

**JetControl 24x  
Version Update  
from V. 3.01 to V. 3.11**



Jetter AG reserves the right to make alterations to its products in the interest of technical progress. These alterations need not to be documented in every single case.

This manual and the information contained herein have been compiled with due diligence. Jetter AG shall not be liable for printing errors contained herein or for other consequential damage.

The brand names and product names used in this manual are trade marks or registered trade marks of the respective title owner.

---

## Table of Contents

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Introduction</b>                                     | <b>5</b> |
| 1.1      | Update Information                                      | 5        |
| 1.1.1    | Registers   | 5        |
| 1.1.2    | JetView-Soft  | 6        |
| <b>2</b> | <b>Expansions</b>                                       | <b>7</b> |
| 2.1      | System Configuration                                    | 7        |
| 2.1.1    | File 'hosts'  | 7        |
| 2.1.2    | Domain Name System                                      | 7        |
| 2.2      | E-Mail  | 8        |
| 2.3      | HTTP Server   | 8        |
| 2.4      | RemoteScan  | 8        |
| 2.4.1    | General information                                     | 8        |
| 2.4.2    | Special functions                                       | 8        |
| 2.4.3    | Registers   | 11       |
| 2.5      | Modbus/TCP  | 12       |
| 2.5.1    | Licensing   | 12       |
| 2.5.2    | Server  | 13       |
| 2.5.3    | Client  | 14       |
| 2.6      | Data Files  | 16       |
| 2.6.1    | General information                                     | 16       |
| 2.6.2    | Special functions                                       | 16       |
| 2.6.3    | Registers   | 19       |
| 2.6.4    | File format   | 20       |
| 2.7      | Floating Point Register                                 | 21       |
| 2.8      | System Bus  | 22       |
| 2.8.1    | Third-party modules                                     | 22       |
| 2.8.2    | Function and power-level terminals on the JX-SIO module | 26       |
| 2.8.3    | Special flags associated to the system bus              | 28       |
| 2.8.4    | Access to JX-SIO analog values                          | 28       |
| 2.8.5    | Timeout register  | 28       |
| 2.8.6    | JetMove 200   | 28       |
| 2.8.7    | Festo CPV-Direct  | 28       |

---

|          |   |           |
|----------|---|-----------|
| 2.8.8    | Sync telegram   | 29        |
| 2.9      | FTP Server  | 29        |
| <b>3</b> | <b>Eliminated Software Bugs</b>                           | <b>30</b> |
| 3.1      | System Bus  | 30        |
| 3.1.1    | Response of analog outputs on Smart I/O modules to errors | 30        |
| 3.1.2    | Registers on Smart I/O modules                            | 30        |
| 3.1.3    | Analog values   | 30        |
| 3.1.4    | Analog terminals on Smart I/O modules                     | 30        |
| 3.2      | File System   | 30        |
| 3.2.1    | File names  | 30        |
| 3.2.2    | Concurrent access to several files                        | 30        |
| 3.3      | Arithmetic Operations                                     | 31        |
| 3.4      | UserInput   | 31        |
| 3.5      | HTTP Server   | 31        |

# 1 Introduction

| Version Updates - Survey |                      |          |           |
|--------------------------|----------------------|----------|-----------|
| Version                  | Function             | upgraded | corrected |
| V 3.00                   | Files system         | ✓        |           |
|                          | Interpreter          | ✓        |           |
|                          | Register interface   | ✓        |           |
|                          | pcom5/JetIP-Server   | ✓        |           |
|                          | E-mail               | ✓        | ✓         |
|                          | Debug interface      | ✓        |           |
|                          | Real-time clock      |          | ✓         |
|                          | Hardware             | ✓        | ✓         |
| V 3.01                   | Application program  | ✓        |           |
|                          | pcom7/JetIP server   |          | ✓         |
|                          | Interpreter          |          | ✓         |
|                          | Hardware             | ✓        |           |
| V 3.11                   | E-mail               | ✓        |           |
|                          | HTTP-Server          | ✓        | ✓         |
|                          | FTP-server           | ✓        |           |
|                          | System bus           | ✓        | ✓         |
|                          | System configuration | ✓        |           |
|                          | Files system         |          | ✓         |
|                          | Interpreter          | ✓        | ✓         |
|                          | Registers            | ✓        |           |
|                          | I/O system           | ✓        |           |

## 1.1 Update Information

### 1.1.1 Registers

Following the update of an os version before V3.11 the registers for dummy modules on the system bus (registers 2023 and 2024), as well as the register containing the start delay (register 2032) will be set to their default values when the controller is switched on for the first time.

| Registers | Default Value |
|-----------|---------------|
| 2023      | -1            |
| 2024      | 255           |
| 2032      | 10            |

## **1.1.2 JetView-Soft**

From this os version on, version V 1.00 of the visualization software JetView-Soft is supported.

## 2 Expansions

### 2.1 System Configuration

Now, names can be specified as IP addresses (e.g. when configuring the SMTP client). To this end, the host name is assigned to the IP address in the file 'hosts' in the directory '/etc'.

If the name cannot be found in the file 'hosts', the operating system tries to achieve name resolution through a DNS server. For this purpose, the IP address of the DNS server is to be specified in the configuration file 'cfgvar.ini' in the directory '/System' or in register 10145.

Due to runtime reasons, addressing through names in the realtime section of the controller (e.g. with the instructions N\_COPY\_TO/FROM) is not possible. Furthermore, direct information on the IP address have to be given there.

#### 2.1.1 File 'hosts'

This file must be stored in the directory '/etc'. It is read out once when the controller is booting. This file has to be created in the same format as the corresponding Windows file. This file is a pure text file.

Example:

```
# Exemplary HOSTS file for JC-24x
192.168.33.209   jetter_mail
192.168.33.208   jetter_demo
192.168.10.211   JC211
```

#### 2.1.2 Domain Name System

If a host name cannot be found in the file 'hosts', the controller tries to resolve the IP address by obtaining the corresponding IP address from a DNS server. During the booting process of the controller the IP address of the DNS server is read out of the configuration memory.

##### 2.1.2.1 Configuration file

The configuration memory can be accessed via the configuration file '/System/cfgvar.ini' by means of an FTP connection with administrator rights.

Example:

```
[CFGVAR]
Version = 4
IP_Address = 192.168. 10. 15
IP_SubNetMask = 255.255.255. 0
IP_DefGateway = 192.168. 10. 1
BasePort = 50000
IP_DNS      = 192.168. 10.209
```

##### 2.1.2.2 Configuration registers

An alternative method to using the configuration file, is accessing the configuration memory through special registers. The IP address of the DNS server can be read out of register 10145.

To modify this address, proceed as follows:

1. Enter value 0x77566152 into the password register 10159
2. Enter the IP address of the DNS server into register 10145
3. Save the configuration by entering any value into register 10100 (the value is of no significance)

## 2.2 E-Mail

The IP addresses in the E-Mail configuration file `./EMAIL/email.ini` can now be specified with their names. The names are resolved during the boot process when the file `./etc/hosts` is read out (see 2.1.1), or by means of a DNS server (see 2.1.2).

Example:

```
[SMTP]
IP=smtp_server

[POP3]
IP=registration
USER=testuser
PASSWORD=test
```

## 2.3 HTTP Server

Some file types have been added to the HTTP server. Now, the HTTP server supports the following file types:

- \*.htm, \*.html, \*.shtml
- \*.txt, \*.ini
- \*.gif, \*.tif, \*.tiff, \*.wbmp
- \*.jpg, \*.jpe, \*.jpeg, \*.png
- \*.xml
- \*.js, \*.jar, \*.java, \*.class, \*.cab
- \*.ocx
- \*.pdf, \*.zip, \*.doc, \*.rtf
- \*.css
- \*.wml, \*.wmlc, \*.wmls, \*.wmlsc

## 2.4 RemoteScan

### 2.4.1 General information

The configurable RemoteScan function is used to cyclically copy register contents from the JetControl to registers of network nodes. On the other hand, they can be read by the nodes and copied in JetControl registers.

The RemoteScan function can be accessed by using special functions 80, 81, and 82.

At the moment, RemoteScan via Modbus/TCP is supported only.

### 2.4.2 Special functions

#### 2.4.2.1 Configuring RemoteScan

To configure RemoteScan, special function 80 is used. However, this function does not start communication (see 2.4.2.2).



SPECIAL\_FUNCTION(80, <Source Register Number>, <Destination Register Number>)

- <Source Register Number> Specifies the number of the first register of the description block.
- <Destination Register Number> Specifies the result register number for this function.

The description block specifies the protocol and the number of communication units. A communication unit specifies the register blocks to be transferred, as well as the address of the communication partner. Up to 10 communication units can be specified. During communication with a partner several different register blocks can be exchanged.

This function can be invoked only with no RemoteScan running. It is not possible to change the configuration during running operation.

2.4.2.1.1 Description Block

| Register Offset      | Description   |  |
|----------------------|---|--|
| 0                    | Protocol  | 1 = JetWay<br>3 = JetIP<br>5 = Modbus/TCP<br>(at the moment, only function 5 possible) |
| 1                    | Amount of following communication units                   | 1 .. 10  |
| Communication Unit 1 |   |  |
| 2                    | Address   | Modbus/TCP: IP-Address   |
| 3                    | Port number   | Modbus/TCP: 502  |
| 4                    | Update rate   | 10 .. 65535 ms   |
| 5                    | Amount of output registers                                | Modbus/TCP: 0 .. 125   |
| 6                    | Output source register number                             | local<br>Modbus/TCP: 8000 .. 8999  |
| 7                    | Output destination register number                        | remote   |
| 8                    | Amount of input registers                                 | Modbus/TCP: 0 .. 125   |
| 9                    | Input source register number                              | remote   |
| 10                   | Input destination register number                         | local<br>Modbus/TCP: 8000 .. 8999  |
| 11                   | Number of the first register of the status register block |  |
| 12                   | Timeout   | in milliseconds  |
| Communication Unit 2 |   |  |
| 13                   | Address   | Modbus/TCP: IP-Address   |
| ..                   | ..  |  |

With modules without input or output registers the corresponding amount has to be set to 0.

If the configuration comprises inputs and outputs, the outputs are sent first, then the inputs are read.

The address (always the first entry of a communication unit) has to be specified directly. It cannot be specified with its name (see 2).

2.4.2.1.2 Status register block

The number of the first register of the status register block, consisting of 3 consecutive registers, has to be specified in the description block of each communication unit. Error messages of this communication unit are stored to this block when RemoteScan is running.

The status block has got the following structure:

| Register Offset | Meaning                                      |  |   |         |
|-----------------|--|--|---|---------|
| 0               | Status (bit-coded)                           | Bit #  | Meaning   |         |
|                 |  | 0  | Scan is running<br>is set after each update cycle               |         |
|                 |  | 1  | An error has occurred<br>is set each time an error has occurred |         |
| 1               | Error code                                   | The code of the last error is displayed. Register Contents | Meaning   |         |
|                 |  | 0  | No error  |         |
|                 |  | < 0  | Application-specific error                                      |         |
|                 |  |  | Modbus/TCP: Code  | Meaning |
|                 |  | -1   | Error in the network driver                                     |         |
|                 |  | -2   | Error in the connection management                              |         |
|                 |  | -3   | Error when sending output registers                             |         |
|                 |  | -4   | Error when reading input registers                              |         |
|                 |  | -5   | Exception response  |         |
|                 |  | -6   | Error when receiving the response                               |         |
| -7              | Wrong transaction ID                         |  |   |         |
| -8              | Timeout                                      |  |   |         |
| 101             | Timeout                                      |  |   |         |
| 102             | Error when reading/writing local registers   |  |   |         |
| 103 / 104       | Error in the lower-level communication layer |  |   |         |
| 2               | Number of errors                             | The number is incremented each time an error occurs.       |   |         |

Note: It is useful to initialize the contents of the status register blocks with 0 before starting the RemoteScan.

#### 2.4.2.1.3 Error messages

After the function has been carried out, the results can be seen in the results register.

| Register Contents | Meaning                                  |
|-------------------|--|
| > 0               | Amount of configured communication units |
| -1                | Protocol not supported                   |
| -2                | Set amount of communication units > 10   |
| -3                | Invalid address or port number           |
| -4                | Invalid register number                  |
| -10               | RemoteScan is already running            |

#### 2.4.2.2 Starting remote scan

Special function 81 is used to start a RemoteScan that has been configured using special function 80.

SPECIAL\_FUNCTION(81, <Source Register Number>, <Destination Register Number>)

<Source Register Number>      Specifies the number of the parameter register.  
 <Destination Register Number>      Specifies the result register number for this function.

No parameters are transferred to this function. Thus, the content of the register to which <Source Register Number> is pointing, is irrelevant.  
 This function always returns value 0 as result.

### 2.4.2.3 Stopping RemoteScan

Special function 82 is used to stop a running RemoteScan. When doing so, all possibly existing communication connections are closed.

SPECIAL\_FUNCTION(82, <Source Register Number>, <Destination Register Number>)

- <Source Register Number> Specifies the number of the parameter register.
- <Destination Register Number> Specifies the result register number for this function.

No parameters are transferred to this function. Thus, the content of the register to which <Source Register Number> is pointing, is irrelevant.  
 This function always returns value 0 as result.

Please mind: Execution of this function may take a relatively long time, depending on the configuration, since it waits until all currently running transfers will be terminated.

## 2.4.3 Registers

| Register 2965: RemoteScan Protocol |   |
|------------------------------------|---|
| Function                           | Description   |
| Read                               | Protocol<br>1 = JetWay<br>3 = JetIP<br>5 = Modbus/TCP |
| Write                              | not possible  |
| Value range                        | 32 bits   |
| Value after reset                  | 0   |

| Register 2966: RemoteScan – Amount of Units |                               |
|---|-------------------------------|
| Function                                    | Description                   |
| Read  | Amount of communication units |
| Write                                       | not possible                  |
| Value range                                 | 32 bits                       |
| Value after reset                           | 0                             |

| Register 2967: RemoteScan Activation |  |
|--------------------------------------|--|
| Function                             | Description                            |
| Read                                 | 0 = Not active<br>1 = active (running) |

|                   |              |
|-------------------|--------------|
| Write             | not possible |
| Value range       |              |
| Value after reset | 0            |

## 2.5 Modbus/TCP

### 2.5.1 Licensing

#### 2.5.1.1 Description

To be able to use the Modbus/TCP communication protocol there has to be a valid license file when JetControl is booting. This license is an individual license, i.e. for each JetControl an individual license file is required.

If the Modbus/TCP protocol is ordered within the framework of a package, a valid license file is included in the delivery. This file must **not** be deleted.

For reorders the serial number on the nameplate of the JetControl has to be indicated.

#### 2.5.1.2 License File

The license file is named "license.dat" and is located in the subdirectory "/licenses". This file is a pure text file.

The file is split-up into certain sections. The key for Modbus/TCP communication is located in the section "MODBUS\_TCP".

The license information for web functions (e-mail, http), if applicable, is also contained in this file. If various functions are purchased at different points in time, the contents of the various license files have to be copied together to create a common "license.dat" file.

#### Example:

```
[ SMTPHTTP ]
CONTROLLER=JetControl24x
SERIALNUMBER=20011023150021
LICENSE=f4d0cf3d577c823f1a45d6119595387000802d2965e3bf55

[ MODBUS_TCP ]
CONTROLLER=JetControl24x
SERIALNUMBER=20011023150021
LICENSE=20ca3ec94895de551afd8d4bf7cc629a5ada0f733fb3e50f
```

#### 2.5.1.3 Registers

The bit-coded register 2930 has been expanded by 2 bits.

| Register 2930: Web Functions |             |  |
|------------------------------|-------------|--|
| Function                     | Description |  |
| Read                         | Bit Number  | Meaning  |
|                              | 0           | 0 = FTP server not available<br>1 = FTP server available           |
|                              | 1           | 0 = HTTP server not available<br>1 = HTTP server available         |
|                              | 2           | 0 = E-mail function not available<br>1 = E-mail function available |
|                              | 3           | 0 = Data file function not available                               |

|                   |                           |   |
|-------------------|---------------------------|---|
|                   |                           | 1 = Data file function available  |
|                   | 4                         | 0 = No Modbus/TCP<br>1 = Modbus/TCP has been licensed                         |
|                   | 5                         | 0 = Modbus/TCP server not available<br>1 = Modbus/TCP server has been started |
| Write             | not possible              |   |
| Value range       | 0 .. 255                  |   |
| Value after reset | Depends on initialization |   |

## 2.5.2 Server

### 2.5.2.1 Description

With a valid license and after a successful start of the Modbus/TCP server, an external client can access registers, inputs and outputs. In doing so, 4 connections may be opened at the same time.

### 2.5.2.2 Supported Instructions

#### 2.5.2.2.1 Register Access

Since only registers with a width of 16 bits can be transferred via Modbus/TCP, access to the high-order 16 bits of JetControl registers is not possible. When receiving register values, sign extension to 32 bits will not be carried out.

#### 2.5.2.2.2 Class 0

##### read multiple registers (fc 3)

Reading register sets.

The number of the start register corresponds to the register number within the JC-24x.

##### write multiple registers (fc 16)

Writing register sets.

The number of the start register corresponds to the register number within the JC-24x.

#### 2.5.2.2.3 Class 1

##### read coils (fc 1)

Reading outputs.

The output number has to be transferred in the internal format of the JC-24x control.

| JetSym – User-oriented numbering format | JC-24x – Internal numbering format |
|---|------------------------------------|
| 101 .. 116                              | 0 .. 15                            |
| 201 .. 216                              | 16 .. 31                           |
| etc.                                    | etc.                               |

##### read input discretes (fc 2)

Reading inputs.

The input number has to be transferred in the internal format of the JC-24x control.

| JetSym – User-oriented numbering format | JC-24x – Internal numbering format |
|---|------------------------------------|
| 101 .. 116                              | 0 .. 15                            |
| 201 .. 216                              | 16 .. 31                           |
| etc.                                    | etc.                               |

##### read input registers (fc 4)

Reading inputs summarized in 16-bit words.

| Inputs     | Register number |
|------------|-----------------|
| 101 .. 116 | 0               |
| 201 .. 216 | 2               |
| 301 .. 316 | 4               |
| etc.       | etc.            |

#### **write coil (fc 5)**

Activating/deactivating an individual output.

The output number has to be transferred in the internal format of the JC-24x control.

| JetSym – User-oriented numbering format | JC-24x – Internal numbering format |
|---|------------------------------------|
| 101 .. 116                              | 0 .. 15                            |
| 201 .. 216                              | 16 .. 31                           |
| etc.                                    | etc.                               |

#### **write single register (fc 6)**

Entering values into the low-order 16 bits of a JC-24x register.

#### 2.5.2.2.4 Class 2

#### **force multiple coils (fc 15)**

Activating/deactivating several outputs.

The output number has to be transferred in the internal format of the JC-24x control.

| JetSym – User-oriented numbering format | JC-24x – Internal numbering format |
|---|------------------------------------|
| 101 .. 116                              | 0 .. 15                            |
| 201 .. 216                              | 16 .. 31                           |
| etc.                                    | etc.                               |

#### **read / write registers**

Writing and simultaneously reading registers.

The number of the start register corresponds to the register number within the JC-24x. First, the registers polled by the client are read, then, the registers transferred from the client are stored.

## 2.5.3 Client

### 2.5.3.1 General information

The Modbus/TCP client in JetControl 24x controllers supports Class 0 Conformance (see 0). This means that read and write multiple registers instructions are used. Up to 125 registers with a width of 16 bits can be transmitted in one frame. When sending 32-bit registers only the lower-order 16 bits are transmitted. When assigning incoming register values to the JetControl-internal 32-bit registers sign extension to 32 bits will not be carried out.

As protocol ID "0" is used, as unit ID "1". Assignment of sent and received frames is carried out using the transaction ID.

Connections to 11 different servers may be opened at the same time.

### 2.5.3.2 RemoteScan

(see 2.4) This function is for cyclically transferring the inputs and outputs 20001 through 36000 that are combined in the 16-bit registers 8000 through 8999 from and to the configured servers. One connection is established to each server (IP address and port) irrespective of the number of communication units configured on this server.

If several communication units are configured on one server, accesses are serialized since servers, as a rule, do not support "command pipelining". If several servers have been configured, communication is carried out in parallel.

2.5.3.2.1 Overlaying of inputs/outputs on registers

| Registers | Inputs / Outputs |
|-----------|------------------|
| 8000      | 20001 – 20016    |
| 8001      | 20017 – 20032    |
| 8002      | 20033 – 20047    |
| etc.      | etc.             |

Since the registers and the inputs/outputs overlaid on them merely are memory cells located in the RAM, and no direct mapping to hardware takes place, it is not determined whether a register contains inputs or outputs. Assignment is made not until configuration in the communication units takes place.

2.5.3.3 Special functions

As acyclic transmission channel to a Modbus/TCP server the special functions 65 (reading registers) and 66 (writing registers) can be used (the functions are available independent of RemoteScan).

While one of these two special functions is being carried out simultaneous calls of this functions in other tasks are blocked until this function will be terminated.

These functions establish a connection to the specified server, transmit the desired data and clear down the connection. In case a connection exists that has been established by RemoteScan, this connection will be used. Setting-up and clearing-down the connection is not required.

The IP address is always to be specified directly. It cannot be specified with its name (see 2).

**Please mind:** It is not advisable to issue TaskBreak or TaskRestart instructions for this task or to restart the program through JetSymb while one of these functions is carried out since in such a case the connection remains established which may block additional transmissions.

2.5.3.3.1 Reading registers

SPECIAL\_FUNCTION(65, <Source Register Number>, <Destination Register Number>)

<Source Register Number> Specifies the number of the first register of a description block.

<Destination Register Number> Specifies the result register number for this function.

| Description Block |                                    |                 |
|-------------------|------------------------------------|-----------------|
| Register Offset   | Meaning                            |                 |
| 0                 | IP-Address                         |                 |
| 1                 | Port number                        | 502             |
| 2                 | Timeout                            | in milliseconds |
| 3                 | Number of the source register      | remote          |
| 4                 | Number of the designation register | local           |
| 5                 | Amount of registers                | 1 .. 125        |

| Result   | Meaning                        |
|----------|--------------------------------|
| 0        | No error                       |
| -1 or -2 | Error during connection set-up |
| -4       | Error during data transfer     |

|     |                           |
|-----|---------------------------|
| -5  | Error message from server |
| -8  | Timeout                   |
| -10 | No Modbus/TCP license     |

### 2.5.3.3.2 Writing to registers

SPECIAL\_FUNCTION(66, <Source Register Number>, <Destination Register Number>)

<Source Register Number> Specifies the number of the first register of a description block.

<Destination Register Number> Specifies the result register number for this function.

| Description Block |                                    |                 |
|-------------------|------------------------------------|-----------------|
| Register Offset   | Meaning                            |                 |
| 0                 | IP-Address                         |                 |
| 1                 | Port number                        | 502             |
| 2                 | Timeout                            | in milliseconds |
| 3                 | Number of the source register      | local           |
| 4                 | Number of the designation register | remote          |
| 5                 | Amount of registers                | 1 .. 125        |

| Result   | Meaning                        |
|----------|--------------------------------|
| 0        | No error                       |
| -1 or -2 | Error during connection set-up |
| -3       | Error during data transfer     |
| -5       | Error message from server      |
| -8       | Timeout                        |
| -10      | No Modbus/TCP license          |

## 2.6 Data Files

### 2.6.1 General information

The latest register values and flag states can be written into a file or read-out of a file with the help of some special functions. This read/write process is controlled by the application program.

The file format is identical to the format of the "data dump" files created by JetSymb (see 2.6.4).

The file names consist of two constant parts and a register content. So, files can be selected by different register values (see 0).

Written files are stored to the root directory (highest directory level). Files to be loaded must also be located in the root directory. Access to data files is carried out with administrator rights and cannot be restricted.

### 2.6.2 Special functions

#### 2.6.2.1 Implementation

Since file operations may take considerable long time, especially with large files, other application tasks are processed while one of the file operations is running. However, only one function can be processed at a time. Tasks which invoke one of these functions while a file operation of another task is running are therefore blocked until this operation is completed.



That implies that data consistency of value blocks to be written or read is not ensured. Data consistency has to be ensured by accordingly programming the application program. The state of the currently running operation can be polled through the registers specified below (see 2.6.3).

**2.6.2.2 File names**

File names always start with "Data\_" followed by a numerical value and the extension "da". The numerical value for drawing a distinction between various files is acquired from the parameter register of the special functions. When doing so, positive or negative values are allowed.

Examples:

- Data\_123456789.da
- Data\_0.da
- Data\_-100.da

Please mind: Observe capitalization. The file system is case-sensitive.

**2.6.2.3 Saving values – creating a file**

Special function **90** is for creating a new data file and inserting a selectable register or flag block into this file.

SPECIAL\_FUNCTION(90, <Source Register Number>, <Destination Register Number>)

- <Source Register Number> Specifies the number of the first register of the parameter block.
- <Destination Register Number> Specifies the result register number for this function.

**2.6.2.3.1 Parameter block**

The function parameters are specified starting from register <Source Register Number>.

| Register Offset | Meaning             |  |
|-----------------|---------------------|--|
| 0               | "File Name"         | Numerical component of the file name (see 0) |
| 1               | Type                | 1 = Register<br>3 = Flag                     |
| 2               | Start of data block | Number of the first register or flag         |
| 3               | End of data block   | Number of the last register or flag          |

**2.6.2.3.2 Function result**

The result of the function can be read out of register <Destination Register Number>.

| Register Contents | Meaning  |
|-------------------|--|
| 0                 | No error                                       |
| -1                | Error when creating a file (e.g. disk is full) |
| -2                | Error when writing data                        |
| -4                | Error when closing a file                      |
| -6                | Invalid register / flag number                 |
| -10               | Data file function not available (see 2.6.3)   |
| -20               | Internal os error                              |

**2.6.2.4 Saving values – append to file**

Special function **91** is for appending a selectable register or flag block to an existing file. In case this file does not exist, a new file will be created.

SPECIAL\_FUNCTION(91, <Source Register Number>, <Destination Register Number>)

<Source Register Number> Specifies the number of the first register of the parameter block.

<Destination Register Number> Specifies the result register number for this function.

#### 2.6.2.4.1 Parameter block

The function parameters are specified starting from register <Source Register Number>.

| Register Offset | Meaning             |  |
|-----------------|---------------------|--|
| 0               | "File Name"         | Numerical component of the file name (see 0) |
| 1               | Type                | 1 = Register<br>3 = Flag                     |
| 2               | Start of data block | Number of the first register or flag         |
| 3               | End of data block   | Number of the last register or flag          |

#### 2.6.2.4.2 Function result

The result of the function can be read out of register <Destination Register Number>.

| Register Contents | Meaning                                      |
|-------------------|--|
| 0                 | No error                                     |
| -1                | Error when opening or creating file          |
| -2                | Error when writing data                      |
| -4                | Error when closing a file                    |
| -6                | Invalid register / flag number               |
| -10               | Data file function not available (see 2.6.3) |
| -20               | Internal os error                            |

### 2.6.2.5 Reading values from a file

Special function **92** is for reading register values and flag states out of a data file and entering them into the corresponding registers or flags. The information is processed in the order specified by the content of the file.

SPECIAL\_FUNCTION(92, <Source Register Number>, <Destination Register Number>)

<Source Register Number> Specifies the number of the register containing the numerical component of the file name.

<Destination Register Number> Specifies the result register number for this function.

#### 2.6.2.5.1 Parameters

The function parameter is specified in register<Source Register Number>.

| Register Offset | Meaning     |  |
|-----------------|-------------|--|
| 0               | "File Name" | Numerical component of the file name (see 0) |

#### 2.6.2.5.2 Function result

The result of the function can be read out of register <Destination Register Number>.

| Register Contents | Meaning |
|-------------------|---------|
|-------------------|---------|

|     |   |
|-----|---|
| 0   | No error                                      |
| -1  | Error when opening file (e.g. file not found) |
| -3  | Error when reading data                       |
| -4  | Error when closing a file                     |
| -10 | Data file function not available (see 2.6.3)  |
| -20 | Internal os error                             |

### 2.6.2.6 Deleting a file

Special function **96** is for deleting a data file from the flash disk.

SPECIAL\_FUNCTION(96, <Source Register Number>, <Destination Register Number>)

<Source Register Number> Specifies the number of the register containing the numerical component of the file name.

<Destination Register Number> Specifies the result register number for this function.

#### 2.6.2.6.1 Parameters

The function parameter is specified in register<Source Register Number>.

| Register Offset | Meaning     |  |
|-----------------|-------------|--|
| 0               | "File Name" | Numerical component of the file name (see 0) |

#### 2.6.2.6.2 Function result

The result of the function can be read out of register <Destination Register Number>.

| Register Contents | Meaning  |
|-------------------|--|
| 0                 | No error   |
| -5                | Error when deleting the file (e.g. file not found) |
| -10               | Data file function not available (see 2.6.3)       |
| -20               | Internal os error                                  |

## 2.6.3 Registers

### 2.6.3.1 Availability

The bit-coded register 2930 has been expanded by 1 bit.

| Register 2930: Web Functions |             |  |
|------------------------------|-------------|--|
| Function                     | Description |  |
| Read                         | Bit Number  | Meaning  |
|                              | 0           | 0 = FTP server not available<br>1 = FTP server available                 |
|                              | 1           | 0 = HTTP server not available<br>1 = HTTP server available               |
|                              | 2           | 0 = E-mail function not available<br>1 = E-mail function available       |
|                              | 3           | 0 = Data file function not available<br>1 = Data file function available |
|                              | 4           | 0 = No Modbus/TCP<br>1 = Modbus/TCP has been licensed                    |
|                              | 5           | 0 = Modbus/TCP server not available                                      |

|                   |  |
|-------------------|--|
|                   | 1 = Modbus/TCP server has been started |
| Write             | not possible                           |
| Value range       | 0 .. 255                               |
| Value after reset | Depends on initialization              |

### 2.6.3.2 Status Register

The processing state of a file operation and the number of the task carrying out the operation can be read from two registers.

| Register 2977: Processing State |   |
|---------------------------------|---|
| Function                        | Description   |
| Read                            | 0: No data file operation in progress<br>1: Processing transferred to file module<br>2: Data is being read/written<br>3: File operation completed |
| Write                           | illegal   |
| Value range                     | 0 .. 255  |
| Value after reset               | 0   |

| Register 2978: Task number |  |
|----------------------------|--|
| Function                   | Description  |
| Read                       | Number of the task performing a file operation<br>0 .. 99: Task number<br>255: No task |
| Write                      | illegal  |
| Value range                | 0 .. 255   |
| Value after reset          | 255  |

## 2.6.4 File format

These files are pure text files with one line for each entry. The entries are to be terminated with "carriage return / line feed". Comment lines are allowed.

A data file is to start with the entry "SD1001".

Data lines start with an identifier for the variable. Now follows the variable number, also separated by a blank or tab. Now follows the value of the variable also separated by a blank or tab.

The IDs at the beginning of a line must not be indented.

| Variable ID | Variable type                   |
|-------------|---------------------------------|
| FS          | Flags                           |
| QS          | Floating point numeral register |
| RS          | Integer Register                |

---

All lines that do not start with such a variable ID are regarded as comment lines with the exception of the first line containing the file ID.

**Example:**

```
SD1001
; JC-24x DATA FILE - Jetter AG
FS    111    1
This is a comment
RS    20112  110
    FS    113    1
QS    65024  -3.141593
QS    65025  6.789e-05
```

The third line from below is also a comment line since the variable's ID ("FS") does not stand at the beginning of the line.

## 2.7 Floating Point Register

Special register **2909** offers the option to define an additional range of floating point registers. This register is for entering the register number starting from which registers 20000 through 49999 are to be treated as floating point registers. The already existing floating point register 65024 and the following registers as well as the integer registers below 20000 remain unaffected.

Register 2909 is protected against power loss and is not initialized by the os. That is, the setting stored here is kept when the controller is switched off.

**Example:**

Content of register 2909 = 25000

Registers from 0 through 11999 and from 20000 through 24999 are treated as integer registers. Registers from 25000 through 49999 and from 65024 through 65279 are treated as floating point registers.

Note: When modifying register 2909 the contents of the storage locations for registers remain unchanged. The only thing is that they are interpreted in a different way.





This register is initialized with the value 65024 by factory default.

## 2.8 System Bus

### 2.8.1 Third-party modules

The JetControl 24x system bus allows Jetter AG brand modules, as well as third-party modules to be connected to it. Basically, these modules are treated as JX-SIO modules.

JetControl 24x with software version V3.11 supports the following modules:

| <b>Modules at the System Bus Produced by Other Manufacturers</b> |   |
|--|---|
| <b>Manufacturer</b>  | <b>Product Designation</b>  |
| Bürkert GmbH & Co. KG  | Valve terminal type 8640<br> Bürkert_BI_100_Benutzerinformation  |
| Festo AG & Co.   | CPV10-GE-CO2-8<br>CPV14-GE-CO2-8<br>CPV18-GE-CO2-8<br>CPX-FB14<br> Festo_BI_100_BenutzerInformation   |
| SMC Pneumatik GmbH   | SI unit EX120 - SCA1<br>SI unit EX121 - SCA1<br>SI unit EX122 - SCA1<br> SMC_BI_100_Benutzerinformation  |
| Lenze GmbH & Co. KG  | Variable speed drive with special field bus function module 2175<br>8201 – 8204<br>8211 – 8218<br>8221 – 8227<br>8241 – 8246<br>8200 vector<br>8200 vector, Cold plate<br> Lenze_8200vector_BI_100_Benutzerinformation |
| Lumberg-Jetter GmbH & Co. KG                                     | LJX-CB-ID16<br>LJX-CB-ID16-NPN<br>LJX-CB-OD8<br>LJX-CB-OD16<br>LJX-CB-IO16  |

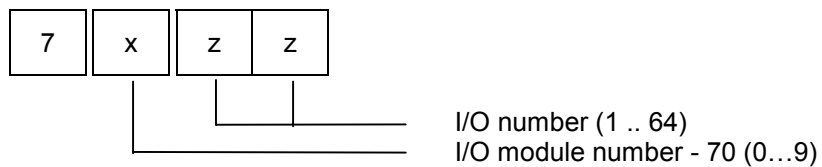
These third-party modules are automatically detected and configured. Additional commissioning software is not required. When connecting such modules, refer to the manuals provided by the corresponding manufacturer. Additional information is available from Jetter AG describing how to operate these modules on the Jetter system bus.

### 2.8.1.1 Register and I/O numbers

Access to inputs and outputs of system bus modules is possible, in the usual way, through I/O instructions of the JetSym(ST) programming language. Registers on these modules can also be accessed by using the register instructions. Distinction between I/Os and registers is made by the numbering.

| I/O and Register Ranges |   |
|-------------------------|---|
| Range                   | Description   |
| IN 7000 ... IN 7999     | Digital inputs of third-party modules                 |
| OUT 7000 ... OUT 7999   | Digital outputs of third-party modules                |
| Reg 5000 ... Reg 5999   | I/O register overlaying of digital and analog inputs  |
| Reg 6000 ... Reg 6999   | I/O register overlaying of digital and analog outputs |
| Reg 7000 ... Reg 7999   | Configuration and diagnostic registers                |

#### Coding of I/O numbers



#### Register code for I/O register overlaying and analog inputs



**Register code for I/O register overlaying and analog outputs****Register code for configuration and diagnostic registers****2.8.1.2 Module array**

In the module array, an unambiguous module code is assigned to each detected module. The module array is accessible via register 2015 and register 2016.

| <b>Register 2015: Pointer to the module array</b> |                                   |
|---|-----------------------------------|
| <b>Function</b>                                   | <b>Description</b>                |
| Read  | Selected module                   |
| Write   | Selection of a specific module    |
| Value range                                       | 0 up to the amount of I/O modules |
| Value after reset                                 | 0                                 |



| <b>Register 2016: Module array</b> |   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
|------------------------------------|---|---|---------|---|---------|---|----------|---|---------|---|---------|---|----------|---|----------|---|----------|---|-----------|----|-------------|----|-----------------|----|------------|----|-------------|----|-------------|----|--------|----|------------------|----|--------------------|----|-----------------------------------|----|--------------------------|----|----------------------------|-----|---------|-----|----------|-----|---------|-----|----------|-----|----------|-----|-----------|-----|--------------------|-----|------------|-----|--------------------|-----|---------------------|-----|------------------------|-----|----------------------|
| <b>Function</b>                    | <b>Description</b>  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| Read                               | <p>Module array</p> <p>2016 = Amount of modules</p> <p>2016 = Code of the first module</p> <p>2016 = Code of the second module</p> <p>Code:</p> <p><b>JX2-I/O Modules</b></p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 5%;">0</td><td>JX2-OD8</td></tr> <tr><td>1</td><td>JX2-ID8</td></tr> <tr><td>2</td><td>JX2-IO16</td></tr> <tr><td>3</td><td>JX2-IA4</td></tr> <tr><td>4</td><td>JX2-OA4</td></tr> <tr><td>5</td><td>JX2-CNT1</td></tr> <tr><td>6</td><td>JX2-PRN1</td></tr> <tr><td>7</td><td>JX2-SER1</td></tr> </table> <p><b>JX-SIO and Third-Party Modules</b></p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 5%;">9</td><td>JX-TP20-R</td></tr> <tr><td>10</td><td>LJX-CB-ID16</td></tr> <tr><td>11</td><td>LJX-CB-ID16-NPN</td></tr> <tr><td>12</td><td>LJX-CB-OD8</td></tr> <tr><td>13</td><td>LJX-CB-OD16</td></tr> <tr><td>14</td><td>LJX-CB-IO16</td></tr> <tr><td>64</td><td>JX-SIO</td></tr> <tr><td>65</td><td>Festo CPV Direct</td></tr> <tr><td>66</td><td>Festo Terminal CPX</td></tr> <tr><td>67</td><td>Buerkert valve terminal type 8640</td></tr> <tr><td>68</td><td>SMC SI unit EX12# - SCA1</td></tr> <tr><td>70</td><td>Lenze fieldbus module 2175</td></tr> </table> <p><b>JX2-Slave Modules</b></p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 5%;">128</td><td>JX2-SV1</td></tr> <tr><td>129</td><td>CAN-DIMA</td></tr> <tr><td>130</td><td>JX2-SM2</td></tr> <tr><td>131</td><td>JX2-SM1D</td></tr> <tr><td>132</td><td>JX2-PID1</td></tr> <tr><td>133</td><td>JX2-PROF1</td></tr> <tr><td>135</td><td>JetMove 200 Series</td></tr> <tr><td>136</td><td>JX2-ProfiM</td></tr> <tr><td>146</td><td>JetMove 600 series</td></tr> </table> <p><b>dummy modules</b></p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 5%;">252</td><td>JX-SIO dummy-module</td></tr> <tr><td>253</td><td>JX2-Slave Dummy Module</td></tr> <tr><td>254</td><td>JX2-I/O Dummy Module</td></tr> </table> | 0 | JX2-OD8 | 1 | JX2-ID8 | 2 | JX2-IO16 | 3 | JX2-IA4 | 4 | JX2-OA4 | 5 | JX2-CNT1 | 6 | JX2-PRN1 | 7 | JX2-SER1 | 9 | JX-TP20-R | 10 | LJX-CB-ID16 | 11 | LJX-CB-ID16-NPN | 12 | LJX-CB-OD8 | 13 | LJX-CB-OD16 | 14 | LJX-CB-IO16 | 64 | JX-SIO | 65 | Festo CPV Direct | 66 | Festo Terminal CPX | 67 | Buerkert valve terminal type 8640 | 68 | SMC SI unit EX12# - SCA1 | 70 | Lenze fieldbus module 2175 | 128 | JX2-SV1 | 129 | CAN-DIMA | 130 | JX2-SM2 | 131 | JX2-SM1D | 132 | JX2-PID1 | 133 | JX2-PROF1 | 135 | JetMove 200 Series | 136 | JX2-ProfiM | 146 | JetMove 600 series | 252 | JX-SIO dummy-module | 253 | JX2-Slave Dummy Module | 254 | JX2-I/O Dummy Module |
| 0                                  | JX2-OD8   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 1                                  | JX2-ID8   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 2                                  | JX2-IO16  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 3                                  | JX2-IA4   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 4                                  | JX2-OA4   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 5                                  | JX2-CNT1  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 6                                  | JX2-PRN1  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 7                                  | JX2-SER1  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 9                                  | JX-TP20-R   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 10                                 | LJX-CB-ID16   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 11                                 | LJX-CB-ID16-NPN   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 12                                 | LJX-CB-OD8  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 13                                 | LJX-CB-OD16   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 14                                 | LJX-CB-IO16   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 64                                 | JX-SIO  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 65                                 | Festo CPV Direct  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 66                                 | Festo Terminal CPX  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 67                                 | Buerkert valve terminal type 8640   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 68                                 | SMC SI unit EX12# - SCA1  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 70                                 | Lenze fieldbus module 2175  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 128                                | JX2-SV1   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 129                                | CAN-DIMA  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 130                                | JX2-SM2   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 131                                | JX2-SM1D  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 132                                | JX2-PID1  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 133                                | JX2-PROF1   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 135                                | JetMove 200 Series  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 136                                | JX2-ProfiM  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 146                                | JetMove 600 series  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 252                                | JX-SIO dummy-module   |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 253                                | JX2-Slave Dummy Module  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |
| 254                                | JX2-I/O Dummy Module  |   |         |   |         |   |          |   |         |   |         |   |          |   |          |   |          |   |           |    |             |    |                 |    |            |    |             |    |             |    |        |    |                  |    |                    |    |                                   |    |                          |    |                            |     |         |     |          |     |         |     |          |     |          |     |           |     |                    |     |            |     |                    |     |                     |     |                        |     |                      |

|                   |                               |                |
|-------------------|-------------------------------|----------------|
|                   | 255                           | not identified |
| Write             | illegal                       |                |
| Value range       | 0 - 255                       |                |
| Value after reset | Quantity of expansion modules |                |

## 2.8.2 Function and power-level terminals on the JX-SIO module

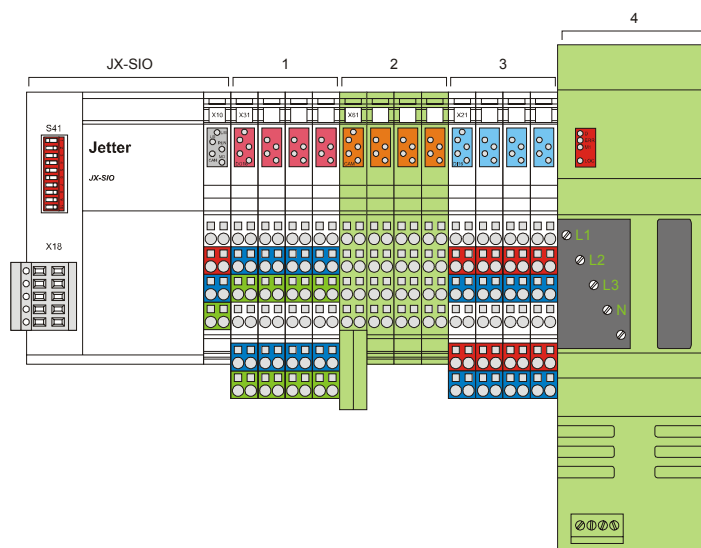
The JX-SIO module allows digital and analog I/O terminals, as well as function and power-level terminals to be connected to it. In the following text the term function terminal is used for both, function and power-level terminals.

The JX-SIO module supports the following function terminals:

- IB IL 400 MLR 1-8A
- IB IL 400 ELR 1-3A
- IB IL 400 ELR R-3A
- INLINE CAM produced by Deutschmann Automation GmbH

| Overview: JX-SIO Function Terminal Registers |                                   |
|--|-----------------------------------|
| Register #                                   | Description                       |
| 7x04   | Index to function terminal        |
| 7x05   | Status of function terminals      |
| 7x06   | Input data of function terminals  |
| 7x07   | Output data of function terminals |

For detailed information on communication between CPU und function terminals please refer to the User Information on Smart I/O modules.



**Abb. 1: Smart I/O Module with function terminal INLINE CAM and power-level terminal IB IL 400 MLR 1-8A**

## 2.8.3 Special flags associated to the system bus

The JetControl 24x module indicates certain error conditions through special flags. This feature facilitates detailed error analysis. After the JetControl 24x module has been switched on it deletes all special flags.

| <b>Overview: Special flags associated to the system bus</b> |   |
|---|---|
| <b>Special Flags</b>  | <b>Description</b>  |
| 2048  | Timeout during an access to a JX2-I/O module  |
| 2049  | Timeout during an access to a JX2-Slave module  |
| 2050  | Timeout during an access to a JX2-I/O register  |
| 2065  | Activated error signalization for output driver errors  |
| 2067  | Fatal system bus error  |
| 2068  | An accumulation of errors on the system bus interface occurred.   |
| 2270  | Timeout during an access to an inactive JX-SIO module   |
| 2272  | Access to an unknown JX-SIO register  |
| 2273  | Access to a register which is not supported by this JX-SIO module, e.g. configuration of an analog input, although there is no analog input terminal. |
| 2274  | Timeout when monitoring a JX-SIO module   |
| 2275  | JX-SIO has carried out an internal reset, or is not ready   |

## 2.8.4 Access to JX-SIO analog values

So far, registers 5x60 through 5x71, as well as registers 6x60 through 6x71 were accessed directly on processing a REGISTER-instruction.

In order to boost performance, analog inputs are now read in the background after all task have been completed. If special flag 2059 is set, analog inputs are read even after each task switch event.

All analog output that have changed are transferred to the JX-SIO at the end of a task.

## 2.8.5 Timeout register

The timeout for reading SDO data can be set in register 2073.

## 2.8.6 JetMove 200

Data transmission via system bus to the JetMove 200 is fully supported.

## 2.8.7 Festo CPV-Direct

Register numbers for condition counter setpoints have been moved from 7x50 through 7x65 to 7x30 through 7x45.

## 2.8.8 Sync telegram

Some of the CANopen modules require a sync telegram to be sent in regular intervals. This interval can be set in register 2074.

Practicable values for frequency inverters, for example, are between 100 and 250 milliseconds.

If this register contains 0, no sync telegrams will be created.

| <b>Register 2074: Sync Time Base</b> |  |
|--------------------------------------|--|
| <b>Function</b>                      | <b>Description</b>                       |
| Read                                 | Set time base for CANopen sync telegrams |
| Write                                | New time base setting                    |
| Value range                          | 0 .. 255 [ms]                            |
| Value after reset                    | 0 (no sync telegram)                     |

## 2.9 FTP Server

So far, on establishing a connection with an FTP client a data connection was immediately established, too. Not before that, the existence of the required file on the flash disk was checked. Some client programs had problems coping with that approach. Starting from this os version, the data connection will be established only in case the required file is located on the flash disk.

## 3 Eliminated Software Bugs

### 3.1 System Bus

#### 3.1.1 Response of analog outputs on Smart I/O modules to errors

The response of analog outputs on Smart I/O modules to errors can be configured. Smart I/O modules are capable of detecting an interrupted system bus line to the JetControl. In such a case they will output previously configured values via their analog outputs. Up to version 3.11, the configuration data would not be transmitted correctly. At the analog outputs, value 0000<sub>hex</sub> would always be output in a fault condition.

#### 3.1.2 Registers on Smart I/O modules

Access to registers 7xzz is always possible with all CANopen modules even if the module is "invalid". This feature facilitates detailed error analysis in case of a failure. Access to registers 7x90 through 7x99 is possible even if the CANopen module has not been detected at all.

#### 3.1.3 Analog values

The sign extension of analog values has been corrected.

#### 3.1.4 Analog terminals on Smart I/O modules

Analog terminals on JX-SIO modules were counted with the incorrect I/O sum. The result was that less I/Os than specified could be connected to the corresponding JetControl.

### 3.2 File System

#### 3.2.1 File names

When reading or writing a file, the names of files already existing on the flash disk were not compared at full length with the passed file names. Due to this bug already existing files could have been overwritten by a write access. With a read access it could have happened that the contents of the wrong file were transferred.

Example 1:

- A file named "jetterAG" already exists on the flash disk.
- A file named "jetter" is to be copied into the same directory.
- It could happen that the existing file "jetterAG" was overwritten with the contents of the file "jetter".

Example 2:

- The files "jetter.txt", "jetter.t1", and "jetter.t" are located in a directory on the flash disk.
- The file "jetter.t" is to be read.
- Which of the 3 files' content was returned depended on the fact which one of the files was found on the disk first.

#### 3.2.2 Concurrent access to several files

When two or more files were concurrently opened and closed, it could happen that the controller got into an internal error condition preventing communication and control

function from being carried out. The only solution to this state was restarting the controller (by turning the power supply off and on again).

### 3.3 Arithmetic Operations

In os versions before V. 3.00 an arithmetic calculation to the "right" of the assignment operator ("="), independent of the type of register being used in this calculation, was carried out as floating point calculation if the register to which the result was to be assigned was a floating point register.

Starting from version V. 3.00 the type of register to the "left" of "=" was not taken into account in calculations so that different results may have occurred, especially with divisions.

Now, in the latest os version the type of the destination register into account as far as local floating point registers in the range from 20000 through 49999 (see 2.7) and from 65024 through 65279 are concerned. If floating point results are to be assigned to floating point slave registers, then floating point registers have also to be used within calculations.

Example:

REG 65024 = 10 / 3 \* 2

In os versions V. 3.00 to versions before V. 3.11 the result assigned to floating point register 65024 was 6.

Starting from os V. 3.11 the assigned result is 6,666667 as in the versions before V. 3.00.

### 3.4 UserInput

When the "USER\_INPUT" instruction was used in os versions before V. 3.00 in connection with an integer register as destination register and a value with decimal positions all entered digits were stored in the destination register. At the same time, the number of decimal positions was stored in special register 2807/2808.

Starting from version V. 3.00 only the rounded part to the left of the decimal point was stored to the destination register. However, if the value was assigned to a floating point register, it was stored correctly.

Starting from this version, the USER\_INPUT instruction shows the same behavior as it did before V. 3.00. However, the entered value will correctly be assigned to floating point registers only if they are local floating point registers in the range from 20000 through 49999 (see 2.7) and from 65024 through 65279. With the USER\_INPUT instruction it is not possible to directly use floating point slave registers.

Example:

The value "123.65" is entered via LCD HMI.

In all cases, on completion of the USER\_INPUT instruction register 2807 (divisor) contains the value 100, and register 2808 (number of decimal positions) the value 2.

| Destination Register          | Register Contents |         |
|-------------------------------|-------------------|---------|
|                               | V. 3.00/V. 3.01   | V. 3.11 |
| 100                           | 124               | 12365   |
| 65024                         | 123.65            | 123.65  |
| Floating point slave register | 123.65            | 12365   |

### 3.5 HTTP Server

The MIME type was not correctly specified when transferring JPEG graphic files. Due to this, some browsers were not able to display such graphics.