored automatically. After storing, they will automatically be restored on the next power-on cycle.

Subindex 5 to 10: At these subindices, user defined parameters of the measuring application can be stored. All relevant data of the measuring application are stored, even some that cannot be accessed by EtherCAT (but by USB). These parameter sets correspond to the device parameter sets "User 1" (Subindex 5) to "User 6" (Subindex 10) available at the serial interface.

[4] From device software version 1.48.



Signature	MSB			LSB
/ISO8859/ character	e	v	a	s
hex	65_h	76_h	61_h	73_h

Storage write access signature

Caution: The GSV-8 stores measuring application parameters by itself. It's not necessary and not recommended to store parameters each time after a change. The EEPROM used for storing is specified for 10 Mio. write cycles.
Instead, this object is useful to prepare for switching between several different parameter sets, e.g. for different measuring tasks.

Index 1011h: Restore Parameters

By writing to this object, device parameters can be loaded from non-volatile memory. By reading, information on loading behaviour can be retrieved.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	10[5]
1	Unsigned32	rw	Restore all parameters	1
2	Unsigned32	rw	Restore Communication related parameters[6]	1
3	Unsigned32	rw	Restore manufacturer default parameters	1
4	Unsigned32	rw	Restore Last power-on values	1
5 to 10d	Unsigned32	rw	Restore previously saved parameter sets	1

By reading the object, it gives Information on parameter loading capabilities:

Value	Meaning
0	From this Parameter set =Subindex nothing can be loaded (not used)
1	From this Parameter set =Subindex, parameters can be loaded

By writing the Signature "load", at most subindices, parameters can be restored as follows:

Subindex 1: All storable parameters are restored, EtherCAT application objects (0x6148 and 0x6149) as well as other measuring application parameters. For the measuring parameters, manufacturer default values are loaded. This Subindex is also present in firmware versions

[5] From device software version 1.47. Before, only Subindex 0 and 1 was supported (Value was =1).

[6] From device software version 1.50

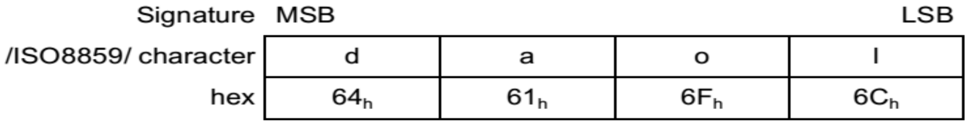
prior to 1.47, but with these, only EtherCAT application objects were restored.

Subindex 2: Restore communication related parameters, i.e. the PDO mapping from SDO 1A00h.

Subindex 3: Restore manufacturer default parameters.

Subindex 4: Restore last power-on values.

Subindex 5 to 10: At these subindices, user defined parameters of the measuring application can be restored, see Obj. 0x1010. These parameter sets correspond to the device parameter sets "User 1" (Subindex 5) to "User 6" (Subindex 10) available at the serial interface.



Signature	MSB			LSB
/ISO8859/ character	d	a	o	l
hex	64_h	61_h	$6F_h$	$6C_h$

Restore write access signature

Index 1018h: Identity Object

This object contains the device identity.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x04
1	Unsigned32	ro	Vendor ID	0x00000270
2	Unsigned32	ro	Product code	0x00080000
3	Unsigned32	ro	Revision	0x0001002F (example)
4	Unsigned32	ro	Serial number	-

Table : Index 1018h

This object is read-only and only implements the sub-index 0 to 4. Any other access will result in an error.

Sub-Index 1: Vendor ID
The vendor ID is a unique manufacturer identification number assigned by the EtherCAT association. ME Meßsysteme GmbH has the vendor ID = **0270h**

Sub-Index 2: Product Code
The product code is a unique identification number of the product assigned by the vendor. **00080000h** is the code for GSV-8

Sub-Index 3: Revision
The revision is the binary equivalent to the software version an in Obj. **100Ah**.

Sub-Index 4: Serial Number

The serial number in its decimal representation, prepended with zeros if the representation has less than 8 digits, can be found on the device's specification plate.

Index 10F0h: Backup Parameter Handling

The Object contains a Checksum value at subindex 1. It was written by the device after it had saved or restored savable EtherCAT application objects.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x01
1	Unsigned32	rw	Checksum Value	-

Index 10F1h: Error Settings

This object contains the EtherCAT error setting.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x02
1	Unsigned32	rw	Local Error Reaction	0x00000000
2	Unsigned32	rw	Sync Error Counter Limit	0x00000000

Table : Index 10F1h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 8. Any other access will result in an error.

Sub-Index 1: Local Error Reaction

The Local Error Reaction defines how the slave shall behave if a local error occurs.

(1) PDO state
(2) Disable SyncManager
(3) Device specific state

The default setting is to report the error via PDO state.

Sub-Index 2: Sync Error Counter Limit

The Sync Error Counter is incremented with every missing Sync Management Event and decremented if an event is received. If the Sync Error Counter exceeds this limit the system changes into the SAFEOP state with the 'Synchronization Lost' error. The Sync Error Counter is reset when the error was acknowledged.

Index 10F8h: Timestamp Object

The Object contains a running time value. The value is internally incremented and not correlated to measuring events.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned64	rw	Local time in ns	-

Index 1A00h: TxPDO Map 1

This object contains the transmit PDO mapping; the description which object value has to be transmitted on request from the EtherCAT master.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	rw	Highest sub-index= Number of mapped objects in the Out-PDO	0x10
1	Unsigned32	rw	Mapped Object 1 = 1st Object in the PDO	0x61300120[1] =AI Process Value of channel 1
2	Unsigned32	rw	Mapped Object 2 = 2nd Object in the PDO	0x61300220 [1] =AI Process Value of channel 2
3	Unsigned32	rw	Mapped Object 3 = 4rd Object in the PDO	0x61300320 [1] =AI Process Value of channel 3
4	Unsigned32	rw	Mapped Object 4 = 4th Object in the PDO	0x61300420 [1] =AI Process Value of channel 4
5	Unsigned32	rw	Mapped Object 5 = 5th Object in the PDO	0x61300520 [1] =AI Process Value of channel 5
6	Unsigned32	rw	Mapped Object 6 = 6th Object in the PDO	0x61300620 [1] =AI Process Value of channel 6
7	Unsigned32	rw	Mapped Object 7= 7th Object in the PDO	0x61300720 [1] =AI Process Value of channel 7
8	Unsigned32	rw	Mapped Object 8 = 8th Object in the PDO	0x61300820 [1] =AI Process Value of channel 8
9	Unsigned32	rw	Mapped Object 9 = 9th Object in the PDO	0x61500120 =AI Status of channel 1
10	Unsigned32	rw	Mapped Object 10 = 10th Object in the PDO	0x61500220=AI Status of channel 2

[1] Def. value applies to device software version **1.26 and higher**. Older versions map Obj. 0x9100 (AI Process value Int32) by default at 1A00.1 =91000120 to 1A00.8 =91000820.

Sub-Index	Data Type	Access	Description	Default Value
11	Unsigned32	rw	Mapped Object 11 = 11th Object in the PDO	0x61500320=AI Status of channel 3
12	Unsigned32	rw	Mapped Object 12 = 12th Object in the PDO	0x61500420=AI Status of channel 4
13	Unsigned32	rw	Mapped Object 13 = 13th Object in the PDO	0x61500520=AI Status of channel 5
14	Unsigned32	rw	Mapped Object 14 = 14th Object in the PDO	0x61500620=AI Status of channel 6
15	Unsigned32	rw	Mapped Object 15 = 15th Object in the PDO	0x61500720=AI Status of channel 7
16	Unsigned32	rw	Mapped Object 16 = 16th Object in the PDO	0x61500820=AI Status of channel 8

Table : Index 1A00h

This object implements the sub-index 0 to 24[7] for read/write-access. A write to sub-index 0 with a value greater than 24 will result in an error. Any access on another sub-index will also result in an error.

Each entry has the following form:

Bits 31:16	Bits 15:8	Bits 7:0
Index of mapped object	Its Sub-Index	Its size in Bits

Table : Sub-Index Bit Mapping for Index 1A00h

Dynamic PDO-Mapping

To change the TxPDO mapping one has to follow the following Steps:

1. The device must be placed into the PREOP-state.
2. The PDO mapping has to be invalidated by writing a Null into the sub-index 0 of Obj. 1A00.
3. Change the PDO mappings by writing the desired object index, sub-index and Bit-size value into the sub-indices 1 up to the last object to be mapped (maximum is 24[5]). Do not leave empty entries in between.
4. Make the PDO mapping valid by writing the number of the highest sub-index, i.e. the number of mapped objects, into sub-index 0 of 1A00.
5. Switch the device back into SafeOP- or OP-state.

Caution: - The EtherCAT implementation only allows entries of the objects **6130h** (AI Process value), **6150h** (AI Status) **9100h** (AI Field value) and **6000h**[8] (DI read state) in the TxPDO mapping object entries. See these Object description on info about their meaning and their sub-indices.

[7] From device software version 1.45 on. Before: Maximum sub-index =16.

[8] Available from firmware Version 1.45 on

- The altered mapping is **not** stored automatically to non-volatile memory, i.e. after rebooting, the default mapping is set again. From device firmware version 1.48 on, the mapping can be stored by writing to SDO 1010.1h or 1010.2h (see there), so that it will be restored on the next power-on cycle.

Index 1C00h: Sync Manager Type

This object contains the Sync Manager types implemented in the device.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x04
1	Unsigned8	ro	Sync Manager Type channel 0	0x01
2	Unsigned8	ro	Sync Manager Type channel 1	0x02
3	Unsigned8	ro	Sync Manager Type channel 2	0x03
4	Unsigned8	ro	Sync Manager Type channel 3	0x04

Table : Index 1C00h

This object is read-only and only implements the sub-index 0 to 4. Any other access will result in an error.

Sync Manager Channel	Value	Description
0	1	Mailbox Write (SDO Request)
1	2	Mailbox Read (SDO Response)
2	3	Process Output Data (RxPDO, unused)
3	4	Process Input Data (TxPDO)

Table : Sync Manager Channels of Index 1C00h

Index 1C12h: RxPDO assign (Sync Manager)

This object contains the Sync Manager RxPDO assignment.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x00
1	Unsigned16	ro	Active RxPDO Map 1	-

Table : Index 1C12h

This object is read-only and only implements the sub-index 0 to 1. Any other access will result in an error.
Since Digital output and Alarm function block (see object 1000h) are not present, this

object has no active RxPDO (sub-index 0 is 0x00), therefore the value of sub-index 1 is left unspecified.

Index 1C13h: TxPDO Assign (Sync Manager)

This object contains the Sync Manager TxPDO assignment.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x01
1	Unsigned16	ro	Active TxPDO Map 1	0x1A00

Table : Index 1C13h

This object is read-only and only implements the sub-index 0 to 1. Any other access will result in an error.
The active TxPDO map is used by the Sync Manger to point to the active TxPDO. The TxPDO Map 1 (**1A00h**) is always used.

Index 1C33h: Sync Manager 3 Input Parameter

This object contains the Sync Manager parameter for ‘inputs’ (TxPDO) assigned in the TxPDO assignment (**1C13h**) object.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x20
1	Unsigned16	rw	Synchronization type	0x0000
2	Unsigned32	ro	Cycle time	0x00000000
3	-	ro	Unused	-
4	Unsigned16	ro	Synchronization types supported	0x4003
5	Unsigned32	ro	Minimum cycle time	0x00005161
6	Unsigned32	ro	Calc and copy time	0x00001644
7	-	ro	Unused	-
8	Unsigned16	rw	Get cycle time	0x0000
9	Unsigned32	ro	Delay time	0x00000000
10	Unsigned32	rw	Sync0 cycle time	0x00000000
11	Unsigned16	ro	Sync Manager event missed	0x00000000
12	Unsigned16	ro	Cycle time too small	0x00000000
13..31	-	ro	Unused	-
32	Bool	ro	Sync error	0

Table : Index 1C33h

This object is read-only and only implements the sub-index 0 to 32. Any other access will result in an error.
All time entries are specified in nanoseconds.

Sub-Index 1: Synchronization Type

If the synchronization type is set to 0x0001, the Sync Mode 'Synchronous with SyncManager 3'is enabled. Otherwise, it remains at the default value 0x0000: free-run mode.

Sub-Index 2: Cycle Time

This defines the minimum time between two SyncManager events.

Sub-Index 4: Synchronization Types supported

The bits indicate which synchronization type the implementation supports. It is for information only.

- Bit 0: Free Run supported
- Bit 1: Synchronous Mode supported
- Bit 14: Dynamic cycle times supported

Sub-Index 5: Minimum Cycle Time

This defines the minimum time between two SyncManager events which the application supports.

Sub-Index 6: Calc and Copy Time

Time needed by the application to perform the necessary calculations and copy the process data from the local memory to the SyncManager.

Sub-Index 8: Get Cycle Time

This entry controls the measurement of the local cycle time.

- Bit 0 controls if the measurement is active (1) or inactive (0).
- Bit 1 controls a reset of all measurement times.

Sub-Index 9: Delay Time

Definition of the delay time of the hardware before the latched value becomes active.
In the present mode of operation this value is unused.

Sub-Index 10: Sync0 Cycle Time

Communication cycle time in nanoseconds, between two Sync0 events.

In the present mode of operation this value is unused.

Sub-Index 11: SyncManager Event missed

The error counter used for missing Sync-Events (see 10F1h).

Sub-Index 12: Cycle Time Too Small

This error counter is incremented, if the time between two Sync-Events is too small, so that a local cycle cannot be completed and input data cannot be provided before the next Sync-Event.

Sub-Index 32: Sync Error

Flag to indicate a SyncManger-Event missed has occurred.

Index 6000h: Digital Input Read 8 Lines

With this object, 8 digital line states can be read.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x02
1	Unsigned8	ro	Read Lines 1 to 8	-
2	Unsigned8	ro	Read Lines 9 to 16	-

Table : Index 6000h

This object is read-only and only implements the sub-index 0 to 1. Any other access will result in an error.

From device software 1.45 or higher, the digital line level can be read with this object. The values in bits 0 to 7 represent the state of the digital lines 1 to 8 at sub-index 1 and of lines 9 to 16 at sub-index 2. The reflect the state of the digital input connection, if they are configured as general-purpose inputs (as by default). The digital I/O functions and types are versatile and can be configured via USB or serial port only.

Index 6112h: AI Operating mode

This object represents the device’s mode of operation.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned8	ro	AI Operating mode channel 1	0x01
2	Unsigned8	ro	AI Operating mode channel 2	0x01
3	Unsigned8	ro	AI Operating mode channel 3	0x01
4	Unsigned8	ro	AI Operating mode channel 4	0x01
5	Unsigned8	ro	AI Operating mode channel 5	0x01
6	Unsigned8	ro	AI Operating mode channel 6	0x01
7	Unsigned8	ro	AI Operating mode channel 7	0x01
8	Unsigned8	ro	AI Operating mode channel 8	0x01
9	Unsigned8	ro	AI Operating mode channel 9	0x00
10	Unsigned8	ro	AI Operating mode channel 10	0x00

Table : Index 6112h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).
The operating mode has two states: Normal operation of channel specified by Sub-index (=1) or channel off (=0).

Index 6114h: AI ADC Sample Rate

This object represents the current sample period of one ADC conversion.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned32	ro	AI ADC sample rate channel 1	0x00000015
2	Unsigned32	ro	AI ADC sample rate channel 2	0x00000015
3	Unsigned32	ro	AI ADC sample rate channel 3	0x00000015
4	Unsigned32	ro	AI ADC sample rate channel 4	0x00000015
5	Unsigned32	ro	AI ADC sample rate channel 5	0x00000015
6	Unsigned32	ro	AI ADC sample rate channel 6	0x00000015
7	Unsigned32	ro	AI ADC sample rate channel 7	0x00000015
8	Unsigned32	ro	AI ADC sample rate channel 8	0x00000015
9	Unsigned32	ro	AI ADC sample rate channel 9	0x00000015
10	Unsigned32	ro	AI ADC sample rate channel 10	0x00000015

Table : Index 6114h

This object is read-only on sub-index 0 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).
Caution: The sample rate of the ADC is set to a constant value of 48000Hz, which corresponds to a time period of 20.83 µs between two samples. The nearest integer value (21d = 15h) is used in this object.

Index 611Ch: AI TEDS control

This object controls/indicates the usage of the TEDS data of sensors.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x08
1	Unsigned8	rw	AI TEDS control channel 1	0x02
2	Unsigned8	rw	AI TEDS control channel 2	0x02
3	Unsigned8	rw	AI TEDS control channel 3	0x02
4	Unsigned8	rw	AI TEDS control channel 4	0x02
5	Unsigned8	rw	AI TEDS control channel 5	0x02

Sub-Index	Data Type	Access	Description	Default Value
6	Unsigned8	rw	AI TEDS control channel 6	0x02
7	Unsigned8	rw	AI TEDS control channel 7	0x02
8	Unsigned8	rw	AI TEDS control channel 8	0x02

Table : Index 611Ch

The object data value of 0 means, that the TEDS data is ignored, if a sensor with TEDS is connected. A value of 2 (default) means that the TEDS data is used, if a sensor with a valid and known TEDS template is connected.

This object is present as of device firmware 1.48 or higher

Index 6125h: AI Autozero

Writing the signature value 0x6f72657a ("zero") to this object will modify the input offset in such a way that the AI Input Process Value (**6130h**) and the AI Input Field Value (**9130h**) both become Zero.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned32	wo	AI Autozero channel 1	-
2	Unsigned32	wo	AI Autozero channel 2	-
3	Unsigned32	wo	AI Autozero channel 3	-
4	Unsigned32	wo	AI Autozero channel 4	-
5	Unsigned32	wo	AI Autozero channel 5	-
6	Unsigned32	wo	AI Autozero channel 6	-
7	Unsigned32	wo	AI Autozero channel 7	-
8	Unsigned32	wo	AI Autozero channel 8	-
9	Unsigned32	wo	AI Autozero channel 9	-
10	Unsigned32	wo	AI Autozero channel 10	-

Table : Index 6125h

This object is read-only on sub-index 0 and write-only on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).
Caution: - The AI Scaling Offset (**6127h**) is not accounted for during the Autozero operation.
- If measurement with Six-Axis sensor is acitive, it's necessary to write using complete access, since single channel zeroing is not allowed (if the user wats to modify particular physical channels, he may change the user Offset value by writing to Obj. 6127h).

Index 6126h: AI Scaling factor

This object represents the factor by which the Field Value is scaled in the calculation to create the Process Value (see **6130h**).

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Float	rw	AI Scaling factor channel 1	3.5
2	Float	rw	AI Scaling factor channel 2	3.5
3	Float	rw	AI Scaling factor channel 3	3.5
4	Float	rw	AI Scaling factor channel 4	3.5
5	Float	rw	AI Scaling factor channel 5	3.5
6	Float	rw	AI Scaling factor channel 6	3.5
7	Float	rw	AI Scaling factor channel 7	3.5
8	Float	rw	AI Scaling factor channel 8	3.5
9	Float	rw	AI Scaling factor channel 9	1
10	Float	rw	AI Scaling factor channel 10	1

Table : Index 6126h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).

Index 6127h: AI Scaling offset

This object represents the offset that is added to the scaled field value to create the process value (see **6130h**).

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Float	rw	AI Scaling offset channel 1	0.0
2	Float	rw	AI Scaling offset channel 2	0.0
3	Float	rw	AI Scaling offset channel 3	0.0
4	Float	rw	AI Scaling offset channel 4	0.0
5	Float	rw	AI Scaling offset channel 5	0.0
6	Float	rw	AI Scaling offset channel 6	0.0
7	Float	rw	AI Scaling offset channel 7	0.0
8	Float	rw	AI Scaling offset channel 8	0.0
9	Float	rw	AI Scaling offset channel 9	0.0
10	Float	rw	AI Scaling offset channel 10	0.0

Table : Index 6127h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).

Index 6130h: AI Process Value

This object represents the measuring values for each analogue input channel, which are processed AI Field Values. It is the result of the following Equation:

AIProcessValue = (((AIFieldValue * 1.05) / 8388608) * AIScalingFactor) + AIScalingOffset

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Float	ro	AI Process Value channel 1	-
2	Float	ro	AI Process Value channel 2	-
3	Float	ro	AI Process Value channel 3	-
4	Float	ro	AI Process Value channel 4	-
5	Float	ro	AI Process Value channel 5	-
6	Float	ro	AI Process Value channel 6	-
7	Float	ro	AI Process Value channel 7	-
8	Float	ro	AI Process Value channel 8	-
9	Float	ro	AI Process Value channel 9	-
10	Float	ro	AI Process Value channel 10	-

Table : Index 6130h

This object is read-only on sub-index 0 to 10. Any other access will result in an error.Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on, if module was enabled by serial interface).
This object on sub-index 1 to 10 can be mapped in the TxPDO (1A00h).

Index 6131h: AI Physical unit

This object represents the unit of the corresponding input (=Subindex). Changing the unit doesn't adjust the scaling automatically, i.e. after altering the unit, the AI process values remain the same.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned32	rw	AI Phys. unit channel 1	0xFD262600 (“mV/V“)
2	Unsigned32	rw	AI Phys. unit channel 2	0xFD262600 (“mV/V“)
3	Unsigned32	rw	AI Phys. unit channel 3	0xFD262600 (“mV/V“)

Sub-Index	Data Type	Access	Description	Default Value
4	Unsigned32	rw	AI Phys. unit channel 4	0xFD262600 (“mV/V“)
5	Unsigned32	rw	AI Phys. unit channel 5	0xFD262600 (“mV/V“)
6	Unsigned32	rw	AI Phys. unit channel 6	0xFD262600 (“mV/V“)
7	Unsigned32	rw	AI Phys. unit channel 7	0xFD262600 (“mV/V“)
8	Unsigned32	rw	AI Phys. unit channel 8	0xFD262600 (“mV/V“)
9	Unsigned32	rw	AI Phys. unit channel 9	0xFD262600 (“mV/V“)
10	Unsigned32	rw	AI Phys. unit channel 10	0xFD262600 (“mV/V“)

The following unit are used; coding is according to CiA 303-2 (CANopen standard):

Unit	**Coding (Data, hex)**	Code serial
mV/V	0xFD.26.26.00	0
kg	0x00.02.00.00	1
g	0x00.4B.00.00	2
N	0x00.21.00.00	3
cN	0xFE.21.00.00	4
V	0x00.26.00.00	5
µm/m	0xFA.01.01.00	6
(keine)	0x00.00.00.00	7
t	0x00.4C.00.00	8
kN	0x03.21.00.00	9
lb	**0x00.EA.00.00**	10
oz	**0x00.EB.00.00**	11
kp	**0x00.EC.00.00**	12
lbf	**0x00.ED.00.00**	13
pdl	**0x00.EE.00.00**	14
mm	0xFD.01.00.00	15
m	0x00.01.00.00	16
cNm	0xFE.56.00.00	17
Nm	0x00.56.00.00	18
°C	0x00.2D.00.00	19
°F	0x00.AC.00.00	20
K	**0x00.E8.00.00**	21
oztr	**0x00.E7.00.00**	22
dwt	**0x00.E6.00.00**	23
kNm	0x03.56.00.00	24

Unit	**Coding (Data, hex)**	Code serial
%	**0x00.E5.00.00**	25
0/00	**0x00.E4.00.00**	26
W	0x00.24.00.00	27
kW	0x03.24.00.00	28
rpm	0x00.00.47.00	29
bar	0x00.4E.00.00	30
Pa	0x00.22.00.00	31
hPa	0x02.22.00.00	32
MPa	0x06.22.00.00	33
N/mm^2	0x06.21.58.00	34
°	0x00.41.00.00	35
Hz	0x00.20.00.00	36
m/s	0x00.01.03.00	37
km/h	0x03.01.48.00	38
m^3/h	0x00.59.48.00	39
mA	0xFD.04.00.00	40
A	0x00.04.00.00	41
m/s^2	0x00.55.00.00	42
flbs	**0x00.E3.00.00**	43
ftlb	**0x00.E2.00.00**	44
J	0x00.23.00.00	45
kWh	**0x00.E1.00.00**	46
<User defined Text Nr. 1>	**0x00.FF.00.00**	-1
< User defined Text Nr. 2>	**0x00.FE.00.00**	-2

Codes shown in bold are manufacturer-defines, but follow the priciples of the CiA 303-2 standard.

Index 6148h: AI Span Start

This object specifies the lower limit of the process value (6130h). If a process value is equal or lower than this limit, the negative overload and invalid flag is set (6150h).

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x08

Sub-Index	Data Type	Access	Description	Default Value
1	Float	rw	AI Span start channel 1	-1048576
2	Float	rw	AI Span start channel 2	-1048576
3	Float	rw	AI Span start channel 3	-1048576
4	Float	rw	AI Span start channel 4	-1048576
5	Float	rw	AI Span start channel 5	-1048576
6	Float	rw	AI Span start channel 6	-1048576
7	Float	rw	AI Span start channel 7	-1048576
8	Float	rw	AI Span start channel 8	-1048576
9	Float	rw	AI Span start channel 9	-1048576
10	Float	rw	AI Span start channel 10	-1048576

Table : Index 6148h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).
With default values, no exceedance error will occur.

Index 6149h: AI Span End

This object specifies the upper limit to the process value (**6130h**). If a process value is higher or equal than this limit, the positive overload and invalid flag is set (**6150h**).

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Float	rw	AI Span end channel 1	1048576
2	Float	rw	AI Span end channel 2	1048576
3	Float	rw	AI Span end channel 3	1048576
4	Float	rw	AI Span end channel 4	1048576
5	Float	rw	AI Span end channel 5	1048576
6	Float	rw	AI Span end channel 6	1048576
7	Float	rw	AI Span end channel 7	1048576
8	Float	rw	AI Span end channel 8	1048576
9	Float	rw	AI Span end channel 9	1048576
10	Float	rw	AI Span end channel 10	1048576

Table : Index 6149h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).

With default values, no exceedance error will occur.

Index 6150h: AI Status

This object reflects the status of each Input Channel.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned8	ro	AI Status channel 1	0x00
2	Unsigned8	ro	AI Status channel 2	0x00
3	Unsigned8	ro	AI Status channel 3	0x00
4	Unsigned8	ro	AI Status channel 4	0x00
5	Unsigned8	ro	AI Status channel 5	0x00
6	Unsigned8	ro	AI Status channel 6	0x00
7	Unsigned8	ro	AI Status channel 7	0x00
8	Unsigned8	ro	AI Status channel 8	0x00
9	Unsigned8	ro	AI Status channel 9	0x00
10	Unsigned8	ro	AI Status channel 10	0x00

Table : Index 6150h

This object is read-only on sub-index 0 to 10. Any other Access will result in an Error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).

Each entry has the following bits defined:[9]

Bits 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
reserved	Object dictionary changed	TEDS sensor present	reserved	reserved	negative overload	positive overload	not valid

Table : Bit Mapping for Index 6150h

Bit 0: The sensor signal (Obj. 6130h and 9100h) is invalid. The reason can be:
- Sensor not connected
- Sensor broken
- Overflow to signal saturation (in combination with bit 1 or 2)

Bit 1 and 2: Positive or negative overload (mutual exclusive). The reason can be:
- Signal greater than value of Obj. 6148h / smaller that 6148h, respectively
- Signal greater that maximum/minimum as defined by sensor parameters (e.g. six-axis sensor)
- Numeric saturation of the signal

Bit 5: A sensor with valid TEDS (Transducer electronic data sheet acc. to IEEE1451) is connected and its data is used.

Bit 6: Any device parameter, that is communicated by the SDO dictionary, may have changed. The

[9] From Firmware Version 1.48. Before: Only Bits 2:0 implemented

user may read back the object dictionary data. Can be reset by using Obj. 6160h.

This object on sub-index 1 to 10 can be mapped in the TxPDO (1A00h).

Index 6160h: AI control byte

By writing to this object, a Set Zero routine can be performed (similar to obj. 6125h) and/or the "Object dictionary changed" flag of obj. 6150h can be reset.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned8	wo	AI Control byte channel 1	-
2	Unsigned8	wo	AI Control byte channel 2	-
3	Unsigned8	wo	AI Control byte channel 3	-
4	Unsigned8	wo	AI Control byte channel 4	-
5	Unsigned8	wo	AI Control byte channel 5	-
6	Unsigned8	wo	AI Control byte channel 6	-
7	Unsigned8	wo	AI Control byte channel 7	-
8	Unsigned8	wo	AI Control byte channel 8	-
9	Unsigned8	wo	AI Control byte channel 9	-
10	Unsigned8	wo	AI Control byte channel 10	-

Table : Index 6150h

Bits 7:4	Bit 3	Bit 2	Bit 1	Bit 0
reserved	Reset bit OD changed	reserved	Perform Auto-Zero	reserved

Table : Bit Mapping for Index 6160h

This object is present as of device firmware 1.48 or higher

Index 61A0h: AI Filter type

This object defines the filter used for calculating the field value (9100h). The only implemented filter is a ‘Repeating Average’ filter (Filter Type 2), that is always active.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned8	ro	AI Filter type channel 1	0x02
2	Unsigned8	ro	AI Filter type channel 2	0x02
3	Unsigned8	ro	AI Filter type channel 3	0x02
4	Unsigned8	ro	AI Filter type channel 4	0x02
5	Unsigned8	ro	AI Filter type channel 5	0x02

Sub-Index	Data Type	Access	Description	Default Value
6	Unsigned8	ro	AI Filter type channel 6	0x02
7	Unsigned8	ro	AI Filter type channel 7	0x02
8	Unsigned8	ro	AI Filter type channel 8	0x02
9	Unsigned8	ro	AI Filter type channel 9	0x02
10	Unsigned8	ro	AI Filter type channel 10	0x02

Table : Index 61A0h

This object is read-only on sub-index 0 to 10. Any other access will result in an error. Subindices 9 and 10 correspond to the counter input module of the GSV-8 (available from Firmware version 1.45 on).

Index 61A1h: AI Filter constant

This object defines the number of samples used by the average filter.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned16	rw	AI Filter constant channel 1	0x12C0
2	Unsigned16	rw	AI Filter constant channel 2	0x12C0
3	Unsigned16	rw	AI Filter constant channel 3	0x12C0
4	Unsigned16	rw	AI Filter constant channel 4	0x12C0
5	Unsigned16	rw	AI Filter constant channel 5	0x12C0
6	Unsigned16	rw	AI Filter constant channel 6	0x12C0
7	Unsigned16	rw	AI Filter constant channel 7	0x12C0
8	Unsigned16	rw	AI Filter constant channel 8	0x12C0
9	Unsigned16	rw	AI Filter constant channel 9	0x12C0
10	Unsigned16	rw	AI Filter constant channel 10	0x12C0

Table : Index 61A1h

This object is read-only on sub-index 0 and read-/writeable on sub-indices 1 to 10. Any other access will result in an error.
Only integers from 4 to 48000 are allowed as filter constants. Trying to set another value will result in an error and no change will be made to this setting.
Any write will set all entries to the exact same value.
The default value is 4800d = 12C0h because of the default data rate of 10 value frames per second. The number of value frames the device acquires per second is: 48000 / AI Filter constant. The corresponding update cycle time is: (AI Filter constant / 48000) s.

Index 9100h: AI Field Value

This object represents the analog input raw value in a twos complement 32Bit integer format for each input channel. Unlike the AI process value, it's not scaled by AI scaling factor.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x08
1	Integer32	ro	AI Field Value channel 1	-
2	Integer32	ro	AI Field Value channel 2	-
3	Integer32	ro	AI Field Value channel 3	-
4	Integer32	ro	AI Field Value channel 4	-
5	Integer32	ro	AI Field Value channel 5	-
6	Integer32	ro	AI Field Value channel 6	-
7	Integer32	ro	AI Field Value channel 7	-
8	Integer32	ro	AI Field Value channel 8	-

Table : Index 9100h

This object is read-only on sub-index 0 to 8. Any other access will result in an error.
This object on sub-index 1 to 8 can be mapped in the TxPDO (1A00h).

Process Data Object (PDO)

The EtherCAT master requests PDOs by sending out PDO frames, which it does typically in a cyclic manner. EtherCAT devices (slaves) add their PDO-in data to these frames.

The GSV-8 adds a data field of a maximum of 96 bytes. The PDO data, its actual size, entries and their meaning are all defined by the content of the SDO 1A00h (see above).

The PDO cycle time for that the master is configured is typically the same or a bit lower than the data sampling frequency of the measuring application, so that each PDO contains actual measuring values. The measuring data sampling frequency can be set with SDO 61A1, its maximum is typically 12000 frames/s (depending on other settings), corresponding to a minimum recommended PDO cycle time of 0.084 ms.

Change log

Version	Date	Changes
5	2018	Structure improved, redundancy deleted, updated
6	04/2019	Updated
7	08/2019	Updated (to FW-ver 1.50), comm. services clarified
8	09/08/2019	Missing SDOs completed, reviewed
9	02/2021	Company adress, reviewed

Subject to modifications

All information herein describes our products in a general manner. They mean no assurance of properties in terms of §459 Abs. 2 BGB and will not cause any liabilities.

Trademarks and Patents

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

Made in Germany

Copyright © 1999-2021

ME-Meßsysteme GmbH