

AXL SE RS485 EF

**Axioline Smart Elements, communication module,
RS-485 interface: 1,
process data width: 64 bytes**

Data sheet
111309_en_00

© Phoenix Contact

2024-09-23



1 Description

You can integrate Axioline Smart Elements into systems with the Smart Element interface.
This Smart Element is used to operate standard I/O devices with a serial interface on a bus system.

Features

- One serial input and output channel in RS-485 format
- Various protocols supported
- Transmission speed can be set up to 230,400 bps
- Number of data bits, stop bits and parity can be set
- Modbus/RTU client and server support
- Process data width: 64 bytes
- Device rating plate stored

Observe these notes



This data sheet is only valid in association with the UM EN AXL SE SYS INST user manual.



Make sure you always use the latest documentation.
It can be downloaded at: phoenixcontact.com/product/1507978

2 Table of contents

1	Description	1
2	Table of contents	2
3	Ordering data	3
4	Technical data	3
5	Internal circuit diagram	7
6	For your safety	8
7	Terminal point assignment as well as diagnostics and status indicators	9
8	Connection examples	10
9	Connection notes	11
10	RS-485 serial interface	12
11	Data storage and transmission	13
12	Process data	16
13	Process data word 0	17
14	Commands	20
15	Parameter, diagnostics and information (PDI).....	24
16	Standard objects	24
17	Application objects	27
18	Device descriptions	31

3 Ordering data

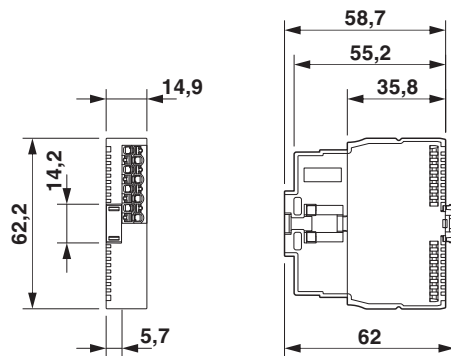
Description	Type	Item no.	Pcs./Pkt.
Axioline Smart Elements, Communication module, interface RS-485: 1, Process data width: 64 bytes, Various protocols supported, degree of protection: IP20	AXL SE RS485 EF	1507978	1
Documentation	Type	Item no.	Pcs./Pkt.
User manual, English, Axioline Smart Elements	UM EN AXL SE SYS INST	-	-
User manual, English, Axioline F: System and installation	UM EN AXL F SYS INST	-	-
User manual, English, Axioline F: Diagnostic registers, and error messages	UM EN AXL F SYS DIAG	-	-
Application note, English, Transmitting and receiving data with devices featuring a serial interface	AH EN RS-232/RS-485/RS-UNI	-	-

Additional ordering data

For additional ordering data (accessories), go to: www.phoenixcontact.com/product/1507978

4 Technical data

Dimensions (nominal sizes in mm)



Width	14.9 mm
Height	62.2 mm
Depth	62 mm

General data

Color	Housing: traffic grey A (RAL 7042)
Weight	33 g
Ambient temperature (operation)	-25 °C ... 60 °C
Ambient temperature (storage/transport)	-40 °C ... 85 °C
Permissible humidity (operation)	5 % ... 95 % (non-condensing)
Permissible humidity (storage/transport)	5 % ... 95 % (non-condensing)
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)
Overvoltage category	II (IEC 60664-1, EN 60664-1)
Degree of pollution	2 (IEC 60664-1, EN 60664-1)
Mounting type	Smart Element slot
Mounting position	See the system in which the Smart Element is used.



Do not use the Smart Element in an atmosphere that contains corrosive gas.

Connection data: I/O

Connection method	Push-in connection
Conductor cross section, rigid	0.25 mm ² ... 1.5 mm ²
Conductor cross section, flexible	0.25 mm ² ... 1.5 mm ²
Conductor cross section [AWG]	24 ... 16
Conductor cross section flexible, with ferrule with plastic sleeve	0.25 mm ² ... 1.5 mm ²
Conductor cross section flexible, with ferrule without plastic sleeve	0.25 mm ² ... 1.5 mm ²
Stripping length	8 mm



Please observe the information provided on conductor cross sections in the "Axioline Smart Elements" user manual.

Interface: Smart Element interface

Number of interfaces	1
Connection method	Card edge connector
Start time until ready to operate	< 500 ms

Interface: RS-485

Number of interfaces	1
Connection method	Push-in connection
Note on the connection method	Use shielded cables.
Transmission speed	100 bps ... 230400 bps (can be parameterized)
Transmission physics	Copper
Supported protocols	Transparent End-to-end Modbus/RTU (client support) Modbus/RTU (server support)
Input buffer	4 kByte
Output buffer	1 kByte
Data bits	7 or 8
Stop bits	1 or 2
Parity	Even, odd or no parity
Termination resistor	120 Ω (active, integrated)
Process data update	540 μ s

Communications power supply of the Smart Elements (U_{SE})

Supply voltage	using card edge connectors
Current draw	See documentation for the system in which the Smart Element is used.

I/O supply (U_P)

Nominal supply voltage	24 V DC (using card edge connectors)
Supply voltage range	19.2 V DC ... 30 V DC (including all tolerances, including ripple)
Current consumption	min. 6 mA (without connected peripherals) typ. 8 mA max. 10 mA
Surge protection	See the system in which the Smart Element is used.
Reverse polarity protection	Polarity protection diode
Protection	See the system in which the Smart Element is used.

Power dissipation

Maximum power dissipation for nominal condition	0.7 W
---	-------

Input and output address area

Input address area	64 Byte
Output address area	64 Byte

Configuration and parameter data in a PROFIBUS system

Required parameter data	17 Byte
Required configuration data	7 Byte

Electrical isolation/isolation of the voltage areas

Test section	Test voltage
Communications supply / 24 V supply (I/O)	500 V AC, 50 Hz, 1 min.
Logic supply / RS-485 interface	500 V AC, 50 Hz, 1 min.
Communications supply / functional ground	500 V AC, 50 Hz, 1 min.
24 V supply (I/O) / RS-485 interface	500 V AC, 50 Hz, 1 min.
24 V supply (I/O) / functional ground	500 V AC, 50 Hz, 1 min.
RS-485 interface / functional ground	500 V AC, 50 Hz, 1 min.

Mechanical tests

Vibration resistance in accordance with EN 60068-2-6/IEC 60068-2-6	5g
Shock in accordance with EN 60068-2-27/IEC 60068-2-27	30g
Continuous shock in accordance with EN 60068-2-27/IEC 60068-2-27	10g

Conformance with EMC Directive 2014/30/EU
Immunity test in accordance with EN IEC 61000-6-2

Electrostatic discharge (ESD) IEC 61000-4-2	Criterion B, ± 6 kV contact discharge, ± 8 kV air discharge
Electromagnetic fields IEC 61000-4-3	Criterion A, Field intensity: 10 V/m
Fast transients (burst) IEC 61000-4-4	Criterion B, ± 2 kV
Transient overvoltage (surge) IEC 61000-4-5	Criterion B, shielded I/O cables: ± 1 kV asymmetrical
Conducted interference IEC 61000-4-6	Criterion A, Test voltage 10 V

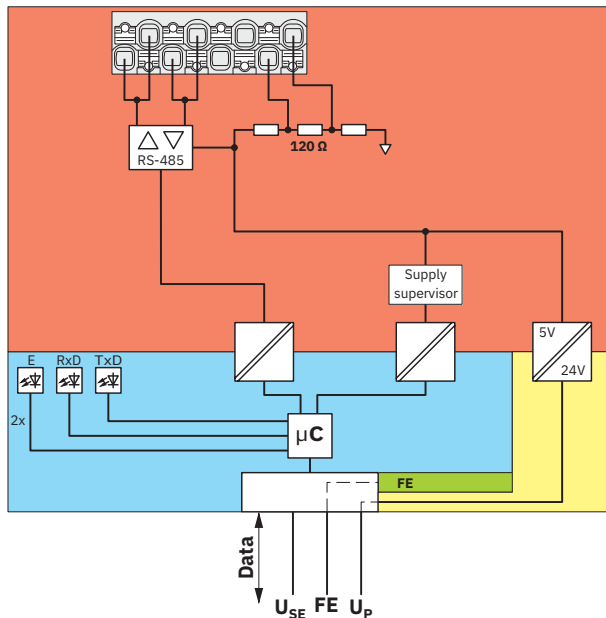
Noise emission test in accordance with EN IEC 61000-6-4	Class A
--	---------

Approvals and manufacturer's declarations

The latest documents can be found at:	www.phoenixcontact.com/product/1507978
---------------------------------------	--

5 Internal circuit diagram

Figure 1 Internal wiring of the terminal points



Key:

Data Data transmission

U_{SE} Communications power supply of the Smart Element

FE Functional ground

U_P I/O supply of the Smart Element



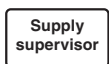
Microcontroller



Electrical isolation for data or power supply



LED



Monitoring the supply voltage



RS-485 interface active



Electrically isolated areas

6 For your safety

6.1 Intended use

Use Smart Elements exclusively in accordance with the specifications in the data sheet and the “Axioline Smart Elements” user manual.

Please also refer to the documentation for the system in which the Smart Elements are used.

If the equipment is used in a manner not specified, the protection provided by the equipment may be impaired.

6.2 Qualification of users

The use of products described in this data sheet is oriented exclusively to electrically skilled persons or persons instructed by them. The users must be familiar with the relevant safety concepts of automation technology as well as applicable standards and other regulations.

6.3 Installation



CAUTION: Fire hazard

- The device must be installed in the final protective housing, which provides sufficient resistance to mechanical strain and protection against the spreading of fire in accordance with the standards UL/IEC/EN 61010-1 and UL/IEC/EN 61010-2-201.
- The external circuits intended to be connected to this device must be galvanically separated from mains supply or hazardous live voltage by reinforced or double insulation and meet the requirements of SELV/PELV (Class III) circuits in accordance with UL/CSA/IEC/EN 61010-1, UL/CSA/IEC/EN 61010-2-201.

6.4 Disconnecting or plugging in a Smart Element



NOTE: Damage to contacts or malfunction

Before performing work on a Smart Element, disconnect the power to the Smart Element.

This means:

- Disconnect the connected I/O devices from the power.
- Switch off the I/O supply voltage U_P !
- Switch off the communications power U_{SE} . For the system in which the Smart Element is used, this means the following: Switch off the voltage that generates the U_{SE} .

6.5 Strain relief



NOTE: damage to the contacts

Physical overloads can result in damage to the terminal points.

- Relieve strain in the connected cables.

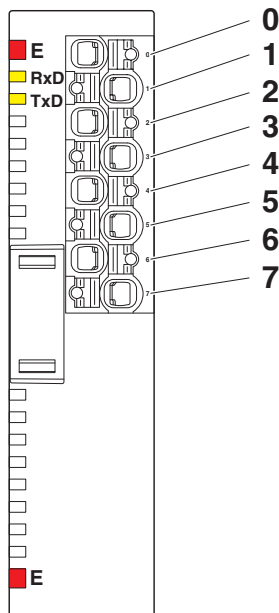
6.6 Locking a Smart Element

Make sure that each Smart Element is locked in its slot. This is only ensured if the unlocking mechanism has been pushed into the guide as far as it will go.

See “Axioline Smart Elements” user manual.

7 Terminal point assignment as well as diagnostics and status indicators

Figure 2 Terminal point assignment as well as diagnostics and status indicators



7.1 Terminal point assignment

Terminal point	Signal	Description	
0	RxD+, TxD+	Receive/transmit data	positive
1	RxD+, TxD+	Receive/transmit data	positive
2	RxD-, TxD-	Receive/transmit data	negative
3	RxD-, TxD-	Receive/transmit data	negative
4	-	Not used	
5	-	Not used	
6	R+	Termination resistor	Positive pole
7	R-	Termination resistor	Negative pole

7.2 Local diagnostics and status indicators

Designation	Color	Description
E	Red	Error
		Off No error
		Flashing (0.5 Hz) Error in Smart Element Replace the Smart Element.
		Flashing (4 Hz) Communication error Check whether the Smart Element has been plugged in correctly.
RxD	Yellow	I/O error Check the connected components and wiring. Remove the error.
		On The Smart Element is receiving data from the connected device.
		Off The Smart Element is not receiving any data.
TxD	Yellow	Receiving data
		On The Smart Element is transmitting data to the connected device.
		Off The Smart Element is not transmitting any data.

See also “Diagnostic state (0018_{hex}: DiagState)” section, “Possible error codes” table.

8 Connection examples

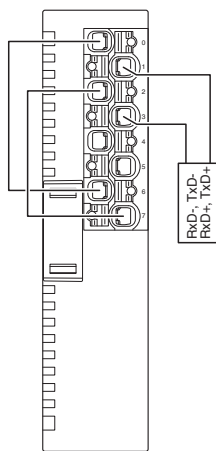
Special feature	Notes
Smart Element as the network end point	Termination resistor required
Smart Element in the middle of a network	Termination resistor not required

8.1 Smart Element as the network end point



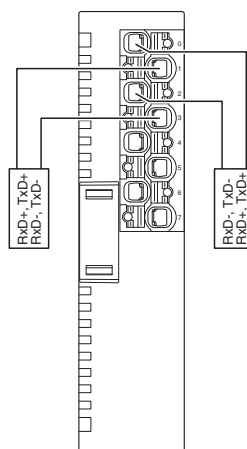
Fit a termination resistor to the RS-485 network at each endpoint.

Figure 3 Smart Element as the network end point



8.2 Smart Element in the middle of a network

Figure 4 Smart Element in the middle of a network



9 Connection notes



Observe the connection instructions of the manufacturer of the device to be connected.

Termination resistor

Fit the receive signals of the RS-485 network with a termination resistor at the relevant end point.

If you use the integrated termination resistor, the polarization of the data cable will also be active.

Shielding

For installation in a control cabinet: Connect the cable shield to the functional ground at a suitable point immediately after entry into the control cabinet. Route the cable in the control cabinet in a shielded manner.

If a closed control cabinet is not available, connect the shield to a shield bus.

Connect the shielding in accordance with the specifications for the system in which you are using the Smart Element.

Within an Axioline F station, the AXL SHIELD SET Axioline shield connection set is available for optimal connection directly in front of the module, see user manual UM EN AXL F SYS INST.

In general, you can use Phoenix Contact products for shielding, see www.phoenixcontact.net/webcode/#0845.

Strain relief

Do not use the shield contact as a strain relief. Carry out the shielding and the strain relief separately.

10 RS-485 serial interface

With RS-485, you can create a network with several devices using an existing network consisting of two signal lines.

Use a twisted-pair, common shielded data cable to connect the devices. Fit a termination resistor to the data cable at both end points of the RS-485 network. For this, you can use the integrated termination resistor in the Smart Element via connections R+ and R-.

If you use the integrated termination resistor, the data cable will also be polarized. This will generate a defined cable idle level.

The Smart Element supports only half duplex transmission. Make sure that data is not sent simultaneously by several devices.

11 Data storage and transmission

The Smart Element stores the received serial data in an intermediate buffer. The data is then requested at the serial interface by the higher-level system or the device.

The serial data traffic is implemented in accordance with various protocols. The protocol used depends on the protocol supported by the remote partner.

11.1 Supported protocols

Protocol	Receive memory	Transmit memory	Special features when receiving
Transparent	4096 bytes	1023 bytes	
End-to-end	3 buffers with 340 bytes each	1023 bytes (including end characters)	Two end characters are filtered out
Modbus/RTU (client support)	1 buffer with 256 bytes	1 buffer with 256 bytes	
Modbus/RTU (server support)	1 buffer with 256 bytes	1 buffer with 256 bytes	

11.2 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 kB) and the receive FIFO can store 4096 bytes (4 kB). If the Smart Element receives another character after the 4095th character, the 24_{hex} (\$) character is stored in the receive FIFO. All other subsequent characters are ignored.

11.3 End-to-end protocol

The serial data is conditioned for the end-to-end protocol. If serial data is sent from the bus side, two additional characters, the first and second delimiters, are attached for transmission to the serial interface.

A block of serial data transmitted by the serial interface is only valid if the Smart Element has received the first and second delimiters. Only then can the data be read via the bus side. The delimiters are not forwarded to the higher-level bus with the user data; they are filtered out.

Unlike in the transparent protocol, the receive memory is not organized as a FIFO but as a buffer.

There are three buffers, each with 340 bytes.

If the maximum buffer size is exceeded without the two delimiters being detected, the previous characters will be ignored and the buffer will be written again.

The transmit FIFO can store 1023 bytes (1 kbyte). The delimiters are attached to, and stored with, the data to be sent.

If a received telegram is read via the process data, it can happen that the telegram length is greater than 61 bytes. In this case, the MSB (most significant bit) is set in the length byte of the response (third input data byte). With the MSB set, the length specification 61_{dec} = 3D_{hex} turns into the value BD_{hex}. This indicates that the telegram has not been read in full yet.

11.4 Modbus/RTU

The “Modbus/RTU (client support)” operating mode provides support for a Modbus/RTU client that is implemented in the form of a function block on a controller, for example.

The “Modbus/RTU (server support)” operating mode provides support for a Modbus/RTU server.

The Smart Element does not contain any programmable logic.

In the Modbus environment, two time values play an important role: 1.5 characters, also referred to as t1.5 and 3.5 characters, also referred to as t3.5. The times for baud rates above 19,200 baud are set to t1.5 = 1.0 ms and t3.5 = 2.0 ms.

These two times describe the pause periods during which nothing is transmitted or received on the serial cable.

Thus, it can be determined whether a new telegram is involved or an active telegram is interrupted.

If a break between two characters is greater than t1.5 and less than t3.5, this is identified as a transmission error. The message is considered incomplete and is indicated with a receive error.

If the break after a character is greater than t3.5, this is identified as the telegram end.

The first character after a t3.5 pause is used as the address ID. Only the addressed Modbus/RTU server may respond to a telegram request from a client.

One transmit and one receive buffer with 256 bytes each are available for the “Modbus/RTU (client support)” protocol.

One transmit and one receive buffer with 256 bytes each are available for the “Modbus/RTU (server support)” protocol.

Telegram structure

Address	Function	Data	CRC
1 byte	1 byte	0 ... 252 bytes	2 bytes
Data			Automatic generation and check

With the Modbus/RTU server, only one telegram is received at a time. If a new telegram is received and the previous telegram has not been read yet, the previous telegram is overwritten. The Smart Element reports the overwriting with the “Receive buffer overwritten by new telegram” receive error. See Section “Status word, receive error”.

If a received telegram is read via the process data, it can happen that the telegram length is greater than 61 bytes. In this case, the MSB (most significant bit) is set in the length byte of the response (third input data byte). With the MSB set, the length specification 61_{dec} = 3D_{hex} turns into the value BD_{hex}. This indicates that the telegram has not been read in full yet.

11.4.1 Data transfer sequence from the perspective of the Modbus/RTU client

Transmit

The two checksum bytes are automatically generated and transmitted for the user data supplied via the bus. Once this has been sent, a response is expected.

If the response to a unicast query (Modbus server address > 0) is not received within the time parameterized via the Modbus response timeout object (050D_{hex}), a transmit error is generated by setting the “Transmit error” error bit in the status word.

In case of a broadcast request (Modbus server address = 0), the response timer (Modbus response timeout) is not started because the Modbus server does not respond to broadcast requests.

Receive

The Smart Element can only receive data when it is waiting for a response. As soon as it has received the response or the timeout time has elapsed, the receiver is blocked again.

If the response is received in time, the checksum is checked.

If one of the following errors occurs, the “Receive error” bits are set in the status word.

- The checksum is invalid.
- A parity error has occurred.
- The time t1.5 has elapsed on receipt.
- The receive buffer was overwritten by a new telegram before the controller had read the previous telegram via process data.

If the data is received without errors, the “Receive buffer not empty” bit is set in the status word.

The checksum bytes are not included when reading the receive data.

11.4.2 Data transfer sequence from the perspective of the Modbus/RTU server

Receive

If one of the following errors occurs, the “Receive error” bits are set in the status word.

- The checksum is invalid.
- A parity error has occurred.
- The time t1.5 has elapsed on receipt.
- The receive buffer was overwritten by a new telegram before the controller had read the previous telegram via process data.

If the data is received without errors, the address in the Modbus telegram is compared with the Modbus/RTU server address stored in the Smart Element.

You can parameterize the permitted address using the “Filter for Modbus/RTU address” object (050C_{hex}).

If the addresses match, the “Receive buffer not empty” bit is set in the status bit.

The checksum bytes are not included when reading the receive data.

Transmit

The two checksum bytes are automatically generated and transmitted for the user data supplied via the bus.

12 Process data

The Smart Element is parameterized via the PDI channel.

The data exchange with the higher-level controller is carried out via process data.

Each Smart Element occupies 64 bytes of process data in input and output direction, of which a maximum of 61 bytes are user data.

Assignment of the process data words

Word	0		1		2		...	31	
Byte (Motorola format)	0	1	2	3	4	5	...	62	63
Byte (Intel format)	1	0	3	2	5	4	...	63	62
OUT	K/P	S	L	D	D	D	...	D	D
IN	K/P	S	L	D	D	D	...	D	D

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

The byte representation in 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 Process data word 0

13.1 Control word

Word	0															
Name	Control word															
	Control word high byte								Control word low byte							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Assignment	0	Command			OUT parameter				Control bits							

Control bits

7	6	5	4	3	2	1	0
0	0	0	0	0	Reset transmit error	Reset receive error	0
					Reset_Tx_error	Reset_Rx_error	

Reset transmit error		
Code (bin)	Meaning	Protocol
0	No action	Modbus/RTU
1	Reset transmit error	

Reset receive error		
Code (bin)	Meaning	Protocol
0	No action	Modbus/RTU
1	Reset receive error	
	All three error bits for receive errors (bit 7, 3, 1) are reset.	

13.2 Status word

Word	0															
Name	Status word															
	Status word high byte								Status word low byte							
Bit	15	14	13	12	11	10	9	8	7	6	6	4	3	2	1	0
Assignment	St	Command			IN parameter				Status bits							

Bit	Meaning	Name
St	Error bit	Error

Reasons for an error bit set:

- Invalid parameter for the specified command
- Failure of I/O supply voltage U_P
- With “Transmit characters” and “Store characters temporarily” commands
- Byte 2 (number of characters to be transmitted) = 0 or > maximum user data length
- Not enough space in the transmit buffer or intermediate buffer

Status bits

7	6	5	4	3	2	1	0
Receive error	Transmit buffer not empty	Transmit buffer full	Receive buffer full	Receive error	Transmit error	Receive error	Receive buffer not empty
Rx_error_3	Tx_buf_not_empty	Tx_buf_full	Rx_buf_full	Rx_error_2	Tx_error	Rx_error_1	Rx_buf_not_empty

Transmit buffer

The transmit buffer flags only relate to the transmit buffer, not the intermediate buffer.

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

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

Protocol	Meaning: transmit buffer full
Modbus/RTU	No more buffer space
Transparent	Space remaining in the transmit buffer \leq 30 characters
End-to-end	Space remaining in the transmit buffer \leq 30 characters

Receive buffer

Receive buffer not empty		
Code (bin)	Meaning	Protocol
0	Empty	All
1	Not empty; characters to be read are available	

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

Protocol	Meaning: receive buffer full
Modbus/RTU	No more buffer space
Transparent	Space remaining $<$ 15 characters
End-to-end	No more buffer space

Transmit error

Transmit error		
Code (bin)	Meaning	Protocol
0	No transmit error	Modbus/RTU (client support)
1	Transmit error; timeout has elapsed with no response received	

Receive error

Receive error				
Bit 7	Bit 3	Bit 1	Meaning	Protocol
0	0	0	No receive error	Modbus/RTU
0	0	1	Invalid checksum	
0	1	1	Parity error	
1	0	1	t1.5 elapsed	
1	1	0	Receive buffer overwritten by new telegram	

Depending on the error, a receive error is indicated via bits 7, 3, and 1.

To reset a receive error, set the "Reset receive error" bit in the control word.

"Receive buffer overwritten by new telegram":

If the receive buffer is overwritten by a new telegram before all data from the previously received telegram has been read, the Smart Element reports the "Receive buffer overwritten by new telegram" receive error.

14 Commands



The application note AH DE RS-232/RS-485/RS-UNI describes an example of which sequences are required to send or receive data using a device with a serial interface. See: "Ordering data, documentation."

Code (bin)	Code (hex)	Command
000	0	Read number of characters received and fill level of the receive buffer
001	1	Transmit characters
010	2	Store characters temporarily
011	3	OUT parameter = 0 _{hex} : Read characters
		OUT parameter = E _{hex} : Read counters
100	4	Reserved
101	5	Toggling command 1: Transmit characters
110	6	Toggling command 2: Store characters temporarily
111	7	Toggling command 3: Read characters

Command toggling

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

This applies for the following commands:

- Read counters
- Transmit characters
- Store characters temporarily
- Read characters
- Read counters

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.

14.1 “Read number of characters received and fill level of the receive buffer” command

For the transparent protocol, the command result is the number of characters that have been received but not yet read.

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

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

The fill level of the receive buffer is specified in byte 4 as the command result.

Process data assignment

Word	0		1		2		...	31	
Byte	0	1	2	3	4	5	...	62	63
OUT	00 _{hex}	xx	xx	xx	xx	xx	...	xx	xx
IN	00 _{hex}	Status bits	Number of characters received		Fill level	xx	...	xx	xx

Fill level			
Byte 4	Protocol		
	Transparent	End-to-end	Modbus/RTU
	Number of kbytes free	Number of buffers free	Number of buffers free
00 _{hex}	≥3, max. 4	3	1
01 _{hex}	< 3	2	-
02 _{hex}	< 2	1	-
03 _{hex}	< 1	0	0
Other values are not permitted.			

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 61 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 via the selected RS interface.

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

Process data assignment with 61 characters (Z1 - Z61)

Word	0		1		2		...	31	
Byte	0	1	2	3	4	5	...	62	63
OUT	10 _{hex}	xx	61 _{dez}	Z1	Z2	Z3	...	Z60	Z61
IN	10 _{hex}	Status bits	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 (61 characters)
- Not enough space in the intermediate buffer

14.3 "Store characters temporarily" command

If more than 61 characters are to be transmitted, the transmit data located in the process data is stored in an intermediate buffer that can store up to 340 characters. No characters are transmitted. The "Transmit characters" command is used to transmit the temporarily stored data. In this way, blocks of up to 340 characters can be transmitted without a break. These are divided, for example, into five process data sequences with 61 characters each and one sequence with the remaining 35 characters.

Process data assignment with 61 characters (Z1 - Z61)

Word	0		1		2		...	31	
Byte	0	1	2	3	4	5	...	62	63
OUT	20 _{hex}	xx	61 _{dez}	Z1	Z2	Z3	...	Z60	Z61
IN	20 _{hex}	Status bits	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 (61 characters)
- Not enough space in the intermediate buffer

14.4 "Read characters" command

A maximum of 61 characters can be read.

Process data assignment with 61 characters (Z1 - Z61)

Word	0		1		2		...	31	
Byte	0	1	2	3	4	5	...	62	63
OUT	30 _{hex}	xx	xx	xx	xx	xx	...	xx	xx
IN	30 _{hex}	Status bits	61 _{dez}	Z1	Z2	Z3	...	Z60	Z61

14.5 "Read counters" command

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

Process data assignment

Word	0	1	...	11	12	...	31
OUT	3E00 _{hex}	-	...	-	-	...	-
IN	3Exx _{hex}	See table below			Reserved		

Word	Contents	Protocol
0	Acknowledgment of the command and status bit (xx)	All
1	Number of valid characters received	All
2	Number of invalid characters received (parity, overrun or framing errors)	
3	Number of characters transmitted	
4	Number of unicast telegrams received without errors	Modbus/RTU
5	Number of broadcast telegrams received without errors	
6	Number of transmitted master telegrams with response	
7	Number of telegrams received with errors	
8	Number of transmitted master telegrams without response	
9	Number of t1.5 timeout errors	
10	Number of checksum errors	
11	Number of telegrams received without errors without address match	
12 ... 31	Reserved	

15 Parameter, diagnostics and information (PDI)

Parameter and diagnostic data as well as other information are transmitted as objects via the PDI channel.

For more detailed information on all possible standard objects for Axioline Smart Elements, please refer to the UM EN AXL SE SYS INST user manual.

The standard objects necessary for operation are described in the following section.

The following applies for the tables below:

Abbreviation	Meaning
Length in bytes	Maximum length of the elements in bytes
R	Read
W	Write
[x]	Number of elements in an array or record



Each visible string is terminated with a null terminator (00_{hex}). The length of a visible-string-type element is therefore at least one byte larger than the number of user data items.

If the number of user data items plus null terminator is smaller than the specified length of the element, the visible string will be populated with a null character (00_{hex}).



You will find detailed information on the PDI objects in the user manual UM EN AXL F SYS INST.

16 Standard objects

Index (hex)	Object name	Data type	Length in bytes	Rights	Meaning/contents	Startup parameters	
Device type							
0037	DeviceType	Octet string	8	R	Device type 0000 0840 0000 1C05 _{hex}	No	
Diagnostics objects							
0018	DiagState	Record [11]	74	R	Diagnostic state	No	*
0019	ResetDiag	UINT8	1	R/W	Handling diagnostic messages	Yes	*
Objects for process data management							
0025	PDIN	Octet string	64	R	Input process data The structure corresponds to the representation in the "Process data" section.	No	
0026	PDOUT	Octet string	64	R	OUT process data The structure corresponds to the representation in the "Process data" section.	No	
Objects for device management							
002D	ResetParam	UINT8	1	R/W	Reset parameterization	No	*

Startup parameters are stored in the non-volatile flash memory.



Exception: object 0019_{hex}

This startup parameter is stored in **volatile** memory.

The objects identified with * in the last column are described in more detail in the following sections.

The description of the other objects is to be found in the user manual UM EN AXL F SYS INST.

16.1 Diagnostics state (0018_{hex}: DiagState)

This object is used for a structured message of an error.

Read off all information via subindex 00 to receive all information on an error number. Access to individual elements of the object is not permitted.

A detailed description of the object is provided in user manual UM EN AXL F SYS INST.

Possible error codes

Element	02	03	04	08	0B		
Error	Priority	Channel	Error code	Function group	Text	E LED	Corrective
	hex	hex	hex				
No error	00	00	0000	General	Status OK	○	
I/O supply voltage (U _p) is not present.	01	FF	3130	General	Supply missing (U _p)	●	Check the supply voltage.
Error in the Smart Element firmware	01	FF	6100	General	Firmware error, update required	●	Replace the Smart Element.
Problem communicating with the Smart Element	01	FF	6130	General	Smart Element missing	☀	Check whether the Smart Element has been plugged in correctly. If the error is still present, replace the Smart Element.
Error in the parameter memory	01	FF	6320	General	Parameter error, repeat parameterization	●	Error in the parameter memory. Parameterize the Smart Element.
Transmit buffer full	02	FF	8152	RS485	Tx buffer full	○	Check the parameters.
Receive buffer full	02	FF	815A	RS485	Rx buffer full	○	Read out the buffer.

Key

Priority	00 _{hex}	No error
	01 _{hex}	Error
	02 _{hex}	Warning
Channel	00 _{hex}	No error
	FF _{hex}	Entire device

LED	○	Off
	●	On
	☀	Flashing (4 Hz)

16.2 Handling diagnostic messages (0019_{hex}: ResetDiag)

You can use this object to specify how the Smart Element should handle diagnostic messages.

Handling diagnostic messages	
Value (hex)	Meaning
00	Permit all diagnostic messages (default)
02	Delete and acknowledge all diagnostic messages that are still pending
06	Delete and acknowledge all diagnostic messages and do not permit new diagnostic messages
Other	Reserved



This startup parameter is stored in **volatile** memory.

Diagnostics are active by default (00_{hex} = Allow all diagnostic messages).

The values 00_{hex} and 06_{hex} are defined as startup parameters.

You can only write the value 02_{hex} from your application program.

16.3 Reset parameterization (002D_{hex}: ResetParam)

Use this object to reset all parameters to the factory default settings (default values) as listed in the “Application objects” section.

To reset the parameters, value 01_{hex} must be transferred during write access.

17 Application objects

Index (hex)	Object name	Data type	Length in bytes	Rights	Meaning/contents	Relevant for protocol
0500	UART_Protocol	UINT8	1	R/W	Protocol	All
0502	UART_Baudrate	UINT32	4	R/W	Baud rate	All
0503	UART_Databits	UINT8	1	R/W	Number of data bits	All
0504	UART_Parity	UINT8	1	R/W	Parity	All
0505	UART_Stopbits	UINT8	1	R/W	Number of stop bits	All
0509	UART_RxErrorProperties	UINT8	1	R/W	Behavior upon faulty character reception	Transparent End-to-end
050B	UART_Delimiter	Octet string	2	R/W	End delimiter	End-to-end
050C	UART_ModbusFilterServer Address	UINT8	1	R/W	Filter for Modbus server address	Modbus/RTU (server support)
050D	UART_ModbusResponse Timeout	UINT16	2	R/W	Modbus response timeout	Modbus/RTU (client support)

All application objects receive startup parameters.

Startup parameters are stored in the non-volatile flash memory.

Parameter value ranges and presets

The values displayed in bold are pre-settings.

17.1 Protocol (0500_{hex}: UART_Protocol)

Serial data traffic can be implemented in accordance with various protocols. The protocol used depends on the protocol supported by the remote partner.

Protocol	
Value (hex)	Meaning
00	Transparent
01	End-to-end
02	Reserved
03	Modbus/RTU (client support)
04	Modbus/RTU (server support)
Other values are not permitted.	

17.2 Baud rate (0502_{hex}: UART_Baudrate)

Use this object to parameterize the transmission speed (baud rate) in bps.

The transmission speeds of 100 bps up to 600 bps can be selected in increments.

Between 1200 bps and 230400 bps, each transmission speed can be set continuously.

Baud rate (bps)
100
110
150
300
600
1200 ... 230400
9600
Other values are not permitted.

17.3 Number of data bits (0503_{hex}: UART_Databits)

Use this object to specify how many bits are transmitted per character.

Number of data bits	
Value (hex)	Meaning
07	7
08	8
Other values are not permitted.	



Modbus/RTU (client or server support):

Parameterization: 8 data bits

Faultless communication is not possible with 7 data bits.

17.4 Parity (0504_{hex}: UART_Parity)

Use this object to define the parity.

Parity	
Value (hex)	Meaning
00	None
01	Odd
02	Even
Other values are not permitted.	



Modbus/RTU (client or server support):

Recommendation: Even parity

All settings are possible.

17.5 Number of stop bits (0505_{hex}: UART_Stopbits)

Use this object to parameterize how many stop bits mark the end of a character.

Number of stop bits	
Value (hex)	Meaning
01	1
02	2
Other values are not permitted.	



Modbus/RTU(client or server support):

Recommendation: 1 stop bit

Exception: If you have parameterized “no parity”, set the parameter for the number of stop bits to 2 to comply with the 11-bit format for each character.

17.6 Behavior in the event of incorrect character reception (0509_{hex}: UART_RxErrorProperties)

Relevant for protocol: Transparent
End-to-end

Use this object to parameterize which character (error pattern) is written to the FIFO if a character was received with errors (e.g., in the event of a parity error).

Behavior upon faulty character reception	
Value (hex)	Meaning
00	If a character is received with an error, no error pattern is stored.
01 ... FE	The value is used as an error pattern.
FF	The invalid character is stored instead of the error pattern.
24	\$
Other values are not permitted.	

17.7 End delimiter (050B_{hex}: UART_Delimiter)

Relevant for protocol: End-to-end

The 1st and 2nd delimiters contain the end delimiters for the end-to-end protocol.

End delimiter				
Byte	Delimiter	Permitted values (hex)	Default value	
			hex	ASCII character
High	1.	00 ... FF	0D	CR = Carriage return
Low	2.	00 ... FF	0A	LF = line feed

17.8 Filter for Modbus server address (050C_{hex}: UART_ModbusFilterServerAddress)

Relevant for protocol: Modbus/RTU
(server support)

You can use this object to specify a Modbus address. The Smart Element indicates the receipt of a Modbus unicast telegram containing this address in the status word. The data from this unicast telegram is stored in the receive buffer.

The receipt of broadcast telegrams (address 0) is always indicated.

Filter	
Value	Meaning
1 ... 247	Only the receipt of valid Modbus telegrams with this server address is indicated in the status word.
255	The receipt of valid Modbus telegrams with a server address 1 to 247 is indicated in the status word.
Other values are not permitted.	

17.9 Modbus response timeout (050D_{hex}: UART_ModbusResponseTimeout)

Relevant for protocol: Modbus/RTU
(client support)

Use this object to parameterize the maximum waiting time for the response of the addressed Modbus server after transmitting a Modbus telegram.

Modbus response timeout	
Value range	100 ms ... 65535 ms
Default	2500 ms
Other values are not permitted.	

18 Device descriptions

The device is described in the device description files.

The device descriptions for controllers from Phoenix Contact are included in PC Worx and PLCnext Engineer, as well as in the corresponding service packs.

The device description files for other systems are available for download at www.phoenixcontact.com/products in the download area of the bus coupler installed.