IB IL RS 232-ECO

Inline ECO communication terminal for serial data transmission (RS-232)

Data sheet

107157_en_01
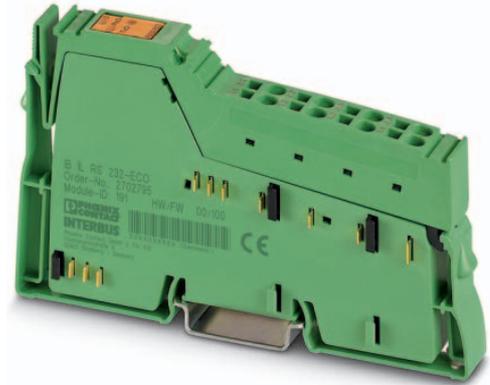
1   Description

The terminal is designed for use within an Inline station.
The terminal is used to operate standard I/O devices with serial interfaces on a bus system.
Inline ECO terminals are approved for the temperature range from 0°C to +55°C. The electronics base and Inline connector are supplied as standard.

Features

– One serial RS-232 input and output channel
– Data is transmitted in transparent mode
– Transmission speeds can be set up to 38400 Baud
– Number of data bits, stop bits and parity can be set
– RTS and CTS handshake signals available
– Parameterization via process data
– Diagnostic and status indicators

|	
i	This data sheet is only valid in association with the IL SYS INST UM E user manual.
i	Make sure you always use the latest documentation. It can be downloaded from the product at phoenixcontact.net/products.

INSPIRING INNOVATIONS

2   Table of contents

3 Ordering data

Description	Type	Order No.	Pcs./Pkt.
Inline ECO, communication terminal, serial data transmission, 1 serial input and output channel in RS-232 format, 500 kbps transmission speed in the local bus, IP20 protection, including Inline connector	IB IL RS 232-ECO	2702795	1

Accessories	Type	Order No.	Pcs./Pkt.
Connector, for digital 1, 2 or 8-channel Inline terminals (Connector/Adapter)	IB IL SCN-8	2726337	10
Inline shield connector (Connector/Adapter)	IB IL SCN-6 SHIELD	2726353	5
Labeling field, width: 12.2 mm (Marking)	IB IL FIELD 2	2727501	10
Insert strip, Sheet, white, unlabeled, can be labeled with: Office printing systems, Plotter: Laser printer, Mounting type: Insert, Lettering field: 62 x 10 mm (Marking)	ESL 62X10	0809492	1
Shield connection clamp, for shield on busbars, contact resistance < 1 mΩ (Assembly)	SK 8	3025163	10
Shield connection clamp, for shield on busbars, contact resistance < 1 mΩ (Assembly)	SK 14	3025176	10
Shield connection clamp, for shield on busbars, contact resistance < 1 mΩ (Assembly)	SK 20	3025189	10
Shield connection clamp, for shield on busbars, contact resistance < 1 mΩ (Assembly)	SK 35	3026463	10
Support bracket for busbars (Assembly)	AB-SK	3025341	10
Support bracket, Bracket for busbars, set every 20 cm, Length: 95.5 mm, Width: 6.2 mm, Color: gray (Assembly)	AB-SK 65	3026489	10
Support bracket, Bracket for busbars, set every 20 cm, Length: 10 mm, Width: 56 mm, Height: 20 mm, Color: silver (Assembly)	AB-SK/E	3026476	10
PEN conductor busbar, 3mm x 10 mm, length: 1000 mm (Assembly)	NLS-CU 3/10 SN 1000MM	0402174	10
Connection terminal block, Connection method Screw connection, Cross section: 0.5 mm² - 6 mm², Width: 7 mm, Color: silver	AK 4	0404017	50
Connection terminal block, Connection method Screw connection, Cross section: 0.5 mm² - 6 mm², Width: 7 mm, Color: green-yellow	AKG 4 GNYE	0421029	50
Connection terminal block, Connection method Screw connection, Cross section: 0.5 mm² - 6 mm², Width: 7 mm, Color: black	AKG 4 BK	0421032	50

Documentation	Type	Order No.	Pcs./Pkt.
User manual, English, Automation terminals of the Inline product range	IL SYS INST UM E	-	-
Data sheet, English, INTERBUS addressing	DB GB IBS SYS ADDRESS	-	-

4 Technical data

Dimensions (nominal sizes in mm)
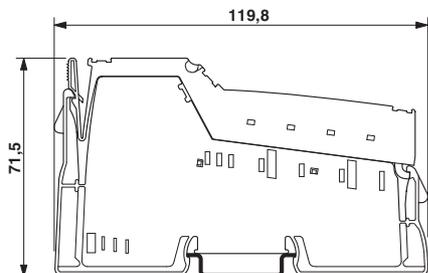
Width	12.2 mm
Height	119.8 mm
Depth	71.5 mm

General data	
Color	green
Mounting type	DIN rail
Ambient temperature (operation)	0 °C ... 55 °C
Ambient temperature (storage/transport)	-25 °C ... 85 °C
Permissible humidity (operation)	10 % ... 95 % (according to DIN EN 61131-2)
Permissible humidity (storage/transport)	10 % ... 95 % (according to DIN EN 61131-2)
Air pressure (operation)	70 kPa ... 106 kPa (up to 3000 m above sea level)
Air pressure (storage/transport)	70 kPa ... 106 kPa (up to 3000 m above sea level)
Degree of protection	IP20
Protection class	III, IEC 61140, EN 61140, VDE 0140-1

Connection data	
Designation	Inline connector
Connection method	Spring-cage connection
Conductor cross section solid / stranded	0.08 mm² ... 1.5 mm² / 0.08 mm² ... 1.5 mm²
Conductor cross section [AWG]	28 ... 16
Stripping length	8 mm

Interface Inline local bus	
Connection method	Inline data jumper
Transmission speed	500 kbps

Interface RS-232	
Connection method	Spring-cage connection
Transmission speed	110 bps ... 38400 bps (configurable)
Transmission physics	Copper
Input buffer	4 kByte
Output buffer	1 kByte
Data bits	6 ... 8
Stop bits	1 or 2

Power consumption	
Communications power U_L	7.5 V (via voltage jumper)
Current consumption from U_L	typ. 70 mA max. 90 mA
Power consumption	typ. 525 mW (at U_L) max. 675 mW (at U_L)

This terminal takes no current from potential jumpers U_M, U_S, and U_{ANA}.

Programming Data	
ID code (hex)	BF
ID code (dec.)	191
Length code (hex)	07
Length code (dec.)	07
Process data channel	112 Bit
Input address area	14 Byte
Output address area	14 Byte
Parameter channel (PCP)	0 Byte
Register length	112 Bit

Electrical isolation/isolation of the voltage areas	
Test section	**Test voltage**
Serial interface/7.5 V supply (bus logic)	500 V AC, 50 Hz, 1 min.
Serial interface/24 V supply U_M	500 V AC, 50 Hz, 1 min.
Serial interface/functional earth ground	500 V AC, 50 Hz, 1 min.
7.5 V supply (bus logics) / functional earth ground	500 V AC, 50 Hz, 1 min.
24 V supply (U_M)/functional earth ground	500 V AC, 50 Hz, 1 min.

Error messages to the higher level control or computer system
None

Approvals
For the latest approvals, please visit phoenixcontact.net/products.

5   Internal circuit diagram
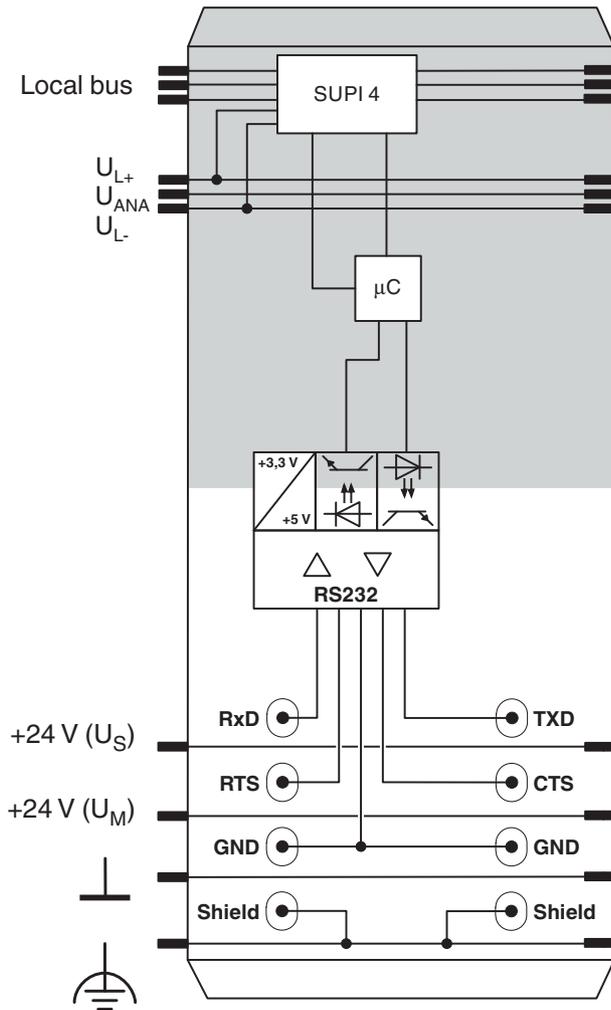
Figure 1   Internal wiring of the terminal points

Key:

SUPI 4   Protocol chip

µC   Microcontroller
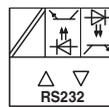
RS232   RS-232 driver block with integrated electrically isolated power supply unit and signal isolation

Electrically isolated area

i   Explanation for other used symbols has been provided in the IL SYS INST UM E user manual.

6   Electrical isolation
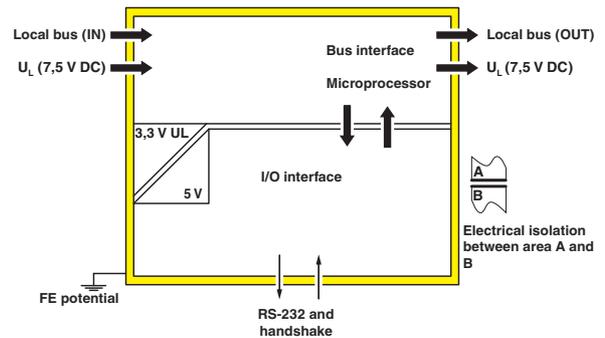
Figure 2   Electrical isolation of the individual function areas

7 Terminal point assignment
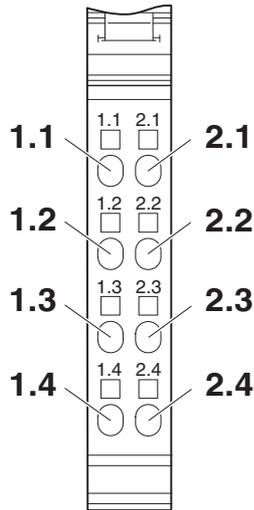
Figure 3 Terminal point assignment

Terminal point	Signal	Description
1.1	RxD	Receive data
2.1	TxD	Transmit data
1.2	RTS	Request to send; handshake signal; output
2.2	CTS	Permission to send; connected device ready to receive; handshake signal; input
1.3, 2.3	GND	Ground for serial interface
1.4, 2.4	Shield	Shield connection

1.3, 2.3	Internally jumpered
1.4, 2.4	Internally jumpered

8 Local diagnostic and status indicators

Figure 4 Local diagnostic and status indicators

Designation	Color	Meaning
D	Green	Diagnostics (bus and logic voltage)
RxD	Yellow	Terminal receives data from the connected device
TxD	Yellow	Terminal transmits data to the connected device

i	For detailed information on diagnostics, please refer to the IL SYS INST UM E user manual.

Function identification

Orange

9 Connection note

In order to increase noise immunity, Phoenix Contact recommends connecting the shield on both sides.

10 Connection examples

ℹ️ Connect the shielding with the shielding accessories given in the ordering data.

## 10.1 RS-232: With 4-wire handshake
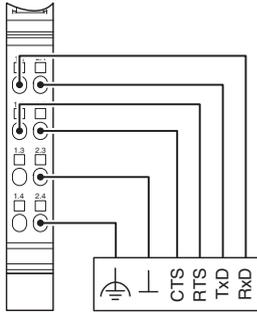
Figure 5   RS-232 interface wiring: four-wire handshake

10.2 RS-232: Without handshake
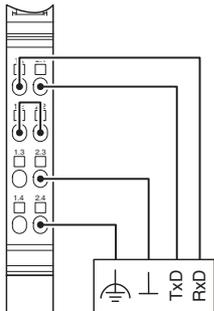
Figure 6   RS-232 interface wiring: without handshake

11 RS-232 serial interface

The RS-232 interface on the terminal represents some form of DTE (data terminal equipment).
This means that terminal point 2.1 (TxD) is always used to transmit and terminal point 1.1 (RxD) is always used to receive.

According to the standard, some form of DCE (data communication equipment) should be connected to the RS-232 interface as a peer. DTE can also be connected.

Measuring the voltage between the connection points for the TxD and GND signals in idle state will determine whether the device to be connected to the RS-232 interface is a form of DTE or DCE. If the voltage measures approximately -5 V, the device is a form of DTE. If the voltage is approximately 0 V, the device is a form of DCE.

Example: when using a 25-pos. standard connector the voltage between pin 2 (TxD) and pin 7 (GND) must be measured.

RS-232 module handshake signals

Any device with a RS-232 interface can be connected to the RS-232 interface on the terminal.

Both the terminal and the device connected to the RS-232 interface can act as a transmitter and a receiver for data exchange.

To avoid errors during data exchange, e.g., a buffer overrun, the handshake is used as a procedure for the mutual signaling of clear to receive and clear to send.

The terminal supports the RTS and CTS handshake signals. Each uses one wire of the connecting cable.

The connecting signals are described from the point of view of the terminal, i.e., from the point of view of the DTE.

12 Mass storage and transmission

The terminal stores the received serial data in an intermediate buffer until it is requested from the serial interface by the bus controller board or the device. Serial data traffic is processed according to the transparent protocol.

12.1 Supported protocols

Protocol	Receive memory	Transmit memory
Transparent	4096 bytes	1023 bytes

12.2 Transparent protocol

If the transparent protocol is used, serial data is transmitted in the same format it is received from the serial interface or from the bus side.

The transmit FIFO (first-in, first-out memory) can store 1023 bytes (1 kbyte) and the receive FIFO can store 4096 bytes (4 kbytes). If the terminal receives another character after the 4095th character, the error pattern is stored in the receive FIFO. All other subsequent characters are ignored.

The protocol supports a hardware handshake. When sending, the system sets the output signal RTS to High and waits until the input signal CTS goes to logic High.
Once all characters have been sent, RTS goes back to Low.

13   Process data

The module uses seven words of IN process data and seven words of OUT process data.

For the assignment of the illustrated (byte.bit) view to your INTERBUS control or computer system, please refer to the DB GB IBS SYS ADDRESS data sheet.						

Word	0		1		2		3		4		5		6	
Byte in the Motorola format	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Byte in the Intel format	1	0	3	2	5	4	7	6	9	8	11	10	13	12
OUT	Control word		L	D	D	D	D	D	D	D	D	D	D	D
	K/P	S												
IN	Status word		L	D	D	D	D	D	D	D	D	D	D	D
	K/P	S												

K/P   Command/parameter

S   Control bits (OUT) or status bits (IN)

L   Length: Number of characters to be written (OUT) or to be read (IN)

D   Data

The byte representation in the Motorola format, also called Big Endian (high order byte at starting address) corresponds to the INTERBUS standard representation. All byte representations in the data sheet have this format.	
The byte representation in the Intel format is also called Little Endian (low order byte at starting address).	

The command is used to determine the function. The actually transmitted data depends on the command.

13.1 Control word

						OUT0									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	Command			OUT parameter				0	0	0	0	0	0	0	0

Commands and OUT parameters

Bit 15	Code (bin)	Code (hex)	Command and OUT parameter
	Bit 14 ... 12	Bit 15 ... 12	
0	000	0	Read number of characters received
0	001	1	Transmit characters
0	010	2	Store characters temporarily
0	011	3	OUT parameter = 0_{hex}: Read characters
			OUT parameter = C_{hex}: Read firmware version
			OUT parameter = D_{hex}: Read configuration
			OUT parameter = E_{hex}: Read counters
0	100	4	Write configuration
0	101	5	Toggling command 1: Transmit characters
0	110	6	Toggling command 2: Store characters temporarily
0	111	7	Toggling command 3: Read characters

Command toggling

Command toggling is used to execute a command on a terminal again. In this way, a second command code is available for the same function.

This applies for the following commands:
– Transmit characters
– Store characters temporarily
– Read characters

Here, bit 14 is used for toggling. If, for example, you wish to transmit character strings in sequence, use command code 001_{bin} for the first transmission and then use 101_{bin} and 001_{bin} alternately.

13.2   Status word

IN0															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
St	Command			IN parameter				Status bits							

St   Error bit

Reasons for an error bit set:

– Invalid parameter for the specified command

– For transmit command: incorrect length

Status bits

7	6	5	4	3	2	1	0
0	Transmit buffer not empty	Transmit buffer full	Receive buffer full	0	0	0	Receive buffer not empty

Transmit buffer not empty	
Code (bin)	**Meaning**
0	Empty
1	Not empty, characters to be sent are still available

Transmit buffer full	
Code (bin)	**Meaning**
0	Not full
1	Full

Receive buffer full	
Code (bin)	**Meaning**
0	Not full
1	Full

Code (bin)	**Meaning**
Receive buffer not empty	
0	Empty
1	Not empty, characters to be read are available

14 Commands

14.1 "Read number of characters received" command

The command result is the number of characters that have been received but not yet read.

This command can be used to first reach a minimum number of characters before transmitting the "Read characters" command.

The number is a 16-bit value and is mapped to word 1.

Process data assignment for the "Read number of characters received" command (process data width: 7 words)

Word	0		1		2		3		4		5		6	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13
OUT	00_{hex}	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
IN	00_{hex}	Status bits	Number of characters received		xx	xx	xx	xx	xx	xx	xx	xx	xx	xx

14.2 "Transmit characters" command

The transmit data located in the process data is stored in the transmit memory. From there the data is transmitted directly via the interface.

A maximum of eleven characters can be transmitted.

Specify the number of characters to be transmitted in the third byte.

If there are characters in the intermediate buffer, these are transmitted first.

After the command has been executed successfully, the intermediate buffer is cleared.

Process data assignment for the "Transmit characters" command with 11 characters (Z1 - Z11; process data width: 7 words)

Word	0		1		2		3		4		5		6	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13
OUT	10_{hex}	xx	$0B_{hex}$	Z1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
IN	10_{hex}	Status bits	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx

Reasons for an error bit set:

– Byte 2 (number of characters to be transmitted)
= 0 or > maximum user data length (11 characters)

14.3 "Store characters temporarily" command

If more than 11 characters are to be transmitted, the transmit data located in the process data is stored in an intermediate buffer which can store up to 330 characters. No characters are transmitted. The "Transmit characters" command is used to transmit temporarily stored data. In this way, blocks of up to 330 characters can be transmitted without a break. They are divided over 30 telegrams with 11 characters each, for example.

Reasons for an error bit set:

– Byte 2 (number of characters to be transmitted)
  = 0 or > maximum user data length (11 characters)
– Not enough space in the intermediate buffer

14.4 "Read characters" command

A maximum of eleven characters can be read.

Process data assignment for the "Read characters" command with eleven characters (C1 - C11)

Word	0		1		2		3		4		5		6	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13
OUT	30_{hex}	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
IN	30_{hex}	Status bits	$0B_{hex}$	Z1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11

14.5 "Read counters" command

This command can be used to read several counters. The counters are used for interface diagnostics.

Process data assignment for the "Read counters" command

Word	0		1		2		3		4		5		6	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13
OUT	$3E_{hex}$	00_{hex}	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx	xx
IN	$3E_{hex}$	Status bits	Number of valid characters received		Number of invalid characters received (parity, overrun or framing errors)		Number of characters transmitted		xx	xx	xx	xx	xx	xx

14.6 "Read firmware version" command

For control word $3C00_{hex}$, word IN1 supplies the firmware version and the module ID.

Example: Firmware version 1.23

Word	IN1															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Assignment (hex)	1				2				3				8			
Meaning	Firmware version												Module ID			

14.7 "Write configuration" command

Output words 0 to 6

Process data assignment for the "Write configuration" command

Word	0		1		2		3		4		5		6	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13
OUT	40$_{hex}$	xx	res.	B/DB	res.	res.	ErrP	res.	res.	res.	res.	res.	res.	res.
IN	40$_{hex}$	Status bits	-	-	-	-	-	-	-	-	-	-	-	-

Element value range

i	The options in bold are default settings.

Byte 3 (B/DB: baud rate/data width)

Baud rate	
Code (hex)	**Wert (baud)**
0	110
1	300
2	600
3	1200
4	1800
5	2400
6	4800
7	**9600**
8	15625
9	19200
A	38400
Other	Reserved

Data width			
Code (hex)	**Meaning**		
	Data bits	**Parity**	**Stop bits**
0	7	Even	1
1	7	Odd	1
2	**8**	**Even**	**1**
3	8	Odd	1
4	8	Without	1
5	7	Without	1
6	7	Even	2
7	7	Odd	2
8	8	Even	2
9	8	Odd	2
A	8	Without	2
B	7	Without	2
C	8	Constantly at 0	1
D	8	Constantly at 1	1
E	6	Without	1
F	Reserved		

Byte 6 (ErrP: error pattern)

The error pattern contains the character that is written to the FIFO if a character was received with errors (e.g., in the event of a parity error) or the receive memory is full.

Error pattern	
Code (hex)	**Meaning**
24	**$**
xx	Any character
00	If a character is received with an error, no error pattern is stored.
FF	The invalid character is stored instead of the error pattern.