MICRO-EPSILON

Operating Instructions

IF2030/ENETIP

Interface Module

MICRO-EPSILON
MESSTECHNIK
GmbH & Co. KG
Koenigbacher Str. 15

94496 Ortenburg / Germany

Tel. +49 (0) 8542/168-0
Fax +49 (0) 8542/168-90
email info@micro-epsilon.com
www.micro-epsilon.com

Contents

1. Safety

System operation assumes knowledge of the operating instructions.

1.1 Symbols Used

The following symbols are used in these operating instructions.

⚠ CAUTION	Indicates a hazardous situation which, if not avoided, may result in minor or moderate injury.
NOTICE	Indicates a situation that may result in property damage if not avoided.
➪	Indicates a user action.
i	Indicates a tip for users.
`Measurement`	Indicates hardware or a software button/menu.

1.2 Warnings

⚠ CAUTION

Connect the power supply and the display/output device according to the safety regulations for electrical equipment.
> Risk of injury

> Damage to or destruction of the interface module

NOTICE

The supply voltage must not exceed the specified limits.
> Damage to or destruction of the interface module

Avoid shocks and impacts to the interface module.
> Damage to or destruction of the interface module

1.3 Notes on CE Marking

The following apply to the IF2030/ENETIP interface module:

- EU Directive 2014/30/EU [1]
- EU Directive 2011/65/EU, “RoHS” Category 9

Products which carry the CE mark satisfy the requirements of the EU directives cited and the European harmonized standards (EN) listed therein. The measuring system is designed for use in industrial environments and meets the requirements.

The EU Declaration of Conformity is available to the responsible authorities according to EU Directive, article 10.

1.4 Intended Use

- The IF2030/ENETIP interface module is designed for use in industrial and laboratory applications. It is used to convert the internal MICRO-EPSILON sensor protocol (RS485, RS422) to EtherNet/IP.
- The IF2030/ENETIP must only be operated within the limits specified in the technical data, see Chap. 2.2.
- The IF2030/ENETIP must be used in such a way that no persons are endangered or machines and other material goods are damaged in the event of malfunction or total failure.
- Take additional precautions for safety and damage prevention in case of safety-related applications.

1.5 Proper Environment

- Protection class: IP20
- Temperature range
  - Operation: 0 ... +50 °C (+32 ... +122 °F)
  - Storage: -20 ... +70 °C (-4 ... +158 °F)
- Humidity: 5 - 95% (non-condensing)
- Ambient pressure: Atmospheric pressure

2. Functional Principle, Technical Data

2.1 Functional Principle

The IF2030/ENETIP interface module is used to convert the internal Micro-Epsilon sensor protocol (RS485 or RS422) to EtherNet/IP.

Features:

- LED status display
- EtherNet/IP interface
- Housing for top-hat rail

2.2 Technical Data

Model		**IF2030/ENETIP**
Power supply		+9 ... +36 V
Speed		1 ms refresh time, minimal
Power consumption (without sensor)		approx. 2.5 W @24 V
Inputs		RS485 (ME protocol) RS422 (ME protocol)
Outputs		EtherNet/IP, sync output, LED status display
Temperature range	operation	0 ... +50 °C (+32 ... +122 °F)
	storage	-20 ... +70 °C (-4 ... +158 °F)

RS485 interface	
Baud rate	9600 Baud ... 4 MBaud
Overvoltage protection	Up to ±60 V, ESD 15 kV
Terminating resistor	120 Ohm, integrated
Supported sensors, controller	ACC5703
	DT6120
	INC5701
	MSC7602

RS422 interface	
Baud rate	9600 Baud ... 4 MBaud
Overvoltage protection	-8 ... +13 V, ESD 15 kV
Terminating resistor	120 Ohm, integrated
Supported sensors, controller	ACS7000
	IFC24x1, IFC242x
	ILD1320, ILD1420, ILD1750, ILD1900, ILD2300
	ODC2520

Synchronization output	
Level	TTL or HTL
	No overvoltage protection

LED status display	System, status, MS, NS

3. Delivery

3.1 Unpacking, Included in Delivery

1 IF2030/ENETIP interface module

1 Operating Instructions

➡ Carefully remove the components of the interface module from the packaging and ensure that the goods are forwarded in such a way that no damage can occur.

➡ After unpacking, check immediately for completeness and transport damage.

➡ If there is damage or parts are missing, immediately contact the manufacturer or supplier.

3.2 Download

EDS file, available at https://www.micro-epsilon.de/service/download/

3.3 Storage

Temperature range storage:	-20 ... +70 °C (-4 ... +158 °F)
Humidity:	5 - 95% (non-condensing)

4. Installation and Assembly

Ensure careful handling during installation and operation.

4.1 Installation of the Interface Module

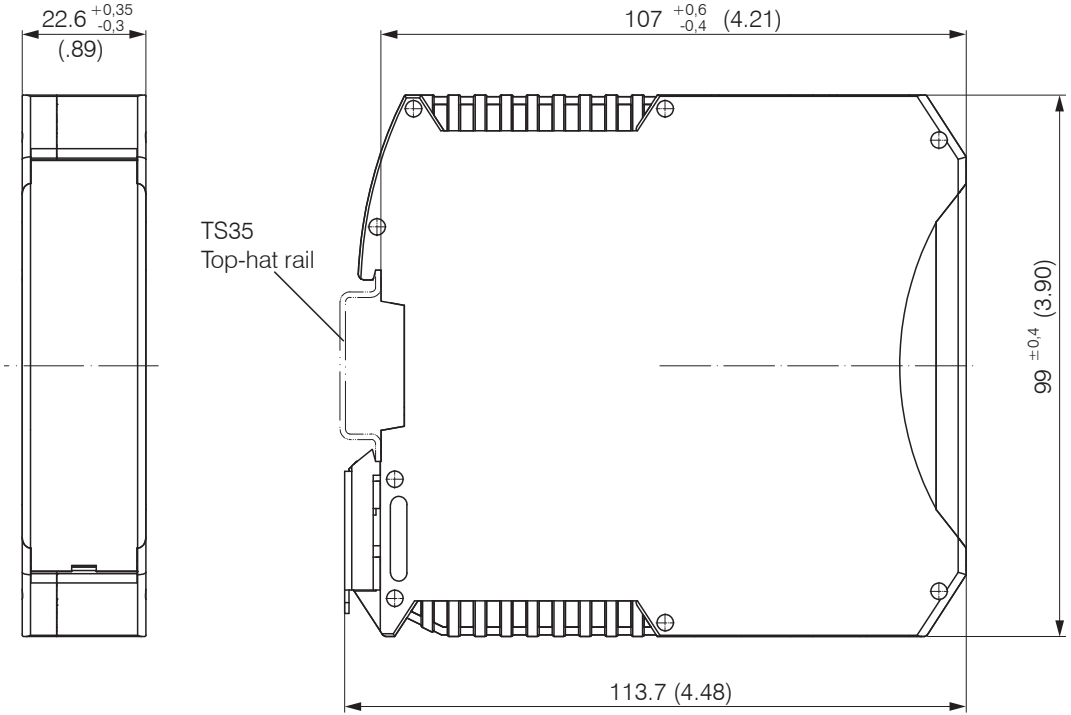
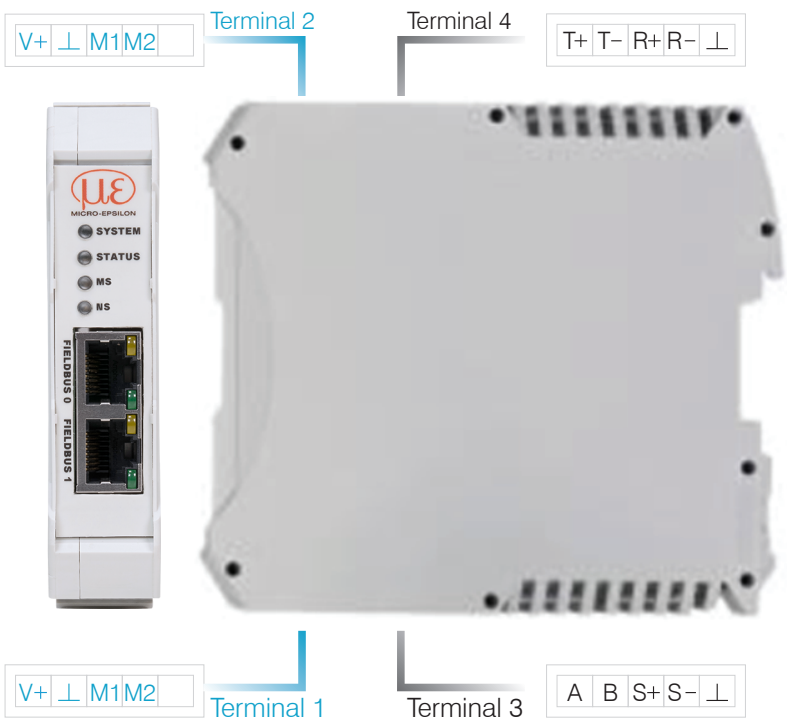


Fig. 1 IF2030/ENETIP dimensional drawing, dimensions in mm (inches)

4.2 Pin Assignment



Terminal 1 and 2	
V+	Supply voltage [2]
⊥	Ground for supply voltage
M1, M2	Multifunction input 1/2 sensor (e.g., for laser on/off)
Terminal 1 and 2 connections daisy-chained	

Terminal 4	
T+	RS422 Tx+
T-	RS422 Tx-
R+	RS422 Rx+
R-	RS422 Rx-
⊥	Ground [1] e.g., for RS422 shield connection

Terminal 3	
A	RS485 A
B	RS485 B
S+	Synchronization output +
S-	Synchronization output -
⊥	Ground [1] e.g., for RS485 shield connection

Fig. 2 Interface module terminals

1) Internally connected to supply ground

2) If the distance between IF2030/PNET and the sensor/controller is long, a separate supply for the sensor/controller may be advisable.

4.2.1 Supply Voltage

The supply voltage is daisy-chained from the supply port (terminal 1) to the sensor port (terminal 2), i.e., the supply voltage must match that of the sensor. Positive voltage must be between 9 V and 36 V.

➡ Connect the inputs V+ and ⊥ on terminal 1 to a voltage supply. Maximum cable length 3 m.

The voltage supply must match that of the connected sensor, because the voltage is internally daisy-chained.

MICRO-EPSILON recommends using the optionally available power supply PS2020,
input 100 - 240 VAC, output 24 VDC/2.5 A, see appendix.

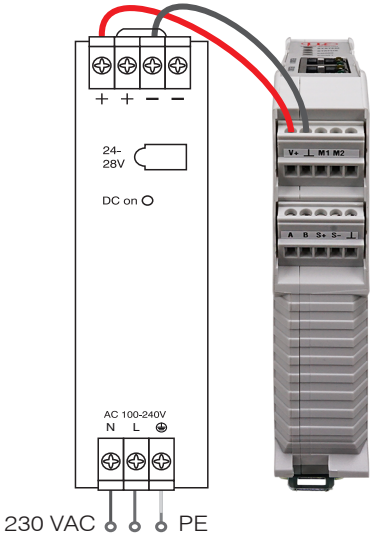


Fig. 3 Interface module with optional PS2020 power supply

i If the distance between IF2030/ENETIP and the connected sensor/controller is long, Micro-Epsilon recommends that a separate supply be used for the sensor/controller.

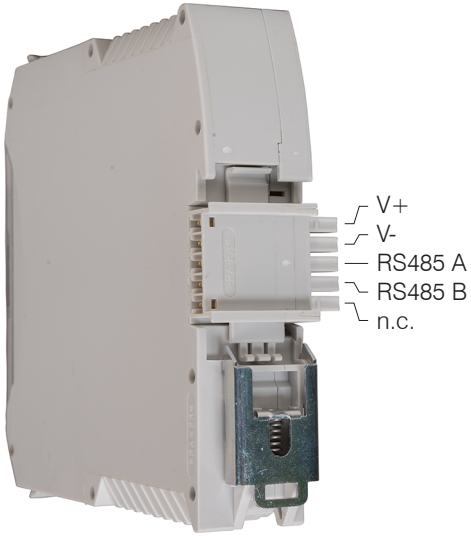
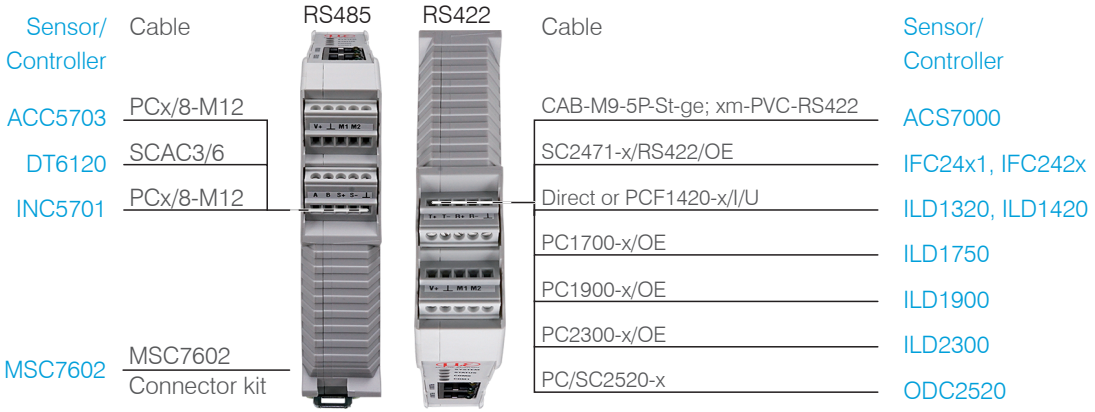


Fig. 4 Optional supply voltage wiring at rear of terminal

4.2.2 Connection Options



The length of the cable between IF2030/ENETIP and sensor/controller is 10 m at most. Because of the PCx/8-M12 cable, the sensor supply for ACC5703 and INC5701 sensors is possible only via the IF2030/ENETIP.

Fig. 5 Connection examples for IF2030/ENETIP

Fig. 6 Connection of an MSC7602 with MSC7602 connector kit

IF2030/ENETIP	Sensor/Controller
RS422	
T+	R+
T -	R -
R+	T+
R -	T -
⊥	Cable shield
RS485	
A	A
B	B
⊥	Cable shield

Fig. 7 Wiring regulation for connections with RS485 or RS422

4.2.4 Cable Termination at Interface

Ensure correct cable termination for an RS485 bus or RS422 bus!

We recommend a 120 Ohm terminating resistor between the signal lines at both the bus start and end. IF2030/ENETIP works as a master for both interfaces; internally, a 120 Ohm terminating resistor has already been permanently incorporated. The IF2030/ENETIP should be at the bus start.

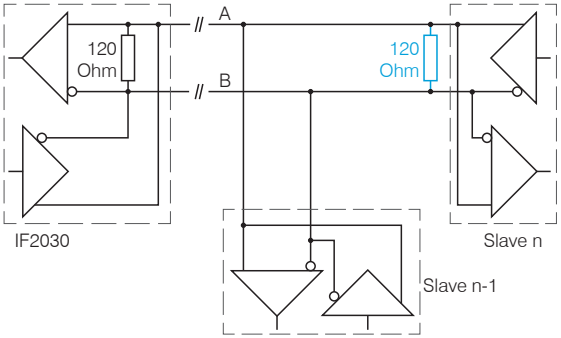


Fig. 8 Cable termination RS485, n = max. 16 slaves

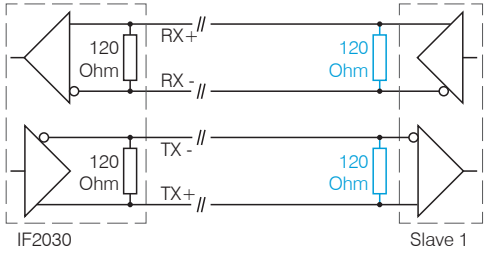


Fig. 9 Cable termination RS422

4.3 Fieldbus Cabling

During cabling, channel 0 of the scanner is connected to a port of adapter 1 (slave device).

The second port of the adapter 1 is connected to the port of the next adapter, etc. One port of the last adapter and channel 1 of the master device (scanner) remain unused.

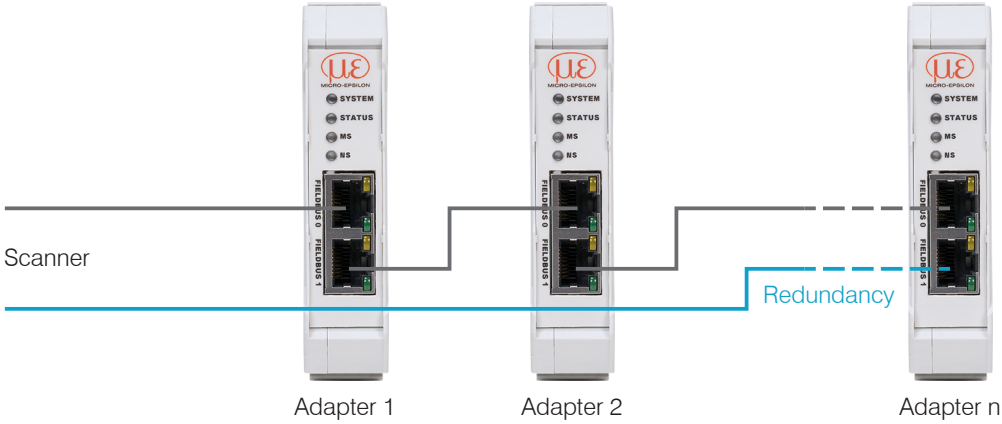


Fig. 10 Cabling in the EtherNet/IP network

Optional: IF2030/ENETIP can participate in a device level ring as a ring node and thereby reduce the threat of failures through redundant cabling.

5. Commissioning

5.1 Configuring the Sensors

The sensor used must be correctly configured to work with the IF2030. Micro-Epsilon recommends that the sensor's base configuration be set by using its web interface. The configuration can later also be adjusted via fieldbus.
Please refer to the operating instructions of the corresponding sensor for detailed information on configuring the sensor.

5.2 Baud Rate and Sensor Interface

IF2030/ENETIP must be set for the interface used and the sensor's baud rate.

Sensor/ Controller	Baud rate [Baud]	Bus address	RS485	RS422
ACC5703	230400	126	•	
ACS7000	230400			•
DT6120	230400	126	•	
IFC24x1	115200			•
IFC242x	115200			•
ILD1320	921600			•
ILD1420	921600			•

Sensor/ Controller	Baud rate [Baud]	Bus address	RS485	RS422
ILD1750	921600			•
ILD1900	921600			•
ILD2300	921600 [1]			•
INC5701	230400	126	•	
MSC7602	256000		•	
ODC2520	115200			•

Fig. 11 Baud rate (factory setting) of the sensors or controllers to be connected

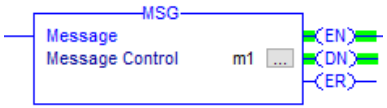
The baud rate and sensor interface are transferred with Class 0xA0 (Object), see Chap. 5.4.

1) The ILD2300 is set for 691.2 kBaud ex factory. Increase the baud rate to 921.6 kBaud in the sensor.

5.2.1 Option 1: MSG Command

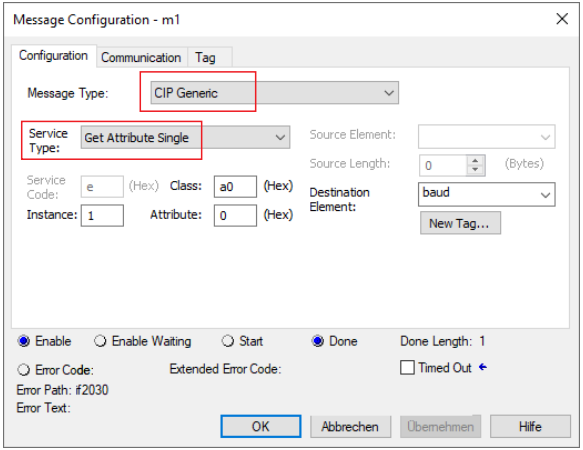
An MSG command is configured as follows:

➡ Click the button ... .

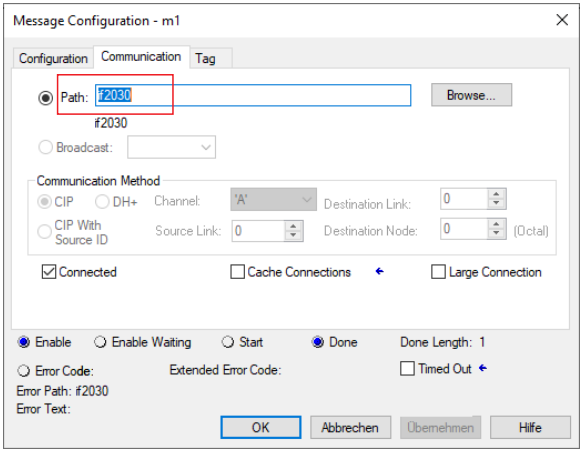


➡ Click the `Configuration` tab and set type `CIP Generic` as `Message Type`.

The following `Service Types` are possible:
- Get Attribute Single or
- Set Attribute Single.

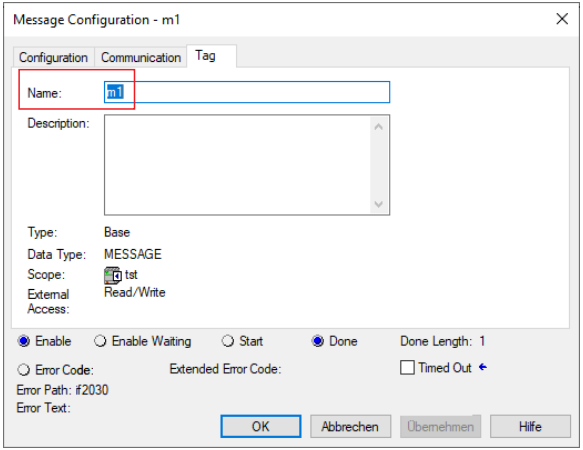


➡ Click the `Communication` tab and select the target device using the `Browse` button in the `Path` field.



➡ Click the `Tag` tab and assign an element name in the `Name` field.

Nothing needs to be set here. The Message Configuration dialog is only available if a tag of type `Message` had been previously entered in the MSG element. In the example above, `m1` was chosen for this purpose.



5.2.2 Option 2: External Software

IF2030/ENETIP can also be configured beyond the PLC (e.g., with a software tool) by using EtherNet/IP Explicit Messaging.

The software used for this purpose must support the following services:

- 0x0E – Get Attribute Single as well as
- 0x10 – Set Attribute Single.

5.3 Data Format

All configuration parameters and data are transmitted in Little Endian format.

Sensors/controllers with RS422: cyclical data are decoded, i.e., a 4th byte is added to the 3 bytes and then transmitted. The sensor signals selected for transfer and their sequence are available on the sensor's web interface.

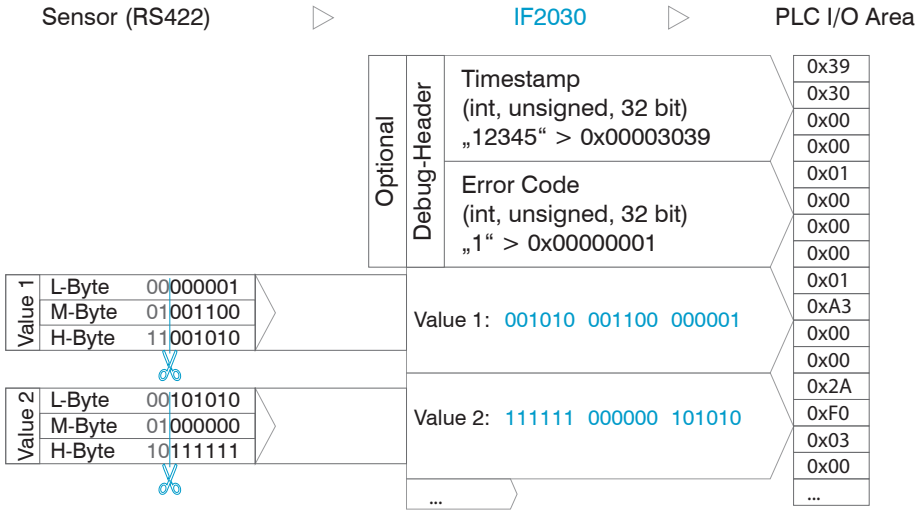


Fig. 12 Interpretation of RS422 sensor data in IF2030/ENETIP

Sensors/controllers with RS485: cyclical data are transmitted via the fieldbus without change, i.e., as a binary block as described and supplied by the sensor. Please refer to the sensor's operating instructions for the data set structure.

5.4 Class Directory

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x80	1	0	Uint8	RW	Select sensor	✓		Address of currently selected sensor
	1	1	Uint8[32]	R	Sensor addresses	✓		Shows address list of available sensors
0x90	0		Uint32[64]	R	Device error log	✓	✓	Reads out the last 32 error codes with time stamp
0xA0	1	0	Uint32	RW	Baudrate	✓	✓	IF2030 baud rate
		1	Uint8	RW	Minimum cycle time	✓	✓	Minimum time for one communication cycle in ms, cycle time = 0: use estimated time
		2	Uint8	RW	Preferred sensor data size	✓	✓	Desired useful data length in bytes per sensor, value = 0: estimated/calculated value
		3	Uint8	RW	Sensor interface	✓	✓	0: RS485, 1: Reserved, 2: ASCII + RS422
		4	Uint8	W	Reset device config	✓	✓	One byte deletes settings from flash, settings are included in RAM until restart
		5	Uint8	W	Reset sensor config	✓		One byte deletes settings from flash, settings are included in RAM until restart
		6	Uint8	W	Reset device	✓	✓	One byte performs reset
		7	Uint8	RW	enable/disable HTTL Sync	✓	✓	0: Disable HTTL synchronization 1: Enable HTTL synchronization
		8	Uint8	RW	enable/disable cyclic status header	✓	✓	0: Disable 8 byte status header in cyclic data 1: Enable 8 byte status header in cyclic data

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0310	1				Device Info	✓		Read out the block of the current sensor
		0	Uint8	R	Number of objects			
		1	Uint8	R	Block version			Block version
		2	Uint8	R	Endianness			Endian
		3	Uint16	R	Software version			Software version
		4	Int32	R	Article number			Part number
		5	Int32	R	Option			Option
		6	Int32	R	Batch number			Batch number
		7	Int32	R	Serial number			Serial number
		8	Uint8	R	Change index			Change index
		9	Uint8	R	Calibration day			Day of calibration
		10	Uint8	R	Calibration month			Month of calibration
		11	Uint8	R	Calibration year			Year of calibration
		12	Uint16	R	Calibration software version			Version of calibration software
		13	Uint16	R	Test software version			
		14	Uint8	R	Test hour			
		15	Uint8	R	Test day			
		16	Uint8	R	Test month			
		17	Uint8	R	Test year			
		18	Int32	R	Article number circuit board			
		19	Int32	R	Serial number circuit board			
		20	Uint8[32]	R	Name			
		21	Uint8	R	sensor/channel count			
		22	Uint8	R	protocol block count			
		23	Uint8[164]	R	protocol blocks			

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0313	1				Diagnostic block	✓		RS485 bus diagnostic block (if available)
		0	Uint8		Number of objects			
		1	Uint8	RW	Page index to read			Specifying an index lets you scroll through existing pages
		2	Uint8	R	Number of pages			
		3	Uint8	R	Diagnose Type			
		4	Uint8[235]	R	String Page			Diagnostic message

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0320	1				Sensor block	✓		Request sensor information
		0	Uint8	R	Number of objects			
		1	Uint8	RW	block index offset			The offset lets you scroll through existing sensor blocks [0 - 0x1F]
		2	Uint8	RW	page index to read			Specifying an index lets you scroll through existing pages
		3	Uint8	R	number of pages			Max. number of pages
		4	Uint8	R	measurement unit			Signal unit
		5	Int32	R	article number			Part number
		6	Int32	R	Option			Option
		7	Int32	R	Batch number			Batch number
		8	Int32	R	serial number			Serial number
		9	Float	R	Nominal measuring range			Nominal measuring range
		10	Float	R	Nominal offset			Nominal offset
		11	Float	R	current measuring range			Actual measuring range
		12	Float	R	current offset			Actual offset
		13	Uint8[32]	R	Target material			Target material
		14	Uint8[32]	R	Sensor/channel name			Sensor/channel name
		15	uint8	R	extension length			Length of block extension
		16	uint8[138]	R	extension			

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0390	1				Parameter Info	✓		Request configuration parameters, e.g., sensor exposure time, request via subindex 1, configure interface with Class 0x2510 through 0x2540
		0	Uint8	R	Number of objects			
		1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
		2	Uint8[14]	R	Name			
		3	Uint8[8]	R	Unit			
		4	Uint8[8]	R	Type			

0x0410	1				Float parameter	✓		Read or write float parameter
		0	Uint8		Number of objects			
		1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
		2	Uint8	RW	Reserved			
		3	Float	RW	Value			Value
		4	Uint8[14]	R	Name			Designation
		5	Uint8[8]	R	Unit			Unit as a string
		6	Float	R	Min			
		7	Float	R	Max			

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0411	1				Int Parameter	✓		Read or write integer parameter
		0	Uint8		NrOfObjects			
		1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
		2	Uint8	RW	Reserved			
		3	Int32	RW	Value			Value
		4	Uint8[14]	R	Name			Designation
		5	Uint8[8]	R	Unit			Unit as a string
		6	Int32	R	Min			
		7	Int32	R	Max			

0x0412	1				Uint Parameter	✓		Read or write unsigned integer parameter
		0	Uint8		NrOfObjects			
		1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
		2	Uint8	RW	Reserved			
		3	Uint32	RW	Value			Value
		4	Uint8[14]	R	Name			Designation
		5	Uint8[8]	R	Unit			Unit as a string
		6	Uint32	R	Min			
		7	Uint32	R	Max			

Class	Instance	Attribute	Data type		Name	RS485	RS422	Description
0x0413					String Parameter	✓		Read or write string parameter
	0		Uint8		NrOfObjects			
	1		Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2		Uint8	RW	Reserved			
	3		Uint8[246]	RW	Value			Value
	4		Uint8[14]	R	Name			Designation

0x0430					RS422 ASCII Access		✓	RS422 commando
	1		Uint8[128]	RW	Send Cmd			Buffer for a 128-character ASCII command, termination with '\n' or 0x0A
	2		Uint8[896]	R	Cmd Answer			Answer from sensor without shortening, e.g., Line feed; if buffer overflows, e.g., PRINT ALL, answer is truncated

If time synchronization has been enabled on the scanner (SPS), a synchronized signal can be tapped at the S+/S- terminal. Signal timing can be configured by using Class 0x43, Instance 1, Attribute 300.

	Parameter	**Unit**	**Default**	**Min**	**Max**	**Comments**
Byte 0 - 3	Sync Intervall	ns	500,000,000	10,000	500,000,000	Synchronization interval 10 μs ... 500 ms
Byte 4 - 7	Sync Offset	ns	0	0	Sync Interval -1	
Byte 8 - 11	Res. Sync Interval	ns	500,000,000	10,000	500,000,000	
Byte 11 - 15	Res. Sync Offset	ns	150	0	Res. Sync Interval - 1	$\lvert$Sync Offset – Res. Sync Offset$\rvert$ > 150
Byte 16 - 19	Pulse Length	μs	4	1	500	Pulse length < min (SyncOffset, Res.Sync Offset)

Changes only take effect after IF2030/ENETIP has been restarted.
Please use Class 0xA0, Instance 1, Attribute 7 to configure the level (TTL/HTL).

6. Liability for Material Defects

All components of the device have been checked and tested for functionality at the factory. However, if defects occur despite our careful quality control, MICRO-EPSILON or your dealer must be notified immediately.

The liability for material defects is 12 months from delivery. Within this period, defective parts, except for wearing parts, will be repaired or replaced free of charge, if the device is returned to MICRO-EPSILON with shipping costs prepaid. Any damage that is caused by improper handling, the use of force or by repairs or modifications by third parties is not covered by the liability for material defects. Repairs are carried out exclusively by MICRO-EPSILON.

Further claims can not be made. Claims arising from the purchase contract remain unaffected. In particular, MICRO-EPSILON shall not be liable for any consequential, special, indirect or incidental damage. In the interest of further development, we reserve the right to make design changes without notification.

For translations into other languages, the German version shall prevail.

7. Service, Repair

Please send us the affected parts for repair or exchange.

If the cause of a fault cannot be clearly identified, please send the entire measuring system to:

MICRO-EPSILON MESSTECHNIK
GmbH & Co. KG
Koenigbacher Str. 15
94496 Ortenburg / Germany

Telephone: +49/8542/168 - 0
Fax: +49/8542/168 - 90
info@micro-epsilon.com
www.micro-epsilon.com

8. Decommissioning, Disposal

➡ Remove all cables from the interface module.

Incorrect disposal may cause harm to the environment.

➡ Dispose of the device, its components and accessories as well as the packaging materials in compliance with the applicable country-specific waste treatment and disposal regulations of the region of use.

Appendix

A 1 Optional Accessories

PS2020

Power supply; installation of top-hat rail, 2.5 A, input 100 - 240 VAC, output 24 VDC/2.5 A, installation type; installation on symmetrical standard rail 35 mm x 7.5 mm, DIN 50022

A 2 Factory Settings

Baud rate	921600 Baud
cycleMinTime	0 (= IF2030 calculates cycle time)
SensorInterface	MEO+RS422
HTTL	OFF
CyclicDebugHeader	OFF

A 3 Sensor Values, Data Format, Conversion

A 3.1 General

The sensors or controllers do not solely output distance values. The overview below describes the conversion during output of distance values. Please refer to the corresponding operating instructions for detailed information on conversion when additional values are output.

A 3.2 ACC5703

Baud rate 230400 b/s RS485 half duplex Max. sampling rate 1 kHz: measurements with variable number ex factory scaled to ±2 g, Little Endian

Bus address 126

Byte Data	Meaning	Data format
Data[0]	Status byte (contains error flags, normally 0x00)	8 bit
Data[1] ... Data[4]	Measured value counter [bit 0:31]	Uint 32 bit
Data[5]	Number of measured values in this package = 3*x mit x [1 ... 19]	8 bit
Data[6]	Padding-Byte	8 bit
Data[7]	Padding-Byte	8 bit
Data[8]	Measuring value 1 x-axis [bit 0:7]	Float 32 bit
Data[9]	Measuring value 1 x-axis [bit 8:15]	
Data[10]	Measuring value 1 x-axis [bit 16:23]	
Data[11]	Measuring value 1 x-axis [bit 24:31]	
...	...	
Data[n] n=8+(4*Data [5]/3)	Measuring value 1 x-axis [bit 0:7]	Float 32 bit
Data[n+1]	Measuring value 1 x-axis [bit 8:15]	
Data[n+2]	Measuring value 1 x-axis [bit 16:23]	
Data[n+3]	Measuring value 1 x-axis [bit 24:31]	
...	...	...
Data[n+m] m=4*Data[5]/3	Measuring value 1 z-axis [bit 0:7]	Float 32 bit
Data[n+m+1]	Measuring value 1 z-axis [bit 8:15]	
Data[n+m+2]	Measuring value 1 z-axis [bit 16:23]	
Data[n+m+2]	Measuring value 1 z-axis [bit 24:31]	

Please refer to the operating instructions for the acceleration sensor for more information.

The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--inertialSENSOR-ACC5703--en.pdf

Fig. 13 Encoding of Measured Data in the Transmission Protocol, ACC5703

A 3.3 ACS7000

RS422	Measuring rate 250 Hz ex factory, all color values and color distances. Up to 32 output values can be transmitted at the same time.
Baud rate	115200 b/s

ACS7000 supplies 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

Group	Name	Index	Raw		Scaled			
			Min	Max	Min	Max	Formula	Unit
Status	Framerate	1	2500	250000	20,00	2000,00	10 ^ 6/(x*12,5*2 ^ 4)*1000	Hz
	Shutter	2	2500	250000	20,00	2000,00	x*12,5*2 ^ 4)/10 ^ 9	µs
	TempDetector	3	-1024	1023	-256,00	255,75	x/4	°C
	TempLightSrc	4	-1024	1023	-256,00	255,75	x/4	°C
LightSensor	Red	5	0	65535	0,00	100,00	x/65536*100	%
	Green	6	0	65535	0,00	100,00	x/65536*100	%
	Blue	7	0	65535	0,00	100,00	x/65536*100	%
	Brightness	8	0	65535	0,00	100,00	x/65536*100	%
Status	Counter	9	0	262143	0	262143	x	-
	Timestamp	10	0	262143	0,00	67,11	x*256/100000	s
Color	XYZ	11-13	0	131072	0,00	256,00	x/512	-
	RGB	14-16	0	131072	0,00	256,00	x/512	-
	LAB	17-19	-131072	131071	-256,00	256,00	x/512	-
	LUV	20-22	-131072	131071	-256,00	256,00	x/512	-
	LCH (L/C)	23-24	-131072	131071	-256,00	256,00	x/512	-
	LCH (H)	25	0	131071	0,00	256,00	x/512	°
	LAB99	26-28	-131072	131071	-256,00	256,00	x/512	-
	LCH99 (L/C)	29-30	-131072	131071	-256,00	256,00	x/512	-
	LCH99 (H)	31	0	184320	0,00	360,00	x/512	°

Group	Name	Index	Raw		Scaled			
			Min	Max	Min	Max	Formula	Unit
Status	Error	32	0	262143	0	262143	x	-
Distance	1_1/2/3	33-35	NA	-				
	…	36-77						
	16_1/2/3	78-80		-				
	Min_1/2/3	81-83	-131072	131071	-256,00	256,00	x/512	-
	DetectedID	84	0	16	0	16	-	-
	MinDistID	85	0	16	0	16	-	-

Fig. 14 Overview of output data via RS422

Please refer to the operating instructions for the color measuring system colorCONTROL ACS7000 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.com/download/manuals/man--colorCONTROL-ACS7000--en.pdf

A 3.4 DT6120

Baud rate	230400 b/s	RS485 half duplex	Measurements ex factory scaled to sensor measuring range, Little Endian
Bus address	126		

Measuring data consist of a counter, the packet length m and the measurements. The packet length m determines how many measurements are transmitted. The packet length m is the number of measurements that have been queried by the electronic system since the last time measuring data were queried, but is limited to the most recent 20 measurements. The first measurement in the data [] package is the oldest value queried, the last one is the most recently queried value.

Byte Data	Meaning	Data format
Data[0]	Counter [7:0]	unsigned short
Data[1]	Counter [15:8]	
Data[2]	Packet length m [7:0]	unsigned char
Data[3]	Filler byte [7:0]	unsigned char
Data[4]	Measuring value 1 [7:0]	signed integer
Data[5]	Measuring value 1 [15:8]	
Data[6]	Measuring value 1 [23:16]	
Data[7]	Measuring value 1 [31:24]	
	. . .	
Data[..]	Measuring value m [7:0]	signed integer
Data[..]	Measuring value m [15:8]	
Data[..]	Measuring value m [23:16]	
Data[..]	Measuring value m [31:24]	

Scaling of measurements

BY default, 24-bit measurements are transmitted.
The following equivalences therfore apply:

0x0 = 0 % of the sensor measuring range

0xF00000 = 100 % of the sensor measuring range

If the sensor is outside the measuring range, accordingly larger measurements are output.

Fig. 15 Encoding of Measured Data in the Transmission Protocol, DT6120

Please refer to the operating instructions for the capacitive displacement measuring system for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--capaNCDT-6110-6120IP--en.pdf

A 3.5 IFC2421, IFC2422, IFC2451, IFC2461, IFC2471

RS422	Up to 32 output values can be transmitted at the same time. The data are configured or selected via ASCII commands or via the web interface.
Baud rate	115200 b/s ex factory

Ex factory, the controller is set for the `Distance measurement` measuring program. Please refer to the associated operating instructions for descriptions of additional measuring programs. IFC24xx supplies 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

$$x = \frac{(d_{OUT} - 98232) * MR}{65536}$$

x	=	Displacement / Thickness in mm
dOUT	=	digital output value
MR	=	Measuring range in mm
131000	=	Midrange for the displacement measurement

Please refer to the operating instructions for the confocal distance measuring system
- confocalDT 2421/2422
- confocalDT 2451/2461/2471

for more information, especially about possible output values.

The current version is available at:

https://www.micro-epsilon.com/download/manuals/man--confocalDT-2421-2422--en.pdf

https://www.micro-epsilon.de/download/manuals/man--confocalDT-2451-2461-2471--en.pdf

A 3.6 ILD1320, ILD1420

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; <643] SMR reserve [643; 64877] Measuring range [>64877; 65520] EMR reserve	$d\ [mm] = \frac{1}{100}\left(\frac{102}{65520}\ x - 1\right) * MR\ [mm]$
	MR = measuring range [mm]	{10/25/50/100/200/500}	
	d = distance [mm]	[-0,01MR; 1,01MR]	

Fig. 16 Calculation of distance value from the digital value, ILD1320/1420

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1320/1420 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1320--en.pdf

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1420--en.pdf

1) Distance value without the Master function.

A 3.7 ILD1750

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; 230604]	$d\ [mm] = \frac{x - 98232}{65536} * MR\ [mm]$
	MR = measuring range [mm]	{2/10/20/50/100/200/500/750}	
	d = distance [mm]	without Mastern [-0,01 MR; 1,01MR]	
		with Mastern [-2MR; 2MR]	

Fig. 17 Calculation of distance value from the digital value, ILD1750

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1750 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1750--en.pdf

A 3.8 ILD1900

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; 230604]	$d\ [mm] = \frac{x - 98232}{65536} * MR\ [mm]$
	MR = measuring range [mm]	{2/10/25/50/100/200/500}	
	d = distance [mm]	without Mastern [-0,01 MR; 1,01MR]	
		with Mastern [-2MR; 2MR]	

Fig. 18 Calculation of distance value from the digital value, ILD1900

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1900 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1900--en.pdf

A 3.9 ILD2300

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	691200 baud ex factory [1]

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). 16 Bit per value are transmitted. The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in mm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; <643] SMR reserve [643; 64877] Measuring range [>64877; 65520] EMR reserve	$d\ [mm] = \frac{1}{100}\left(\frac{102}{65520}\,x - 1\right) * MR\ [mm]$
	MR = measuring range [mm]	{10/25/50/100/200/500}	
	d = distance [mm]	[-0,01MR; 1,01MR]	

Fig. 19 Calculation of distance value from the digital value, ILD2300

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 2300 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-2300--en.pdf

1) When delivered, ILD2300 is set for 691.2 kBaud. Increase the baud rate to 921.6 kBaud in the sensor.

A 3.10 INC5701

Baud rate 230400 b/s RS485 half duplex max. sampling rate 250 Hz, ex factory INC5701D, Little Endian
Bus address 126

Byte	Meaning	Data format
Data[0]	Status byte (contains error flags, normaly 0x00)	8 bit
Data[1]	Long term values counter [bit 0:7]	Uint 32 bit
Data[2]	Long term values counter [bit 8:15]	
Data[3]	Long term values counter [bit 16:23]	
Data[4]	Long term values counter [bit 24:31]	
Data[5]	Number of measured values in this package	8 bit
Data[6]	Padding byte	8 bit
Data[7]	Padding byte	8 bit
Data[8]	Measured value 1 [bit 0:7]	Float 32 bit
Data[9]	Measured value 1 [bit 8:15]	
Data[10]	Measured value 1 [bit 16:23]	
Data[11]	Measured value 1 [bit 24:31]	
Data[12]	Measured value 2 [bit 0:7]	Float 32 bit
Data[13]	Measured value 2 [bit 8:15]	
Data[14]	Measured value 2 [bit 16:23]	
Data[15]	Measured value 2 [bit 24:31]	

Fig. 20 Encoding of Measured Data in the Transmission Protocol, INC5701S

Byte Data	Meaning	Data format
Data[0]	Status byte (contains error flags, normally 0x00)	8 bit
Data[1] ... Data[4]	Measured value counter [bit 0:31]	Uint 32 bit
Data[5]	Number of measured values in this package	8 bit
Data[6], Data[7]	Padding-Byte	8 bit
Data[8]	Measuring value 1 LP [1] [bit 0:7]	Float 32 bit
Data[9]	Measuring value 1 LP [1] [bit 8:15]	
Data[10]	Measuring value 1 LP [1] [bit 16:23]	
Data[11]	Measuring value 1 LP [1] [bit 24:31]	
Data[12]	Measuring value 2 LP [1] [bit 0:7]	
Data[13]	Measuring value 2 LP [1] [bit 8:15]	
Data[14]	Measuring value 2 LP [1] [bit 16:23]	
Data[15]	Measuring value 2 LP [1] [bit 24:31]	
...	...	...
Data[n] n=8+(4*Data [5])	Measuring value 2 SF [2] [bit 0:7]	Float 32 bit
Data[n + 1]	Measuring value 2 SF [2] [bit 8:15]	
Data[n + 2]	Measuring value 2 SF [2] [bit 16:23]	
Data[n + 3]	Measuring value 2 SF [2] [bit 24:31]	
Data[n + 4]	Measuring value 2 SF [2] [bit 24:31]	
Data[n + 5]	Measuring value 2 SF [2] [bit 24:31]	
...	...	

Fig. 21 Encoding of Measured Data in the Transmission Protocol, INC5701D

1) LP = Low pass filter 2) SF = SensorFUSION filter

Please refer to the operating instructions for the inclination sensor for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--inertial-SENSOR-INC5701--de.pdf

The measurement data consits of one status byte, measured values counter, number of measured values, and the measured data. The measured values counter increases continuously with each sampled value. It represents the number of measured values transmitted in this package (floats). The first measurement value in the Data [] package is the oldest measured value. A measured value is represented as 4-byte float data type in the unit angular degrees [°].

A 3.11 MSC7602

Baud rate 256000 baud ex factory, [9600 ... 256000] RS485 half duplex Measurements ex factory scaled to analog value, Little Endian

Bus address 126 [2 ... 126]

Sequence for a measurement value request:

Send	0x10	0x7E [1]	0x01 [2]	0x4C	0xCB [3]	0x16									
Receive	0x68	0x0B	0x0B	0x68	0x01 [2]	0x7E [1]	0x08	0xAE	0x47	0x61	0x3F	0x00	0x00	0x00	0x00
	0x1C [4]	0x16													

Result	Description	Format	Example
	Unscaled value	Bytes 8 - 11: 4 Bytes, float, Little-Endian	0x3F6147AE (float) = 0.88 V
	Scaled value	Bytes 12 - 15: 4 Bytes, float, Little-Endian	If this value is 0, the controller was not set up. Otherwise, the digital counterpart of the analog output will be sent according the setting you have done in the controller before.
	Maximum speed for data transmission (1x send + 1x receive): ~3 ms @ 256.000 Baud		

1) DA: 126
2) SA: 1
3) CH: Checksum Send: Bytes 2 - 4
4) CH: Checksum Receive: Bytes 5 - 15

Fig. 22 Encoding of Measured Data in the Transmission Protocol, MSC7602

Please refer to the operating instructions for the inductive displacement measuring system for more information.
The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--induSENSOR-7xxx--en.pdf

A 3.12 ODC2520

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	115200 Baud ex factory

Ex factory, the controller outputs the measurements in the `Edge light-dark` measuring program to the web diagram, i.e., output must be redirected to the RS422 interface.
The ODC2520 supplies 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.
The linearized measuring values can be converted in µm using the following formula:

$$x\,[\mu m] = d_{out} - 131000$$

x = Measuring value (edge position, difference, center axis) in µm

d_{out} = digital output value; $d_{out} \geq 262072$ are error values

Fig. 23 Calculation of edge position from the digital value, ODC2520

Please refer to the operating instructions for the laser micrometer optoCONTROL 2520 for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoCONTROL-2520--en.pdf

MICRO-EPSILON

Operating Instructions

IF2030/PNET

Interface Module

MICRO-EPSILON
MESSTECHNIK
GmbH & Co. KG
Koenigbacher Str. 15

94496 Ortenburg / Germany

Tel. +49 (0) 8542/168-0
Fax +49 (0) 8542/168-90
email info@micro-epsilon.com
www.micro-epsilon.com

Contents

1. Safety

System operation assumes knowledge of the operating instructions.

1.1 Symbols Used

The following symbols are used in these operating instructions.

⚠ CAUTION	Indicates a hazardous situation which, if not avoided, may result in minor or moderate injury.
NOTICE	Indicates a situation that may result in property damage if not avoided.
➔	Indicates a user action.
i	Indicates a tip for users.
`Measurement`	Indicates hardware or a software button/menu.

1.2 Warnings

⚠ CAUTION

Connect the power supply and the display/output device according to the safety regulations for electrical equipment.
> Risk of injury

> Damage to or destruction of the interface module

NOTICE

The supply voltage must not exceed the specified limits.
> Damage to or destruction of the interface module

Avoid shocks and impacts to the interface module.
> Damage to or destruction of the interface module

1.3 Notes on CE Marking

The following apply to the IF2030/PNET interface module:

- EU Directive 2014/30/EU [1]
- EU Directive 2011/65/EU, “RoHS” Category 9

Products which carry the CE mark satisfy the requirements of the EU directives cited and the European harmonized standards (EN) listed therein. The measuring system is designed for use in industrial environments and meets the requirements.

The EU Declaration of Conformity is available to the responsible authorities according to EU Directive, article 10.

1.4 Intended Use

- The IF2030/PNET interface module is designed for use in industrial and laboratory applications.
  It is used to convert the internal MICRO-EPSILON sensor protocol (RS485, RS422) to PROFINET.
- The IF2030/PNET must only be operated within the limits specified in the technical data, see Chap. 2.2.
- The IF2030/PNET must be used in such a way that no persons are endangered or machines and other material goods are damaged in the event of malfunction or total failure.
- Take additional precautions for safety and damage prevention in case of safety-related applications.

1.5 Proper Environment

- Protection class: IP 20
- Temperature range
  - Operation: 0 ... +50 °C (+32 ... +122 °F)
  - Storage: -20 ... +70 °C (-4 ... +158 °F)
- Humidity: 5 - 95% (non-condensing)
- Ambient pressure: Atmospheric pressure

2. Functional Principle, Technical Data

2.1 Functional Principle

The IF2030/PNET interface module is used to convert the internal Micro-Epsilon sensor protocol (RS485 or RS422) to PROFINET IO.

Features:

- Synchronization output, LED status display
- PROFINET interface
- Housing for top-hat rail

2.2 Technical Data

Model		**IF2030/PNET**
Power supply		+9 ... +36 V
Speed		1 ms (refresh time PROFINET)
Power consumption (without sensor)		approx. 2.5 W @24 V
Inputs		RS485 (ME protocol) RS422 (ME protocol)
Outputs		PROFINET, sync output, LED status display
Temperature range	operation	0 ... +50 °C (+32 ... +122 °F)
	storage	-20 ... +70 °C (-4 ... +158 °F)

RS485 interface	
Baud rate	9600 Baud ... 4 MBaud
Overvoltage protection	Up to ±60 V, ESD 15 kV
Terminating resistor	120 Ohm, integrated
Supported sensors, controller	ACC5703
	DT6120
	INC5701
	MSC7602

RS422 interface	
Baud rate	9600 Baud ... 4 MBaud
Overvoltage protection	-8 ... +13 V, ESD 15 kV
Terminating resistor	120 Ohm, integrated
Supported sensors, controller	ACS7000
	IFC24x1, IFC242x
	ILD1320, ILD1420, ILD1750, ILD1900, ILD2300
	ODC2520

Synchronization output	
Level	TTL or HTL (PROFINET IRT only)
	No overvoltage protection

LED status display	System, status, COM0, COM1

3. Delivery

3.1 Unpacking, Included in Delivery

1 IF2030/PNET interface module

1 Operating Instructions

- Carefully remove the components of the interface module from the packaging and ensure that the goods are forwarded in such a way that no damage can occur.
- After unpacking, check immediately for completeness and transport damage.
- If there is damage or parts are missing, immediately contact the manufacturer or supplier.

3.2 Download

GSDML file, available at https://www.micro-epsilon.de/service/download/

TIA function components for easier configuration, available at https://www.micro-epsilon.de/service/download/

3.3 Storage

Storage temperature:	-20 ... +70 °C (-4 ... +158 °F)
Humidity:	5 - 95% (non-condensing)

4. Installation and Assembly

Ensure careful handling during installation and operation.

4.1 Installation of the Interface Module

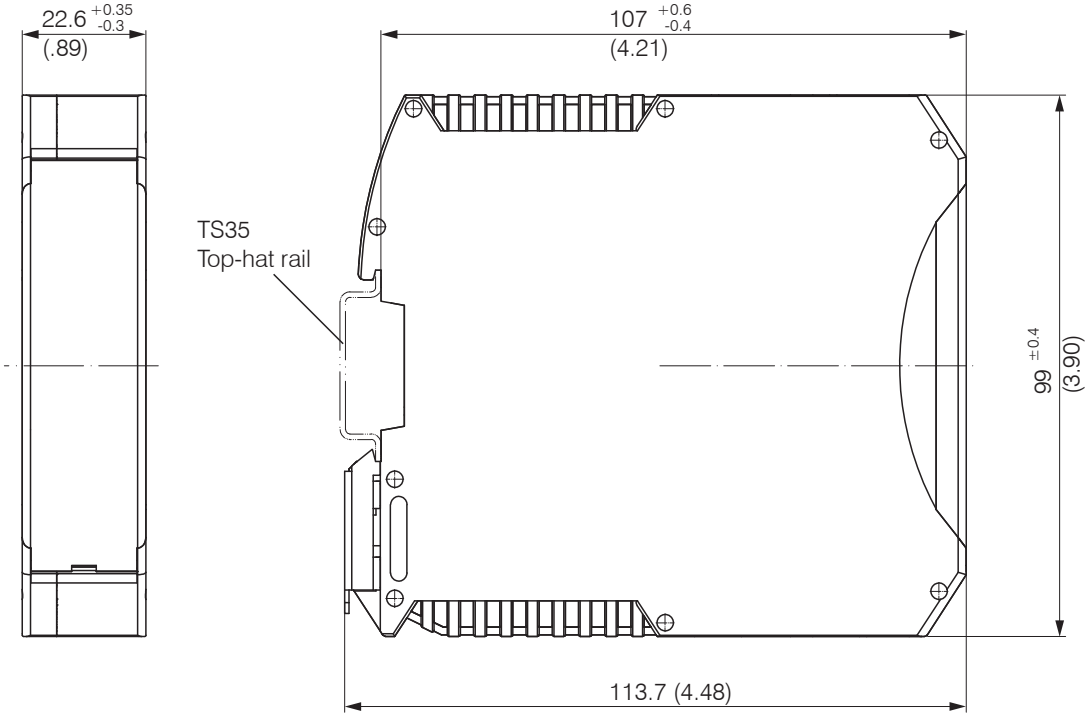
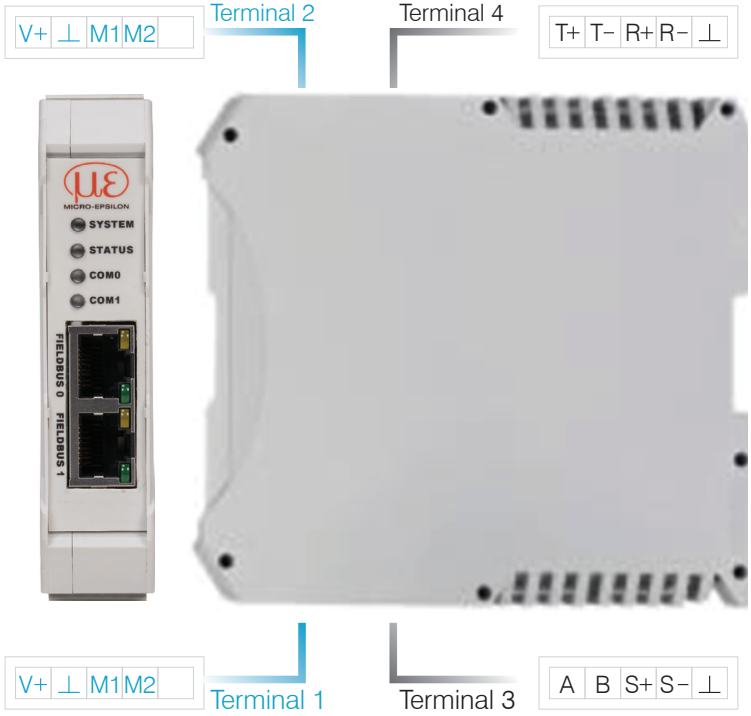


Fig. 1 IF2030/PNET dimensional drawing, dimensions in mm (inches)

4.2 Pin Assignment



Terminal 1 and 2	
V+	Supply voltage [2]
⊥	Ground for supply voltage
M1, M2	Multifunction input 1/2 sensor (e.g., for laser on/off)
Terminal 1 and 2 connections daisy-chained	

Terminal 4	
T+	RS422 Tx+
T-	RS422 Tx-
R+	RS422 Rx+
R-	RS422 Rx-
⊥	Ground [1] e.g., for RS422 shield connection

Terminal 3	
A	RS485 A
B	RS485 B
S+	Synchronization output +
S-	Synchronization output -
⊥	Ground [1] e.g., for RS485 shield connection

Fig. 2 Interface module terminals

1) Internally connected to supply ground

2) If the distance between IF2030/PNET and the sensor/controller is long, a separate supply for the sensor/controller may be advisable.

4.2.1 Supply Voltage

The supply voltage is daisy-chained from the supply port (terminal 1) to the sensor port (terminal 2), i.e., the supply voltage must match that of the sensor. Positive voltage must be between 9 V and 36 V.

➡ Connect the inputs V+ and ⊥ on terminal 1 to a voltage supply. Maximum cable length 3 m.

The voltage supply must match that of the connected sensor, because the voltage is internally daisy-chained.

MICRO-EPSILON recommends using the optionally available power supply PS2020,
input 100 - 240 VAC, output 24 VDC/2.5 A, see appendix.

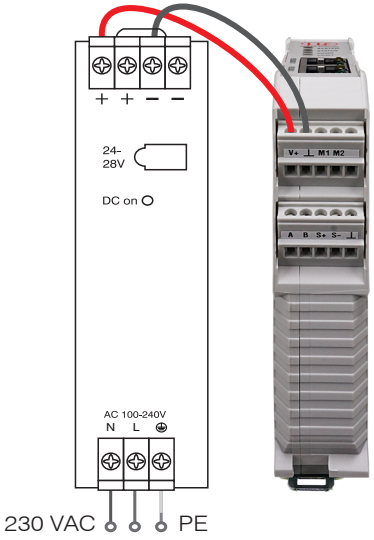


Fig. 3 Interface module with optional PS2020 power supply

i If the distance between IF2030/PNET and the connected sensor/controller is long, Micro-Epsilon recommends that a separate supply be used for the sensor/controller.

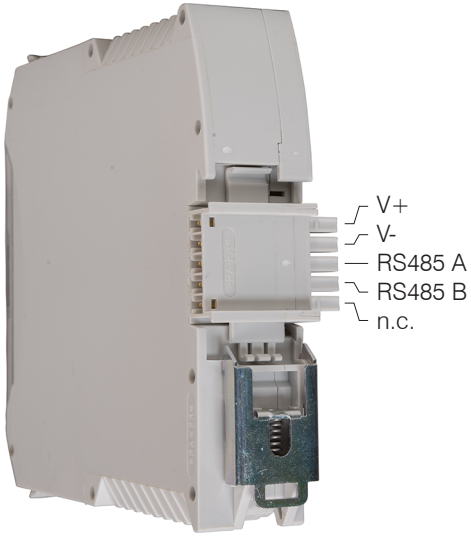
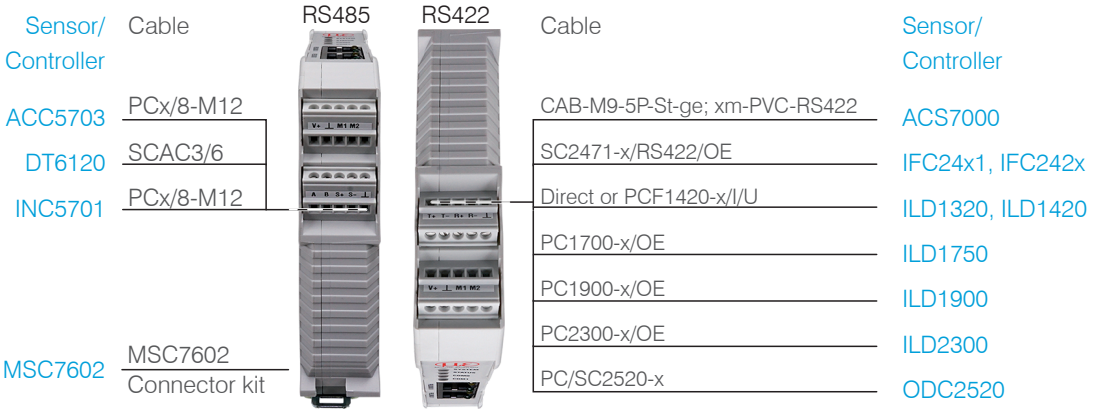


Fig. 4 Optional supply voltage wiring at rear of terminal

4.2.2 Connection Options



The length of the cable between IF2030/PNET and sensor/controller is 10 m at most. Because of the PCx/8-M12 cable, the sensor supply for ACC5703 and INC5701 sensors is possible only via the IF2030/PNET.

Fig. 5 Connection examples for IF2030/PNET

Fig. 6 Connection of an MSC7602 with MSC7602 connector kit

IF2030/PNET	Sensor/Controller
RS422	
T+	R+
T -	R -
R+	T+
R -	T -
⊥	Cable shield
RS485	
A	A
B	B
⊥	Cable shield

Fig. 7 Wiring regulation for connections with RS485 or RS422

4.2.3 Cable Termination at Interface

Ensure correct cable termination for an RS485 bus or RS422 bus!

We recommend a 120 Ohm terminating resistor between the signal lines at both the bus start and end. IF2030/PNET works as a master for both interfaces; internally, a 120 Ohm terminating resistor has already been permanently incorporated. The IF2030/PNET should be at the bus start.

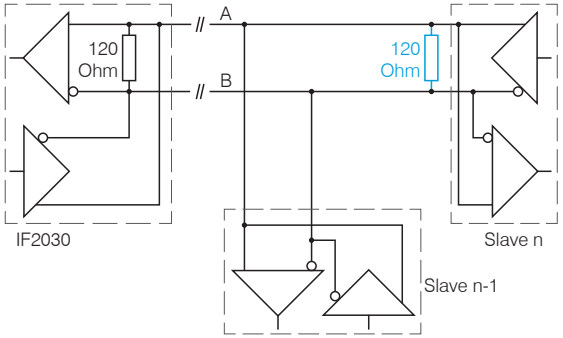


Fig. 8 Cable termination RS485, n = max. 16 slaves

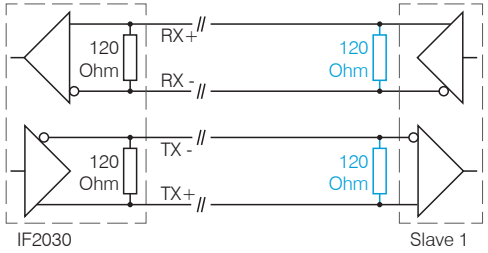


Fig. 9 Cable termination RS422

4.3 Fieldbus Cabling

During cabling, channel 0 of the IO controller is connected to a port of the first IO device (slave device). The second port of the first slave device is connected to the input port of the next slave device, etc. One port of the last slave device and channel 1 of the master device remain unused.

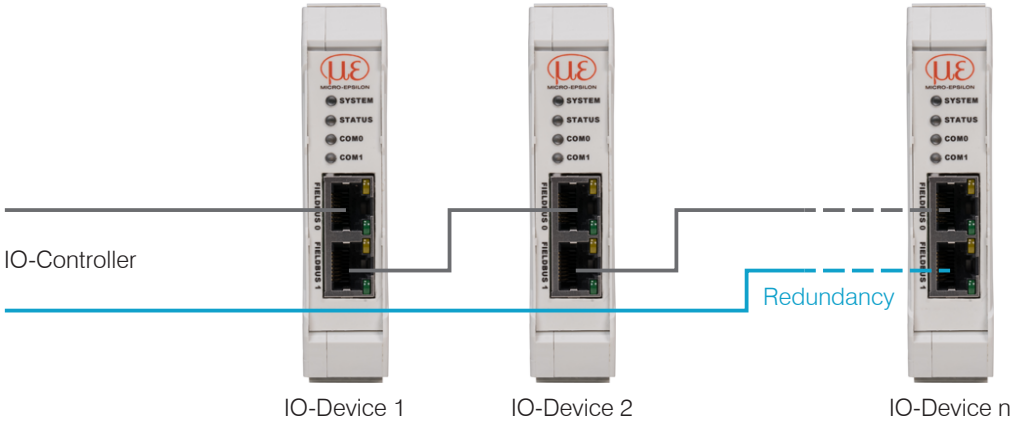


Fig. 10 Cabling in the PROFINET IO network

Optional: You achieve greater failsafe network performance if you implement an additional redundant connection (MRP = Media Redundancy Protocol) between the output port of the last slave device and channel 1 of the IO controller. IF2030 can participate in an MRP ring as a client; however, it cannot manage the ring. To achieve ring functionality, all participants must be configured as ring participants.

5. Initial Operation

5.1 Configuring the Sensors

The sensor used must be correctly configured to work with the IF2030. Micro-Epsilon recommends that the sensor's base configuration be set by using its web interface. The configuration can later also be adjusted via fieldbus.
Please refer to the operating instructions of the corresponding sensor for detailed information on configuring the sensor.

5.2 Baud Rate and Sensor Interface

IF2030/PNET must be set for the interface used and the sensor's baud rate. The baud rate and sensor interface can be configured in various ways.

Sensor/Controller	Baud rate [Baud]	RS485	RS422
ACC5703	230400	•	
ACS7000	230400		•
DT6120	230400	•	
IFC24x1	115200		•
IFC242x	115200		•
ILD1320	921600		•
ILD1420	921600		•

Sensor/Controller	Baud rate [Baud]	RS485	RS422
ILD1750	921600		•
ILD1900	921600		•
ILD2300	921600 [1]		•
INC5701	230400	•	
MSC7602	256000	•	
ODC2520	115200		•

Fig. 11 Baud rate (factory setting) of the sensors or controllers to be connected

5.2.1 Option 1: Basic Settings Module

After setting up the IF2030/PNET in the TIA portal, see Chap. A 3, the `Basic Settings` module offers an easy way to specify the required settings.

➡ Select the baud rate and interface as necessary, see Chap. 5.7.4.

1) The ILD2300 is set for 691.2 kBaud ex factory. Increase the baud rate to 921.6 kBaud in the sensor.

5.2.2 Option 2: TIA Components

The download package with the GSDML file includes preset function components that allow for easy access to options for the IF2030/PNET.

➡ Select the `IF2030_BaudrateInterface` component and transfer the parameters, see Chap. 5.7.3.

5.2.3 Option 3: Directly Access the Object Directory

Use the `WRREC_DB` function component to send the desired baud rate and sensor interface to the IF2030/PNET, see Chap. 5.5.

5.3 Data Format

All configuration parameters and data are transmitted in Little Endian format.

Sensors/controllers with RS485: cyclical data are transmitted via the fieldbus without change, i.e., as a binary block as described and supplied by the sensor. Please refer to the sensor's operating instructions for the data set structure.

Sensors/controllers with RS422: cyclical data are decoded, i.e., a 4th byte is added to the 3 bytes and then transmitted. The sensor signals selected for transfer and their sequence are available on the sensor's web interface.

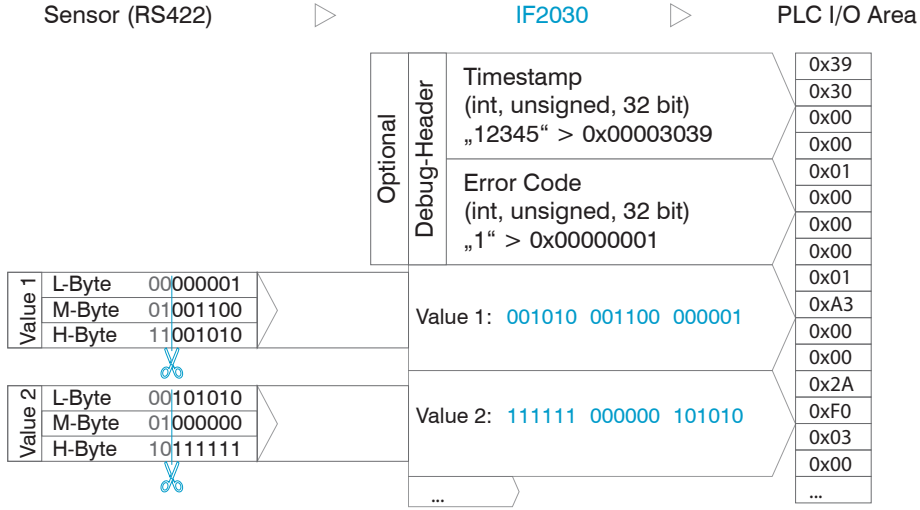


Fig. 12 Interpretation of RS422 sensor data in IF2030/PNET

5.4 Object Directory

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2000	0	Uint8	RW	select sensor	✓		Address of currently selected sensor
0x2001	0	Uint8[32]	R	sensor addresses	✓		Shows address list of available sensors
0x2010	0	Uint32[64]	R	device error log	✓	✓	Reads out the last 32 error codes with time stamp
0x2020	0	Uint32	RW	baud rate	✓	✓	IF2030 baud rate
0x2021	0	Uint8	RW	minimum cycle time	✓	✓	Minimum time for one communication cycle in ms, cycle time = 0: use estimated time
0x2022	0	Uint8	RW	preferred sensor data size	✓	✓	Desired useful data length in bytes per sensor, value = 0: estimated/calculated value
0x2023	0	Uint8	RW	serial sensor interface	✓	✓	0: RS485, 1: Reserved, 2: ASCII + RS422
0x2024	0	Uint8	W	clear device config	✓	✓	One byte deletes settings from flash, settings are included in RAM until restart
0x2025	0	Uint8	W	clear sensor config	✓		One byte deletes settings from flash, settings are included in RAM until restart
0x2026	0	Uint8	W	reset device	✓	✓	One byte performs reset
0x2027	0	Uint8	RW	enable/disable HTTL Sync	✓	✓	0: Disable HTTL synchronization 1: Enable HTTL synchronization
0x2028	0	Uint8	RW	enable/disable cyclic status header	✓	✓	0: Disable 8 byte status header in cyclic data 1: Enable 8 byte status header in cyclic data

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2210				Device Info	✓		Read out the block of the current sensor
	0	Uint8	R	NrOfObjects			
	1	Uint8	R	Block version			Block version
	2	Uint8	R	Endianness			Endian
	3	Uint16	R	Software version			Software version
	4	Int32	R	Article number			Part number
	5	Int32	R	Option			Option
	6	Int32	R	Batch number			Batch number
	7	Int32	R	Serial number			Serial number
	8	Uint8	R	Change index			Change index
	9	Uint8	R	Calibration day			Day of calibration
	10	Uint8	R	Calibration month			Month of calibration
	11	Uint8	R	Calibration year			Year of calibration
	12	Uint16	R	Calibration software version			Version of calibration software
	13	Uint16	R	Test software version			
	14	Uint8	R	Test hour			
	15	Uint8	R	Test day			
	16	Uint8	R	Test month			
	17	Uint8	R	Test year			
	18	Int32	R	Article number circuit board			
	19	Int32	R	Serial number circuit board			
	20	Uint8[32]	R	Name			
	21	Uint8	R	sensor/channel count			
	22	Uint8	R	protocol block count			
	23	Uint8[164]	R	protocol blocks			

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2213				Diagnostic block	✓		RS485 bus diagnostic block (if available)
	0	Uint8		NrOfObjects			
	1	Uint8	RW	page index to read			Specifying an index lets you scroll through existing pages
	2	Uint8	R	number of pages			
	3	Uint8	R	diagnose Type			
	4	Uint8[235]	R	String Page			Diagnostic message

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2220				Sensor block	✓		Request sensor information
	0	Uint8	R	NrOfObjects			
	1	Uint8	RW	block index offset			The offset lets you scroll through existing sensor blocks [0 - 0x1F]
	2	Uint8	RW	page index to read			Specifying an index lets you scroll through existing pages
	3	Uint8	R	number of pages			Max. number of pages
	4	Uint8	R	measurement unit			Signal unit
	5	Int32	R	article number			Part number
	6	Int32	R	Option			Option
	7	Int32	R	Batch number			Batch number
	8	Int32	R	serial number			Serial number
	9	Float	R	Nominal measuring range			Nominal measuring range
	10	Float	R	Nominal offset			Nominal offset
	11	Float	R	current measuring range			Actual measuring range
	12	Float	R	current offset			Actual offset
	13	Uint8[32]	R	Target material			Target material
	14	Uint8[32]	R	Sensor/channel name			Sensor/channel name
	15	uint8	R	extension length			Length of block extension
	16	uint8[138]	R	extension			

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2501				Parameter Info	✓		Request configuration parameters, e.g., sensor exposure time, request via subindex 1, configure interface with objects 0x2510 through 0x2540
	0	Uint8	R	NrOfObjects			
	1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2	Uint8[14]	R	Name			
	3	Uint8[8]	R	Unit			
	4	Uint8[8]	R	Type			

0x2510				Float parameter	✓		Read or write float parameter
	0	Uint8		NrOfObjects			
	1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2	Uint8	RW	Reserved			
	3	Float	RW	Value			Value
	4	Uint8[14]	R	Name			Designation
	5	Uint8[8]	R	Unit			Unit as a string
	6	Float	R	Min			
	7	Float	R	Max			

Index	Subindex	Data type		Name	RS485	RS422	Description
0x2520				Int Parameter	✓		Read or write integer parameter
	0	Uint8		NrOfObjects			
	1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2	Uint8	RW	Reserved			
	3	Int32	RW	Value			Value
	4	Uint8[14]	R	Name			Designation
	5	Uint8[8]	R	Unit			Unit as a string
	6	Int32	R	Min			
	7	Int32	R	Max			

0x2530				Uint Parameter	✓		Read or write unsigned integer parameter
	0	Uint8		NrOfObjects			
	1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2	Uint8	RW	Reserved			
	3	Uint32	RW	Value			Value
	4	Uint8[14]	R	Name			Designation
	5	Uint8[8]	R	Unit			Unit as a string
	6	Uint32	R	Min			
	7	Uint32	R	Max			

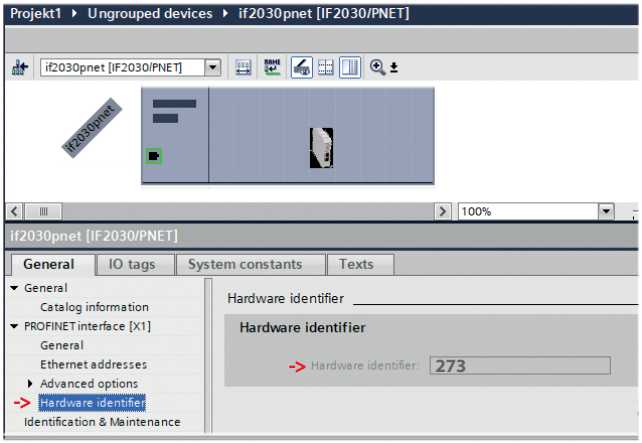
Index	Subindex	Data type		Name	RS485	RS422	Description
0x2540				String Parameter	✓		Read or write string parameter
	0	Uint8		NrOfObjects			
	1	Uint16	RW	Parameter ID			Please refer to the sensor documentation for available parameter IDs and their types
	2	Uint8	RW	Reserved			
	3	Uint8[246]	RW	Value			Value
	4	Uint8[14]	R	Name			Designation

0x2600				RS422 ASCII Access		✓	RS422 commando
	1	Uint8[128]	RW	Send Cmd			Buffer for a 128-character ASCII command, termination with '\n' or 0x0A
	2	Uint8[896]	R	Cmd Answer			Answer from sensor without shortening, e.g., Line feed; if buffer overflows, e.g., PRINT ALL, answer is truncated

5.5 Sequence When Writing and Reading Acyclical Data

➡ Determine the hardware identification (ID) of the module. To do so, switch to the `General` > `PROFINET-Interface` > `Advanced Options` tab.

In the example to the right, you get the value 273.



On the SPS, `WRREC_DB` with input parameters (:=) is called.

`REQ` // Start execution

`ID` // Hardware ID of the target device addressed

`INDEX` // Target address in the object directory

`LEN` // Length of the binary data block to be written

`RECORD` // Usable data for writing

`RECORD`, `VALID`, `BUSY`, `ERROR`, `STATUS` and `LEN` contain return parameters (=>) that allow for determining the success or progress of the write command.

WRREC_DB		
REQ :=	1 → 0	Enable-Flag
ID :=	273	HW-ID
INDEX :=	0x2600	Objekt Index
LEN :=	20	8 Byte + Data Length
RECORD :=		
DONE =>	Status/Result Output	
BUSY =>		
ERROR =>		
STATUS =>		

Write-Header (8 Byte)							
0	0	0x01	0	0x0d	0	0	0
Reserved	Reserved	Subindex	Reserved	Data length (UINT32-LE)			

Command: "LASERPOW ON\n"												
'L'	'A'	'S'	'E'	'R'	'P'	'O'	'W'	0x20	'O'	'N'	0x0A	

Command: "LASERPOW ON\n"												
'L'	'A'	'S'	'E'	'R'	'P'	'O'	'W'	0x20	'O'	'N'	0x0A	

Fig. 13 SPS write command with 8 Byte prefix to turn on the laser light source on the sensor

The write and read commands, see Fig. 14, belong together. The write command `WRREC` that is used to send the command to the IF2030 is executed first. IF2030 forwards the command to the sensor. IF2030 immediately notes the sensor answer in a buffer. On the SPS `RDREC` is executed and thus the last command and the answer buffer are read back at the same time; the answer from IF2030 is saved in the RECORD buffer.

Fetch answer, see Fig. 14.

This chronology is a reaction to the prior write command, see Fig. 13.

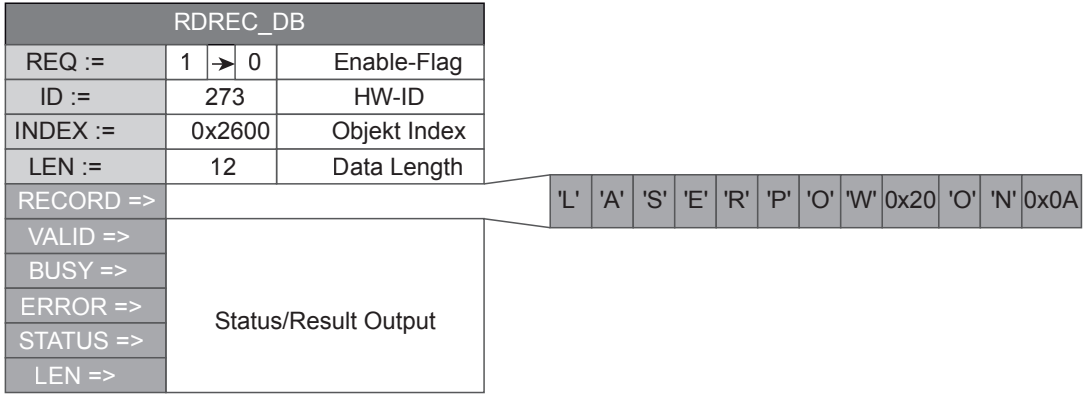


Fig. 14 SPS read command

The examples below show how to turn off the laser light source on the sensor.

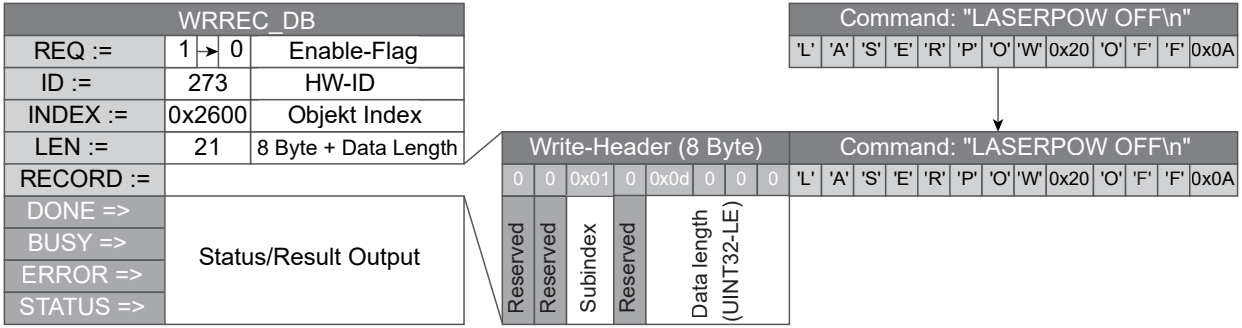


Fig. 15 SPS write command with 8 Byte prefix to turn off the laser ligth source on the sensor

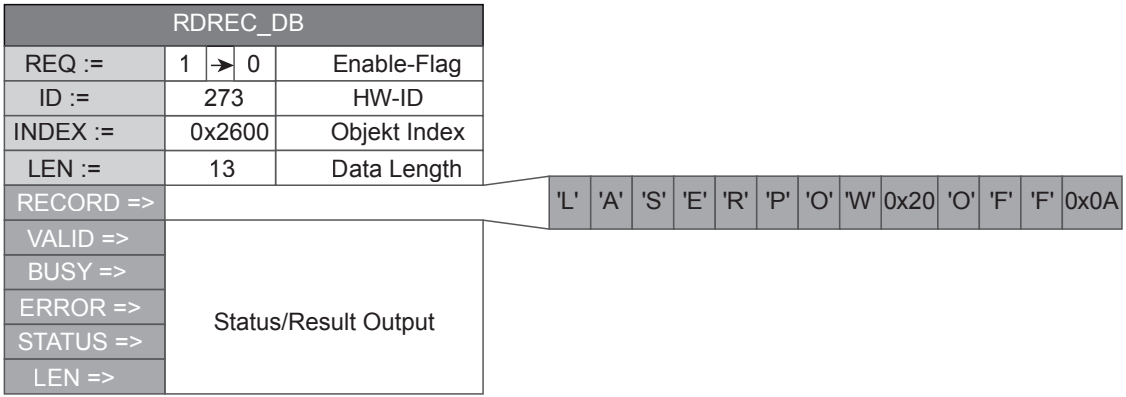


Fig. 16 SPS read command to turn off the laser light source on the sensor

5.6 Sequence When Writing Structured Data

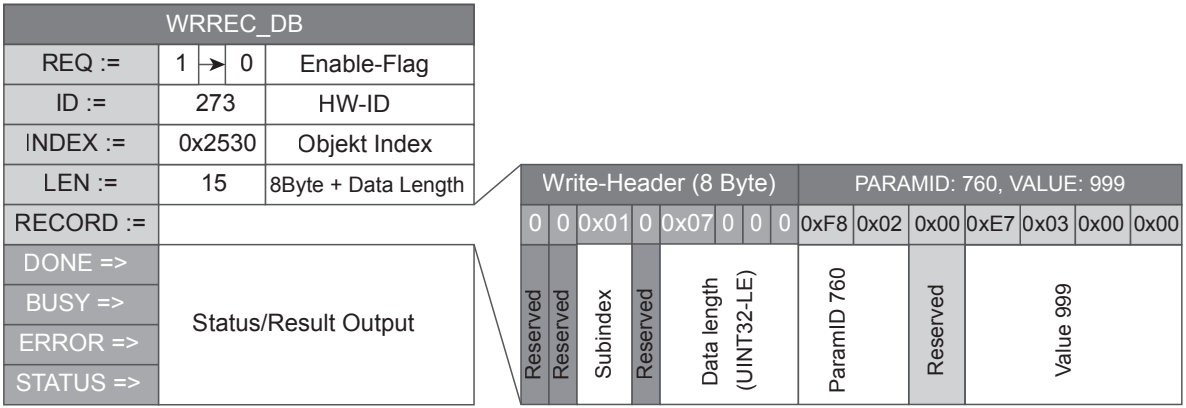


Fig. 17 Write command with data from SPS to IF2030/PNET

5.7 TIA Function Components

5.7.1 General

You can configure your IF2030/PNET via S7 by using several function components. They cover core functions that can be used for all compatible Micro-Epsilon sensors.

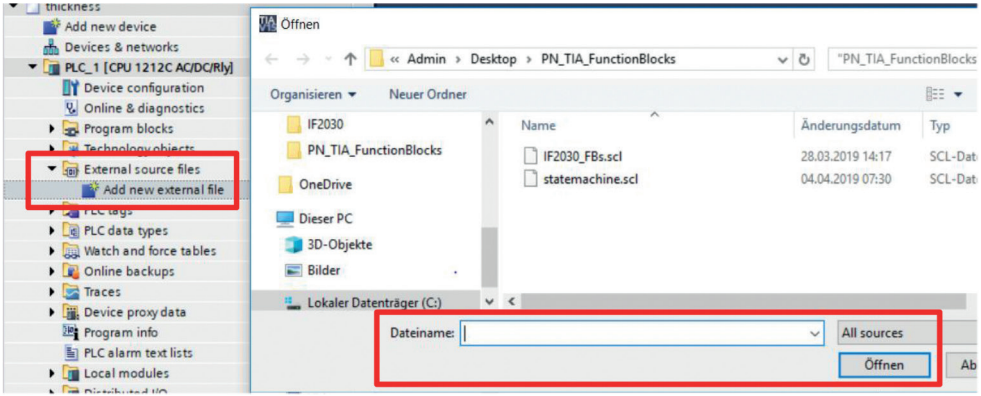
The components are available in an unencrypted form to allow you to view the code (“Structured Control Language”) and use it as a template for your own programs. The function components are provided together with the GSDML file. An overview of configuration examples, incl. register addresses in hexadecimal format, is provided below.

- IF2030_BaudrateInterface: R/W sensor interface (0x2023) and baud rate (0x2020)
- IF2030_CycletimeDatasize: R/W cycle time (0x2021) and data size (0x2022)
- IF2030_HTTL-Debug: R/W switching between HTL/TTL (0x2027) and enabling of the DebugHeader (0x2028)
- IF2030_MEB_floatparam: R/W Float-type sensor parameter (0x2510)
- IF2030_MEB_intparam: R/W Int-type sensor parameter (0x2520)
- IF2030_MEB_uintparam: R/W UInt-type sensor parameter (0x2530)
- IF2030_Reset: Delete W IF2030/PNET- (0x2024) or sensor configuration (0x2025) and perform a reboot (0x2026)
- IF2030_SelectSensor: R/W sensor selection (0x2000) and R sensor list (0x2001)

The `device_id` of the connected sensor or controller must be transferred to each function component. A trigger for `TRUE` in a program cycle triggers the desired action, a Read command is prioritized over a Write command, and the action’s end is signaled by `done = TRUE`. A `status != 0` shows an error when data is sent or received. With `reset_after_write = TRUE`, the IF2030/PNET is rebooted after successful configuration to allow the changes made to take effect immediately.

5.7.2 Importing Function Components

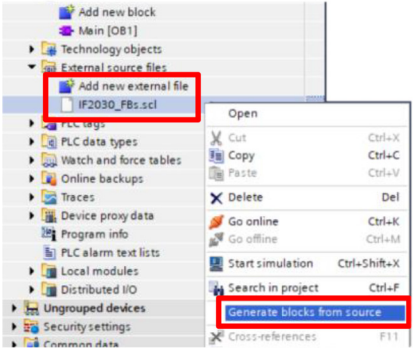
- Go to `Project navigation`. In your SPS, follow the path `External source files > Add new external file`. Double-clicking that link opens a `Dialog window`.



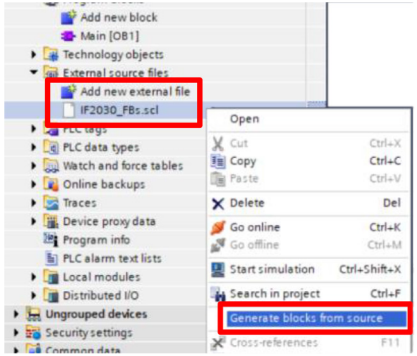
- Now select the path for the source file `IF2030_FBs.scl` and click the `Open` button.

The file is located in the `External source files` folder. Next, the function components must be transferred to the program components.

- Open the context menu by right-clicking the file and select the `Generate blocks from source` function in that menu. If displayed, confirm a message that existing blocks will be overwritten.

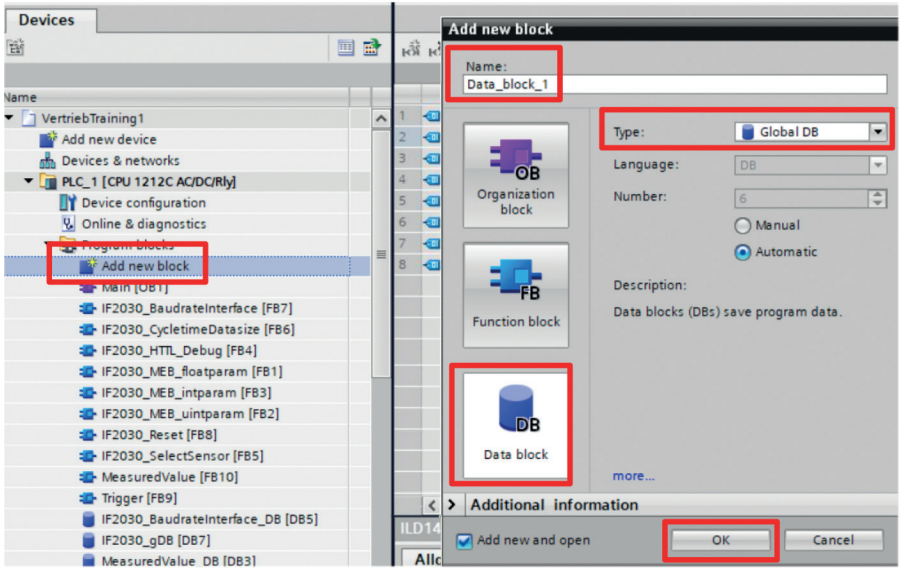


The function components generated are now available in the `Program components` folder. You can also view the result of generating them in the `Inspection window` on the `Info` > `Compile` tab. Please note that these messages refer to the source file.



5.7.3 Executing Function Component

- To create a global data component, select your CPU's `Program components` folder in the `Project navigation` and then click `Add new block`.
- In the next dialog window, select the option `Data block` and change the name, if necessary. Select `Global DB` for the type. Enable the check mark `Add new and open`, unless this has already been done automatically. Confirm the selection with OK.

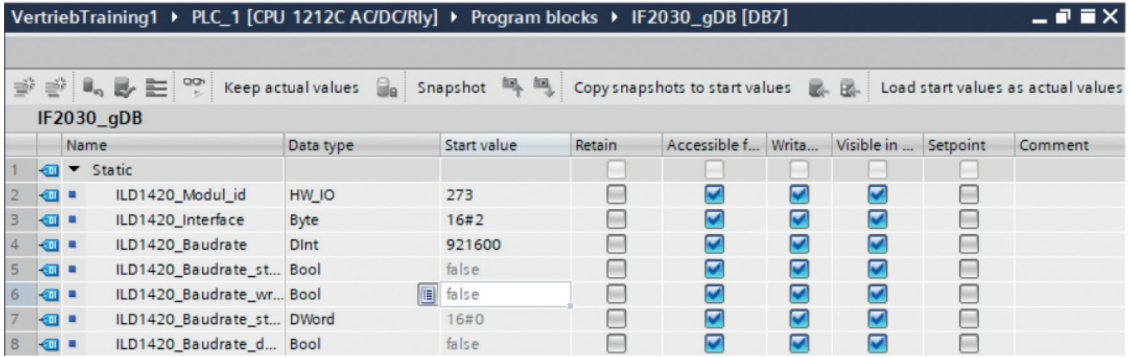


The data component is automatically displayed.

- Now create the necessary variables, depending on the function component.

The starting value is the value used when the data component is loaded into the CPU's memory.

- Next, click the `Save project` button (top left on the function bar).



i Ensure that you use the correct data types. You must correctly link the global variables with those in the function component.

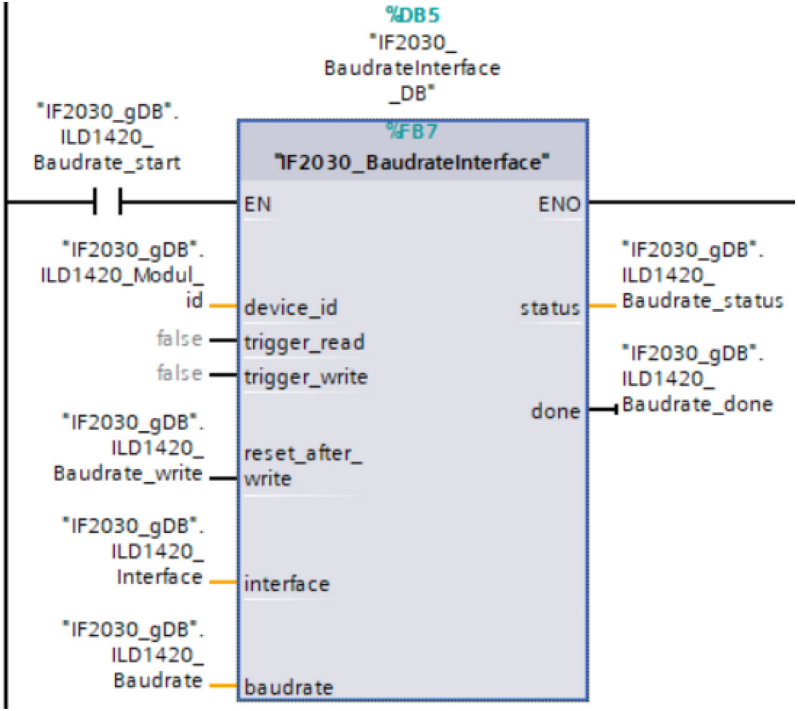
For a function component to be processed, it must be called in the program.

➡ Open the `Main [OB1]` organization component by double-clicking it. Mark your function component and drag it into the program of the previously opened organization component.

i There is no requirement that the call be performed via the main program OB1; that program is always processed by the CPU by default.

➡ Now use your mouse to drag-and-drop the variables needed for the circuit from your data component to the connections of your function component that is to be called. Alternatively, you can also enter the variables manually.

i If you vertically separate the editing area by clicking the symbol button on the function bar, the data component and the `Main [OB1]` organization component are displayed next to each other.



➡ Save your project. To translate all components, click the `Program components` folder in `Project navigation` and select the symbol button for `Translation` on the function bar.

i The components that were translated successfully are then displayed under `Inspection window` > `Info` > `Translation`.

After successful translation, the entire control system with the program generated, including the hardware configuration, can be loaded by using the `Load to device` symbol button.

5.7.4 Basic Settings Module

You can specify basic settings using the TIA portal interface. Proceed as follows to configure the IF2030/PNET based on a few basic parameters.

- In the `Hardware catalog`, select the `Basic settings` output module and place it in the next free slot in the `Device overview`.

The input module must be located before the basic settings module.

- Double-click the module in the `Device overview`. In the `Inspection window`, navigate to the `Properties` > General > Component parameters tab.

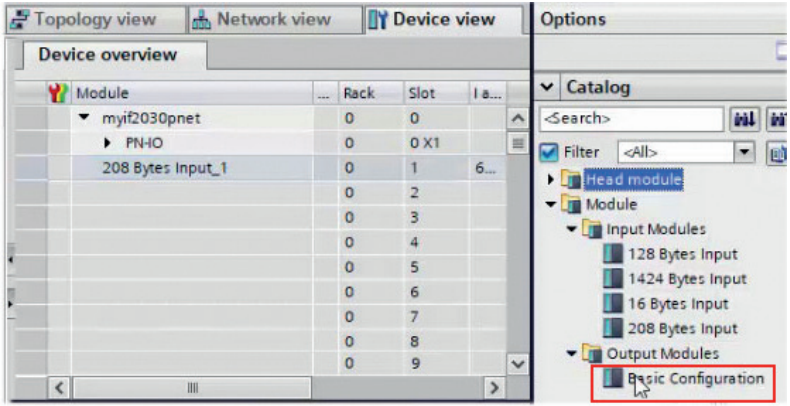
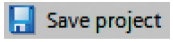


Fig. 18 Basic settings module in the hardware catalog

➡ Specify the required settings, e.g., baud rate, sensor interface. Enable the settings by using the parameter `Init-Config`.

Detailed information about this process is available in the Baud rate and Interface section, see Chap. 5.2.

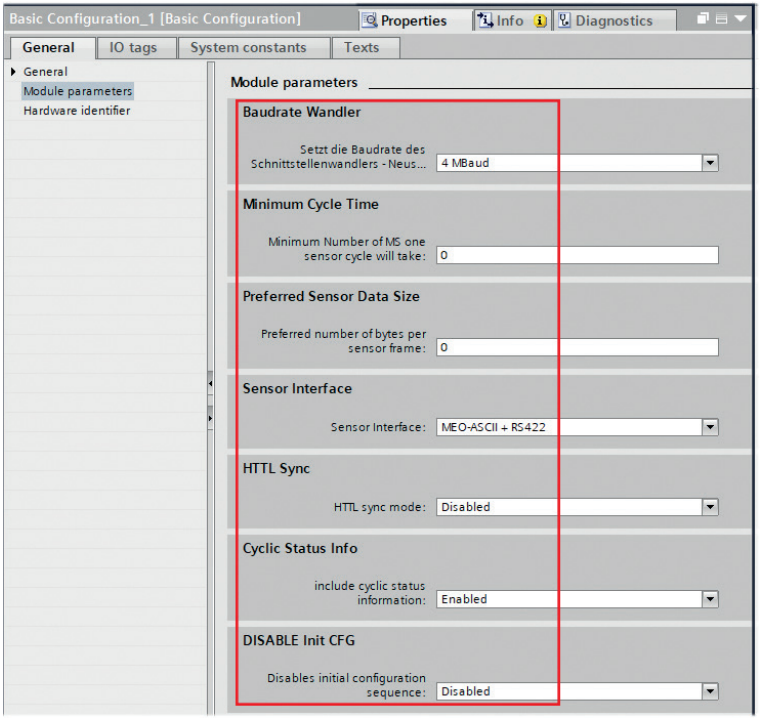
➡ Save your changes by clicking the `Save` 

button (at top left on the function bar).

Now you can load the settings into the CPU.

➡ Select your S7 device in the `Working window` if this has not been done already, and click the `Load to device` symbol button on the `Function bar`.

➡ Reboot the IF2030/PNET to have the changes take effect!



i This step is necessary because of the selected/used mechanism for establishing a GUI to parameterize the IF2030/PNET; it prevents the configuration from being sent repeatedly to the CPU as programs are run.

Please refer to the instructions for commissioning for more information. The current version is available at https://www.micro-epsilon.de/download/manuals/mav--IF2030-PNET--en.pdf.

6. Liability for Material Defects

All components of the device have been checked and tested for functionality at the factory. However, if defects occur despite our careful quality control, MICRO-EPSILON or your dealer must be notified immediately.

The liability for material defects is 12 months from delivery. Within this period, defective parts, except for wearing parts, will be repaired or replaced free of charge, if the device is returned to MICRO-EPSILON with shipping costs prepaid. Any damage that is caused by improper handling, the use of force or by repairs or modifications by third parties is not covered by the liability for material defects. Repairs are carried out exclusively by MICRO-EPSILON.

Further claims can not be made. Claims arising from the purchase contract remain unaffected. In particular, MICRO-EPSILON shall not be liable for any consequential, special, indirect or incidental damage. In the interest of further development, we reserve the right to make design changes without notification.

For translations into other languages, the German version shall prevail.

7. Service, Repair

Please send us the affected parts for repair or exchange.

If the cause of a fault cannot be clearly identified, please send the entire measuring system to:

MICRO-EPSILON MESSTECHNIK
GmbH & Co. KG
Koenigbacher Str. 15
94496 Ortenburg / Germany

Tel. +49/8542/168 - 0
Fax +49/8542/168 - 90
info@micro-epsilon.com
www.micro-epsilon.com

8. Decommissioning, Disposal

➡ Remove all cables from the interface module.

Incorrect disposal may cause harm to the environment.

➡ Dispose of the device, its components and accessories as well as the packaging materials in compliance with the applicable country-specific waste treatment and disposal regulations of the region of use.

Appendix

A 1 Optional Accessories

PS2020

Power supply; installation of top-hat rail, 2.5 A, input 100 - 240 VAC, output 24 VDC/2.5 A, installation type; installation on symmetrical standard rail 35 mm x 7.5 mm, DIN 50022

A 2 Factory Settings

Baudrate	9600 Baud
cycleMinTime	0 (= IF2030 calculates cycle time)
SensorInterface	MEO+RS422
HTTL	OFF
CyclicDebugHeader	OFF

A 3 Integration Into TIA Portal

The GSDML file contains information about a PROFINET device. This file is needed for the PROFINET controller and must be integrated into the corresponding configuration software.

➡ Import the GSDML file. To do so, in the `Extras` > `Manage device description files (DDF)` menu, select the path for the file <`GSDML-Vx-MICRO-EPSILON-IF2030.xml`>.

➡ Click the `Install` button.

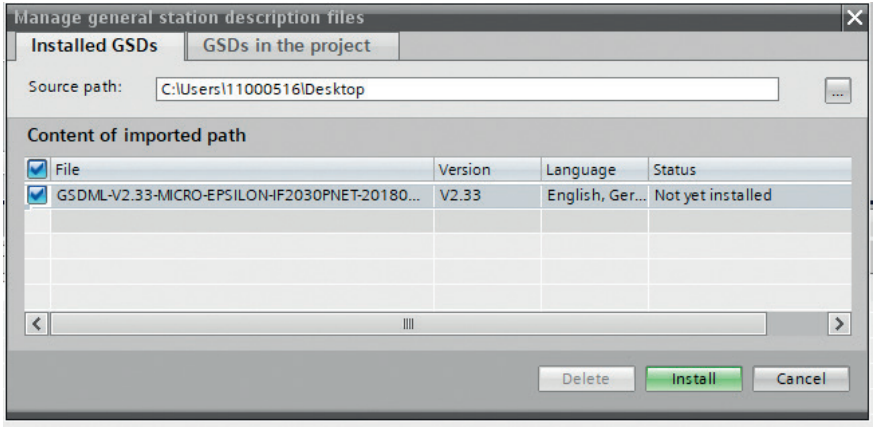
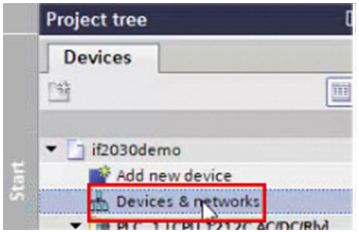


Fig. 19 Importing the device description file

After installation, switch to the project view.

➡ In `Project navigation`, click `Devices & networks`.



Add IF2030/PNET to the project.

➡ Switch to the hardware catalog tab.

➡ In the menu, select `Other field devices > PROFINET IO > I/O > MICRO-EPSILON MESSTECHNIK GmbH > PNS > IF2030/PNET.`

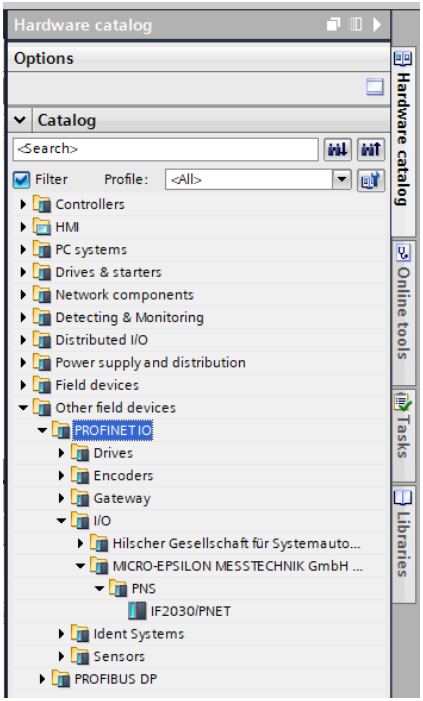
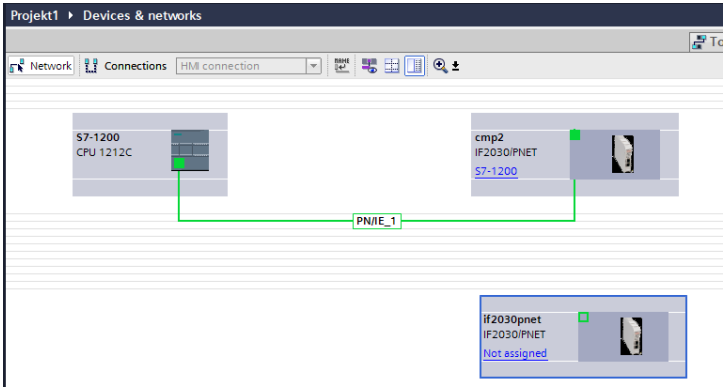
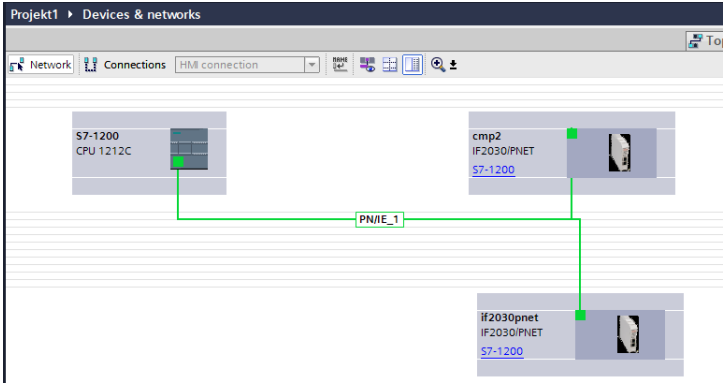


Fig. 20 Selecting IF2030/PNET as the hardware

➡ Drag `IF2030/PNET` into the project.



➡ Connect the green PN port in the device diagram to the PN network or to the PN connection of the SPS.



Enter the device name for identification in the PN network.

➡ Switch to the `Device view`, double-click your IF2030/PNET and set its device name in the `Inspection window` (`Properties` > `General` tab).

i The device name is used to identify the device on the PN network and as an address; it must be unique across the entire system.

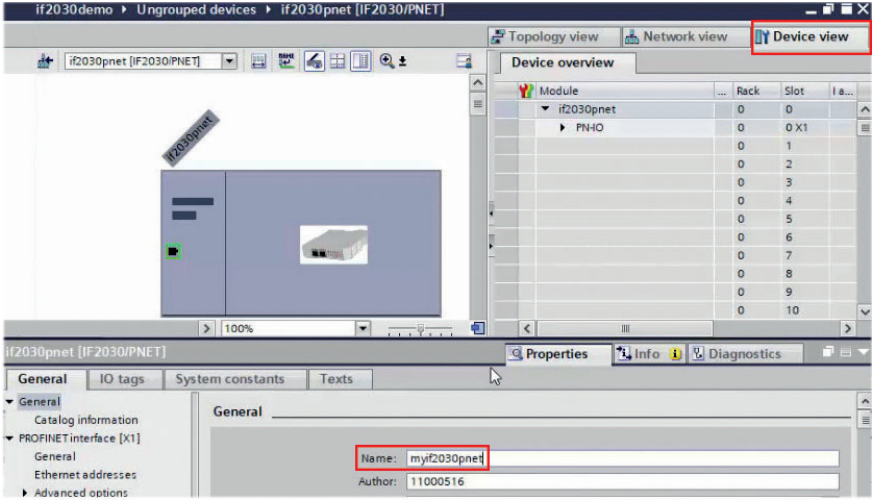


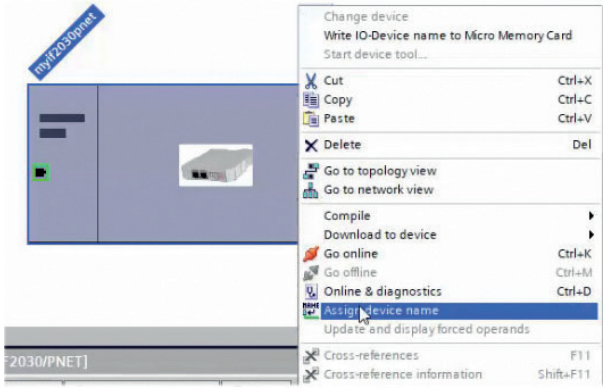
Fig. 21 Assigning a device name

The change of name must be communicated to the PN network.

➡ Right-click the IF2030/PNET.

You now reach the context menu shown.

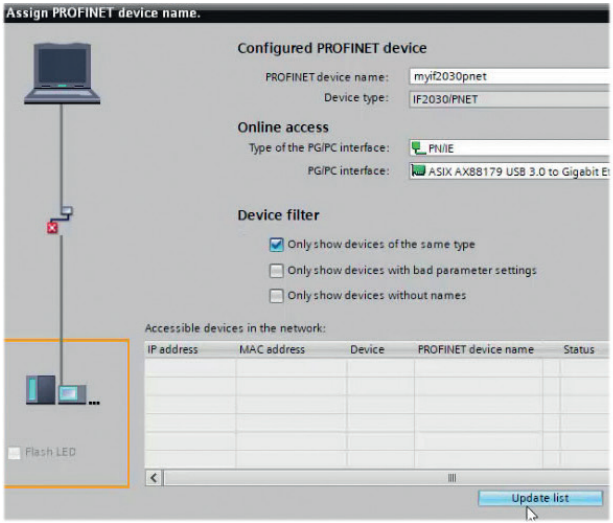
➡ Select the `Assign device name` entry.

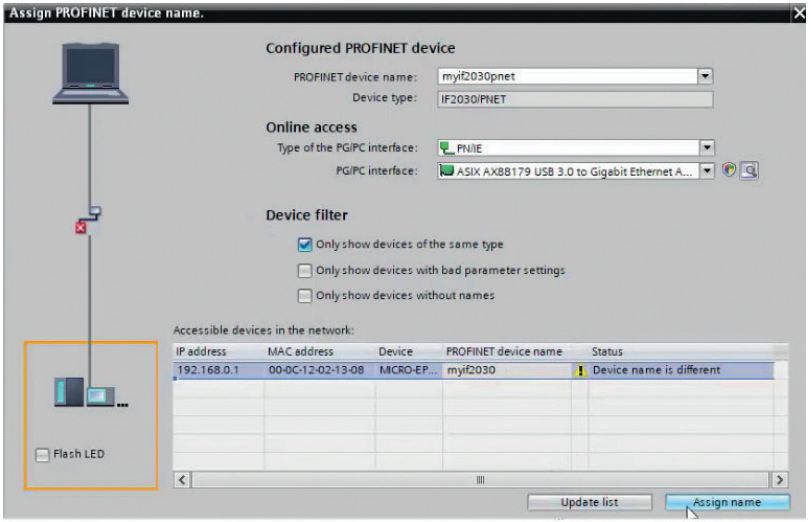


➡ In the open dialog window, click the `Update list` button.

Potential devices on the PN network are displayed.

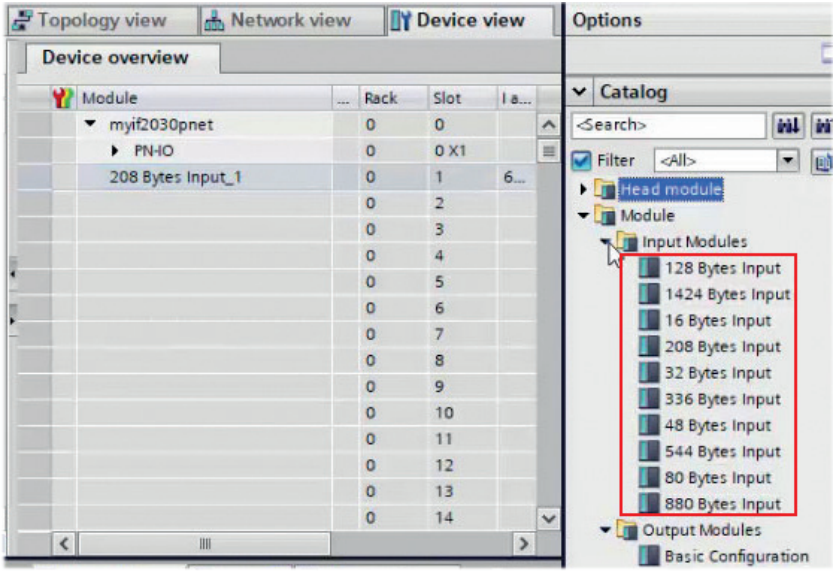
➡ In the list that is now displayed, mark the row with your IF2030/PNET that is to be renamed; field `Status`, "*Device name is different*". Finally, click the `Assign name` button.





Add modules to the device.

- Double-click the device.
- In the `Hardware catalog`, first select the matching input module for the usable data (the module is determined by the component and must match the one selected in TIA); drag it to the first free slot in the `Device overview`.
- In the hardware catalog, select the output module `Basic settings` and drag it to the next free slot in the device overview.



A 4 Sensor Values, Data Format, Conversion

A 4.1 General

The sensors or controllers do not solely output distance values. The overview below describes the conversion during output of distance values. Please refer to the corresponding operating instructions for detailed information on conversion when additional values are output.

A 4.2 ACC5703

Baud rate 230400 b/s RS485 half duplex Max. sampling rate 1 kHz: measurements with variable number ex factory scaled to ±2 g, Little Endian

Bus address 126

Byte Data	Meaning	Data format
Data[0]	Status byte (contains error flags, normally 0x00)	8 bit
Data[1] ... Data[4]	Measured value counter [bit 0:31]	Uint 32 bit
Data[5]	Number of measured values in this package = 3*x mit x [1 ... 19]	8 bit
Data[6]	Padding-Byte	8 bit
Data[7]	Padding-Byte	8 bit
Data[8]	Measuring value 1 x-axis [bit 0:7]	Float 32 bit
Data[9]	Measuring value 1 x-axis [bit 8:15]	
Data[10]	Measuring value 1 x-axis [bit 16:23]	
Data[11]	Measuring value 1 x-axis [bit 24:31]	
...	...	
Data[n] n=8+(4*Data [5]/3)	Measuring value 1 x-axis [bit 0:7]	Float 32 bit
Data[n+1]	Measuring value 1 x-axis [bit 8:15]	
Data[n+2]	Measuring value 1 x-axis [bit 16:23]	
Data[n+3]	Measuring value 1 x-axis [bit 24:31]	
...	...	...
Data[n+m] m=4*Data[5]/3	Measuring value 1 z-axis [bit 0:7]	Float 32 bit
Data[n+m+1]	Measuring value 1 z-axis [bit 8:15]	
Data[n+m+2]	Measuring value 1 z-axis [bit 16:23]	
Data[n+m+2]	Measuring value 1 z-axis [bit 24:31]	

Please refer to the operating instructions for the acceleration sensor for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--inertialSENSOR-ACC5703--en.pdf

Fig. 22 Encoding of Measured Data in the Transmission Protocol, ACC5703

A 4.3 ACS7000

RS422	Measuring rate 250 Hz ex factory, all color values and color distances. Up to 32 output values can be transmitted at the same time.
Baud rate	115200 b/s

ACS7000 supplies 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

Group	Name	Index	Raw		Scaled			
			Min	Max	Min	Max	Formula	Unit
Status	Framerate	1	2500	250000	20,00	2000,00	10 ^ 6/(x*12,5*2 ^ 4)*1000	Hz
	Shutter	2	2500	250000	20,00	2000,00	x*12,5*2 ^ 4)/10 ^ 9	µs
	TempDetector	3	-1024	1023	-256,00	255,75	x/4	°C
	TempLightSrc	4	-1024	1023	-256,00	255,75	x/4	°C
LightSensor	Red	5	0	65535	0,00	100,00	x/65536*100	%
	Green	6	0	65535	0,00	100,00	x/65536*100	%
	Blue	7	0	65535	0,00	100,00	x/65536*100	%
	Brightness	8	0	65535	0,00	100,00	x/65536*100	%
Status	Counter	9	0	262143	0	262143	x	-
	Timestamp	10	0	262143	0,00	67,11	x*256/100000	s
Color	XYZ	11-13	0	131072	0,00	256,00	x/512	-
	RGB	14-16	0	131072	0,00	256,00	x/512	-
	LAB	17-19	-131072	131071	-256,00	256,00	x/512	-
	LUV	20-22	-131072	131071	-256,00	256,00	x/512	-
	LCH (L/C)	23-24	-131072	131071	-256,00	256,00	x/512	-
	LCH (H)	25	0	131071	0,00	256,00	x/512	°
	LAB99	26-28	-131072	131071	-256,00	256,00	x/512	-
	LCH99 (L/C)	29-30	-131072	131071	-256,00	256,00	x/512	-
	LCH99 (H)	31	0	184320	0,00	360,00	x/512	°

Group	Name	Index	Raw		Scaled			
			Min	Max	Min	Max	Formula	Unit
Status	Error	32	0	262143	0	262143	x	-
Distance	1_1/2/3	33-35	NA	-				
	…	36-77						
	16_1/2/3	78-80		-				
	Min_1/2/3	81-83	-131072	131071	-256,00	256,00	x/512	-
	DetectedID	84	0	16	0	16	-	-
	MinDistID	85	0	16	0	16	-	-

Fig. 23 Overview of output data via RS422

Please refer to the operating instructions for the color measuring system colorCONTROL ACS7000 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.com/download/manuals/man--colorCONTROL-ACS7000--en.pdf

A 4.4 DT6120

Baud rate	230400 b/s	RS485 half duplex	Measurements ex factory scaled to sensor measuring range, Little Endian
Bus address	126		

Measuring data consist of a counter, the packet length m and the measurements. The packet length m determines how many measurements are transmitted. The packet length m is the number of measurements that have been queried by the electronic system since the last time measuring data were queried, but is limited to the most recent 20 measurements. The first measurement in the data [] package is the oldest value queried, the last one is the most recently queried value.

Byte Data	Meaning	Data format
Data[0]	Counter [7:0]	unsigned short
Data[1]	Counter [15:8]	
Data[2]	Packet length m [7:0]	unsigned char
Data[3]	Filler byte [7:0]	unsigned char
Data[4]	Measuring value 1 [7:0]	signed integer
Data[5]	Measuring value 1 [15:8]	
Data[6]	Measuring value 1 [23:16]	
Data[7]	Measuring value 1 [31:24]	
	. . .	
Data[..]	Measuring value m [7:0]	signed integer
Data[..]	Measuring value m [15:8]	
Data[..]	Measuring value m [23:16]	
Data[..]	Measuring value m [31:24]	

Scaling of measurements

BY default, 24-bit measurements are transmitted.
The following equivalences therfore apply:

0x0 = 0 % of the sensor measuring range

0xF00000 = 100 % of the sensor measuring range

If the sensor is outside the measuring range, accordingly larger measurements are output.

Fig. 24 Encoding of Measured Data in the Transmission Protocol, DT6120

Please refer to the operating instructions for the capacitive displacement measuring system for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--capaNCDT-6110-6120IP--en.pdf

A 4.5 IFC2421, IFC2422, IFC2451, IFC2461, IFC2471

RS422	Up to 32 output values can be transmitted at the same time. The data are configured or selected via ASCII commands or via the web interface.
Baud rate	115200 b/s ex factory

Ex factory, the controller is set for the `Distance measurement` measuring program. Please refer to the associated operating instructions for descriptions of additional measuring programs. IFC24xx supplies 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

$$x = \frac{(d_{OUT} - 98232) * MR}{65536}$$

x	=	Displacement / Thickness in mm
dOUT	=	digital output value
MR	=	Measuring range in mm
131000	=	Midrange for the displacement measurement

Please refer to the operating instructions for the confocal displacement measuring system
- confocalDT 2421/2422
- confocalDT 2451/2461/2471

for more information, especially about possible output values.

The current version is available at:

https://www.micro-epsilon.com/download/manuals/man--confocalDT-2421-2422--en.pdf

https://www.micro-epsilon.de/download/manuals/man--confocalDT-2451-2461-2471--en.pdf

A 4.6 ILD1320, ILD1420

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; <643] SMR reserve [643; 64877] Measuring range [>64877; 65520] EMR reserve	$d\ [mm] = \frac{1}{100}\left(\frac{102}{65520}\ x - 1\right) * MR\ [mm]$
	MR = measuring range [mm]	{10/25/50/100/200/500}	
	d = distance [mm]	[-0,01MB; 1,01MR]	

Fig. 25 Calculation of distance value from the digital value, ILD1320/1420

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1320/1420 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1320--en.pdf

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1420--en.pdf

1) Distance value without the Master function.

A 4.7 ILD1750

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; 230604]	$d\ [mm] = \frac{x - 98232}{65536} * MR\ [mm]$
	MR = measuring range [mm]	{2/10/20/50/100/200}	
	d = distance [mm]	without Mastern [-0,01 MR; 1,01MR]	
		with Mastern [-2MR; 2MR]	

Fig. 26 Calculation of distance value from the digital value, ILD1750

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1750 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1750--en.pdf

A 4.8 ILD1900

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	921600 baud ex factory

The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/ENETIP into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; 230604]	$d\ [mm] = \frac{x - 98232}{65536} * MR\ [mm]$
	MR = measuring range [mm]	{2/10/25/50/100/200/500}	
	d = distance [mm]	without Mastern [-0,01 MR; 1,01MR]	
		with Mastern [-2MR; 2MR]	

Fig. 27 Calculation of distance value from the digital value, ILD1900

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 1900 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-1900--en.pdf

A 4.9 ILD2300

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	691200 baud ex factory [1]

i The sensor can continue to supply measurements to the RS422 output even while the sensor is communicating.

The digital measurements are output at the sensor as unsigned digital values (raw values). 16 Bit per value are transmitted. The sensors supply 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.

The linearized measuring values can be converted in µm using the following formula:

Value	Variables	Value range	Formula
Distance	x = digital value	[0; <643] SMR reserve [643; 64877] Measuring range [>64877; 65520] EMR reserve	$d\ [mm] = \frac{1}{100}\left(\frac{102}{65520}\ x - 1\right) * MR\ [mm]$
	MR = measuring range [mm]	{10/25/50/100/200/500}	
	d = distance [mm]	[-0,01MR; 1,01MR]	

Fig. 28 Calculation of distance value from the digital value, ILD2300

Please refer to the operating instructions for the laser-optical displacement sensors optoNCDT 2300 for more information, especially about possible output values. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoNCDT-2300--en.pdf

1) When delivered, ILD2300 is set for 691.2 kBaud. Increase the baud rate to 921.6 kBaud in the sensor.

A 4.10 INC5701

Baud rate 230400 b/s RS485 half duplex max. sampling rate 250 Hz, ex factory INC5701D, Little Endian
Bus address 126

Byte	Meaning	Data format
Data[0]	Status byte (contains error flags, normaly 0x00)	8 bit
Data[1]	Long term values counter [bit 0:7]	Uint 32 bit
Data[2]	Long term values counter [bit 8:15]	
Data[3]	Long term values counter [bit 16:23]	
Data[4]	Long term values counter [bit 24:31]	
Data[5]	Number of measured values in this package	8 bit
Data[6]	Padding byte	8 bit
Data[7]	Padding byte	8 bit
Data[8]	Measured value 1 [bit 0:7]	Float 32 bit
Data[9]	Measured value 1 [bit 8:15]	
Data[10]	Measured value 1 [bit 16:23]	
Data[11]	Measured value 1 [bit 24:31]	
Data[12]	Measured value 2 [bit 0:7]	Float 32 bit
Data[13]	Measured value 2 [bit 8:15]	
Data[14]	Measured value 2 [bit 16:23]	
Data[15]	Measured value 2 [bit 24:31]	

Fig. 29 Encoding of Measured Data in the Transmission Protocol, INC5701S

Byte Data	Meaning	Data format
Data[0]	Status byte (contains error flags, normally 0x00)	8 bit
Data[1] ... Data[4]	Measured value counter [bit 0:31]	Uint 32 bit
Data[5]	Number of measured values in this package	8 bit
Data[6], Data[7]	Padding-Byte	8 bit
Data[8]	Measuring value 1 LP [1] [bit 0:7]	Float 32 bit
Data[9]	Measuring value 1 LP [1] [bit 8:15]	
Data[10]	Measuring value 1 LP [1] [bit 16:23]	
Data[11]	Measuring value 1 LP [1] [bit 24:31]	
Data[12]	Measuring value 2 LP [1] [bit 0:7]	
Data[13]	Measuring value 2 LP [1] [bit 8:15]	
Data[14]	Measuring value 2 LP [1] [bit 16:23]	
Data[15]	Measuring value 2 LP [1] [bit 24:31]	
...	...	...
Data[n] n=8+(4*Data [5])	Measuring value 2 SF [2] [bit 0:7]	Float 32 bit
Data[n + 1]	Measuring value 2 SF [2] [bit 8:15]	
Data[n + 2]	Measuring value 2 SF [2] [bit 16:23]	
Data[n + 3]	Measuring value 2 SF [2] [bit 24:31]	
Data[n + 4]	Measuring value 2 SF [2] [bit 24:31]	
Data[n + 5]	Measuring value 2 SF [2] [bit 24:31]	
...	...	

Fig. 30 Encoding of Measured Data in the Transmission Protocol, INC5701D

1) LP = Low pass filter 2) SF = SensorFUSION filter

Please refer to the operating instructions for the inclination sensor for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--inertial-SENSOR-INC5701--de.pdf

The measurement data consits of one status byte, measured values counter, number of measured values, and the measured data. The measured values counter increases continuously with each sampled value. It represents the number of measured values transmitted in this package (floats). The first measurement value in the Data [] package is the oldest measured value. A measured value is represented as 4-byte float data type in the unit angular degrees [°].

A 4.11 MSC7602

Baud rate 256000 baud ex factory, [9600 ... 256000] RS485 half duplex Measurements ex factory scaled to analog value, Little Endian

Bus address 126 [2 ... 126]

Sequence for a measurement value request:

Send	0x10	0x7E [1]	0x01 [2]	0x4C	0xCB [3]	0x16									
Receive	0x68	0x0B	0x0B	0x68	0x01 [2]	0x7E [1]	0x08	0xAE	0x47	0x61	0x3F	0x00	0x00	0x00	0x00
	0x1C [4]	0x16													

Result	Description	Format	Example
	Unscaled value	Bytes 8 - 11: 4 Bytes, float, Little-Endian	0x3F6147AE (float) = 0.88 V
	Scaled value	Bytes 12 - 15: 4 Bytes, float, Little-Endian	If this value is 0, the controller was not set up. Otherwise, the digital counterpart of the analog output will be sent according the setting you have done in the controller before.
	Maximum speed for data transmission (1x send + 1x receive): ~3 ms @ 256.000 Baud		

1) DA: 126
2) SA: 1
3) CH: Checksum Send: Bytes 2 - 4
4) CH: Checksum Receive: Bytes 5 - 15

Fig. 31 Encoding of Measured Data in the Transmission Protocol, MSC7602

Please refer to the operating instructions for the inductive displacement measuring system for more information.
The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--induSENSOR-7xxx--en.pdf

A 4.12 ODC2520

RS422	The data are configured or selected via ASCII commands or via the web interface.
Baud rate	115200 Baud ex factory

Ex factory, the controller outputs the measurements in the `Edge light-dark` measuring program to the web diagram, i.e., output must be redirected to the RS422 interface.
The ODC2520 supplies 3 bytes per value at the output. These bytes are coded by the IF2030/PNET into 4 bytes, see Chap. 5.3.
The linearized measuring values can be converted in µm using the following formula:

$$x\ [\mu m] = d_{out} - 131000$$

x = Measuring value (edge position, difference, center axis) in µm

d_{out} = digital output value; $d_{out} \geq 262072$ are error values

Fig. 32 Calculation of edge position from the digital value, ODC2520

Please refer to the operating instructions for the laser micrometer optoCONTROL 2520 for more information. The current version is available at:

https://www.micro-epsilon.de/download/manuals/man--optoCONTROL-2520--en.pdf

MICRO-EPSILON MESSTECHNIK GmbH & Co. KG
Koenigbacher Str. 15 · 94496 Ortenburg / Germany
Tel. +49 (0) 8542/168-0 · Fax +49 (0) 8542/168-90
info@micro-epsilon.com · www.micro-epsilon.com
Your local contact: www.micro-epsilon.com/contact/worldwide/

X9751394-A032050SWE