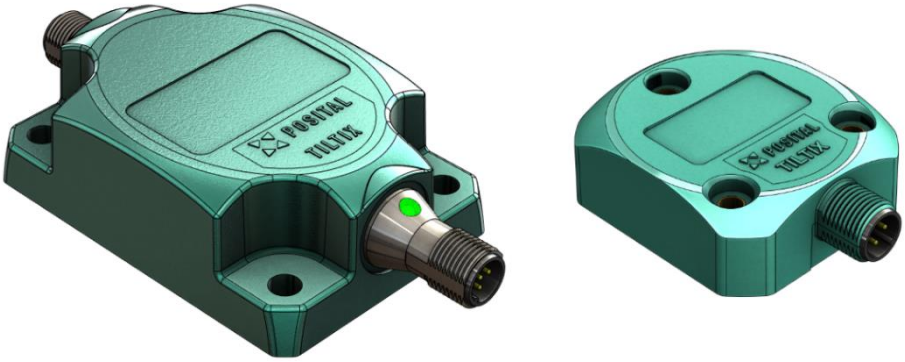
TILTIX MEMS ACS INCLINOMETER
WITH CANOPEN INTERFACE



User Manual

TILTIX MEMS ACS INCLINOMETER
WITH CANOPEN INTERFACE

1 Content

2 General Safety Advice

2.1 Important Information

Read these instructions carefully, and have a look at the equipment to become familiar with the device before trying to install, operate, or maintain it.

The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention towards information that clarifies/simplifies a procedure.

Please Note: Electrical equipment should be serviced only by qualified personnel. No responsibility is assumed by POSITAL for any consequences arising out of the use of this material. This document is not intended as an instruction manual for untrained personnel.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

3 About this Manual

3.1 Background

This user manual explains how to install and configure the TILTIX ACS inclinometer with a CANopen interface with illustrations from a Schneider TWIDO® PLC.

3.2 Version Management

- Updated On: 20231027
- Document Name: Manual_ACS3_CANopen_General.pdf

3.3 Imprint

FRABA B.V.
Jan Campertstraat 5,
NL-6416 SG Heerlen
T +49 221-96213-0
info@fraba.com, www.fraba.com

3.4 Copyright

CANopen® and CiA® are registered community trademarks of CAN in Automation e.V.

3.5 User Annotation

All readers are highly welcome to send us feedback and comments about this document.
Depending on your region you can reach us by e-mail at the following e-mail addresses. For America info@fraba.com, for Asia info@fraba.sg, for Europe info@fraba.de.

3.6 Document History

Document	Manual_ACS_CANopen_General.pdf
Created	2022/05
Author	ADO
Reviewers	TSC/ALI
Versions	20240517

4 Introduction

This manual explains how to install and configure the TILTIX ACS inclinometers with a CANopen interface, suitable for industrial, military and heavy duty applications. The TILTIX CANopen is based on the DS410 standard.

4.1 TILTIX Inclinomter

TILTIX inclinometers sense and measure the angle of tilt (inclination/slope/elevation) of an object with respect to the force of gravity.

The basic principle behind this TILTIX inclinometer is a Micro-Electro-Mechanical Systems (MEMS) sensor cell that is embedded to a fully molded ASIC. The angle is measured with the relative change in electrical capacitance in the MEMS cell.

4.2 CANopen Interface

CANopen is based on the Controller Area Network (CAN) that was developed by automotive industries in the 80s. The application protocol CANopen was introduced by the multi vendor association CAN in Automation (CiA) to ensure a full compatibility of industrial automation products. It is a multiple access system (maximum: 127 devices), which means that all devices can access the bus. These devices are the components of the CANopen bus.

In simple terms, CANopen works as a client-server model. Each device checks whether the bus is free, and if it is free the device can send messages. If two devices try to access the bus at the same time, the device with the higher priority level has permission to send its message first. Devices with the lowest priority level must cancel their data transfer and wait before re-trying to send their message.

4.3 TILTIX ACS CANopen

The TILTIX CANopen inclinometer corresponds to the inclinometer profile DS 410 CANopen standards. In addition to high resolution, accuracy and protection class of IP69K, the inclinometer has a built-in active temperature compensation. This makes TILTIX ACS suitable for machines, harsh environments and versatile applications in industrial, heavy duty and military applications.

The inclinometer supports the following operating modes:

- Polled mode: The position value is transmitted only on request.
- Cyclic mode: The position value is sent cyclically (regular, adjustable intervals) on the bus.
- SYNC mode: The position value is sent after a synchronization message (SYNC) is received. The position value is sent every n SYNCs (n ≥ 1).
- State change mode: The position value is transmitted whenever the position of the

inclinometer changes (minimum time interval settable).

Various software tools for configuration and parameter-setting are available from different suppliers. It is easy to align and program the inclinometers using the EDS (electronic data sheet) configuration file provided. (Refer to section 3.1)

4.4 Typical Applications of TILTIX ACS

- Cranes and Construction Machinery
- Mobile Machinery
- Agriculture Machinery
- Elevated Platforms
- Mobile Lifts and Fire Engines
- Automated Guided Vehicles (AGV)
- Automatic Assembling Machinery
- Boring and Drilling Applications
- Leveling and Flattening
- Robotic applications

5 TILTIX – Modes and Parameter

The purpose of this chapter is to describe the available configuration parameters of the TILTIX inclinometers with CANopen interface. Before going into details the following information describes useful technical terms and acronyms for CANopen network communication.

Node-ID	In the CAN network each device has a Node-ID that is used to address the device in the network and to define it's priority of
EDS (Electronic Data Sheet)	An EDS file describes the communication properties of a device on the CAN network (baud rates, transmission types, I/O features, etc.). It is provided by the device manufacturer and is used in the configuration tool to configure a node (like a driver in an operating system).
PDO (Process Data Object)	CANopen frame containing I/O data. We distinguish between: Transmit-PDO (TPDO): data provided by a node Receive-PDO (RPDO): data to be consumed by a node The transmission direction is always seen from a node's point of view.
SDO (Service Data Object)	SDOs are typically used to read or write parameters while the application is running.
COB-ID (Communication Object Identifier)	Each CANopen frame starts with a COB-ID working as the Identifier. During the configuration phase, each node receives the COB-ID(s) for which it is the provider (or consumer).
NMT (Network Management Transition)	The NMT protocols are used to issue state machine change commands (i.e. to start and stop the devices), detect remote device boot ups and error conditions.

5.1 Pre-Operational Mode

If the device is in this state, its configuration can be modified. However, only SDOs can be used to read or write device-related data. The device goes into Pre-Operational Mode:

- After the power up
- On receiving the Enter Pre-Operational NMT indication, if it was in Operational Mode before

If configuration is complete, the device goes into one of the following states on receiving the corresponding indication:

- "Stopped" on receiving the "Stop Remote Node" NMT indication
- "Operational" on receiving the "Start Remote Node" NMT indication

To set a node to pre-operational mode, the master must send the following message:

Identifier	Byte 0	Byte 1	Description
0 h	80 h	00	NMT-PreOp, All Nodes
0 h	80 h	NN	NMT-PreOp, NN

NN: Node Number

5.2 Operational Mode

The device goes into the "Operational" state if it was in the "Pre-Operational" state on receiving the "Start Remote Node" indication. If the CANopen network is started using the "Node Start" NMT services in "Operational" state, all device functionalities can be used. Communication can use PDOs or SDOs.

Note: Modifications to the configuration in "Operational" mode may have unexpected consequences and should therefore only be made in "Pre-Operational" mode.

To put one or all nodes in “Operational” mode, the master has to send the following message:

Identifier	Byte 0	Byte 1	Description
0h	01h	00h	NMT-Start, all nodes
0h	01h	NN (in hex)	NMT-Start, NN

NN: Node Number

5.3 Stopped Mode

The device goes into the "Stopped" mode on receiving the "Node Stop" indication (NMT service) if it was in "Pre-Operational" or "Operational" mode. In this mode, the device cannot be configured. No service is available to read and write device-related data (SDO). Only the slave monitoring function "Node Guarding" remains active. To put one or all nodes in the ”Stopped” mode, the master has to send the following message:

Identifier	Byte 0	Byte 1	Description
0 h	02 h	00h	NMT-Stop, all nodes
0 h	02 h	NN (in hex)	NMT-Stop, NN

NN: Node Number

5.4 Re-Initialization of the Inclinometer

If a node is not operating correctly, it is advisable to carry out a re-initialization. After re-initialization, the inclinometer accesses the bus in pre-operational mode.

Identifier	Byte 0	Byte 1	Description
0 h	82 h	00h	Reset Communication
0 h	81 h	NN (in hex)	Reset Node

Polled Mode	By a Remote Transmission Request Telegram, the connected host calls for the current process value. The inclinometer uses the current position value, calculates eventually set-parameters and sends back the obtained process value by the same identifier.
Cyclic Mode	The inclinometer transmits cyclically the current process value, without being called by the host. The cycle time can be programmed in milliseconds for values between 1 ms and 65536 ms.
Sync Mode	The inclinometer answers with current process value after receiving a sync telegram. The parameter sync counter can be programmed to skip a certain number of sync telegrams before answering again.
Heartbeat Function	A node signals its communication status by cyclically transmitting a heartbeat message. This message is received by one or any number of members (Heartbeat Consumers) in the bus and so they can control the dedicated node (Heartbeat Producer).

5.5 Boot-up Procedure

The general boot-up procedure for the TILTIX CANopen and the mapping of various modes are illustrated below:

Number	Description
1	Module Power up
2	After initialization, the module automatically goes into pre-operational mode
3	NMT: Start Remote Node
4	NMT: Pre-operational Mode
5	NMT: Stop Remote Node
6	NMT: Reset Node
7	NMT: Reset Communication

5.6 Usage of Layer Settings Service (LSS)

The integrated Layer Setting Services (LSS) functionality is designed according to the CiA normative DS305V200 CANopen Layer Setting Services. These services and protocols can be used to enquire or to change settings of several parameters (of the physical, data link, and application layer) on a CANopen device with LSS slave capability by a CANopen device with LSS master capability via the CAN network. In case of TILTIX, the TILTIX inclinometer will be the LSS slave device, and the PLC (control) has to support LSS master device functionality. The LSS-functionality of the TILTIX series is limited to the following parameters of the application layer, namely node number and baud rate. The LSS master device requests services that are performed by the inclinometer (LSS slave devices). The LSS master device requests the LSS address from the LSS slave device. The LSS address is defined in object 1018h Identity Object; it consists of vendor-id, product-code, revision-number and serial-number as shown in the table below. After receiving this information, the control can identify the inclinometer, the node-number and baud rate can then be set. The exact procedure varies in detail, dependent on the utilized PLC tool. This object provides the general identification of the inclinometer.

Subindex	Description	Data Type	Default Value
0	Number of entries	Unsigned 8	0x4
1	Vendor Id	Unsigned 32	0x42
2	Product Code	Unsigned 32	
3	Revision Number	Unsigned 32	0x00010700
4	Serial Number	Unsigned 32	–

1) Prone to change with every revision.

6 Installation

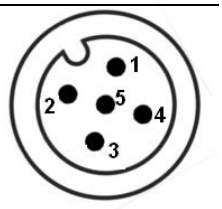
6.1 Accessories

Article No	Article	Description
ACS	Inclinometer	ACS series of Inclinometers
Download	Datasheet[1]	TILTIX ACS Data Sheet, Specifications and Drawings
Download	User Manual[1]	Installation and Configuration User Manual (English)
Download	EDS-File[1]	Electronic Datasheet (EDS) File for Configuration
10017217	Connecting cable	2m PUR Cable with M12 5pin, A-Coded, female straight connector (other lengths available)
10017219	Connecting cable	2m PUR Cable with M12 5pin, A-Coded, female angled connector (other lengths available)
10005631	Termination Res	External terminal resistors for higher baud rate transmissions

1) The latest documentation and the EDS file can also be downloaded from our website.

6.2 Pin Assignment

The inclinometer is connected via a 5-pin round M12 connector. (Standard M12, Male side at sensor, Female at connector counterpart or connection cable).

Signal	5 pin round connector	Pin Assignment
CAN Ground	1	
VS Supply Voltage	2	
0 V Supply Voltage	3	
CAN High	4	
CAN Low	5	

6.3 Installation Precautions

Warning: Do not remove or mount while the inclinometer is under power!

Avert any modifications to the housing!

Avoid mechanical load!

Prior to installation, please check for all connections and mounting instructions to be complied with. Please also observe the general rules and regulations on operating low voltage technical devices, for safety and sustainability of TILTIX Inclinometers over long period of time.

Please read the installation leaflet for detailed instructions and precautions during mounting and installation.

6.4 Mounting Instructions

TILTIX is a pre-calibrated device which can be put into immediate operation, upon simple and easy installation with a four point mount. The mounting surface must be plane and free of dust and grease. We recommend hex-head screws with M6 or UNCbolts ¼ for the best possible and secure mounting.

Use all four screws for mounting but restrict the tightening torque in the range of 1.5 – 2.5 Nm for the screws. The M12 connectors are to be perfectly aligned and screwed till the end with a tightening torque in the range of 0.4 – 0.6 Nm. Use all four screws for mounting and also note to use the same tightening torque for all the screws.

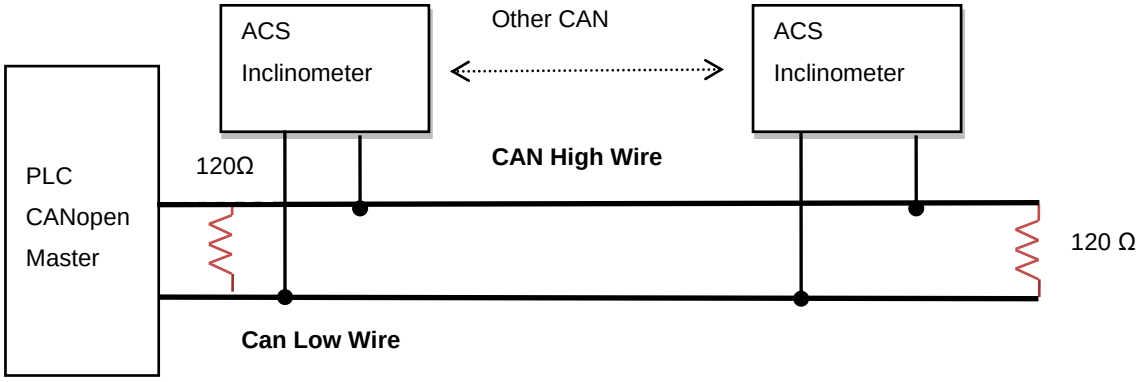
Prior to installation, please check for all connection and mounting instructions to be complied with. Please do also observe the general rules and regulations on low voltage technical devices.

6.5 Bus Termination

If the inclinometer is connected at the end or beginning of the bus or is used at transmission ≥ 50 kBaud a termination resistor of 120 Ohm must be used in order to prevent reflection of information back into the CAN bus. TILTIX sensors have built-in termination resistors that can be activated (1) or deactivated (0) if required by setting object 3002h accordingly. Please refer to the Appendix A to learn more about the objects.

The bus wires can be routed in parallel or twisted, with or without shielding in accordance with the electromagnetic compatibility requirements. A single line structure minimizes reflection.

The following diagram shows the components for the physical layer of a two-wire CAN bus:



6.6 LED Definition

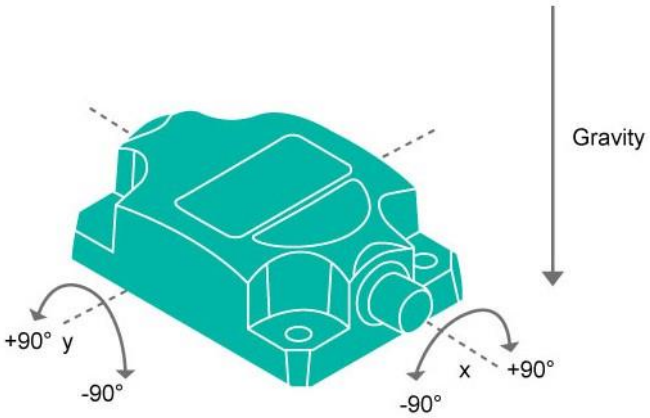
Status LED (Dual colored) **Green / Red LED**	**Meaning**
Green off	No power supply
Green single flash	Device in CAN state STOPPED.
Green blinking	Boot Up message sent, device configuration is possible Device in CAN state PRE-OPERATIONAL
Green on	Normal operation mode, Inclinometer in CAN state OPERATIONAL
Red off	Normal operation mode
Red single flash	At least one of the error counters of the CAN controller has reached or exceeded the warning level (too many error frames)
Red double flash	A guard event or heartbeat event has occurred
Red blinking	General configuration error (e.g. wrong baud rate)
Red on	The can controller is in state bus off. No communication is possible anymore. Too many error frames in the network.

Since a dual color LED is used, only the red LED is shown in cases where both green and red LED would light up.

6.7 Measurement Axes

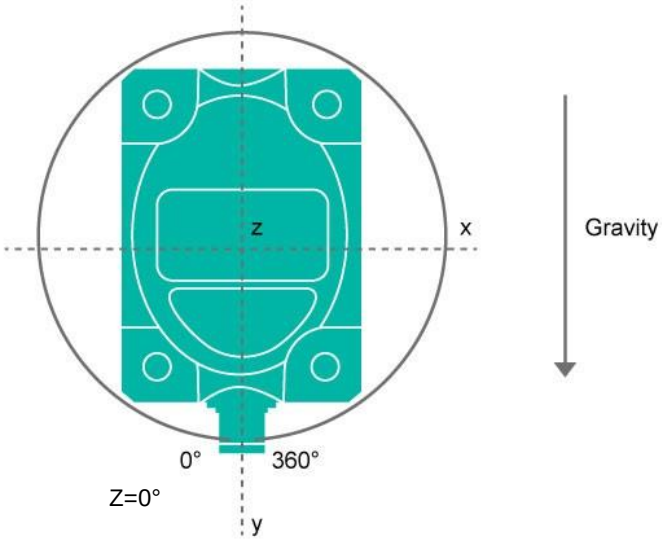
TILTIX ACS -090-2 – Dual Axis Inclinometer

X- and Y-Axis measure the angle position in space. Both axes are limited to ± 90°. The sensor is mounted horizontally. The X- and Y-Axis output 0°, when the inclinometer is flat.



TILTIX ACS -360-1 – Single Axis Inclinometer

The sensor is mounted vertically. A clockwise rotation increases the angle value from 0°-360°. When the male connector is pointing downwards, the output is 0°.



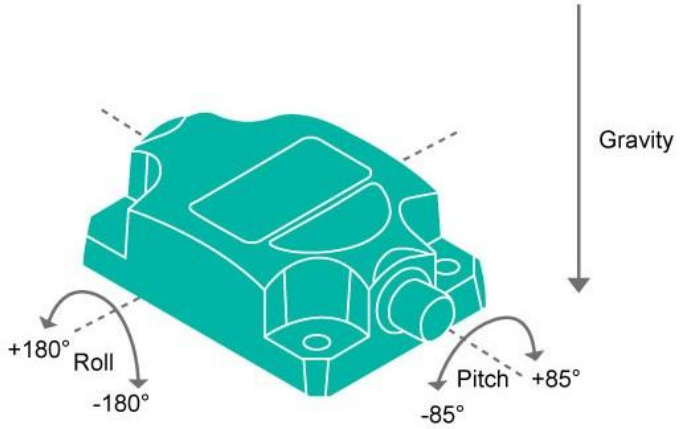
POSITAL

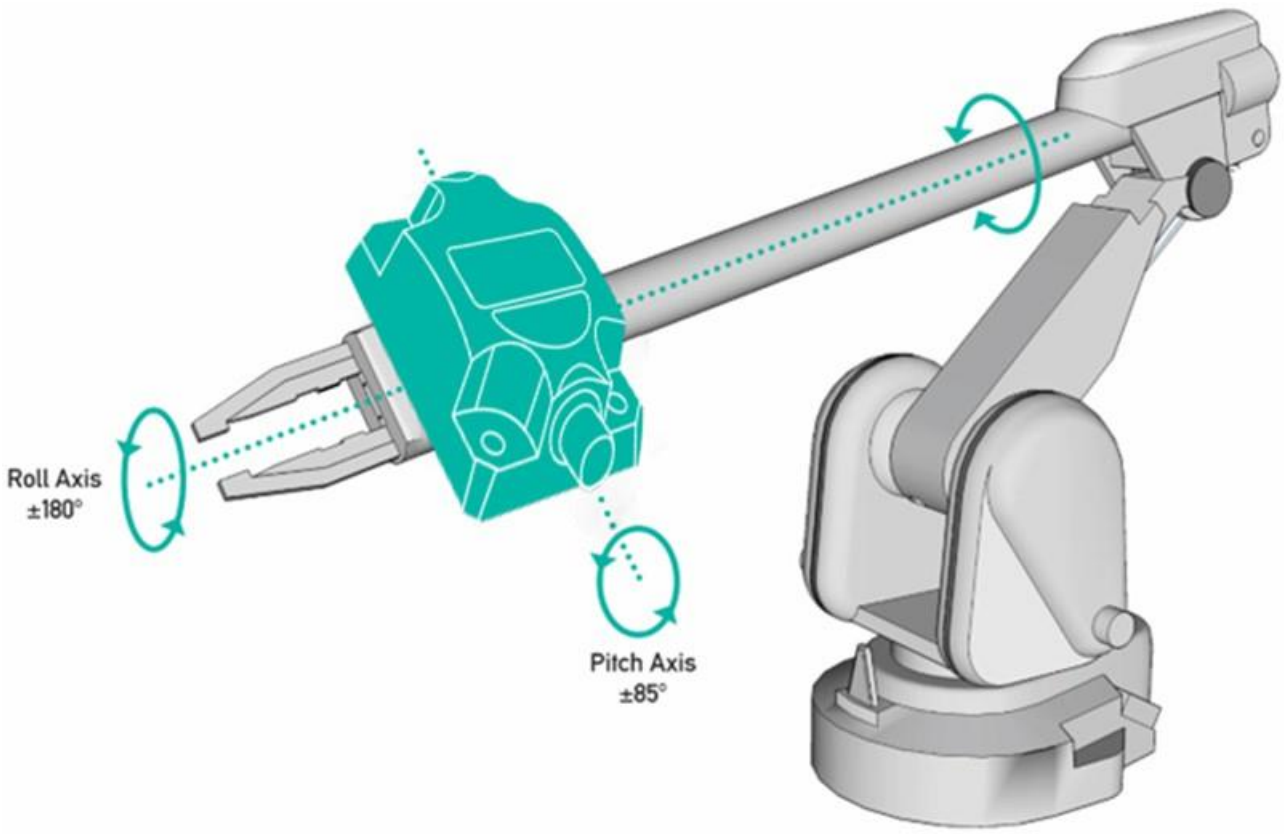
TILTIX MEMS ACS INCLINOMETER
WITH CANOPEN INTERFACE

TILTIX ACS -180-E – Pitch and Roll Axis Inclinometer

The Pitch axis monitors the position angle in space, while the Roll axis shows the rotation with respect to the pitch angle.
In case of a robot arm (graphic below), the pitch axis monitors the position angle in space and the roll axis measures the rotation of the robot head around the arm (just like a Singleturn encoder).

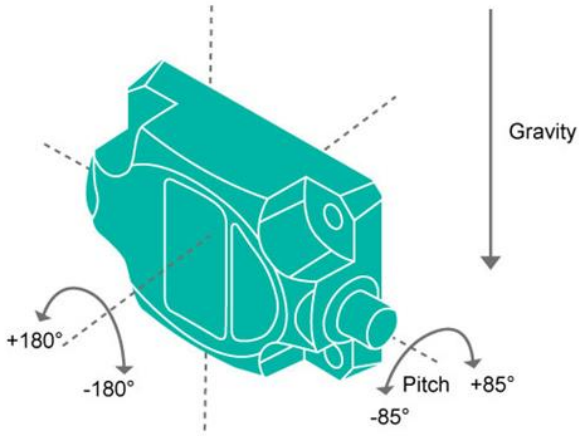
The Pitch and Roll version is very usefull for any machine where a part of the application is rotating around another segment.
The Pitch axis is limited to ± 85°, while the Roll axis is limited to ± 180°.





TILTIX ACS-180-F – Pitch and Roll Axis Inclinometer

This version is intended for vertical mounting. Like the ACS-180-E, the Pitch axis monitors the position angle in space, while the Roll axis shows the rotation with respect to the pitch angle.



7 TILTIX Software Configuration

This chapter succeeds the hardware configuration as in real time. TILTIX is a very flexible device and hence all the parameters are programmable via CAN bus. This enables remote configuration.

This chapter is divided into two major parts - one describing the methodology for putting the TILTIX into operation and the other for PDO/SDO programming of TILTIX.

7.1 Default Factory Settings

Description	Object	Value
Device Type	1000h	0x4019A – dual axis 0x3019A – single axis
Cyclic Timer	2200h	00h (0 ms)
Resolution	6000h	0Ah (0.01°)
Node Number	3000h	00h (Node number: 1)
Baud Rate	3001h	03h (125 kB)

Note: The factory settings can be restored at any time. A few of the parameters have to be re-programmed in order to make the TILTIX inclinometers compatible with the controller or the already existing CAN bus to which it is going to be installed on.

7.2 Active Programming Objects

Active CANopen objects depending on the state of TILTIX: The crosses in the table below indicate which CANopen objects are active in each state.

Initialization	Pre-Operational	Operational	Stopped	
PDO Object			x	
SDO Object		x	x	
Boot-Up	x		x	
NMT		x	x	x

7.3 Programmable Parameters

Objects are based on the CiA-DS410 CANopen profile for inclinometer (www.can-cia.org). The following table gives a list of the command identifiers sent and received by the inclinometer. These are the standard commands used for communication and transmission between a master and slave in a CAN bus. It is quite useful for the analysis of communication logs between the master and slave and for better understanding of the system under observation.

7.4 Command Description

Command	Function	Telegram	Description
22h	Domain Download	Request	Parameter to TILTIX
60h	Domain Download	Confirmation	Parameter Received
40h	Domain Upload	Request	Parameter Request
43h, 4Bh, 4Fh [1)]	Domain Upload	Reply	Parameter to Master
80h	Warning	Reply	Transmission Error

1) The value of the command byte depends on the data length of the called parameter.

7.5 Data Length of Commands

Command	Data length	Data Length
43h	4 Byte	Unsigned 32
4Bh	2 Byte	Unsigned 16
4Fh	1 Byte	Unsigned 8

7.6 Frequently Used Commands

The following list shows the most frequently used objects while programming the TILTIX inclinometer. The whole list of objects is provided in Appendix A.

Section 6 offers important programming tips and FAQ which are necessary for the proper use of the inclinometer.

Position Value (Objects 6010h, 6020h)	The objects 6010h and 6020h are used to get the scaled tilt angle (integer 16 variables) For dual axis ACS-090-2: Object 6010h refers to the X-Axis Object 6020h refers to the Y-Axis For one axis ACS-360-1: Object 6010h refers to the Z-Axis For pitch and roll axis ACS-180-E: Object 6010h refers to the Pitch-Axis Object 6020h refers to the Roll-Axis
Store Parameters (Objects 1010h, 2300h)	Object 1010h is used to store either all parameters or only parts of the object dictionary if necessary. Whereas object 2300h stores and saves all current parameters with an additional NMT reset of the TILTIX.
Resolution in Degree (°) (Object 6000h)	This parameter is used to program the desired resolution. The resolution can be set to 1º (1000d), 0.1º (100d) or 0.01º (10d). (Default 10d – 0.01º)
Preset Value (Objects 6012h/6022h)	You set the inclinometers output position value to a desired position value in your machine. The value is set in the inclinometer when the telegram is sent and confirmed. Do this operation during standstill of the application to increase the accuracy, because the device is calculating and automatically applying an offset value. If you set the preset dynamically, which is not recommended, then you also have to take bus latency time and inclinometer internal cycle time into consideration. For 2 axes: Object 6012h is used for setting the preset value for X-Axis and Object 6013h for Y-Axis. For 1 axis Object 6012h is used for Z-Axis.
Baud Rate (Object 3001h)	The Baud rate can be programmed via SDO. (Default 125 kBaud)
Node Number (Object 3000h)	The setting of the Node Number can be achieved via SDO-Object. Valid addresses range from 1 to 127, each address can only be used once. The value set in this object is incremented by one to prevent setting the Node Number 0. For inclinometers programmed via SDO, the default is 0h = Node Number 1

7.7 PDO Transmission

Process Data Objects (PDOs) communicate process information/data and enable them to be exchanged in real time. A CANopen device's PDO set describes the implicit exchanges between this device and its communication partners on the network. The exchange of PDOs is only authorized if the device is in "Operational" mode.

Note: The PDOs can be directly mapped into memory locations on the controller and can be viewed upon reading those memory locations. An example with a SCHNEIDER-TWIDO controller is provided in the next section.

7.8 Object 1800h: 1st Transmit PDO Communication Parameter

This object contains the communication parameter of the 1st transmit PDO.

Subindex1	Description	Data Type	Default Value	Access	Restore after BootUp
00h	Number of sub indices	Unsigned 8	5	ro	yes
01h	COB-ID	Unsigned 32	180h + Node ID	rw	yes
02h	Transmission Mode	Unsigned 8	1	rw	yes
03h	Inhibit Time	Unsigned 32	0	rw	yes
04h	Not Available				
05h	Event Timer	Unsigned 32	0	rw	yes

1) Subindex: Second degree identifier used in combination with the object. (Follows the object number)

Programming example (for NN=1): The inclinometer should send the PDO every 100 ms (default setting is that the sensor answers to synch messages from the CANopen master)

Step 1: Set all sensors to pre-operational mode

Identifier	DLC	Byte 0	Byte 1
00	2	80	00

Step 2: Set Event timer 1800Sub05 to 100ms (100dec = 64h)

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	00	18	05	64	00	00	00

Step 3: Set Transmission Type (1800Sub2) to "Send PDO on event" (256dec = FEh)

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	00	18	02	FE	00	00	00

Step 4: Set all sensors to operational mode

Identifier	DLC	Byte 0	Byte 1
00	2	01	00

7.9 Object 1801h: 2nd Transmit PDO Communication Parameter

This object contains the communication parameter of the 2nd transmit PDO.

Subindex1	Description	Data Type	Default Value	Access	Restore after BootUp
00h	Number of sub indices	Unsigned 8	5	ro	yes
01h	COB-ID	Unsigned 32	280h + Node ID	rw	yes
02h	Transmission Mode	Unsigned 8	254	rw	yes
03h	Inhibit Time	Unsigned 32	0	rw	yes
04h	Not Available				
05h	Event Timer	Unsigned 32	0	rw	yes

1) Subindex: Second degree identifier used in combination with the object. (Follows the object number)

7.10 Transmission Mode

The transmission mode (Sub index 2) for Objects 1800 and 1801 can be configured as described below:

Transfer Value (Dec)	Transmission Mode					Note
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	RTR Only	
0		X	X			Send PDO on first sync message following an event
1 – 240	X		X			Send PDO every x sync messages
241 – 251	Reserved					
252			X		X	Receive Sync and send PDO on remote request
253					X	Update data and send PDO on remote request
254				X		Send PDO on event
255				X		Send PDO on Event

7.11 Inhibit Time

The inhibit time is the minimum interval for PDO transmission if the transmission type is set to 254d (FEh) and 255d (FFh). The value is defined as multiple of 100 µs. The Inhibit Time cannot be changed while the respective PDO is in operation.

7.12 Event Timer

The Event Timer only works in asynchronous transmission mode (transmission mode 254d and 255d). The value is defined as multiple of 1 ms. A transmit PDO is sent after the Event Timer expires (for values > 0). The range is 1 – 65536 ms. The Event Timers of TPDO1 and TPDO2 are hardwired, i.e. if one timer is changed, the other one is adjusted accordingly.

7.13 Cyclic Timer

The Cyclic Timer is hardwired with both Event Timers of TPDO1 and TPDO2. Due to the fact that it is possible to save either communication parameters (Event Timers) or manufacturer parameter (Cyclic Time), the parameters could hold different values after a power off/on cycle. The inclinometer prevents this mismatch by checking these values during startup. The event timer value is copied to the Cyclic Time if they do not match.

7.14 Object 1A00h: 1st Transmit PDO Mapping Parameter

This object contains the mapping parameter of the 1st transmit PDO.

Subindex	Description	Data Type	Default Value	Access	Restore after BootUp
0	Number of sub indices	Unsigned 8	2 – dual axis 1 – single axis	rw	yes
1	Mapped object	Unsigned 16	0x60100010	rw	yes
2	Mapped object	Unsigned 16	0x60200010	rw	yes

7.15 Object 1A01h: 2nd Transmit PDO Mapping Parameter

This object contains the mapping parameter of the 2nd transmit PDO.

Subindex	Description	Data Type	Default Value	Access	Restore after BootUp
0	Number of sub indices	Unsigned 8	2 – dual axis 1 – single axis	rw	yes
1	Mapped object	Unsigned 16	0x60100010	rw	yes
2	Mapped object	Unsigned 16	0x60200010	rw	yes

Programming example (for NN=1): The Sensor should transmit the following values in the 2nd TPDO

TPDO2			
Acceleration x	Acceleration y		

Step 1: Disable 2nd Transmit PDO

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	18	01	00	00	00	80

Step 2: Disable TPDO2 mapping

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	1A	00	00	00	00	00

Step 3: Map acceleration x 16-bit value (3403Sub0) to TPDO2 Object 1

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	1A	01	10	00	03	34

Step 4: Map acceleration y 16-bit value (3404Sub0) to TPDO2 Object 2

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	1A	02	10	00	04	34

Step 5: Map acceleration z 16-bit value (3405Sub0) to TPDO2 Object 3

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	1A	03	10	00	05	34

Step 6: Enable TPDO mapping Parameter again

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	1A	00	03	00	00	00

Step 7: Enable 2nd Transmit PDO

Byte 4 to Byte 5 is the COB-ID -> 280h + Node-ID = 281h. For TPDO 1 this would be 180h + Node-ID

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	18	01	81	02	00	00

Step 8: Save this configuration by writing “save” to 1010Sub01

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	10	10	01	73	61	76	65

7.16 Types of SDO

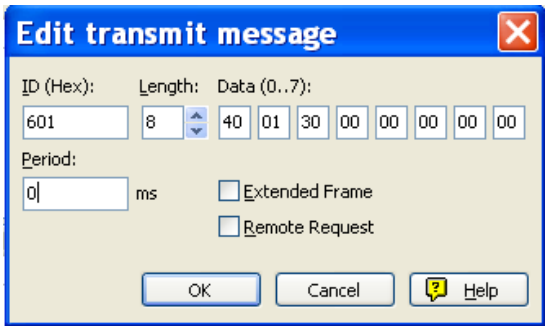
Service Data Objects (SDOs) allow a device's data to be accessed by using explicit requests. The SDO service is available if the device is in the "Operational" or "Pre-Operational" state. There are two types of SDOs:

- Read SDOs (Download SDOs)
- Write SDOs (Upload SDOs).

The SDO protocol is based on a 'Client/Server' model:

Download SDO	The client sends a request indicating the object to be read. The server returns the data contained within the object.
Upload SDO	The client sends a request indicating the object to be written to with the desired value. After the object has been updated, the server returns a confirmation message.
Unprocessed SDO	The server returns an error code (80) in both cases if an SDO could not be processed.

A typical illustration of an SDO for explicitly reading the current baud rate value is given below:

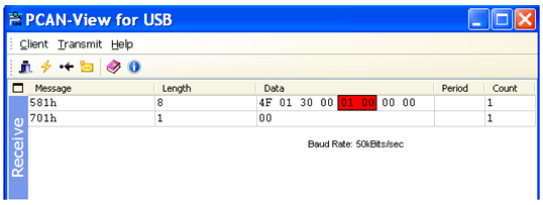


SDO passed as a new message to the device

We used a PEAK™ CAN master for this illustration. The PCAN®-USB adapter enables simple connection to CAN networks.

The PCAN®-USB's compact plastic casing makes it suitable for mobile applications. It works as a master on the CAN bus connection via D-Sub, 9-pin and is in accordance with CiA102 standards. (To learn more about Peak CAN click here)

- ID: 601- Message to NN1 (600+1)
- Length: 8bit word
- Data 0: Read (40) / Write (22)
- Object 3001h allows to read the baud rate value from the inclinometer.
- Data 1 & 2 : Object in Little Endian (3001s is 0130 in Little Endian format)
- Data 3: Sub-Index (NA)
- Data 4-7: Data to be written (NA in read command)
- The Received message 581h
- reads out the data



Received Message from the Device

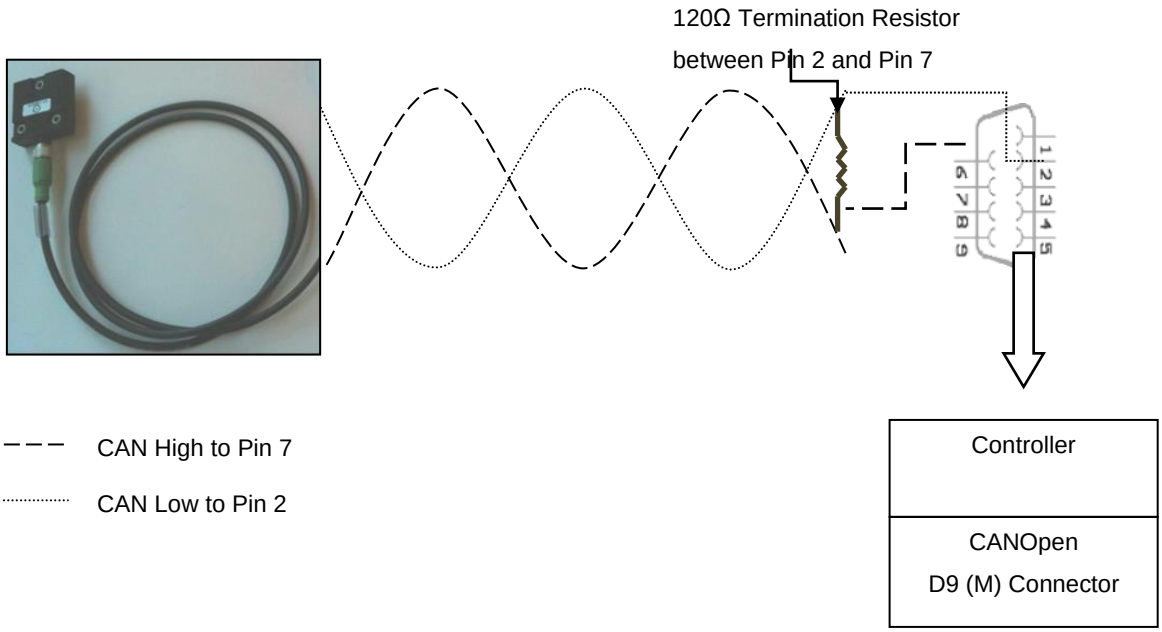
So, SDOs can be used to explicitly read or write data in TILTIX CANopen inclinometers. All the relevant objects that can be configured are described in Appendix A.

In the above example, 701h is the boot up message received. Then once we transmit the SDO command as shown above, we receive a reply. The received message 581h, consists of the domain downloaded. In this case it is the baud rate (01h – 50 kB) as indicated in the above figure.

8 Working with Schneider PLC

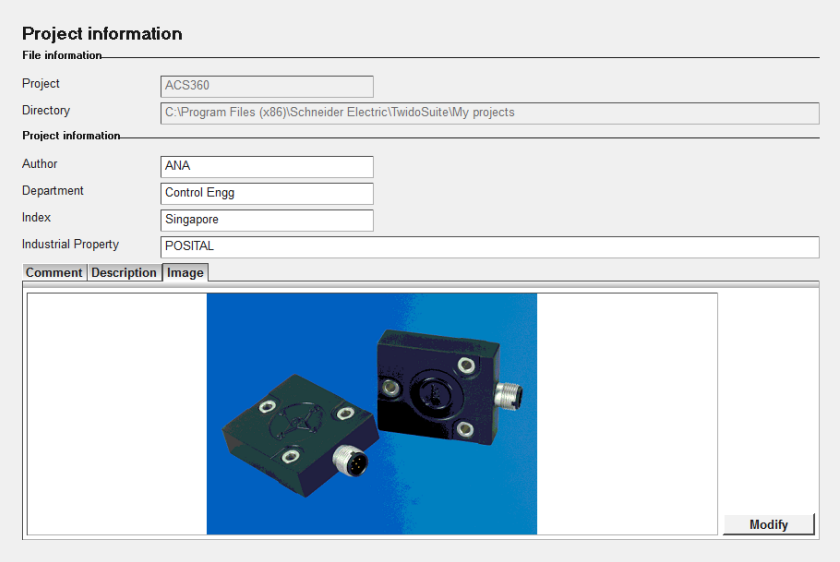
8.1 Types of SDO

A single axis inclinometer was connected to a TWIDO programmable logic controller with a CANopen communication interface. The step-by-step connection procedure and the working of the inclinometer in a CAN bus is illustrated in the following sections. Please note that, the programming in other control systems may vary individually. Please have this section as a reference for TILTIX' working with programmable logic controllers.



8.2 Software Project Information

Once the hardware setup is done the TILTIX should be configured in such a way that it is compatible to the already existing setup and gives a proper position output.

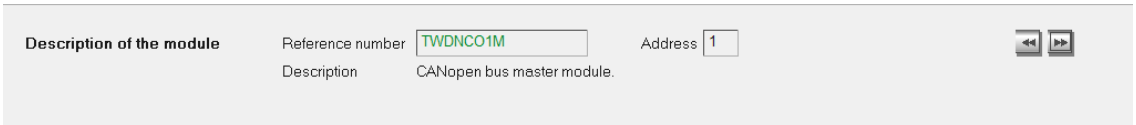


TWDLMDA20DTK

Modular base controller, 12 24V DC inputs, 8 ouputs (0.3A source transistors). Removable MIL connectors.

8.3 Controller Description

8.4 CANopen Master Configuration



TWDNCO1M

CANopen bus master module.

FRABA (V1.1)

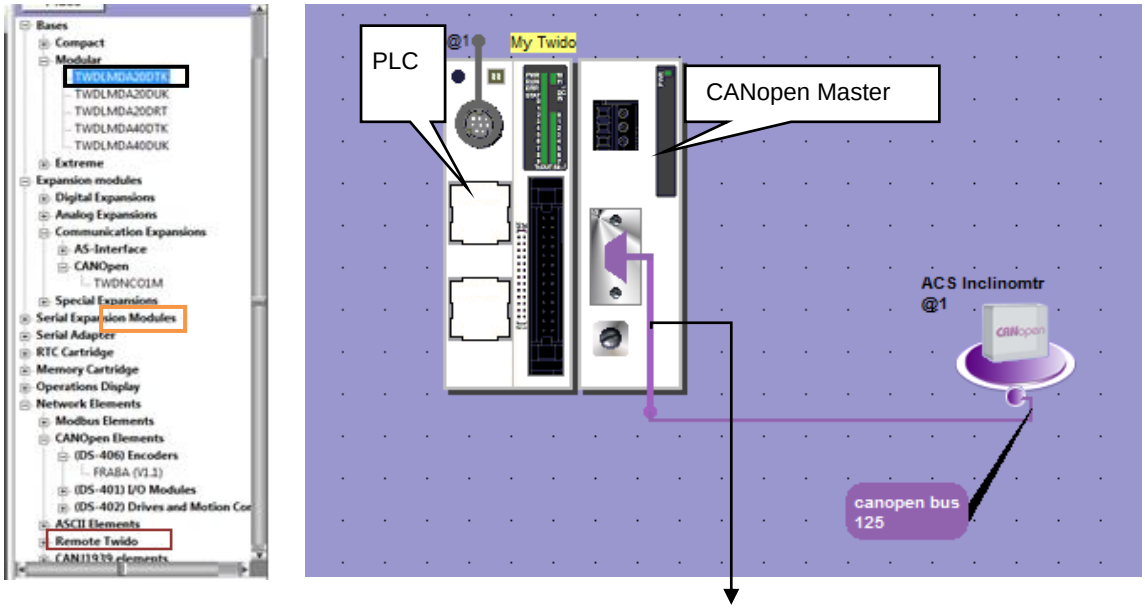
Vendor:FRABA Posital GmbH
Description:EDS for ACS inclinometer (option 360°)
Author:FRABA Posital GmbH, Mr. Jakschies
Creation:06-29-2004

Once the TILTIX EDS file is being uploaded all objects – including the PDOs – are added to the controller. The Schneider system automatically

identifies the PDOs and maps them on to the slave device.

8.5 Connection Network Setup

The illustration below, describes the connection of the elements in the CAN bus. At first, the CANopen communication interface is connected to the main controller. Then the inclinometer is connected to the CANopen communication interface.



Linking of Master and Slave

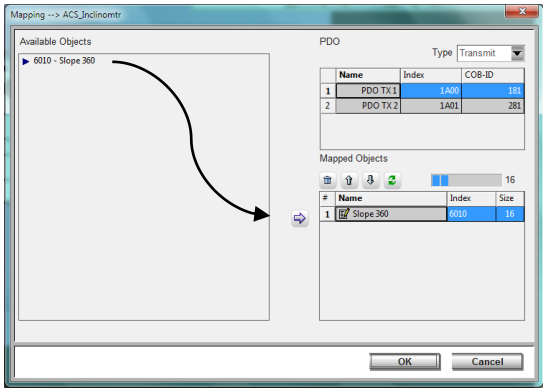
The next step after the setup of the network is the configuration of all the parameters and settings, to facilitate the communication between the master, slave and the controller. This picture is the overall description of the setup, with the TWIDO TWDLMDA20DTK controller, TWDNCO1M CANopen communication expansion module and the TILTIX360 EDS file.

9 Configuration

9.1 TILTIX Process Data Objects (PDO) Mapping

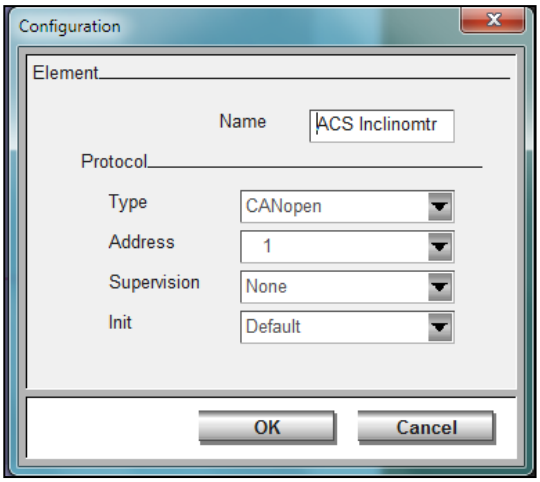
A list of all available objects is pre-programmed in the EDS file. Select the TILTIX inclinometer on the bus and click on Configuration. A list of all the mapped PDOs appears.

Then, according to need, the objects are mapped on to the Transmit-PDO's of the TILTIX.



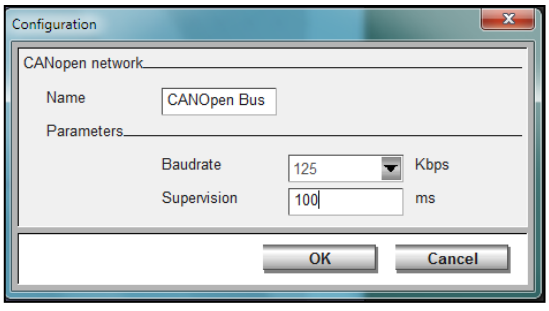
9.2 TILTIX CANopen Node Configuration

Click on the TILTIX inclinometer on the bus and select the CANopen configuration option. It is used to define the name, type, address and supervision of the node. Make sure the node number and the address coincide for the inclinometer selected.



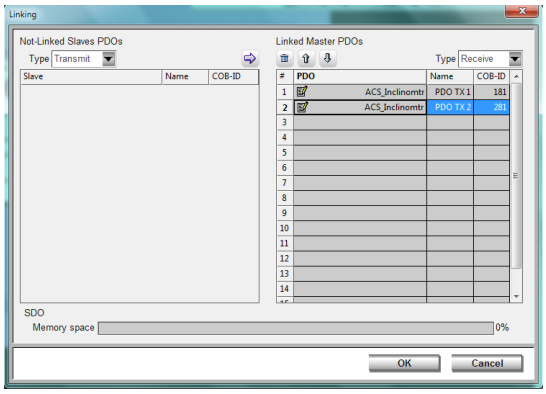
9.3 CANopen bus network configuration

Click on the bus connecting the TILTIX inclinometer and the PLC. Select the bus configuration option to define the name of the bus, the transmission speed and supervision time. Make sure that the TILTIX is configured with the appropriate baud. (All devices on the bus must have the same baud rate)



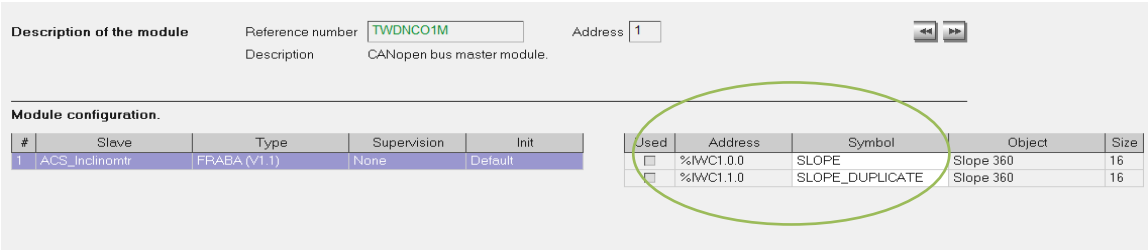
9.4 Linking of CANopen Master and TILTIX Transmit-PDOs

Select the CANopen link on the controller. Click on the configuration option. The PDOs of the slave are mapped on to the CANopen master so that the information contained in the objects of the slave are transmitted and saved on to the controller's memory



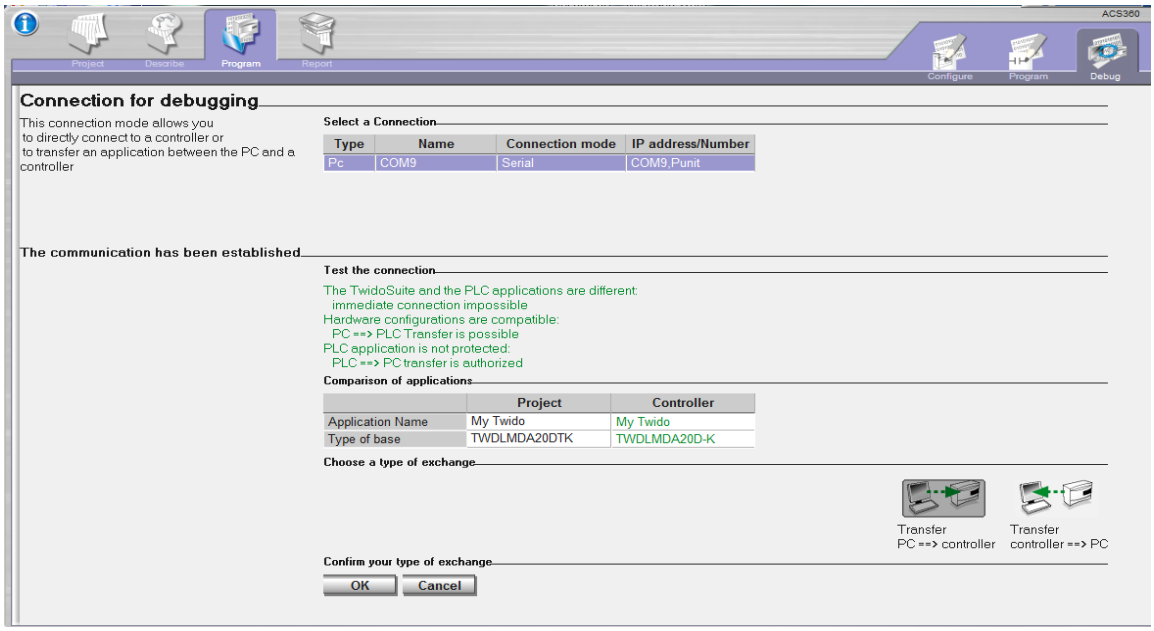
9.5 TILTIX & Controller Memory Configuration

The current and updated position values from the TILTIX inclinometer are mapped on to an EEPROM memory location in the controller. This memory location, in this case %IWC1.0.0 and %IWC1.1.0 will always contain the slope values obtained from the object 6010h of the TILTIX when the controller is online.

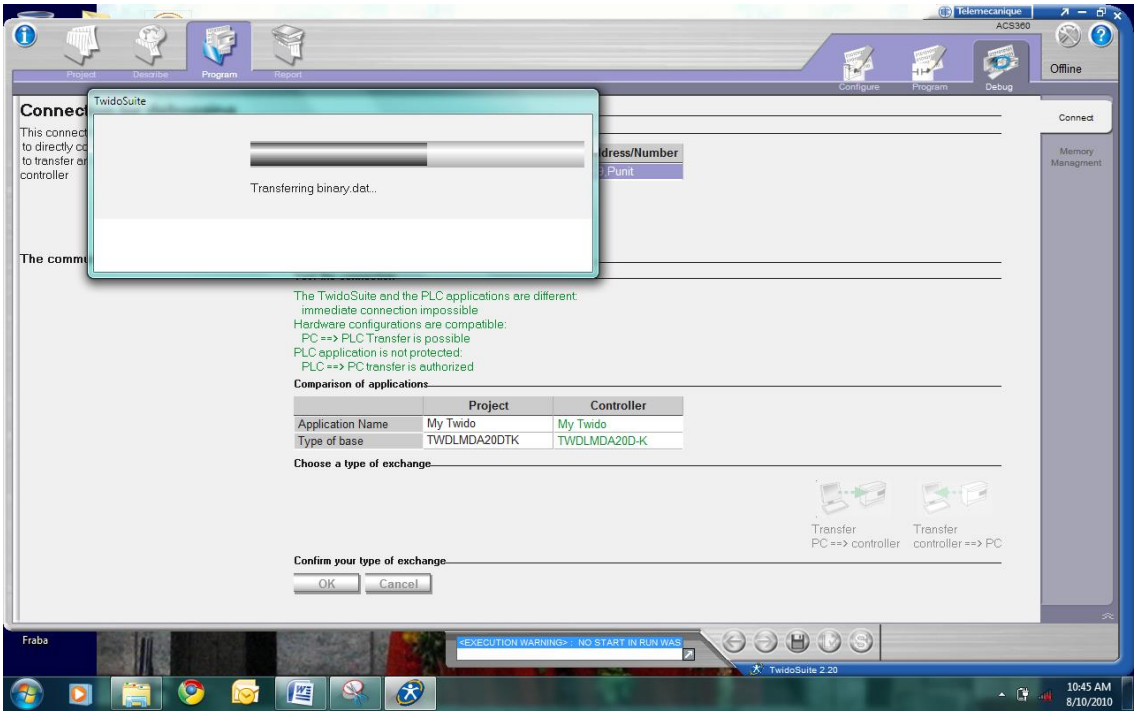


9.6 Debugging

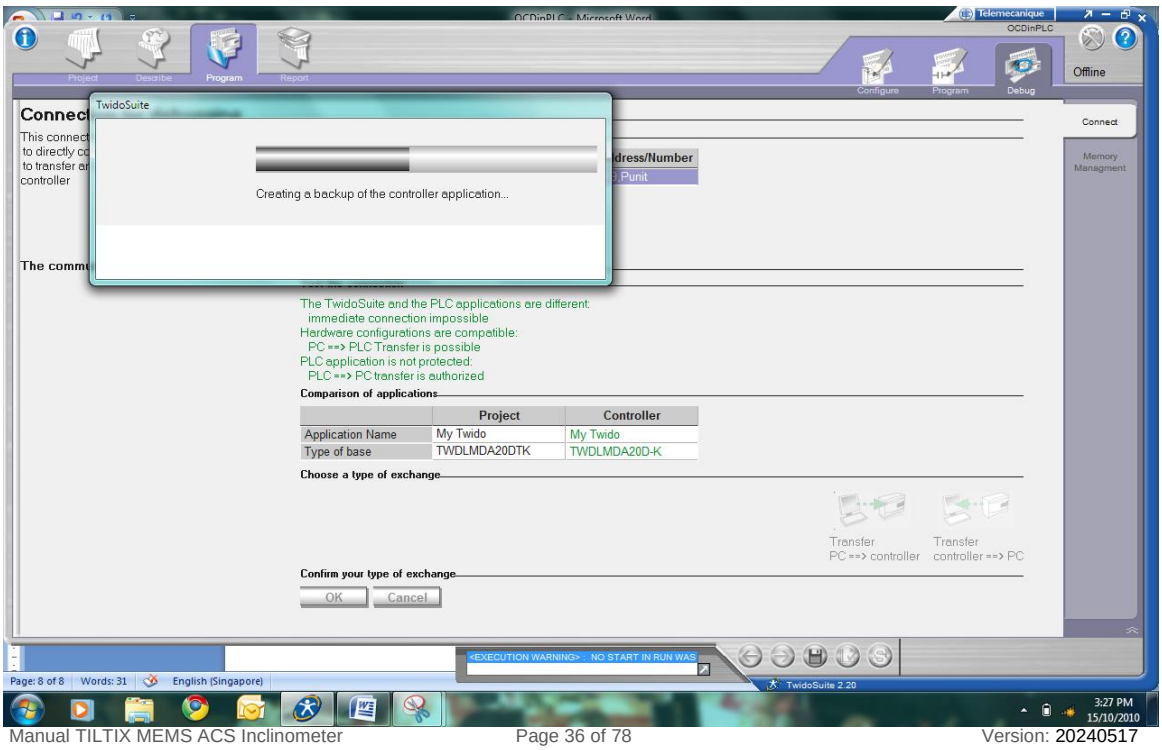
The debugging stage is done on completing the configuration of the PDO's. It involves the following steps:



The serial communication port is selected and PC → controller transfer is initiated. Once the transfer is initiated the configured parameters and the programming done on the PC is debugged and transferred to the controller for real time application. The following illustrations are the intermediate tasks during debugging.



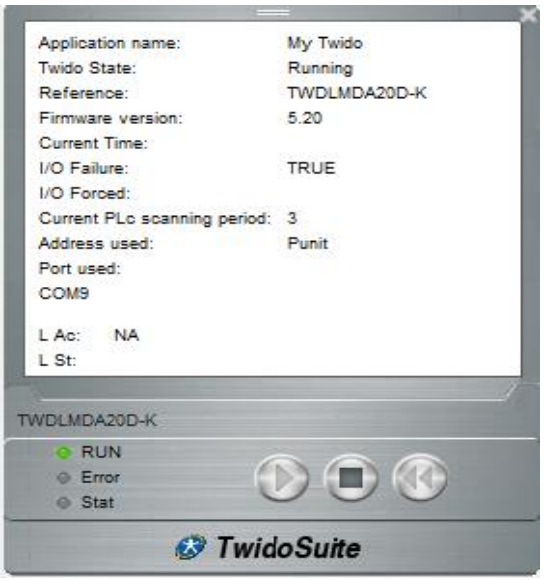
Converting all the configured parameters to binary formats.



Once the controller goes into the online mode, the PDOs cannot be changed. However, SDOs still can be configured if necessary.

9.7 Run

Once debugged, the controller goes into online mode. SDOs then can be configured if needed before running the controller.



Upon start up, we can create an animation table to monitor the necessary controller parameters and the system variables which contain the position value. Now, we will program the PLC in order to obtain the position values.

9.8 Resetting CANopen Communication

```
0  *                                    LD    1
1  1/*                                  [ %MW0 := 16#0001 ]
2  0/*                                  [ %MW1 := 16#0000 ]
3  1                                    LD    %SW81:X3
4  * /*                                 [ CAN_CMD1 %MW0:2 ]
5  0                                    LDN   %SW81:X4
```

9.9 Resetting CANopen Nodes

```
0  *                                    LD    1
1  1/*                                  [ %MW0 := 16#0001 ]
2  1/*                                  [ %MW1 := 16#0001 ]
3  1                                    LD    %SW81:X3
4  * /*                                 [ CAN_CMD1 %MW0:2 ]
5  0                                    LDN   %SW81:X4
```

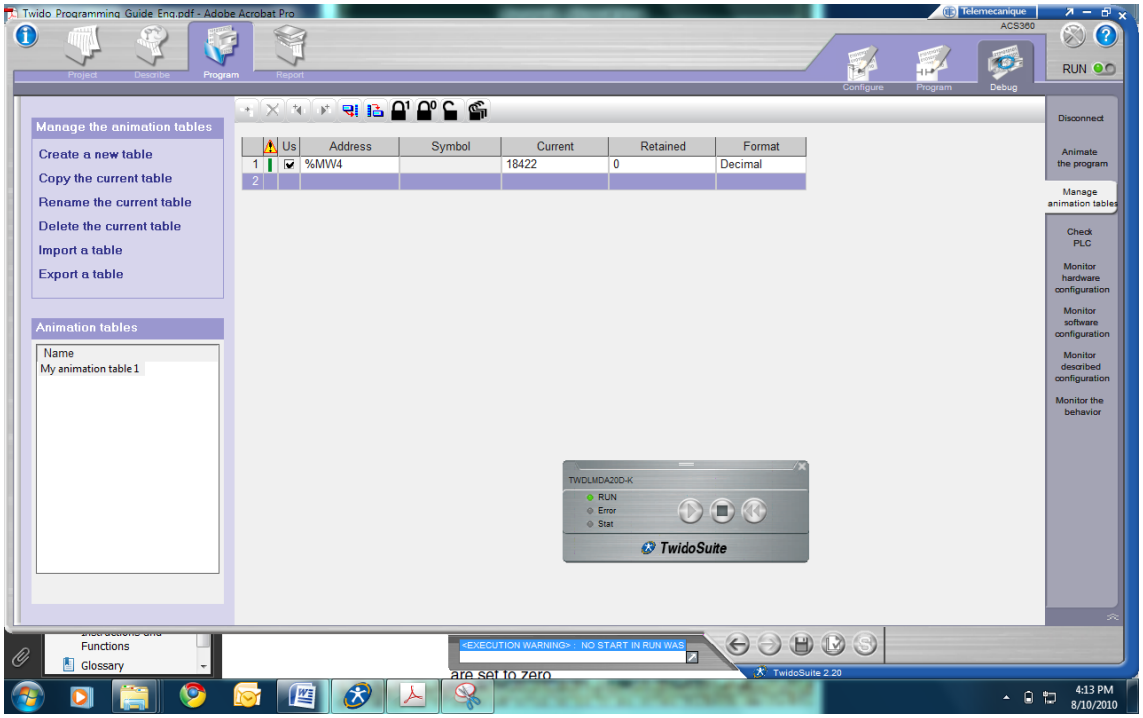
```
0  *                                    LD    1
1  2/*                                  [ %MW0 := 16#0002 ]
2  1/*                                  [ %MW1 := 16#0001 ]
3  1                                    LD    %SW81:X3
4  * /*                                 [ CAN_CMD1 %MW0:2 ]
5  0                                    LDN   %SW81:X4
```

9.10 Position Readout

```
0  *                                        LD    1
1  3/*                                      [ %MW0 := 16#0003 ]
2  1/*                                      [ %MW1 := 16#0001 ]
3  24592/*                                  [ %MW2 := 16#6010 ]
4  0/*                                      [ %MW3 := 16#0000 ]
5  18416/*                                  [ %MW4 := 16#0000 ]
6  0/*                                      [ %MW5 := 16#0000 ]
7  * /*                                     [ CAN_CMD1 %MW0:6 ]
```

Position Value

9.11 Readout Using Animation Table



The position Readout is 18416 through the memory location %MW4 (Shown in the programming). We know that the resolution is set to 0.01°. TILTIX Position Value = 18416 x 0.01° = 184.16°

Illustration of measurement over full range: At initial position (approximately 0°):

		Us	Address	Symbol	Current	Retained	Format
1		☑	%MW4		32	0	Decimal
2							

TILTIX Position Value = 32 * 0.01° = 0.32°

At approximately 90°:

		Us	Address	Symbol	Current	Retained	Format
1		☑	%MW4		9138	0	Decimal
2							

TILTIX Position Value = 9138 * 0.01° = 91.38°

At approximately 180°:

		Us	Address	Symbol	Current	Retained	Format
1		☑	%MW4		18052	0	Decimal
2							

TILTIX Position Value = 18052 * 0.01° = 180.52°

At approximately 270°:

		Us	Address	Symbol	Current	Retained	Format
1		☑	%MW4		27256	0	Decimal
2							

TILTIX Position Value = 27256 * 0.01° =272.56°

All the above position values where obtained by programming the position value output explicitly. The other method is very simple and direct.

Just run the controller and the same position values are obtained. The position is mapped with the memory location %IWC1.0.0 or %IWC1.1.0 through PDO mapping done in the earlier steps. The steps for the mapping have been illustrated in above parts so that, in real time application, end users can directly follow the above steps to read out the position values from the mapped memory locations.

10 Frequently Asked Question

What if the inclinometer doesn't respond after power on?

If the bus is active, then, the installed inclinometer is transmitting a false node number. If the bus is inactive, then, it was connected with an incorrect baud rate.

Possible solution may include addressing the node in pre-operational mode, addressing the inclinometer via SDO, trying to reset or power off or reprogramming the baud rate.

How to avoid malfunction of the position value during transmission?

During the transmission of the position value occasional malfunctions can occur. The CAN bus can be temporarily in the bus off state, too.

In order to avoid this problem, please check if the last bus nodes have the terminal resistor. The termination resistors at the ends of the CAN bus are generally used to avoid the reflection of information back into the bus. If the last bus node is a TILTIX Inclinometer the terminal resistor is to be activated through object 3002h.

When can too many ERROR-Frames appear?

The bus load maybe too high in case of too many error frames. So in this case check if all bus nodes have the same baud rate. Even if one node has a different baud rate, error frames are produced automatically.

Always ensure that the baud rate is set according to the defined TILTIX Baud rate table and once the baud rate has been changed, the CAN bus has to be restarted with the communication changed to the recently set baud rate.

What does unexpected module / Module missing / Wrong Module errors refer to?

Improper definition of node address or improper loading of EDS file generally leads to such errors. It is best to reinitialize the CAN bus or re-install the EDS file.

The node state stopped upon loading and initialization!

It occurs mostly because the bus transmission timeout is defined shorter than the TILTIX transmission time. So it is best to increase the bus timeout period (Approximately 2 – 3 seconds).

Unable to change to another node number?

If all nodes are found to be in operational mode, then follow the next few steps to set the required node number to a selected device.

Calculate the required node number in hexadecimal. (TILTIX is internally programmed to add 1 to any node number change fed to it, in order to avoid the node number 0; Range of Node Numbers: 1 – 127). For example if we want a NN=28 decimal, we need to feed 27 decimal (27+1=28). So the NN 1B hex has to be fed in order to set the selected device to node number 28.

Send a write telegram to the particular node, with 1B as data on the object 3000h.

Use 2300h to save the parameters with the reset.

A boot up message with the new node number pops us.

What is the significance of boot-up message?

Rx	0701	1	00

The boot up message signifies that the node is active and can communicate. The node number can be identified from the boot-up message.
NN = 701h – 700h = 1h = 1

What is the difference between operational and pre-operational mode?

The main difference between the modes is that in pre-operational mode only service data objects (SDOs) can be programmed whereas in operational mode all device functionalities SDOs, PDOs can be programmed.

Modification of parameters in the operational mode may sometimes have unexpected consequences and should therefore be avoided.

How to read position value?

Send a message to the TILTIX to read out the position data object (6010h/6020h). The current position is calculated and is given out as a reply to the above message.

For example, the position data is given out as 0EF8 hex, this is 3832 in decimal. If the resolution is 0Ahex (0.01°) then the position value is (position output in decimal X Resolution), 3832 X 0.01° = 38.32º.

Why are there different types of storing and re-storing?

Object 2300h can be used to save all the parameters in the non-volatile memory at once. But when we use object 1010h for storing and object 1011h for re-storing there are a lot of options available – Communication, Application, Manufacturer or all parameters. It enables the user to store specific types of objects without disturbing the other configurations. Hence, this gives the user ease and flexibility for storing only particular objects.

How to send a sync message?

The default COB-ID sync message value is 80h. Once a TPDO transmission type is set to sync, the TILTIX sensor will react to the following message:

- COB-ID 80h (Default)
- Data Bits 0

How to set the cyclic timer?

The cyclic timer can be set with the object 2200h. Write the required cycle time in ms (0 - 65536) to the object and save the parameters.

The cyclic timer is hardwired to the TPDO event timers to avoid conflicts. The TPDO timers are given higher priority. Hence, upon boot up, if the cyclic timer and event timers have different values, the TILTIX immediately writes the value of the event timers to the cyclic timer to make it uniform.

How to configure PDO Transmission?

The position value can be transmitted in various modes using PDO objects. The modes can be configured according to the user's requirement.

- Sync Mode
  This mode allows the user to receive the position value only if requested. Change the transmission type of the corresponding TPDO communication object (1800h/1801h) to sync mode by writing 1HEX to sub-index 2.
  Once back in operational mode, the user needs to transmit a sync message when a position value is required. The TILTIX replies with current position value through the TPDO COB-ID (Default: 0x181 for TPDO 1 & 0x281 for TPDO 2).
- Cyclic Mode
  This mode allows the user to receive the

position values at a constant interval.
Change the transmission type of the corresponding TPDO communication object (1800h/1801h) to cyclic mode by writing 254
(FE Hex) to sub-index 2. Make sure that the cyclic timer is set to the value of the required interval.
Once in operational mode, the TILTIX shall start transmitting the position output at a constant interval, with the TPDO COB-ID (Default: 0x181 for TPDO 1 & 0x281 for TPDO 2).

- State Change Mode
  This mode enables the user to receive position data only when there is a change in its value.
  Change the transmission type of the corresponding TPDO communication object (1800h/1801h) to state change mode by writing 255 (FF Hex) to sub-index 2. Make sure that the inhibit timer is also set as required.
  Once in operational mode the TILTIX outputs the position only if there is a change in its value.

How to calculate the inhibit time?

The inhibit time sets the minimum time between two process values. The value is defined as multiples of 100µs. For example, assume we set the inhibit timer (Object 1800h/1801h) sub-index 3 to 1000d (03E8hex). The inhibit time is set to 100ms (1000 x 100µsIt is not allowed to change the inhibit time while the PDO exists.

Unable to set preset?

While setting preset there are a few considerations like resolution and range that have to be taken into consideration. The resolution of the TILTIX set by the user defines the range of the values that can be set. Hence, while setting the preset the user has to be careful to be within range.

What is a moving average filter?

The moving average filter is a kind of simple low pass filter. In TILTIX the moving average is used to dampen the effects due to external influences like vibrations. 0 means that the filter is deactivated. The user can define the array size in the range of 1-250 (decimal). Once the array size is defined the TILTIX takes an average of the position data array for the final position output.

What are the considerations while setting the resolution of TILTIX?

The resolution can be set with the object 6000h. If attribute “resolution” is changed, side effects will occur while the sensor is adapting the set parameters to the configured resolution. This is because many other objects/parameters are set based on the resolution. Rounding effects have to be taken into account. Attributes affected:

Write/Read: Preset Slope Long16, Preset Slope Lateral16, Differential Slope Long16 Offset, Differential Slope Lateral16 Offset, Preset Slope Long32, Preset Slope Lateral32, Differential Slope Long32 Offset, Differential Slope Lateral32 Offset;

Read Only: Slope Long16, Slope Lateral16, Slope Long16 Offset, Slope Lateral16 Offset, Slope Long32, Slope Lateral32, Slope Lateral32 Offset, Slope Lateral32 Offset.

What is scaling and inversion? Explain with example?

Scaling and inversion are functionalities which are included in TILTIX to make the user handle and process the position output according to his need.

This is generally controlled by the operation parameter objects (6011h / 6021h).

For example: Assume that the slope offset value is A8 and differential offset value is 0.

Scaling	Inversion	Position Value (in hex)
0	0	1043
0	1	EFBD
1	0	10EB

1	1	F065

- If 00, then position value is the actual position 1043 hex.
- If 01, then the position value is just inverted (-1043). This is (10000h – 1043) = EFBD hex.
- If 10, then the position is just scaled. (1043 + A8) is 10EB hex.
- If 11, then the position is inverted and scaled. Therefore, the new position value is (EFBD + A8) F065 hex.

If saving or restoring not all but only part of the parameters (communication, application or manufacturer parameters), Cyclic Time (object 2200) and Preset X, Preset Y (80° version, objects 2600 and 2601) or Preset (360° version, object 2600) show a special behavior: Cyclic Time is hardwired with the Event Timers of TxPDO1 and TxPDO2. If only the Event Timers (communication parameters) or Cyclic Time (Manufacturer parameter) is/are saved, the parameters would hold different values after a power off/on cycle. Therefore, if the values do not match, the value of the event timers is copied to the Cycle Time, too. Preset parameters work the same way, with the application parameters (objects 6xxx) having the higher priority.

Slope long16, slope lateral16, slope long32 and slope lateral32 show the sensor output in the current resolution without any decimal point.

Terminal resistor is only activated if the device is powered! This could lead under specific network conditions to problems.

11 Appendix A: TILTIX CANopen Objects

(ro- Read Only, wo- Write Only & rw – Read or Write)

11.1 Object 1000h: Device Type

The object at index 1000h describes the type of device and its functionality. It is composed of a 16-bit field which describes the device profile that is used and a second 16-bit field which gives additional information about optional functionality of the device. The additional information parameter is device profile specific.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 32	0X4019A[1] 0X3019A[2]	ro

1) Dual Axis
2) Single Axis

11.2 Object 1001h: Error Register

This object is used by the device to display internal faults. If a fault is detected, the corresponding bit is activated. The following errors are supported.

Bit	Description	Comments
0	Generic Error	The generic error is signaled at any error situation

The object description for error register.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 8	N/A	ro

11.3 Object 1003h: Pre-Defined Error Field

The object holds the errors that have occurred on the device and have been signaled via the Emergency Object. The error code is located in the least significant word and additional information is located in the most significant word. Sub-index 0 contains the number of recorded errors.

Subindex	Description	Data Type	Default Value	Access
0	Number of recorded errors	Unsigned 8	0	rw
1	Most recent errors	Unsigned 32	–	ro

To clear error Log: Write data 0 into sub-index 0 of object 1003.

11.4 Object 1005h: COB-ID ~~Sync~~SYNC Message

This object indicates the configured COB-ID of the synchronization object. In operational mode, the sensor sends process data if SYNC message with same COB-ID is received. (Correct setting of transmission type of TxPDO(s) implied)

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 32	0x80	rw

11.5 Object 1008h: Mfr Device Name

This object contains the device name.

Subindex	Description	Data Type	Default Value	Access
0	–	String	–	ro

11.6 Object 1009h: Mfr Hardware Version

This object contains the hardware version of the circuit board.

Subindex	Description	Data Type	Default Value	Access
0	–	String	–	ro

11.7 Object 100Ah: Mfr Software Version

This object contains the manufacturer software version. The CA01 line has a major revision of 2. (02.xx)

Subindex	Description	Data Type	Default Value	Access
0	–	String	–	ro

11.8 Object 100Ch: Guard Time

This object contains the guard time in milliseconds

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 16	0	rw

11.9 Object 100Dh: Life Time Factor

This object contains the life time factor parameters. The life time factor multiplied with the guard time gives the life time for the node guarding protocol.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 8	0	rw

11.10 Object 1010h: Store Parameters

This object controls the saving of all writable parameters in non-volatile memory.

Subindex	Description	Data Type	Default Value	Access
0	Number of sub-indices	Unsigned 8	4	ro
1	Save all parameters	Unsigned 32	–	rw
2	Save communication parameters	Unsigned 32	–	rw
3	Save application parameters	Unsigned 32	–	rw
4	Save manufacturer parameters	Unsigned 32	–	rw

Storing Procedure: To save the parameters to non-volatile memory, the access signature “save” has to be sent to the corresponding sub-index of the device.

	Most Significant Word		Least significant word	
ASCII	e	v	a	s
Hex value	65h	76h	61h	73h

11.11 Object 1011h: Restore Parameters

This object is used to restore device and CANopen related writeable parameters to default factory settings. **Restore command does not affect baud rate, node number, termination resistor and auto bootup.**

Subindex	Description	Data Type	Default Value	Access
0	Number of sub-indices	Unsigned 8	4	ro
1	Restore all parameters	Unsigned 32	–	rw
2	Restore communication parameters	Unsigned 32	–	rw
3	Restore application parameters	Unsigned 32	–	rw
4	Restore manufacturer parameters	Unsigned 32	–	rw

Loading procedure: To load the parameters from non volatile memory the access signature “load” has to be sent to the corresponding subindex of the device.

	Most Significant Word		Least significant word	
ASCII	d	a	o	l
Hex value	64h	61h	6Fh	6Ch

Note: A power off/on sequence is required for the restoration to take place after the message has been transmitted.

11.12 Object 1014h: COB-ID Emergency

This object indicates the configured COB-ID for the emergency write service. This has the highest priority among transmitted messages.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 32	0x81	rw

11.13 Object 1015h: Inhibit time EMCY

This object defines the inhibit time used for the emergency message.

The time must be a multiple of 100 milliseconds.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 16	0	rw

11.14 Object 1016h: Consumer Heartbeat Time (Not Approved)

The consumer heartbeat time defines the expected heartbeat cycle time in ms. The device can only monitor one corresponding device. If the time is set to 0 the monitoring is not active. The value of this object must be higher than the corresponding time (object 1017) of the monitored device.

Subindex	Description	Data Type	Default Value	Access
0	Number of indices	Unsigned 8	1	ro
1	Consumer heartbeat time	Unsigned 32	0	rw

The context of subindex 1 is as follows:

Bit	31 to 24	23 to 16	15 to 0
Value	0h (reserved)	Address of monitored device (Node-ID)	Monitoring time (ms)

11.15 Object 1017h: Producer Heartbeat Time (Not Approved)

The object contains the time interval in milliseconds in which the device has to produce a heartbeat message. The values are given in multiples of 1ms.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 16	0	rw

11.16 Object 1018h: Identity Object

This object provides the general identification of the inclinometer.

Subindex	Description	Data Type	Default Value	Access
0	Number of entries	Unsigned 8	4	ro
1	Vendor Id	Unsigned 32	0x42	ro
2	Product Code	Unsigned 32	0x41435333 (ACS3)	ro

3	Revision Number	Unsigned 32		ro
4	Serial Number	Unsigned 32		ro

11.17 Object 1020h Verify Configuration

This object indicates the date and time that the configuration was stored. A configuration tool or a master can use this object to verify the configuration after a reset and, if necessary, perform a new configuration.

Subindex	Description	Data Type	Default Value	Access
0	Highest sub-index supported	Unsigned 8	0x2	ro
1	Configuration date	Unsigned 32		rw
2	Configuration time	Unsigned 32		rw

11.18 Object 1029h Error behavior object

This object is used to define what the NMT state of the controller should be in case of an error.

Subindex	Description	Data Type	Default Value	Access
0	Highest sub-index supported	Unsigned 8	0x1	ro
1	Communication error	Unsigned 8		rw

11.19 Object 2200h: Cyclic Timer

This object is used to determine the transmission interval during cyclic transmission mode. This object is hardwired with the event timers of TxPDO1 and TxPDO2. If cyclic timer and event timers of TxPDOs hold different values after power off/on cycle, the value of the event timers is copied to the cycle time.

Subindex	Description	Data Type	Default Value	Access
0	–	Unsigned 16	0	rw

Please note that this object is not stored when using object 0x1010sub4 (Store manufacturer specific parameters)

11.20 Object 2300h: Save Parameter with Reset

With this object all parameters can be stored in the non volatile memory. All parameters are saved with an additional reset executed afterwards.

Subindex	Description	Data Type	Default Value	Access
0	Access code	Unsigned 32	55AAAA55h	wo

11.21 Object 2600h: Preset X-Axis

In ACS3 inclinometers, this object sets the X-axis to the desired value. In TILTIX360 inclinometers, this object sets the Z-axis to the desired value. It is hard-wired with objects 6012 and 6112.

Subindex	Description	Data Type	Default Value	Access
0	–	Integer 16	–	rw

The valid preset value ranges depend on the current resolution of the TILTIX. An error message will appear if the Preset value is out of range.

Angular Resolution	Range of Position Values		
	ACS-090-2	ACS-360-1	ACS-180-E/-F
1º	± 90	360	± 90
0.1º	± 900	3600	± 900
0.01º	± 9000	36000	± 9000
0.001º	± 90000	360000	± 90000

11.22 Object 2601h: Preset Y-Axis

In ACS3 inclinometers, this object sets the Y-axis to a desired value. It is hard-wired with objects 6022 and 6122.

Subindex	Description	Data Type	Default Value	Access
0	–	Integer 16	–	rw

The valid preset value ranges depend on the current resolution of the TILTIX. An error message will appear if the preset value is out of range.

Angular Resolution	Range of Position Values		
	ACS-090-2	ACS-360-1	ACS-180-E/-F
1º	± 90	-	± 180
0.1º	± 900	-	± 1800
0.01º	± 9000	-	± 18000
0.001º	± 90000	-	± 180000

After setting the Preset value a save command must be given in order to set the Preset value permanently.

11.23 Object 3000h: Node Number

This object contains the node number of the device.
The POSITAL standard node number is 01 decimal.
The valid node numbers range from 01 to 127.

Subindex	Description	Data Type	Default Value	Access
0	Node Number	Unsigned 8	0	rw

NOTE: To avoid the node number zero (0), one (1) will be added to the value of this object. E.g.: To set node number 32 decimal – Write 1F hex to the object and save. 1Fh+1h = 20h = 32 (dec) will finally be set as the node number. **The new node number is only adopted after the device is reset, i.e. the node number needs to be saved prior to the reset to have an effect.**

Programming example (for NN=1): Set Node Number (NN) to 50 Decimal

Step 1: Write 31h in object 3000Sub0 (the saved value is increased by 1; 31h + 01h = 32h = 50dec)

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	00	30	00	31	00	00	00

Step 2: Save this configuration by writing "save" to 1010Sub01

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	10	10	01	73	61	76	65

Step 3: The sensor will adapt the new Node number after a reset!

11.24 Object 3001h: Baud rate

This object contains the baud rate of the device. Valid value ranges from 0 to 7.

Subindex	Description	Data Type	Default Value	Access
0	Baud rate	Unsigned 8	–	rw*

Eight different baud rates are provided. To adjust the baud rate only one byte is used. The default baud rate is 125 kB (03h).

Baud rate in kB	Stored Value in Object
20	00h
50	01h
100	02h
125	03h
250	04h
500	05h
800	06h
1000	07h

The new baud rate is only adopted after the device is reset, i.e. the baud rate needs to be saved prior to the reset to have an effect.

Programming example (for NN=1): **Set Baudrate to 500 kBaud**

Step 1: Set Baudrate 3001Sub0 to 500kBaud (05h)

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	01	30	00	05	00	00	00

Step 2: Save Parameters and execute a reset. The reset is needed to restart the sensor with the configured Baudrate

Identifier	DLC	Command	Index	Subindex	Service/Process data

NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
~~6~~01	8	22	00	23	00	55	AA	AA	55

11.25 Object 3002h: Termination Resistor (not usable yet)

This object is used to activate the termination resistor in case the TILTIX is used at the edges of the transmission line (to avoid reflection of data back into the line) and/or high transmission rates (>50kB).

Subindex	Description	Data Type	Default Value	Access
0	Activate Termination Resistor	Unsigned8	0	rw

For Activation[1] – Write I decimal to the sub index 0 of the object

For De-activation – Write 0 decimal to the sub index 0 of the object

1) Note: If activated, the termination resistor is only enabled while the device is powered.

11.26 Object 3010h: Filter Mode

The filter mode needs to be set to define the active filter. The individual filter parameters need to be set in the related object.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Filter Mode	Unsigned 8	0	6	2	rw

Value	Description	Related Object
0	No filter active	-
1	Recursive Filter	3022h
2	Moving Average Filter Short	3100h
3	Moving Average Filter Long	3110h
4	Butterworth Filter	3120h
5	Critically Damped Filter	3130h
6	Kalman Filter	3140h

When the active filter is switched between Moving Average Filter (backwards compatible) and Moving Average Filter Long, the related moving average filter length needs to be set again by the customer.

11.27 Object 3022h: Recursive Filter

This object controls the coefficient (weighting factor) of the recursive filter. If set to 0, the filter is deactivated. Valid values range from 0 to 999. The value specifies the weighting factor divided by 1000. Therefore, mathematically the weighting factor ranges from 0 to 0.999.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Recursive Filter Coefficient	Unsigned 16	0	999	970	rw

For calculation of the position values with the recursive filter activated:

$$y_n = a \cdot y_{n-1} + (1 - a) \cdot x_n$$

n: iteration cycle; x: filter input value; y: filter output value; a: filter coefficient

11.28 Object 3100h: Moving Average Filter (backwards compatible)

This object contains the filter length for the calculation of the arithmetic mean value. A value of 100 will result in a filter length of 200 ms. The filter is deactivated if set to 0.

Subindex	Description	Data Type	Min	Max	Default Value	Access
0	Moving Average Filter (*2ms)	Unsigned 16	0	250	100	rw

Internal calculation of position values when MAF is activated with length N:

New_Sensor_Valuex = [Sensor_Valuex + Sensor_Valuex-1 + …..+ Sensor_Valuex-N] / N

$$y_n = \frac{\sum_{i=1}^{N}(x_i)}{N}$$

11.29 Object 3110h: Moving Average Filter Long

The Moving Average Filter Long is similar to the moving average filter of Object 3100h but allows for longer filter length times of up to 5s.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	4	ro
01h	Moving Average Filter Long Mode 0 – Standard 1 – Expert	Unsigned 8	0	1	0	rw
02h	Moving Average Filter Length [ms]	Unsigned 16	1	5000	200	rw
03h	MAF buffer length AF [ms]	Unsigned 8	1	50	2	rw
04h	MAF buffer length MAF	Unsigned 8	1	100	100	rw

Index 01h offers the choice between two modes: Standard and Expert.

Opting for the Standard Mode allows you to effortlessly configure the total Filter Length via sub index 02h, as the internal buffer lengths are automatically adjusted.

In contrast, activating the Expert Mode empowers you to personally define the two Buffer Lengths through sub index 03h and 04h.

$$\text{Sub index } 03\text{h} * \text{Sub index } 04\text{h} = \text{Sub index } 02\text{h}$$

11.30 Object 3120h: Butterworth Filter

The Butterworth Filter is a classical low pass frequency filter with the focus on frequency response. The Cut-off Frequency is set in mHz and suppresses frequencies above the set threshold.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	2	ro
01h	Filter Order	Unsigned 8	1	10	1	rw
02h	Cut-off Frequency [mHz]	Unsigned 16	1	65535	1000	rw

11.31 Object 3130h: Critically Damped Filter

The Critically Damped Filter is a low pass frequency filter like the Butterworth Filter. The Cut-off Frequency is set in mHz and suppresses frequencies above the set threshold. This filter will not overshoot, even for higher filter orders, while its band transition is less sharp compared to the Butterworth Filter.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	2	ro
01h	Filter Order	Unsigned 8	1	10	1	rw
02h	Cut-off Frequency [mHz]	Unsigned 16	1	65535	1000	rw

11.32 Object 3140h: Kalman filter

The Kalman Filter is an adaptive filter that automatically shifts between static (stationary without many movements) and dynamic (in motion) mode. The Static Damping parameter defines the filter strength in the static mode, while the Dynamic Factor determines how much this filter strength is lowered upon transition to the dynamic mode.

The motion threshold is set in m*g* (milli *g*) and needs to be adjusted to the system noise. The best way to find this out is to record the accelerometer data of this inclinometer in a machine idle state, where the machine is powered on, but not in motion. The motion threshold should be set well above the peak-to-peak noise value. Ten times the standard deviation is typically a good starting point. The parameters Static Damping and Dynamic Factor set the filter strength for static and dynamic mode where the value 100 reflects maximum and 0 minimum.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	3	ro
01h	Static Damping [%]	Unsigned 8	0	100	70	rw
02h	Motion Threshold [m*g*]	Unsigned 16	0	1000	2	rw
03h	Dynamic Factor [%]	Unsigned 8	0	100	50	rw

11.33 Object 3200h: Angle Mode

The Angle Mode (measurement axis) of the inclinometer can be changed by setting of the Angle Mode Object 3200h, e.g. a single axis inclinometer can be re-configured as dual axis inclinometer and vice versa

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Angle Mode	Unsigned 8	1	4	-	rw

Value	Description	Related Type key ACS-xxx-x-CA01-xx3-xx
1	Pitch, Roll ± 85°/± 180° horizontal mounting	ACS-**180-E**-CA01-**H**x3-xx
2	2 Axes ±-90°	ACS-**090-2**-CA01-**H**x3-xx
3	1 Axis 360°	ACS-**360-1**-CA01-**V**x3-xx
4	Pitch, Roll ±85°/± 180° vertical mounting	ACS-**180-F**-CA01-**V**x3-xx

Whenever the angle mode is changed, the following objects are reset to default values:

- **1000h Device type**
- **1A00h TPDO1 mapping parameters**
- **1A01h TPDO2 mapping parameters**
- **2600h Preset X-axis**
- **2601h Preset Y-axis**
- **3300h Range limit slope long**
- **3301h Range limit slope lateral**
- **6000h Resolution**
- **6011h Slope long16 operating parameter**
- **6012h Slope long16 preset value**
- **6013h Slope long16 offset**
- **6014h Differential slope long16 offset**
- **6021h Slope lateral16 operating parameter**
- **6022h Slope lateral16 preset value**
- **6023h Slope lateral16 offset**
- **6024h Differential slope lateral16 offset**
- **6111h Slope long32 operating parameter**

- **6112h Slope long32 preset value**
- **6113h Slope long32 offset**
- **6114h Differential slope long32 offset**
- **6121h Slope lateral32 operating parameter**
- **6122h Slope lateral32 preset value**
- **6123h Slope lateral32 offset**
- **6124h Differential slope lateral32 offset**

For saving the angle mode change, the Save all parameters command of object 1010h Sub 01 must be used.

11.34 Object 3250h: Preset Mode

There are two different preset modes available for the angle versions with pitch and roll output.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Preset Mode	Unsigned 8	0	1	0	rw/ro

0 = Backwards compatible preset logic. The physical measurement range is preserved while the offset value conducted from the preset value is added to the output angle. For ±90° angles the displayable range changes according to the preset while for 360°/±180° angles the range is preserved.
1 = New preset logic based on preserved angle ranges with shifted 0 position. The position limits stay the same and the turning point where the graph switches from positive to negative slope moves across the physical position. For the new preset logic and the restricted angles (±90°) both the preset input value (by customer) and the current position value (used to calculate the offset) are interpreted as if the sensor was oriented in a plain way pointing down with its bottom side (not upside-down). This means that any preset operation in an upside-down position will lead to unexpected values for the customer output.
Check diagram in appendix 12.3 for reference.

11.35 Object 3300h: Range Limit Slope Long

With this object, the minimum and maximum output range of the inclinometers slope long angle can be limited. Range limits are set with the current resolution.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	2	ro
01h	Range Minimum	(Un)signed 32				rw
02h	Range Maximum	(Un)signed 32				rw

11.36 Object 3301h: Range Limit Slope Lateral

With this object, the minimum and maximum output range of the inclinometers slope lateral angle can be limited. Range limits are set with the current resolution.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Number of Entries	Unsigned 8	-	-	2	ro
01h	Range Minimum	Signed 32				rw
02h	Range Maximum	Signed 32				rw

11.37 Object 3310h: Angle Hysteresis

With this object, the hysteresis can be set. Depending on the direction of rotation, the position value deviates from the physical position by the set hysteresis value [0.001°]

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Angle Hysteresis [0.001°]	Unsigned 16	0	65535	0	rw

11.38 Object 3403h: Acceleration X-Axis 16

This object provides the acceleration value in x axis. The value is given in m*g* with a specified range of ±6 *g*, reliable output range (specified by the MEMS manufacturer) is ±1.5 *g*.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Acceleration X-Axis 16	Signed 16				ro

11.39 Object 3404h: Acceleration Y-Axis 16

This object provides the acceleration value in x axis. The value is given in mg with a specified range of ±1.5 g. (Usually up to ±6 g can be displayed but cannot be guaranteed.)

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Acceleration Y-Axis 16	Signed 16				ro

11.40 Object 3405h: Acceleration Z-Axis 16

This object provides the acceleration value in x axis. The value is given in mg with a specified range of ±1.5 g. (Usually up to ±6 g can be displayed but cannot be guaranteed.)

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Acceleration Z-Axis 16	Signed 16				ro

11.41 Object 3450h: Diagnostics 16

This object provides the Process and Diagnostic data.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Diagnostics 16	Unsigned 16			0	ro

Process data / Diagnostic data

Bit 8-15	Bit 6-7	Bit 4-5	Bit 2-3	Bit 0-1
reserved	Temperature range status	Self-test output status	Slop lateral range status	Slope long range status

Value	Slope long/ lateral range status
0	Angle within range
1	Angle smaller than range
2	Angle larger than range

Value	Self-test output status
0	Sensor working fine
1	STO lower limit exceeded
2	STO upper limit exceeded

Value	Temperature range status
0	Temperature within range of -40°C … +85°C
1	Temperature lower than -40°C
2	Temperature higher than +85°C*

*Temperature is measured within sensor and will differ from ambient temperature. If the temperature status is unequal to zero, the sensor does not necessarily have a problem.

11.42 Object 3451h: Self Test Output (STO) 16

STO is used to monitor if the accelerometer is functioning correctly. STO threshold monitoring should be implemented on application software. Failure thresholds and failure tolerant time of the system are application specific and should be carefully validated. Monitoring can be implemented by counting the subsequent “STO signal exceeding threshold” events.

Component failure can be suspected if the STO signal exceeds the threshold level continuously after performing component hard reset in static (no vibration) condition.

Subindex	Description	Data Type	Min	Max	Default Value	Access
00h	Self Test Output (STO) 16	Signed 16				ro

11.43 Object 6000h: Resolution

This object shall indicate the resolution of Slope long16 (object 6010h) and Slope lateral16 (object 6020h) objects, as well as Slope long32 (object 6110h) and Slope lateral32 (object 6120h).

Subindex	Description	Data Type	Default Value	Access
0	Resolution	Unsigned 16	10	rw

Four possible resolutions can be used:

Angular Resolution	Value decimal	Byte hex
0.001°	1	01h
0.01º	10	0Ah
0.1°	100	64h
1°	1000	3e8h

⚠ For a resolution of 0.001° the 32bit objects have to be used in the PDO mapping to ensure proper output values.

11.44 Object 6010h: Slope Long16

Position Value X-Axis/Longitudinal Axis (dual axes), Z-Axis (single axis), or Pitch-Axis (Pitch and Roll)

For dual axes inclinometers, this object provides the X-axis position value and for single axis inclinometers, this object provides the Z-axis value. This is a read only object and the position values are limited depending on the resolution set.

Subindex	Description	Data Type	Default Value	Access
0	Slope Long16	Signed Integer 16 – dual axis Unsigned Integer 16 – single axis	–	ro

Angular Resolution	Range of Position Values		
	ACS-090-2	ACS-360-1	ACS-180-E/-F
1º	± 90	± 360	± 180
0.1º	± 900	± 3600	± 1800
0.01º	± 9000	± 36000	± 18000
0.001º	± 90000	± 360000	± 180000

Calculation of position / slope value from CANopen readout:

Inclination (Degree) = Slope Long 16[Object 6010] x Resolution [Object 6000]

If attribute “resolution” is changed, side effects will occur, as the sensor is adapting the already set parameters to the configured resolution. Rounding effects have to be taken into account. Attributes affected: write/read: preset slope long16, preset slope lateral16, differential slope long16 offset, differential slope lateral16 offset, preset slope long32, preset slope lateral32, differential slope long32 offset, differential slope lateral32 offset; read only: slope long16, slope lateral16, slope long16 offset, slope lateral16 offset, slope long32, slope lateral32, slope lateral32 offset, slope lateral32 offset.

11.45 Object 6011h: Slope long16 operating parameter

This object controls the scaling and inversion of the Slope long16 values. If bit 0 is set to 1, inversion of Slope long16 is enabled. If bit 1 is set to 1, Slope long16 offset and Differential Slope long16 offset are added to the Slope long16. Both bits can be set at the same time.

Subindex	Description	Data Type	Default Value	Access
0	Position Scaling and Inversion	Unsigned 8	2	rw

Calculation of position value based on the operating parameter:

Slope Long16 = [(Physical inclination of slope long16 * Multiplier) + Slope long16 offset + Differential Slope long16 offset]. With:

- Physical inclination of slope long16 (X)
- Multiplier: if inversion is disabled: 1, if inversion is enabled: -1
- Slope Long16 Offset (6013) = Slope Long16 Preset (6012) – (Physical inclination of slope long16 (X) at tacc * Multiplier), with tacc = preset acquisition time
- Differential slope long16 offset (6014): Adds an absolute value (positive or negative) to the sensor

Bit 1	Bit 0	Position Calculation
0	0	X
0	1	-X
1	0	(X + Object 6013h + Object 6014h)
1	1	(-X + Object 6013h + Object 6014h)

For detailed explanation please see the corresponding objects.

11.46 Object 6012h: Slope Long16 preset value

This object is used to define a new, desired longitudinal slope value. The values will be set only according to the current resolution, so extreme care should be taken in order to be within the range of the position value limits.

Subindex	Description	Data Type	Default Value	Access
0	Preset Value	(Un)signed 16	0	rw

Angular Resolution	Range of Position Values ACS-090-2	Range of Position Values ACS-360-1	Range of Position Values ACS-180-E/-F
1º	± 90	± 360	± 180
0.1º	± 900	± 3600	± 1800
0.01º	± 9000	± 36000	± 18000
0.001º	± 90000	± 360000	± 180000

Programming example (for NN=1): Preset Value - Set the current position to 1°

Step 1: Set preset value to 64h. If the resolution is set to 0,01°, 64h is equal to 1° (Note: If the resolution is set to 0.1°, the preset value must be set to Ah to set the current position to 1°)

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	00	26	00	64	00	00	00

Step 2: Save this configuration by writing "save" to 1010Sub01

Identifier	DLC	Command	Index		Subindex	Service/Process data			
NN = 1		Download				Byte 4	Byte 5	Byte 6	Byte 7
601	8	22	10	10	01	73	61	76	65

If a range limit is active, the preset angle value will only be shown correctly within the defined angle range. If you preset in a sensor position outside the restricted angle range, you might get unexpected angle outputs outside this range.

11.47 Object 6013h: Slope Long16 offset

This is a read only parameter. The parameter is recalculated every time Slope Long16 preset (6012h), Slope Long32 Preset (6112h) or Preset X (2600h) objects are changed.

Subindex	Description	Data Type	Default Value	Access
0	Offset Value	(Un)signed 16	–	ro

Calculation of Slope Long16 Offset:

Offset Value = Preset Value - (Physical Value x Inversion) - Differential Offset*

[Inversion is -1 if activated; 1 if deactivated]

[* at time of preset input]

11.48 Object 6014h: Differential slope long16 offset

This parameter adds an additional, independent offset to Slope long16, but only if scaling of slope long16 is enabled. Please note that the values should be within the range of position attribute according to the resolution already set.

Subindex	Description	Data Type	Default Value	Access
0	Offset Value	(Un)signed 16	–	rw

The main difference between the objects 6013h and 6014h is that 6013h is calculated automatically depending on the preset value whereas object 6014h is user defined.

11.49 Object 6020h: Slope lateral16

Position Value Y-Axis (ACS-090-2 Lateral Axis) or Roll-Axis (ACS-180-E Pitch and Roll)

For dual axis inclinometers, this object provides the Y- or Roll-Axis position value. This is a read only object. The position values are limited depending on the resolution set.

Important: This object will output 0 for a sensor with one axis output.

Subindex	Description	Data Type	Default Value	Access
0	–	Signed 16	–	ro

Angular Resolution	Range of Position Values	
	ACS-090-2	ACS-180-E/-F
1º	± 90	± 180
0.1º	± 900	± 1800
0.01º	± 9000	± 18000
0.001º	± 90000	± 180000

Calculation of position / slope value from CANopen readout:

Inclination (Degree) = Slope Lateral 16[Object 6020] x Resolution [Object 6000]

11.50 Object 6021h: Slope lateral16 operating parameter

This object controls the scaling and inversion of the Slope lateral16 values. If bit 0 is set, inversion of Slope lateral16 is enabled. If bit 1 is set, Slope lateral16 offset and Differential slope lateral16 offset are added to the Slope lateral16. Both bits can be set at the same time.

Subindex	Description	Data Type	Default Value	Access
0	Position Scaling and Inversion	Unsigned 8	2	rw

Calculation of position value based on the operating parameter:

Slope Lateral16 = [(Physical inclination of slope lateral16 * Multiplier) + Slope lateral16 Offset + Differential Slope lateral16 Offset]

With:

- Physical inclination of slope lateral16 (X)
- Multiplier: if inversion is disabled: 1, if inversion is enabled: -1
- Slope Lateral16 Offset (6023) = Slope Lateral16 Preset (6022) – (Physical inclination of slope lateral16 (X) at t_{acc} * Multiplier), with t_{acc} = preset acquisition time
- Differential slope lateral16 offset (6024): Adds an absolute value (positive or negative) to the sensor output.

Bit 1	Bit 0	Position Calculation
0	0	X
0	1	-X
1	0	(X + Object 6023h + Object 6024h)
1	1	(-X + Object 6023h + Object 6024h)

For detailed explanation please see the corresponding objects.

11.51 Object 6022h: Slope lateral16 preset value

This object is used to define a new, desired lateral slope value. The values will be set only according to the current resolution, so extreme care should be taken in order to be within the range of the position value limits.

Subindex	Description	Data Type	Default Value	Access
0	Preset Value	Signed 16	0	rw

Resolution	Preset Range (in decimal)	Preset Range (in decimal)
Type	ACS-090-2	ACS-180-E/-F
1º	± 90	± 180
0.1º	± 900	± 1800
0.01º	± 9000	± 18000
0.001º	± 90000	± 180000

If a range limit is active, the preset angle value will only be shown correctly within the defined angle range. If you preset in a sensor position outside the restricted angle range, you might get unexpected angle outputs outside this range.

11.52 Object 6023h: Slope lateral16 offset

This is a read only parameter. The parameter is recalculated every time Slope Lateral16 preset (6022h), Slope Lateral32 Preset (6122) or Preset Y (2601) objects are changed. This object does not exist for TILTIX360

Subindex	Description	Data Type	Default Value	Access
0	Offset Value	Signed 16	–	ro

Calculation of Slope Lateral16 Offset:

Offset Value = Preset Value - (Physical Value x Inversion) - Differential Offset*

[Inversion is -1 if activated; 1 if deactivated]

[* at time of preset input]

11.53 Object 6024h: Differential slope lateral16 offset

This parameter adds an additional, independent offset to Slope lateral16, but only if scaling of Slope lateral16 is enabled. Please note that the values should be within the range of position attribute according to the resolution already set.

Subindex	Description	Data Type	Default Value	Access
0	Offset Value	Signed 16	–	rw

The main difference between the objects 6023h and 6024h is that 6023h is calculated automatically depending on the preset value whereas object 6024h is user defined.

32-Bit Objects (61xxh)

Objects 6110h – 6114h: Hardwired with 16-bit slope long16 objects.

Objects 6120h – 6124h: Hardwired with 16-bit slope lateral16 objects.

Overview of Objects

Object Name	Object No. ID hex	Access	Data Type
Device type	1000	Read-only	UNSIGNED32
Error Register	1001	Read-only	UNSIGNED8
Predefined Error Field	1003	–	ARRAY
Sub 0: Number of errors	1003.0	Read/write	UNSIGNED8
Sub 1: Standard error field	1003.1	Read-only	UNSIGNED32
COB-ID SYNC	1005	Read/write	UNSIGNED32
Communication cycle period	1006	Read/write	UNSIGNED32
Synchronous window length	1007	Read/write	UNSIGNED32
Device Name	1008	Read-only	VISIBLE_STRING
Hardware Version	1009	Read-only	VISIBLE_STRING
Software Version	100a	Read-only	VISIBLE_STRING
Guard Time	100c	Read/write	UNSIGNED16
Life Time Factor	100d	Read/write	UNSIGNED8
Store Parameters	1010	–	ARRAY
Highest sub-index supported	1010.0	Read-only	UNSIGNED8
Save all parameters	1010.1	Read/write	UNSIGNED32
Save communication parameters	1010.2	Read/write	UNSIGNED32
Save Application parameters	1010.3	Read/write	UNSIGNED32
Save Manufacturer parameters	1010.4	Read/write	UNSIGNED32
Restore Default Parameters	1011	–	ARRAY
Highest sub-index supported	1011.0	Read-only	UNSIGNED8
Restore all parameters	1011.1	Read/write	UNSIGNED32
Restore communication parameters	1011.2	Read/write	UNSIGNED32
Restore application parameters	1011.3	Read/write	UNSIGNED32
Restore manufacturer parameters	1011.4	Read/write	UNSIGNED32
COB-ID Emergency	1014	Read-only	UNSIGNED32
Inhibit time EMCY	1015	Read/write	UNSIGNED16
Consumer Heartbeat Time	1016	–	ARRAY
Sub 0: No. of Entries	1016.0	Read-only	UNSIGNED8
Sub 1: Consumer Heartbeat Time	1016.1	Read/write	UNSIGNED32
Producer Heartbeat Time	1017	Read/write	UNSIGNED16
Identity Object	1018	–	RECORD
Highest sub-index supported	1018.0	Read-only	UNSIGNED8
Vendor ID	1018.1	Read-only	UNSIGNED32
Product Code	1018.2	Read-only	UNSIGNED32

Revision Number	1018.3	Read-only	UNSIGNED32
Serial Number	1018.4	Read-only	UNSIGNED32
Verify configuration	1020		ARRAY
Highest sub-index supported	1020.0	Read-only	UNSIGNED8
Configuration date	1020.1	Read/write	UNSIGNED32
Configuration time	1020.2	Read/write	UNSIGNED32
Error behavior object	1029		ARRAY
Highest sub-index supported	1029.0	Read-only	UNSIGNED8
Communication error	1029.1	Read/write	UNSIGNED8
TxPDO1 Comm Parameter	1800	–	RECORD
Sub 0: No. of Entries	1800.0	Read-only	UNSIGNED8
Sub 1: COB-ID	1800.1	Read/write	UNSIGNED32
Sub 2: Transmission Type	1800.2	Read/write	UNSIGNED8
Sub 3: Inhibit Time	1800.3	Read/write	UNSIGNED16
Sub 5: Event Timer	1800.5	Read/write	UNSIGNED16
TxPDO2 Comm Parameter	1801	–	RECORD
Sub 0: No. of Entries	1801.0	Read-only	UNSIGNED8
Sub 1: COB-ID	1801.1	Read/write	UNSIGNED32
Sub 2: Transmission Type	1801.2	Read/write	UNSIGNED8
Sub 3: Inhibit Time	1801.3	Read/write	UNSIGNED16
Sub 5: Event Timer	1801.5	Read/write	UNSIGNED16
TxPDO1 Mapping	1A00	–	RECORD
Sub 0: Number of entries	1A00.0	Read/write	UNSIGNED8
Sub 1: 1st mapped object	1A00.1	Read/write	UNSIGNED32
Sub 2: 2nd mapped object	1A00.2	Read/write	UNSIGNED32
Sub 3: 3rd mapped object	1A00.3	Read/write	UNSIGNED32
Sub 4: 4th mapped object	1A00.4	Read/write	UNSIGNED32
TxPDO2 Mapping	1A01	–	RECORD
Sub 0: Number of entries	1A01.0	Read/write	UNSIGNED8
Sub 1: 1st mapped object	1A01.1	Read/write	UNSIGNED32
Sub 2: 2nd mapped object	1A01.2	Read/write	UNSIGNED32
Sub 3: 3rd mapped object	1A01.3	Read/write	UNSIGNED32
Sub 4: 4th mapped object	1A01.4	Read/write	UNSIGNED32
Cyclic Time	2200	Read/write	UNSIGNED16
Store Parameters	2300	Write-only	UNSIGNED32
Preset X-Axis (dual axis), Preset Z-Axis (one axis)	2600	Read/write	INTEGER16
Preset Y-Axis (only dual axis)	2601	Read/write	INTEGER16
Node Number	3000	Read/write	UNSIGNED8

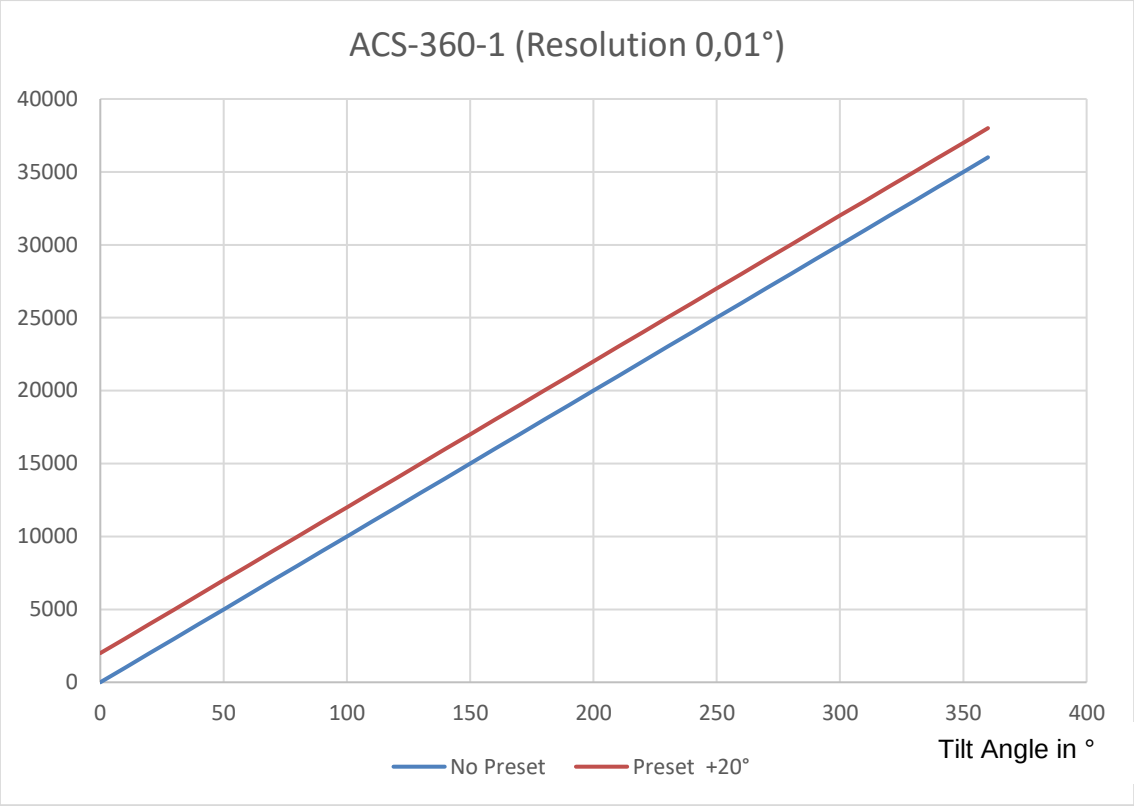
Baud rate	3001	Read/write	UNSIGNED8
Termination Resistor	3002	Read/write	UNSIGNED8
Filter mode	3010	Read/write	UNSIGNED8
Recursive filter	3022	Read/write	UNSIGNED16
Moving Average Filter	3100	Read/write	UNSIGNED16
Moving Average Filter long	3110		RECORD
Highest sub-index supported	3110.0	Read/write	UNSIGNED8
Moving Average Filter long Mode	3110.1	Read/write	Unsigned 8
Moving Average Filter long Length	3110.2	Read/write	UNSIGNED16
Butterworth filter	3120		RECORD
Highest sub-index supported	3120.0	Read-only	UNSIGNED8
Filter order	3120.1	Read/write	UNSIGNED8
Cut-off frequency	3120.2	Read/write	UNSIGNED16
Critically damped filter	3130		RECORD
Highest sub-index supported	3130.0	Read-only	UNSIGNED8
Filter order	3130.1	Read/write	UNSIGNED8
Cut-off frequency	3130.2	Read/write	UNSIGNED16
Kalman filter	3140		RECORD
Highest sub-index supported	3140.0	Read-only	UNSIGNED8
Static damping	3140.1	Read/write	UNSIGNED8
Motion threshold	3140.2	Read/write	UNSIGNED16
Dynamic factor	3140.3	Read/write	UNSIGNED8
Angle mode	3200	Read/write	UNSIGNED8
Preset mode	3250	Read/write	UNSIGNED8
Range limit slope long	3300		RECORD
Highest sub-index supported	3300.0	Read-only	UNSIGNED8
Range minimum	3300.1	Read/write	INTEGER32
Range maximum	3300.2	Read/write	INTEGER32
Range limit slope lateral	3301		RECORD
Highest sub-index supported	3301.0	Read-only	UNSIGNED8
Range minimum	3301.1	Read/write	INTEGER32
Range maximum	3301.2	Read/write	INTEGER32
Angle hysteresis	3310	Read/write	UNSIGNED16
Acceleration X-axis	3403	Read-only	INTEGER16
Acceleration Y-axis	3404	Read-only	INTEGER16
Acceleration Z-axis	3405	Read-only	INTEGER16
Diagnostics	3450	Read-only	UNSIGNED16
Self test output	3451	Read-only	INTEGER16
Resolution	6000	Read/write	UNSIGNED16

Slope long16	6010	Read-only	INTEGER16
Slope long16 operating parameter	6011	Read/write	UNSIGNED8
Slope long16 preset value	6012	Read/write	INTERGER16
Slope long16 offset	6013	Read-only	INTEGER16
Differential slope long16 offset	6014	Read/write	INTEGER16
Slope lateral16 (only 90° version)	6020	Read-only	INTEGER16
Slope lateral16 operating parameter (only 90° version)	6021	Read/write	UNSIGNED8
Slope lateral16 preset value (only 90° version)	6022	Read/write	INTEGER16
Slope lateral16 offset (only 90° version)	6023	Read-only	INTEGER16
Differential slope lateral16 offset (only 90° version)	6024	Read/write	INTEGER16
Slope long32	6110	Read-only	INTEGER32
Slope long32 operating parameter	6111	Read/write	UNSIGNED8
Slope long32 preset value	6112	Read/write	INTEGER32
Slope long32 offset	6113	Read-only	INTEGER32
Differential slope long32 offset	6114	Read/write	INTEGER32
Slope lateral32	6120	Read-only	INTEGER32
Slope lateral32 operating parameter	6121	Read/write	UNSIGNED8
Slope lateral32 preset value	6122	Read/write	INTEGER32
Slope lateral32 offset	6123	Read-only	INTEGER32
Differential slope lateral32 offset	6124	Read/write	INTEGER32

If saving or restoring not all but only a part of the parameters (communication, application or manufacturer parameters), Cyclic Time (object 2200h) and Preset X, Preset Y (80° version, objects 2600 and 2601) or Preset (360° version, object 2600) show a special behavior: Cyclic Time is hardwired with the Event Timers of TxPDO1 and TxPDO2. If only the Event Timers (communication parameters) or Cyclic Time (Manufacturer parameter) is/are saved, the parameters would hold different values after a power off/on cycle. Therefore, if the values do not match, the value of the event timers is copied to the Cycle Time, too. Preset parameters work the same way, with the application parameters (objects 6xxx) having the higher priority.

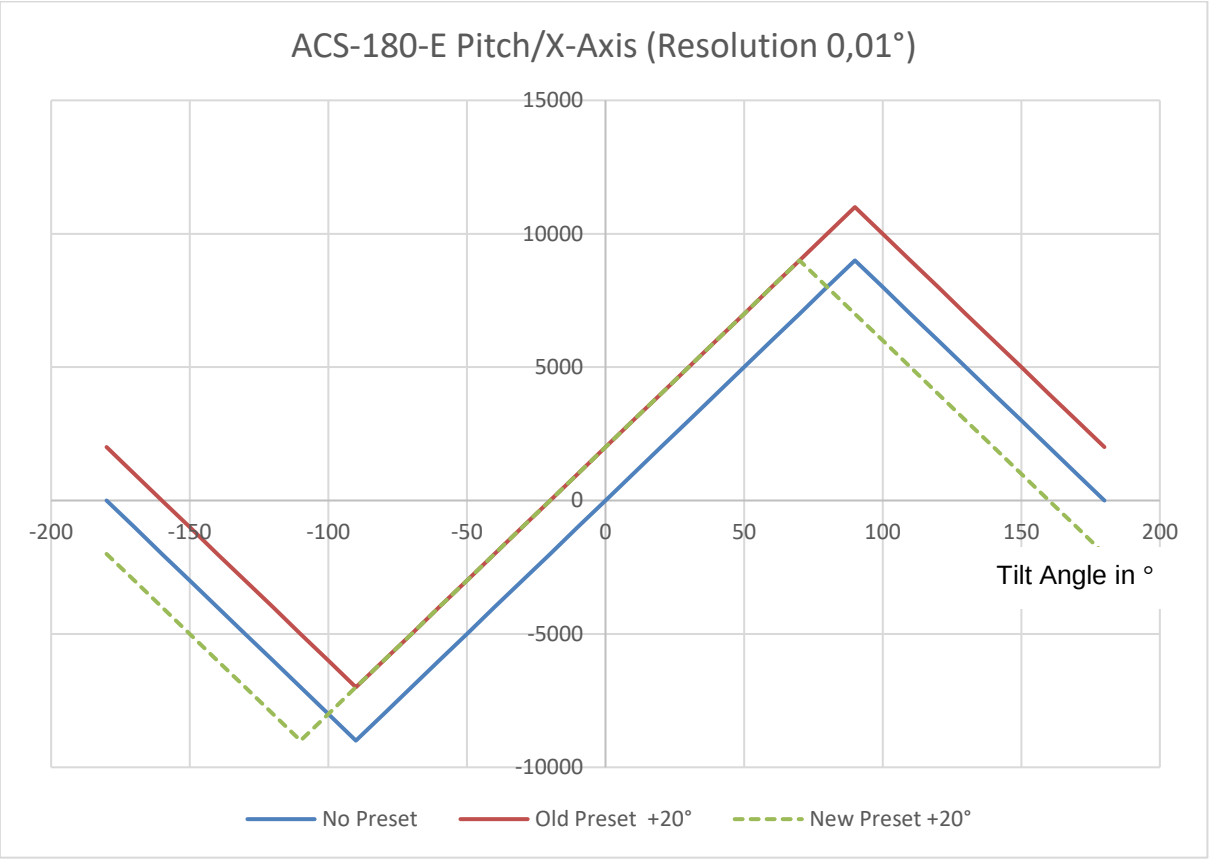
12 Appendix C: Output Graphs

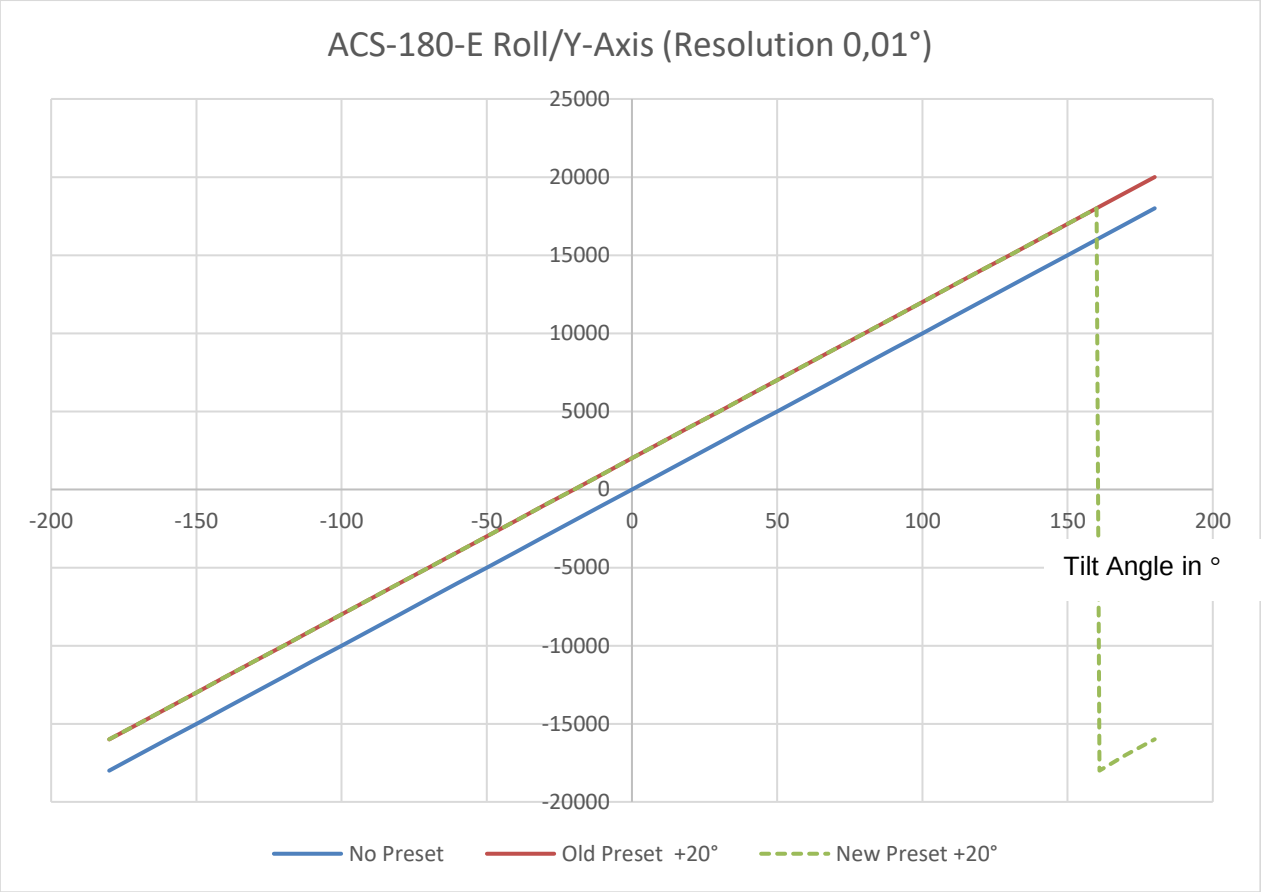
12.1 TILTIX 360-1 Output Values

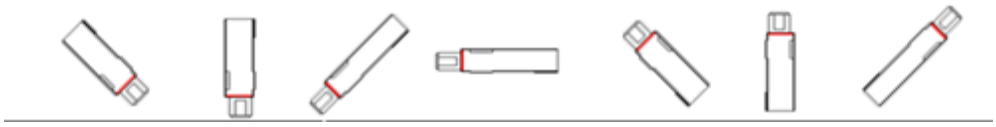


12.2 TITIX 090-2 Output Values

12.3 TILTIX 180-E Output Values




ACS-180-E Roll/Y-Axis (Resolution 0,01°)
25000
20000
15000
10000
5000
0
-5000
-10000
-15000
-20000
-200
-150
-100
-50
0
50
100
150
200
Tilt Angle in °
No Preset
Old Preset +20°
New Preset +20°



13 Glossary

(For more detailed information refer to CiA's CAN Dictionary)

A

Address	Number, assigned to each node, irrespective of whether it is a master or slave. The inclinometer address (non-volatile) is configured in the base with rotary switches.
APV	Absolute Position Value.

B

Bandwidth transmitted	Bandwidth is the value, which denominates the size of information in a defined time unit.
Baud rate	Transmission speed formulated in number of bits per second. Bus node Device that can send and/or receive or amplify data by means of the bus.
Bit Rate	Number of bits per time during transmission, independent of bit representation. The bit rate in CAN networks is limited to 1 Mbit/s.
Bus	Topology of a communication network, where all nodes are reached by passive links. This allows transmission in both directions.
Byte	8-bit unit of data = 1 byte.

C

CAL	CAN application layer. Application layer developed by CiA (CAN in Automation) members providing several communication services and corresponding protocols.
CAN	Controller Area Network or CAN multiplexing network.
CANopen	Application layer of an industrial network based on the CAN bus.
CCW	Counter-clockwise
CiA	CAN In Automation, organization of manufacturers and users of devices that operate on the CAN bus.
CiA 410	The CANopen device profile for inclinometer supports 16-bit as well as 32-bit sensors
COB	Elementary communication object on the CAN network. All data is transferred using a COB.
COB-ID	COB-Identifier. Identifies an object in a network. The ID determines the transmission priority of this object. The COB-ID consists of a function code and a node number.
CW	Clockwise

E

Event Timer	The event timer is assigned in CANopen to one PDO. It defines the frequency of PDO transmission.

F

FC Function Code	Determines the type of message sent via the CAN network.

H

Heartbeat	CANopen and DeviceNet use the heartbeat message to indicate that a node is still alive. This message is transmitted periodically.
Heartbeat Consumer	The heartbeat consumer time defines time interval required by the monitoring node Time to verify,whether a monitored node is alive or not. In case, the monitored node transmits no heartbeat message within that time interval, it is regarded as not alive.

L

Line terminator	Resistor terminating the main segments of the bus.
LMT	Network management object. This is used to configure the parameters of each layer in the CAN. Master "Active" device within the network, that can send data without having received a request. It controls data exchange and communication management.

N

NMT	Network management object. This is responsible for managing the execution, configuration and errors in a CAN network.
NN	Node number

P

PCV	Process Value
PDO	Communication object, with a high priority for sending process data.
PV	Preset Value: Configuration value

R

RO	Read Only: Parameter that is only accessible in read mode.
ROMAP	Read Only MAPable: Parameter that can be polled by the PDO.
RW	Read/Write: Parameter that can be accessed in read or write mode.

S

SDO	Communication object, with a low priority for messaging (configuration, error handling, diagnostics). Slave Bus node that sends data at the request of the master. The inclinometers are always slaves.

W

WO	Write Only: Parameter that is only accessible in write mode.

14 Contact

FRABA America

T +1 609 750-8705

info@posital.com

FRABA Europe

T +49 221 96213-0

info@posital.eu

FRABA Asia

T +65 6514 8880

info@posital.sg