



User Manual

JetControlMobile 631
Controller

60885160_01

We automate your success.

This document has been compiled by Jetter AG with due diligence, and based on the known state of the art. Revisions and further development of our products are not automatically mentioned in a reviewed document. Jetter AG shall not be liable for errors in form or content, or for missing updates, as well as for damages or disadvantages resulting from such failure.



Jetter AG
Graeterstrasse 2
71642 Ludwigsburg
Germany

Phone

Switchboard	+49 7141 2550-0
Sales	+49 7141 2550-531
Technical Hotline	+49 7141 2550-444

E-mail

Technical Hotline	hotline@jetter.de
Sales	sales@jetter.de

Translation of the german original User
Manual

Revision	1.00.2
Date of issue	7/6/2021

Table of Contents

- 1 Introduction 6**
 - 1.1 Information on this document 6
- 2 Safety 7**
 - 2.1 General information 7
 - 2.2 Purpose 7
 - 2.2.1 Intended use 7
 - 2.2.2 Usage other than intended 7
 - 2.3 Warnings used in this document 7
- 3 Product description 8**
 - 3.1 Design 8
 - 3.2 Product features 9
 - 3.3 LED indicators 9
 - 3.3.1 Diagnostic capability via LEDs 9
 - 3.4 Nameplate 10
 - 3.5 Scope of delivery 10
- 4 Technical specifications 11**
 - 4.1 Dimensions 11
 - 4.2 Mechanical specifications 11
 - 4.3 Electrical properties 12
 - 4.3.1 CPU core and operating system 13
 - 4.3.2 Ports and interfaces 13
 - 4.3.3 Inputs and outputs 15
 - 4.4 Environmental conditions 16
 - 4.5 EMC values 18
- 5 Mechanical installation 19**
 - 5.1 Requirements for installation location and mounting surface 19
 - 5.2 Preparing for installation 20
 - 5.3 Mounting orientation 20
 - 5.3.1 Allowed mounting orientations 20
 - 5.3.2 Prohibited mounting orientations 21
- 6 Electrical connection 22**
 - 6.1 Pin assignment 23
 - 6.1.1 X71 - Connector (male) 23
 - 6.1.2 X13 - Service connector 1 (USB + RS-232) 24
 - 6.1.3 X14 – Service connector 2 (Ethernet) 25

- 6.1.4 X61 ... X66 – CAM1 ... CAM6..... 25
- 6.1.5 X81, X82 – Display 1 and Display 2 25
- 6.2 Commissioning..... 26
 - 6.2.1 Connecting the voltage supply 26
 - 6.2.2 Booting up the controller 26
 - 6.2.3 Setting up an Ethernet connection between controller and PC..... 27
- 7 Identification and Configuration..... 28**
 - 7.1 Identification 28
 - 7.1.1 Device web site 28
 - 7.1.2 Electronic Data Sheet (EDS)..... 31
 - 7.1.3 Version registers 31
 - 7.2 Operating system 33
 - 7.2.1 Operating system update of the controller 33
 - 7.3 File System..... 39
 - 7.3.1 Properties 39
 - 7.4 Installing the USB/CAN adapter 41
- 8 Programming..... 42**
 - 8.1 Abbreviations, module register properties and formats..... 42
 - 8.2 Storage options - Overview 43
 - 8.2.1 Operating system memory 43
 - 8.2.2 File system memory 43
 - 8.2.3 Application program memory 43
 - 8.2.4 Memory for volatile application program variables 44
 - 8.2.5 Memory for non-volatile application program registers 44
 - 8.2.6 Memory for non-volatile application program variables 45
 - 8.2.7 Special registers 45
 - 8.2.8 Inputs and outputs 45
 - 8.2.9 Flag 46
 - 8.3 Saving and loading an application program 47
 - 8.4 CANopen® STX API 48
 - 8.4.1 STX Functions 48
 - 8.4.2 Heartbeat monitoring 49
 - 8.4.3 CANopen® Object dictionary 49
 - 8.5 SAE J1939 STX API..... 51
 - 8.5.1 STX Functions 51
 - 8.6 Serial interface RS-232 53
 - 8.6.1 Operating principle 53
 - 8.6.2 Registers - Description 54
 - 8.6.3 Programming 59

8.7	Controlling displays	61
8.8	FTP server.....	63
8.9	Monitoring interface activities	64
8.10	User-programmable IP interface	65
9	Registers - Overview.....	66
10	Maintenance and repairs	75
10.1	Maintenance, repairs and disposal.....	75
10.2	Storage and shipment	75
11	Service	76
11.1	Customer service	76
12	Spare parts and accessories	77
12.1	Accessories	77
12.1.1	Monitors.....	77
12.1.2	USB flash drives	77

1 Introduction

1.1 Information on this document

This document forms an integral part of the product and must be read and understood prior to using it. It contains important and safety-related information for the proper use of the product as intended.

Target groups

This document is intended for specialists with appropriate qualifications. Only competent and trained personnel is allowed to put this device into operation. During the whole product life cycle, safe handling and operation of the device must be ensured. In the case of missing or inadequate technical knowledge or knowledge of this document any liability is excluded.

Availability of information

Make sure this document is kept at the ready in the vicinity of the product throughout its service life.

For information on new revisions of this document, visit the download area on our website. This document is not subject to any updating service.

[Start | Jetter - We automate your success.](#)

For further information refer to the following information products:

- Version updates
Information about new versions of software products or of the operating system of your controller.
- JetSym software Online Help
Detailed description of software functions with application examples
- Application-oriented manuals
Cross-product documentation

2 Safety

2.1 General information

When placed on the market, this product corresponds to the current state of science and technology.

In addition to the operating instructions, the laws, regulations and guidelines of the country of operation or the EU apply to the operation of the product. The operator is responsible for compliance with the relevant accident prevention regulations and generally accepted safety rules.

RoHS 2

The device conforms to the EU directive 2011/65/EU (RoHS 2).

2.2 Purpose

2.2.1 Intended use

The device is intended for controlling applications in the field of commercial vehicles and mobile machines.

Operate the device only in accordance with the intended conditions of use, and within the limits set forth in the technical specifications.

Intended use of the product includes its operation in accordance with this manual.




2.2.2 Usage other than intended

Do not use the device in technical systems for which a high level of fail-safety is required.

Machinery Directive

This device is no safety-related part as per Machinery Directive 2006/42/EC, and must, therefore, not be used for safety-relevant applications. This device is NOT intended for the purpose of personal safety, and must, therefore, not be used to protect persons.

2.3 Warnings used in this document

 DANGER	High risk Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.
 WARNING	Medium risk Indicates a potential hazardous situation which, if not avoided, could result in death or serious injury.
 CAUTION	Low risk Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.
NOTICE	Material damage Indicates a situation which, if not avoided, could result in malfunctions or material damage.

3 Product description

The multi-monitor controller JCM-631 lets you connect 2 displays via FPD-LinkII. You can connect up to 8 cameras. All common interfaces such as CAN, Ethernet, RS-232 and USB, as well as digital inputs and outputs are available.

3.1 Design

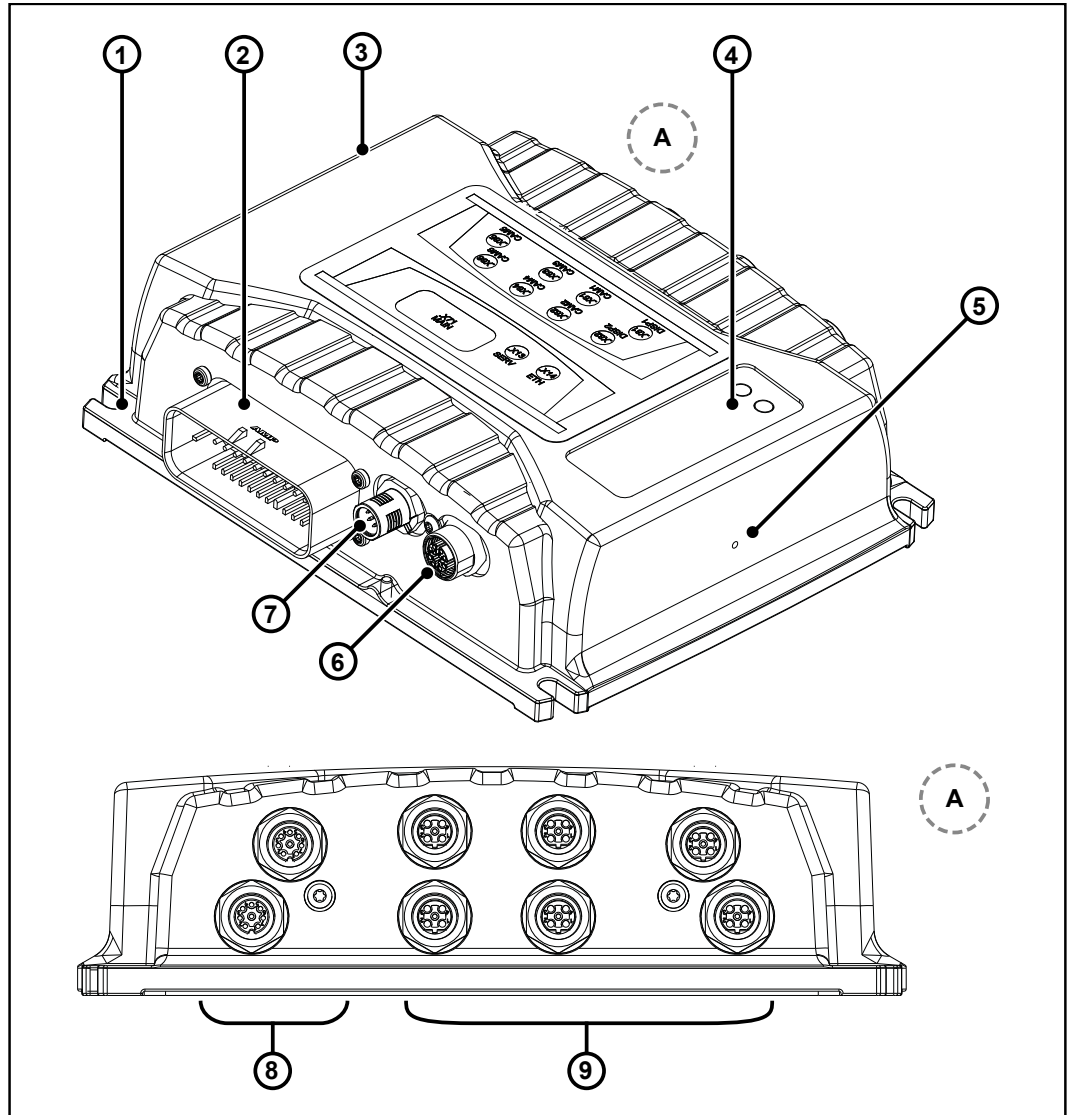


Fig. 1: Design

1	Fastening eye
2	X71: Power supply connector and CAM7 ... CAM8
3	Position of nameplate
4	Diagnostic LEDs
5	Protective vent
6	X14: Service interface 2 (ETH)
7	X13: Service interface 1 (USB + RS-232)
8	X81/X82: Connector for display 1 and display 2
9	X61 ... X66: Connector for CAM1 ... CAM6

3.2 Product features

- iMX6, 800 MHz
- 1 GB RAM
- 1 GB flash memory (available to the user)
- Programming to IEC 61131-3 using STX or in C/C++
- 8 cameras
- 2x FPD-LinkII-OUT
- 3 digital inputs
- 2 digital outputs
- 4x CAN
- USB
- RS-232
- Ethernet

3.3 LED indicators

The JCM-631 has 2 LEDs. LED D2 can be configured by the user.

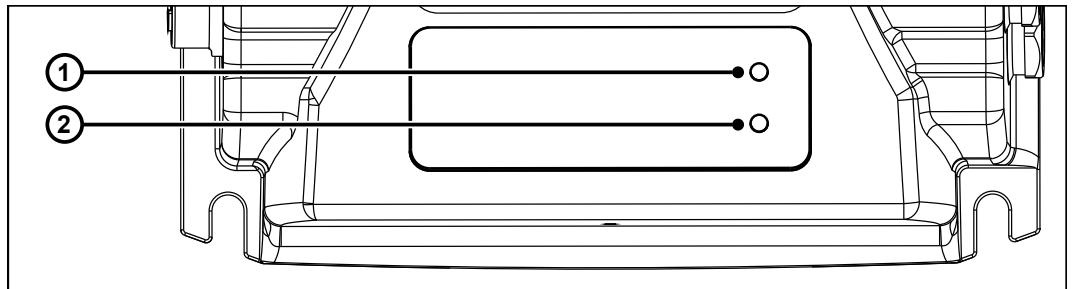


Fig. 2: LED indicators

1	"D2" LED
2	"D1" LED

3.3.1 Diagnostic capability via LEDs

The color and status of the LEDs provide diagnostic options for various states. In the programming tool JetSym, diagnostics can be performed in the Hardware Manager or via the setup window by entering the corresponding register number.

LEDs	Status	Color	Description
D1	ON	Orange	System error. Contact Jetter AG.
	Flashing slowly	Red	System started, internal communication not yet active
	Flashing slowly	Green	System started, internal communication active
	ON	Red	System error - graphics unit. Contact Jetter AG.
	Flashing slowly	Red/ green	Temperature warning

LEDs	Status	Color	Description
D2	OFF	---	This LED is controlled by register 700000063. If the register has the value 0, the LED is off.
	ON	Red	Register 700000063 has the value 1.
	ON	Green	Register 700000063 has the value 2.
	ON	Orange	Register 700000063 has the value 3.

3.4 Nameplate

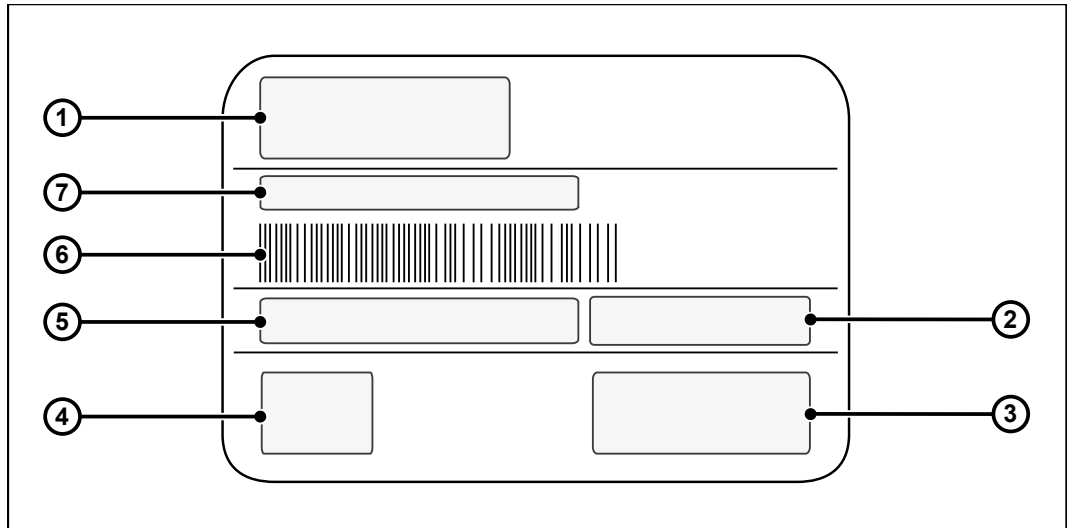


Fig. 3: Sample nameplate

1	Company logo
2	Item number and hardware revision
3	Certification mark
4	Power supply specifications
5	Model code number
6	Bar code
7	Serial number

3.5 Scope of delivery

Scope of delivery	Item number	Quantity
JCM-631	10001553	1

4 Technical specifications

This chapter contains information on electrical and mechanical data, as well as on operating data of the JCM-631.

4.1 Dimensions

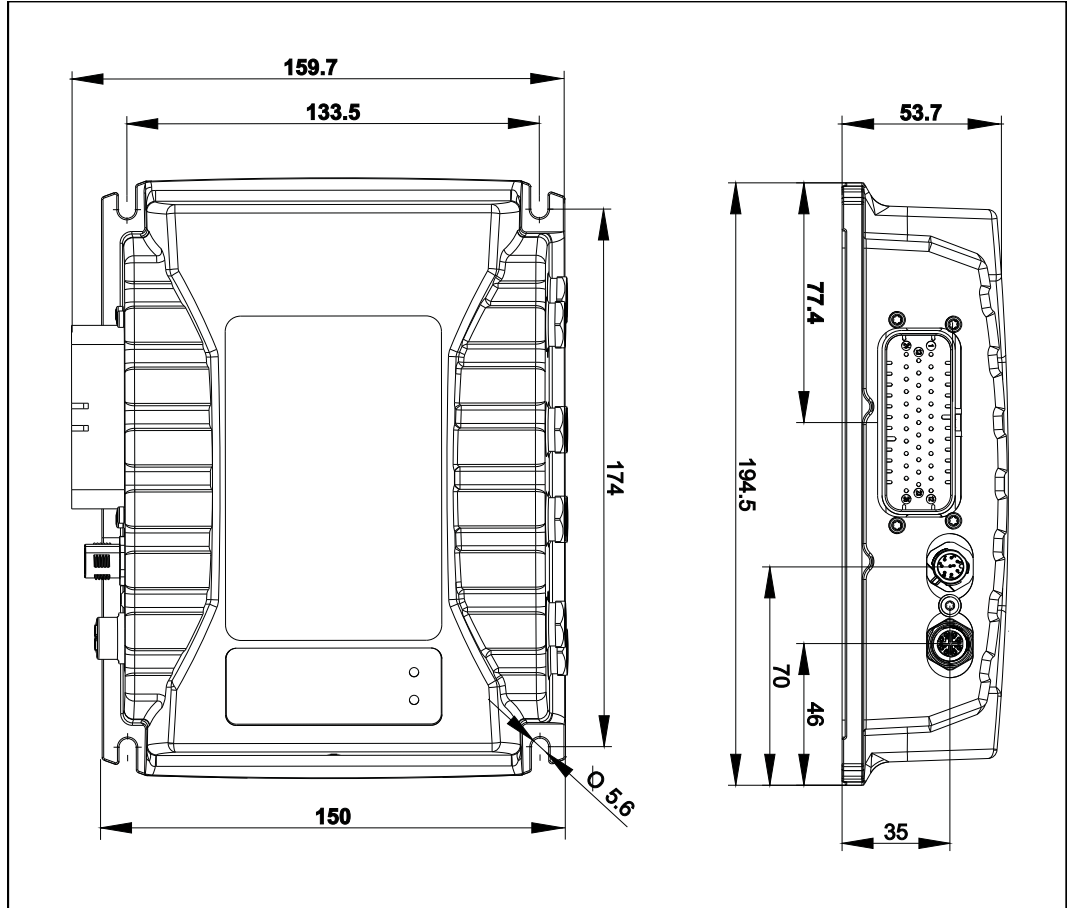


Fig. 4: Dimensions in mm

4.2 Mechanical specifications

Parameter	Description	Standards
Weight	1250 g	
Housing specifications		
Material	Die-cast aluminum housing	
Housing potential	Galvanically isolated	
Housing film	Film in the logo area with viewing window for diagnostic LEDs	
Coating	None. Option: Coating for category C4	

Parameter	Description	Standards
Screw connections	All screw connections on the outside of the housing should be resistant to salt spray.	
Vibration resistance		
Floating frequency (10 to 150 Hz)	6 h	DIN EN 60068-2-6
Shock resistance		
Type of shock	Half-sine wave	DIN EN 60068-2-27
Intensity and duration	30 g for 11 ms	
Number and direction	6000 shocks (1000 shocks in 6 directions)	
Testing temperature:	Room temperature	
Degree of protection		
Degree of protection	IP65	DIN EN 60529
Conditions for IP67	The mating connectors including the wire glands are also IP67 rated. The prescribed mounting orientation must be observed.	

Tab. 1: Mechanical specifications

4.3 Electrical properties

Controller

Parameter	Description	Standards
Operating voltage	8 ... 32 V DC	
	12 V and 14 V from vehicle electrical system	
External fusing	8 A	
	No separate operating voltage supplies for the outputs; housing and GND are routed separately	
Total current consumption of unit at 24 V	Standby current via terminal 30 < 3 mA	
Load dump protection	Up to 70 V	ISO 7637-2
Protection against polarity reversal	Yes, up to 32 V permanently	
Voltage measurement for diagnostics	12-bit resolution	

Tab. 2: Electrical Characteristics – Controller

4.3.1 CPU core and operating system

Parameter	Description
CPU	
Type	CORTEX A9 Dualcore
Switching frequency	1 GHz per core
Flash memory capacity	4 GB, 1 GB of which is available for the user
RAM	1 GB, DDR3
Coprocessor	
Type	CORTEX M4
Switching frequency	120 MHz
Flash memory capacity	512 kB
RAM	128 kB
FPGA	
RAM	128 MB, DDR2
Operating system	
Operating system	Windows Compact 2013 + NET Framework
Boot loader	Coprocessor CAN bootloader (via CANopen®)
Firmware	Coprocessor firmware (CAN interface and video controller)

Tab. 3: CPU core and operating system

4.3.2 Ports and interfaces

CAN port

Parameter	Description
Adjustable Baud rates	125 kBaud
	250 kBaud
	500 kBaud
	1 MBaud
Default Baud rate	250 kBaud
Protocol	CANopen®
	SAE J1939
Default node ID on the CANopen® bus	127 (0x7F)
Terminating resistor	Does not exist. Must be connected externally.
Number of CAN interfaces	4

Tab. 4: CAN port

USB port

Parameter	Description
Data rate	USB 2.0
High-speed type	480 MBaud max.
Protocol	USB host
Connection topology	Connection via service connector
Maximum cable length	0.3 m

Tab. 5: USB port

Ethernet interface

Parameter	Description
Baud rate	100 Mbit/s
Protocols	All typical LAN protocols
Connection topology	Connection via M12 connector
Default Ethernet IP	192.168.10.15

Tab. 6: Ethernet interface

Serial interface RS-232

Parameter	Description
Adjustable Baud rates	9.6 kBaud
	19.2 kBaud
	38.4 kBaud
	57.6 kBaud
	115.2 kBaud
Protocols	All typical COM protocols, printer
Connection topology	1x connection via main connector
	1x connection via service connector M12

Tab. 7: Serial interface RS-232

Video interfaces

FPD Link II

Parameter	Description
Protocols	FPD Link II
Color depth	18-bit (determined by FPD-Link II)
Supported resolutions	480x272 ... 1280x800
Bandwidth	1032 Mbps
Connection topology	Connection is doubled. The displays are supplied with U_b (24 V) max. 500 mA (PTC protected).

Tab. 8: FPD Link II

CAM IN

Parameter	Description
Signal type	Analog CVBS, 1 Vpp
Protocols	PAL, NTSC
Connection topology	Connection is implemented once. The cameras are supplied by CAMERA POWER IN (looped through).

Tab. 9: CAM IN

4.3.3 Inputs and outputs

Digital outputs

Parameter	Description
Quantity	2
Function	Digital output signal
Design	H-side switch
Rated voltage	DC 8 V ... DC 32 V(UB)
Output current	Max. 500 mA per output, short-circuit proof
Protective features	Thermal tripping (hardware)
	Reverse polarity protection over entire device
	Short-circuit-proof
Electrical isolation	None

Tab. 10: Digital outputs

CAMERA POWER OUT

Parameter	Description
Quantity	1
Power consumption	500 mA per camera

Tab. 11: CAMERA POWER OUT

Digital inputs

Parameter	Description
Quantity	3
Function	Digital input signal
Design	PNP
Operating point ON:	7.5 V min.
Operating point OFF:	5.0 V max.
Input resistance	3.30 kΩ
Frequency band	0 ... 100 Hz (static)
Electrical isolation	None

Tab. 12: Digital inputs

4.4 Environmental conditions

Parameter	Description	Standards
Operating temperature	-30 ... +75 °C In case of direct mounting of the rear wall on a heat dissipating unpainted metal plate (aluminum, anodized aluminum, V2A, V4A) with expected thickness of 5 mm	DIN EN 61131-2 DIN EN 60068-2-1 DIN EN 60068-2-2
Max. housing temperature	80 °C	
Climatic conditions	Humid heat	
Storage temperature	-40 ... +85 °C	
Air humidity	10 ... 95 %	
Salt water resistance	Not designed for maritime applications	
Degree of pollution - Electronics	Degree of pollution 2	DIN EN 61131-2

Tab. 13: Environmental conditions

Climatic tests

Humidity/heat, cyclic

Parameter	Description
Temperature cycling range	+25 °C ... +55 °C
Rel. humidity	93 % ± 3%
Duration	24 h
Cycles	6
Functional state	During the switch-on phases and randomly, functional state A

Tab. 14: Humidity/heat, cyclic

Cold constant temperature

Parameter	Description
Temperature cycling range	-30 °C
Duration	24 h
Cycles	6
Functional state	During the switch-on phases and randomly, functional state A

Tab. 15: Cold constant temperature

Warm constant temperature

Parameter	Description
Temperature cycling range	+70 °C
Duration	96 h
Testing method	DUT on, 10x off/on
Visual monitoring	On power-up

Parameter	Description
Functional state	Functional state A

Tab. 16: Warm constant temperature

Combined vibration and temperature test

Parameter	Description	Standards
Frequency band	10 Hz ... 2000 Hz	
Power spectral density (PSD)	10 Hz, 2 * 0.1 (g ² /Hz)	
	20 Hz, 2 * 0.1 (g ² /Hz)	
	100 Hz, 1 * 0,001 (g ² /Hz)	
	500 Hz, 1 * 0,001 (g ² /Hz)	
	2000 Hz, 1 * 0.0001 (g ² /Hz)	
RMS value of acceleration	2.2 g	
Test direction	X, Y, Z direction	
Test duration	22 h per room axis	
Total test duration	66 h mere vibration time	
Temperature profile	See thermal cycling test	DIN EN 60068-2-14

Tab. 17: Combined vibration and temperature test

Thermal cycling test

Parameter	Description
Testing temperature	-20 °C ... +70 °C
Dwell times	see temperature curve
dT/dt	1 K/min
Temperature profile	Start at +20 °C
	Change to -20 °C after 40 min.
	90 min at -20 °C
	Change to +20 °C after 40 min.
	Change to +70 °C after 50 min.
	100 min at +70 °C
Status of DUT	Change to +20 °C after 50 min.
	In operation

Tab. 18: Thermal cycling test

4.5 EMC values

EMV to EN 13309

Emitted interference

Parameter	Description	Standards
Frequency band		Directive 2004/104/EC, supplemented by Directives 2005/49/EC, 2005/83/EC, 2006/28/EC and 2009/19/EC with 100 V/m (E1 type-approval) and EN 61000-6-2:2001 (CE)
Limit value		

Tab. 19: Emitted interference

Interference immunity to external magnetic field

Parameter	Description	Standards
Frequency		Directive 2004/104/EC, supplemented by Directives 2005/49/EC, 2005/83/EC, 2006/28/EC and 2009/19/EC with 100 V/m (E1 type-approval) and EN 61000-6-2:2001 (CE)
Magnetic field		

Tab. 20: Interference immunity to external magnetic field

5 Mechanical installation

WARNING

Risk of burns

Contact with hot surface may cause burns.

- ▶ Take protective measures to prevent inadvertent contact with the device.
- ▶ Allow the device to cool down for some time before you start working on it.

NOTICE

Damages to material or functional impairment

Welding on the chassis may cause damages to material of the device, or impair its functions.

- ▶ Before you start welding, disconnect all connections between the device and the electric system of the vehicle.
- ▶ Protect the device from flying sparks and welding beads (splatter).
- ▶ Do not touch the device with the welding electrode or earth clamp.

NOTICE

Dirt and moisture can affect the electrical connections.

- ▶ Plug unused pins with blanking plugs.
- ▶ Protect all electrical connections with appropriate single wire seals.

5.1 Requirements for installation location and mounting surface

NOTICE

Overheating

Heat build-up can impair the function of the device

- ▶ Mount the device on a heat-conductive surface.
- ▶ Do not mount the device in the vicinity of parts which may become very hot.

The device can directly be fastened to the vehicle or to a mounting plate.

Installation location The installation location must meet the following requirements:

- The installation location must allow air to circulate.
- The installation location must be of sufficient size.
- The device must be easily accessible to allow for service work. It must be possible to disconnect the connectors at any time.
- Sufficient distance to parts with high heat generation must be maintained.

Mounting surface The mounting surface must meet the following requirements:

- Heat dissipating materials, e.g. galvanized sheet steel or anodized aluminum must be used.
- The installation surface must be level.
- The mounting surface must not be painted.

5.2 Preparing for installation

Fastening material Use the following fastening material:

Material	Size	Surface	Strength class
Screws/bolts	M5	galvanized	8.8
Washers	M5	galvanized	-
Screw nuts	M5	galvanized	8.8

Tab. 21: Fastening material

Mounting surface Prepare the mounting surface as follows:

1. Mark out the positions of the 4 fastening lugs.
2. Center-punch the 4 holes.

5.3 Mounting orientation

When mounting, observe the permitted and prohibited mounting orientations.

5.3.1 Allowed mounting orientations

When installing the controller, make sure that the connector plugs point to the side.

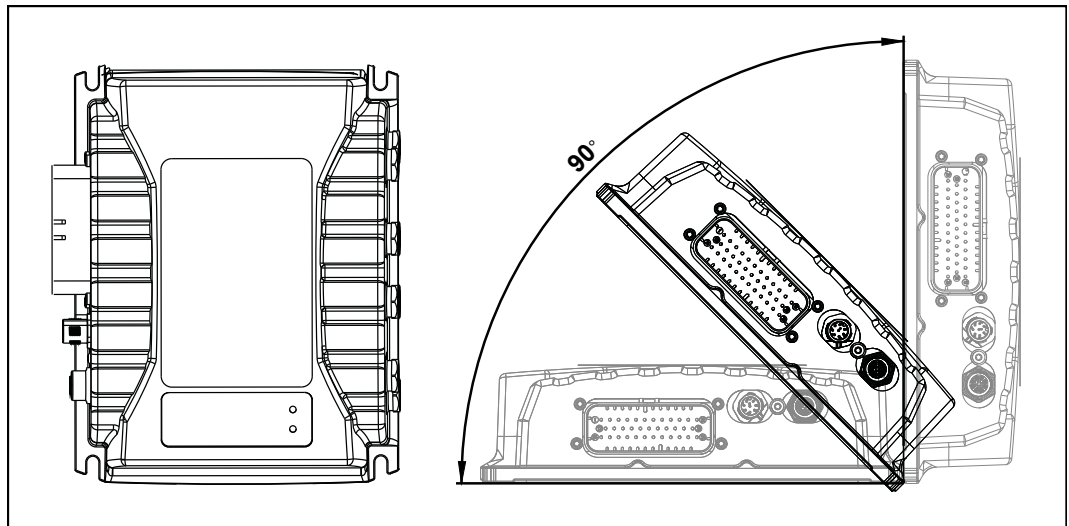


Fig. 5: Allowed mounting orientations

5.3.2 Prohibited mounting orientations

Mounting overhead or with the connection plugs pointing upwards are not allowed.

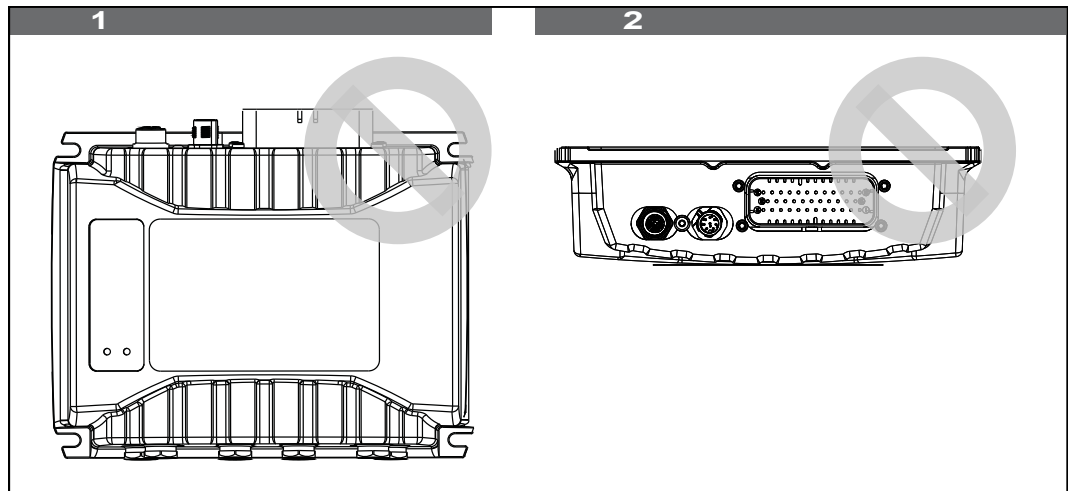


Fig. 6: Prohibited mounting orientations

6 Electrical connection

WARNING

Transient overvoltages

Damage to the device, in extreme cases consequential damage to persons due to malfunction of the device.

- ▶ For lines longer than 30 m, install separate overvoltage protection for each line.

WARNING

Risk of burns

Contact with hot surface may cause burns.

- ▶ Take protective measures to prevent inadvertent contact with the device.
- ▶ Allow the device to cool down for some time before you start working on it.

WARNING

CAN line - Interferences

Faulty communication due to unshielded CAN lines resulting in extreme cases in personal injury due to malfunction of the device.

- ▶ Use shielded connecting cables to CAN interfaces.
- ▶ Connect termination resistors of 120 ohms at both ends of the CAN bus.

NOTICE

Improving electromagnetic compatibility

Improper implementation of the wiring harness may impair electromagnetic compatibility.

- ▶ Keep the cables as short as possible.
- ▶ Lay power lines and signal lines separated from each other.

NOTICE

Damages to material or functional impairment

Improper implementation of the wiring harness may cause mechanical stress.

- ▶ Protect the cables from bending, twisting or chafing.
- ▶ Install strain reliefs for the connecting cables.

NOTICE

Surges resulting from missing protection or fusing

Surges may cause malfunctions or damage to the product.

- ▶ Protect the voltage inputs from surges according to the requirements.
- ▶ Ensure that the device is handled in accordance with ESD regulations.

6.1 Pin assignment

6.1.1 X71 - Connector (male)

Purpose

Lets you connect the following:

- Power supply
- CAN 1 ... 4
- Camera 7 ... 8
- DO1 ... DO3
- DI1 ... DI3
- Serial interface RS-232

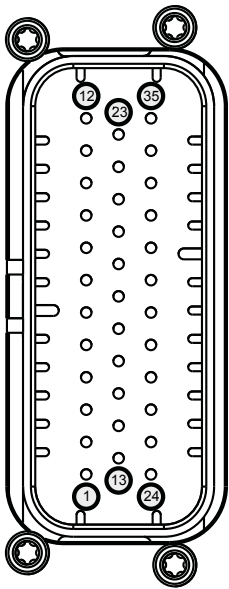


Fig. 7: Connector (male), 35-pin, AMPSEAL

Pin	I/O	Signal	Description
1	I	U _{ign}	Ignition (vehicle terminal 15)
2	I	GND (Ground)	Battery - (vehicle terminal 31)
3	I	U _{bat}	Battery + (vehicle terminal 30)
4	I/O	CAN3_L	CAN3-Low
5	I/O	CAN2_L	CAN2 Low
6	I/O	CAN1_L	CAN1 Low
7	I/O	CAN4_L	CAN4 Low
8	I/O	CAN4_H	CAN4 High
9	O	GND (Ground)	Reference ground RS-232 (COM1)
10	O	GND (Ground)	GND supply for camera 8
11	O	DO1	
12	O	DO2	
13		n. c.	
14	O		Video input for camera 7
15	I		Video input + for camera 8
16	I/O	CAN3_H	CAN3 High
17	I/O	CAN2_H	CAN2 High
18	I/O	CAN1_H	CAN1 High
19	O	GND (Ground)	
20	O	GND (Ground)	
21	O	GND (Ground)	GND supply for camera 7
22	O	GND (Ground)	
23	-		Shield
24		n. c.	
25	I		Video input + for camera 7
26	O		Video input for camera 8
27	O		RS-232 TX (COM1)
28	I		RS-232 RX (COM1)
29	I	DI1	
30	O	DO3	Output power of the camera up to 3 A.
31	I	DC 12 V/24 V	Power consumption of the camera
32	I	DI2	
33	I	DI3	

Pin	I/O	Signal	Description
34	-		Shield
35	-		Shield

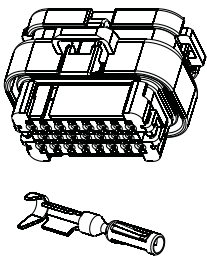
Cable for the male connector

The connecting cable must meet the following requirements:

Parameter	Description
Core cross-section	0.75 mm ² ... 1.5 mm ² (AWG 20 ... 16)
Minimum diameter of the isolation	1.7 mm
Maximum diameter of the insulation	2.7 mm

Mating part of the connector

Compatible mating part of the 35-pin AMPSEAL connector:



Parameter	Description
Manufacturer	AMPSEAL
Item no. of housing	776164-4
Item no. of crimp contact (jack)	0-0770520-1
Wire size range	0.75 mm ² ... 1.5 mm ² (AWG 20 ... 16)

6.1.2 X13 - Service connector 1 (USB + RS-232)

Purpose

Lets you connect the following:

- USB
- Serial interface RS-232

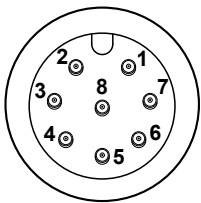


Fig. 8: M12, male, 8-pin, A-coded

Pin	Signal
1	USB VCC5 + IN/OUT
2	USB DATA -
3	USB DATA +
4	n. c.
5	USB shield -
6	RS-232 RX (COM2)
7	RS-232 TX (COM2)
8	GND (Ground)

6.1.3 X14 – Service connector 2 (Ethernet)

Purpose

Lets you connect the following:

- Ethernet interface

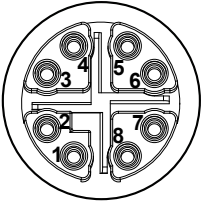


Fig. 9: M12, female, 8-pin , X-coded

Pin	Signal
1	D1+
2	D1-
3	D2+
4	D2-
5	D4+
6	D4-
7	D3-
8	D3+

Cable for service connector 1

To connect devices to service connector 1, you can order cables separately as [accessories \[▶ 77\]](#).

6.1.4 X61 ... X66 – CAM1 ... CAM6

Purpose

This connector (female) lets you connect the following:

- Camera 1 ... 6

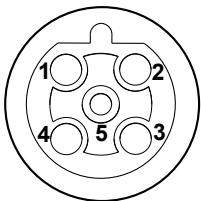


Fig. 10: M12, female, 5-pin B-coded

Pin	Signal	Description
1	Video -	
2	GND (Ground)	
3	Video +	
4	12 V / 24 V	Power supply 12 V / (24 V via CAM Power IN)
5	GND (Ground)	

6.1.5 X81, X82 – Display 1 and Display 2

Purpose

This connector (female) lets you connect the following:

- FPD-Link II for connecting JVM-Mxx displays

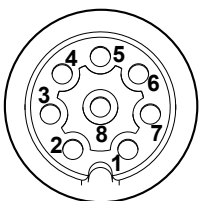


Fig. 11: M12, female, 8-pin , A-coded

Pin	Signal	
1	24 V supply	
2	GND (Ground)	
3	GND (Ground)	
4	FPD-LINK II	
5	CAN1_L	Must be provided with a suitable terminating resistor.
6	FPD-LINK II	
7	24 V supply	
8	CAN1_H	Must be provided with a suitable terminating resistor.

6.2 Commissioning

This chapter outlines the steps for initial commissioning of this device.

DANGER

Uncontrolled axis movements

High mechanical forces due to axis movements and accelerations.

- ▶ Keep out of the danger zone of the machine.
- ▶ Do not disable any safety equipment.
- ▶ Have malfunctions eliminated by qualified personnel.

NOTICE

Surges resulting from missing protection or fusing

Surges may cause malfunctions or damage to the product.

- ▶ Protect the voltage inputs from surges according to the requirements.
- ▶ Ensure that the device is handled in accordance with ESD regulations.

6.2.1 Connecting the voltage supply

Follow the steps below to connect the power supply:

- ✓ Make sure that the enabling relay is connected and jumpered.
- 1. Connect the following pins:
 - Power supply
 - Terminating resistor of 120 Ω at the beginning and end of the CAN bus
- 2. Switch on the ignition.
- ⇒ The controller is supplied with power.

6.2.2 Booting up the controller

- ✓ You need the cable with the designation Programming Cable **2m M12/8-pin/ X-coded to RJ45 (part no. 60880135)**
- 1. Connect the cable to the **service connector 2**.
- 2. Connect controller and PC by means of the Ethernet cable.
- 3. Power-up the controller.
 - ⇒ The controller goes through the boot process.
 - ⇒ LED **D1** flashes green when the controller is ready for operation.
- 4. Establish a network connection between your PC and the controller. The default IP address of the controller is **192.168.10.15**.
- 5. Launch the JetSym programming tool.
- ⇒ Now, you can start configuring the controller.

INFO

Further information

For more information on this topic refer to the Online Help in our software products or to the application-oriented manuals on our [homepage](#).

6.2.3 Setting up an Ethernet connection between controller and PC

Before you can program a Jetter controller, you must first establish an Ethernet connection between your PC and the controller.

- ✓ You need the appropriate Ethernet cable (part number: 60880135)
 - ✓ The controller and PC are switched on and connected with an Ethernet cable.
 - 1. Set up a new network connection in the Windows® operating system of your PC. How you set up a network connection depends on the version of the Windows® operating system.
 - 2. Configure your PC's network connection to connect to the controller. It is important that the IP addresses of the controller and the network connection match in the first three IP address elements (default IP address: 192.168.10.15).
- ⇒ The network connection is established and you can program the controller.

7 Identification and Configuration

7.1 Identification

This chapter describes how to identify the JCM-631:

- Determining the hardware revision
- Retrieving Electronic Data Sheet (EDS) information. The EDS holds numerous non-volatile production-relevant data.
- Determining the OS version of the device and its software components

7.1.1 Device web site

The device has a web interface with a graphical user interface. You can access the Web Interface using a Web browser and the IP address of the device. The user interface consists of several pages for input and output of values and offers various functions, such as operating system update.

Info

The *Info* page provides information about the product and the embedded software.

Product	
Name	JCM-631-E02-G06-K00-O03
Series	JCM-63x
Serial Number	20180425090000
HW Revision	03.26
SOM Name	PCBY_SMARC_TES_01_MAGIK2_IMX6
SOM Serial Number	17W34D0108
Baseboard Name	PCBY_JCM630_01_BASE_VAR1
Baseboard Serial Number	20171221700139

Software Version	
System Version	1.0.1.7
Kernel Version	1.3.3.13
Coprocessor Version	0.9.0.2
FPGA Version	1.0.1.31
JDS Version	1.00.1.19

Fig. 12: "Info" page

Systemlog

The *Systemlog* page provides a list of system events that have occurred since the device was started.



Fig. 13: "Systemlog" page

I/O Info

The *I/O Info* page provides a list of device-specific registers. The current value of the registers is also displayed.

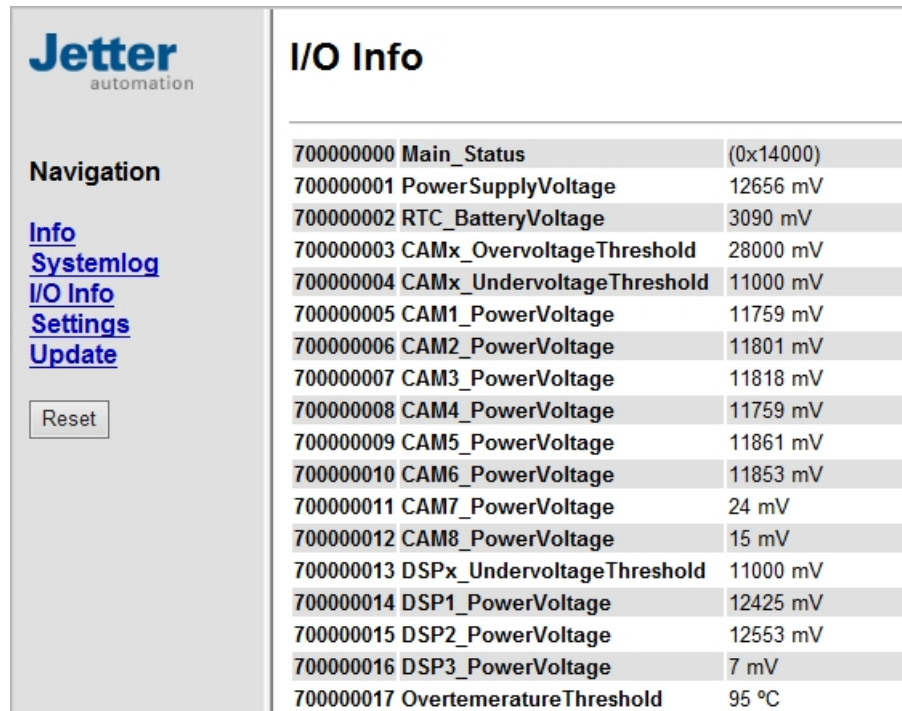


Fig. 14: "I/O Info" page

Settings

The *Settings* page lets you configure the controller, e.g. its IP address or baud rates.

INFO

CAN Settings

Please note that the CAN settings on the device web page are taken over by the STX program.

Fig. 15: "Settings" page

To change values on the *Settings* page, proceed as follows:

1. Click on the **Edit** button to edit the respective value.
 2. Enter the new value or select a new value.
 3. Click the button **Save**.
 4. Click the **Reset** button in the navigation area.
- ⇒ The device restarts and the values are applied.

7.1.2 Electronic Data Sheet (EDS)

Each JCM-631 has an electronic nameplate EDS. Numerous production-relevant data are permanently stored in the EDS. The EDS data can be read out via files in the file system of the controller or via special registers.

EDS registers

EDS registers let you retrieve entries made in the Electronic Data Sheet (EDS). These registers mirror 1:1 the contents of the EDS file. They are to be read only (ro, read only).

Registers - Overview

An overview of the EDS registers can be found in the chapter Register Overview.

7.1.3 Version registers

The operating system provides several registers which can be used to read out the hardware revision or OS version of the device and its components. You will need this information when contacting the hotline of Jetter AG in case of a problem.

Hardware revisions

The device has special registers, the content of which lets you identify the hardware.

Registers - Overview

The registers listed below let you retrieve the hardware revisions:

Register	Description
108021	Hardware revision - CPU board
200170	Controller model

Tab. 22: Register overview - Hardware revisions

Operating system version

The device has special registers, the content of which are unique OS version numbers.

Format of software version numbers

The software version numbers of the JCM-631 are represented by 4 numbers.

1 . 2 . 3 . 4

Fig. 16: Software version numbers

Digits	Description
1	Major or main version number
2	Minor or secondary version number
3	Branch or intermediate version number
4	Build version number

Tab. 23: Format of software version numbers

Released version A released version can be recognized by both Branch and Build having got value 0.

Registers - Overview

The registers listed below let you retrieve the operating system versions:

Register	Description
200169	Operating system version
210001	JetVM version

Tab. 24: Registers - Overview

7.2 Operating system

We are continuously striving to enhance the operating systems of our products. Enhancing means adding new features, and upgrading existing functions. Current OS files are available for download on our homepage in the downloads area of the respective product.

i INFO

Further information

More information on this subject is available on our website.

[Start | Jetter - We automate your success.](#)

7.2.1 Operating system update of the controller

This chapter describes how to carry out an operating system update on the JCM-631 controller. You have got several options for transferring the OS file to the controller:

- With the programming tool JetSym
- Via FTP connection
- From a USB flash drive
- From the application program
- Via device website
- Via JetEasyDownload

Operating system update using the programming tool

The JetSym programming tool offers an easy way to transfer an OS file to the controller.

Performing the update

- ✓ An OS file for the controller must be available.
 - ✓ An Ethernet connection or a CAN connection via the first CAN interface has been set up via the hardware manager in JetSym .
1. Select in the JetSym menu **Build** menu item **Update OS...** Alternative: In the **System commands** dialog of the Hardware Manager, click the button **Update OS**.
 - ⇒ The file selection dialog opens.
 2. Select the new OS file here.
 - ⇒ JetSym opens a confirmation dialog.
 3. Launch the OS upload by clicking the button **Yes**.
 4. Wait until the update process is completed.
 5. To activate the transferred OS, re-boot the controller.

Updating the operating system by means of FTP

An FTP client lets you transfer an OS file to the controller.

Performing the update

- ✓ An OS file for the controller must be available.
 - ✓ An FTP connection to the controller must be possible.
 - ✓ The login parameters for a user with administrator or system rights must be at hand.
 - ✓ The OS is running.
 - ✓ Make sure the controller remains energized during the operating system update.
1. Open an FTP connection to the JCM-631.
 2. Log in with administrator or system rights
User: *admin*
Password: *admin*
 3. Navigate to the directory */System/OS*.
 4. Transfer the OS file
 5. Wait until the update process is completed
 6. Clear the FTP connection.
 7. To activate the transferred OS, re-boot the controller.

OS update from a USB flash drive

A USB flash drive lets you transfer an OS file to the JCM-631.

Contents of the batch file

The batch file **autostart.bat** must have the following contents:

```
@echo off
if not exist \App\autostart.ini goto end
if exist \App\update.ini goto end
if exist \App\usb_update_process_marker goto dm
echo \Data\update\jvm_ce0.os >\App\update.ini
if not exist \Data\update md \Data\update
copy %1\jvm_ce0.os \Data\update\jvm_ce0.os
echo.>\App\usb_update_process_marker
del \System\reset.exe
goto end
:dm
del \App\usb_update_process_marker
:end
```

Updating the operating system

- ✓ An OS file for the JCM-631 must available.
 - ✓ The device must remain powered on during the operating system update.
1. Copy the OS file to your USB flash drive. This drive must be FAT formatted.
 2. Create a batch file on the USB flash drive and name it **autostart.bat**. Contents of the batch file: See above.
 3. Replace all entries **jvm_ce0** with the actual name of the OS file.
 4. Switch off the JCM-631.
 5. Insert the USB flash drive into the JCM-631.

- 6. Switch on the JCM-631.
 - ⇒ About 10 seconds later the device automatically launches the update process.
 - ⇒ The device reboots automatically and starts the application.

Update sequence

The operating system update is performed in the following chronological order:

Level	Description
1	The files are transferred.
2	The device is reset.
3	The files are installed.
4	The device is reset.
5	The application appears

Log file for documentation purposes

The update process is documented in the **\App\update.log** file.

Updating the OS from within the application program

The file functions included in the STX language let you carry out a program-controlled OS update of a JCM-631 from within an OS file.

Performing the update

- ✓ An operating system file must be located in the file system of the JCM-631.
 - ✓ The operating system and the application program must be running.
 - ✓ Make sure the controller remains energized during the operating system update.
1. Copy the OS file to a file of any name and of the extension ***.os** in the directory */System/OS*.
 2. To activate the updated OS, for example by writing to the system command register, re-boot the controller.

STX program

```

Var
    SourceName:      String;
    DestinationName: String;
    UpdateIt:        Bool;
End_Var;

//*****
// Name: OSupdate
// 1. Activating Tracing in JetSym
// 2. Setting the name of the source file in 'SourceName'
// 3. Setting the flag 'UpdateIt'
//*****

Task OSupdate Autorun
    Var
        ResCopy: Int;
    End_Var;

    DestinationName := '/System/OS/OperatingSystem.os';
    Loop
        UpdateIt := False;
        When UpdateIt Continue;
        ResCopy := FileCopy(SourceName,
                            DestinationName);
        Trace('Result : ' + IntToStr(ResCopy) + '$n');
    End_Loop;
End_Task;
    
```

OS update via device website

The device web site in Internet Explorer shows all properties and version numbers of your device. It also lets you configure the Ethernet and CANopen® interfaces.

Accessing the device web site

To have the device web site displayed on your PC, enter the **current** IP address of the device in the Internet Explorer.

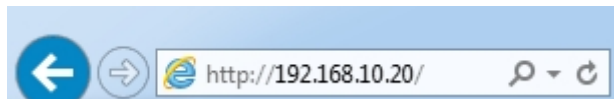


Fig. 17: Accessing the device web site

Prerequisites

To enter the IP address in the range of exceptions of the Internet Explorer, proceed as follows:

1. Click menu item **Tools** and open the dialog **Internet options**.
2. Go to tab **Connections** and click the button **LAN settings**.
3. In section **Proxy Server** click the button **Advanced**.
 - ⇒ The **Proxy Settings** dialog opens where you can enter proxy parameters.
4. In the **Exceptions** pane of the dialog **Proxy Settings**, enter IP address **192.168.***, as is shown in the illustration below.

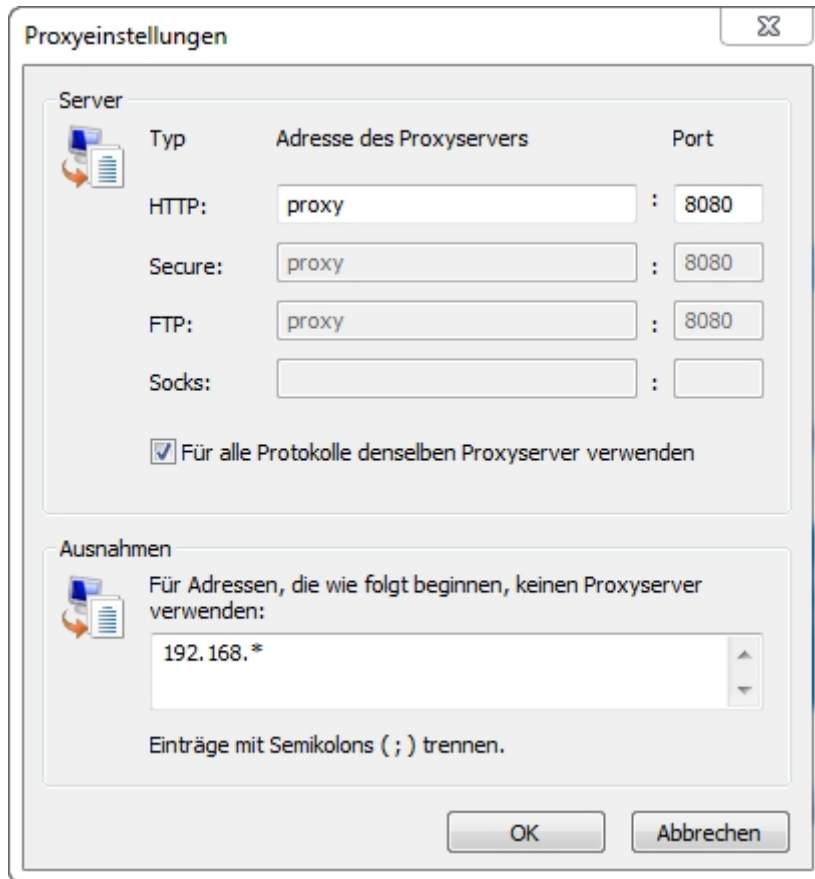


Fig. 18: Device website - Proxy settings

Performing the update

1. On the **Update** page, select an OS file (*Browse...*).
2. Copy the OS file to the device (*Upload*).
⇒ A message appears when the upload is complete.
3. Reboot the controller.

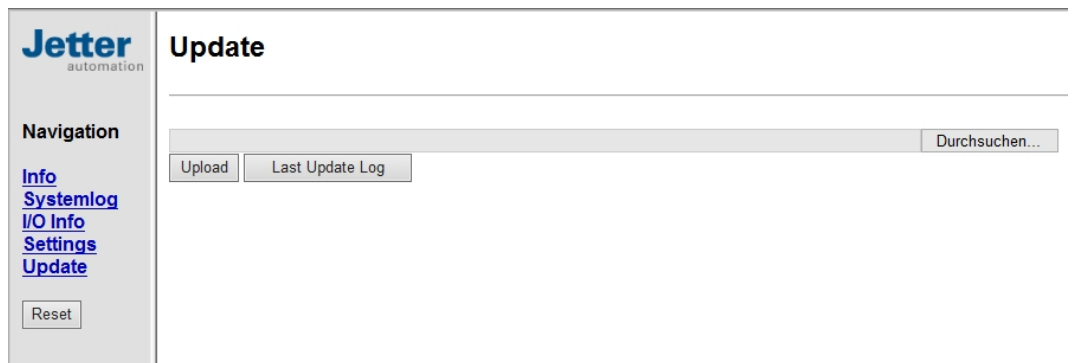


Fig. 19: Device website - Update

i INFO

Displaying the log of the last update

If you click the **Last Update Log** button, the history of the last update is displayed. This will help you track any errors that may have occurred.

Operating system update via JetEasyDownload

You can update the operating system file of the device using a Peak CAN dongle and the command line tool JetEasyDownload (version 1.00.0.15 or higher) by Jetter .

JetEasyDownload parameters

To call JetEasyDownload you need specific parameters.

Parameter	Description	Values	
-H<Num>	Hardware	0=	PCAN_ISA1CH
		1=	PCAN_ISA2CH
		2=	PCAN_PCI_1CH
		3=	PCAN_PCI_2CH
		4=	PCAN_PCC_1CH
		5=	PCAN_PCC_2CH
		6=	PCAN_USB_1CH
		7=	PCAN_USB_2CH
		8=	PCAN_Dongle Pro
		9=	PCAN_Dongle
		10=	PCAN_NET Jetter
		11=	PCAN_DEV default device
		20=	IXXAT V2.18
22=	IXXAT V3		
100=	CAN hardware detected first		
-T<nodeID>	Target NodeID		
-B<Num>	Baud rate	0=	20 kB
	
		7=	1 MB
-S<Num>	SDO Timeout	Default	300 ms
-L<name>	OS file name	e.g. JCM-631_x.xx.x.xx.os	

Performing the update

JetEasyDownload -H100 -T48 -B5 -S8000 -LJCM-631.os

- ✓ JetEasyDownload and Peak CAN dongle must be ready for use.
- ✓ There must be a CAN connection between Peak CAN dongle and JCM-631.
- 1. Call up JetEasyDownload with the above parameters and a valid OS file.
 - ⇒ The device carries out a reset.
 - ⇒ The device starts in boot loader mode with a single heartbeat in init state (data = 0x00).
- 2. Wait for approx. 7 seconds while the device formats the flash memory.
 - ⇒ The device starts the download process.
- ⇒ The device starts automatically with the new firmware.

7.3 File System

The file system lets you access files located on the internal flash disk or an USB flash drive. Some files may be protected against read/write access or deletion. This is normal behavior. Some of these files are virtual files, such as firmware images, or protected files, such as EDS files.

NOTICE

Malfunctions caused by missing or damaged system files

Careless working with system files can result in malfunctions of the device.

- ▶ Do not delete or move any system files.

File categories

The files of the file system are categorized as follows:

- System directories or system files used by the operating system
- Files accessible to the user

System directories

The user is not allowed to delete system directories. System directories even survive formatting.

Directory	Description
/System	<ul style="list-style-type: none"> ■ System configuration ■ System information
/USB	Root directory of the USB flash drive
/App	Directory for applications
/Data	Folder for storing data.
/Windows	
/	RAM disk drive

Tab. 25: System directories

NOTICE

Delays during start-up

Application files in the system directory */Data* cause delays during startup.

- ▶ Copy all application files to the */App* directory and **not to** the */Data* directory.

Formatting and checking of data

The JCM-631 can only format the flash disk. Formatting or checking the USB flash drive is not possible.

INFO

Further information

For more information on this subject refer to the application-oriented manual *File System* available for download from our [homepage](#).

7.3.1 Properties

The following properties apply to the internal flash disk and USB flash drive:

- 8 files max. to be opened simultaneously
- Separate directory names by a slash "/", not by a backslash "\".
- When the controller creates a file, the file contains date and time assigned by the controller.
- Date, time, and file size are not available for all system files.

Flash disk - Properties

Size The following disk space is available to the user:

Parameter	Value
Flash disk capacity	4 GB
Size of folder App (of the a.m. capacity)	450 MB
Size of folder Data (of the a.m. capacity)	2 GB

NOTICE

Limitation of storage cycles

The memory is based on a NAND flash which has a limited number of storage cycles. You should consider this in the type and number of times you store customer data.

- ▶ Save the data only when it is really necessary.
- ▶ NV RAM is available for a high number of storage operations.

Properties The internal flash disk drive has got the following properties:

- Up to 7 directory levels and 1 file level are allowed.
- No case sensitivity.
- Directory and file names with a length of up to 63 characters are possible.
- All characters except "/" and "." are permitted for directory and file names
- The location of the directories "App" and "Data" is on the flash disk drive.

USB flash drive - Properties

Size The available capacity depends on the USB flash drive used. Tested capacity:

- 2 ... 64 GB

Properties The USB flash drive has got the following properties:

- The USB flash drive must be compatible with FAT 12, FAT 16, or FAT 32.
- No case sensitivity.
- Directory and file names with a maximum of 63 characters each are possible.
- The following characters are not permitted in directory and file names: "/", "\", ":", "*", "?", "<", ">" and "|"
- The number of directory levels depends on the formatting.
- There is no user/access administration.

Guarantee Jetter AG only guarantee the proper functioning of USB flash drives offered as options by Jetter AG. You will find suitable USB flash drives in the [accessories](#) [▶ 77](#) section.

7.4 Installing the USB/CAN adapter

i INFO

Driver for USB CAN adapter

The driver for the PEAK Systems USB/CAN adapter is supplied along with the JetSym software.

...C:\Program Files (x86)\Jetter\JetSym x.x.x\Tools\PcanDrv

To install the USB/CAN adapter, proceed as follows:

- ✓ You need an up-to-date version of the software JetSym and a USB/CAN adapter.
 - 1. Plug the USB/CAN adapter into a USB port of your PC.
 - 2. If the hardware installation wizards opens, close it.
 - 3. If you use the USB/CAN adapter by Peak Systems, install the PcanDrv driver.
 - 4. Connect the CAN adapter to the CAN interface of the controller.
- ⇒ You can start configuring your controller in JetSym.

i INFO

Further information

For more information on this topic refer to the Online Help in our software products or to the application-oriented manuals on our [homepage](#).

8 Programming

The JCM-631 is programmed with the integrated development environment JetSym.

System data and system functions are accessed via system registers.

8.1 Abbreviations, module register properties and formats

Abbreviations

The abbreviations used in this document are listed in the table below:

Abbreviation	Description
R 100	Register 100
MR 150	Module register 150

Tab. 26: Abbreviations

Module register properties

Each module register is characterized by certain properties. Most properties are identical for many module registers. In the following description, module register properties are mentioned only if a property deviates from the default properties listed below.

Property	Standard design
Type of access	Read/write
Value after reset	0 or undefined (e.g. revision/version number)
Takes effect	Immediately
Write access	Always
Data type	Integer

Tab. 27: Module register properties

Numerical formats

The numerical formats used in this document are listed in the table below:

Notation	Format of numerical values
100	Decimal
0x100	Hexadecimal
0b100	Binary

Tab. 28: Numerical formats

JetSym sample programs

The notation for sample programs used in this document is listed in the table below:

Notation	Format of numerical values
<code>Var, When, Task</code>	Keyword
<code>BitClear();</code>	Commands
<code>100 0x100 0b100</code>	Constant numerical values
<code>// This is a comment</code>	Comment
<code>// ...</code>	Further program processing

Tab. 29: JetSym sample programs

8.2 Storage options - Overview

The JCM-631 features several types of program and data memory. There is volatile memory that requires power to maintain the stored information, and non-volatile memory which does not require power to maintain the stored information. This memory is located directly on the CPU.

This chapter gives an overview of the available memory.

8.2.1 Operating system memory

The OS is located in the non-volatile flash memory in the CPU. It starts immediately after the device is switched on.

Properties

- Internal flash memory for storing OS data
- Internal volatile RAM for storing OS data

Type of access

- The user is not allowed to directly access the OS memory.
- The operating system can be changed via update.

8.2.2 File system memory

The file system memory is for storing data and program files.

Properties

- Non-volatile
- Size of the internal flash disk: 2,45 GB

Type of access

- By operating system
- By JetSym
- Via FTP connection
- By browser (via HTTP server)
- By means of file commands from within the application program

8.2.3 Application program memory

By default, the application program is uploaded from JetSym to the JCM-631 and is stored there.

Properties

- Stored as file within the file system
- Default directory *\app\program name*
- Files may also be stored to other directories (or USB flash drive)

Type of access

- By operating system
- By JetSym
- Via FTP connection
- By means of file commands from within the application program

8.2.4 Memory for volatile application program variables

Volatile variables are used to store data which need not be maintained when the JCM-631 is de-energized.

- Properties**
- Global variables which are not assigned to permanent addresses (not %VL or %RL)
 - Local variables
 - Variables are stored in a compact manner, according to the size of their type
 - Variables are initialized with value 0 when they are created.

- Type of access**
- By JetSym
 - From within the application program

JetSym STX program In the following program, a global volatile variable is incremented by 1 every 2 seconds.

```

Var
    Count: Int; //volatile, since not localized
End_Var;

Task Inkrement Autorun
    Loop
        Inc (Count) ;
        Delay (T#2s) ;
    End_Loop;
End_Task;
    
```

Setup pane The JetSym setup pane shows the content of the variable:

Number	Description	Function
1	Present content of the variable	The content of the variable is incremented by 1 every 2 seconds

8.2.5 Memory for non-volatile application program registers

Non-volatile registers let you store data which must be saved when the controller is de-energized.

- Properties**
- Global variables with dedicated addresses (%VL)
 - Register variables always occupy 4 bytes.
 - Register variables are not initialized by the operating system.
 - Number of register variables: 30,000
 - Register numbers: 1,000,000 ... 1,029,999

- Type of access**
- By JetSym
 - By browser (via HTTP server)
 - From HMIs
 - From the application program
 - From other controllers

8.2.6 Memory for non-volatile application program variables

Non-volatile variables let you store data which must be maintained when the controller is de-energized.

- Properties**
- Global variables which are assigned to permanent registers (%RL)
 - Variables are stored in a compact way.
 - Size: 120,000 bytes
 - Register numbers: 1,000,000 ... 1,029,999

- Type of access**
- By JetSym
 - From the application program
 - From HMIs

8.2.7 Special registers

Special registers let you control OS functions and retrieve status information.

- Properties**
- Global variables with dedicated addresses (%VL)
 - When the operating system is launched, special registers are initialized using their default values.
 - Register numbers: 100,000 ... 999,999

- Type of access**
- By JetSym
 - By browser (via HTTP server)
 - From HMIs
 - From the application program
 - From other controllers

8.2.8 Inputs and outputs

Inputs and outputs are 1-bit variables. This means they can either have the value TRUE or FALSE.

- Properties of inputs/outputs**
- Variables which are assigned to permanent addresses (%XL)
 - I/O registers: 361000 ... 362200

- Memory access**
- By JetSym
 - From HMIs
 - From the application program

8.2.9 Flag

Flags are one-bit operands. This means they can either have the value TRUE or FALSE.

Properties of user flags

- Global variables which are assigned to permanent addresses (%MX)
- Non-volatile
- Quantity: 256
- Flag numbers: 0 ... 255

Properties of overlaid user flags

- Global variables which are assigned to permanent addresses (%MX)
- Non-volatile
- Overlaid by registers 1000000 ... 1000055
- Quantity: 1,792
- Flag numbers: 256 ... 2047

Properties of special flags

- Global variables which are assigned to permanent addresses (%MX)
- When the operating system is launched, special flags are initialized using their default values.
- Quantity: 256
- Flag numbers: 2048 ... 2303

Memory access

- By JetSym
- From HMIs
- From the application program

8.3 Saving and loading an application program

The user determines the program that is to be executed. When uploading the application program to the controller, this program is stored as a file to the internal flash disk. The device enters the path and file name into the **\App\start.ini** file.

Path and file name By default, JetSym creates in the */app* directory a subdirectory and assigns the project name to it. Then, JetSym stores the application program to this subdirectory assigning the extension ***.es3** to it. Path and file names are always converted into lower case letters.

\App\start.ini - Structure

This file is a text file with one section holding 2 entries:

Element	Description
[Startup]	Section name
Project	Path to the application program file. This path is relative to <i>\ app</i> .
Program	Name of the application program file

Example

```
[Startup]
Project = test_program
Program = test_program.es3
```

The application program is loaded from the file **\App\test_program\test_program.es3**.

Loading the application program

When the application program is restarted via JetSym or after booting the JCM-631, the application program is loaded via the file system and executed.

The application program is loaded by the OS of the JCM-631 as follows:

Step	Description
1	The operating system reads the file \App\start.ini from the internal flash disk.
2	The OS evaluates the Project entry. It contains the path leading to the application program file.
3	The OS evaluates the Program entry. This entry contains the program name.
4	The OS loads the application program from the file <Project>/<Program> .

8.4 CANopen® STX API

The CANopen® STX API provides platform where the user can send and receive CANopen® messages via STX functions.

These STX functions are used in communication between this device and other CANopen® nodes.

The CANopen® standard

CANopen® is an open standard for networking and communication, e.g. in the automotive sector. The CANopen® protocol has been further developed by the CiA e.V. (CAN in Automation) and works on the physical layer with CAN Highspeed in accordance with ISO 11898.

Specifications

The CANopen® specifications can be obtained from the **CiA e.V.** homepage at <http://www.can-cia.org>.

The key specification documents are:

- CiA DS 301 - This document is also known as the communication profile and describes the fundamental services and protocols used under CANopen®.
- CiA DS 302 - Framework for programmable devices (CANopen® Manager, SDO Manager)
- CiA DR 303 - Information on cables and connectors

i INFO

Further information

For more information on this subject refer to the application-oriented manual *CANopen® STX API* available for download from our [homepage](#).

8.4.1 STX Functions

Application

STX functions are used in the communication between the JCM-631 and other CANopen® nodes.

The JCM-631 supports the following STX functions:

Function	Description
CanOpenInit ()	CanOpenInit () lets you initialize one of the CAN busses.
CanOpenSetCommand ()	CanOpenSetCommand () lets you change the heartbeat status of the device itself and of all other devices (NMT slaves) on the CAN bus.
CanOpenUploadSDO ()	CanOpenUploadSDO () lets you access a particular object in the Object Directory of the message recipient and read the value of the object. Data is exchanged in accordance with the SDO upload protocol.
CanOpenDownloadSDO ()	CanOpenDownloadSDO () lets you access a particular object in the Object Directory of the message recipient and specify the value of the object. Data is exchanged in accordance with the SDO download protocol.

Function	Description
CanOpenAddPDORx ()	By calling up the CanOpenAddPDORx () function, process data, sent by other CANopen® devices, can be entered for reception.
CanOpenAddPDOTx ()	By calling up the CanOpenAddPDOTx () function, process data can be placed on the bus.

Tab. 30: Supported STX functions

INFO Further information
 For more information on this subject refer to the application-oriented manual *CANopen® STX API* available for download from our [homepage](#).

INFO Further information
 For more information on this subject, refer to JetSym online help.

8.4.2 Heartbeat monitoring

The heartbeat protocol is for monitoring the activity of communication partners. If the inactivity exceeds the set interval (Heartbeat consumer time), the status is set to "offline".

The application program lets you define heartbeat functions, such as

- Displaying information to the user
- Rebooting the device
- Ignoring process data

INFO Further information
 For more information on this subject refer to the application-oriented manual *CANopen® STX API* available for download from our [homepage](#).

8.4.3 CANopen® Object dictionary

The operating system of the JCM-631 supports the following objects:

Index (hex)	Object (code)	Object name	Data type	Type of access
1000	VAR	Device type	Unsigned32	RO (read only)
1001	VAR	Error Register	Unsigned8	RO
1002	VAR	Manufacturer status	Unsigned32	RO
1003	ARRAY	Pre-defined error field	Unsigned32	RO
1008	VAR	Manufacturer Device Name	String const	1008
1009	VAR	Manufacturer Hardware Version	String const	1009
100A	VAR	Manufacturer Software Version	String const	100A
100B	VAR	Node ID	Unsigned32	RO
1017	VAR	Producer Heartbeat Time	Unsigned16	R/W (read & write)

Index (hex)	Object (code)	Object name	Data type	Type of access
1018	RECORD	Identity	Identity	RO
1200	RECORD	Server 1 - SDO parameter	SDO parameter	RO
1201	RECORD	Server 2 - SDO parameter	SDO parameter	R/W
1203	RECORD	Server 3 - SDO parameter	SDO parameter	R/W
1203	RECORD	Server 4 - SDO parameter	SDO parameter	R/W

i INFO**Further information**

For more information on this subject refer to the application-oriented manual *CANopen® STX API* available for download from our [homepage](#).

8.5 SAE J1939 STX API

SAE J1939 is an open standard for networking and communication in the commercial vehicle sector. The focal point of the application is the networking of the power train and chassis. The J1939 protocol originates from the international **Society of Automotive Engineers (SAE)** and works on the physical layer with CAN high-speed according to ISO 11898.

Specifications

The SAE J1939 specifications can be obtained from the SAE homepage <http://www.sae.org>.

The key SAE J1939 specifications are:

- J1939-11 - Information on the physical layer
- J1939-21 - Information on the data link layer
- J1939-71 - Information on the application layer vehicles
- J1939-73 - Information on the application layer range analysis
- J1939-81 - Network management

i INFO

Further information

For more information on this subject refer to the application-oriented manual *SAE J1939 STX API* available for download from our [homepage](#).

8.5.1 STX Functions

Application

STX functions are used in the communication between the JCM-631 and other devices in a vehicle.

The JCM-631 supports the following STX functions:

Function	Description
SAEJ1939Init()	By calling the SAEJ1939Init() function SAE-J1939Init(), one of the existing CAN busses (not CAN 0 because it is reserved for CANopen®) is initialized for the J1939 protocol.
SAEJ1939SetSA()	The function SAEJ1939SetSA() lets you change your own SA (Source Address) during runtime.
SAEJ1939GetSA()	The function SAEJ1939GetSA() lets you determine your own SA (Source Address).
SAEJ1939AddRx()	Calling up the function SAEJ1939AddRx() prompts the JCM-631 to receive a specific message.
SAEJ1939AddTx()	Calling up the function SAEJ1939AddTx() prompts the JCM-631 to cyclically send a specific message via the bus.
SAEJ1939RequestPGN()	Calling up the function SAEJ1939RequestPGN() sends a request to the DA (Destination Address) following a PGN.
SAEJ1939GetDM1()	Calling up the function SAEJ1939GetDM1() requests the current diagnostics error codes (also see SAE J1939-73 No. 5.7.1).
SAEJ1939GetDM2()	Calling up the function SAEJ1939GetDM2() requests the diagnostics error codes that preceded the current ones (also see SAEJ1939-73 No. 5.7.2).

Function	Description
SAEJ1939 SetSPNConversion()	Calling up the function SAEJ1939SetSPNConversion() defines the arrangement of bytes in the message which is requested by the SAEJ1939GetDM1() or SAEJ1939GetDM2() function.
SAEJ1939 GetSPNConversion()	Calling up the function SAEJ1939GetSPNConversion() determines the currently set conversion method.

i INFO**Further information**

For more information on this subject refer to the application-oriented manual *SAE J1939 STX API* available for download from our [homepage](#).

8.6 Serial interface RS-232

The JCM-631 has a user-programmable RS-232 interface.

8.6.1 Operating principle

The OS provides a receive buffer and a transmit buffer for the user-programmable serial interface. These buffers can be used to adjust the transfer rate between application program and serial interface.

Block diagram

The following illustration shows the block diagram of the user-programmable serial interface:

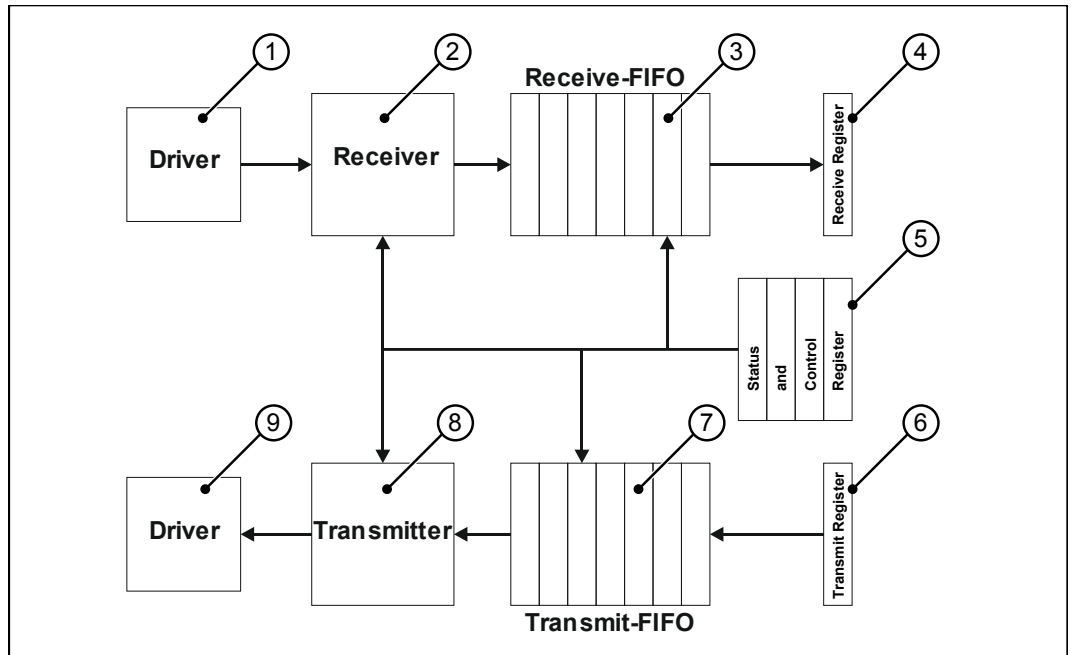


Fig. 20: Block Diagram - Serial Interface

Elements of the interface

The user-programmable serial interface consists of the following elements:

Number	Section	Function
1	Interface driver	Converts the signals of the interface into internal signal levels
2	Addressee	Performs serial/parallel conversion
3	Receive buffer	Buffer for received characters
4	Receive register	Read access to this register reads the received characters in the receive buffer (3).
5	Interface driver	Converts internal signal levels into interface signals.
6	Transmitter	Performs parallel/serial conversion
7	Sending buffer	Buffer for characters to be sent
8	Transmit register	Write access to this register causes the characters to be entered into the transmit buffer (7) and to be sent by the transmitter (6).
9	Status and control register	Query of filling levels and error states of buffers; setting of transmission parameters

Tab. 31: Elements of the serial interface

Receiving a character

Receiving a character is done in 4 steps:

Step	Description
1	The interface driver converts signals "on the line" into internal signal levels and forwards them to the receiver.
2	The receiver performs serial/parallel conversion of this character and checks the set communication parameters.
3	The receiver enters the character into the receive buffer if there is any place left. Otherwise, the character is discarded and buffer overflow is signaled.
4	Via receive register the character can be read out of the receive buffer.

Sending a character

A character is sent in 4 steps:

Step	Description
1	Via transmit register the character is entered into the transmit buffer if there is any place left. Otherwise the character is discarded.
2	Once the transmitter has sent a character, it reads the next character from the transmit buffer.
3	The transmitter performs parallel/serial conversion and sends this character to the interface driver using the set communication parameters.
4	The interface driver converts internal signal levels into the various interface standards

8.6.2 Registers - Description

This chapter describes the registers associated with the user-programmable serial interface. These registers are used for the following tasks:

- Parameterizing the interface
- Sending characters
- Receiving characters

Register numbers

The registers of the interface are combined into one register block. The basic register number of this block is dependent on the device.

Device	Basic register number	Register numbers
JCM-631	103000	103000 ... 103019 for COM 1 on X71
	103100	103100 ... 103119 for COM 2 on the service connector

Tab. 32: Register numbers of the serial interface

Determining register numbers

In this chapter, only the last two figures of a register number are specified. e.g. MR 14. To calculate the complete register number, add the basic register number of the corresponding device, e.g. 103100, to the module register number.

i INFO

Re-initializing the interface

When entering values into registers MR 1 through MR 6, the entire interface is re-initialized and the transmit and receive buffers are cleared.

MR 2

Baud rate

This register lets you set the baud rate.

Property	Description
Values	1,200 ... 115,200
Value after reset	9,600

Tab. 33: Baud rate

MR 3

Number of data bits per character

This register lets you set the number of data bits per character.

Property	Description
Values	5, 6, 7, 8
Value after reset	8

Tab. 34: Number of data bits per character

MR 4

Stop bits

This register lets you set the number of stop bits per character.

Property	Description
Values	1 1 stop bit
	2 1.5 stop bits if MR 3 = 5 2 stop bits if MR 3 = 6, 7, 8
	Value after reset

Tab. 35: Stop bits

MR 5

Parity

This register lets you set the parity of a character.

Property	Description
Values	0 None (no parity)
	1 Odd parity
	2 Even parity
	3 1 (mark)
	4 0 (space)
Value after reset	2

Tab. 36: Parity

MR 7 Activating the serial interface

This register lets you activate the serial interface.

Property	Description	
Values	0	The serial interface is inactive, no data can be received/transmitted.
	1	The serial interface is active, data can be received/transmitted.
Value after reset	0	

Tab. 37: Activating the serial interface

MR 10 Sending buffer

The character that has to be sent must be entered into this register.

- If the sending buffer is able to accommodate the character, it is entered into this buffer. This character will be sent once all previously entered characters have been sent.
- Prior to sending characters from the application program, it must be checked whether the sending buffer is able to accommodate characters. This can be checked by reading out MR 11.
- The sending buffer functions according to the FIFO principle. The first character entered is sent first.

Property	Description	
Values	0 ... 31	5 bits per character
	0 ... 63	6 bits per character
	0 ... 127	7 bits per character
	0 ... 255	8 bits per character
Type of access	Read	Character written last
	Write access	Sending a character

Tab. 38: Sending buffer

MR 11 Sending buffer filling level

This register shows how many characters the sending buffer accommodates. There is space for 32,768 characters max. within the buffer.

Property	Description
Values	0 ... 32,768

Tab. 39: Sending buffer filling level

MR 12

Receive buffer, 8 bits (without deleting the character on reading)

This register shows the "oldest" character stored in the receive buffer. On reading, this character will not be removed from the buffer.

Property	Description	
Values	0 ... 31	5 bits per character
	0 ... 63	6 bits per character
	0 ... 127	7 bits per character
	0 ... 255	8 bits per character
Type of access	Read	Oldest character in buffer
Takes effect	if MR 14 > 0	

Tab. 40: Receive buffer, 8 bits (without deleting the character on reading)

MR 13

Receive buffer, 8 bits (with deleting the character on reading)

This register shows the "oldest" character stored in the receive buffer. This character is removed from the buffer. Thus, the character received next can be read out during the next read access.

Property	Description	
Values	0 ... 31	5 bits per character
	0 ... 63	6 bits per character
	0 ... 127	7 bits per character
	0 ... 255	8 bits per character
Type of access	Read	Oldest character in buffer
Takes effect	if MR 14 > 0	

Tab. 41: Receive buffer, 8 bits (with deleting the character on reading)

MR 14

Receive buffer filling level

This register shows how many characters the receive buffer accommodates. Each read access to MR 13 decrements this register by 1.

Property	Description
Values	0 ... 32,768

Tab. 42: Receive buffer filling level

MR 15

Receive buffer, 16-bit, little endian

Read access to this register removes 2 characters from the receive buffer and returns them as 16-bit value.

Assignment:

Character	Bits in register
First	Bit 0 ... 7
Second	Bit 8 ... 15

Property	Description	
Values	0 ... 65,535	
Type of access	Read	Removes 2 characters from the buffer
Takes effect	if MR 14 > 1	

Tab. 43: Receive buffer, 16-bit, little endian

MR 16 Receive buffer; 16-bit; big endian

Read access to this register removes 2 characters from the receive buffer and returns them as 16-bit value.

Assignment:

Character	Bits in register
First	Bit 8 ... 15
Second	Bit 0 ... 7

Property	Description
Values	0 ... 65,535
Type of access	Read Removes 2 characters from the buffer
Takes effect	if MR 14 > 1

Tab. 44: Receive buffer; 16-bit; big endian

MR 17 Receive buffer, 32-bit, little endian

Read access to this register removes 4 characters from the receive buffer and returns them as 32-bit value.

Assignment:

Character	Bits in register
First	Bit 0 ... 7
Second	Bit 8 ... 15
Third	Bit 16 ... 23
Fourth	Bit 24 ... 31

Property	Description
Values	-2,147,483,648 ... 2,147,483,647
Type of access	Read Removes 4 characters from the buffer
Takes effect	if MR 14 > 3

Tab. 45: Receive buffer, 32-bit, little endian

MR 18 Receive buffer; 32-bit; big endian

Read access to this register removes 4 characters from the receive buffer and returns them as 32-bit value.

Assignment:

Character	Bits in register
First	Bit 24 ... 31
Second	Bit 16 ... 23
Third	Bit 8 ... 15
Fourth	Bit 0 ... 7

Property	Description
Values	-2,147,483,648 ... 2,147,483,647
Type of access	Read Removes 4 characters from the buffer
Takes effect	if MR 14 > 3

Tab. 46: Receive buffer; **32-bit; big endian**

MR 19

Error counter

This register shows the number of detected errors.

Property	Description
Values	0 ... 2,147,483,647
Type of access	Read/write (clearing)

Tab. 47: Error counter

8.6.3 Programming

Activating the serial interface

Configuring the interface

Module register MR 7 lets you activate the user-programmable serial interface.

- ✓ The JCM-631 and the device to communicate with have been wired according to the RS-232 interface standard.
- ▶ Enter value 1 into MR 7.
- ⇒ The serial interface is activated.

Configuring the serial interface

Configuring the interface

Module registers MR 2 ... MR 6 are used to configure the user-programmable serial interface.

- ✓ The JCM-631 and the device to communicate with have been wired according to the RS-232 interface standard..
- ▶ Enter the desired communication parameters into MR 2 ... MR 6.
- ⇒ The serial interface is set as a user-programmable interface. Transmit and receive buffer are cleared.

Sending characters

Sending characters

A character is sent by entering it into the register *Transmit buffer*.

- ✓ The user-programmable serial interface is configured.
- 1. Check the transmit buffer filling level, whether there is enough space in the transmit buffer.
- 2. If there is no space in the transmit buffer, wait, until there is enough space.
- 3. Enter the character to be sent into register *Transmit buffer*.
- ⇒ The character is written into the transmit buffer and will be sent from there.

Receiving characters

Characters are received by reading characters from register *Receive Buffer*.

Receiving characters

- ✓ The user-programmable serial interface is configured.
 - 1. Check the filling level of the receive buffer to make sure that it contains at least 1 character.
 - 2. Read the character from the register *Receive buffer*.
- ⇒ The character is taken from the receive buffer.

Receiving values

Values are received by reading characters from registers MR 15 through MR 18 *Receive buffer registers*.

Receiving values

- ✓ The user-programmable serial interface is configured.
 - 1. Check the filling level of the receive buffer to make sure that it contains at least 2 or 4 characters.
 - 2. Read the values from *Receive buffer* registers MR 15 through MR 18.
- ⇒ The characters are read from the receive buffer.

8.7 Controlling displays

Connection

JVM-Mxx displays are connected to the FPD Link II interfaces (sockets X81, X82 [▶ 25]) of the controller.

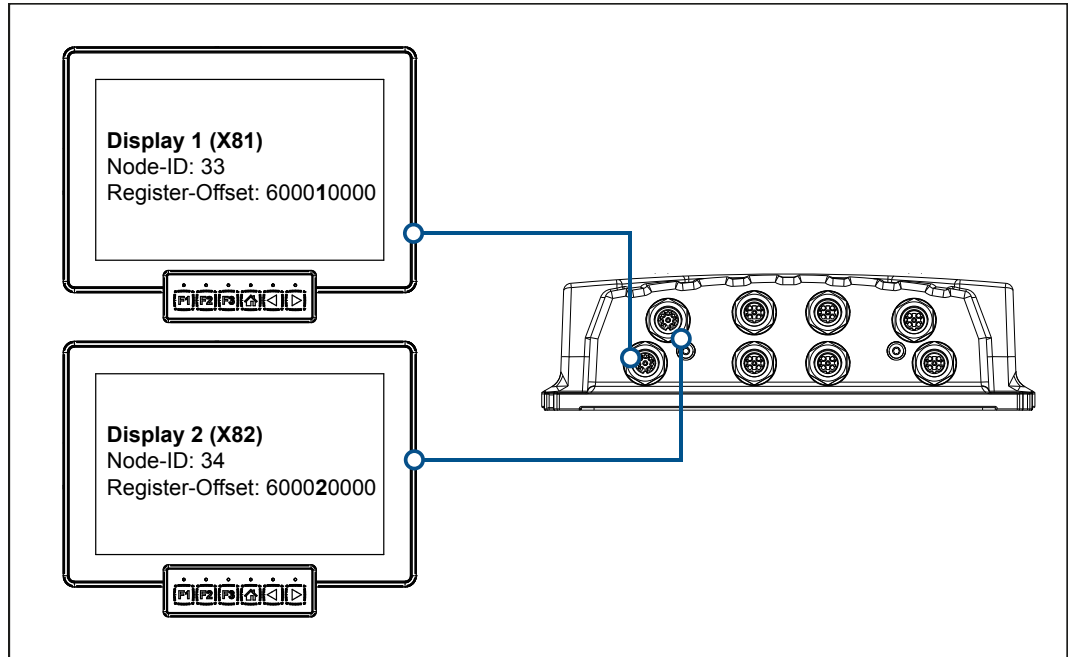


Fig. 21: Controlling displays

Bit-coded mapping of input keys

There are 6 input keys on the left and right side as well as at the bottom. The keys are arranged on the displays as follows:

Display 1 (X81)

Reg Bit	F-Code	Reg Bit	F-Code
0x80	F1	0x01	F7
0x40	F2	0x02	F8
0x20	F3	0x10	F9
0x10	F4	0x20	F10
0x02	F5	0x40	F11
0x01	F6	0x80	F12

Reg Bit	0x80	0x40	0x20	0x10	0x02	0x01
F-Code	F13	F14	F15	F16	F17	F18

Display 2 (X82)

Reg Bit	F-Code	Reg Bit	F-Code
0x80	S1	0x01	S7
0x40	S2	0x02	S8
0x20	S3	0x10	S9
0x10	S4	0x20	S10
0x02	S5	0x40	S11
0x01	S6	0x80	S12

Reg Bit	0x80	0x40	0x20	0x10	0x02	0x01
F-Code	S13	S14	S15	S16	S17	S18

Bit-coded mapping of LEDs

The signal LEDs are set via registers:

Display 1

			Top/ bot- tom	Left	Right
600012000	0x	00	F3	F3	F3

Display 2

			Top/ bot- tom	Left	Right
600022000	0x	00	F3	F3	F3

See also

 Display PLC [[▶ 67](#)]

8.8 FTP server

The controller supports the *FTP server* function. The FTP server lets you handle directories and files using an FTP client.

The files can be stored to the following storage media:

- Internal flash disk
- USB flash drive

FTP server function activation

The *FTP server* function is always enabled on this device. It is enabled when you order this feature.

That is, bit 0 in Web Status register 202930 is always set.

INFO

Further information

For more information on this subject refer to the application-oriented manual *File System* available for download from our [homepage](#).

8.9 Monitoring interface activities

In order to make the variables used within the application program accessible from outside, several Ethernet servers are integrated in the JCM-631. These servers support several protocols on different interfaces. The servers do not require any programming in the application program, but process requests from external clients on their own.

Monitored activities The following interface activities can be monitored:

- JetIP server via Ethernet interface
- STX debug server via Ethernet interface

Purpose The monitoring function for interface activities can be used, amongst others, for the following scenarios:

- Plants requiring process visualization to ensure safe operation. They can be transferred into a save state if communication fails.
- When the service technician connects an HMI, the application program automatically displays additional status information.

INFO

Further information

For more information on this subject refer to the application-oriented manual *System Registers* available for download from our [homepage](#).

8.10 User-programmable IP interface

The user-programmable IP interface lets you send or receive any data via Ethernet interface on the JCM-631 using TCP/IP or UDP/IP. The data is processed in the application program.

Applications

The user-programmable IP interface allows the programmer to carry out data exchange via Ethernet connections which do not use standard protocols, such as FTP, HTTP, JetIP, or Modbus/TCP. The following applications are possible:

- Server
- Client
- TCP/IP
- UDP/IP

Required programmer's skills

To be able to program user-programmable IP interfaces the following knowledge of data exchange via IP networks is required:

- IP addressing (e.g. IP address, port number, subnet mask)
- TCP (e.g. connection establishment/termination, data stream, data backup)
- UDP (e.g. datagram)

Restrictions

For communication via user-programmable IP interface, the programmer must not use any ports which are already used by the operating system of the controller. Therefore, do not use the following ports:

Protocol	Port number	Default value	User
TCP	Depending on the FTP client	20	FTP server (data)
TCP	21		FTP server (controller)
TCP	23		System logger
TCP	80		HTTP server
TCP	From the file / EMAIL/email.ini	25, 110	E-mail client
TCP	502		Modbus/TCP server
TCP, UDP	1024 - 2047		Various
TCP, UDP	IP configuration	50000, 50001	JetIP
TCP	IP configuration	52000	Debug server

INFO

Further Information

For more information on this subject refer to the application-oriented manual *User-Programmable Interfaces* available for download from our [homepage](#).

9 Registers - Overview

This register quick reference summarizes in brief the registers and flags used by the JCM-631 running OS version 1.01.1.05.

Default IP address

IP address	192.168.10.15
Subnet mask	255.255.255.0

Default address on the CANopen® bus

This device is delivered with the following default address on the CANopen® bus:

Node ID	127 (0x7F)
---------	------------

General overview - Registers

Register range	Description
100000 ... 100999	Electronic Data Sheet (EDS)
102000 ... 102999	Real-time clock
103000 ... 103999	Serial interface RS-232
104000 ... 104999	Ethernet
106000 ... 106499	CAN
107000 ... 107999	Flash disk
108000 ... 108599	System information
109000 ... 109003	Revisions
200000 ... 209999	General system registers
210000 ... 219999	Application program
310000 ... 319999	File system/data files
360000 ... 369999	Display PLC
1000000 ... 1029999	Application registers (non-volatile; integer/float)

Register range	Description
600000000 ... 600299999	JCM registers (JCM-631 only)

Flags - General overview

Register range	Description
0 ... 255	Application flags (non-volatile)
256 ... 2047	Overlaid by registers R 1000000 through 1000055
2048 ... 2303	Special flags

Electronic nameplate (entire device)

Register range	Description
100900 ... 100993	Production
100900	Internal version number
100901 ... 100907	Serial number (register string)
100992	Navision ID
100993	FBG version

Ethernet

Registers	Description
101100	IP address (r/w)

Real-time clock

Register	Description
102810 ... 102820	Direct access (localized)
102810	Milliseconds
102811	Seconds
102812	Minutes
102813	hours
102814	Day of the week (0 = Sunday)
102815	Day
102816	Month
102817	Year
102818	Deviation in minutes
102819	Standard deviation plus daylight savings time
102820	Daylight savings time is active

Register	Description
102910 ... 102917	Direct access (UTC)
102910	Milliseconds
102911	Seconds
102912	Minutes
102913	hours
102914	Day of the week (0 = Sunday)
102915	Day
102916	Month
102917	Year
102920 ... 102928	Buffer access (UTC - not suitable for JetSym setup)
102920	Milliseconds
102921	Seconds
102922	Minutes
102923	hours
102924	Day of the week (0 = Sunday)
102925	Day
102926	Month
102927	Year
102928	Read/write trigger

Serial port

Register	Description
X = COM port number - 1	
103X02	Baud rate (1200 ... 115200)
103X03	Bits per character (5 ... 8)
103X04	Stop bits (1, 2)
103X05	Parity
0	None
1	Odd
2	Even
3	1
4	0
103X07	Port activation
0	OFF
1	ON
103X10	Sending buffer
103X11	Sending buffer filling level
103X12	Receiving buffer (without immediate clearing)
103X13	Receiving buffer (with immediate clearing)
103X14	Receive buffer filling level
103X15	Receive buffer, 16-bit, little endian
103X16	Receive buffer; 16-bit; big endian
103X17	Receive buffer, 32-bit, little endian

Register	Description
103x18	Receive buffer; 32-bit; big endian
103X19	Error counter

Revisions

Registers	Description
190000	Kernel version
190001	File collection version
190002	Coprocessor version
190003	FPGA revision

Display PLC

Register range	Description
X = 1 for display 1 at X81 (offset 600010000)	
X = 2 for display 2 at X82 (offset 600020000)	
6000X1000	Input keys
6000X2000	LEDs
...	
6000X2492	
6000X2000	Bit-coded mapping of LEDs
...	
6000X2006	
6000X3000	Digipot
...	
6000X3003	
6000X3000	Count Value
6000X3001	Digipot button
6000X3002	Minimum count value
6000X3003	Maximum count value
6000X4000	Sensors and actuators
...	
6000X4011	
6000X4000	Display backlight 0 ... 100%
6000X4001	Keys backlight 0 ... 100%
6000X4010	Display temperature in °C
6000X4012	Display operating voltage in mV
6000X7600	Display EDS
...	
6000X7712	
	Page 0
6000X7600	Internal version number
6000X7601	Device identifier
6000X7602	Device name (register string)
...	
6000X7612	
6000X7613	PCB revision
6000X7614	PCB options
6000X7615	Min. OS version

Register range	Description
6000X7616	Min. bootloader version
	Page 1
6000X7700	Internal version number
6000X7701	Serial number (register string)
...	
6000X7707	
6000X7708	Day
6000X7709	Month
6000X7710	Year
6000X7711	TestNum.
6000X7712	TestRev.
	Page 2
6000X7800	Internal version number
6000X7801	Display type
6000X7802	CAN-ID
6000X7803	Touch keys
6000X7804	Pushbuttons
6000X7805	LEDs
6000X7806	Digital input
6000X7807	Analog input
6000X7808	RotaryEncoder
6000X7809	Horn
6000X7810	Vendor ID
6000X7900	Electronic nameplate
...	
6000X7999	
6000X7900	Version
6000X7901	Serial number
6000X7913	Item number
6000X7919	Hardware revision
6000X9000	Version registers
...	
6000X9099	
6000X9000	Firmware version (register string)
6000X9010	FPGA version (register string)

General system registers

Register	Description
200000	OS version (major * 100 + minor)
200001	Application program is running (bit 0 = 1)
0/2	Stop program
1	Start program
3	Continue program
200169	OS version (IP format)
200170	Controller model
202980	Error history: Number of entries

Register	Description
202981	Error history: Index
202982	Error history: Entry
203000	Interface monitoring: JetIP
203001	Interface monitoring: SER
203005	Interface monitoring: STX debug server
203100 ... 203107	32-bit overlaying - Flag 0 ... 255
203108 ... 203123	16-bit overlaying - Flag 0 ... 255
203124 ... 203131	32-bit overlaying - Flag 2048 ... 2303
203132 ... 203147	16-bit overlaying - Flag 2048 ... 2303
209700	System logger: Global enable
209701 ... 209754	Enabling system components

System Functions

For reasons of compatibility, the system functions are listed below.

In JetSym STX, use the corresponding JetSym STX functions instead of the system functions.

System function	Description
4	Converting BCD to HEX
5	Converting HEX to BCD
20	Square root
21	Sine
22	Cosine
23	Tangent
24	Arc sine
25	Arc cosine
26	Arc tangent
27	Exponential function
28	Natural logarithm
29	Absolute value
30	Separation of digits before and after the decimal point
50	Sorting register values
60	CRC generation for Modbus RTU
61	CRC check for Modbus RTU
65/67	Reading register block via Modbus/TCP
66/68	Writing register block via Modbus/TCP
80/85	Initializing RemoteScan
81	Starting RemoteScan
82	Stopping RemoteScan

System function	Description
90	Writing a data file
91	Appending a data file
92	Reading a data file
96	Deleting a data file
150	Configuring NetCopyList
151	Deleting NetCopyList
152	Sending NetCopyList

System function	Corresponding JetSym STX function
4	Function Bcd2Hex(Bcd: int): Int;
5	Function Hex2Bcd(Hex: int): Int;
50	Function QSort(DataPtr: Int, ElementCnt: Int, ElementSize: Int, SortOffset: Int, SortType: STXBASETTYPE, SortMode: QSORTMODE): Int;
60	Function ModbusCRC-gen(FramePtr: Int, Length: int): Int;
61	Function ModbusCRC-check(FramePtr: Int, Length: int): Int;
65/67	Function ModbusReadReg(Const Ref MbParam: MODBUS_PARAM): Int;
66/68	Function ModbusWriteReg(Const Ref MbParam: MODBUS_PARAM): Int;
80/85	Function RemoteScanConfig(Protocol: RSCAN_PROTOCOL, Elements: Int, Const Ref Configuration: RSCAN_DSCR): Int;
81	Function RemoteScanStart(Protocol: int): Int;
82	Function RemoteScanStop(Protocol: int): Int;
90/91	Function FileDAWrite(Const Ref FileName: String, Const Ref Mode: String, VarType: DAWRITE_TYPE, First: Int, Last: int): Int;
92	Function FileDARead(Const Ref FileName: String): Int;
110	Function EmailSend(Const Ref FileName: String): Int;
150	Function NetCopyListConfig(IPAddr: Int, IPPort: Int, Const Ref List: TNetCopyLinstL): Int;
151	Function NetCopyListSend(Handle: int): Int;
152	Function NetCopyListDelete(Handle: int): Int;

Application program

Register	Description
210000	Application program is running (bit 0 = 1)
210001	JetVM version
210004	Error register (bit-coded)
Bit 8	Illegal jump
Bit 9	Illegal call
Bit 10	Illegal index
Bit 11	Illegal opcode
Bit 12	Division by 0
Bit 13	Stack overflow
Bit 14	Stack underflow
Bit 15	Illegal stack
Bit 16	Error when loading the application program
Bit 24	Timeout - Cycle time
Bit 25	Timeout - Task lock
Bit 31	Unknown error
210006	Highest task number
210007	Minimum program cycle time
210008	Maximum program cycle time
210009	Current program cycle time
210011	Current task number
210050	Current program position within an execution unit
210051	ID of the execution unit being processed
210056	Desired total cycle time in μ s
210057	Calculated total cycle time in μ s
210058	Maximum time slice per task in μ s
210060	Task ID (for R210061)
210061	Priority for task [R210060]
210063	Length of scheduler table
210064	Index in scheduler table
210065	Task ID in scheduler table
210070	Task ID (for R210071)
210071	Timer number (0 ... 31)
210072	Manual triggering of a timer event (bit-coded)
210073	End of cyclic task (task ID)
210074	Command for cyclic tasks
210075	Number of timers
210076	Timer number (for R210077)
210077	Timer value in milliseconds
210100 ... 210199	Task state Please use the STX function TaskGetInfo() as described in the JetSym online help.

Register	Description
210400 ... 210499	Task - Program address
210600	Task ID of a cyclical task (for R210601)
210601	Processing time of a cyclical task in per mil figure
210609	Task lock timeout in ms
-1	Monitoring disabled
210610	Timeout (bit-coded, bit 0 → timer 0 etc.)

File system/data file function

Register	Description
312977	Status of file operation
312978	Task ID

Register interfaces

Register range	Description	Type of access	Unit
700,000,000	Main_Status	R	BIT-FIELD
Bit 0	PowerSupplyUnder-voltage		
Bit 1	RTC_BatteryUnder-voltage		
Bit 2	CAM1_Undervoltage		
Bit 3	CAM1_Overvoltage		
Bit 4	CAM2_Undervoltage		
Bit 5	CAM2_Overvoltage		
Bit 6	CAM3_Undervoltage		
Bit 7	CAM3_Overvoltage		
Bit 8	CAM4_Undervoltage		
Bit 9	CAM4_Overvoltage		
Bit 10	CAM5_Undervoltage		
Bit 11	CAM5_Overvoltage		
Bit 12	CAM6_Undervoltage		
Bit 13	CAM6_Overvoltage		
Bit 14	CAM7_Undervoltage		
Bit 15	CAM7_Overvoltage		
Bit 16	CAM8_Undervoltage		
Bit 17	CAM8_Overvoltage		
Bit 18	DISP1_Undervoltage		
Bit 19	DISP2_Undervoltage		
Bit 20	DISP3_Undervoltage		
Bit 21	Base-board_Overtemperature		
Bit 22	Baseboard_Under-temperature		
Bit 23	FP-GAboard_Overtemperature		
Bit 24	FPGAboard_Under-temperature		
Bit 25	CPUboard_Overtemperature		
Bit 26	CPUboard_Under-temperature		
Bit 27	Baseboard_Power-Fault		
Bit 28	FPGABoard_Power-Fault		
Bit 29	CPUBoard_Power-Fault		
Bit 30	PowerON_INPUT_IS_OFF		

Register range	Description	Type of access	Unit
700,000,001	PowerSupplyVoltage	R	1 mV
700,000,002	RTC_BatteryVoltage	R	1 mV
700,000,003	CAMx_OvervoltageThreshold	R/W	1 mV
700,000,004	CAMx_UndervoltageThreshold	R/W	1 mV
700,000,005	CAM1_PowerVoltage	R	1 mV
700,000,006	CAM2_PowerVoltage	R	1 mV
700,000,007	CAM3_PowerVoltage	R	1 mV
700,000,008	CAM4_PowerVoltage	R	1 mV
700,000,009	CAM5_PowerVoltage	R	1 mV
700,000,010	CAM6_PowerVoltage	R	1 mV
700,000,011	CAM7_PowerVoltage	R	1 mV
700,000,012	CAM8_PowerVoltage	R	1 mV
700,000,013	DSPx_UndervoltageThreshold	R/W	1 mV
700,000,014	DSP1_PowerVoltage	R	1 mV
700,000,015	DSP2_PowerVoltage	R	1 mV
700,000,016	DSP3_PowerVoltage	R	1 mV
700,000,017	OvertemperatureThreshold	R/W	0.1 °C
700,000,018	UndertemperatureThreshold	R/W	0.1 °C
700,000,019	Baseboard_Temperature	R	0.1 °C
700,000,020	FPGAboard_Temperature	R	0.1 °C
700,000,021	CPUboard_Temperature	R	0.1 °C
700,000,022	Baseboard_5V	R	1 mV
700,000,023	Operating_time (Betriebsminuten Zähler Persistent im RTC gespeichert.)	R	1 min

Register range	Description	Type of access	Unit
700,000,040	DIGITAL_IO_Status	R	BIT-FIELD
Bit 0	DI1_DigitalValue		
Bit 1	DI2_DigitalValue		
Bit 2	DI3_DigitalValue		
Bit 3	DO1_DigitalFeedbackValue		
Bit 4	DO2_DigitalFeedbackValue		
Bit 5	DO3_DigitalFeedbackValue		
Bit 6	DO1_OverCurrent		
Bit 7	DO2_OverCurrent		
Bit 8	DO3_OverCurrent		
700,000,041	DIGITAL_IO_Mode	R/W	BIT-FIELD
Bit 0	DI1_RatiometricThreshold		
Bit 1	DI2_RatiometricThreshold		
Bit 2	DI3_RatiometricThreshold		
700,000,042	DI1_Threshold	R/W	1 mV
700,000,043	DI1_Voltage	R	1 mV
700,000,044	DI1_Threshold	R/W	1 mV
700,000,045	DI2_Voltage	R	1 mV
700,000,046	DI3_Threshold	R/W	1 mV
700,000,047	DI3_Voltage	R	1 mV
700,000,048	DO1_DigitalValue	R/W	1 or 0
700,000,049	DO1_VoltageThreshold	R/W	1 mV
700,000,050	DO1_Voltage	R	1 mV
700,000,051	DO1_OverCurrentThreshold	R/W	1 mA
700,000,052	DO1_Current	R	1 mA
700,000,053	DO2_DigitalValue	R/W	1 or 0
700,000,054	DO2_VoltageThreshold	R/W	1 mV
700,000,055	DO2_Voltage	R	1 mV
700,000,056	DO2_OverCurrentThreshold	R/W	1 mA
700,000,057	DO2_Current	R	1 mA
700,000,058	DO3_DigitalValue	R/W	1 or 0
700,000,059	DO3_VoltageThreshold	R/W	1 mV
700,000,060	DO3_Voltage	R	1 mV
700,000,061	DO3_OverCurrentThreshold	R/W	1 mA
700,000,062	DO3_Current	R	1 mA

Register range	Description	Type of access	Unit
700,000,063	LED_D2 (value from table in chapter 5.3.2)	R/W	
700,000,080	If 1 is entered, a re-boot is performed. If POWER_ON = 0, the ShutdownDelay is shortened	W	1 or 0
701,000,000 ... 704,999,999	FPGA Shared Memory	R/W	
700,000,064	DISP_x Power Control (default is on = 1)	R/W	1 or 0

Application registers

Register	Description
1000000 ... 1005999	32-bit integer (remanent)

32 combined flags

Register	Description
203100	0 ... 31
203101	32 ... 63
203102	64 ... 95
203103	96 ... 127
203104	128 ... 159
203105	160 ... 191
203106	192 ... 223
203107	224 ... 255

16 combined flags

Register	Description
203108	0 ... 15
203109	16 ... 31
203110	32 ... 47
203111	48 ... 63
203112	64 ... 79
203113	80 ... 95
203114	96 ... 111
203115	112 ... 127
203116	128 ... 143
203117	144 ... 159
203118	160 ... 175

Register	Description
203119	176 ... 191
203120	192 ... 207
203121	208 ... 223
203122	224 ... 239
203123	240 ... 255

32 combined special flags

Register	Description
203124	2048 ... 2079
203125	2080 ... 2111
203126	2112 ... 2143
203127	2144 ... 2175
203128	2176 ... 2207
203129	2208 ... 2239
203130	2240 ... 2271
203131	2272 ... 2303

16 combined special flags

Register	Description
203132	2048 ... 2063
203133	2064 ... 2079
203134	2080 ... 2095
203135	2096 ... 2111
203136	2112 ... 2127
203137	2128 ... 2143
203138	2144 ... 2159
203139	2160 ... 2175
203140	2176 ... 2191
203141	2192 ... 2207
203142	2208 ... 2223
203143	2224 ... 2239
203144	2240 ... 2255
203145	2256 ... 2271
203146	2272 ... 2287
203147	2288 ... 2303

Application registers with overlaid flags

Register	Description
1000000	256 ... 287
1000001	288 ... 319
1000002	320 ... 351
1000003	352 ... 383
1000004	384 ... 415

Register	Description
1000005	416 ... 447
1000006	448 ... 479
1000007	480 ... 511
1000008	512 ... 543
1000009	544 ... 575
1000010	576 ... 607
1000011	608 ... 639
1000012	640 ... 671
1000013	672 ... 703
1000014	704 ... 735
1000015	736 ... 767
1000016	768 ... 799
1000017	800 ... 831
1000018	832 ... 863
1000019	864 ... 895
1000020	896 ... 927
1000021	928 ... 959
1000022	960 ... 991
1000023	992 ... 1023
1000024	1024 ... 1055
1000025	1056 ... 1087
1000026	1088 ... 1119
1000027	1120 ... 1151
1000028	1152 ... 1183
1000029	1184 ... 1215
1000030	1216 ... 1247
1000031	1248 ... 1279
1000032	1280 ... 1311
1000033	1312 ... 1343
1000034	1344 ... 1375
1000035	1376 ... 1407
1000036	1408 ... 1439
1000037	1440 ... 1471
1000038	1472 ... 1503
1000039	1504 ... 1535
1000040	1536 ... 1567
1000041	1568 ... 1599
1000042	1600 ... 1631
1000043	1632 ... 1663
1000044	1664 ... 1695
1000045	1696 ... 1727
1000046	1728 ... 1759
1000047	1760 ... 1791
1000048	1792 ... 1823
1000049	1824 ... 1855
1000050	1856 ... 1887
1000051	1888 ... 1919
1000052	1920 ... 1951

Register	Description
1000053	1952 ... 1983
1000054	1984 ... 2015
1000055	2016 ... 2047

10 Maintenance and repairs

10.1 Maintenance, repairs and disposal

Maintenance	<p>This device is maintenance-free. Therefore, for the operation of the device no inspection or maintenance are required.</p>
Repairs	<p>Defective components could cause dangerous malfunctions and could compromise safety. Only the manufacturer is allowed to repair the device. Do not open the device!</p>
Disposal of obsolete equipment	<p>The device must be disposed of in accordance with the Environmental Product Declaration EPD. Applicable local environmental directives and regulations must be complied with. This product must be disposed of as waste electronic equipment. Waste packaging material must be recycled or reused. You can return the device to Jetter AG for disposal.</p>
Modifications and alterations to the device	<p>Modifications and alterations to the device and its functions are not allowed. In the case of modifications to the device, any liability is excluded. The original parts are specifically designed for the device. Parts and equipment from other manufacturers must, therefore, not be used. Any liability for any damages resulting from the use of non-original parts and equipment is excluded.</p>

10.2 Storage and shipment

Storage	<p>When storing the device observe the environmental conditions given in chapter "Technical specifications".</p>
Shipment and packaging	<p>The device contains electrostatically sensitive components which can be damaged if not handled properly. Damages to the device may impair its reliability. To protect the device from impact or shock, it must be shipped in its original packaging, or in an appropriate protective ESD packaging. In case of damaged packaging inspect the device for any visible damage, and inform your freight forwarder and the Jetter AG of the damage caused during shipment. If the device is damaged or has been dropped, it is strictly forbidden to use it.</p>

11 Service

11.1 Customer service

Should you have any questions, suggestions, or problems, please don't hesitate to contact our service representatives. To contact them, please call our technical hotline or use the contact form on our homepage:

[*Technical hotline | Jetter - We automate your success.*](#)

You are also welcome to send an e-mail to our technical hotline:

[*hotline@jetter.de*](mailto:hotline@jetter.de)

Please supply the following information when contacting our technical hotline:

- Hardware revision and serial number
For the hardware revision number, please refer to the nameplate.
- OS version
To determine the operating system version, use the functions of the development environment.

12 Spare parts and accessories

NOTICE

Inadequate accessories might cause damage to the product

Parts and equipment from other manufacturers might impede the function of the device and cause damage to the product.

- ▶ Only use accessories recommended by Jetter AG.

12.1 Accessories

INFO

Ordering accessories

The accessories are not part of the scope of delivery.
Suitable accessories are available from Jetter AG.

Component	Item number
EVAL Kit bestehend aus <ul style="list-style-type: none"> ■ Cable set 0.25 m ■ Connector boxes with 4 mm banana jacks ■ Labeling strips JCM-631 	On request
Programming cable 2 m M12/8-pin/X-coded, to RJ45 (Cat5e)	60880135
Programming and service cable USB socket and RS-232 to sub-D	60882337

Tab. 48: Accessories

12.1.1 Monitors

Component	Item number
Display 4.3" resolution 480 x 272 wide, buzzer, RAM mount	10001678
Display 10.1" resolution 1024 x 600 wide, PCAP touch, Buzzer, VESA mount	10001679
Connecting cable for 10001678 and 10001679 on the JCM-631, length 3 m.	60882820
Connecting cable for 10001678 and 10001679 on the JCM-631 with angled connector, length 3.5 m.	60883231

Tab. 49: Accessories – Monitors

12.1.2 USB flash drives

Component	Item number
USB flash drive, 4 GB	60875553
USB flash drive XMORE, 4 GB, XQC8	60876836
USB flash drive, 8 GB	60881279

Tab. 50: Accessories - USB flash drives

Jetter AG
Graeterstrasse 2
71642 Ludwigsburg
www.jetter.de

E-mail info@jetter.de
Phone +49 7141 2550-0

We automate your success.