KxD0140 / K6D55ec / GSV-61
FT sensors with measuring amplifier GSV-61
EtherCat Protocol

Last edit: 13/06/2024 Applicable device software revision: from 3.47

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

The customer interface of the GSV-61 measuring amplifier and the K6D55ec/KxD0140 6-axis Force/Torque sensor follows the EtherCAT field bus standard. The application layer is implemented according to the EtherCAT standard, using CoE (CANopen over EtherCAT). The EtherCAT protocol is defined in IEC61158. The device is supplied by EtherCAT-P, i.e. through the EtherCAT interface. The K6D55ec smart FT sensor contains a GSV-61 amplifier.

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

The GSV-61/K6D55ec/KxD0140 is powered through the M8 EtherCAT connector conforming to the EtherCAT P technology.

Signal	Function	Pin-No. M8 connector	Pin-No. RJ45
TD+	Transmit data +, GNDs	1	1
TD-	Transmit data - , Us:+24VDC	4	2
RD+	Receive data +	2	3
RD-	Receive data -	3	6

The device requires an EtherCAT P power source device, that superposes the supplying DC and splits the Ethernet signals for the EtherCAT master.

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. 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.

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-61/K6D55ec/KxD0140 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

Field	Value / Description
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)
- 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-61/K6D55ec/KxD0140 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.

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 KxD0140

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.
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.

CMD	Abbr.	Name	Description
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-61/K6D55ec/KxD0140 EtherCat Implementation

K6D55ec and KxD0140 have a build-in measuring amplifier that has six analogue inputs, which are digitized with a 24 Bit Sigma-Delta analog-to-digital converter, who converts all channels simultaneously and then calculates physically scaled process values. The same applies to the GSV-61 EtherCAT amplifier. In conjunction with the buffered-mode Sync-Manager of the EtherCat interface, the data of the six channels representing the forces and torques belong exactly to the same point of time. The rate, with which the measuring controller updates the values, is configurable from 1 to 1200 values/second.
The KxD0140 implements the parts of the CoE CanOpen 404 device profile, which are 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 6 often refer to the corresponding analogue input channel configuration.

The EtherCAT communication interface is described by a standardized device description

file (available from the vendor), the **ESI file**. It has a specialized xml format. Many EtherCAT master programs can read it, so that they have information about the communication properties of the device, e.g. the SDO object dictionary and the PDO properties.

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	**Not used**. (Process data from master to device)
Sync Manager 4 (SM4)	3	PDO in	Process data from device to master. Used for measurement data and sensor status. Mapping configurable (see SDO 1A00 description).

Service-Data Objects (SDO)

The relevant device parameters can be accessed through SDOs. 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
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)
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 0 method.

After a measurement data frame acquisition is completed, 1.4 µs later the slave controller captures the system time of the positive edge of its Latch 0 input (ESC registers

0x09C0..0x09C7).[1]

When the calculation of the physical PDO values is completed, the system time of the negative edge of the Latch 0 input is captured by the ESC (registers 0x09C8..0x09CF).11

1 Not yet available with Firmware Rev. 3.45

Object Dictionary

This chapter specifies the objects of the GSV-61/K6D55ec/KxD0140-EtherCAT implementation. Many of them follow the CANopen device profile 404, since is's a CoE device.

Index (Hex)	Name (as in ESI-file & device)	Type	Page
6112	Operating Mode	Application, Analog Input	27
6114	ADC Sample Rate	Application, Analog Input	28
6125	Autozero	Application, Analog Input	28
6126	Scaling Factor	Application, Analog Input	28
6127	Scaling Offset	Application, Analog Input	29
6130	FLOAT Process Value	Application, Analog Input	29
6131	Physical Unit	Application, Analog Input	30
6148	FLOAT Span Start	Application, Analog Input	32
6149	FLOAT Span End	Application, Analog Input	32
6150	AI Status	Application, Analog Input	33
6160	AI control byte	Application, Analog Input	34
9100	INT32 Field Value	Application, Analog Input	34

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	0x80020194

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 0x80020194 contains the following two fields:

- Device Profile Number 0194h = 404d
- Additional Information 02h = 00000010b

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

- Bit 17: Analogue input 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.
This object is reserved for future use and reads always 0.

Index 1008h: Device Name

This object contains the device name.

Sub-Index	Data Type	Access	Description	Default Value
0	String(7)	ro	Device Name	"KxD0140" or "GSV-61"

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(3)	ro	Hardware version	"0.1"

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	"03.47" (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	1

Sub-Index	Data Type	Access	Description	Default Value
1	Unsigned32	rw	Store EtherCAT application Objects	1

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

Value	Meaning
0	At this Parameter set =Subindex no values are stored
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 (1A00h, 6148h and 6149h) can be stored. When saving, a checksum value is written to Object 0x10F0.

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

Storage write access signature

Remark: The KxD0140 stores most measuring application parameters by itself (unless otherwise noted in the object description). 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.

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	1
1	Unsigned32	rw	Restore all parameters	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)

Value	Meaning
1	From this Parameter set =Subindex, parameters can be loaded

By writing the Signature "load", subindex 1, parameters can be restored as follows:

Subindex 1: EtherCAT application objects (0x6148, 0x6149 and 0x1A00) are restored.

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	0x00060001 or 0x0006008C
3	Unsigned32	ro	Revision	0x0003002F (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. It is: 0x00060001 for GSV-61 or 0x0006008C for KxD0140.[2]

Sub-Index 3: Revision

The revision is the binary equivalent to the software version in Obj. 100Ah. Bits<23:16> contain the major revision No, bits <16:0> the minor rev. No.

Sub-Index 4: Serial Number

The serial number in its decimal representation. It can be found on the device's type plate.

2 0x00060000h is the preliminary code for KxD0140 proto-type

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	ro	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	0x00000001
2	Unsigned16	rw	Sync Error Counter Limit	0x0004

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 (PreOp)	Highest sub-index= Number of mapped objects in the Out-PDO	0x0C
1	Unsigned32	rw (PreOp)	Mapped Object 1 = 1st Object in the PDO	0x61300120 =AI Process Value of channel 1
2	Unsigned32	rw (PreOp)	Mapped Object 2 = 2nd Object in the PDO	0x61300220 =AI Process Value of channel 2
3	Unsigned32	rw (PreOp)	Mapped Object 3 = 4rd Object in the PDO	0x61300320 =AI Process Value of channel 3
4	Unsigned32	rw (PreOp)	Mapped Object 4 = 4th Object in the PDO	0x61300420 =AI Process Value of channel 4
5	Unsigned32	rw (PreOp)	Mapped Object 5 = 5th Object in the PDO	0x61300520 =AI Process Value of channel 5
6	Unsigned32	rw (PreOp)	Mapped Object 6 = 6th Object in the PDO	0x61300620 =AI Process Value of channel 6
7	Unsigned32	rw (PreOp)	Mapped Object 7 = 7th Object in the PDO	0x61500120 =AI Status of channel 1
8	Unsigned32	rw (PreOp)	Mapped Object 8 = 8th Object in the PDO	0x61500220=AI Status of channel 2
9	Unsigned32	rw (PreOp)	Mapped Object 9 = 9th Object in the PDO	0x61500320=AI Status of channel 3
10	Unsigned32	rw (PreOp)	Mapped Object 10 = 10th Object in the PDO	0x61500420=AI Status of channel 4
11	Unsigned32	rw (PreOp)	Mapped Object 11 = 11th Object in the PDO	0x61500520=AI Status of channel 5
12	Unsigned32	rw (PreOp)	Mapped Object 12 = 12th Object in the PDO	0x61500620=AI Status of channel 6

Table : Index 1A00h. Default mapping shown only.

This object implements the sub-index 0 to 18 for read/write-access. A write to sub-index 0 with a value greater than 18 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 these 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 18). 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) in the TxPDO mapping object entries. See the Object description on info about their meaning and their sub-indices.
- The altered mapping is not stored automatically to non-volatile memory, i.e. after rebooting, the default mapping is set again. The mapping can be stored by writing to SDO 1010.1h (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.
This object has no active RxPDO (sub-index 0 is 0x00), therefore the value of sub-index 1 is left unspecified. GSV-61/K6D55ec/KxD0140 doesn't use RxPDO.

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 (PreOp)	Synchronization type	0x0001
2	Unsigned32	ro	Cycle time	0x00000000
3	-	ro	Unused	-
4	Unsigned16	ro	Synchronization types supported	0x4003
5	Unsigned32	ro	Minimum cycle time	0x0005C814
6	Unsigned32	ro	Calc and copy time	0x0001D4C0
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 in ns.

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 in ns between two SyncManager events which the

application supports.

Sub-Index 6: Calc and Copy Time

Time in ns 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 2020h FT Sensor Information

The Object 2020h can be used to determine the number of stored data sets for six-axis sensors (=FT sensors) and to read and set the activated array. The calculation of the physical six-axis values can also be enabled or disabled.

Sub-Index	Data type	Access	Meaning	Default value
0	unsigned8	ro	Highest sub-index supported	0x03
1	unsigned8	rw (PreOp)	Number of the activated FT sensor	0x00
2	unsigned8	ro	Number of saved FT data records	0x00
3	unsigned8	ro	Maximum number of FT savable records	0x08

Sub-indices 0 to 3 are supported. An access to other sub-indexes is acknowledged with an error message. This object is Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.

Sub-Index 1:
The value range of the number of activated FT sensor at sub-index 1 is:
0 to <number of saved FT records>. A value of 0 means that the calculation of the physical six-axis values is deactivated, so that the default TxPDOs No. 1 to 3 (= Obj. 6130.1-6) display raw values, scaled in mV/V.
If the value is greater than 0, it corresponds to the number of the sensor data that is activated. Writing a value >0 stores the data if Obj. 2021 and 202B ... 2024 and 202E in non-volatile memory and activates the FT sensor calculation.
The number of the sensor data corresponds to the FT sensor data objects as follows:

Number	Index of FT Basic Data	Index of FT Sensor Extended Data
1	2021h	202Bh
2	2022h	202Ch
3	2023h	202Dh
4	2024h	202Eh

If FT sensor calculation is activated, the measured values have the following meaning:

Channel No.	Meaning	Physical unit
1	Force in X direction	N
2	Force in Y direction	N
3	Force in Z direction	N
4	Torque in X direction	Nm
5	Torque in Y direction	Nm
6	Torque in Z-direction	Nm

Writing a number 1..<number of saved FT records> activates the corresponding sensor data for matrix calculation.
Sub-Index 2:
Number of saved FT records. At this sub-index, the number of sensor record that are already saved can be read. It also defines the maximum for activating and storing sensor data, which is: No. of saved data records +1.
Sub-Index 3:
The maximum number of sensor record sets that can be stored can be read here. GSV-61/K6D55ec/KxD0140 with FW-Rev. 3.45 only supports one FT sensor matrix No. 1. but those with Rev. 3.47 supports up to 4 sensor data sets.

Index 2021h..Index 2024h FT Sensor Basic Data 1..4

With the objects 2021h..2024h, the contents of the basic six-axis sensor data sets can be read or written (Write in PreOP only). These objects are "Manufacturer defined"
The object 2021h is assigned to the FT sensor No. 1, Obj. 2022h No. 2, etc. until 2024h the

sensor No. 4. Data read of a sensor with a No. greater than the value of Obj. 2020.2h is not meaningful. The layout of the objects 2021h to 2024h is identical:

Sub-Index (dec)	Data type	Access	Meaning	Physic. Unit	Default-value
0	unsigned8	ro	Highest sub-index supported	-	55
1	unsigned32	rw (PreOp)	**Serial number of the FT sensor**	-	0
2	Float	rw (PreOp)	**Maximum value of force in X direction (Fx max)**	N	0
3	Float	rw (PreOp)	**Maximum value of force in Y direction (Fy max)**	N	0
4	Float	rw (PreOp)	**Maximum value of force in Z-direction (Fz max)**	N	0
5	Float	rw (PreOp)	**Maximum torque in X direction (Mx max)**	Nm	0
6	Float	rw (PreOp)	**Maximum torque in Y direction (My max)**	Nm	0
7	Float	rw (PreOp)	**Maximum torque in Z direction (Mz max)**	Nm	0
8	Float	rw (PreOp)	Geometric offset in X direction	m	0
9	Float	rw (PreOp)	Geometric offset in Y direction	m	0
10	Float	rw (PreOp)	Geometric offset in Z direction	m	0
11	Float	rw (PreOp)[3]	Zero signal of component 0	mV/V	0
12	Float	rw (PreOp)[3]	Zero signal of component 1	mV/V	0
13	Float	rw (PreOp)[3]	Zero signal of component 2	mV/V	0
14	Float	rw (PreOp)[3]	Zero signal of component 3	mV/V	0
15	Float	rw (PreOp)[3]	Zero signal of component 4	mV/V	0
16	Float	rw (PreOp)[3]	Zero signal of component 5	mV/V	0
17	unsigned32	rw (PreOp)	Sensor Type in Bits<7:0>: =0: Standard solution =1: With "Matrix Plus" 2nd Order =2: Matrix sizes MxN defined: Number of rows M in Bits<15:12> of this value, number of columns in Bits<19:16>	-	0
18	Float	rw (PreOp)	**Matrix scaling value Out. If=1: output in physical units**	-	0
19	Float	rw (PreOp)	**Matrix scaling value In. If =1: Raw value units mV/V.**	-	0
20...55	Float	rw (PreOp)	**Matrix Elements, from 1.1 ... 1.6 (1st row), then 2.1...2.6 to 6.1 ... 6.6 (6th row)**	N/mV/V or Nm/mV/V	0

Only sub-indices 0 to 55 are supported. An access to other sub-indexes is acknowledged with an error message. Values shown in bold should be set as a minimum requirement. This object is

3 RW (PreOp) from FW-Ver 3.48. Beore, it's RO

Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.
After writing to the object, the data is NOT stored to non-volatile memory! Storing is performed by writing the corresponding number >0 to Obj. 2020.1 (see p. 22), which will also store corresponding Obj. 202B..202E content.

Index 202Bh..Index 202Eh FT Sensor Extended Data 1..4

With the objects 202Bh..202Eh, the contents of the extended six-axis sensor data sets can be read or written (Write in PreOP only). These objects are "Manufacturer defined"
The object 202Bh is assigned to the FT sensor No. 1, Obj. 202Ch No. 2, etc. until 202Eh the sensor No. 4. Data read of a sensor with a No. greater than the value of Obj. 2020.2h is not meaningful. The layout of the objects 202Bh to 202Eh is identical:

Sub-Index (dec)	Data type	Access	Meaning	Physic. Unit	Default-value
0	unsigned8	ro	Highest sub-index supported	-	58
1	String	rw (PreOp)	Model name of the sensor	-	"Sensor 1..4"
2	unsigned8	rw (PreOp)	Unit Enumerator for Fx (see 6131h, p.30)	-	3 (=N)
3	unsigned8	rw (PreOp)	Unit Enumerator for Fx (see 6131h, p.30)	-	3 (=N)
4	unsigned8	rw (PreOp)	Unit Enumerator for Fz (see 6131h, p.30)	-	3 (=N)
5	unsigned8	rw (PreOp)	Unit Enumerator for Mx (see 6131h, p.30)	-	18 (=Nm)
6	unsigned8	rw (PreOp)	Unit Enumerator for My (see 6131h, p.30)	-	18 (=Nm)
7	unsigned8	rw (PreOp)	Unit Enumerator for Mz (see 6131h, p.30)	-	18 (=Nm)
8	unsigned8	rw (PreOp)	Index of 1st factor 1st input for Matrix B	-	0
9	unsigned8	rw (PreOp)	Index of 1st factor 2nd input for Matrix B input	-	0
10	unsigned8	rw (PreOp)	Index of 1st factor 3rd input for Matrix B input	-	0
11	unsigned8	rw (PreOp)	Index of 1st factor 4th input for Matrix B input	-	0
12	unsigned8	rw (PreOp)	Index of 1st factor 5th input for Matrix B input	-	0
13	unsigned8	rw (PreOp)	Index of 1st factor 6th input for Matrix B input	-	0
14	unsigned8	rw (PreOp)	Index of 2nd factor 1st input for Matrix B input	-	0
15	unsigned8	rw (PreOp)	Index of 2nd factor 2nd input for Matrix B input	-	0
16	unsigned8	rw (PreOp)	Index of 2nd factor 3rd input for Matrix B input	-	0
17	unsigned8	rw (PreOp)	Index of 2nd factor 4th input for Matrix B input	-	0
18	unsigned8	rw (PreOp)	Index of 2nd factor 5th input for Matrix B input	-	0
19	unsigned8	rw (PreOp)	Index of 2nd factor 6th input for Matrix B input	-	0

20...55	Float	rw (PreOp)	Matrix B Elements, from 1.1 ... 1.6 (1st row), then 2.1...2.6 to 6.1 ... 6.6 (6th row)	N/(mV/V)² or Nm/(mV/V)²	0
56	unsigned8	rw (PreOp)	Calibration date Day (1..31)	-	0
57	unsigned8	rw (PreOp)	Calibration date Month (1..12)	-	0
58	unsigned16	rw (PreOp)	Calibration date Year	-	0

Only sub-indices 0 to 58 are supported. An access to other sub-indexes is acknowledged with an error message. This object is Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.
This object shall contain meaningful data only if data is or will be stored (see 2020.1h and 2020.2h) and if the Sensor Type defined in 2021-2024.17 is =1 = "Matrix Plus". For other Sensor Types, the data is optional and can be left as default.
After writing to the object, the data is NOT stored to non-volatile memory! Storing is performed by writing the corresponding number >0 to Obj. 2020.1 (see p. 22) which will also store corresponding Obj. 2021..2024 content.

Index 2035h: AI Input Sensitivity

This object represents the input sensitivity, which is composed of the total input range (see 2036h) and an internal scaling factor. It also defines the measurable range in mV/V.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x06
1	Float	rw (PreOp)	AI Sensitivity channel 1	2
2	Float	rw (PreOp)	AI Sensitivity channel 2	2
3	Float	rw (PreOp)	AI Sensitivity channel 3	2
4	Float	rw (PreOp)	AI Sensitivity channel 4	2
5	Float	rw (PreOp)	AI Sensitivity channel 5	2
6	Float	rw (PreOp)	AI Sensitivity channel 6	2

Only sub-indices 0 to 6 are supported. An access to other sub-indexes is acknowledged with an error message. This object is Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.

Index 2036h: AI Input Range

This object represents the input range, which is composed of the maximum input range and an internal gain factor, which can be 1, 2, 4 or 8. Therefore, only discrete values can be set, defined by Obj. 2037h. The value is given in Range [mV/V] x 100.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x06

Sub-Index	Data Type	Access	Description	Default Value
1	Unsigned32	rw (PreOp)	AI Range channel 1 x100	400
2	Unsigned32	rw (PreOp)	AI Range channel 2 x100	400
3	Unsigned32	rw (PreOp)	AI Range channel 3 x100	400
4	Unsigned32	rw (PreOp)	AI Range channel 4 x100	400
5	Unsigned32	rw (PreOp)	AI Range channel 5 x100	400
6	Unsigned32	rw (PreOp)	AI Range channel 6 x100	400

Only sub-indices 0 to 58 are supported. An access to other sub-indexes is acknowledged with an error message. This object is Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.

Index 2037h: AI Supported Input Ranges

This Object lists all available input ranges, given in mV/V x100.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x04
1	Unsigned32	ro	AI Range for gain=1 x100	400
2	Unsigned32	ro	AI Range for gain=2 x100	200
3	Unsigned32	ro	AI Range for gain=4 x100	100
4	Unsigned32	ro	AI Range for gain=8 x100	50

This object is read-only. Any other access will result in an error. This object is Manufacturer defined, i.e. not predefined in the CANopen nor in the EtherCAT standard.

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	0x06
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

This object is read-only on sub-index 0 and read/write on sub-index 1 to 6. Any other access will result in an error.

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 in µs. Writing to this object is possible in PreOP state and will change the sampling rate of the AD conversion. The default value corresponds to 10 values/s x6 channles.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Unsigned32	rw (PreOp)	AI ADC sample rate channel 1	0x000186A0
2	Unsigned32	rw (PreOp)	AI ADC sample rate channel 2	0x000186A0
3	Unsigned32	rw (PreOp)	AI ADC sample rate channel 3	0x000186A0
4	Unsigned32	rw (PreOp)	AI ADC sample rate channel 4	0x000186A0
5	Unsigned32	rw (PreOp)	AI ADC sample rate channel 5	0x000186A0
6	Unsigned32	rw (PreOp)	AI ADC sample rate channel 6	0x000186A0

Only sub-indices 0 to 6 are supported. An access to other sub-indexes is acknowledged with an error message.

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	-

Table : Index 6125h This object is read-only on sub-index 0 and write-only on sub-index 1 to 6. Any other access will result in an error.

Caution: - The AI Scaling Offset (6127h) is not accounted for during the Autozero operation. - If measurement with Six-Axis sensor is active (see 2021.1h), all input channels will be set to zero, regardless of the sub-index . Nevertheless, if the user wants 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). With FT sensor activated, the content of this object

has no effect.

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A
1	Float	rw (PreOp)	AI Scaling factor channel 1	2
2	Float	rw (PreOp)	AI Scaling factor channel 2	2
3	Float	rw (PreOp)	AI Scaling factor channel 3	2
4	Float	rw (PreOp)	AI Scaling factor channel 4	2
5	Float	rw (PreOp)	AI Scaling factor channel 5	2
6	Float	rw (PreOp)	AI Scaling factor channel 6	2

Table : Index 6126h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 6 in PreOP state only. Any other access will result in an error.

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 (PreOp)	AI Scaling offset channel 1	0.0
2	Float	rw (PreOp)	AI Scaling offset channel 2	0.0
3	Float	rw (PreOp)	AI Scaling offset channel 3	0.0
4	Float	rw (PreOp)	AI Scaling offset channel 4	0.0
5	Float	rw (PreOp)	AI Scaling offset channel 5	0.0
6	Float	rw (PreOp)	AI Scaling offset channel 6	0.0

Table : Index 6127h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 6 in PreOP state only. Any other access will result in an error.

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[4]
If FT sensor is enabled, Sub-Indices 1..3 represent Fx,y,z and 4..6 Mx,y,z. In that case, the above equation doesn't apply.

4 This equation doesn't apply if FT sensor is enabled.

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	-

Table : Index 6130h

This object is read-only on sub-index 0 to 6. Any other access will result in an error.
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 (PreOp)	AI Phys. unit channel 1	0xFD262600 (“mV/V“)
2	Unsigned32	rw (PreOp)	AI Phys. unit channel 2	0xFD262600 (“mV/V“)
3	Unsigned32	rw (PreOp)	AI Phys. unit channel 3	0xFD262600 (“mV/V“)
4	Unsigned32	rw (PreOp)	AI Phys. unit channel 4	0xFD262600 (“mV/V“)
5	Unsigned32	rw (PreOp)	AI Phys. unit channel 5	0xFD262600 (“mV/V“)
6	Unsigned32	rw (PreOp)	AI Phys. unit channel 6	0xFD262600 (“mV/V“)

This object is read-only on sub-index 0 and read/write on sub-index 1 to 6 in PreOP state only. Any other access will result in an error.
The following unit are used; coding is according to CiA 303-2 (CANopen standard):

Unit	**Coding (Data, hex)**	Enum Obj. 202B..202E.2..7
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
(none)	0x00.00.00.00	7
t	0x00.4C.00.00	8

Unit	**Coding (Data, hex)**	Enum Obj. 202B..202E.2..7
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
%	**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/mm²	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
m³/h	0x00.59.48.00	39
mA	0xFD.04.00.00	40
A	0x00.04.00.00	41

Unit	Coding (Data, hex)	Enum Obj. 202B..202E.2..7
m/s²	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-defined, but follow the principles 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).[5]

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x08
1	Float	rw (PreOp)	AI Span start channel 1	-1048576
2	Float	rw (PreOp)	AI Span start channel 2	-1048576
3	Float	rw (PreOp)	AI Span start channel 3	-1048576
4	Float	rw (PreOp)	AI Span start channel 4	-1048576
5	Float	rw (PreOp)	AI Span start channel 5	-1048576
6	Float	rw (PreOp)	AI Span start channel 6	-1048576

Table : Index 6148h This object is read-only on sub-index 0 and read/write on sub-index 1 to 6. Any other access will result in an error. With default values, no exceedance error will occur.

After writing to this object, the data won't be stored to non-volatile memory. To store them, Obj. 1010.1 must be used.

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).p.32

Sub-Index	Data Type	Access	Description	Default Value
0	Unsigned8	ro	Highest sub-index	0x0A

5 This behaviour is implemented from Rev.No 3.46 on.

Sub-Index	Data Type	Access	Description	Default Value
1	Float	rw (PreOp)	AI Span end channel 1	1048576
2	Float	rw (PreOp)	AI Span end channel 2	1048576
3	Float	rw (PreOp)	AI Span end channel 3	1048576
4	Float	rw (PreOp)	AI Span end channel 4	1048576
5	Float	rw (PreOp)	AI Span end channel 5	1048576
6	Float	rw (PreOp)	AI Span end channel 6	1048576

Table : Index 6149h

This object is read-only on sub-index 0 and read/write on sub-index 1 to 6. Any other access will result in an error.
With default values, no exceedance error will occur.

After writing to this object, the data won't be stored to non-volatile memory. To store them, Obj. 1010.1 must be used.

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

Table : Index 6150h

This object is read-only on sub-index 0 to 6. Any other Access will result in an Error.

Each entry has the following bits defined:

Bits 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
reserved	Object dictionary changed	reserved	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 6: Any device parameter, that is communicated by the SDO dictionary, may have changed. The user may read back the object dictionary data. Can be reset by using Obj. 6160h.

This object on sub-index 1 to 6 can be mapped in the TxPDO (1A00h).

Index 6160h: AI Control Byte

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	wo	AI Control Byte ch. 1	0x00
2	Unsigned8	wo	AI Control Byte ch. 2	0x00
3	Unsigned8	wo	AI Control Byte ch. 3	0x00
4	Unsigned8	wo	AI Control Byte ch. 4	0x00
5	Unsigned8	wo	AI Control Byte ch. 5	0x00
6	Unsigned8	wo	AI Control Byte ch. 6	0x00

Table : Index 6160h

This object triggers some function when written:

Bits 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
reserved	reserved	reserved	reserved	Reset bit OD changed	reserved	Auto-zero	reserved

Bit 3 / Value=8: Reset Bit 6 of Obj. 6150.n (all channels). Bit 6 of 6150.n and this flag is the same for all channels.

Bit 1 / Value=2: Trigger the the Set zero function (see object 6125 h) at specified channel(s). If Multi-axis measuring is enabled (Obj. 2020.1>0), writing 2 to any sub-index will perform Set zero at all channels.

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. The positive full-scale input is represented by 0x0079E79E and the negative by 0xFF861862.

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

Sub-Index	Data Type	Access	Description	Default Value
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	-

Table : Index 9100h

This object is read-only on sub-index 0 to 6. Any other access will result in an error.
This object on sub-index 1 to 6 can be mapped in the TxPDO (1A00h).
The raw value in mV/V can be calculated as follows:
Raw value= (Obj.9100.n data / 7989150d) x InputSensitivity
The InputSensitivity can be read by Obj. 2035.n

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 KxD0140 adds a data field of a maximum of 54 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 6114h, its maximum is typically 1200 frames/s, corresponding to a minimum recommended PDO cycle time of 0.84 ms.

Change log

Version	Date	Changes
1	2024	Initial release, valid for device software ver. 3.45
2	05/2024	New objects, valid for device software ver. 3.47
3	06/2024	Added GSV-61, some clarifications

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