JC-340

Version Update

from V. 1.14 to V. 1.16



Version Update

Jetter

Revision 1.01
October 2012 / Printed in Germany

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

This Version Update and the information contained herein have been compiled with due diligence. However, Jetter AG assume no liability for printing or other errors or damages arising from such errors.

The brand names and product names used in this document are trademarks or registered trademarks of the respective title owner.

Table of Contents

1 Introduction 4

Operating System Update....5
JC-340 Version Update - Overview....6

2 New Features 11

2.1 Various New Features and Modifications....12
System Functions as STX Functions....13
New Data Types: RegString and dword....16
STX Memory Protection....17

3 Fixed Software Bugs 18

Wrong Values in Program Cycle Time Registers After Program Start....19
NetCopyVarToReg Does Not Work Properly in Conjunction With Repetitions....20
Wrong Value in Task Status Register if a Cyclic Task is Aborted....21
Wrong Values in Registers of CANopen® Modules With Profile DS402....22
Incorrect Formatting of CANopen® Application Registers....23

1 Introduction

Introduction

This chapter shows the history of OS versions for the controller JC-340.

Operating System Update - Why?

An OS update allows you to:

- add new functions to your controller
- fix software bugs
- make sure your controller is working with a definite OS version, for example, if a definite OS version has been released for a certain customer

Contents

Topic	Page
Operating System Update	5
JC-340 Version Update - Overview	6

Operating System Update

OS File for Updating the Operating System

For updating the OS the following file is needed:

OS File	Description
JC-340_1.16.0.00.os	OS file for JC-340 with version 1.16

Downloading the OS File

Jetter AG make operating system files available for download from their **homepage at http://www.jetter.de**. OS files can be found in the support area or on the page of the JC-340 controller via quicklink.

Operating System Update by means of JetSym

To update your OS proceed as follows:

Step	Action
1	Download the OS file from www.jetter.de
2	Establish a connection between PC and controller
3	In JetSym: Select menu item "Build -> Update OS" or Click on the button "OS Update" in the CPU window of the hardware manager
4	Select the OS File
5	Initiate the OS update by clicking OK
6	**Result:** Following Power OFF / Power ON the new OS is launched.

Minimum Requirements

For programming a JC-340 with version 1.16 JetSym 5.0.0 or higher is required.

JC-340 Version Update - Overview

V 1.04

The following table gives an overview of newly added features and fixed software bugs in OS version 1.04:

Description	New	Fixed
JX2 system bus:		
Register overlaying for digital inputs/outputs	✓	
Support of JX-SIO modules and third-party CANopen® devices	✓	
JX3 system bus:		
Register overlaying for digital inputs/outputs	✓	
System bus special registers for status and control	✓	
Operating system update:		
Via FTP: On completion notification the OS has actually been stored.		✓
Updating a JX2 slave module while registers are being accessed blocks communication		✓
Application program:		
Task switch could fail to happen		✓
Error signal in case of invalid file "/app/start.ini"		✓
Display commands:		
Redirection to JX2-SER1 works only if JX2-PRN1 has been configured, too		✓

V 1.05

The following table gives an overview of newly added features and fixed software bugs in OS version 1.05:

Description	New	Fixed
JX2 system bus: V1.05.0.00		
AS interface gateway BWU1821 is supported	✓	
Frequency inverter 8200 vector is supported	✓	
JetMove 1xx is not detected during boot process		✓
Automatic baud rate recognition does not work reliably for some of the baud rates and configurations of IP67 modules.		✓
Repetition counter does not work when polling I/O modules		✓
AutoCopy function:		
Automatic copying of controller data	✓	
Application program:		
Pending cyclic tasks are started immediately after Taskunlock	✓	
For function pow(x,y) a floating point number can be entered as exponent	✓	
Cyclic tasks can be debugged	✓	

Description	New	Fixed
Length of project and program names > 39 characters		✓
Restart of an elapsed timer		✓
The value returned by DateTimeDecode() was always 1 day short of the actual day.		✓
DateTimeEncode and -IsValid might return the value TRUE irrespective of an invalid date		✓
User registers:		
The register type can be set up without having to start the application program	✓	
Displays and HMIs:		
A floating point value can be used as default for UserInput	✓	
The default value for UserInput is not displayed correctly		✓
It is not possible to enter LED register numbers		✓

V 1.08

The following table gives an overview of newly added features and fixed software bugs in OS version 1.08:

Description	New	Fixed
System configuration:		
System rights for configuration file	✓	
JX2 system bus: V1.11.0.00		
Timeout after CAN-PRIM message		✓
Registers of LJX7-CSL modules		✓
Write access to analog outputs of CANopen® modules		✓
State of digital inputs when the controller is powered on		✓
Digital outputs on JX-SIO or CANopen® modules		✓
Input/output 64 on JX-SIO or CANopen® modules		✓
User-programmable CAN interface		✓
Application program:		
NetCopyList Functions	✓	
StrCopy()		✓
Crash in the case of "invalid" application program		✓
NetCopyVarFromReg()		✓
JX3 system bus:		
Module registers for digital I/Os	✓	
Displays and HMIs:		
UserInput()		✓

V 1.09

The following table gives an overview of newly added features and fixed software bugs in OS version 1.09:

Description	New	Fixed
System:		
System command register	✓	
JX2 system bus: V1.13.0.00		
Status change of inputs on JX2-ID8		✓
Status change of fast inputs		✓
Application program:		
FTP client	✓	
Axis instructions		✓
Taskrestart in the case of Delay()		✓
Crash in the case of missing library		✓
Floating-point number registers in data files		✓
NetCopyVarToReg with floating-point number registers		✓
JX3 system bus:		
Dummy Modules	✓	
AutoCopy:		
FTP commands	✓	
Serial interface:		
Initialization after booting		✓

V 1.10

The following table gives an overview of newly added features and fixed software bugs in OS version 1.10:

Description	New	Fixed
System:		
LED registers		✓
SD memory card		✓
JX2 system bus: V1.17.0.00		
Additional modules	✓	
CAN-PRIM	✓	
Application program:		
Task commands with variable parameters	✓	
UserInput()		✓
NetCopyListSend()		✓
Task status register		✓

Description	New	Fixed
Real-time clock:		
Additional register for milliseconds	✓	
User-programmable IP interface:		
More connections	✓	

V 1.12

The following table gives an overview of newly added features and fixed software bugs in OS version 1.12:

Description	New	Fixed
System:		
System command register	✓	
JX2 system bus: V1.21.0.00		
Initialization		✓
CAN-PRIM		✓
CANopen® sync interval		✓
CANopen® application registers		✓
CANopen® type string		✓
CANopen® writing output values		✓
CANopen® version number		✓
WAGO 750		✓
JX3 system bus:		
Register accesses		✓
Application program:		
Program control	✓	✓
Assigning structures	✓	
Sorting data	✓	
Displaying variables in JetSym		✓
HTTP server:		
New file type	✓	
Serial interface:		
Error detection		✓

V 1.14

The following table gives an overview of newly added features and fixed software bugs in OS version 1.14:

Description	New	Fixed
JX2 system bus: V1.22.0.00		
OS update		✓
Application program:		
New instructions	✓	

V 1.16

The following table gives an overview of newly added features and fixed software bugs in OS version 1.16:

Description	New	Fixed
JX2 system bus: V1.23.0.00		
CANopen® registers		✓
Application program:		
New data types	✓	
New functions	✓	
Memory protection	✓	
Cyclic tasks		✓
NetCopyVarToReg		✓
Cycle time registers		✓

2 New Features

Introduction

This chapter describes the features which have been added or enhanced in the new software release.

Contents

Topic	Page
Various New Features and Modifications	12

2.1 Various New Features and Modifications

Introduction

This chapter covers the new features and modifications

Contents

Topic	Page
System Functions as STX Functions	13
New Data Types: RegString and dword	16
STX Memory Protection	17

System Functions as STX Functions

Introduction

Several system functions have already been provided by the STX instruction set, so far. Now, using this extension the other functions can be invoked by STX functions, too. To ensure compatibility with existing programs the system functions can be used as before not requiring changes to the program.

System Functions

For function parameters and data on which system functions are based only registers (%VL) can be used.

STX Functions

For function parameters and data on which STX functions are based only STX variables (%RL or not localized variables) can be used. The results of these functions correspond to the results returned by the system functions.

Description of Functions

For a more detailed description of the STX programming language, including sample programs, refer to JetSym online help.

Functions

The table below compares system functions and STX functions:

SystemFunction	STX Function Declaration
4	`function Bcd2Hex(Bcd: int): Int`
5	`function Hex2Bcd(Hex: int): Int;`
50	`function QSort(DataPtr: Int, ElementCnt: Int, ElementSize: Int, SortOffset: Int, SortType: STXBASETYPE, SortMode: QSORTMODE): Int;`
60	`function ModbusCRCgen(FramePtr: Int, Length: int): Int;`
61	`function ModbusCRCcheck(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	`function FileDAWrite(Const Ref FileName: String, Const Ref Mode: String, VarType: DAWRITE_TYPE, First: Int, Last: int): Int;`
110	`function EmailSend(Const Ref FileName: String): Int;`

Example: Format Conversion

SystemFunction

```
SystemFunction(4, RegNoBcd, RegNoHex);
```

STX Function

```
VarHex := Bcd2Hex(VarBcd);
```

SystemFunction

```
SystemFunction(5, RegNoHex, RegNoBcd);
```

STX Function

```
VarBcd := Hex2Bcd(VarHex);
```

Example: Sorting Data

SystemFunction

```
SystemFunction(50, RegNoParam, RegNoResult);
```

STX Function

```
nResult := QSort(&aSort, nItems, sizeof(SORTTYPE), nSortIdx,
nSortType, nSortMode);
```

Example: Modbus RTU

SystemFunction

```
SystemFunction(60, RegNoFirst, RegNoLast);
```

STX Function

```
// Adds the CRC also to the frame
nCRC := ModbusCRCgen(&abyFrame, nLen);
```

SystemFunction

```
SystemFunction(61, RegNoFirst, RegNoLast);
```

STX Function

```
// 0: CRC wrong; 1: CRC correct
nResult := ModbusCRCcheck(&abyFrame, nLen);
```

Example: Modbus/TCP

SystemFunction

```
SystemFunction(67, RegNoParam, RegNoResult);
```

STX Function

```
nResult := ModbusReadReg(stModbusParam);
```

SystemFunction

```
SystemFunction(68, RegNoParam, RegNoResult);
```

STX Function

```
nResult := ModbusWriteReg(stModbusParam);
```

Example: RemoteScan

SystemFunction

```
SystemFunction(85, RegNoParam, RegNoResult);
```

STX Function

```
nResult := RemoteScanConfig(RSCAN_PROTOCOL.MODBUSTCP, nElements,
stRScanParam[0]);
```

SystemFunction

```
SystemFunction(81, RegNoParam, RegNoResult);
```

STX Function

```
nResult := RemoteScanStart(RSCAN_PROTOCOL.MODBUSTCP);
```

SystemFunction

```
SystemFunction(82, RegNoParam, RegNoResult);
```

STX Function

```
nResult := RemoteScanStop(RSCAN_PROTOCOL.MODBUSTCP);
```

Example: Writing DA file

SystemFunction

```
SystemFunction(90, RegNoParam, RegNoResult);
```

STX Function

```
// The file name is subject only to file system restrictions
nResult := FileDAWrite(szFileName, 'w', eVarType, nFirstReg,
nLastReg);
```

SystemFunction

```
SystemFunction(91, RegNoParam, RegNoResult);
```

STX Function

```
// The file name is subject only to file system restrictions
nResult := FileDAWrite(szFileName, 'a', eVarType, nFirstReg,
nLastReg);
```

Example: Sending E-Mails

SystemFunction

```
SystemFunction(110, RegNoParam, RegNoResult);
```

STX Function

```
// The file name is subject only to file system restrictions
nResult := EmailSend(szFileName);
```

New Data Types: RegString and dword

Introduction	Starting from this OS version the controller JC-340 supports the STX data types `RegString` and `dword`
Prerequisites	The new data types can be used starting from JetSym version 5.0.0.
Reference	For more information on these data types and on how they can be used refer to JetSym online help.

STX Memory Protection

Introduction

Starting from this OS version the controller JC-340 supports the memory protection feature provided by JetSym STX.

Prerequisites

The memory protection feature can be used starting from JetSym version 5.0.0.

Size of Data Memory

So far, the memory size for non-localized variables of an STX application program could not always be specified. Thanks to the implementation of the new memory administration required for memory protection the size can be quantified:

Controller	Data memory
JC-340/JC-350	2 MB
JC-360/JC-360MC	4 MB
JC-940MC	8 MB
JetSTX-VM	2 MB
FMC01	0.5 MB

Reference

For more information on STX memory protection refer to JetSym online help.

3 Fixed Software Bugs

Introduction

This chapter describes the software bugs which have been fixed in the new operating system release.

Contents

Topic	Page
Wrong Values in Program Cycle Time Registers After Program Start	19
NetCopyVarToReg Does Not Work Properly in Conjunction With Repetitions	20
Wrong Value in Task Status Register if a Cyclic Task is Aborted	21
Wrong Values in Registers of CANopen® Modules With Profile DS402	22
Incorrect Formatting of CANopen® Application Registers	23

Wrong Values in Program Cycle Time Registers After Program Start

Symptoms

After program start, both registers for minimum and maximum program cycle time indicate wrong values. This is true for program starts caused by energizing the controller or triggered by JetSym.

The following registers are affected:

- 210007
- 210008

Affected Versions/Revisions

The following versions/revisions are affected by this bug:

OS version	JC-340/350 JC-360/360MC	< 1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Remedy/Workaround

In your application program, enter at the beginning of the first task after a short delay time a very large value into register 210007 and 0 into registers 210008.

For example:

```
Task tMain Autorun
    Delay(0);
    Regs[210007] := 1000000;
    Regs[210008] := 0;
    ...
```

Fixed Versions/Revisions

Starting from the following versions/revisions this bug has been fixed:

OS version	JC-340/350 JC-360/360MC	1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

NetCopyVarToReg Does Not Work Properly in Conjunction With Repetitions

Symptoms

If automatic repetitions have been enabled for networking via JetIP and such repetitions actually do occur, the function `NetCopyVarToReg()` enters wrong values into registers.

Affected Versions/Revisions

The following versions/revisions are affected by this bug:

OS version	JC-340/350	< 1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Remedy/Workaround

Do not enable automatic repetitions for `NetCopyVarToReg()`, but handle occurring errors by appropriately programming your application program.

Fixed Versions/Revisions

Starting from the following versions/revisions this bug has been fixed:

OS version	JC-340/350	1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Wrong Value in Task Status Register if a Cyclic Task is Aborted

Symptoms

When the cycle time monitoring of the OS aborts a cyclic task, it fails to reset bit 1 in the task status register.

Affected Versions/Revisions

The following versions/revisions are affected by this bug:

OS version	JC-340/350 JC-360/360MC	< 1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Remedy/Workaround

There is no remedy/workaround for affected versions/revisions.

Fixed Versions/Revisions

Starting from the following versions/revisions this bug has been fixed:

OS version	JC-340/350 JC-360/360MC	1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Wrong Values in Registers of CANopen® Modules With Profile DS402

Symptoms

The contents in several registers of CANopen® modules with profile DS402 are incorrectly displayed:

Register number	Error
200007x07, 200007x12, 200007x50, 200007x63, 200007x64	Negative values are incorrectly displayed.
200007x57, 200007x58	An incorrect value or an error is displayed.

Affected Versions/Revisions

The following versions/revisions are affected by this bug:

OS version	JC-340/JC-350	< 1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Remedy/Workaround

There is no remedy/workaround for affected versions/revisions.

Fixed Versions/Revisions

Starting from the following versions/revisions this bug has been fixed:

OS version	JC-340/JC-350	1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Incorrect Formatting of CANopen® Application Registers

Symptoms

If the CANopen® device does not send format information, the CANopen® application registers 200.007.x65 through 200.007.x84 are not displayed in the format set in register 200.002.088.

Affected Versions/Revisions

The following versions/revisions are affected by this bug:

OS version	JC-340/JC-350	< 1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	

Remedy/Workaround

There is no remedy/workaround for affected versions/revisions.

Fixed Versions/Revisions

Starting from the following versions/revisions this bug has been fixed:

OS version	JC-340/JC-350	1.16.0.00
Hardware revision	not relevant	
Configuration or operating mode	not relevant	