

JC-310-JM

Versionsupdate von V. 1.22 auf V. 1.28

Version 1.00

September 2017 / Printed in Germany

Dieses Dokument hat die Jetter AG mit der gebotenen Sorgfalt und basierend auf dem ihr bekannten Stand der Technik erstellt.

Bei Änderungen, Weiterentwicklungen oder Erweiterungen bereits zur Verfügung gestellter Produkte wird ein überarbeitetes Dokument nur beigelegt, sofern dies gesetzlich vorgeschrieben oder von der Jetter AG für sinnvoll erachtet wird. Die Jetter AG übernimmt keine Haftung und Verantwortung für inhaltliche oder formale Fehler, fehlende Aktualisierungen sowie daraus eventuell entstehende Schäden oder Nachteile.

Die im Dokument aufgeführten Logos, Bezeichnungen und Produktnamen sind geschützte Marken der Jetter AG, der mit ihr verbundenen Unternehmen oder anderer Inhaber und dürfen nicht ohne Einwilligung des jeweiligen Inhabers verwendet werden.

Inhaltsverzeichnis

1	Einleitung	4
	Betriebssystemupdate	5
	JC-310-JM Versionsupdate Übersicht	6
2	Erweiterungen	7
2.1	Diverse Funktionserweiterungen und Änderungen	8
	Taskcontinue startet einen Anwendungstask	9
	Neue Funktion FileEnd()	10
	Neue Funktionen DirLister	11
	Neue Funktionen BitSetReg() und BitClearReg()	12
	STX-Speichernutzung	13
	Mehr Verbindungen beim JetIP/TCP-Server	14
	Erweitertes Fehlerregister 200009	15
	NetConsistency kopiert die Konfigurationsdateien	16
2.2	Startverzögerungszeit	17
	Startverzögerungszeit einstellen	18
2.3	Jetter-Ethernet-Systembus	19
2.3.1	Funktion NetConsistency	21
	Funktion NetConsistency	23
	Zuweisung der Netzwerkparameter in Abhängigkeit von der GNN und Übertragung der Parameter- und Konfigurationsdaten	26
	JetIPScan in der JetControl aktivieren und deaktivieren	33
	Zeitlicher Ablauf beim Systemhochlauf	34
	Registerbeschreibung vom NetConsistency-Basistreiber	35
	Registerbeschreibung der NetConsistency-Instanz	43
	Fehlerauswertung bei NetConsistency	44
2.3.2	JetIPScan-Registerbeschreibung	46
	Registernummern	47
	Globaler Status - Registerbeschreibung	48
	Warnungen und Fehler - Registerbeschreibung	51
	Konfiguration - Registerbeschreibung	55
2.4	Freiprogrammierbare IP-Schnittstelle	57
2.4.1	Programmierung	59
	Initialisieren der freiprogrammierbaren IP-Schnittstelle	60
	Verbindung öffnen	61
	Daten senden	65
	Daten empfangen	67
	Verbindung schließen	70
2.4.2	Register	71
	Registernummern	72
	Registerbeschreibung	73
3	Beseitigte Software-Bugs	76
	Ungültiges Verbindungs-Handle bei der freiprogrammierbaren IP-Schnittstelle	77
	Lange Schlüsselnamen führen zum Absturz	78
	Umbenennen einer Datei mit maximaler Namenslänge führt zum Absturz	79
	Eingangswerte eines Netzwerkteilnehmers sind eingefroren	80
	Inkonsistente Register bei der IP-Konfiguration	81

1 Einleitung

Einleitung

Dieses Kapitel zeigt eine Historie der Betriebssystemversionen.

Betriebssystemupdate - wozu?

Ein Betriebssystemupdate bietet Ihnen die folgenden Möglichkeiten:

- Die Funktionalität erweitern
 - Software-Bugs beheben
 - Einen bestimmten Betriebssystemstand übertragen, z. B. nach Freigabe einer Betriebssystemversion
-

Inhalt

Thema	Seite
Betriebssystemupdate	5
JC-310-JM Versionsupdate Übersicht	6

Betriebssystemupdate

OS-Datei zum Betriebssystemupdate

Zum Betriebssystemupdate benötigen Sie die folgende Datei:

OS-Datei	Beschreibung
JC-310-JM_1.28.0.00.os	Betriebssystemdatei für JC-310-JM mit der Version 1.28

Download eines Betriebssystems

Die Jetter AG stellt die Betriebssysteme auf ihrer **Homepage** <https://www.jetter.de/downloads.html> zur Verfügung. Sie finden die Betriebssystemdateien zum Download beim jeweiligen Produkt.

Betriebssystemupdate mit JetSym

Führen Sie zum Betriebssystemupdate folgende Schritte aus.

Schritt	Vorgehen
1	Download der OS-Datei von www.jetter.de .
2	Herstellung einer Verbindung zwischen PC und Steuerung.
3	In JetSym: Ausführen des Menüpunkts Build -> Betriebssystemupdate oder Klicken auf die Schaltfläche OS-Update im CPU-Fenster des Hardware-Managers.
4	Auswahl der OS-Datei.
5	Start des Betriebssystemupdates mit OK.
6	Ergebnis: Nach Power Off/Power On startet das neue Betriebssystem.

Mindestvoraussetzungen

Für die Programmierung des JC-310-JM ab Version 1.28 ist JetSym ab Version 5.3.0 erforderlich.

JC-310-JM Versionsupdate Übersicht

V 1.28

Die folgende Tabelle zeigt eine Übersicht über neu hinzugekommene Funktionen und die beseitigten Software-Bugs der Betriebssystemversion 1.28:

Funktion	Neu	Bug
Anwendungsprogramm:		
Taskbearbeitung (Systemkommando 170/171)	✓	
Taskcontinue	✓	
Speicherverwaltung	✓	
Neue Funktionen	✓	
Ethernet-Systembus:		
NetConsistency	✓	
Kommunikation:		
STX-Debug-Server	✓	✓
JetIP-Server	✓	
NetConsistency	✓	
Dateisystem:		
Rename		✓
Freiprogrammierbare IP-Schnittstelle:		
Puffer-Verwaltung	✓	
Senden/Empfangen	✓	✓
System:		
Startverzögerung	✓	
IP-Konfiguration	✓	
Fehleranzeige	✓	

2 Erweiterungen

Einleitung

Die Entwicklung der Jetter AG erweitert ständig die Funktionen der Steuerung JC-310-JM. Durch ein Betriebssystemupdate lässt sich die Funktionalität der Steuerung um die neuen Funktionen erweitern. Sie benötigen dazu ...

- eine OS-Datei
 - das Softwaretool JetSym
 - eine Verbindung zwischen dem PC und der Steuerung
-

Inhalt

Thema	Seite
Diverse Funktionserweiterungen und Änderungen	8

2.1 Diverse Funktionserweiterungen und Änderungen

Einleitung

Dieses Kapitel beschreibt diverse Funktionserweiterungen und Änderungen.

Inhalt

Thema	Seite
Taskcontinue startet einen Anwendungstask	9
Neue Funktion FileEnd()	10
Neue Funktionen DirLister	11
Neue Funktionen BitSetReg() und BitClearReg()	12
STX-Speichernutzung	13
Mehr Verbindungen beim JetIP/TCP-Server	14
Erweitertes Fehlerregister 200009	15
NetConsistency kopiert die Konfigurationsdateien	16
Startverzögerungszeit	17
Jetter-Ethernet-Systembus	19
Freiprogrammierbare IP-Schnittstelle	57

Taskcontinue startet einen Anwendungstask

Einleitung	Die Befehle <code>Taskcontinue</code> und <code>TaskContinueById()</code> setzen einen Anwendungstask fort, der zuvor mit <code>Taskbreak</code> oder <code>TaskBreakById()</code> gestoppt wurde.
Bisherige Funktion	Ein Task, der seit dem Programmstart noch nicht gelaufen ist (kein Attribut <code>Autorun</code>), kann mit <code>Taskcontinue</code> oder <code>TaskContinueById()</code> nicht gestartet werden.
Neue Funktion	Ein Task, der seit dem Programmstart noch nicht gelaufen ist, kann mit <code>Taskcontinue</code> oder <code>TaskContinueById()</code> gestartet werden.
Grund der Änderung	Damit die Funktionalität zu anderen Steuerungsfamilien angeglichen ist.

Neue Funktion FileEnd()

Einleitung	Ab dieser Betriebssystemversion der Steuerung JC-310-JM wird die STX-Funktion <code>FileEnd()</code> unterstützt.
Voraussetzungen	Für die Verwendung dieser Funktionen ist die Programmierumgebung JetSym in der Version 5.3 oder höher erforderlich.
Deklaration	<code>Function FileEnd(Ref F:File):Int</code>
Verweis	Eine detaillierte Beschreibung des Befehls und seiner Anwendung finden Sie in der Online-Hilfe von JetSym.

Neue Funktionen DirLister

Einleitung	Ab dieser Betriebssystemversion der Steuerung JC-310-JM werden die STX-Funktionen des DirListers unterstützt.
Voraussetzungen	Für die Verwendung dieser Funktionen ist die Programmierumgebung JetSym in der Version 5.3 oder höher erforderlich.
Deklaration	<pre>Type DirKind : Enum (Files = 0, Folders = 1); End_Type Function DirListPath(enKind: DirKind, Const Ref strPath : String, Const Ref strFilter : String := '*.*') : Bool; Function DirListGetEntry(Ref strEntry : String) : Bool; Function DirListClose();</pre>
Verweis	Eine detaillierte Beschreibung der Befehle und ihrer Anwendung finden Sie in der Online-Hilfe von JetSym.

Neue Funktionen BitSetReg() und BitClearReg()

Einleitung	Ab dieser Betriebssystemversion der Steuerung JC-310-JM werden die STX-Funktionen <code>BitSetReg()</code> und <code>BitClearReg()</code> unterstützt.
Voraussetzungen	Für die Verwendung dieser Funktionen ist die Programmierumgebung JetSym in der Version 5.3 oder höher erforderlich.
Deklaration	<pre>Function BitSetReg(RegNr:Int, BitNr:Int); Function BitClearReg(RegNr:Int, BitNr:Int);</pre>
Verweis	Eine detaillierte Beschreibung der Befehle und ihrer Anwendung finden Sie in der Online-Hilfe von JetSym.

STX-Speichernutzung

Einleitung Ab dieser Betriebssystemversion der Steuerung JC-310-JM wird die Anzeige der Speichernutzung von JetSym STX unterstützt.

Voraussetzungen Für die Anzeige der Speichernutzung ist die Programmierumgebung JetSym in der Version 5.3.1 oder höher erforderlich.

Registerübersicht Aus diesen Registern kann die Speichernutzung des Anwendungsprogramms in Bytes ausgelesen werden. Im CPU-Fenster des Hardware-Managers von JetSym finden Sie eine grafische Darstellung der Werte.

Register	Beschreibung
R 211010	Gesamter Speicher: Insgesamt
R 211011	Gesamter Speicher: Verwendet
R 211012	Gesamter Speicher: Frei
R 211013	Systemspeicher: Insgesamt
R 211014	Systemspeicher: Verwendet
R 211015	Systemspeicher: Frei
R 211016	Anwenderspeicher: Insgesamt
R 211017	Anwenderspeicher: Verwendet
R 211018	Anwenderspeicher: Frei
R 211019	Verwendeter Speicher: Programm
R 211020	Verwendeter Speicher: Daten
R 211021	Verwendeter Speicher: Konstanten
R 211022	Verwendeter Speicher: Stack
R 211023	Verwendeter Speicher: JIT-Compiler
R 211024	Verwendeter Speicher: System

Verweis Eine detaillierte Beschreibung der Anzeige der STX-Speichernutzung finden Sie in der Online-Hilfe von JetSym.

Mehr Verbindungen beim JetIP/TCP-Server

Bisherige technische Daten

Parameter	Beschreibung
Anzahl Verbindungen	4

Neue technische Daten

Parameter	Beschreibung
Anzahl Verbindungen	8

Grund der Änderung

Es können mehr Verbindungen gleichzeitig geöffnet sein.

Erweitertes Fehlerregister 200009

R 200009

Die in dieser Version neu hinzugekommenen Fehlerbits sind grau unterlegt:

Bedeutung der Bits

Bit 3	Fehler in Datei Modconfig.da
Bit 5	Interner fataler Fehler der Ausführungseinheit des Anwendungsprogramms
Bit 10	Fehlermeldung von Teilnehmer am Jetter-Ethernet-Systembus
Bit 12	Fehlermeldung von JetIPScan
Bit 16	Fehlermeldung von NetConsistency
Bit 20	Interner Fehler der Speicherverwaltung des Betriebssystems
Bit 21	Interner Fehler der Speicherverwaltung des Anwendungsprogramms
Bit 22	Zum Boot-Zeitpunkt ist der System-Logger aktiv (Register 209700 = 213)
Bit 24	IP-Adresskonflikt erkannt

Modulregistereigenschaften

Zugriff	Lesen
Wert nach Reset	0

NetConsistency kopiert die Konfigurationsdateien

Bisherige Funktion	Die Funktion NetConsistency prüft die IP-Einstellungen der konfigurierten Netzwerkteilnehmer und stellt sie gegebenenfalls ein.
Neue Funktion	Die Funktion NetConsistency kopiert zusätzlich die Konfigurations- und Parameterdateien der konfigurierten Netzwerkteilnehmer und startet sie neu.
Einschränkung	Die Netzwerkteilnehmer müssen sternförmig vernetzt sein.

2.2 Startverzögerungszeit

Einleitung	Das Gerät hat ein Register, in das Sie eine Verzögerungszeit eintragen können.				
Verwendung	Der Boot-Vorgang des Geräts wird um die eingetragene Verzögerungszeit verzögert.				
Inhalt					
	<table><thead><tr><th>Thema</th><th>Seite</th></tr></thead><tbody><tr><td>Startverzögerungszeit einstellen.....</td><td>18</td></tr></tbody></table>	Thema	Seite	Startverzögerungszeit einstellen.....	18
Thema	Seite				
Startverzögerungszeit einstellen.....	18				

Startverzögerungszeit einstellen

Einleitung

Wenn andere Geräte am Bus eine längere Hochlaufzeit haben, ist das Verzögern des Boot-Vorgangs erforderlich.

Verzögerungszeit einstellen

Um die Verzögerungszeit einzustellen, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Geben Sie das Passwort ein. Dazu beschreiben Sie das R 202970 mit dem Wert 1112502132 (0x424f6f74).
3	Beschreiben Sie das Register 202971 mit der gewünschten Verzögerungszeit in 100-ms-Schritten.

Ergebnis: Beim nächsten Booten ist der Boot-Vorgang vor der Initialisierung des JX2- und JX3-Systembusses um die eingestellte Startverzögerungszeit verzögert.

R 202970

Passwortregister

Schreiben Sie in dieses Register das 1112502132 (0x424F6F74). Anschließend beschreiben Sie das Register der Startverzögerungszeit mit dem gewünschten Wert. Jetzt setzt die Steuerung den Wert dieses Registers auf 0.

Registereigenschaften

Wert	1112502132 (0x424F6F74)
------	-------------------------

R 202971

Startverzögerungszeit

Schreiben Sie in dieses Register die Verzögerungszeit in Vielfachen von 100 Millisekunden.

Registereigenschaften

Werte	0 (aus) ... 3.000 (300 Sekunden)
-------	----------------------------------

Wert nach Reset	Wie zuvor beschrieben (remanent)
-----------------	----------------------------------

Ablauf

- Die Steuerung führt die Startverzögerung nur aus, wenn der Schalter S11 in Stellung *RUN* ist.
- Die Startverzögerung wird abgebrochen, indem die Schalterstellung *RUN* verlassen wird.

Anzeige

- Die LED **D1** blinkt während der ersten Hälfte der Startverzögerungszeit langsam (ca. 1 Hz).
- Die LED **D1** blinkt während der zweiten Hälfte der Startverzögerungszeit schnell (ca. 4 Hz).

2.3 Jetter-Ethernet-Systembus

Einleitung

Der Jetter-Ethernet-Systembus setzt auf TCP, UDP/IP auf und kann somit parallel zu anderen TCP-, UDP/IP-Protokollen genutzt werden.

Er ist für den Datenaustausch über das Standard-Ethernet zwischen folgenden Geräten konzipiert:

- Programmiergerät
- Steuerungen
- Busknoten
- Kommunikationsmodule

Datenaustausch

Der Jetter-Ethernet-Systembus unterscheidet zwischen dem zyklischen und dem azyklischen Austausch der Daten zwischen den Kommunikationsteilnehmern. Beide Arten des Datenaustausches können gleichzeitig innerhalb eines Netzwerks ausgeführt werden.

Datenaustausch	Eigenschaften
Zyklisch	<ul style="list-style-type: none"> ▪ Architektur: Publish/Subscribe ▪ Teilnehmer: Steuerungen, Busknoten und Kommunikationsmodulen ▪ Zugriff: Automatisch vom Betriebssystem ▪ Zugriffszeit: Schnell, deterministisch ▪ Daten: Register, Ein-/Ausgänge ▪ Konfiguration: Hardware-Manager in JetSym ▪ Reichweite: Subnetz
Azyklisch	<ul style="list-style-type: none"> ▪ Architektur: Client/Server ▪ Client: PC und Steuerungen ▪ Server: PC, Steuerungen, Busknoten und Kommunikationsmodule ▪ Daten: Z. B. Register, Ein-/Ausgänge, STX-Variablen, Anwendungsprogramm ▪ Zugriff: PC oder Anwendungsprogramm ▪ Zugriffszeit: Abhängig von der Reaktionszeit des Servers ▪ Konfiguration: Nur bei der Verwendung von Netzwerkregistern ▪ Reichweite: Weltweit

2 Erweiterungen

Mindestanforderungen

Das Gerät wird in einem System bestehend aus verschiedenen Komponenten der Jetter AG betrieben. Um eine einwandfreie Zusammenarbeit der Komponenten zu gewährleisten, müssen die verwendeten Betriebssysteme und die Programmiersoftware JetSym mindestens die unten aufgeführten Versionsnummern haben.

Komponente	Ab Version
JC-310-JM	1.22.0.00
JC-340	1.22.0.00
JC-350	1.22.0.00
JC-360	1.22.0.00
JC-360MC	1.22.0.00
JC-365	1.26.0.00
JC-365MC	1.26.0.00
JC-440(MC)	1.02.0.00
JC-940MC	1.06.0.20
JC-945MC	1.01.0.00
JX3-BN-ETH	1.18.0.02
JX3-COM-EIPA	1.01.0.00
JX3-COM-PND	1.03.0.06
JM-200-ETH	1.22.0.00
JetSym	5.1.2

Inhalt

Thema	Seite
Funktion NetConsistency	21
JetIPScan-Registerbeschreibung	46

2.3.1 Funktion NetConsistency

Ziel	Das Ziel von NetConsistency ist die automatisierte Prüfung der IST-Systemeigenschaften gegenüber den SOLL-Systemeigenschaften von Netzwerkteilnehmern. Wenn die IST-Systemeigenschaften nicht den SOLL-Systemeigenschaften entsprechen, wird das System automatisch an den entsprechenden Stellen durch die SOLL-Systemeigenschaften korrigiert.
Anwendung	<p>Der Anwender hat durch NetConsistency die folgenden Möglichkeiten:</p> <ul style="list-style-type: none">▪ Eine defekte Systemkomponente, einen Netzwerkknoten, durch eine einfache Anpassung an der neuen Systemkomponente in einer projektierten Anlage auszutauschen. Die JetControl, der NetConsistency-Master, konfiguriert automatisch die neue Systemkomponente mit allen Informationen der bisherigen Systemkomponente.▪ Eine bestehende Anlage auf einfache Weise updaten. Ein Download der neuen Systemeigenschaften auf die JetControl, dem NetConsistency-Master, ist erforderlich. Die JetControl erkennt automatisch den Unterschied zwischen der bisherigen und neuen Systemkonfiguration und korrigiert die entsprechenden Stellen durch die neuen Systemeigenschaften.
Systemeigenschaften	<p>Systemeigenschaften sind:</p> <ul style="list-style-type: none">▪ Netzwerkparameter (IP-Adresse, Subnetzmaske, Default-Gateway)▪ Parameterdaten▪ Konfigurationsdaten
Konfigurationsdaten	Der Hardware-Manager von JetSym erzeugt die Konfigurations- und Parameterdaten. Der Hardware-Manager überträgt die Daten mit Hilfe der Funktion Vergleichen und Downloaden auf die JetControl.
NetConsistency-Master	Die Funktionalität NetConsistency stellt ein im System definierten NetConsistency-Master bereit. Der NetConsistency-Master kann nur eine JetControl sein.

2 Erweiterungen

Voraussetzungen

Die Nutzung von NetConsistency setzt folgendes voraus:

- JetSym ab V 5.1.0
- Mindestens ein NetConsistency-Master:

Produkt	Ab Version
JC-940MC	1.05.0.08 1.06.6.01 (Test-OS)
JC-945MC	1.01.0.00
JC-440(MC)	1.02.0.00
JC-340, JC-350	1.23.0.04
JM-310-JM	1.28.0.00
JC-360(MC)	1.28.0.00
JC-365(MC)	1.28.0.00

- Mindestens einer bis max. 64 NetConsistency-Slaves:

Produkt	Ab Version
JC-310-JM	1.22.0.00
JM-200-ETH	1.22.0.00
Ethernet-Achse JM-xxx (JM-2xx-OEM)	2.07.0.37
Ethernet-Achse MC-JM-xxx (JM-2xx-OEM)	2.07.0.37
JX3-BN-ETH	1.18.0.02
JX3-COM-EIPA	1.01.0.00
JX3-COM-PND	1.03.0.06

Inhalt

Thema	Seite
Funktion NetConsistency	23
Zuweisung der Netzwerkparameter in Abhängigkeit von der GNN und Übertragung der Parameter- und Konfigurationsdaten	26
JetIPScan in der JetControl aktivieren und deaktivieren	33
Zeitlicher Ablauf beim Systemhochlauf	34
Registerbeschreibung vom NetConsistency-Basistreiber	35
Registerbeschreibung der NetConsistency-Instanz	43
Fehlerauswertung bei NetConsistency	44

Funktion NetConsistency

Einschränkungen

- NetConsistency ist nur für den Jetter-Ethernet-Systembus verfügbar.
- Die Netzwerkteilnehmer müssen im selben Subnetz angeschlossen sein.
- Nur wenn JetIPScan aktiv ist, wird NetConsistency ausgeführt. JetIPScan ist aktiv, wenn Bit 2 von R 202962 gesetzt ist.
- Die JetControl führt die Funktionalität des NetConsistency nur einmalig beim Boot-Vorgang der JetControl, dem NetConsistency-Master, aus.

Funktion

Die Funktionalität des NetConsistency umfasst in der aktuellen Version die Systemeigenschaft:

- Netzwerkparameter
 - IP-Adresse
 - Subnetzmaske
 - Default-Gateway
- Parameterdaten
- Konfigurationsdaten

Netzwerkparameter

NetConsistency verwendet dazu JetIPScan. JetIPScan bietet die Funktionalität, den Busknoten die Netzwerkparameter über die GNN zuzuweisen.

Die Steuerung weist die Netzwerkparameter den Busknoten zu, die Sie im Hardware-Manager konfiguriert haben.

Als IP-Adresse weist die Steuerung die IP-Adresse den Busknoten zu, die Sie im Hardware-Manager konfiguriert haben.

Als Subnetzmaske weist die Steuerung ihre eigene Subnetzmaske den Busknoten zu.

Als Default-Gateway weist die Steuerung ihre eigene IP-Adresse oder ihr eigenes Default-Gateway den Busknoten zu:

Produkt	Zugewiesenes Default-Gateway
JC-940MC und JC-945MC wenn nur ETH1 konfiguriert ist	Default-Gateway der Steuerung
JC-940MC und JC-945MC wenn auch ETH2 und/oder ETH3 konfiguriert sind	IP-Adresse von ETH1 der Steuerung
JC-340, JC-350 und JC-440	Default-Gateway der Steuerung

Parameter- und Konfigurationsdaten

NetConsistency verwendet dazu FTP. Über FTP werden Parameter- und Konfigurationsdateien an die Busknoten übertragen.

Die Steuerung hält Parameter- und Konfigurationsdateien aller Busknoten in einem Backup-Verzeichnis. Für jeden Busknoten gibt es im Backup-Verzeichnis einen Ordner mit der Bezeichnung *NetNode* und der GNN am Namensende.

Beispiel: Dateisystem der Steuerung: `/SysConfig/Backup/NetNode002`

JetSym überträgt mit dem Vergleichen und Download alle Parameter- und Konfigurationsdateien in das Backup-Verzeichnis der Steuerung.

Zur Übernahme der Parameter- und Konfigurationsdaten erfolgt nach der Übertragung der Dateien ein Reboot auf die betroffenen Busknoten.

Busknoten **ohne** Parameter- und/oder Konfigurationsdateien erfahren keinen Reboot.

Folgende Produkte besitzen Parameter- und Konfigurationsdateien und erfahren einen Reboot:

Produkt	Ab Version
JX3-BN-ETH	V 1.18.0.02
JX3-COM-EIPA	V 1.01.0.00
JX3-COM-PND	V 1.03.0.06

Folgende Produkte besitzen **keine** Parameter- und Konfigurationsdateien und erfahren **keinen** Reboot:

Produkt	Ab Version
JC-310-JM	V 1.22.0.00
JM-200-ETH	V 1.22.0.00
Ethernet-Achse JM-xxx (JM-2xx-OEM)	V 2.07.0.37
Ethernet-Achse MC-JM-xxx (JM-2xx-OEM)	V 2.07.0.37

Systemhochlauf der Busknoten ohne remanente Speicherung der IP-Adresse

Die Busknoten verwenden beim Systemhochlauf die über den eigenen DIP-Schalter 1 bis 8 eingestellte GNN als feste IP-Adresse. Das gilt, bis der Busknoten die im Hardware-Manager konfigurierten Netzwerkparameter über die JetControl, den NetConsistency-Master, zugewiesen bekommt.

Eine remanente Speicherung der zuletzt zugewiesenen Netzwerkparameter über das NetConsistency ist nicht implementiert.

Als Empfehlung gilt: Verwenden Sie bei der Konfiguration der Busknoten im Hardware-Manager die GNN als niederwertigstes Byte der IP-Adresse.

Folgende Produkte speichern die IP-Adresse **nicht** remanent:

Produkt	Ab Version
JC-310-JM	V 1.22.0.00
JM-200-ETH	V 1.22.0.00
Ethernet-Achse JM-xxx (JM-2xx-OEM)	V 2.07.0.37
Ethernet-Achse MC-JM-xxx (JM-2xx-OEM)	V 2.07.0.37
JX3-COM-EIPA	V 1.01.0.00
JX3-COM-PND	V 1.03.0.06

Systemhochlauf der Busknoten mit remanenter Speicherung der IP-Adresse

Wenn die DIP-Schalter 9 bis 12 des JX3-BN-ETH in folgender Stellung sind, werden die Netzwerkparameter, die NetConsistency zugewiesen hat, remanent in der Datei **config.ini** der Busknoten gespeichert.

DIP-Schalter	Stellung
9	ON
10	OFF
11	OFF
12	OFF

Über die DIP-Schalter 1 bis 8 der Busknoten wird dessen GNN konfiguriert. Die Kodierung ist binär, z. B. nur Schalter 3 in Stellung ON bedeutet GNN = 4.

Die Busknoten verwenden beim Systemhochlauf die Netzwerkparameter, die in der Datei **/System/config.ini** stehen. Unmittelbar danach werden den Busknoten die im Hardware-Manager konfigurierten Netzwerkparameter über die JetControl, den NetConsistency-Master, zugewiesen. Wenn NetConsistency den Busknoten zuvor einmal die im Hardware-Manager konfigurierten Netzwerkparameter zugewiesen hat, verwenden die Busknoten beim Systemhochlauf bereits diese Netzwerkparameter.

Die zugewiesenen Netzwerkparameter speichern die Busknoten in der Datei **/System/config.ini** auf dem Dateisystem. Die bestehende Datei **/System/config.ini** wird dabei überschrieben.

Die Identifizierung der Busknoten im System, zur Zuweisung der im Hardware-Manager konfigurierten Netzwerkparameter, geschieht über die am DIP-Schalter der Busknoten eingestellten GNN.

Folgende Produkte speichern die IP-Adresse remanent:

Produkt	Ab Version
JX3-BN-ETH	V 1.18.0.02
JX3-COM-EIPA	V 1.05.0.02 (Beta-OS)
JX3-COM-PND	V 1.05.0.02 (Beta-OS)

Zuweisung der Netzwerkparameter in Abhängigkeit von der GNN und Übertragung der Parameter- und Konfigurationsdaten

Einleitung

Das NetConsistency stellt über das JetIPScan die Netzwerkparameter automatisch ein und überträgt per FTP die Parameter- und Konfigurationsdaten automatisch.

Das NetConsistency stellt über das JetIPScan die Netzwerkparameter bei den folgenden Geräten automatisch ein:

- Ethernet-Achsen JM-xxx (JM-2xx-OEM, JM-200-ETH, JC-310-JM)
- Ethernet-Achsen MC-JM-xxx (JM-2xx-OEM, JM-200-ETH, JC-310-JM)
- JX3-BN-ETH
- JX3-COM-EIPA
- JX3-COM-PND

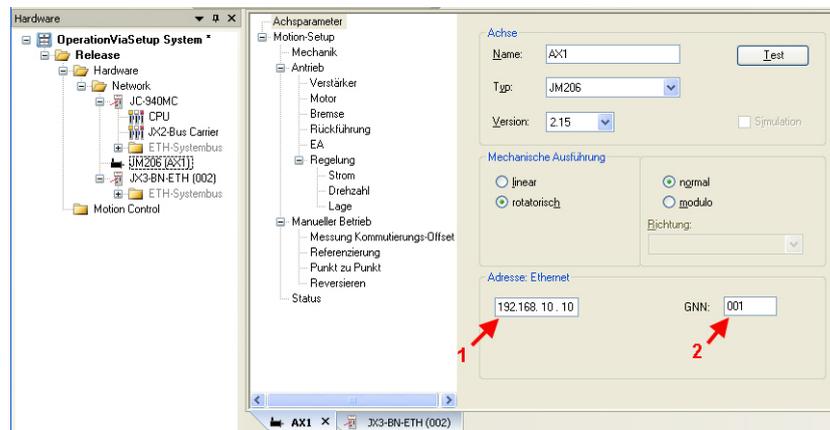
Das NetConsistency überträgt per FTP die Parameter- und Konfigurationsdateien auf die folgenden Geräte automatisch:

- JX3-BN-ETH
- JX3-COM-EIPA
- JX3-COM-PND

Automatisch bedeutet, dass Sie bei einem Austausch eines Netzwerkteilnehmers **nur** die GNN (Global Node Number) gleichbedeutend mit der Einstellung des DIP-Schalters vom bisherigen Netzwerkteilnehmer übernehmen müssen.

Alle weiteren Einstellungen erhält der Netzwerkteilnehmer über die JetControl. Das NetConsistency weist über das JetIPScan die Netzwerkparameter zu und überträgt per FTP die Parameter- und Konfigurationsdateien, die Sie für den Netzwerkteilnehmer im Hardware-Manager vergeben haben.

Einem JM-200 mit Option -ETH die IP-Adresse und GNN zuweisen



Schritt	Vorgehen
1	Stellen Sie die GNN am DIP-Schalter (Schalter 1 bis 8) des MC-JM-xxx oder JM-xxx ein.
2	Starten Sie JetSym.
3	Wählen Sie das Gerät MC-JM-xxx oder JM-xxx im Hardware-Manager aus.
4	Wählen Sie die Registerkarte Achsparameter aus.
5	Tragen Sie als Adresse Ethernet (1) die IP-Adresse ein. Tipp: Verwenden Sie die GNN als niederwertigstes Byte der IP-Adresse.
6	Tragen Sie als GNN (2) die Global Node Number für das Gerät ein. Die Zahl muss mit der Einstellung des DIP-Schalters am Gerät übereinstimmen.

Ergebnis: IP-Adresse und GNN sind dem Gerät zugewiesen.

DIP-Schalter beim MC-JM-xxx oder JM-xxx einstellen

Die Einstellung der DIP-Schalter 1 bis 8 verwendet der MC-JM-xxx oder JM-xxx als GNN. Die Kodierung ist binär.

Beispiele

GNN = 4: Schalter 3 ist ON. Alle anderen Schalter sind OFF.

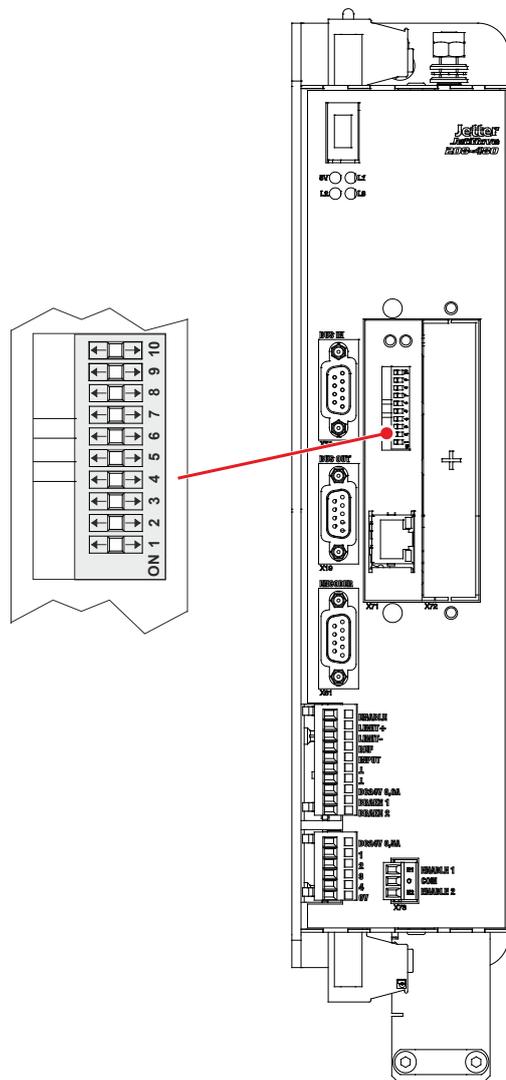
GNN = 5: Schalter 1 und 3 sind ON. Alle anderen Schalter sind OFF.

GNN = 8: Schalter 4 ist ON. Alle anderen Schalter sind OFF.

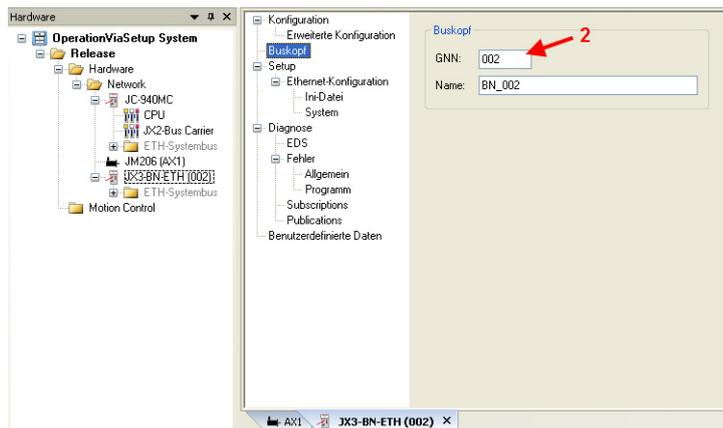
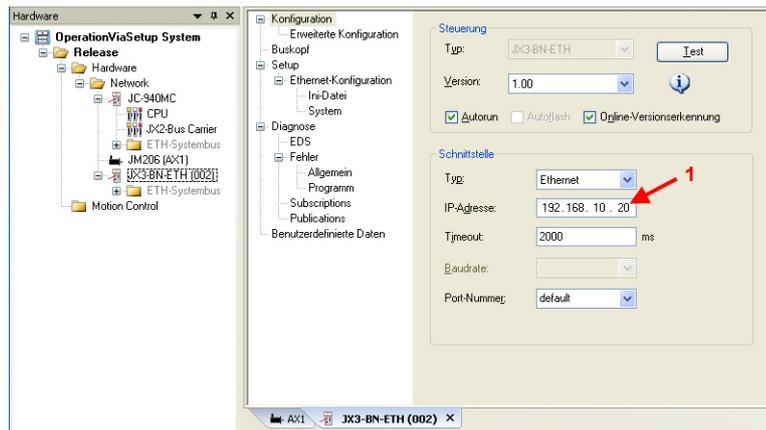
2 Erweiterungen

Position der DIP-Schalter beim MC-JM-xxx oder JM-xxx

Wenn beim digitalen Servoverstärker eine Ethernet-Schnittstelle integriert ist, steht ein 10-poliger DIP-Schalter zur Verfügung. Die folgende Abbildung zeigt die Position der DIP-Schalter.



Einem JX3-BN-ETH und JX3-COM-EIPA/PND die IP-Adresse und GNN zuweisen



Schritt	Vorgehen
1	Stellen Sie die GNN am DIP-Schalter (Schalter 1 bis 8) des JX3-BN-ETH oder JX3-COM-EIPA/PND ein.
2	Stellen Sie die Betriebsart GNN am DIP-Schalter (Schalter 9 bis 12) des JX3-BN-ETH oder JX3-COM-EIPA/PND ein.
3	Starten Sie JetSym.
4	Wählen Sie das Gerät JX3-BN-ETH oder JX3-COM-EIPA/PND im Hardware-Manager aus.
5	Wählen Sie die Registerkarte Konfiguration aus.
6	Tragen Sie als IP-Adresse (1) die IP-Adresse ein.
7	Wählen Sie die Registerkarte Busknoten aus.
8	Tragen Sie als GNN (2) die Global Node Number für das Gerät ein. Die Zahl muss mit der Einstellung des DIP-Schalters am Gerät übereinstimmen.

Ergebnis: IP-Adresse und GNN sind dem Gerät zugewiesen.

2 Erweiterungen

DIP-Schalter beim JX3-BN-ETH und JX3-COM-EIPA/-PND einstellen

Die Einstellung der DIP-Schalter 9 bis 12 aktiviert die remanente Speicherung der zugewiesenen Netzwerkparameter in der Datei **config.ini**.

Stellen Sie den DIP-Schalter 9 auf ON und die DIP-Schalter 10 bis 12 auf OFF.

Die Einstellung der DIP-Schalter 1 bis 8 konfiguriert die GNN. Die Kodierung ist binär.

Beispiele

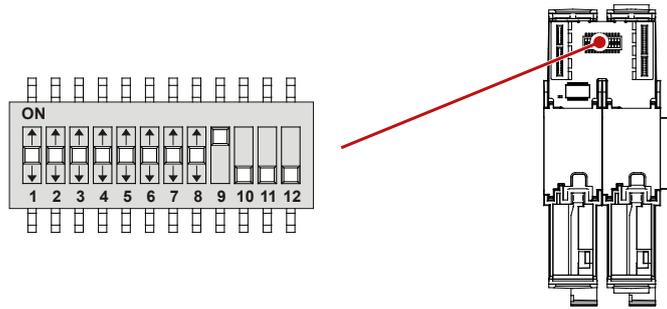
GNN = 4: Schalter 3 ist ON. Alle anderen Schalter sind OFF.

GNN = 5: Schalter 1 und 3 sind ON. Alle anderen Schalter sind OFF.

GNN = 8: Schalter 4 ist ON. Alle anderen Schalter sind OFF.

Position der DIP-Schalter beim JX3-BN-ETH und JX3-COM-EIPA/-PND

Die folgende Abbildung zeigt die Position der DIP-Schalter.



Netzwerk-Topologie

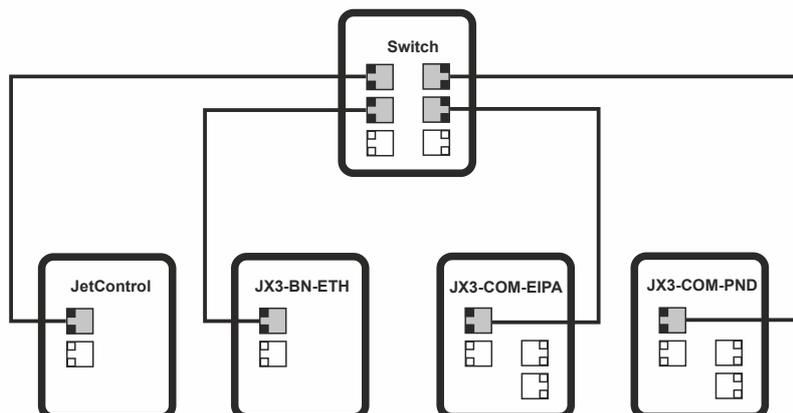
Bei der Verwendung des NetConsistency **müssen** die Netzwerkteilnehmer in einer Stern-Topologie, siehe Abbildung *Stern-topologie*, vernetzt sein.

Hintergrund

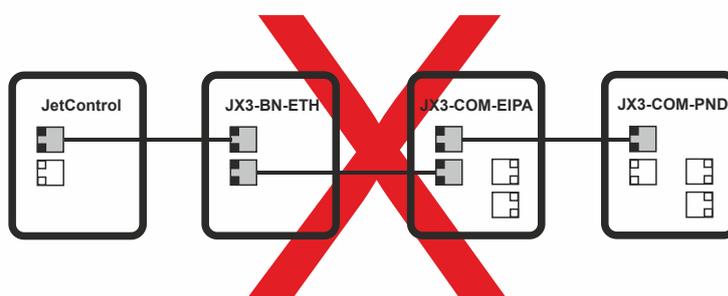
Nach der Übertragung der Parameter- und Konfigurationsdateien per FTP, erzeugt die Steuerung einen Reboot der betreffenden Busknoten.

Dabei erfährt auch der Ethernet-Switch auf den Busknoten einen Reboot und kann keine Netzwerktelegramme mehr weiterleiten. In der Linien-Topologie, siehe Abbildung *Linientopologie*, führt das dazu, dass z. B. der erste Busknoten JX3-BN-ETH in der Linien-Topologie von der JetControl das Reboot-Kommando erhält. Der nachfolgende Busknoten JX3-COM-EIPA aber kein Reboot-Kommando mehr von der Steuerung empfangen kann, weil der Ethernet-Switch des ersten Busknotens JX3-BN-ETH gerade neu bootet. Deshalb ist bei Verwendung von NetConsistency die Linientopologie nicht erlaubt.

Stern-Topologie



Linien-Topologie



Vergleichen und Downloaden

Wenn Sie im Hardware-Manager alles eingestellt haben, übertragen Sie die Einstellungen über **Vergleichen und Downloaden** auf die Systemteilnehmer. Dazu gibt es den folgenden Befehl im Hardware-Manager:

- Vergleichen und downloaden (rechte Maustaste auf **Release**)

Zugewiesene Netzwerkparameter

Beim Systemhochlauf weist die Steuerung den angeschlossenen Netzwerkteilnehmern die folgenden Netzwerkparameter zu:

- IP-Adresse
- Subnetzmaske
- Default-Gateway

IP-Adresse

Die Steuerung weist die im Hardware-Manager eingestellte IP-Adresse zu.

Subnetzmaske

Die Steuerung weist ihre eigene Subnetzmaske zu.

Default-Gateway

Der zugewiesene Default-Gateway ist vom Steuerungstyp abhängig:

Produkt	Zugewiesenes Default-Gateway	
JC-340, JC-350, JC-440MC	Default-Gateway der Steuerung.	
JC-940MC	Wenn dann ...
	... weder bei ETH2 noch bei ETH3 Netzwerkparameter konfiguriert sind,	... weist die Steuerung das Default-Gateway von ETH1 zu.
	... bei ETH2 oder bei ETH3 Netzwerkparameter konfiguriert sind,	... weist die Steuerung die IP-Adresse von ETH1 als Default-Gateway zu.
JC-945MC	Wenn dann ...
	... bei ETH3 keine Netzwerkparameter konfiguriert sind,	... weist die Steuerung das Default-Gateway von ETH1 zu.
	... bei ETH3 Netzwerkparameter konfiguriert sind,	... weist die Steuerung die IP-Adresse von ETH1 als Default-Gateway zu.

Anschließend überträgt die Steuerung die Parameter- und Konfigurationsdateien auf die Netzwerkteilnehmer und führt einen Reboot auf den betroffenen Netzwerkteilnehmer aus.

JetIPScan in der JetControl aktivieren und deaktivieren

Einleitung

JetIPScan müssen Sie über einen Eintrag in das Systemkommandoregister aktivieren. Die Einstellung ist remanent.

JetIPScan aktivieren

Um JetIPScan zu aktivieren, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Passwortregister 202960 mit dem Wert 1112502132 (0x424f6f74).
3	Beschreiben Sie das Systemkommandoregister 202961 mit dem Wert 331.
⇒	Bit 2 von R 202962 ist gesetzt und JetIPScan ist aktiviert.

JetIPScan deaktivieren

Um JetIPScan zu deaktivieren, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Passwortregister 202960 mit dem Wert 1112502132 (0x424f6f74).
3	Beschreiben Sie das Systemkommandoregister 202961 mit dem Wert 330.
⇒	Bit 2 von R 202962 ist gelöscht und JetIPScan ist deaktiviert.

Zeitlicher Ablauf beim Systemhochlauf

Zeitlicher Ablauf beim Systemhochlauf

Die folgende Tabelle zeigt den zeitlichen Ablauf beim Systemhochlauf:

Stufe	Beschreibung
1	In der Boot-Phase verwenden die Netzwerkteilnehmer mit remanenter Speicherung der IP-Adresse die IP-Adresse aus der Datei config.ini .
2	Während des Boot-Vorgangs der JetControl wird jedem Netzwerkteilnehmer bei der Ausführung der Funktionalität NetConsistency über JetIPScan eine Netzwerkkonfiguration (IP-Adresse, Subnetzmaske, Gateway-Adresse) zugewiesen. Den Teilnehmern, die Parameterdateien und/oder Konfigurationsdateien besitzen, werden über FTP die Dateien übertragen. Zur Übernahme der neuen Parameter- und Konfigurationsdaten erfolgt ein Reboot der Netzwerkteilnehmer.
3	Nach Ende des Boot-Vorgangs der JetControl und damit nach Ausführung der Funktionalität NetConsistency sind die Netzwerkteilnehmer unter den im Hardware-Manager konfigurierten Netzwerkkonfigurationen erreichbar und besitzen die im Hardware-Manager konfigurierten Parameter- und Konfigurationsdaten.

Zeitlicher Ablauf bei NetConsistency

NetConsistency durchläuft die folgenden Zustände in der Boot-Phase der JetControl.

Stufe	Beschreibung
1	Der Basistreiber ist initialisiert.
2	Eine Instanz ist initialisiert.
3	Die Funktionalität des NetConsistency wird ausgeführt.

Registerbeschreibung vom NetConsistency-Basistreiber

Registerübersicht

Register	Beschreibung
470000 ... 470008	Cookie
470009	Versionsnummer
470010	Status
470011	Kommando
470020	Maximal mögliche Anzahl an Instanzen
470021	Anzahl funktionsbereiter Instanzen
470030 ... 470035	Begrenzungen
470040 ... 470157	Fehlerauswertung

R 470000 ... R 470008

Cookie

Dieses Register zeigt den Beginn der NetConsistency-Register auf. Das vereinfacht die Orientierung.

Modulregistereigenschaften

Zugriff	Lesen
Wert nach Reset	NetConsistency
Datentyp	RegString

R 470009

Version von NetConsistency

R 470009 zeigt die Version von NetConsistency an.

Modulregistereigenschaften

Werte	IP#0.00.0.00 ... IP#9.99.9.99
Zugriff	Lesen
Wert nach Reset	Version von NetConsistency

R 470010

Statusregister

R 470010 zeigt den Status des NetConsistency-Basistreibers an.

Bedeutung der Bits

Bit 0	Fehler
0 =	kein Fehler
1 =	Fehler

Bit 2 Status der Initialisierung

0 = Basistreiber ist nicht initialisiert

1 = Basistreiber ist initialisiert

Modulregistereigenschaften

Zugriff Lesen

Wert nach Reset 0x00000004

R 470011**Kommandoregister**

Der Wert ist 0, weil es keine Kommandos gibt.

R 470020**Maximale Anzahl an Instanzen**

R 470020 zeigt die maximale Anzahl an NetConsistency-Instanzen an. Der aktuelle Wert ist immer 1.

Modulregistereigenschaften

Werte 1

Zugriff Lesen

Wert nach Reset 1

R 470021**Anzahl funktionsbereiter Instanzen**

R 470021 zeigt die Anzahl funktionsbereiter NetConsistency-Instanzen an.

Modulregistereigenschaften

Werte 0 ... 1

Zugriff Lesen

Wert nach Reset 1

R 470030

Maximale Anzahl an Fehlermeldungen für den Logger

R 470030 legt die maximale Anzahl an Fehlermeldungen fest, die NetConsistency an den Logger weiterleitet.

Modulregistereigenschaften

Werte	10
Zugriff	Lesen
Wert nach Reset	10

R 470031

Anzahl an den Logger weitergeleitete Fehlermeldungen

R 470031 zeigt die Anzahl Fehlermeldungen an, die NetConsistency an den Logger weitergeleitet hat.

Modulregistereigenschaften

Werte	0 ... 10
Zugriff	Lesen

R 470032

Maximale Anzahl an Warnungen für den Logger

R 470032 legt die maximale Anzahl an Warnungen fest, die NetConsistency an den Logger weiterleitet.

Modulregistereigenschaften

Werte	10
Zugriff	Lesen
Wert nach Reset	10

R 470033

Anzahl an den Logger weitergeleitete Warnungen

R 470033 zeigt die Anzahl Warnungen an, die NetConsistency an den Logger weitergeleitet hat.

Modulregistereigenschaften

Werte	0 ... 10
Zugriff	Lesen

R 470034

Maximale Anzahl an Fehlerhistorieneinträgen

R 470034 legt die maximale Anzahl an Fehlerhistorieneinträge fest.

Modulregistereigenschaften

Werte	10
Zugriff	Lesen
Wert nach Reset	10

R 470035

Anzahl an Einträgen in der Fehlerhistorie

R 470035 zeigt die Anzahl an Fehlermeldungen an, die NetConsistency in die Fehlerhistorie eingetragen hat.

Modulregistereigenschaften

Werte	0 ... 30
Zugriff	Lesen

R 470040

Fehlernummern

R 470040 zeigt die Fehlernummern an.

Fehlername	Fehlernummer
NoError	0
GroupFunction	-1
GroupCStandard	-2
GroupJetterFileSystem	-3
GroupJetterLogger	-4
GroupJetterOS	-5
GroupJetterParserXml	-6
GroupJetterPcom	-7
GroupUtility	-8
GroupJetIpScan	-9
Api	-100
Manager	-110
ManagerInit	-111
ManagerDeinit	-112
ManagerMultipleInit	-113
Instance	-120
InstanceInit	-121

Fehlername	Fehlernummer
InstanceDeinit	-122
StateMachine	-140
StateMachineInit	-141
StateMachineDeinit	-142
Error	-150
ErrorInit	-151
ErrorDeinit	-152
Warning	-160
WarningInit	-161
WarningDeinit	-162
Register	-170
RegisterInit	-171
RegisterDeinit	-172
Xml	-180
XmlInit	-181
XmlDeinit	-182
XmlInvalidGnn	-183
XmlInvalidIpAddress	-184
XmlTagNetConsistencyAttrVersion	-185
XmlTagNetNodesAttrCount	-186
XmlTagNetNodeAttrName	-187
XmlTagNetNodeAttrType	-188
XmlTagNetNodeAttrGnn	-189
XmlTagPcomAttrName	-190
XmlTagPcomAttrCommand	-191
XmlTagPcomAttrModuleId	-192
XmlTagPcomAttrTypeId	-193
XmlTagIpAddress	-194
XmlTagJetIPAttrPort	-195
XmlTagJx3SystembusAttrCrcEdsModuleCount	-196
XmlTagFilesAttrCount	-197
XmlTagFilesAttrCrc	-198
XmlTagFileAttrCrc	-199
XmlTagFileAttrPath	-200
XmlTagFileAttrName	-201
JetModuleReadReg	-300
JetModuleWriteReg	-301

Fehlername	Fehlernummer
Utility	-310
JetIPScan	-320
JetIPScanInit	-321
JetIPScanDeinit	-322
Processing	-330
ProcessingInit	-331
ProcessingDeinit	-332

Modulregistereigenschaften

Werte	$-2^{16} \dots 0$
Zugriff	Lesen

R 470041

Zeitpunkt des Fehlers in Millisekunden

R 470041 zeigt den Zeitpunkt des Fehlers in Millisekunden an. Wenn die JetControl 50 Tage eingeschaltet ist, tritt ein Überlauf auf.

Modulregistereigenschaften

Werte	$0 \dots 2^{32} \text{ ms} = 0 \dots 50 \text{ Tage}$
Zugriff	Lesen

R 470042

Instanz, wo der Fehler aufgetreten ist

R 470042 zeigt an, bei welcher Instanz der Fehler aufgetreten ist. Aktuell ist nur eine Instanz möglich.

Modulregistereigenschaften

Werte	0: Erste Instanz
Zugriff	Lesen

R 470043

Anzahl an Fehlerparametern

R 470043 zeigt die Anzahl an Fehlerparametern an.

Modulregistereigenschaften

Werte	$0 \dots 5$
Zugriff	Lesen

R 470044**Fehlerparameter 1**

R 470044 zeigt den Fehlerparameter 1 an. Der Wert ist nur gültig, wenn $R\ 470043 \geq 1$.

Modulregistereigenschaften

Werte 0 ... 2^{32}

Zugriff Lesen

R 470045**Fehlerparameter 2**

R 470045 zeigt den Fehlerparameter 2 an. Der Wert ist nur gültig, wenn $R\ 470043 \geq 2$.

Modulregistereigenschaften

Werte 0 ... 2^{32}

Zugriff Lesen

R 470046**Fehlerparameter 3**

R 470046 zeigt den Fehlerparameter 3 an. Der Wert ist nur gültig, wenn $R\ 470043 \geq 3$.

Modulregistereigenschaften

Werte 0 ... 2^{32}

Zugriff Lesen

R 470047**Fehlerparameter 4**

R 470047 zeigt den Fehlerparameter 4 an. Der Wert ist nur gültig, wenn $R\ 470043 \geq 4$.

Modulregistereigenschaften

Werte 0 ... 2^{32}

Zugriff Lesen

R 470048**Fehlerparameter 5**

R 470048 zeigt den Fehlerparameter 5 an. Der Wert ist nur gültig, wenn $R\ 470043 = 5$.

2 Erweiterungen

Modulregistereigenschaften

Werte	0 ... 2^{32}
-------	----------------

Zugriff	Lesen
---------	-------

R 470049

Anzahl Buchstaben der Fehlermeldung

R 470049 zeigt die Anzahl an Buchstaben der Fehlermeldung an. Die Fehlermeldung ist gespeichert in den Registern 470050 ... 470157.

Modulregistereigenschaften

Werte	0 ... 300
-------	-----------

Zugriff	Lesen
---------	-------

R 470050 ... R 470157

Text der Fehlermeldung

Diese Register enthalten den Text der Fehlermeldung.

Modulregistereigenschaften

Zugriff	Lesen
---------	-------

Wert nach Reset	""
-----------------	----

Datentyp	RegString
----------	-----------

Registerbeschreibung der NetConsistency-Instanz

Registerübersicht

Register	Beschreibung
471010	Status
471011	Kommando

R 471010

Statusregister

R 470010 zeigt den Status der ersten NetConsistency-Instanz an.

Bedeutung der Bits

Bit 0 Fehler

0 = kein Fehler

1 = Fehler

Bit 2 Status der Initialisierung

0 = Erste Instanz ist nicht initialisiert

1 = Erste Instanz ist initialisiert

Bit 3 Status der Ausführung

0 = keine Ausführung

1 = Ausführung läuft

Modulregistereigenschaften

Zugriff Lesen

Wert nach Reset 0x00000004

R 471011

Kommandoregister

Der Wert ist 0, weil es keine Kommandos gibt.

Fehlerauswertung bei NetConsistency

Möglichkeiten der Fehlerausgabe

Die folgenden Möglichkeiten der Fehlerausgabe gibt es:

- Über den Logger von NetConsistency und JetIPScan
- Über das erweiterte Fehlerregister R 200009
- Über das Fehlernummerregister R 200051 von JetIPScan
- Über das Fehlernummerregister R 200061 von NetConsistency

R 200009

Erweitertes Fehlerregister

R 200009 ist ein bitkodiertes Register.

Bedeutung der Bits

Bit 12 Fehlermeldung von JetIPScan

- 0 = kein Fehler
- 1 = JetIPScan hat einen Fehler gemeldet.
Die Fehlernummer steht in R 200051.

Bit 16 Fehlermeldung von NetConsistency

- 0 = kein Fehler
- 1 = NetConsistency hat einen Fehler gemeldet.
Die Fehlernummer steht in R 200061 und R 470040.

Modulregistereigenschaften

Zugriff	Lesen
---------	-------

R 200051**Fehlernummern von JetIPScan**

R 200051 zeigt die Fehlernummern von JetIPScan an. Der Inhalt dieses Registers ist identisch mit dem MR 13 der Funktion JetIPScan.

Modulregistereigenschaften

Werte	0	Ohne Fehler oder Warnung
	5	Funktion durch Anwender abgebrochen
	1001	Die erste empfangene Antwort stimmt nicht mit Antwort 2 und 3 überein (siehe MR 101x)
	1002	Die zweite empfangene Antwort stimmt nicht mit Antwort 1 und 3 überein (siehe MR 102x)
	1003	Die dritte empfangene Antwort stimmt nicht mit Antwort 1 und 2 überein (siehe MR 103x)
	-1	Alle 3 Antworten sind unterschiedlich (siehe MR 100x)
	-2	Die IP-Einstellungen mindestens eines Teilnehmers konnte nicht geändert werden (siehe MR 140x)
	-3	Die Funktion JetIPScan wurde aufgerufen, obwohl sie bereits läuft
	-10	Die Länge der SOLL-Liste ist < 1 oder > 255 oder der Zeiger auf die Liste ist ungültig
	-11	Eine GNN der SOLL-Liste ist < 1 oder > 255 oder mehrfach vorhanden
	-20 ... -40	Interner Fehler
	-1001 ... -1199	Teilnehmer hat die falsche CtrlID oder CtrlIDopt gemeldet (siehe MR 110x)
	-2001 ... -2199	Teilnehmer hat sich nicht gemeldet (siehe MR 120x)
	-3001 ... -3199	Mehrere Teilnehmer mit der gleichen GNN haben sich gemeldet (siehe MR 130x)
Zugriff	Lesen	

R 200061**Fehlernummern von NetConsistency**

R 200061 zeigt die Fehlernummern von NetConsistency an, siehe R 470040.

Verwandte Themen

- **Registerbeschreibung vom NetConsistency-Basistreiber** (siehe Seite 35)
- **Registerbeschreibung von JetIPScan** (siehe Seite 46)

2.3.2 JetIPScan-Registerbeschreibung

Einleitung

Dieses Kapitel beschreibt die Register, aus denen die Statusinformationen der Funktionalität JetIPScan ausgelesen werden. Sie können die Register zu Debug- oder Diagnosezwecken verwenden. Weitere Funktionen, wie z. B. die Prüfung der Netzwerkkonfiguration, können hierüber nicht ausgelöst werden.

Inhalt

Thema	Seite
Registernummern	47
Globaler Status - Registerbeschreibung	48
Warnungen und Fehler - Registerbeschreibung	51
Konfiguration - Registerbeschreibung	55

Registernummern

Einleitung Die Statusinformationen erscheinen in den Registern eines zusammenhängenden Registerblocks. Die Basisregisternummer dieses Blocks ist steuerungsabhängig.

Registernummern

Basisregisternummer	Registernummern
520000	520000 ... 522999

Registernummer ermitteln

In diesem Kapitel sind jeweils nur die letzten vier Ziffern der Registernummer angegeben, z. B. MR 1499. Addieren Sie zu dieser Modulregisternummer die Basisregisternummer des jeweiligen Geräts, um die vollständige Registernummer, z. B. 521499, zu ermitteln.

Registerübersicht

Register	Beschreibung
MR 0 ... MR 13	Globaler Status
MR 1000 ... MR 1499	Warnungen und Fehler
MR 2000 ... MR 2399	SOLL- und IST-Konfiguration

Globaler Status - Registerbeschreibung

Einleitung

Mit Hilfe dieser Register lassen sich globale Statusinformationen auslesen.

MR 0

Summenstatus

Im MR 0 signalisiert die Steuerung bitkodiert eine Zusammenfassung der Statusmeldungen.

Bedeutung der Bits

Bit 0 Funktionsfreigabe

Dieses Bit entspricht Bit 2 des Systemstatusregisters 202962.

0 = JetIPScan-Client AUS

1 = JetIPScan-Client EIN

Bit 1 Sammelfehlermeldung

1 = MR 13 enthält einen Wert kleiner 0

Modulregistereigenschaften

Zugriff Lesen

Wert nach Reset Bit 0: Je nach Freigabestatus
Bit 1: 0

MR 10

Ausführungszustand

Entspricht dem Rückgabewert *State*.

Modulregistereigenschaften

Werte	0	Funktion läuft nicht Funktion ist beendet
	1	Warte auf eine Antwort der Netzwerkteilnehmer
	2	Sende ein Anfragetelegramm
	3	Prüfe die Antworten der Netzwerkteilnehmer
	4	Schreibe die Konfiguration der Netzwerkteilnehmer

Zugriff Lesen

MR 11

Anzahl der Durchläufe

Entspricht dem Rückgabewert *Count*.

Modulregistereigenschaften

Werte	0 ... 3	Anzahl der Durchläufe
Zugriff	Lesen	

MR 12

Anzahl der Änderungen

Entspricht dem Rückgabewert *Changed*.

Modulregistereigenschaften

Werte	0 ... 199	Anzahl der geänderten Netzwerkteilnehmer
Zugriff	Lesen	

MR 13

Funktionsergebnis

Entspricht dem Rückgabewert *Result* und dem Registerinhalt der globalen Fehlernummer 2000051. Dieses Register zeigt den Wert des letzten aufgetretenen Fehlers oder der letzten Warnung an. Werte größer Null zeigen Warnungen an. Werte kleiner Null sind Fehlermeldungen.

Modulregistereigenschaften

Werte	0	Ohne Fehler oder Warnung
	5	Funktion durch Anwendung abgebrochen
	1001	Die erste empfangene Antwort stimmt nicht mit Antwort 2 und 3 überein (siehe MR 101x)
	1002	Die zweite empfangene Antwort stimmt nicht mit Antwort 1 und 3 überein (siehe MR 102x)
	1003	Die dritte empfangene Antwort stimmt nicht mit Antwort 1 und 2 überein (siehe MR 103x)
	-1	Alle 3 Antworten sind unterschiedlich (siehe MR 100x)
	-2	Die IP-Einstellungen mindestens eines Teilnehmers konnte nicht geändert werden (siehe MR 140x)
	-3	Die Funktion JetIPScan wurde aufgerufen, obwohl sie bereits läuft
	-10	Die Länge der SOLL-Liste ist < 1 oder > 255 oder der Zeiger auf die Liste ist ungültig
	-11	Eine GNN der SOLL-Liste ist < 1 oder > 255 oder mehrfach vorhanden

2 Erweiterungen

Modulregistereigenschaften

Werte	-20 ... -40	Interner Fehler
	-1001 ... -1199	Teilnehmer hat die falsche CtrlID oder CtrlIDopt gemeldet (siehe MR 110x)
	-2001 ... -2199	Teilnehmer hat sich nicht gemeldet (siehe MR 120x)
	-3001 ... -3199	Mehrere Teilnehmer mit der gleichen GNN haben sich gemeldet (siehe MR 130x)
Zugriff	Lesen	

Warnungen und Fehler - Registerbeschreibung

Einleitung

Eine detaillierte Diagnose der aufgetretenen Warnungen und Fehler ist mit Hilfe dieser Register möglich.

Wenn während der Prüfung und Einstellung der IP-Einstellungen aller Netzwerkteilnehmer eine Warnung oder ein Fehler auftritt, setzt die Steuerung das entsprechende Bit in den folgend beschriebenen Registern. Dabei entspricht das Bit der GNN des Netzwerkteilnehmers.

Zwischen der GNN des Netzwerkteilnehmers und der Bitnummer gilt folgender Zusammenhang:

$$\text{Bitnummer} = \text{GNN} - 1$$

Da ein Register 32 Bit enthält, sind jeweils 7 aufeinander folgende Register zusammengefasst (siehe Tabelle).

Registerbit	GNN
Register.0	1
Register.31	32
(Register + 1).0	33
(Register + 1).31	64
(Register + 2).0	65
(Register + 2).31	96
(Register + 3).0	97
(Register + 3).31	128
(Register + 4).0	129
(Register + 4).31	160
(Register + 5).0	161
(Register + 5).31	192
(Register + 6).0	193
(Register + 6).6	199

MR 1000 ... 1006

Alle 3 Antworten sind unterschiedlich

Die Steuerung fragt dreimal die Netzwerkkonfiguration ab und vergleicht diese drei Antworten. Wenn alle drei Antworten unterschiedlich sind, setzt die Steuerung das entsprechende Bit in diesen Registern.

Bedeutung der Bits

Bit = 0 Kein Fehler

Bit = 1 Fehler

Modulregistereigenschaften

Bitnummer	GNN - 1
-----------	---------

Zugriff	Lesen
---------	-------

MR 1010 ... 1016

Antwort 1 ist unterschiedlich

Die Steuerung fragt dreimal die Netzwerkkonfiguration ab und vergleicht diese drei Antworten. Wenn die Antworten 2 und 3 gleich sind, die Antwort 1 aber unterschiedlich, setzt die Steuerung das entsprechende Bit in diesen Registern.

Bedeutung der Bits

Bit = 0	Keine Warnung
---------	---------------

Bit = 1	Warnung
---------	---------

Modulregistereigenschaften

Bitnummer	GNN - 1
-----------	---------

Zugriff	Lesen
---------	-------

MR 1020 ... 1026

Antwort 2 ist unterschiedlich

Die Steuerung fragt dreimal die Netzwerkkonfiguration ab und vergleicht diese drei Antworten. Wenn die Antworten 1 und 3 gleich sind, die Antwort 2 aber unterschiedlich, setzt die Steuerung das entsprechende Bit in diesen Registern.

Bedeutung der Bits

Bit = 0	Keine Warnung
---------	---------------

Bit = 1	Warnung
---------	---------

Modulregistereigenschaften

Bitnummer	GNN - 1
-----------	---------

Zugriff	Lesen
---------	-------

MR 1030 ... 1036**Antwort 3 ist unterschiedlich**

Die Steuerung fragt dreimal die Netzwerkkonfiguration ab und vergleicht diese drei Antworten. Wenn die Antworten 1 und 2 gleich sind, die Antwort 3 aber unterschiedlich, setzt die Steuerung das entsprechende Bit in diesen Registern.

Bedeutung der Bits

Bit = 0 Keine Warnung

Bit = 1 Warnung

Modulregistereigenschaften

Bitnummer GNN - 1

Zugriff Lesen

MR 1100 ... 1106**Falsche CtrlID oder CtrlIDopt**

Ein Teilnehmer mit der geforderten GNN hat sich gemeldet, die CtrlID oder die CtrlIDopt stimmen aber nicht überein.

Bedeutung der Bits

Bit = 0 Kein Fehler

Bit = 1 Fehler

Modulregistereigenschaften

Bitnummer GNN - 1

Zugriff Lesen

MR 1200 ... 1206**Teilnehmer hat sich nicht gemeldet**

Ein Teilnehmer mit der geforderten GNN hat sich nicht gemeldet.

Bedeutung der Bits

Bit = 0 Kein Fehler

Bit = 1 Fehler

Modulregistereigenschaften

Bitnummer GNN - 1

Zugriff Lesen

MR 1300 ... 1306

Mehrfachmeldung

Mehrere Teilnehmer haben sich mit derselben GNN gemeldet. Jeder Netzwerkteilnehmer muss aber eine eindeutige GNN haben.

Bedeutung der Bits

Bit = 0 Kein Fehler

Bit = 1 Fehler

Modulregistereigenschaften

Bitnummer GNN - 1

Zugriff Lesen

MR 1400 ... 1406

IP-Einstellung konnte nicht geändert werden

Wenn die IP-Einstellungen eines Netzwerkteilnehmers geändert wurden, prüft die Steuerung anschließend, ob der Netzwerkteilnehmer diese Änderungen übernommen hat.

Wenn der Netzwerkteilnehmer diese Änderungen nicht übernommen hat, setzt die Steuerung das entsprechende Bit in diesen Registern.

Bedeutung der Bits

Bit = 0 Kein Fehler

Bit = 1 Fehler

Modulregistereigenschaften

Bitnummer GNN - 1

Zugriff Lesen

Konfiguration - Registerbeschreibung

Einleitung

Mit Hilfe dieser Register lassen sich die SOLL-Konfiguration und die drei empfangenen IST-Konfigurationen prüfen. Nachdem Sie die GNN in MR 2000 eingetragen haben, überträgt die Steuerung die Werte in die 4 Registersätze.

MR 2000

GNN

Tragen Sie hier die GNN ein.

Modulregistereigenschaften

Werte 1 ... 199

Wert nach Reset 1

MR 2010 ... 2015

SOLL-Konfiguration

Aus diesen Registern können Sie die SOLL-Konfiguration, wie sie beim Start der Funktion mitgegeben wurde, auslesen.

Register	Aufrufparameter
2010	NodeID (GNN)
2011	CtrlID
2012	CtrlIDopt
2013	IpAddr
2014	IpMask
2015	Gateway

MR 2110 ... 2123

IST-Konfiguration 1

Aus diesen Registern können Sie die erste empfangene IST-Konfiguration auslesen.

Register	Aufrufparameter
2110	NodeID (GNN)
2111	CtrlID
2112	CtrlIDopt
2113	IpAddr
2114	IpMask
2115	Gateway
2120	Anzahl
2121	MAC-Adresse high

Register	Aufrufparameter
2122	MAC-Adresse low
2123	Gesendete IP-Adresse

MR 2210 ... 2223

IST-Konfiguration 2

Aus diesen Registern können Sie die zweite empfangene IST-Konfiguration auslesen.

Register	Aufrufparameter
2210	NodeID (GNN)
2211	CtrlID
2212	CtrlIDopt
2213	IpAddr
2214	IpMask
2215	Gateway
2220	Anzahl
2221	MAC-Adresse high
2222	MAC-Adresse low
2223	Gesendete IP-Adresse

MR 2310 ... 2323

IST-Konfiguration 3

Aus diesen Registern können Sie die dritte empfangene IST-Konfiguration auslesen.

Register	Aufrufparameter
2310	NodeID (GNN)
2311	CtrlID
2312	CtrlIDopt
2313	IpAddr
2314	IpMask
2315	Gateway
2320	Anzahl
2321	MAC-Adresse high
2322	MAC-Adresse low
2323	Gesendete IP-Adresse

2.4 Freiprogrammierbare IP-Schnittstelle

Die freiprogrammierbare IP-Schnittstelle

Die freiprogrammierbare IP-Schnittstelle bietet die Möglichkeit, beliebige Daten mit TCP/IP oder UDP/IP über die Ethernet-Schnittstelle des Geräts zu versenden oder zu empfangen. Die Verarbeitung der Daten geschieht dabei ganz im Anwendungsprogramm.

Anwendungen

Die freiprogrammierbare IP-Schnittstelle erlaubt dem Programmierer Daten über Ethernet-Verbindungen auszutauschen, die sich nicht Standardprotokollen - wie z. B. FTP, HTTP, JetIP oder Modbus/TCP - bedienen. Folgende Anwendungen sind dabei möglich:

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

Voraussetzungen an den Programmierer

Die Funktionalität der freiprogrammierbaren IP-Schnittstelle setzt folgende Kenntnisse der Datenübertragung über IP-Netzwerke voraus:

- IP-Adressierung (z. B. IP-Adresse, Port-Nummer, Subnetzmaske)
- TCP (z. B. Verbindungsaufbau / -abbau, Datastream, Datensicherung)
- UDP (z. B. Datagram)

Einschränkungen

Für die Kommunikation über die freiprogrammierbare IP-Schnittstelle darf der Programmierer keine Ports verwenden, die schon das Betriebssystem verwendet. Verwenden Sie deshalb folgende Ports nicht:

Protokoll	Port-Nummer	Standardwert	Benutzer
TCP	Abhängig vom FTP-Client	20	FTP-Server (Daten)
TCP	21		FTP-Server (Steuerung)
TCP	23		System-Logger
TCP	80		HTTP-Server
TCP	Aus der Datei /EMAIL/email.ini	25, 110	E-Mail-Client
TCP	502		Modbus/TCP-Server
TCP, UDP	1024 - 2047		Diverse
TCP, UDP	IP-Konfiguration	50000, 50001	JetIP
TCP	IP-Konfiguration	52000	Debug-Server

2 Erweiterungen

Inhalt

Thema	Seite
Programmierung	59
Register.....	71

2.4.1 Programmierung

Einleitung

Bei der freiprogrammierbaren IP-Schnittstelle werden die Daten über TCP/IP- oder UDP/IP-Verbindungen zwischen dem Anwendungsprogramm und einem Netzwerkteilnehmer ausgetauscht. Verwendung finden hierbei Aufrufe von Funktionen, die im Sprachumfang des Geräts enthalten sind. Führen Sie zur Programmierung folgende Schritte aus:

Schritt	Vorgehen
1	Freiprogrammierbare IP-Schnittstelle initialisieren
2	Verbindungen öffnen
3	Daten übertragen
4	Verbindungen schließen

Technische Daten

Die technischen Daten der freiprogrammierbaren IP-Schnittstelle:

Funktion	Beschreibung
Anzahl Verbindungen	20
Maximale Datengröße	4.000 Byte
Anzahl Empfangspuffer pro Verbindung	4

Einschränkungen

Während das Gerät gerade eine der Funktionen der freiprogrammierbaren IP-Schnittstelle bearbeitet, dürfen die Tasks, die die Funktionen aufgerufen haben, nicht durch `TaskBreak` angehalten oder durch `TaskRestart` neu gestartet werden.

Dies kann zu folgenden Fehlern führen:

- Verbindungen öffnen sich nicht
- Datenverlust beim Senden oder Empfangen
- Verbindungen bleiben ungewollt offen
- Verbindungen werden ungewollt geschlossen

Inhalt

Thema	Seite
Initialisieren der freiprogrammierbaren IP-Schnittstelle.....	60
Verbindung öffnen.....	61
Daten senden	65
Daten empfangen	67
Verbindung schließen	70

Initialisieren der freiprogrammierbaren IP-Schnittstelle

Einleitung Die Initialisierung muss mindestens einmal bei jedem Anwendungsprogrammstart ausgeführt werden.

Funktionsdeklaration `Function ConnectionInitialize():Int;`

Rückgabewert Folgender Rückgabewert ist möglich:

Rückgabewert

0	Immer
---	-------

Verwenden der Funktion So wird die Funktion verwendet und der Rückgabewert einer Variablen zur weiteren Auswertung zugewiesen:

```
Result := ConnectionInitialize();
```

Funktionsweise Das Gerät arbeitet die Funktion in folgenden Stufen ab:

Stufe	Beschreibung
1	Das Gerät schließt alle geöffneten Verbindungen der freiprogrammierbaren IP-Schnittstelle.
2	Das Gerät initialisiert alle betriebssysteminternen Datenstrukturen der freiprogrammierbaren IP-Schnittstelle.

Verwandte Themen

- **Verbindung öffnen** (siehe Seite 61)
 - **Verbindung schließen** (siehe Seite 70)
 - **Daten senden** (siehe Seite 65)
 - **Daten empfangen** (siehe Seite 67)
-

Verbindung öffnen

Einleitung

Bevor Daten gesendet oder empfangen werden können, muss eine Verbindung geöffnet werden. Dabei gilt Folgendes zu unterscheiden:

- Welches Transportprotokoll (TCP oder UDP) ist zu verwenden?
- Ist ein Client oder ein Server einzurichten?

Funktionsdeklaration

```
Function ConnectionCreate (ClientServerType: Int,
                          IPType: Int,
                          IPAddr: Int,
                          IPPort: Int,
                          Timeout: Int): Int;
```

Funktionsparameter

Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
ClientServerType	Client = 1 = CONNTYPE_CLIENT Server = 2 = CONNTYPE_SERVER	
IPType	UDP/IP = 1 = IPTYPE_UDP TCP/IP = 2 = IPTYPE_TCP	
IPAddr	Gültige IP-Adresse	Nur bei TCP/IP-Client erforderlich
IPPort	Gültige IP-Port-Nummer	Wird bei UDP/IP-Client ignoriert
Timeout	0 ... 1.073.741.824 [ms]	0 = unendlich

Rückgabewert

Bei einem positiven Rückgabewert konnte die Verbindung geöffnet werden. Bei einem negativen Rückgabewert ist ein Fehler aufgetreten und die Verbindung konnte nicht geöffnet werden.

Rückgabewert

> 0	Ein positiver Rückgabewert muss in einer Variablen gesichert werden. Er muss als Handle beim Aufruf der Funktionen für Daten senden, Daten empfangen und Verbindung schließen mitgegeben werden.
-1	Fehler beim Verbindungsaufbau
-2	Interner Fehler
-3	Ungültiger Parameter
-8	Zeitüberschreitung

2 Erweiterungen

Verwenden der Funktion bei einem TCP/IP-Client

Wenn ein Client eine TCP/IP-Verbindung zu einem Server aufbauen soll, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionCreate(CONNTYPE_CLIENT,  
                           IPTYPE_TCP,  
                           IP#192.168.75.123,  
                           46000,  
                           T#10s);
```

Funktionsweise bei einem TCP/IP-Client

Der Task bleibt bei der Programmzeile stehen, bis die Verbindung aufgebaut wurde oder der angegebene Timeout abgelaufen ist. Die Funktion läuft in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät versucht eine TCP/IP-Verbindung zu Port 46000 zu dem Netzwerkteilnehmer mit der IP-Adresse 192.168.75.123 aufzubauen.	
2	Wenn dann ...
	... der Netzwerkteilnehmer die Verbindung akzeptiert hat,	... wird die Funktion beendet und ein positiver Wert als Handle zum weiteren Zugriff auf die Verbindung zurückgeliefert.
	... die Verbindung nicht aufgebaut werden konnte und die Timeout-Zeit von 10 Sekunden noch nicht abgelaufen ist,	... wird mit Stufe 1 fortgefahren.
... ein Fehler aufgetreten ist oder der Timeout abgelaufen ist,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.	

Verwenden der Funktion bei einem TCP/IP-Server

Wenn ein Server eine TCP/IP-Verbindung mit einem Client aufbauen soll, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionCreate(CONNTYPE_SERVER,  
                           IPTYPE_TCP,  
                           0,  
                           46000,  
                           T#100s);
```

Funktionsweise bei einem TCP/IP-Server

Der Task bleibt bei der Programmzeile stehen, bis die Verbindung aufgebaut wurde oder der angegebene Timeout abgelaufen ist. Die Funktion läuft in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät richtet den TCP/IP-Port 46000 zum Empfang von Verbindungsanfragen ein.	
2	Wenn dann ...
	... der Client eines Netzwerkteilnehmers eine Verbindung aufgebaut hat,	... werden keine weiteren Verbindungsanfragen auf diesen Port akzeptiert, die Funktion beendet und ein positiver Wert als Handle zum weiteren Zugriff auf die Verbindung zurückgeliefert.
	... die Verbindung nicht aufgebaut werden konnte und die Timeout-Zeit von 100 Sekunden noch nicht abgelaufen ist,	... wird auf einen Verbindungsaufbau gewartet.
	... ein Fehler aufgetreten ist oder der Timeout abgelaufen ist,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verwenden der Funktion bei einem UDP/IP-Client

Wenn ein Client eine UDP/IP-Verbindung aufbauen soll, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionCreate(CONNTYPE_CLIENT,
                           IPTYPE_UDP,
                           0,
                           0,
                           0);
```

Funktionsweise bei einem UDP/IP-Client

UDP ist eine verbindungslose Art der Kommunikation. Deshalb öffnet das Gerät nur einen Kommunikationskanal, über den Daten an einen Netzwerkteilnehmer gesendet werden. Die Funktion läuft in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät richtet einen UDP/IP-Verbindungskanal zum Senden von Daten ein.	
2	Wenn dann ...
	... kein Fehler aufgetreten ist,	... wird die Funktion beendet und ein positiver Wert als Handle zum weiteren Zugriff auf die Verbindung zurückgeliefert.
	... ein Fehler aufgetreten ist,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

2 Erweiterungen

Verwenden der Funktion bei einem UDP/IP-Server

Wenn ein Server eine UDP/IP-Verbindung aufbauen soll, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionCreate(CONNTYPE_SERVER,  
                           IPTYPE_UDP,  
                           0,  
                           46000,  
                           0);
```

Funktionsweise bei einem UDP/IP-Server

UDP ist eine verbindungslose Art der Kommunikation. Deshalb öffnet das Gerät nur einen Kommunikationskanal über den Daten von einem Netzwerkteilnehmer empfangen werden. Die Funktion läuft in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät richtet einen UDP/IP-Verbindungskanal am Port 46000 zum Empfangen von Daten ein.	
2	Wenn dann ...
	... kein Fehler aufgetreten ist,	... wird die Funktion beendet und ein positiver Wert als Handle zum weiteren Zugriff auf die Verbindung zurückgeliefert.
	... ein Fehler aufgetreten ist,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verwandte Themen

- **Verbindung schließen** (siehe Seite 70)
 - **Daten senden** (siehe Seite 65)
 - **Daten empfangen** (siehe Seite 67)
 - **Initialisieren der freiprogrammierbaren IP-Schnittstelle** (siehe Seite 60)
-

Daten senden

Einleitung Daten können über eine zuvor geöffnete Verbindung gesendet werden.

Funktionsdeklaration

```
Function ConnectionSendData (IPConnection: Int,
                             IPAddr: Int,
                             IPPort: Int,
                             Const Ref SendData,
                             DataLen: Int) : Int;
```

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
IPConnection	Handle	Rückgabewert der Funktion ConnectionCreate ()
IPAddr	Gültige IP-Adresse	Nur bei UDP/IP-Client erforderlich
IPPort	Gültige IP-Port-Nummer	Nur bei UDP/IP-Client erforderlich
SendData	Adresse des zu sendenden Datenblocks	
DataLen	1 ... 4.000	Länge des Datenblocks in Byte

Rückgabewert Folgende Rückgabewerte sind möglich:

Rückgabewert	
0	Daten erfolgreich gesendet
-1	Fehler beim Senden, z. B. Verbindung abgebrochen
-3	Ungültiges Handle, z. B. senden über einen UDP/IP-Server

Verwenden der Funktion bei einer TCP/IP-Verbindung Wenn Daten über eine TCP/IP-Verbindung gesendet werden sollen, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionSendData (hConnection,
                              0,
                              0,
                              SendBuffer,
                              SendLen);
```

2 Erweiterungen

Funktionsweise bei einer TCP/IP-Verbindung

Bei TCP/IP werden die Daten über eine Verbindung übertragen, die vorher geöffnet wurde. Deshalb ist die Angabe von IP-Adresse und IP-Port-Nummer nicht mehr erforderlich und werden in der Funktion ignoriert.

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Die Daten wurden gesendet und ihr Empfang bestätigt.
- Ein Fehler ist aufgetreten.

Verwenden der Funktion bei einer UDP/IP-Verbindung

Wenn Daten bei einem Client über eine UDP/IP-Verbindung gesendet werden sollen, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionSendData(hConnection,  
                             IP#192.168.75.123,  
                             46000,  
                             SendBuffer,  
                             SendLen);
```

Funktionsweise bei einer UDP/IP-Verbindung

Da bei UDP/IP keine Verbindung zwischen zwei Netzwerkteilnehmern besteht, können Daten bei jedem Funktionsaufruf an einen anderen Teilnehmer oder anderen Port gesendet werden. Der Task bleibt bei diesem Funktionsaufruf stehen, bis die Daten gesendet wurden.

Sie erhalten keine Bestätigung, dass der andere Netzwerkteilnehmer die Daten empfangen hat.

UDP/IP-Client und -Server

Über eine UDP/IP-Client-Verbindung kann nur gesendet werden. Der Sende-Port wird vom Betriebssystem festgelegt.

Über eine UDP/IP-Server-Verbindung kann gesendet und empfangen werden. Als Sende-Port wird der bei der Verbindungseröffnung angegebene Port verwendet.

Verwandte Themen

- **Initialisieren der freiprogrammierbaren IP-Schnittstelle** (siehe Seite 60)
- **Verbindung öffnen** (siehe Seite 61)
- **Verbindung schließen** (siehe Seite 70)
- **Daten empfangen** (siehe Seite 67)

Daten empfangen

Einleitung Daten können über eine zuvor geöffnete TCP/IP-Verbindung oder über die UDP/IP-Verbindung eines Servers empfangen werden.
Über die UDP/IP-Verbindung eines Clients können keine Daten empfangen, sondern nur gesendet werden.

Einschränkungen Wenn Datenpakete über das Netzwerk empfangen werden, müssen diese vom Anwendungsprogramm abgeholt werden. Vom Betriebssystem der Steuerung werden pro Verbindung maximal vier Pakete zwischengespeichert. Alle weiteren Pakete werden verworfen.

Funktionsdeklaration

```
Function ConnectionReceiveData(IPConnection: Int,
                               Ref IPAddr: Int,
                               Ref IPPort: Int,
                               Ref ReceiveData,
                               DataLen: Int,
                               Timeout: Int): Int;
```

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
IPConnection	Handle	Rückgabewert der Funktion ConnectionCreate()
IPAddr	Adresse einer Variablen, um die IP-Adresse des Senders zu speichern	Nur bei UDP/IP-Server erforderlich
IPPort	Adresse einer Variablen, um die IP-Port-Nummer des Senders zu speichern	Nur bei UDP/IP-Server erforderlich
ReceiveData	Adresse des Empfangsdatenblocks	
DataLen	1 ... 4.000	Maximale Länge des Datenblocks in Byte
Timeout	0 ... 1.073.741.824 [ms]	0 = unendlich

Rückgabewert Folgende Rückgabewerte sind möglich:

Rückgabewert	
> 0	Anzahl empfangener Datenbyte
-1	Fehler beim Empfang, z. B. Verbindung abgebrochen
-3	Ungültiges Handle, z. B. empfangen über einen UDP/IP-Client
-8	Timeout

2 Erweiterungen

Verwenden der Funktion bei einer TCP/IP-Verbindung

Wenn Daten über eine TCP/IP-Verbindung empfangen werden sollen, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionReceiveData(hConnection,  
                                Dummy,  
                                Dummy,  
                                ReceiveBuffer,  
                                sizeof(ReceiveBuffer),  
                                T#10s);
```

Funktionsweise bei einer TCP/IP-Verbindung

Bei TCP/IP werden die Daten über eine Verbindung übertragen, die vorher geöffnet wurde. Deshalb ist die Angabe von IP-Adresse und IP-Port-Nummer nicht mehr erforderlich und werden in der Funktion ignoriert.

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Die Daten wurden empfangen.
- Ein Fehler ist aufgetreten.

Die Daten werden bei einer TCP/IP-Verbindung als Datenstrom übertragen. Das Gerät arbeitet die Funktion in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät wartet, bis Daten empfangen wurden, längstens jedoch die mitgegebene Timeout-Zeit.	
2	Wenn dann ...
	... der Timeout abgelaufen oder die Verbindung geschlossen wurde,	... wird die Funktion mit einer Fehlermeldung verlassen.
	... Daten empfangen wurden,	... werden sie in den mitgegebenen Empfangspuffer kopiert (höchstens jedoch bis zur mitgegebenen Anzahl) und bei Stufe 3 fortgefahren.
3	Wenn dann ...
	... mehr Daten empfangen wurden als in den Empfangspuffer kopiert werden konnten,	... werden diese vom Gerät gepuffert und können durch weitere Funktionsaufrufe abgeholt werden.
4	Die Funktion wird verlassen und die Anzahl der in den Empfangspuffer kopierten Daten zurückgegeben.	

Verwenden der Funktion bei einem UDP/IP-Server

Wenn Daten bei einem Server über eine UDP/IP-Verbindung empfangen werden sollen, können Sie die Funktion so aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionReceiveData(hConnection,  
                                IPAddr,  
                                IPPort,  
                                ReceiveBuffer,  
                                sizeof(ReceiveBuffer),  
                                T#10s);
```

Funktionsweise bei einem UDP/IP-Server

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Alle Daten wurden empfangen.
- Ein Fehler ist aufgetreten.

Die Daten werden bei einer UDP/IP-Verbindung als Datagramme übertragen. Das Gerät arbeitet die Funktion in folgenden Stufen ab:

Stufe	Beschreibung	
1	Das Gerät wartet bis alle Daten eines Datagramms empfangen wurden, längstens jedoch die mitgegebene Timeout-Zeit.	
2	Wenn dann ...
	... der Timeout abgelaufen oder die Verbindung geschlossen wurde,	... wird die Funktion mit einer Fehlermeldung verlassen.
	... Daten empfangen wurden,	... werden sie in den mitgegebenen Empfangspuffer kopiert (höchstens jedoch bis zur mitgegebenen Anzahl) und bei Stufe 3 fortgefahren.
3	Wenn dann ...
	... mehr Daten empfangen wurden als in den Empfangspuffer kopiert werden konnten, also das gesendete Datagramm zu groß ist,	... werden diese verworfen.
4	IP-Adresse und IP-Port-Nummer des Senders werden in die mitgegebenen Variablen übertragen.	
5	Die Funktion wird verlassen und die Anzahl der in den Empfangspuffer kopierten Daten zurückgegeben.	

Verwandte Themen

- **Initialisieren der freiprogrammierbaren IP-Schnittstelle** (siehe Seite 60)
- **Verbindung öffnen** (siehe Seite 61)
- **Verbindung schließen** (siehe Seite 70)
- **Daten senden** (siehe Seite 65)

Verbindung schließen

Einleitung Schließen Sie nicht mehr benötigte Verbindungen, da die Anzahl gleichzeitig geöffneter Verbindungen begrenzt ist.

Funktionsdeklaration `Function ConnectionDelete (IPConnection: Int) : Int;`

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
IPConnection	Handle	Rückgabewert der Funktion ConnectionCreate()

Rückgabewert Folgende Rückgabewerte sind möglich:

Rückgabewert

0	Verbindung geschlossen und gelöscht
-1	Ungültiges Handle

Verwenden der Funktion So können Sie die Funktion aufrufen und den Rückgabewert einer Variablen zur weiteren Auswertung zuweisen:

```
Result := ConnectionDelete (hConnection);
```

Verwandte Themen

- **Verbindung öffnen** (siehe Seite 61)
- **Daten senden** (siehe Seite 65)
- **Daten empfangen** (siehe Seite 67)
- **Initialisieren der freiprogrammierbaren IP-Schnittstelle** (siehe Seite 60)

2.4.2 Register

Einleitung

Dieses Kapitel beschreibt die Register des Geräts, in denen die aktuelle Verbindungsliste der freiprogrammierbaren IP-Schnittstelle enthalten ist. Die Register können zu Debug- oder Diagnosezwecken verwendet werden. Weitere Funktionen, wie Verbindung öffnen oder schließen, können hierüber nicht ausgelöst werden.

Inhalt

Thema	Seite
Registernummern	72
Registerbeschreibung.....	73

Registernummern

Einleitung

Die Daten jeweils einer Verbindung erscheinen in den Registern eines zusammenhängenden Registerblocks. Die Basisregisternummer dieses Blocks ist steuerungsabhängig.

Registernummern

Basisregisternummer	Registernummern
350000	350000 ... 350007

Registernummer ermitteln

In diesem Kapitel ist jeweils nur die letzte Ziffer der Registernummer angegeben, z. B. MR 1. Addieren Sie zu dieser Ziffer die Basisregisternummer des jeweiligen Geräts, z. B. 350000, um die vollständige Registernummer zu ermitteln.

Registerübersicht

Register	Beschreibung
MR 0	Verbindungsauswahl
MR 1	Verbindungstyp
MR 2	Transportprotokoll
MR 3	IP-Adresse
MR 4	IP-Port-Nummer
MR 5	Zustand
MR 6	Anzahl gesendeter Bytes
MR 7	Anzahl empfangener Bytes
MR 8	Anzahl verworfener Bytes
MR 9	Anzahl verworfener Pakete

Registerbeschreibung

Einleitung

Das Betriebssystem verwaltet die geöffneten Verbindungen in einer Liste. Mit Hilfe des Modulregisters MR 0 *Verbindungsauswahl* werden die Verbindungsdaten einer Verbindung in die anderen Register des Registerblocks kopiert.

MR 0

Verbindungsauswahl

Die Anwahl der Verbindungen geschieht, indem Werte in dieses Register geschrieben werden. Dieses Register zeigt, ob die folgenden Register Verbindungsdaten enthalten.

Modulregistereigenschaften

Werte lesen	0	Verbindung vorhanden
	-1	Verbindung nicht vorhanden

Modulregistereigenschaften

Werte schreiben	0	Erste Verbindung in der Liste anwählen
	> 0	Nächste Verbindung in der Liste anwählen
	< 0	Vorherige Verbindung in der Liste anwählen

MR 1

Verbindungstyp

Der Wert in diesem Register zeigt an, ob es sich um eine Client- oder eine Serververbindung handelt.

Modulregistereigenschaften

Werte	1	Client
	2	Server

MR 2

Transportprotokoll

Der Wert in diesem Register zeigt die Art des Transportprotokolls an, UDP oder TCP.

Modulregistereigenschaften

Werte	1	UDP
	2	TCP

MR 3

IP-Adresse

Der Wert in diesem Register zeigt die konfigurierte IP-Adresse an.

Modulregistereigenschaften

Werte	0.0.0.0 ... 255.255.255.255
-------	-----------------------------

MR 4

IP-Port-Nummer

Der Wert in diesem Register zeigt die konfigurierte IP-Port-Nummer an.

Modulregistereigenschaften

Werte	0 ... 65.535
-------	--------------

MR 5

Zustand

Der Wert in diesem Register zeigt an, in welchem Zustand sich die Verbindung befindet.

Modulregistereigenschaften

Werte	0	Verbindung geschlossen
	1	Verbindung wird geöffnet
	2	Verbindung ist geöffnet
	3	TCP/IP-Server: Warten auf Verbindungsanfrage von Client
	4	Interne Verwendung

MR 6

Anzahl gesendeter Bytes

Der Wert in diesem Register zeigt die Anzahl der über diese Verbindung gesendeten Datenbytes an. Da es sich um ein vorzeichenbehaftetes 32-Bit-Register handelt und die gesendeten Bytes jeweils hinzuaddiert werden, können Zahlenüberläufe vom positiven zum negativen Maximalwert auftreten.

Modulregistereigenschaften

Werte	-2.147.483.648 ... 2.147.483.647
-------	----------------------------------

MR 7**Anzahl empfangener Bytes**

Der Wert in diesem Register zeigt die Anzahl der über diese Verbindung empfangenen Datenbytes an. Da es sich um ein vorzeichenbehaftetes 32-Bit-Register handelt und die empfangenen Bytes jeweils hinzuaddiert werden, können Zahlenüberläufe vom positiven zum negativen Maximalwert auftreten.

Modulregistereigenschaften

Werte	-2.147.483.648 ... 2.147.483.647
-------	----------------------------------

MR 8**Anzahl verworfener Bytes**

Der Wert in diesem Register zeigt die Anzahl der Datenbytes an, die nicht empfangen werden konnten, weil das Anwendungsprogramm die zwischengespeicherten Datenbytes nicht abgeholt hat.

Modulregistereigenschaften

Werte	0 ... 2.147.483.647
-------	---------------------

MR 9**Anzahl verworfener Pakete**

Der Wert in diesem Register zeigt die Anzahl der Datenpakete an, die nicht empfangen werden konnten, weil das Anwendungsprogramm die zwischengespeicherten Datenpakete nicht abgeholt hat.

Modulregistereigenschaften

Werte	0 ... 2.147.483.647
-------	---------------------

3 Beseitigte Software-Bugs

Einleitung Dieses Kapitel beschreibt die in der neuen Betriebssystemversion beseitigten Software-Bugs.

Inhalt

Thema	Seite
Ungültiges Verbindungs-Handle bei der freiprogrammierbaren IP-Schnittstelle	77
Lange Schlüsselnamen führen zum Absturz	78
Umbenennen einer Datei mit maximaler Namenslänge führt zum Absturz..	79
Eingangswerte eines Netzwerkteilnehmers sind eingefroren.....	80
Inkonsistente Register bei der IP-Konfiguration	81

Ungültiges Verbindungs-Handle bei der freiprogrammierbaren IP-Schnittstelle

Fehlerbild

Wenn bei `ConnectionReceiveData()`, `ConnectionSendData()` oder `ConnectionDelete()` ein ungültiges Verbindungs-Handle mitgegeben wird stürzt die Steuerung ab.

Betroffene Versionen/Revisionen

Der Fehler tritt in folgenden Versionen/Revisionen auf:

Betriebssystemversion	JC-340/350	< 1.28.0.00
	JC-360/365 (MC)	< 1.28.0.00
	JC-940MC	< 1.10.0.00
	JC-310-JM	< 1.28.0.00
	JCM-350	< 1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Abhilfe/Workaround

Prüfen Sie vor der Verwendung das Verbindungs-Handle auf Gültigkeit.

Fehlerbehebung

Der Fehler ist ab den folgenden Versionen/Revisionen behoben:

Betriebssystemversion	JC-340/350	1.28.0.00
	JC-360/365 (MC)	1.28.0.00
	JC-940MC	1.10.0.00
	JC-310-JM	1.28.0.00
	JCM-350	1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Lange Schlüsselnamen führen zum Absturz

Fehlerbild

Nachdem für mehrere Schlösser/Schlüssel in der Datei **/System/keys.ini** längere Namen zugeordnet wurden stürzt die Steuerung ab.

Betroffene Versionen/Revisionen

Der Fehler tritt in folgenden Versionen/Revisionen auf:

Betriebssystemversion	JC-340/350	< 1.28.0.00
	JC-360/365 (MC)	< 1.28.0.00
	JC-940MC	< 1.10.0.00
	JC-310-JM	< 1.28.0.00
	JM-200-ETH	< 1.28.0.00
	JCM-350	< 1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Abhilfe/Workaround

Stellen Sie sicher, dass die Länge (Anzahl Zeichen) aller Namen in der Datei **/System/keys.ini** den Wert 224 nicht übersteigt.

Fehlerbehebung

Der Fehler ist ab den folgenden Versionen/Revisionen behoben:

Betriebssystemversion	JC-340/350	1.28.0.00
	JC-360/365 (MC)	1.28.0.00
	JC-940MC	1.10.0.00
	JC-310-JM	1.28.0.00
	JM-200-ETH	1.28.0.00
	JCM-350	1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Umbenennen einer Datei mit maximaler Namenslänge führt zum Absturz

Fehlerbild

Wenn eine Datei umbenannt wird und der neue Namen hat die maximal zulässige Länge von 63 Zeichen stürzt die Steuerung ab.

Betroffene Versionen/Revisionen

Der Fehler tritt in folgenden Versionen/Revisionen auf:

Betriebssystemversion	JC-340/350	< 1.28.0.00
	JC-360/365 (MC)	< 1.28.0.00
	JC-940MC	< 1.10.0.00
	JC-310-JM	< 1.28.0.00
	JM-200-ETH	< 1.28.0.00
	JCM-350	< 1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Abhilfe/Workaround

Stellen Sie sicher, dass die Länge des neuen Dateinamens 62 Zeichen nicht übersteigt.

Fehlerbehebung

Der Fehler ist ab den folgenden Versionen/Revisionen behoben:

Betriebssystemversion	JC-340/350	1.28.0.00
	JC-360/365 (MC)	1.28.0.00
	JC-940MC	1.10.0.00
	JC-310-JM	1.28.0.00
	JM-200-ETH	1.28.0.00
	JCM-350	1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Eingangswerte eines Netzwerkteilnehmers sind eingefroren

Fehlerbild

Wenn mehrere Netzwerkteilnehmer mit einer Steuerung über Publish/Subscribe kommunizieren, ändern sich die Eingangswerte eines der Teilnehmer nicht mehr. Die Ausgangswerte zu diesem Teilnehmer stimmen weiterhin. Ebenfalls die Eingänge und Ausgänge zu den anderen Teilnehmern. Es werden keine Fehler angezeigt. Das Problem lässt sich durch einen Neustart des Subscribers auf der Steuerung beheben.

Betroffene Versionen/Revisionen

Der Fehler tritt in folgenden Versionen/Revisionen auf:

Betriebssystemversion	JC-340/350	< 1.28.0.00
	JC-360/365 (MC)	< 1.28.0.00
	JC-940MC	< 1.10.0.00
	JC-310-JM	< 1.28.0.00
	JCM-350	< 1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Abhilfe/Workaround

Verbinden Sie bei jedem Netzwerkteilnehmer einen nicht verwendeten Ausgang mit einem nicht verwendeten Eingang. Schalten Sie im Anwendungsprogramm den Ausgang ein und aus. Prüfen Sie ob der Eingang dem Zustand des Ausgangs folgt. Wenn dies nicht mehr der Fall ist, starten Sie den Subscriber auf der Steuerung neu.

Fehlerbehebung

Der Fehler ist ab den folgenden Versionen/Revisionen behoben:

Betriebssystemversion	JC-340/350	1.28.0.00
	JC-360/365 (MC)	1.28.0.00
	JC-940MC	1.10.0.00
	JC-310-JM	1.28.0.00
	JCM-350	1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Inkonsistente Register bei der IP-Konfiguration

Fehlerbild

Beim Ändern der IP-Einstellungen in den Register 101200 bis 101202 werden die neuen Werte sofort in den Register 104531 bis 104533 sichtbar. Sie werden jedoch nicht sofort wirksam.

Betroffene Versionen/Revisionen

Der Fehler tritt in folgenden Versionen/Revisionen auf:

Betriebssystemversion	JC-340/350	< 1.28.0.00
	JC-360/365 (MC)	< 1.28.0.00
	JC-310-JM	< 1.28.0.00
	JCM-350	< 1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Abhilfe/Workaround

Eine Abhilfe oder ein Workaround ist in den betroffenen Version nicht möglich.

Fehlerbehebung

Der Fehler ist ab den folgenden Versionen/Revisionen behoben:

Betriebssystemversion	JC-340/350	1.28.0.00
	JC-360/365 (MC)	1.28.0.00
	JC-310-JM	1.28.0.00
	JCM-350	1.20.0.00
Hardwarerevision	Nicht relevant	
Konfiguration oder Betriebsart	Nicht relevant	

Jetter AG
Gräterstraße 2
71642 Ludwigsburg | Germany

Tel +49 7141 2550-0
Fax +49 7141 2550-425
info@jetter.de
www.jetter.de

We automate your success.