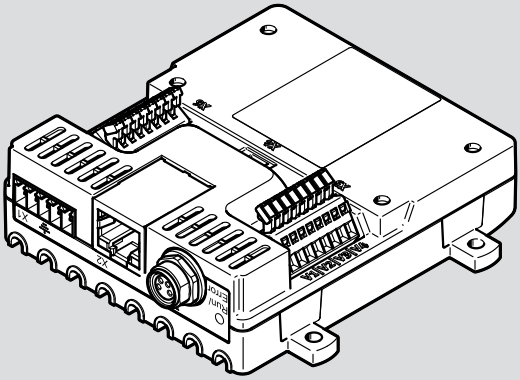
VAEM-V-S8EPRS2

Valve control module



FESTO

Operating instructions

8144870
2021-10a
[8144872]

Translation of the original instructions

Table of contents

1 Applicable documents

All available documents for the product ➔ www.festo.com/sp.

Documents	Product	Contents
Brief instruction	valve control module VAEM-V-S8EPRS2	Installation, Parameterisation
Assembly instructions	H-rail mounting VAME-V3-H	–
Application Note	valve control module VAEM-V-S8EPRS2	Connection via Ethernet in CODESYS
Application Note	valve control module VAEM-V-S8EPRS2	Combination of VTOE-8... and VAEM-V-S8...
Software	valve control module VAEM-V-S8EPRS2	GUI for the valve control module
Software	valve control module VAEM-V-S8EPRS2	API for the valve control module

Tab. 1: Applicable documents

2 Safety

2.1 Safety instructions

- Only use the product in original status without unauthorised modifications.
- Only use the product if it is in perfect technical condition.
- Take into consideration the ambient conditions at the location of use.
- Prior to mounting, installation and maintenance work: Switch off power supply and secure it from being switched back on.

2.2 Intended use

The product is designed for control of solenoid valves by holding current reduction and it is intended for industrial operation in enclosed spaces.

2.3 Training of skilled personnel

Installation, commissioning, maintenance and disassembly should only be conducted by qualified personnel.
The specialized personnel must be familiar with the installation and operation of electrical and pneumatic control systems.

3 Additional information

- Accessories ➔ www.festo.com/catalogue.

4 Service

Contact your regional Festo contact person if you have technical questions ➔ www.festo.com.

5 Product overview

The product is an electronic control with integrated, adjustable holding current reduction for controlling solenoid valves. Communication takes place via a COM interface using the ASCII protocol or via an Ethernet interface using the Modbus TCP protocol in accordance with the client-server principle.

5.1 Functions

5.1.1 Valve parameter settings

The product offers the following functions, which can be set via parameters.

- set nominal voltage
- read nominal voltage
- select valve
- read valve selection
- set switching time
- read switching time
- set delay time
- read delay time
- set pickup time
- read pickup time
- set inrush current
- read inrush current
- set holding current
- read holding current
- set current reduction
- read current reduction

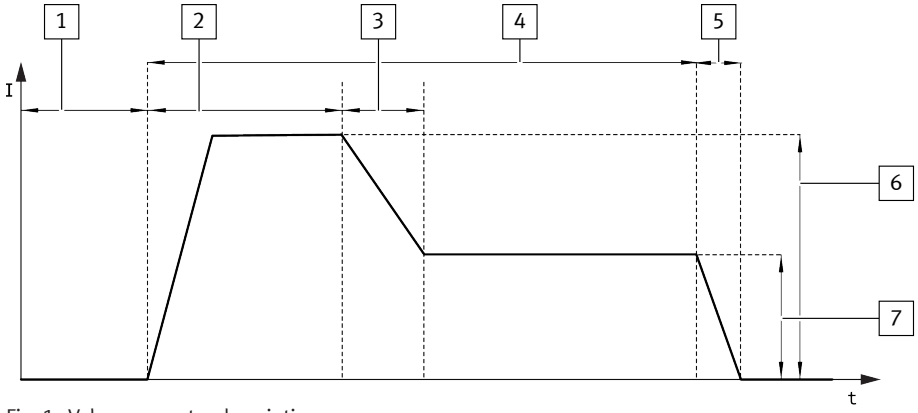


Fig. 1: Valve parameter description

Legend	Valve parameters	Description
1	T_Delay	Delay time
2	T_Inrush	Pickup time

Legend	Valve parameters	Description
3	T_Tau	Current reduction time
4	T_On	Switching time
5	T_Switch_Off	Cooldown Parameters cannot be set individually T_Switch_Off is a constant 4 ms
6	I_Inrush	Inrush current
7	I_Hold	Holding current

Tab. 2

5.1.2 Holding current reduction

Function description

The integrated holding current reduction reduces the current consumption to the set holding current after the adjustable pickup time has elapsed.

With holding current reduction, the output current of the valve is measured and the signal at the VAEM output is adjusted and controlled in accordance with the set parameters.

- Reduction of the heat development of the solenoid valve coil
- Increased service life of solenoid valves
- Lower power consumption
- Improvement of the switching times of solenoid valves

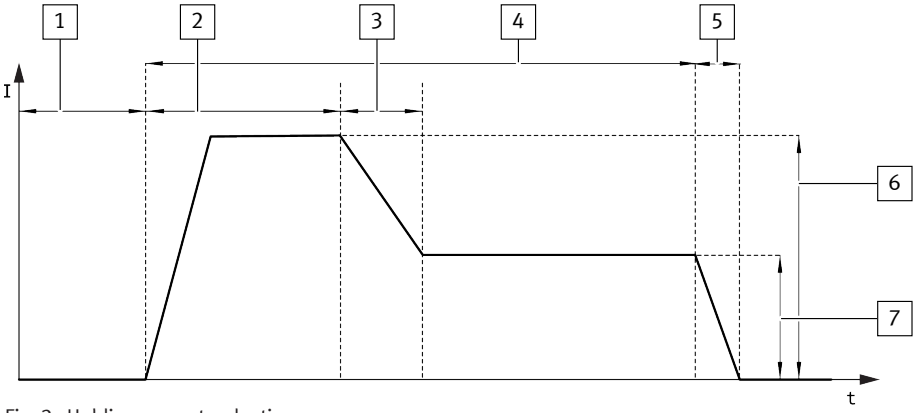


Fig. 2: Holding current reduction

1 Delay time
2 Pickup time
3 Current reduction time
4 Switching time
5 Cooldown
6 Inrush current
7 Holding current

5.2 Product design

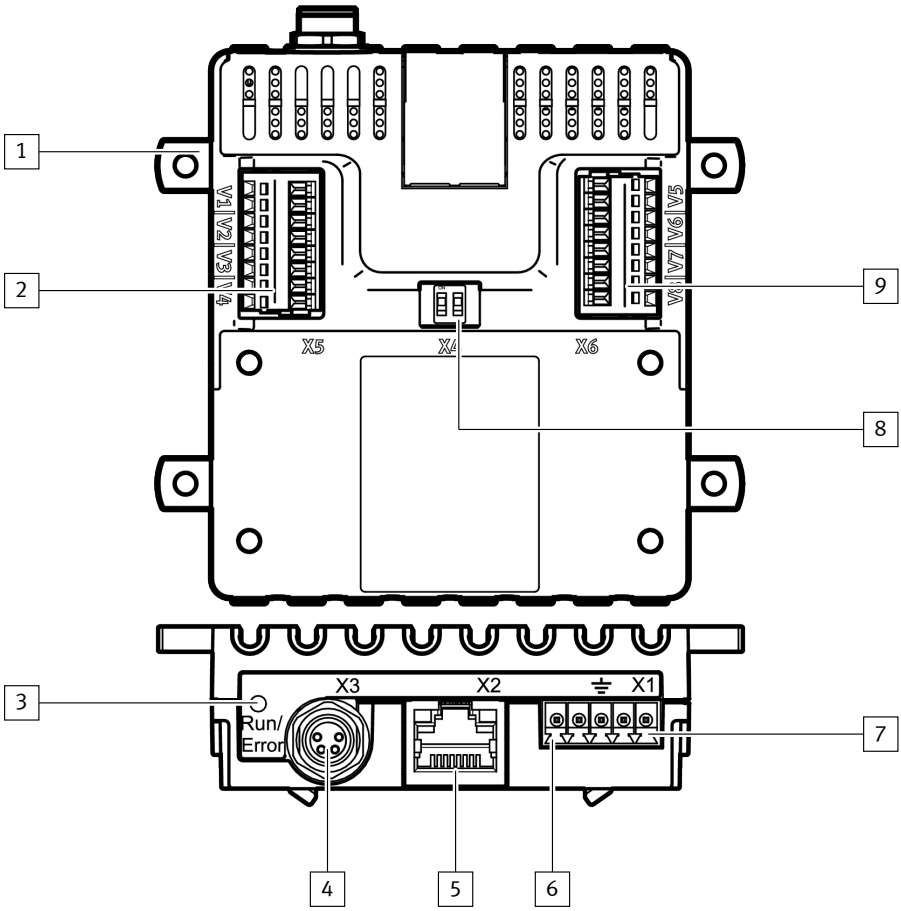


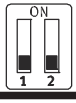
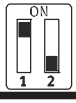
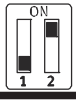
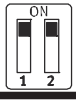
Fig. 3: Product design and connections

1 Housing with mounting flange
2 Terminal [X5]: valves 1 ... 4
3 Status indication: LED
4 Port [X3]: RS232
5 Port [X2]: Ethernet
6 Connection [X1], pin 5: trigger input
7 Connection [X1], pin 1: power supply
8 Switch [X4]: baud rate
9 Terminal [X6]: valves 5 ... 8

5.2.1 Control elements

Switch [X4]

The transmission speed of the communication interface [X3] is set at switch [X4].

baudrate [kBd]	9.6	19.2	57.6	115.2
Switching position				

Tab. 3: Setting the baud rate

5.2.2 Connecting elements

⚠ WARNING

Risk of injury due to electric shock.

- For the electrical power supply with extra-low voltages, use only PELV circuits that guarantee a reinforced isolation from the mains network.
- Observe IEC 60204-1/EN 60204-1.

NOTICE

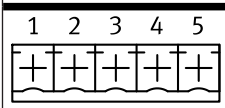
Material damage due to incorrect or incomplete installation.

The following conditions must be fulfilled for commissioning:

- The system must be fully assembled.
- The electrical installation must be complete and checked.

Connection [X1]

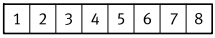
The connection [X1] shows the power supply and the trigger input.

Plug pattern	Pin	Function
	1	Power supply: 24 V DC
	2	Power supply: GND
	3	Functional earth ⏚
	4	Trigger input: GND
	5	Trigger input: 24 V DC

Tab. 4: Pin allocation connection [X1]

Ports [X5] and [X6]

The connections [X5] and [X6] are each provided with a terminal to connect the valves.

Plug pattern	Pin	Function
terminal [X5]		
	1	Connection valve 1
	2	
	3	Connection valve 2
	4	
	5	Connection valve 3

Plug pattern	Pin	Function
1 2 3 4 5 6 7 8	6	Connection valve 3
	7	Connection valve 4
	8	
terminal [X6]		
1 2 3 4 5 6 7 8	1	Connection valve 8
	2	
	3	Connection valve 7
	4	
	5	Connection valve 6
	6	
	7	Connection valve 5
	8	

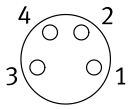
Tab. 5: Pin assignment of terminals [X5] and [X6]

Port [X2]: Ethernet

The port [X2] in the version VAEM-V-S8EPRS2 is a TCP/IP communication interface via Ethernet.

Port [X3]: RS232

The port [X3] is for communications via the RS232 interface.

Plug pattern	Pin	Function
	1	GND - Common Ground
	2	TxD - Transmit Data
	3	RxD - Receive Data
	4	NC - Not connected

Tab. 6: Port pin allocation [X3]

6 Commissioning

Commissioning and parameterisation via Festo software

Easier commissioning and parameterisation via Festo software ➔ www.festo.com/sp.

1. Connect the valves to the ports [X5] and [X6].
2. Connect communications cable with the COM interface and interface [X3].
   Alternatively, connect the Ethernet communication cable and interface [X2].
3. Assemble plug in accordance with the plug pattern ➔ Tab. 4 Pin allocation connection [X1] and plug into port [X1].

4. With RS232 communication:
   Set required baud rate at the switch [X4].
5. Apply nominal voltage to the product.
   ↳ The LED is green.
6. Establish and parameterise communications connection.

6.1 Connection settings

NOTICE

Unauthorised Access to the Device Can Cause Damage or Malfunction.

- When connecting the device to a network, protect the network from unauthorised access.
  Standards for security in information technology can be used for network protection measures, e.g. IEC 62443, ISO/IEC 27001.

The system and configuration parameters of the VAEM-V-S8EPRS2 can be set by
– ASCII telegrams via RS232
– Modbus TCP/TCP protocol via Ethernet
In order to communicate with the VAEM-V-S8EPRS2 via RS232, serial terminal software is required to transmit the ASCII telegrams.

Connection settings via an RS232 connection

Connection settings are required to use the product with the serial interface module [X3] and terminal software.

Setting	Value
Port	See connected COM-Port to Client
baudrate	See setting on the product ➔ Tab. 3 Setting the baud rate
Data size	8
Parity	None
Stop bit	1
Handshaking	None
Mode	Free

Tab. 7: Connection settings for RS232 communication

Ethernet connection settings

Setting	Value
IP	192.168.178.1 (default)
Port	502 (Modbus TCP default)

Tab. 8: Ethernet connection settings

i

When using multiple network interfaces, the Ethernet (TCP/IP) interface must be prioritised.

7 Parameterisation

7.1 Parameter structure

Access types and data types

Access type	ASCII abbreviation	HEX value	Description
Read access	R	00	The read access returns the set values in decimal.
Write access	W	01	The write access overwrites the previously set values.

Tab. 9: Access types

Data type	ASCII abbreviation	HEX value	Value range	Description
uint64	U64	04	0 ... 2^{64}-1	64 bit unsigned integer
uint32	U32	03	0 ... 2^{32}-1	32 bit unsigned integer
uint16	U16	02	0 ... 2^{16}-1	16 bit unsigned integer
uint08	U08	01	0 ... 2^{8}-1	8 bit unsigned integer

Tab. 10: Data types

Ethernet system parameters

Function	Index	ASCII abbreviation	HEX value
controllword	Index 1	I1	01
statusword	Index 2	I2	02
Saving parameter values	Index 11	I11	0B
nominal voltage	Index 4	I4	04
inrush current	Index 5	I5	05
holding current	Index 6	I6	06
switching time	Index 7	I7	07
pickup time	Index 8	I8	08
operating mode	Index 9	I9	09
select valve	Index 19	I19	13

Function	Index	ASCII abbreviation	HEX value
delay time	Index 22	I22	16
current reduction time	Index 46	I46	01

Tab. 11: Parameter listing index with abbreviations

Addressing	Subindex	ASCII abbreviation	HEX value
System	Subindex 0	S0	00
Valve 1	Subindex 0	S0	00
Valve 2	Subindex 1	S1	01
Valve 3	Subindex 2	S2	02
...	...	...	...
Valve 8	Subindex 7	S7	07

Tab. 12: Listing of subindex and addressing

Function	Index I	Hex value index	Data type	Value range	Default value	Description
IPV4Address-Soll	60 (with sub-index 0)	3C	Uint32	0 ... 4294967294 (decimal)	3232281089 (decimal) ➔ C0.A8.B2.01 ➔ 192.168.178.1	Change destination IP address in the network In the event of a change, a restart is required. Invalid IP addresses: – 0.x.x.x – 127.x.x.x – 224.x.x.x – 255.255.255.255
IPV4Address-sIst	61	3D	Uint32	0 ... 4294967294 (decimal)	3232281089 (decimal)	Display destination IP address in the network
ModbusTcp-PortSoll	54	36	Uint16	0 ... 50265535	502 (decimal)	Change Modbus TCP port. In the event of a change, a restart is required.
ModbusTcp-PortIst	55	37	Uint16	0 ... 50265535	502 (decimal)	Display Modbus TCP port.

Function	Index I	Hex value index	Data type	Value range	Default value	Description
ModbusTcpTimeoutSoll	56	38	Uint16	0 ... 50265535	0 (Timeout inactive)	Change TCP/IP connection timeout. Minimum timeout is 1000 ms. Values between 0 ... 1000 are increased to 1000.
ModbusTcpTimeoutIst	57	39	Uint16	0 ... 50265535	0	Display TCP/IP connection timeout.

Tab. 13: Valve control functions

7.2 Architecture ASCII telegram

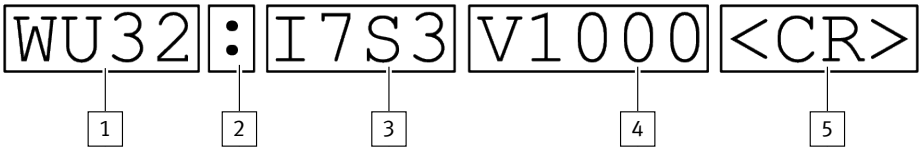


Fig. 4: Structure of the ASCII telegram

1 Command area with access type and data type
2 Separator
3 Identification area with index and sub-index
4 Transfer value
5 Command end character

Command area
The access type and the data type are specified in the command area.

Identification area
The identification area consists of the index and the subindex. The index denotes the system parameter. The subindex denotes the addressing at which the function of the system parameter is to be executed.

Transfer value
When writing an ASCII telegram the transfer value must always be specified with V in front. When reading, the transfer value is omitted and is returned as an answer.

Command end character
The command end character is not specified in this documentation because the need for a command end character depends on the terminal software used.

7.3 Architecture of Modbus TCP

Parameter structure for sending via Modbus TCP

Modbus TCP is an open communication protocol with Client-Server architecture. The byte sequence is based on the big endian system.

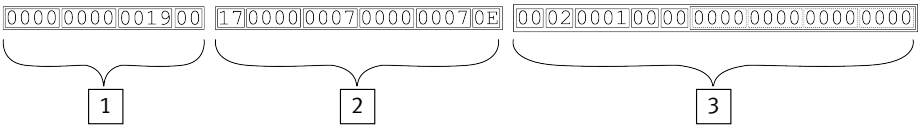


Fig.5

1 Modbus TCP header

2 Modbus TCP addressing function code 0x17

3 Structure of Modbus TCP function and data area function code 0x17

Modbus TCP header

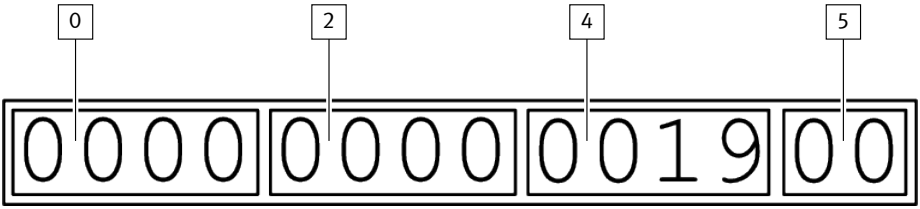


Fig. 6: Modbus TCP header

0 Transaction number

2 Protocol identifier

4 Number of following bytes

5 Device identification

Byte no.	Number of bytes	Designation	Description	Byte order
0	2	Transaction number	Freely selectable	most significant byte
1				least significant byte
2	2	Protocol identifier	0x0000 (default)	most significant byte
3				least significant byte
4	2	Number of bytes to follow	Number of bytes of device identifier, function code, addressing and data area	most significant byte

Byte no.	Number of bytes	Designation	Description	Byte order
5	2	Number of bytes to follow	Number of bytes of device identifier, function code, addressing and data area	least significant byte
6	1	Device identification	Freely selectable, unique address, Slave-ID 00 (default)	

Tab. 14: Structure of Modbus TCP Header

Structure of Modbus TCP addressing function code 0x17

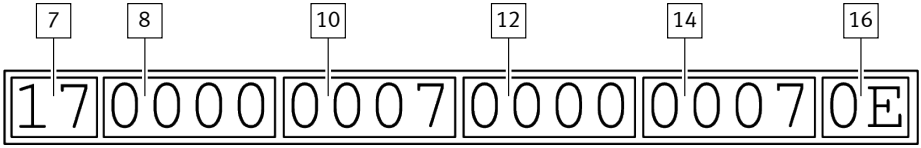


Fig. 7: Structure of Modbus TCP addressing

7 Function code
8 Start address of the read registers
10 Number of registers read
12 Start address of the write registers
14 Number of registers written
16 Number of bytes written

Byte no.		Number of bytes	Designation	Description	Byte order
transmit	receive				
7	7	1	Function code	0x17 (VAEM standard) Reading and writing of data	
8	-	2	Start address of the read registers	0x0000 (default value)	most significant byte
9	-				least significant byte
10	-	2	Number of registers read	0x0007 (default value)	most significant byte
11	-				least significant byte
12	-	2	Start address of the write registers	0x0000 (default value)	most significant byte
13	-				least significant byte
14	-	2	Number of registers written	0x0007 (default value)	most significant byte

Byte no.		Number of bytes	Designation	Description	Byte order
transmit	receive				
15	-	2	Number of registers written	0x0007 (default value)	least significant byte
16	8	1	Number of bytes written	0x0E (default value)	

Tab. 15: Structure of Modbus TCP addressing function code 0x17

Structure of Modbus TCP function and data area function code 0x17

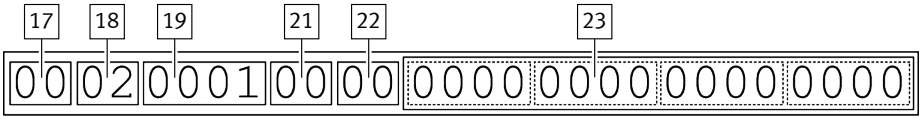


Fig. 8: Function and data area function code 0x17

17 Read/write function
18 Data type
19 System parameters index
21 Addressing parameter subindex
22 Error (reserved)
23 Data area

Send Byte no.	Receive Byte no.	Number of bytes	Designation	Description	Byte order
17	9	1	Read or write access	0x00: read access 0x01: write access	
18	10	1	Data type	Data type dependent on system parameter index ➔ Tab. 10 Data types	
19	11	2	Index	Function to be executed ➔ Tab. 11 Parameter listing index with abbreviations ➔ Tab. 31 Valve control functions	Highest-value byte
20	12				Lowest-value byte
21	13	1	Subindex	Addressing of the function at system or valve level➔ Tab. 12 Listing of subindex and addressing	

Send Byte no.	Receive Byte no.	Number of bytes	Designation	Description	Byte order
22	14	1	Error	Reserved for error return	
23	15	8	Data area Write: transfer value Read: place holder 0 Return: return value	Transfer value dependent on the data type uint64: Byte 23 ... 30 uint32: Byte 27 ... 30 uint16: Byte 29 ... 30 uint8: Byte 30 Unused bytes have the default value 0x00	Highest-value byte
24	16				
25	17				
26	18				
27	19				
28	20				
29	21				
30	22				Lowest-value byte

Tab. 16: Structure of Modbus TCP function and data area function code 0x17

7.4 Parameter description

7.4.1 System parameters

controllword

The control word is used for:

- Stopping the valves before the operating time has elapsed
- Resetting the active error bit
- Manual start (only in operating mode 1)

The control word is a 16-digit binary code. To transfer the binary code of the check word, it must be converted into a decimal (RS232 communication) or hexadecimal (Ethernet communication) value. The individual digits of the binary code represent the individual operating functions of the product from right to left. Bits 4 ... 15 are reserved and without function.

Bit	Designation	Status 0	Change of status 0 → 1
0	Start	Normal status	Starts all selected valves (only possible in operating mode 1)
1	-	Reserved, without function	
2	Stop	Normal status	Stops the operation of all selected valves until the next change of status 1 → 0 (only possible in operating mode 1)

Bit	Designation	Status 0	Change of status 0 → 1
3	Reset error message	Normal status	Stops the operation of all selected valves and resets the active error bit in the status word
4 ... 15	-	Reserved, without function	

Tab. 17: Meanings of the bits in the control word

Access	Data type	Index I	Subindex S	Value V	Example of an ASCII telegram
W (Write)	U 16	1	0	x	`WU16:I1S0Vx`
R (Read)	U16	1	0	-	`RU16:I1S0`

Tab. 18: Architecture ASCII telegram control word

Access	Data type	System parameters index	Addressing parameter subindex	Transfer value	Example of data area Modbus TCP
0x01 (write)	0x02 (Uint16)	0x01	0x00	xx	`01 02 0001 00 00 xxxx xxxx xxxx xxxx`
0x00 (read)	0x02 (Uint16)	0x01	0x00	0x00	`00 02 0001 00 00 0000 0000 0000 0000`

Tab. 19: Architecture of Modbus TCP: control word

Example of use of the check word

Situation: starting all selected valves

1. Reset control word.
   - Via ASCII telegram: `WU16: I1S0V0`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 02 0001 00 00 0000 0000 0000 0000`
2. Set bit 1 of the control word.
   - Via ASCII telegram: `WU16: I1S0V1`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 02 0001 00 00 0000 0000 0000 0001`

   ↳ Start all selected valves.

statusword

The status word is used for:

- Reading the operating mode and readiness for operation
- Error status and error detection

The status word is a 16-digit binary code. To interpret the status word, the returned decimal (RS232 communication) or hexadecimal (Ethernet communication) value must be converted into a binary code.
The individual digits of the binary code represent the individual status indicators from right to left. The individual bits are shown in the table from top to bottom.

Bit	Designation	Meaning	
		Status 0	Status 1
0	Status	No valve active	≥ 1 valve active
1	-	Reserved, without function	
2	-	Reserved, without function	
3	Error bit	no error	Error
4	Operational readiness	– ≥ 1 valve in operation – No valve selected – Error – Stop active	≥ 1 valve selected and ready to start
5	-	Reserved, without function	
6 7	operating mode	00: operating mode 1 01: operating mode 2 10: operating mode 3 11: reserved	
8	Valve 1	– Valve not selected – Error on selected valve – No valve connected	Valve selected
...	...	...	...
15	Valve 8	– Valve not selected – Error on selected valve – No valve connected	Valve selected

Tab. 20: Meanings of the bits in the status word

Access	Data type	Index I	Subindex S	Value V	Example of an ASCII telegram
R	U16	2	0	-	`RU16:I2S0`

Tab. 21: Architecture of ASCII telegram: query status word

Access	Data type	System parameters index	Addressing parameter sub-index	Transfer value	Example of data area Modbus TCP
0x00 (read)	0x02 (Uint16)	0x02	0x00	0x00	00 02 0002 00 00 0000 0000 0000 0000

Tab. 22: Architecture of Modbus TCP: query status word

Examples of meanings of the statuses of the status word

The following combinatorics are based on the fact that irrelevant bits are omitted or occur without a value in the table.

Combinatorics of the binary code						Meaning
Bit $_{valve}$	Bit 7	Bit 6	Bit 4	Bit 3	Bit 0	
	0	0				Product is in operating mode 1.
	0	1				Product is in operating mode 2.
	1	0				Product is in operating mode 3.
1						The valves that have state 1 in the $_{valve}$ bit are selected valves.
			1	0	0	Product is ready to go.
0			0	1		Error at all selected valves which have the state 0 at the corresponding bit $_{valve}$.

Tab. 23: Combinatorics of the states of the status word

set operating mode

The product can be operated in three different modes. They differ in cycle time, cycle start and trigger.

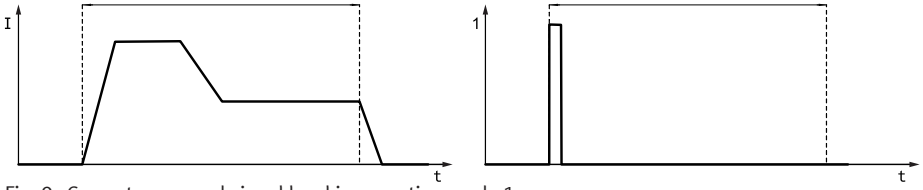


Fig. 9: Current curve and signal level in operating mode 1

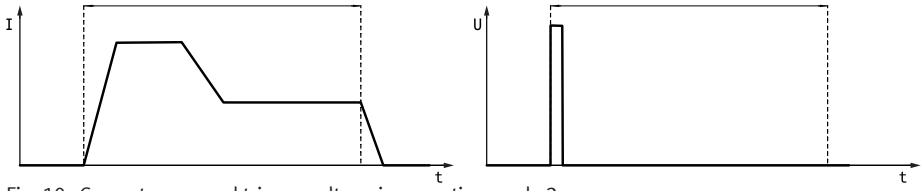


Fig. 10: Current curve and trigger voltage in operating mode 2

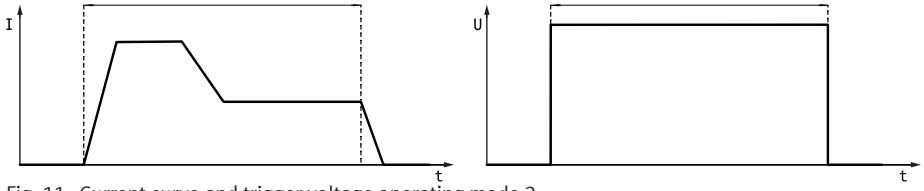


Fig. 11: Current curve and trigger voltage operating mode 3

	operating mode 1	**operating mode 2**	**operating mode 3**
Cycle start	Internal via control word via communication interface	External by 24 V trigger input	External by 24 V trigger input
Triggers	Change of the logical state from 0 → 1 at bit 0 of the control word	Change to active state at trigger input	Change to active state at trigger input
Cycle time	Set switching time of the individual valves	Set switching time of the individual valves	Duration of the active state at the trigger input
Variable value	0 0x00	1 0x01	2 0x02

Tab. 24: Differences according to operating mode

Access	**Data type**	**Index I**	**Subindex S**	**Value V**	**Example of an ASCII telegram**
W	U08	9	0	0 1 2	`WU08:I9S0V2`

Tab. 25: Architecture of the ASCII telegram: set operating mode

Access	Data type	System parameters index	Addressing parameter sub-index	Transfer value	Example of data area Modbus TCP
0x01 (write)	0x01 (Uint08)	0x09	0x00	0x00 0x01 0x02	01 01 0009 00 00 0000 0000 0000 0001

Tab. 26: Architecture of Modbus TCP: set operating mode

Example setting operating mode 2

1. Check the set operating mode using
   - ASCII telegram: `RU08: I9SO`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 00 01 0009 00 00 0000 0000 0000 0000`
2. Send required operating mode, using
   - ASCII telegram: `WU08: I9S0V1`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 01 0009 00 00 0000 0000 0000 0001`

select valve

With the system parameter [Select valve] it is possible to individually select valves for the control process. Here it is possible to store all valves in the valve selection or to limit this to individual valves or valve combinations (e.g. V1, V4, V5 and V7). It is also possible to query which valves are selected. The valve selection is transferred as a decimal value originating in a binary code. Each bit of the binary code represents a valve from right to left.

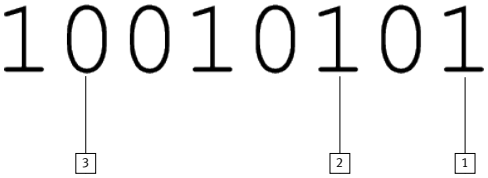


Fig. 12: Valve assignment in binary code

1 Valve 1
2 Valve 3
3 Valve 7

Access	Data type	Index I	Subindex S	Value V	Example of an ASCII telegram
W	U08	19	0	89 (selected valves 1, 4, 5 and 7)	`WU08:I19S0V89`

Tab. 27: Architecture of the ASCII telegram: select valve

Access	Data type	System parameters index	Addressing parameter subindex	Transfer value	Example of data area Modbus TCP
0x01 (write)	0x01 (Uint08)	0x13	0x00	0x59 (selected valves 1, 4, 5 and 7)	`01 01 0013 00 00 0000 0000 0000 0059`

Tab. 28: Architecture of the Modbus TCP: select valve

Example of selecting valves 1, 4 and 6

Situation: previous valve selection is unknown and should be overwritten.

The binary code 101001 results from the selection of valves 1, 4 and 6. The binary code corresponds to the decimal value 41.

1. Interrogation of the selected valves using
   - ASCII telegram: `RU08: I19S0`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 00 01 0013 00 00 0000 0000 0000 0000`

     ↳ Decimal value is returned.
2. Convert decimal value to binary code and interpret ➔ Fig. 12.
3. Select the desired valves by setting the value 1 to the corresponding position in the binary code. All non-selected valves have the value 0 at the corresponding positions.
4. Convert binary code into decimal or hexadecimal value.
5. Transfer function using
   - ASCII telegram: `WU08: I19S0V41`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 01 0013 00 00 0000 0000 0000 0029`

save setting values

Via the system parameter [Save setting values], valve-specific configuration parameters are saved on the product and are immediately available again after the de-energised state. Thus, with renewed energisation, the parameterisation does not have to be carried out again.

The following configuration parameters are saved:
- nominal voltage
- inrush current
- holding current
- switching time
- pickup time
- operating mode
- selected valves
- delay time
- current reduction time

Access	Data type	Index I	Subindex S	Value V	Example of an ASCII telegram
W	U32	11	0	99999	`WU32:I11S0V99999`

Tab. 29: Architecture of the ASCII telegram: save setting values

Access	Data type	System parameters index	Addressing parameter subindex	Transfer value	Example of data area Modbus TCP
0x01 (write)	0x03 (Uint32)	0x0B	0x00	0x0001 869F	`01 03 000B 00 00 0000 0000 0001 869F`

Tab. 30: Architecture of Modbus TCP: save setting values

7.4.2 Configuration parameters per valve

Variables

The variables x and y must be replaced.
y = subindex for valve 0 ...7
x = value to be accepted as decimal number.

Function	Index I	Hex value index	Data type	Unit	Value range	Default value	Example of an ASCII telegram
set nominal voltage	4	04	uint16	mV	8000 ... 24000	24000	`WU16:I4SyVx`
read nominal voltage	4	04	uint16	mV	-	-	`RU16:I4Sy`
set switching time	7	07	uint32	0.2 ms	1 ... 2^{32}-1	500	`WU32:I7SyVx`

Function	Index I	Hex value index	Data type	Unit	Value range	Default value	Example of an ASCII telegram
read switching time	7	07	uint32	0.2 ms	-	-	RU32:I7Sy
set delay time	22	16	uint32	0.2 ms	0 ... 2^{32}-1	0	WU32:I22SyVx
read delay time	22	16	uint32	0.2 ms	-	-	RU32:I22Sy
set pickup time	8	08	uint16	0.2 ms	1 ... 500	125	WU16:I8SyVx
read pickup time	8	08	uint16	0.2 ms	-	-	RU16:I8Sy
set inrush current	5	05	uint16	mA	20 ... 1000	300	WU16:I5SyVx
read inrush current	5	05	uint16	mA	-	-	RU16:I5Sy
set holding current	6	06	uint16	mA	20 ... 400	100	WU16:I6SyVx
read holding current	6	06	uint16	mA	-	-	RU16:I6Sy
set current reduction	46	2E	uint32	0.2 ms	0 ... 1000	100	WU32:I46SyVx
read current reduction	46	2E	uint32	0.2 ms	-	-	RU32:I46Sy

Tab. 31: Valve control functions

8 Cleaning

1. Switch off the following energy sources to clean the outside:
   - Compressed air
   - Operating voltage
2. Clean the outside of the product with a soft cloth. Do not use aggressive cleaning agents.

9 Malfunctions

9.1 Diagnostics

LED green	LED red	Description
		Error-free operation
		Device identification failed

LED green	LED red	Description
○		Error is present
		Operating system error
		Bootloader mode This mode shows that a firmware update has failed.
		Firmware update is being run

Tab. 32: Status indicator

Diagnostics via status word

The status word facilitates the diagnosis of faults and their origin. Structure of the status word ➔ Tab. 20 Meanings of the bits in the status word.

Bit	Bit combination 1	Bit combination 2
Bit 0	1	0
Bit 3	1	1
Bit 4	0	0
Meaning	Error in the operating phase and ≥1 connected valve error-free	No operating phase and faults on one or more valves

Tab. 33: Frequent bit combinations in the status word

9.2 Fault clearance

9.2.1 Resetting the active error bit

If an error occurs, the error bit, bit 3 in the status word, is activated. This must be reset manually. To reset this, a state change of 0 → 1 must take place in the control word at bit 3. To ensure continued operation, the control word must be reset to its original state.

1. Reset control word.
   - Via ASCII telegram: `WU16: I1S0V0`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 02 0001 00 00 0000 0000 0000 0000`
2. Set bit 3 in the control word.
   - Via ASCII telegram: `WU16: I1S0V8`
   - Using Modbus TCP: `0000 0000 0019 00 17 0000 0007 0000 0007 0E 01 02 0001 00 00 0000 0000 0000 0008`

9.2.2 System faults

In the event of a system fault, all selected valves are deactivated in the status word. The red LED flashes and error bit 3 in the status word becomes active.

System faults	Inspection interval	Remedy
Voltage fluctuations	– Constant review	– Check fixed power supply – Ensure constant nominal voltage At least 80% of the nominal voltage – ➔ 9.2.1 Resetting the active error bit
Total current consumption	– At start-up – During the operating time of the valves	– Check the total current consumption of the product. Total current consumption < 1.8 A – Check current consumption of valves and adjust configuration parameters if necessary – ➔ 9.2.1 Resetting the active error bit

Tab. 34: Fault clearance of system faults

9.2.3 Valve faults

In the event of a valve fault, only the faulty valve is deactivated in the status word. The red LED flashes and error bit 3 in the status word becomes active. The valves without faults are not affected by this. Valve faults can be identified by the fact that selected faulty valves are indicated as deactivated in the status word and the red LED flashes.

Valve faults	Inspection interval	Remedy
– Idling of the valve – Cable break at valve connection	– At start-up	– Check valve and wiring and replace if necessary – Error in case of current consumption after start command < 20 mA (> 100 ms) – ➔ 9.2.1 Resetting the active error bit
– Current consumption of a valve exceeds permissible value – Valve short-circuit	– At start-up – During the operating time of the valves	– Check the current consumption of the valve – Error in case of current consumption during tightening phase > 1.35 A – Error in current consumption during pickup phase > 1 A (> 4 ms) – Error in current consumption during the holding current phase > 400 mA (> 10 ms) – Check valve and wiring and replace if necessary – ➔ 9.2.1 Resetting the active error bit

Tab. 35: Fault clearance of valve faults

9.2.4 Communication errors

The error codes for communication errors are returned by the product as return values after incorrect read and write commands.

Error code	Description	Fault clearance
E0	Ready for operation, no error	
E34	Invalid index	Check transfer command and use valid index
E35	Invalid subindex	Check transfer command and use valid subindex
E36	Read request cannot be processed	Check and correct input
E37	Write request cannot be processed	Check and correct input

Error code	Description	Fault clearance
E41	Specified value falls below the minimum value	Check and correct value
E42	The specified value exceeds the maximum value	Check and correct value
E43	Incorrect transfer value	Check and correct the transfer value
E44	Data type incorrect	Use correct data type
E93	General syntax error	Check and correct the ASCII structure of the input
E94	Syntax error index (variable x)	Check and correct index and value
E95	Syntax error subindex (variable y)	Check and correct subindex and value
E96	Syntax error value	Check and correct value
E97	Command execution aborted	Repeat command or restart

Tab. 36: Error codes communication

10 Disposal

ENVIRONMENT

Send the packaging and product for environmentally sound recycling in accordance with the current regulations ➔ www.festo.com/sp.

11 Technical data

VAEM-V-S8EPRS2		
General		
approval		RCM Mark
ambient temperature	[°C]	0 ... 50
storage temperature	[°C]	–20 ... +70
degree of protection		IP20
Pollution degree		2
product weight	[g]	98
dimensions W x L x H	[mm]	92 x 100 x 28
Relative air humidity	[%]	0 ... 95 Non-condensing
Nominal altitude of use	m ASL	≤ 2000
vibration resistance		Shock test with severity level 2 in accordance with EN 60068-2-6 (FN 942017-4)

VAEM-V-S8EPRS2		
shock resistance		Shock test with severity level 2 in accordance with EN 60068-2-27 (FN 942017-5)
note on materials		Contains PWIS substances
Cable length		
Valve	[m]	≤ 5
RS232	[m]	≤ 10
Power supply	[m]	≤ 30
Ethernet	[m]	≤ 25
Electronics		
nominal operating voltage DC	[V]	24
internal current consumption, operating voltage	[mA]	~ 36
perm. voltage fluctuations	[%]	± 10
Number of outputs		≤ 8
nominal operating voltage DC of load	[V]	8 ... 24 (PWM)
inrush current, per Output	[mA]	20 ... 1000 (≤ 100 ms)
inrush current, total	[A]	≤ 4 (≤ 100 ms)
holding current, per Output	[mA]	20 ... 400
holding current, total	[A]	≤ 1,8
trigger level	[V]	Level 14 ... 24
Ethernet interface, transmission rate	[Mbit/s]	10/100
Time resolution	[ms]	0,2

Tab. 37: Technical data

Type of severity level (SL)					
Vibration load					
Frequency range [Hz]		Acceleration [m/s²]		Deflection [mm]	
SL1	SL2	SL1	SG2	SL1	SL2

Type of severity level (SL)					
2 … 8	2 … 8	–	–	±3.5	±3.5
8 … 27	8 … 27	10	10	–	–
27 … 58	27 … 60	–	–	±0.15	±0.35
58 … 160	60 … 160	20	50	–	–
160 … 200	160 … 200	10	10	–	–
Shock load					
Acceleration [m/s²]		Duration [ms]		Shocks per direction	
SL1	SL2	SL1	SL2	SL1	SL2
±150	±300	11	11	5	5
Continuous shock load					
Acceleration [m/s²]		Duration [ms]		Shocks per direction	
±150		6		1000	

Tab. 38: Type of severity level (SL)

Copyright:
Festo SE & Co. KG
73734 Esslingen
Ruiter Straße 82
Deutschland

Phone:
+49 711 347-0

Internet:
www.festo.com