

Themenhandbuch

Dateisystem

60881371

We automate your success.

Artikelnummer: 60881371

Version 1.10

April 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	Dateisystem	5
1.1	Verzeichnisse	6
	Verzeichnisse bei Steuerungen.....	7
	Verzeichnisse bei Bediengeräten.....	8
1.2	Eigenschaften	11
	Eigenschaften der Flash-Disk	12
	Eigenschaften der SD-Karte	13
	Eigenschaften des USB-Sticks	14
1.3	Benutzerverwaltung	15
	Benutzer verwalten	17
	Auslieferungszustand/Vordefinierte Benutzer und Schlüssel	19
	Schloss anbringen	20
	Namen einrichten für Schlüssel/Schlösser	22
1.4	Belegung der Flash-Disk einsehen	24
	Belegung der Flash-Disk.....	25
1.5	Formatieren und Prüfen	28
	Flash-Disk formatieren.....	29
	SD-Karte formatieren	30
	USB-Stick formatieren.....	31
	SD-Karte prüfen	32
	USB-Stick überprüfen	33
2	FTP-Server	35
	Anmeldung	37
	Beispiel: Windows FTP-Client.....	38
3	HTTP-Server	39
3.1	Server Side Includes	40
	Erster Eintrag in der HTML-Datei.....	41
	Einfügen von Echtzeit-Steuerungswerten	42
	Beispiel einer HTML-Seite	47
4	FTP-Client	49
4.1	Programmierung	51
	FTP-Client initialisieren	52
	Verbindung zum FTP-Server öffnen	53
	Verbindung schließen	55
	Datei lesen	56
	Datei schreiben	58
	Datei löschen	60
	Verzeichnis wechseln.....	62
	Verzeichnis anlegen.....	64
	Verzeichnis löschen	66
	Aktuelles Verzeichnis ermitteln	68
4.2	Register	70
	Registernummern.....	71
	Registerbeschreibung	72

5	AutoCopy - Automatisches Kopieren von Steuerungsdaten	75
5.1	Funktionsweise	77
	AutoCopy-Funktion starten.....	78
	AutoCopy-Funktion ausführen	79
	AutoCopy-Funktion beenden	82
5.2	autocopy.ini - Aufbau	83
	Die Sektion [OPTIONS].....	84
	Die Kommandosektionen	85
	Beispiel für eine Kommandodatei	93
5.3	Protokolldatei	96
	Dateiinhalt	97
5.4	Datendateien	98
	Dateiformat.....	99
6	Anwendungsprogramm	101
	Standardablage des Anwendungsprogramms	102
	Anwendungsprogramm auf der SD-Karte oder dem USB-Stick ablegen	103
	Anwendungsprogramm laden	105

1 Dateisystem

Einleitung Dieses Kapitel beschreibt das Dateisystem. Das Dateisystem ermöglicht den Zugriff auf die Dateien der internen Flash-Disk, der SD-Karte und eines USB-Sticks.
Wenn zu lösende Probleme auftreten, ist es hilfreich, das Dateisystem zu verstehen.

Hinweis Arbeiten Sie vorsichtig im Umgang mit dem Dateisystem, insbesondere mit den Systemdateien. Fehlerhafte Systemdateien können zu einem Gerät führen, das nicht mehr bootet.
Nicht alle Dateien sind lesbar, schreibbar oder löscherbar. Dieses Verhalten ist normal. Einige dieser Dateien sind virtuelle Dateien, z. B. Firmware Images, oder geschützte Dateien, z. B. EDS-Dateien.

Aufteilung Das Dateisystem unterscheidet zwischen folgenden Dateien:

- Vom Betriebssystem verwendete Systemverzeichnisse/-dateien
- Vom Anwender verwendete Dateien

Inhalt

Thema	Seite
Verzeichnisse.....	6
Eigenschaften	11
Benutzerverwaltung.....	15
Belegung der Flash-Disk einsehen.....	24
Formatieren und Prüfen.....	28

1.1 Verzeichnisse

Systemverzeichnisse

Systemverzeichnisse sind nicht löscherbar. Sie sind auch nach dem Formatieren noch vorhanden.

Steuerungen

Verzeichnisnamen durch einen Schrägstrich "/" und nicht durch einen umgekehrten Schrägstrich "\" trennen.

Verzeichnis	Beschreibung
/System	<ul style="list-style-type: none">▪ Systemkonfiguration▪ Systeminformationen
/SD	<ul style="list-style-type: none">▪ Root-Verzeichnis der SD-Karte
/USBx	<ul style="list-style-type: none">▪ Root-Verzeichnis des USB-Sticks x

Bediengeräte

Verzeichnisnamen durch einen umgekehrten Schrägstrich "\" und nicht durch einen Schrägstrich "/" trennen.

Verzeichnis	Beschreibung
\System	<ul style="list-style-type: none">▪ Systemkonfiguration▪ Systeminformationen▪ Begrüßungsbild (Boot-Image)▪ Screenshot
\SD	<ul style="list-style-type: none">▪ Root-Verzeichnis der SD-Karte
\USB	<ul style="list-style-type: none">▪ Root-Verzeichnis des USB-Sticks
\App	<ul style="list-style-type: none">▪ Speicherbereich für die Applikation
\Data	<ul style="list-style-type: none">▪ Speicherbereich für die Daten
\Windows	<ul style="list-style-type: none">▪ Systemverzeichnis von Windows CE
\	<ul style="list-style-type: none">▪ RAM-Disk

Inhalt

Thema	Seite
Verzeichnisse bei Steuerungen	7
Verzeichnisse bei Bediengeräten	8

Verzeichnisse bei Steuerungen

Verzeichnis */SD****/SD***

Wenn Sie eine SD-Karte in den SD-Karten-Steckplatz stecken, wird das Verzeichnis */SD* dynamisch erstellt. Sonst ist dieses Verzeichnis nicht sichtbar.

Verzeichnis */System****/System***

Dieses Verzeichnis enthält systemrelevante Dateien, wie den Kernel, die Co-Prozessor-Firmware, Konfigurationen, EDS usw.

Verzeichnis */USBx****/USBx***

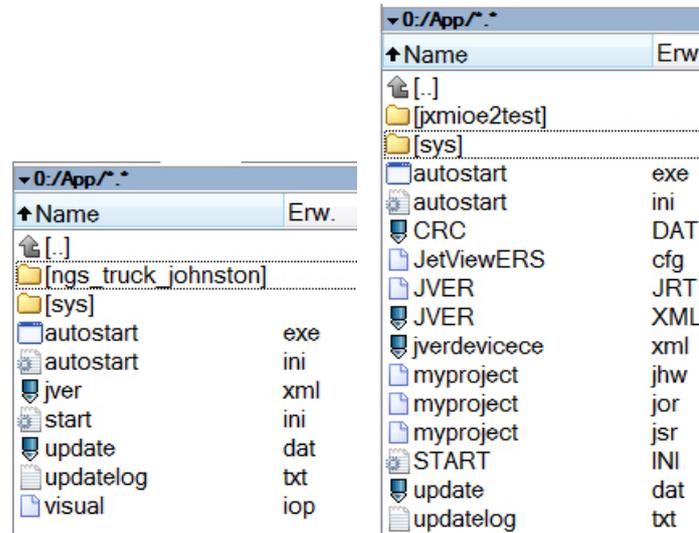
Wenn Sie einen USB-Stick in den USB-Anschluss x stecken, wird das Verzeichnis */USBx* dynamisch erstellt. Sonst ist dieses Verzeichnis nicht sichtbar.

Verzeichnisse bei Bediengeräten

Verzeichnis *\App*

\App

Dieses Verzeichnis enthält die Anwendung und die Visualisierung. In diesem Beispiel ist die STX-Anwendung in den Ordnern **ngs_truck_johnston** (links) oder **jxmioe2test** (rechts) gespeichert.



Links basieren die Daten auf der älteren S-Plattform. Die Visualisierung ist in einer .iop-Datei gespeichert, hier ist es die Datei **visual.iop**.

Rechts basieren die Daten auf der neuen CE-Plattform. Da gibt es keine .iop-Datei. Stattdessen speichert JetViewSoft mehrere Visualisierungsdateien ab.

Hinweis!

Kopieren Sie alle Anwendungs- und Visualisierungsdateien in den Ordner **App** und nicht in den Ordner **Data**. Sonst verursacht das Verzögerungen beim Hochfahren, siehe Verzeichnis *\Data*.

start.ini

Der Inhalt dieser Textdatei legt fest, welche Anwendung gestartet wird.

\App\sys

Dieses Verzeichnis enthält die Interpreter der STX-Programmiersprache und der Visualisierung. **Hier nichts verändern!**

autostart.exe

Diese Anwendung lässt ein Betriebssystemupdate zu. **Hier nichts verändern!** Außerdem startet die Datei das Gerät mitsamt der Visualisierung.

updatelog.txt

Das ist eine Log-Datei, die während des Betriebssystemupdates geschrieben wird.

Verzeichnis \Data**\Data**

Dieses Verzeichnis ist der große Speicher des Bediengeräts. Sie können einige Parameter- oder Konfigurationsdateien hier speichern.

Wichtiger Hinweis!

Auf der Datenpartition können größere Datenmengen abgelegt werden. Um einen zügigen Systemstart sicherzustellen, wird diese Partition ggf. zeitversetzt nach dem Start der STX-Applikation eingebunden. Die Applikation kann nicht auf dieser Partition abgelegt werden.

Verzeichnis \SD**\SD**

Wenn Sie eine SD-Karte in den SD-Karten-Steckplatz des Bediengeräts stecken, wird das Verzeichnis \SD dynamisch erstellt. Sonst ist dieses Verzeichnis nicht sichtbar.

Verzeichnis \System**\System**

Dieses Verzeichnis enthält systemrelevante Dateien, wie den Kernel, die Co-Prozessor-Firmware, Konfigurationen, EDS usw.

Name	Erw.
[EDS]	
[OS]	
_eeprom_base	vol
_eeprom_som	vol
_JVCN_kernel-1	os
_JVCN_kernel-2	os
_perreg1	fdf
_perreg2	fdf
_xldr	bin
bootupscreen	bmp
co-processor1	os
co-processor2	os
co-processor3	os
fpga	os
JVCN_kernel	os
perreg	fdf
reset	exe
screenshot	bmp
sysconfig	os
systemlog	txt

bootupscreen.bmp

Das ist eine 16-Bit-Bmp-Datei (r5, g6, b5), die als Bild direkt nach dem Einschalten sichtbar ist.

Sie können eine eigene Bilddatei erstellen und diese Datei ersetzen.

co-processor1

Diese virtuelle Datei enthält die Firmware eines versteckten Co-Prozessors, der die meisten Interaktionen (Schaltflächen, Summer, Hintergrundlicht usw.) mit dem Anwender steuert.

reset.exe

Wenn Sie diese Datei löschen, führt das Bediengerät sofort danach einen Neustart aus. Diese Funktion können Sie z. B. in Batch-Dateien anwenden, die nach der Abarbeitung einen automatischen Neustart erfordern.

Verzeichnis *USB*

USB

Wenn Sie einen USB-Stick in das Bediengerät stecken, wird das Verzeichnis *USB* dynamisch erstellt. Sonst ist dieses Verzeichnis nicht sichtbar.

Verzeichnis *Windows*

Windows

Dieses Verzeichnis enthält Dateien von Windows CE. **Hier nichts verändern!**

1.2 Eigenschaften

Einleitung

Dieses Kapitel beschreibt die Eigenschaften des Dateisystems. Unterschieden wird dabei zwischen der internen Flash-Disk, der SD-Karte und des USB-Sticks.

Allgemeine Eigenschaften

Für die interne Flash-Disk, die SD-Karte und den USB-Stick gelten folgende Eigenschaften:

- Maximal 8 Dateien gleichzeitig öffnen.
- Ausschließlich Kleinschreibung für Verzeichnis- und Dateinamen verwenden.
- Wenn das Gerät eine Datei erstellt, erhält die Datei Datum und Uhrzeit des Geräts.
- Datum, Uhrzeit und Dateigröße sind nicht bei allen Systemdateien verfügbar.

Inhalt

Thema	Seite
Eigenschaften der Flash-Disk.....	12
Eigenschaften der SD-Karte.....	13
Eigenschaften des USB-Sticks	14

Eigenschaften der Flash-Disk

Größe

Die Speichergröße ist geräteabhängig.

Steuerungen: Eigenschaften

Die interne Flash-Disk hat folgende weitere Eigenschaften:

- 7 Verzeichnisebenen und 1 Dateiebene sind zulässig.
 - Groß- und Kleinschreibung wird unterschieden.
 - Verzeichnis- und Dateinamen mit jeweils maximal 63 Zeichen Länge sind möglich.
 - Für Verzeichnis- und Dateinamen sind alle Zeichen außer "/" und ".." erlaubt.
 - Benutzer-/Zugriffsverwaltung mit maximal 31 Schlössern für maximal 33 Benutzer.
-

Bediengeräte: Eigenschaften

Die interne Flash-Disk hat folgende weitere Eigenschaften:

- 7 Verzeichnisebenen und 1 Dateiebene sind zulässig.
 - Eine Unterscheidung zwischen Groß- und Kleinschreibung gibt es nicht.
 - Für die Zeichenlänge von Verzeichnis- und Dateinamen sind gesamt 63 Zeichen zulässig.
 - Für Verzeichnis- und Dateinamen sind alle Zeichen außer "\" und ".." erlaubt.
 - Die Ordner **App** und **Data** liegen auf der Flash-Disk.
 - Es gibt keine Benutzer-/Zugriffsverwaltung.
-

Eigenschaften der SD-Karte

Größe

Die Speichergröße ist geräteabhängig.

Eigenschaften

Die SD-Karte hat folgende weitere Eigenschaften:

- Die SD-Karte muss FAT 16 kompatibel sein.
- Verzeichnis- und Dateinamen mit gesamt 260 Zeichen sind möglich.
- In Verzeichnis- und Dateinamen sind nicht erlaubt: "/", "\", ":", "*", "?", "", "<", ">" und "|"
- Es gibt keine Benutzer-/Zugriffsverwaltung.

Die Jetter AG übernimmt nur für die als Option erhältlichen Karten die Gewähr.

Eigenschaften des USB-Sticks

Größe

Die Speichergröße ist geräteabhängig.

Eigenschaften

Der USB-Stick hat folgende weitere Eigenschaften:

- Der USB-Stick muss FAT 16 oder FAT 32 kompatibel sein.
- Verzeichnis- und Dateinamen mit gesamt 260 Zeichen sind möglich.
- In Verzeichnis- und Dateinamen sind nicht erlaubt: "/", "\", ":", "*", "?", "''", "<", ">" und "|"
- Es gibt keine Benutzer-/Zugriffsverwaltung.

Die Jetter AG übernimmt nur für die als Option erhältlichen USB-Sticks die Gewähr.

1.3 Benutzerverwaltung

Einleitung

Im Dateisystem für die interne Flash-Disk besteht die Möglichkeit, Zugriffsrechte (Schlösser) für Verzeichnisse zu definieren und Benutzer einzurichten.

Für jeden Benutzer sind die Zugriffsrechte (Schlüssel) einstellbar.

Auf Verzeichnisse und Dateien, für die der Benutzer nicht den benötigten Schlüssel besitzt, ist der Zugriff nicht möglich. Diese Verzeichnisse und Dateien erscheinen bei einer FTP-Verbindung auch nicht.

Voraussetzungen

Für die Benutzerverwaltung sind Administratorrechte erforderlich.

Eigenschaften

Die Benutzerverwaltung hat folgende Eigenschaften:

Eigenschaft	Maximalwert
Anzahl Benutzer	33
Anzahl vordefinierter Benutzer	2
Länge Benutzername	31 alphanumerische Zeichen
Länge Passwort	31 alphanumerische Zeichen
Anzahl Leseschlüssel	31
Anzahl Schreischlüssel	31
Anzahl vordefinierter Schlüssel	2

Dateien

Die Einstellungen der Benutzerverwaltung geschieht in drei Dateien im Ordner **System**:

Datei	Funktion
flashdisklock.ini	Schlösser an Verzeichnisse anbringen
keys.ini	Namen für Schlösser/Schlüssel vergeben
users.ini	Benutzer verwalten

Die Dateien sind immer vorhanden. Sie sind nicht löscherbar, sondern sind nur änderbar oder überschreibbar.

Einschränkungen

Beachten Sie bitte die folgenden Einschränkungen:

- Die Benutzerverwaltung ist nur für die interne Flash-Disk möglich. Sie ist nicht auf die SD-Karte und den USB-Stick anwendbar.
- Wenn eine Datei der Benutzerverwaltung übertragen wurde, ist ihr Inhalt sofort lesbar. Die Einstellungen werden erst nach dem nächsten Boot-Vorgang gültig.

Inhalt

Thema	Seite
Benutzer verwalten	17
Auslieferungszustand/Vordefinierte Benutzer und Schlüssel	19
Schloss anbringen	20
Namen einrichten für Schlüssel/Schlösser	22

Benutzer verwalten

Einleitung Die Benutzer des Dateisystems verwalten Sie in der Konfigurationsdatei **/System/users.ini**.

Voraussetzungen Wenn Sie Namen für die Schlüssel verwenden wollen, muss das Gerät die Namen zuvor kennen. Richten Sie die Namen deshalb zuerst ein, *Einrichten von Namen für Schlüssel/Schlösser* (siehe Seite 22).

Benutzer verwalten Um die Benutzer zu verwalten, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Stellen Sie eine FTP-Verbindung zum Gerät her. Melden Sie sich hierbei mit Administratorrechten an.
2	Öffnen Sie die Datei /System/users.ini .
3	Ändern Sie die Dateieinträge.
4	Speichern Sie die geänderte Datei wieder auf dem Gerät.
5	Booten Sie das Gerät neu.

Ergebnis: Die geänderten Benutzereinstellungen sind aktiv.

Aufbau der Konfigurationsdatei

Die Konfigurationsdatei ist eine Textdatei, deren Einträge in Sektionen gruppiert sind.

- Für jeden Benutzer wird eine eigene Sektion verwendet.
- In diesen Sektionen werden Werte gesetzt, mit denen das Dateisystem arbeitet.
- Leerzeilen können Sie beliebig einfügen.
- Kommentarzeilen werden mit "!", "#" oder ";" eingeleitet.

Sektionen

Die Namen der Sektionen sind **[USER1]** bis **[USER33]**. Hier legen Sie jeweils Benutzernamen und Passwort sowie die Lese- und Schreibrechte fest.

Beispiel:

```
[USER4]
NAME=TestUser3
PW=testpass
READKEYS=5,openLock2,10,11
WRITEKEYS=openLock2,10,11
SYSKEYS=
```

NAME	
Im Beispiel	TestUser3
Beschreibung	Anmeldename des Benutzers
Gültige Werte	Maximal 31 alphanumerische Zeichen
Bei ungültigem Wert oder nicht vorhandenem Eintrag	Der Benutzer wird nicht angelegt
PW	
Im Beispiel	testpass
Beschreibung	Anmeldepasswort des Benutzers
Gültige Werte	Maximal 31 alphanumerische Zeichen
Bei nicht vorhandenem Eintrag	Für die Anmeldung des Benutzers ist kein Passwort erforderlich
READKEYS	
Im Beispiel	5,openLock2,10,11
Beschreibung	Schlüssel für Lesezugriffe
Gültige Werte	1 ... 31 (oder entsprechende Namen)
Bei nicht vorhandenem Eintrag	Der Benutzer erhält keine Leseschlüssel
WRITEKEYS	
Im Beispiel	openLock2,10,11
Beschreibung	Schlüssel für Schreibzugriffe
Gültige Werte	1 ... 31 (oder entsprechende Namen)
Bei nicht vorhandenem Eintrag	Der Benutzer erhält keine Schreibschlüssel
SYSKEYS	
Beschreibung	Keine Funktion; reserviert für zukünftige Erweiterungen

Auslieferungszustand/Vordefinierte Benutzer und Schlüssel

Einleitung

Im Dateisystem sind zwei Benutzer mit festgelegten Rechten vordefiniert. Diese beiden Benutzer können nicht gelöscht werden. Mit Hilfe der Benutzerverwaltung können Sie für diese Benutzer lediglich ein anderes Passwort festlegen.

Auslieferungszustand

Im Auslieferungszustand der Steuerung hat die Konfigurationsdatei folgenden Inhalt:

```
[USER1]
NAME=admin
PW=admin
READKEYS=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
WRITEKEYS=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
SYSKEYS=

[USER33]
NAME=system
PW=system
READKEYS=2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
WRITEKEYS=2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31
SYSKEYS=
```

Vordefinierte Benutzer

Der Benutzer *admin* hat alle Schlüssel und kann somit auf alle Verzeichnisse und Dateien lesend und schreibend zugreifen.

Der Benutzer *system* hat bis auf Schlüssel **1** ebenfalls alle Schlüssel.

Vordefinierte Schlüssel

Von den 31 Schlüsseln haben zwei eine vordefinierte Funktion.

Schloss/Schlüssel	Funktion
1	<ul style="list-style-type: none"> ▪ Ethernet-Konfiguration ▪ Benutzerverwaltung
2	<ul style="list-style-type: none"> ▪ Betriebssystemupdate der CPU ▪ Betriebssystemupdate der JX2- und JX3-Module

Schloss anbringen

Einleitung In der Konfigurationsdatei **/System/flashdisklock.ini** versehen Sie Verzeichnisse der Flash-Disk mit Schlössern. Nur Benutzer mit dem passenden Schlüssel können Dateien und Unterverzeichnisse in diesen Verzeichnissen lesen oder schreiben (löschen).

Voraussetzungen Wenn Sie Namen für die Schlösser verwenden wollen, muss das Gerät die Namen zuvor kennen. Richten Sie die Namen deshalb zuerst ein, *Namen einrichten für Schlüssel/Schlösser* (siehe Seite 22).

Schloss anbringen Um ein Schloss an ein Verzeichnis anzubringen, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Stellen Sie eine FTP-Verbindung zu dem Gerät her; melden Sie sich hierbei mit Administratorrechten an.
2	Öffnen Sie die Datei /System/flashdisklock.ini .
3	Ändern Sie die Dateieinträge.
4	Speichern Sie die geänderte Datei wieder auf dem Gerät.
5	Booten Sie das Gerät neu.

Ergebnis: Das Verzeichnis ist mit einem Schloss versehen.

Aufbau der Konfigurationsdatei Die Konfigurationsdatei ist eine Textdatei, die eine Sektion enthält.

- In dieser Sektion werden Werte gesetzt, mit denen das Dateisystem arbeitet.
 - Jedes Verzeichnis geben Sie mit seiner Schlossnummer auf einer eigenen Zeile an.
 - Leerzeilen können Sie beliebig einfügen.
 - Kommentarzeilen werden mit "!", "#" oder ";" eingeleitet.
-

Sektion Der Name der Sektion ist **[LOCKS]**. Hier ordnen Sie die Schlösser den Verzeichnissen nach folgendem Schema zu:

Verzeichnis=Schloss

Beispiel:

```
[LOCKS]
test1=0
test1/sub1=2
test1/sub2=5
test2=userlock2
```

Schlossnummern

Die Schlossnummern haben folgende Eigenschaften:

- Gültige Schlossnummern sind 0 ... 31.
 - Schlossnummer 0: Am Verzeichnis ist kein Schloss angebracht. Sie benötigen keine besondere Zugangsberechtigung.
 - Sie können Nummern oder zuvor definierte Namen verwenden.
-

Namen einrichten für Schlüssel/Schlösser

Einleitung

Die Schlüssel/Schlösser sind von 1 ... 31 durchnummeriert. Um einfacher mit Schlüssel/Schloss umzugehen, kann jedem Schlüssel/Schloss ein Name zugeordnet werden. In der Konfigurationsdatei **/System/keys.ini** richten Sie die Namen ein.

Namen einrichten

Um Namen für Schlüssel/Schlösser einzurichten, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Stellen Sie eine FTP-Verbindung zu dem Gerät her; melden Sie sich hierbei mit Administratorrechten an.
2	Öffnen Sie die Datei /System/keys.ini .
3	Ändern Sie die Dateieinträge.
4	Speichern Sie die geänderte Datei wieder auf dem Gerät.
5	Booten Sie das Gerät neu.

Ergebnis:

Die Namen stehen jetzt zur Verfügung. Benutzer können die Namen jetzt beim Anbringen von Schlössern und Verwalten von Benutzern verwenden.

Aufbau der Konfigurationsdatei

Die Konfigurationsdatei ist eine Textdatei, die eine Sektion enthält.

- In dieser Sektion werden Werte gesetzt, mit denen das Dateisystem arbeitet.
- Jeder Schlüssel ist mit seinem Namen auf einer eigenen Zeile angegeben.
- Leerzeilen können Sie beliebig einfügen.
- Kommentarzeilen werden mit "!", "#" oder ";" eingeleitet.

Sektion

Der Name der Sektion ist **[KEYS]**. Hier ordnen Sie die Namen den Schlüssel/Schlössern nach folgendem Schema zu:

KEYxx=Name

xx: Nummer des Schlüssels (01 ... 31)

Beispiel:

```
[KEYS]
KEY01=Admin
KEY02=System
KEY03=
KEY04=
KEY05=service
...
KEY31=
```

**Namen für Schlösser
und Schlüssel**

Für die Namen gelten folgende Festlegungen:

- Maximal 15 alphanumerische Zeichen
 - Schloss und Schlüssel müssen denselben Namen haben
-

1.4 Belegung der Flash-Disk einsehen

Einleitung

Den Anwendungsbereich der internen Flash-Disk können Sie einsehen. Einzelheiten zur Belegung des Anwendungsbereichs finden Sie in diesem Kapitel.

Inhalt

Thema	Seite
Belegung der Flash-Disk.....	25

Belegung der Flash-Disk

Infodatei

Sie können den Bereich mit den Anwendungsdaten auf der internen Flash-Disk einsehen.

Die Belegung des Bereichs mit den Anwendungsdaten lesen Sie aus der Datei **/System/flashdiskinfo.txt**.

Beispiel

Das Beispiel zeigt die mögliche Belegung der Flash-Disk einer JetControl 340 (4 MB):

```
Name : flash disk
Date : 25.11.2008
Time : 15:04
Tracks: 64
```

```
Track 0: sectors: 128 (used: 81 / blocked: 47 / free: 0)
Track 1: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 2: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 3: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 4: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 5: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 6: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 7: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 8: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 9: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 10: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 11: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 12: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 13: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 14: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 15: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 16: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 17: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 18: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 19: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 20: sectors: 128 (used: 64 / blocked: 64 / free: 0)
Track 21: sectors: 128 (used: 85 / blocked: 43 / free: 0)
Track 22: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 23: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 24: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 25: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 26: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 27: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 28: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 29: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 30: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 31: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 32: sectors: 128 (used: 128 / blocked: 0 / free: 0)
Track 33: sectors: 128 (used: 105 / blocked: 0 / free: 23)
```

1 Dateisystem

```
Track 34: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 35: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 36: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 37: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 38: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 39: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 40: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 41: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 42: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 43: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 44: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 45: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 46: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 47: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 48: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 49: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 50: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 51: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 52: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 53: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 54: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 55: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 56: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 57: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 58: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 59: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 60: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 61: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 62: sectors: 128 (used: 0 / blocked: 0 / free: 128)
Track 63: sectors: 128 (used: 0 / blocked: 0 / free: 128)
```

```
Total: sectors: 8192 (used: 4175 / blocked: 154 / free: 3863)
```

```
Used   : 2120900 byte
Blocked: 78232 byte
Free   : 1962404 byte
Total  : 4161536 byte
```

Elemente der Infodatei

Tracks und Sektoren bilden die Verwaltungseinheiten der Flash-Disk. Die Infodatei enthält folgende Elemente:

Element	Beschreibung
Name	Fester Name der Flash-Disk
Date/Time	Zeitpunkt, wann die Flash-Disk zuletzt formatiert wurde
Tracks	Gesamtzahl der Tracks
Track xx: sectors: 128	Belegung der Sektoren eines Tracks
Total: sectors:	Gesamtstatistik der Sektoren
Used	Gesamtzahl der belegten Bytes
Blocked	Gesamtzahl der blockierten Bytes
Free	Gesamtzahl der freien Bytes
Total	Gesamtgröße der Flash-Disk

Zustände des Sektors

Die kleinste Verwaltungseinheit der Flash-Disk, der Sektor, kann folgende Zustände einnehmen:

Zustand	Bedeutung
used	Der Sektor ist durch Daten belegt.
blocked	Der Sektor ist nicht mehr belegt, kann aber aus Verwaltungsgründen noch nicht wieder benutzt werden.
free	Der Sektor ist nicht belegt und kann benutzt werden.

1.5 Formatieren und Prüfen

Einleitung

Dieses Kapitel beschreibt:

- Flash-Disk formatieren
- SD-Karte formatieren
- SD-Karte prüfen
- USB-Stick formatieren
- USB-Stick prüfen

Die interne Flash-Disk braucht nicht durch eine separate Funktion geprüft werden, da sie auf maximale Sicherheit der Verwaltungsstrukturen ausgelegt ist.

Funktionsweise

Wenn das Gerät bootet, prüft das Betriebssystem den Inhalt des Steuerregisters. Das Steuerregister ist Bestandteil des Dateisystems. Abhängig vom Wert in diesem Register werden die folgenden Funktionen ausgeführt:

- Flash-Disk formatieren
 - SD-Karte formatieren
 - USB-Stick formatieren
 - SD-Karte überprüfen
 - USB-Stick überprüfen
-

Registernummer

Die Nummer des Steuerregisters ist 202936.

Inhalt

Thema	Seite
Flash-Disk formatieren.....	29
SD-Karte formatieren.....	30
USB-Stick formatieren.....	31
SD-Karte prüfen.....	32
USB-Stick überprüfen.....	33

Flash-Disk formatieren

Einleitung

In den folgenden Fällen ist es erforderlich, die Flash-Disk neu zu formatieren:

- Wenn Sie eine Betriebssystemversion mit einem geänderten Format der Flash-Disk einspielen
- Wenn Informationen zur Verwaltung der Flash-Disk zerstört wurden

Auswirkungen

- Alle Dateien und Verzeichnisse des Anwenderbereichs werden gelöscht.
- Systemdateien und Systemverzeichnisse werden nicht beeinflusst.

Flash-Disk formatieren

Damit das Gerät die interne Flash-Disk formatiert, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Steuerregister 202936 des Dateisystems mit dem Wert -999720373 (0xc4697a4b).
3	Schalten Sie das Gerät aus.
4	Schalten Sie das Gerät ein.

Ergebnis: Während des Boot-Vorgangs wird die Flash-Disk formatiert und das Steuerregister 202936 auf **0** gesetzt.

SD-Karte formatieren

Einleitung

Im folgenden Fall ist es erforderlich, die SD-Karte neu zu formatieren:

- Wenn Informationen zur Verwaltung der SD-Karte zerstört wurden

Auswirkungen

Alle Dateien und Verzeichnisse auf der SD-Karte werden gelöscht.

SD-Karte formatieren

Damit das Gerät die SD-Karte formatiert, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Steuerregister 202936 des Dateisystems mit dem Wert -748362163 (0xd364e64d).
3	Schalten Sie das Gerät aus.
4	Schalten Sie das Gerät ein.

Ergebnis: Während des Boot-Vorgangs wird die SD-Karte formatiert und das Steuerregister 202936 auf **0** gesetzt.

USB-Stick formatieren

Einleitung Unter Umständen kann es notwendig sein, den USB-Stick neu zu formatieren. Dies kann dann der Fall sein, wenn Informationen zur Verwaltung des USB-Sticks zerstört wurden.

Auswirkungen Alle Dateien und Verzeichnisse auf dem USB-Stick werden gelöscht.

Formatieren Führen Sie folgende Schritte aus, damit der USB-Stick formatiert wird:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Steuerregister des Dateisystems mit dem Wert (0x8f3d5185).
3	Schalten Sie das Gerät aus.
4	Schalten Sie das Gerät ein.

Ergebnis: Während der Boot-Phase des Geräts wird der USB-Stick formatiert und das Steuerregister auf 0 gesetzt.

SD-Karte prüfen

Einleitung

Im folgenden Fall ist es erforderlich, die SD-Karte auf Fehler zu prüfen:

- Wenn das Gerät während eines Zugriffs auf die SD-Karte ausgeschaltet wurde

Auswirkungen

- Alle Dateien und Verzeichnisse auf der SD-Karte werden geprüft und Fehler ggf. korrigiert.
Anschließend sind die Verwaltungsstrukturen auf der SD-Karte in einem konsistenten Zustand.
- Abhängig von der Größe der SD-Karte und der Anzahl der Verzeichnisse und Dateien verlängert sich die Boot-Phase bis auf mehrere Minuten.

SD-Karte prüfen

Damit das Gerät die SD-Karte prüft, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Steuerregister 202936 des Dateisystems mit dem Wert 748371092 (0x2c9b3c94).
3	Schalten Sie das Gerät aus.
4	Schalten Sie das Gerät ein.

Ergebnis: Während des Boot-Vorgangs prüft das Gerät die SD-Karte. Der Wert im Steuerregister bleibt erhalten, so dass die Prüfung bei jedem Neustart des Geräts durchgeführt wird.

Einschränkungen

Die Funktion **repariert** die Verwaltungsstrukturen der SD-Karte. Sie können die SD-Karte somit weiter verwenden. Zu beachten ist, dass z. B. das Gerät die Daten einer unvollständig geschriebenen Datei nicht immer restaurieren kann.

USB-Stick überprüfen

Einleitung

Unter Umständen kann es notwendig sein, den USB-Stick auf Fehler zu überprüfen. Dies kann dann der Fall sein, wenn das Gerät ausgeschaltet wurde während auf den USB-Stick zugegriffen wurde.

Auswirkungen

- Alle Dateien und Verzeichnisse auf dem USB-Stick werden überprüft und Fehler gegebenenfalls korrigiert. Anschließend sind die Verwaltungsstrukturen auf dem USB-Stick in einem konsistenten Zustand.
- Abhängig von der Größe des USB-Sticks und der Anzahl der Verzeichnisse und Dateien kann die Boot-Phase bis auf mehrere Minuten verlängert werden.

Überprüfung

Führen Sie folgende Schritte aus, damit das Gerät den USB-Stick überprüft:

Schritt	Vorgehen
1	Schalten Sie das Gerät ein.
2	Beschreiben Sie das Steuerregister des Dateisystems mit dem Wert (0x17dbd42a).
3	Schalten Sie das Gerät aus.
4	Schalten Sie das Gerät ein.

Ergebnis: Während der Boot-Phase des Geräts wird der USB-Stick überprüft. Der Wert im Steuerregister bleibt erhalten, so dass die Überprüfung bei jedem Neustart des Geräts durchgeführt wird.

Einschränkungen

Die Funktion **repariert** nur die Verwaltungsstrukturen des USB-Sticks, damit dieser weiter verwendet werden kann. Zum Beispiel können die Daten einer unvollständig geschriebenen Datei nicht immer restauriert werden.

2 FTP-Server

Einleitung

Der FTP-Server erlaubt den Zugriff auf Verzeichnisse und Dateien mit einem FTP-Client. Die Dateien können auf folgenden Speichermedien abgespeichert sein:

- Interne Flash-Disk
- SD-Karte
- USB-Stick

Dieses Kapitel beschreibt den Vorgang der Anmeldung und die vom FTP-Server unterstützten Kommandos.

FTP-Clients

Außer dem Kommandozeilen-FTP-Client, wie er bei vielen PC-Betriebssystemen mitgeliefert wird, können Sie auch grafische FTP-Tools verwenden.

Anzahl möglicher Verbindungen

Der FTP-Server verwaltet maximal vier gleichzeitig geöffnete FTP-Verbindungen.

Jeder weitere FTP-Client, der sich versucht mit dem FTP-Server zu verbinden, erhält auf seine Anfrage zum Aufbau einer Verbindung keine Antwort.

Unterstützte Kommandos

Der FTP-Server unterstützt die standardisierten Kommandos. Sie finden diese unter:

- Help-Menü des FTP-Servers; mit FTP-Server verbinden und als Kommando *help* oder *help binary* eingeben
- Im Internet nach FTP und Kommandos suchen

Wenn Sie sich nicht um die Kommandos kümmern wollen, empfehlen wir Ihnen ein FTP-Programm wie TotalCommander zu verwenden.

R 202930

Webstatus

Das Webstatusregister zeigt bitkodiert die Funktionen an, die verfügbar sind.

Bedeutung der Bits

Bit 0	FTP-Server
1 =	verfügbar

Bit 1	HTTP-Server
1 =	verfügbar

Bit 2	E-Mail
1 =	verfügbar

Bit 3	Datendateifunktion
1 =	verfügbar

Bit 4	Modbus/TCP
1 =	vorhanden

Bit 5	Modbus/TCP
1 =	verfügbar

Bit 7	FTP-Client
1 =	verfügbar

Modulregistereigenschaften

Zugriff	Lesen
Wert nach Reset	Je nach bestellter Option

Kenntnisse des Programmierers

Dieses Kapitel setzt folgende Kenntnisse voraus:

- Dateisystem
- IP-Netzwerke
- FTP-Kommandos

Inhalt

Thema	Seite
Anmeldung	37
Beispiel: Windows FTP-Client	38

Anmeldung

Anmeldung

Um auf das Dateisystem über FTP zugreifen zu können, muss sich der FTP-Client bei der Aufnahme der Kommunikation wie folgt anmelden:

- Bei **Servername** geben Sie die IP-Adresse des Geräts ein.
 - Bei **Benutzername** geben Sie den Benutzernamen ein, z. B. *admin*.
 - Bei **Passwort** geben Sie das Passwort ein, z. B. *admin*.
-

Auslieferungszustand

Im Auslieferungszustand ist ein Benutzer (User) eingerichtet:

NAME=admin

PW=admin

Beispiel: Windows FTP-Client

Aufgabe

Bearbeiten Sie mit dem FTP-Client von z. B. Windows XP die folgenden Aufgaben:

- Aufruf des FTP-Clients mit Verbindung öffnen und Eingabe der IP-Adresse
- Login mit Benutzer *admin* und Passwort *admin*
- Anzeigen des Inhalts des aktuellen Verzeichnisses mit *dir*
- Wechseln in das Verzeichnis *App* mit *cd app*
- Anzeigen des Inhalts des aktuellen Verzeichnisses mit *dir*
- Mögliche Kommandos anzeigen lassen mit *help*.
- Beenden der Sitzung und des FTP-Clients mit *bye*

Vorgehen

```
C:\WINDOWS\system32\cmd.exe
C:\>ftp 192.168.10.15
Verbindung mit 192.168.10.15 wurde hergestellt.
220 Service ready for new user.
Benutzer (192.168.10.15:(none)): admin
331 User name okay, need password.
Kennwort:
230 User logged in, proceed.
ftp> dir
200 Command okay.
150 File status okay; about to open data connection.
01-01-98 14:00 <DIR> App
01-01-98 14:00 <DIR> Data
01-01-06 14:00 <DIR> Windows
01-01-98 14:00 <DIR> System
226 Closing data connection.
FTP: 64d Bytes empfangen in 0,33Sekunden 0,54KB/s
ftp> cd data
250 Requested file action okay, completed.
ftp> dir
200 Command okay.
150 File status okay; about to open data connection.
08-22-13 03:02 94500 visual.iop
226 Closing data connection.
FTP: 64d Bytes empfangen in 0,16Sekunden 0,31KB/s
ftp> help
Befehle können abgekürzt werden. Befehle sind:
?          delete          literal          prompt          send
?          debug           ls              put             status
append    dir             mdelete        pwd            trace
ascii     disconnect     mdir           quit           type
bell      get            mget          quote         user
binary    glob           mkdir         recu         verbose
bye       hash          mls           remotehelp
cd        help          mput          rename
close    lcd           open          rmdir
ftp> bye
221 Service closing control connection.
C:\>
```

3 HTTP-Server

Einleitung	<p>Ein Standardbrowser reicht für den Zugriff auf den HTTP-Server aus. Mit dem Browser können Dateien, die per FTP auf die Steuerung geladen wurden, gelesen und angezeigt werden.</p> <p>Um Zugriff auf bestimmte Seiten zu erhalten, kann es, je nach Konfiguration des Dateisystems, erforderlich sein, Benutzername und Passwort einzugeben.</p>				
Vorgegebene Dateinamen	Die vorgegebenen Dateinamen sind index.htm und index.html .				
Unterstützte Dateitypen	<p>Folgende Dateitypen werden unterstützt:</p> <ul style="list-style-type: none"> ▪ *.htm, *.html, *.shtml ▪ *.txt, *.ini ▪ *.gif, *.tif, *.tiff, *.bmp, *.wbmp ▪ *.jpg, *.jpe, *.jpeg, *.png ▪ *.xml ▪ *.js, *.jar, *.java, *.class, *.cab ▪ *.ocx ▪ *.pdf, *.zip, *.doc, *.rtf ▪ *.css ▪ *.wml, *.wmlc, *.wmls, *.wmlsc ▪ *.ico, *.svg 				
Funktion HTTP-Server aktivieren	<p>Bei der Steuerung <Produktname einfügen> ist die Funktion HTTP-Server immer aktiviert.</p> <p>Das Bit 1 im Webstatusregister 202930 ist dann immer gesetzt.</p>				
Kenntnisse des Programmierers	<p>Dieses Kapitel setzt folgende Kenntnisse voraus:</p> <ul style="list-style-type: none"> ▪ Dateisystem ▪ IP-Netzwerke 				
Inhalt	<table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">Thema</th> <th style="text-align: right;">Seite</th> </tr> </thead> <tbody> <tr> <td>Server Side Includes</td> <td style="text-align: right;">40</td> </tr> </tbody> </table>	Thema	Seite	Server Side Includes	40
Thema	Seite				
Server Side Includes	40				

3.1 Server Side Includes

Einleitung	Der HTTP-Server enthält die Funktionalität <i>Server Side Includes (SSI)</i> . Mit Hilfe dieser Funktion sind in einer HTML-Seite aktuelle Echtzeit-Steuerungswerte darstellbar.								
Regeln	Wenn Sie aktuelle Echtzeit-Steuerungswerte in die HTML-Seite eintragen wollen, müssen Sie am Beginn dieser HTML-Seite den Tag Name Space angeben. Im Tag Name Space wird der verwendete Name Space definiert. Im Bodybereich der HTML-Seite folgen die Data -Tags.								
Echtzeit-Steuerungswerte aktualisieren	Wenn die HTML-Seite in den Browser geladen wird, ersetzt der HTTP-Server die Data -Tags einmalig durch aktuelle Echtzeit-Steuerungswerte. Um die Werte zu aktualisieren, muss die HTML-Seite immer wieder neu geladen werden. Angestoßen wird der Ladevorgang vom Anwender, indem er im Browser die Adresse der Steuerung und den Namen der gewünschten Seite einträgt, z. B. <i>http://192.168.10.209/Homepage/SSI/ssiTimeAndDate.htm</i> .								
Inhalt	<table><thead><tr><th>Thema</th><th>Seite</th></tr></thead><tbody><tr><td>Erster Eintrag in der HTML-Datei</td><td>41</td></tr><tr><td>Einfügen von Echtzeit-Steuerungswerten</td><td>42</td></tr><tr><td>Beispiel einer HTML-Seite</td><td>47</td></tr></tbody></table>	Thema	Seite	Erster Eintrag in der HTML-Datei	41	Einfügen von Echtzeit-Steuerungswerten	42	Beispiel einer HTML-Seite	47
Thema	Seite								
Erster Eintrag in der HTML-Datei	41								
Einfügen von Echtzeit-Steuerungswerten	42								
Beispiel einer HTML-Seite	47								

Erster Eintrag in der HTML-Datei

Aufbau

Der Tag **Name Space** muss der erste Eintrag in der HTML-Datei sein. Er hat den folgenden Aufbau:

```
<NS:DTAG xmlns:NS=http://jetter.de/ssi/jetcontrol/
```

Wobei **NS** den Name Space darstellt. Der Name Space besteht aus einer Zeichenfolge mit einer maximalen Länge von 63 Zeichen.

Der hier eingeführte Name Space wird bei den folgenden **Data** Tags wieder verwendet. Die übrigen Teile der Zeile sind festgelegt und müssen genauso angegeben werden.

In den folgenden Beispielen ist der verwendete Name Space **JC**.

Einfügen von Echtzeit-Steuerungswerten

Einleitung

Aktuelle Echtzeit-Steuerungswerte werden über Tag-Funktionen in die Parameter der Sektionen integriert. Damit können Zustände von Registern, Textregistern, Eingängen, Ausgängen und Merkern angezeigt werden.

Tag-Kennzeichnung

Alle Tags beginnen und enden mit bestimmten Zeichenfolgen. Zwischen diesen Tag-Begrenzern sind die Variablen definiert.

Kennzeichnung	Zeichenfolge
Beginn eines Tags	<JC:DTAG
Ende eines Tags	/>

Variablendefinition

Die Variablendefinition in einem Tag enthält Attribute, die z. B. die Darstellung des Variablenwerts festlegen.

name

Beschreibung	Variablenname
Bemerkungen	Kennbuchstabe gefolgt von der Variablennummer
Beispiel	name="R1000023"

type

Beschreibung	Variablentyp der Darstellung
Beispiel	type="REAL"

format

Beschreibung	Darstellungsformat
Bemerkungen	Siehe Formatdefinition
Beispiel	format="+0#####.###"

factor

Beschreibung	Faktor, mit dem der Echtzeit-Steuerungswert multipliziert wird
Bemerkungen	Die Multiplikation geschieht vor der Addition des Offsets
Beispiel	factor="1.5"

offset

Beschreibung	Wert, der zum Echtzeit-Steuerungswert addiert wird
Bemerkungen	Die Addition geschieht nach der Multiplikation mit dem Faktor
Beispiel	offset="1000"

Formatdefinition

Die Darstellung der Variablen können Sie über das Attribut steuern.

- Die Anzahl der Stellen/Zeichen, mit denen eine Variable dargestellt wird, legt das Zeichen "#" fest.
- Durch Voranstellen einer "0" wird bei den Registertypen INT, INTX und REAL die Ausgabe von führenden Nullen festgelegt.
- Durch Voranstellen eines "+" wird bei den Registertypen INT und REAL die Ausgabe eines Vorzeichens festgelegt.
- Durch Voranstellen eines Leerzeichens wird bei den Registertypen INT und REAL die Ausgabe eines Leerzeichens für positive Werte festgelegt.

Register/Textregister

Der Variablenname beginnt mit einem großen "R" gefolgt von der Registernummer.

Folgende Typen sind möglich:

Typ	Darstellung
INT	Ganzzahl dezimal
INTX	Ganzzahl hexadezimal
INTB	Ganzzahl binär
BOOL	Registerinhalt = 0 --> Anzeige: 0 Registerinhalt != 0 --> Anzeige: 1
REAL	Gleitkommazahl dezimal
STRING	Textregister

Standardtyp: INT

Beispiel:

```
<JC:DTAG name="R1000250" type="REAL" format="+0#####.###"
factor="3.25" offset="500" />
```

Ergebnis:

Aufgrund dieses Befehls wird der Inhalt von Register 1000250 mit 3,25 multipliziert und zum Produkt 500 addiert. Im Webbrowser erscheint das Ergebnis mit Vorzeichen und mindestens fünf Vorkommastellen. Wenn erforderlich, werden führende Nullen hinzugefügt. Darüber hinaus werden drei Nachkommastellen angefügt.

Merker

Der Variablenname beginnt mit einem großen "F" gefolgt von der Merkernummer.

Folgende Typen sind möglich:

Typ	Darstellung
BOOL	Merker = 0 --> Anzeige: 0 Merker = 1 --> Anzeige: 1
STRING	Merker = 0 --> Anzeige: FALSE Merker = 1 --> Anzeige: TRUE

Standardtyp: BOOL

Beispiel:

```
<JC:DTAG name="F100" type="STRING" format="#" />
```

Ergebnis:

Der Zustand des Merkers 100 wird als String "T" oder "F" angezeigt.

Eingänge

Der Variablenname beginnt mit einem großen "I" gefolgt von der Eingangsnummer.

Folgende Typen sind möglich:

Typ	Darstellung
BOOL	Eingang = 0 --> Anzeige: 0 Eingang = 1 --> Anzeige: 1
STRING	Eingang = 0 --> Anzeige: OFF Eingang = 1 --> Anzeige: ON

Standardtyp: BOOL

Beispiel:

```
<JC:DTAG name="I201200308" type="STRING" />
```

Ergebnis:

Der Zustand des Eingangs 201200308 wird als String "ON" oder "OFF" angezeigt.

Ausgänge

Der Variablenname beginnt mit einem großen "O" gefolgt von der Ausgangsnummer.

Folgende Typen sind möglich:

Typ	Darstellung
BOOL	Ausgang = 0 --> Anzeige: 0 Ausgang = 1 --> Anzeige: 1
STRING	Ausgang = 0 --> Anzeige: OFF Ausgang = 1 --> Anzeige: ON

Standardtyp: BOOL

Beispiel:

```
<JC:DTAG name="O201100308" />
```

Ergebnis:

Der Zustand des Ausgangs 201100308 wird als "1" oder "0" eingefügt.

**Zugriff über
Zeigerregister**

Zugriff über ein Zeigerregister ist möglich durch Einfügen eines großen "P" vor dem Variablennamen. Es wird jeweils der Wert der Variablen angezeigt, deren Nummer dem Inhalt des im Variablennamen spezifizierten Registers entspricht.

Beispiele:

```
<JC:DTAG name="PR1000300" />
```

Ergebnis: Der Inhalt des Registers, dessen Nummer im Register 1000300 enthalten ist, wird angezeigt.

```
<JC:DTAG name="PF1000300" />
```

Ergebnis: Der Zustand des Merkers, dessen Nummer im Register 1000300 enthalten ist, wird angezeigt.

```
<JC:DTAG name="PI1000300" />
```

Ergebnis: Der Zustand des Eingangs, dessen Nummer im Register 1000300 enthalten ist, wird angezeigt.

```
<JC:DTAG name="PO1000300" />
```

Ergebnis: Der Zustand des Ausgangs, dessen Nummer im Register 1000300 enthalten ist, wird angezeigt.

**Zugriff über
Zeigerregister und Offset**

Zusätzlich zum Wert aus dem Zeigerregister kann noch ein konstanter Wert oder ein weiterer Registerinhalt addiert werden, um die Nummer der anzuzeigenden Variablen zu bilden.

Beispiele:

```
<JC:DTAG name="PR1000300 + 100" />
```

Ergebnis: Der Inhalt des Registers, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Wert 100 ergibt, wird angezeigt.

```
<JC:DTAG name="PR1000300 + R1000100" />
```

Ergebnis: Der Inhalt des Registers, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Inhalt von Register 1000100 ergibt, wird angezeigt.

```
<JC:DTAG name="PF1000300 + 100" />
```

Ergebnis: Der Zustand des Merkers, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem 100 ergibt, wird angezeigt.

```
<JC:DTAG name="PF1000300 + R1000100" />
```

Ergebnis: Der Zustand des Merkers, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Inhalt von Register 1000100 ergibt, wird angezeigt.

```
<JC:DTAG name="PI1000300 + 100" />
```

Ergebnis: Der Zustand des Eingangs, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Wert 100 ergibt, wird angezeigt.

```
<JC:DTAG name="PI1000300 + R1000100" />
```

Ergebnis: Der Zustand des Eingangs, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Inhalt von Register 1000100 ergibt, wird angezeigt.

```
<JC:DTAG name="PO1000300 + 100" />
```

Ergebnis: Der Zustand des Ausgangs, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Wert 100 ergibt, wird angezeigt.

```
<JC:DTAG name="PO1000300 + R1000100" />
```

Ergebnis: Der Zustand des Ausgangs, dessen Nummer sich aus der Addition des Inhalts von Register 1000300 und dem Inhalt von Register 1000100 ergibt, wird angezeigt.

Beispiel einer HTML-Seite

Aufgabe	Binden Sie aktuelle Echtzeit-Steuerungswerte in eine HTML-Seite ein. Die HTML-Seite soll dann mit Hilfe der Funktionalität <i>Server Side Includes</i> mit einem Browser darstellbar sein.
Vorgehen	<pre> <JC:DTAG xmlns:JC="http://jetter.de/ssi/jetcontrol" /> <html> <head> <meta http-equiv="Content-Type" content="text/html; charset=windows-1252"> <meta name="GENERATOR" content="Microsoft FrontPage 4.0"> <meta name="ProgID" content="FrontPage.Editor.Document"> <title>Index</title> </head> <body> Hallo Welt, &nbsp; <p>So können aktuelle Steuerungswerte in eine html-Seite eingebunden werden:&nbsp;</p> <p>Register 201000 = <JC:DTAG name="R201000" type = INT format="+####" />, oder Hex: 0x<JC:DTAG name="R201000" type="INTX" format="0###" />, oder lieber so <JC:DTAG name="R201000" type="BOOL" />, wenn man nur boolsch abfragt. Es geht aber auch binär: <JC:DTAG name="R201000" type="INTB" format="#####" />b.&nbsp;</p> <p>Strings könnte man auch definieren "<JC:DTAG name="R201000" type="STRING" />".&nbsp;</p> <p>So sieht eine Realzahl aus: <JC:DTAG name="R1001500" type="REAL" /> oder so <JC:DTAG name="R1001500" type="REAL" factor="1.3" format="###.###" />.&nbsp;</p> <p>Der Wert eines Merkers wird so dargestellt: <JC:DTAG name="F10" /> oder <JC:DTAG name="F10" type="STRING" />.&nbsp;</p> <p>Bei den Ein- und Ausgängen erfolgt es analog: <JC:DTAG name="I100000205" type="BOOL" /> bzw. <JC:DTAG name="I100000205" type="STRING" />.&nbsp;</p> <p>R201000 = <JC:DTAG name="R201000" type="INT" format="+0#####" />&nbsp;</p> <p>Viele Grüße&nbsp;</p> <p>Dein JetControl</p> </body> </html> </pre>
Speicherort	Speichern Sie die HTML-Seite im Dateisystem der Steuerung ab.

4 FTP-Client

Der FTP-Client	Der FTP-Client ermöglicht den Zugriff auf Dateien und Verzeichnisse eines fernen Netzwerkteilnehmers aus dem Anwendungsprogramm heraus. Der FTP-Client kommuniziert hierzu mit dem FTP-Server dieses Netzwerkteilnehmers.
Funktionen	<p>Folgende Funktionen sind dabei möglich:</p> <ul style="list-style-type: none">▪ Anlegen von Verzeichnissen im fernen Dateisystem▪ Löschen von Verzeichnissen im fernen Dateisystem▪ Kopieren von Dateien vom lokalen Dateisystem zum fernen Dateisystem▪ Kopieren von Dateien vom fernen Dateisystem zum lokalen Dateisystem
Voraussetzungen	<ul style="list-style-type: none">▪ Die Funktionalität des FTP-Clients setzt grundsätzliche Kenntnisse über FTP-Verbindungen und Dateisysteme voraus.▪ Die IP-Adresse des FTP-Servers muss bekannt sein.▪ Wenn die IP-Adresse des FTP-Servers nicht bekannt ist, muss die Namensauflösung über einen DNS-Server möglich sein.▪ Die Parameter für Anmeldung am FTP-Server (User-Name und Passwort) müssen bekannt sein.▪ Für die Programmierung muss JetSym ab der Version 4.3 verwendet werden.
Verarbeitung im Anwendungsprogramm	<ul style="list-style-type: none">▪ Die Steuerung führt immer nur gleichzeitig einen FTP-Zugriff aus.▪ Der entsprechende Task im Anwendungsprogramm bleibt an dem Befehl stehen, bis der Zugriff abgeschlossen ist.▪ Während dieser Zeit werden andere Tasks des Anwendungsprogramms bearbeitet.▪ Während ein FTP-Zugriff eines Tasks gerade läuft, werden andere Tasks, die einen FTP-Befehl aufrufen, blockiert, bis der laufende Zugriff abgeschlossen ist.

R 202930

Webstatus

Das Webstatusregister zeigt bitkodiert die Funktionen an, die verfügbar sind.

Bedeutung der Bits

Bit 0 FTP-Server
1 = verfügbar

Bit 1 HTTP-Server
1 = verfügbar

Bit 2 E-Mail
1 = verfügbar

Bit 3 Datendateifunktion
1 = verfügbar

Bit 4 Modbus/TCP
1 = vorhanden

Bit 5 Modbus/TCP
1 = verfügbar

Bit 7 FTP-Client
1 = verfügbar

Modulregistereigenschaften

Zugriff	Lesen
Wert nach Reset	Je nach bestellter Option

Inhalt

Thema	Seite
Programmierung	51
Register.....	70

4.1 Programmierung

Einleitung

Der FTP-Client ermöglicht, vom Anwendungsprogramm auf Dateien und Verzeichnisse eines Netzwerkteilnehmers zuzugreifen. Verwendung finden hierbei Aufrufe von Funktionen, die im Sprachumfang der Steuerung enthalten sind. Führen Sie zur Programmierung folgende Schritte aus:

Schritt	Vorgehen
1	FTP-Client initialisieren
2	Verbindungen zu FTP-Servern öffnen
3	Daten übertragen
4	Verbindungen schließen

Einschränkungen

Während die Steuerung gerade eine der Funktionen des FTP-Clients bearbeitet, dürfen die Tasks, die den FTP-Client bedienen, nicht durch `TaskBreak` angehalten oder durch `TaskRestart` neu gestartet werden. Sonst beendet die Steuerung diese Funktion nicht und damit werden alle weiteren Funktionsaufrufe des FTP-Client blockiert.

Inhalt

Thema	Seite
FTP-Client initialisieren.....	52
Verbindung zum FTP-Server öffnen	53
Verbindung schließen	55
Datei lesen.....	56
Datei schreiben.....	58
Datei löschen	60
Verzeichnis wechseln	62
Verzeichnis anlegen	64
Verzeichnis löschen.....	66
Aktuelles Verzeichnis ermitteln.....	68

FTP-Client initialisieren

Einleitung Bei jedem Anwendungsprogrammstart muss mindestens einmal der FTP-Client initialisiert werden.

Funktionsdeklaration `Function FtpInitialize():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 := FtpInitialize();
```

Funktionsweise Die Steuerung arbeitet die Funktion in folgenden Stufen ab:

Stufe	Beschreibung
1	Die Steuerung schließt alle geöffneten Verbindungen des FTP-Client.
2	Die Steuerung initialisiert alle betriebssysteminternen Datenstrukturen des FTP-Client.

Verbindung zum FTP-Server öffnen

Einleitung

Bevor Daten gesendet oder empfangen werden können, muss eine Verbindung zu einem FTP-Server geöffnet werden. Hierbei erfolgt die Anmeldung mit einem Benutzernamen und Passwort (Login).

Funktionsdeklaration

```
Function FtpConnect(Const Ref ServerAddr: String,
                   Const Ref UserName: String,
                   Const Ref PassWord: String):Handle;
```

Funktionsparameter

Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
ServerAddr	IP-Adresse oder Name	Namensauflösung über DNS-Server
UserName	Benutzername für die Anmeldung	Login
PassWord	Passwort für die Anmeldung	Login

Rückgabewert

Bei einem positiven Rückgabewert konnte die Verbindung geöffnet und die Anmeldung erfolgreich durchgeführt werden. Bei einem Rückgabewert von **0** 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 folgenden Funktionen mitgegeben werden: <ul style="list-style-type: none"> ▪ Verbindung schließen ▪ Datei vom FTP-Server lesen ▪ Datei auf dem FTP-Server schreiben ▪ Datei auf dem FTP-Server löschen ▪ Verzeichnis auf dem FTP-Server wechseln ▪ Verzeichnis auf dem FTP-Server anlegen ▪ Verzeichnis auf dem FTP-Server löschen ▪ Aktuelles Verzeichnis auf dem FTP-Server ermitteln
0	Fehler beim Verbindungsaufbau oder der Anmeldung beim FTP-Server

Funktionsweise

Der Task bleibt bei der Programmzeile stehen, bis die Verbindung aufgebaut wurde oder der Timeout des FTP-Clients abgelaufen ist.

Die Funktion läuft in folgenden Stufen ab:

Stufe	Beschreibung	
1	Die Steuerung versucht eine TCP/IP-Verbindung zum FTP-Server aufzubauen.	
2	Wenn dann ...
	... der Netzwerkteilnehmer die Verbindung akzeptiert hat,	... weiter bei Stufe 3.
	... die Verbindung nicht aufgebaut werden konnte und der Timeout noch nicht abgelaufen ist,	... weiter bei Stufe 1.
	... ein Fehler aufgetreten ist oder der Timeout abgelaufen ist,	... wird die Funktion beendet und der Wert 0 zurückgeliefert.
3	Die Steuerung meldet sich mit dem Benutzernamen <i>Administrator</i> und dem Passwort <i>AdminPassword</i> am FTP-Server an.	
4	Wenn dann ...
	... der FTP-Server die Anmeldung akzeptiert hat,	... wird die Funktion beendet und ein positiver Wert als Handle zum weiteren Zugriff auf diese Verbindung zurückgeliefert.
	... der FTP-Server die Anmeldung z. B. wegen eines ungültigen Benutzernamens oder falschen Passworts nicht akzeptiert hat,	... wird die Funktion beendet und der Wert 0 zurückgeliefert.

Verwandte Themen

- **Verbindung schließen** (siehe Seite 55)
-

Verbindung schließen

Einleitung

Schließen Sie nicht mehr benötigte Verbindungen, da dadurch der Verwaltungsaufwand für die Verbindungen innerhalb der Steuerung reduziert wird.

Funktionsdeklaration

```
Function FtpDisconnect (FtpConnection:Handle) :Int;
```

Funktionsparameter

Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()

Rückgabewert

Folgende Rückgabewerte sind möglich:

Rückgabewert

0	Verbindung geschlossen und gelöscht
-1	Ungültiges Handle
-2, -3	Kommunikationsfehler, z. B. FTP-Server antwortet nicht

Verwandte Themen

- **Verbindung zum FTP-Server öffnen** (siehe Seite 53)

Datei lesen

Einleitung

Mit dieser Funktion wird der Inhalt einer Datei eines fernen Netzwerkteilnehmers in eine Datei im lokalen Dateisystem der Steuerung kopiert.

Funktionsdeklaration

```
Function FtpFileRead(FtpConnection:Handle,
                    Const Ref ServerFile: String,
                    Const Ref ClientFile: String):Int;
```

Funktionsparameter

Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
ServerFile	Dateiname	Name der Datei im Dateisystem des FTP-Servers, die die Steuerung lesen soll
ClientFile	Dateiname	Name der Datei, unter dem die Steuerung die gelesene Datei im lokalen Dateisystem speichern soll

Rückgabewert

Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** hat die Steuerung die Datei gelesen und lokal gespeichert.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2, -6	Fehler beim Speichern der lokalen Datei
-3, -5, -7, -8	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-4	Fehlermeldung vom FTP-Server, z. B. Datei nicht vorhanden

Funktionsweise

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Die Steuerung muss zuerst z. B. die Datei **ServerTestFile.txt** lesen und sie in ihrem lokalen Dateisystem z. B. unter dem Namen **LocalTestFile.txt** speichern.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er den Inhalt der Datei ServerTestFile.txt übermitteln soll.	
2	Die Steuerung empfängt den Inhalt der Datei ServerTestFile.txt .	
3	Die Steuerung schreibt den Inhalt in die Datei LocalTestFile.txt .	
4	Wenn dann ...
	... keine Fehler aufgetreten sind,	... wurde die Datei erfolgreich kopiert, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Dateinamen

- Der Funktionsparameter für die lokale Datei kann auch den Pfad zu dieser Datei enthalten, z. B. `'/Data/TestFiles/LocalTestFile.txt'`.
- Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für die Datei auf dem FTP-Server auch den Pfad zu dieser Datei enthalten. Sonst ist zuvor mit dem Befehl `FtpDirChange()` das Verzeichnis einzustellen.
- Das Dateisystem unterstützt beide Varianten.

Verwandte Themen

- **Datei schreiben** (siehe Seite 58)

Datei schreiben

Einleitung Mit dieser Funktion wird der Inhalt einer Datei im lokalen Dateisystem in eine Datei eines entfernten Netzwerkteilnehmers kopiert.

Funktionsdeklaration

```
Function FtpFileWrite (FtpConnection:Handle,
                      Const Ref ServerFile: String,
                      Const Ref ClientFile: String):Int;
```

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
ServerFile	Dateiname	Name der Datei, unter dem der FTP-Server die geschriebene Datei speichern soll
ClientFile	Dateiname	Name der Datei in dem lokalen Dateisystem, deren Inhalt die Steuerung an den FTP-Server senden soll

Rückgabewert Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** hat die Steuerung die Datei gelesen und auf das Dateisystem des fernen Netzwerkteilnehmers geschrieben.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2	Fehler beim Lesen der lokalen Datei, z. B. Datei nicht vorhanden
-3, -5, -8	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-4, -7	Fehlermeldung des FTP-Servers, z. B. Datei kann nicht angelegt werden

Funktionsweise

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Die Steuerung muss z. B. zuerst die Datei **LocalTestFile.txt** lesen und sie in das Dateisystem des fernen Netzwerkteilnehmers z. B. unter dem Namen **ServerTestFile.txt** schreiben.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er den Inhalt der Datei ServerTestFile.txt speichern soll.	
2	Die Steuerung sendet den Inhalt der Datei LocalTestFile.txt .	
3	Der FTP-Server schreibt den Inhalt in die Datei ServerTestFile.txt .	
4	Wenn dann ...
	... keine Fehler aufgetreten sind,	... wurde die Datei erfolgreich kopiert, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Dateinamen

- Der Funktionsparameter für die lokale Datei kann auch den Pfad zu dieser Datei enthalten, z. B. `'/Data/TestFiles/LocalTestFile.txt'`.
- Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für die Datei auf dem FTP-Server auch den Pfad zu dieser Datei enthalten. Sonst ist zuvor mit dem Befehl `FtpDirChange()` das Verzeichnis einzustellen.
- Das Dateisystem unterstützt beide Varianten.

Verwandte Themen

- **Datei lesen** (siehe Seite 56)

Datei löschen

Einleitung

Mit dieser Funktion kann eine Datei im Dateisystem eines fernen Netzwerkteilnehmers gelöscht werden.

Funktionsdeklaration

```
Function FtpFileRemove (FtpConnection:Handle,
                        Const Ref ServerFile: String):Int;
```

Funktionsparameter

Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
ServerFile	Dateiname	Name der Datei, die gelöscht werden soll

Rückgabewert

Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** konnte die Datei im Dateisystem des fernen Netzwerkteilnehmers gelöscht werden.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-3	Fehlermeldung vom FTP-Server, z. B. Datei nicht vorhanden

Funktionsweise

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Der FTP-Server muss zuerst die Datei **ServerTestFile.txt** löschen. Bitte beachten Sie: Der Dateiname dient nur als Beispiel.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er die Datei ServerTestFile.txt löschen soll.	
2	Der FTP-Server löscht die Datei ServerTestFile.txt .	
3	Wenn dann ...
	... keine Fehler aufgetreten sind,	... ist die Datei erfolgreich gelöscht, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Dateinamen

- Der Funktionsparameter für die lokale Datei kann auch den Pfad zu dieser Datei enthalten, z. B. `'/Data/TestFiles/LocalTestFile.txt'`.
 - Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für die Datei auf dem FTP-Server auch den Pfad zu dieser Datei enthalten. Sonst ist zuvor mit dem Befehl `FtpDirChange()` das Verzeichnis einzustellen.
 - Das Dateisystem unterstützt beide Varianten.
-

Verzeichnis wechseln

Einleitung Mit dieser Funktion kann das aktuelle Verzeichnis im Dateisystem eines fernen Netzwerkteilnehmers gewechselt werden.

Funktionsdeklaration

```
Function FtpDirChange (FtpConnection:Handle,
                      Const Ref ServerDir: String):Int;
```

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
ServerDir	Verzeichnisname	Name des Verzeichnisses, in das gewechselt werden soll

Rückgabewert Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** konnte das Verzeichnis gewechselt werden.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-3	Fehlermeldung des FTP-Servers, z. B. Verzeichnis nicht vorhanden

Funktionsweise Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Der FTP-Server muss zuerst das Verzeichnis wechseln.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er in ein Unterverzeichnis wechseln soll.	
2	Der FTP-Server wechselt das Verzeichnis.	
3	Wenn dann ...
	... keine Fehler aufgetreten sind,	... ist das neue Verzeichnis eingestellt, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verzeichnisnamen

- Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für das Verzeichnis auf dem FTP-Server auch den kompletten Pfad - über mehrere Verzeichnisebenen - zu diesem Verzeichnis enthalten.
 - Sonst muss zuvor mit dem Befehl `FtpDirChange()` von einer Verzeichnisebene zur nächsten verzweigt werden, bis das gewünschte Verzeichnis erreicht ist.
 - Das Dateisystem des Geräts unterstützt beide Varianten.
-

Verwandte Themen

- **Aktuelles Verzeichnis ermitteln** (siehe Seite 68)
-

Verzeichnis anlegen

Einleitung Mit dieser Funktion wird ein neues Verzeichnis im Dateisystem eines fernen Netzwerkteilnehmers angelegt.

Funktionsdeklaration `Function FtpDirCreate (FtpConnection:Handle,
Const Ref ServerDir: String):Int;`

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
ServerDir	Verzeichnisname	Name des Verzeichnisses, das angelegt werden soll

Rückgabewert Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** konnte das Verzeichnis im Dateisystem des fernen Netzwerkteilnehmers angelegt werden.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-3	Fehlermeldung vom FTP-Server, z. B. Verzeichnis bereits vorhanden

Funktionsweise Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Der FTP-Server muss zuerst ein Unterverzeichnis anlegen.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er ein Unterverzeichnis anlegen soll.	
2	Der FTP-Server legt das Verzeichnis an.	
3	Wenn dann ...
	... keine Fehler aufgetreten sind,	... ist das neue Verzeichnis angelegt, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verzeichnisnamen

- Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für das Verzeichnis auf dem FTP-Server auch den kompletten Pfad - über mehrere Verzeichnisebenen - zu diesem Verzeichnis enthalten.
 - Sonst muss zuvor mit dem Befehl `FtpDirChange()` von einer Verzeichnisebene zur nächsten verzweigt werden, bis das gewünschte Verzeichnis erreicht ist.
 - Das Dateisystem des Geräts unterstützt beide Varianten.
-

Einschränkungen im Dateisystem einer JetControl

Wenn Sie als Funktionsparameter ein Verzeichnis mit einem Pfad angeben, müssen alle Verzeichnisse bis zu dem, das Sie anlegen wollen, vorhanden sein. Ein rekursives Anlegen von mehreren Verzeichnissen wird nicht unterstützt.

Beispiel:

```
Result := FtpDirCreate(FtpHandle,  
                      '/DataFiles/TextFiles/Release');
```

Damit der Ordner **Release** im Verzeichnisbaum `/DataFiles/TextFiles` angelegt werden kann, müssen diese vorhanden sein.

Verwandte Themen

- **Verzeichnis löschen** (siehe Seite 66)
-

Verzeichnis löschen

Einleitung Mit dieser Funktion wird ein Verzeichnis im Dateisystem eines fernen Netzwerkteilnehmers gelöscht.

Funktionsdeklaration `Function FtpDirRemove (FtpConnection:Handle, Const Ref ServerDir: String):Int;`

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion <code>FtpConnect ()</code>
ServerDir	Verzeichnisname	Name des Verzeichnisses, das gelöscht werden soll

Rückgabewert Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** konnte das Verzeichnis im Dateisystem des fernen Netzwerkteilnehmers gelöscht werden.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-2	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-3	Fehlermeldung des FTP-Servers, z. B. Verzeichnis nicht vorhanden

Funktionsweise Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Der FTP-Server muss zuerst das Unterverzeichnis löschen.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er das Unterverzeichnis löschen soll.	
2	Der FTP-Server löscht das Unterverzeichnis.	
3	Wenn dann ...
	... keine Fehler aufgetreten sind,	... ist das Verzeichnis gelöscht, die Funktion beendet und der Wert 0 zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verzeichnisnamen

- Wenn das Dateisystem des entfernten Netzwerkteilnehmers dies unterstützt, kann der Funktionsparameter für das Verzeichnis auf dem FTP-Server auch den kompletten Pfad - über mehrere Verzeichnisebenen - zu diesem Verzeichnis enthalten.
 - Sonst muss zuvor mit dem Befehl `FtpDirChange()` von einer Verzeichnisebene zur nächsten verzweigt werden, bis das gewünschte Verzeichnis erreicht ist.
 - Das Dateisystem des Geräts unterstützt beide Varianten.
-

Verwandte Themen

- **Verzeichnis anlegen** (siehe Seite 64)
-

Aktuelles Verzeichnis ermitteln

Einleitung Mit dieser Funktion wird das aktuell eingestellte Verzeichnis im Dateisystem eines fernen Netzwerkteilnehmers ermittelt.

Funktionsdeklaration

```
Function FtpDirPrint (FtpConnection:Handle,
                    Ref str: String):Int;
```

Funktionsparameter Beschreibung der Funktionsparameter:

Parameter	Wert	Bemerkung
FtpConnection	Handle	Rückgabewert der Funktion FtpConnect ()
str	Adresse eines Strings	Aktuelles Verzeichnis mit Pfadangabe

Rückgabewert Bei einem negativen Rückgabewert ist ein Fehler aufgetreten. Bei einem Rückgabewert von **0** konnte das aktuell eingestellte Verzeichnis im Dateisystem eines fernen Netzwerkteilnehmers ermittelt werden.

Rückgabewert

0	Kein Fehler
-1	Ungültiges Handle
-3	Kommunikationsfehler, z. B. FTP-Server antwortet nicht
-4	Fehlermeldung des FTP-Servers
-5	Ungültige Rückmeldung vom Server

Funktionsweise

Der Task bleibt in folgenden Fällen bei diesem Funktionsaufruf stehen:

- Der FTP-Server muss zuerst das aktuell eingestellte Verzeichnis ermitteln.
- Ein Fehler ist aufgetreten.

Die Funktion wird in folgenden Stufen abgearbeitet:

Stufe	Beschreibung	
1	Die Steuerung schickt zum FTP-Server das Kommando, dass er das aktuell eingestellte Verzeichnis ermitteln soll.	
2	Der FTP-Server übermittelt der Steuerung das aktuell eingestellte Verzeichnis mit Pfadangabe.	
3	Wenn dann ...
	... keine Fehler aufgetreten sind,	... enthält die Variable den kompletten Pfad des aktuellen Verzeichnisses, die Funktion wird beendet und der Wert 0 wird zurückgeliefert.
	... Fehler aufgetreten sind,	... wird die Funktion beendet und ein negativer Wert zurückgeliefert.

Verwandte Themen

- **Verzeichnis wechseln** (siehe Seite 62)

4.2 Register

Einleitung

Dieses Kapitel beschreibt die Register der Steuerung, in denen Statusinformationen des FTP-Clients enthalten sind. 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	71
Registerbeschreibung	72

Registernummern

Einleitung

Die Daten jeweils einer Verbindung erscheinen in den Registern eines zusammenhängenden Registerblocks. Zwei weitere Register zeigen den Status des aktuell vom Anwendungsprogramm ausgeführten Befehls an. Die Basisregisternummer dieser Register ist steuerungsabhängig.

Registernummern

Basisregisternummer	Registernummern
320000	320000 ... 320101

Registernummer ermitteln

In diesem Kapitel sind jeweils nur die letzten drei Ziffern der Registernummer angegeben, z. B. MR 002. Addieren Sie zu dieser Modulregisternummer die Basisregisternummer des jeweiligen Geräts, z. B. 320000, um die vollständige Registernummer zu ermitteln.

Registerübersicht

Übersicht über die Modulregister des FTP-Clients:

Register	Beschreibung
MR 000	Anzahl offener Verbindungen
MR 002	Timeout in Sekunden
MR 003	Port-Nummer des FTP-Servers
MR 004	Index in der Verbindungstabelle
MR 005	Handle der Verbindung
MR 006	IP-Adresse des FTP-Servers
MR 007	Port-Nummer des FTP-Servers
MR 008	IP-Adresse des FTP-Clients
MR 009	Port-Nummer des FTP-Clients
MR 100	Status der FTP-Client-Bearbeitung
MR 101	Task-ID

Registerbeschreibung

Einleitung

Die geöffneten Verbindungen werden vom Betriebssystem der Steuerung in einer Liste verwaltet. Mit Hilfe der Modulregister MR 004 oder MR 005 werden die Verbindungsdaten einer Verbindung in die Register MR 006 bis MR 009 kopiert.

MR 000

Anzahl geöffneter Verbindungen

Der Wert in diesem Register zeigt an, wie viele Verbindungen zurzeit geöffnet sind.

Modulregistereigenschaften

Werte lesen	0 ... 2.147.483.647	Anzahl Verbindungen
-------------	---------------------	---------------------

MR 002

Timeout

In diesem Register stellen Sie den Timeout des FTP-Clients beim Zugriff auf den FTP-Server ein.

Modulregistereigenschaften

Werte	0 ... 2.147.483.647	in Sekunden
Wert nach Reset	20	

MR 003

Port-Nummer des FTP-Servers

Der Wert in diesem Register zeigt die IP-Port-Nummer des FTP-Servers an.

Modulregistereigenschaften

Werte	0 ... 65.535	
Wert nach Reset	21	

MR 004

Index in der Verbindungstabelle

In dieses Register wird der Index der Verbindungstabelle eingetragen. Wenn eine Verbindung an dem eingetragenen Index vorhanden ist, können Sie anschließend das Handle der Verbindung aus Modulregister MR 005 und die Verbindungsdaten aus den Modulregistern MR 006 bis MR 009 lesen.

Modulregistereigenschaften

Werte	0 ... [MR 000] - 1
Wert nach Reset	-1

MR 005**Handle der Verbindung**

In dieses Register wird das Handle der Verbindung eingetragen. Wenn eine Verbindung mit diesem Handle vorhanden ist, können Sie anschließend den Index der Verbindungstabelle aus Modulregister MR 004 und die Verbindungsdaten aus den Modulregistern MR 006 bis MR 009 lesen.

Modulregistereigenschaften

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

MR 006**IP-Adresse des FTP-Servers**

Der Wert in diesem Register zeigt die IP-Adresse des FTP-Servers an.

Modulregistereigenschaften

Zugriff	Lesen
Wird wirksam	Wenn MR 004 >= 0

MR 007**Port-Nummer des FTP-Servers**

Der Wert in diesem Register zeigt die Port-Nummer des FTP-Servers an.

Modulregistereigenschaften

Zugriff	Lesen
Wird wirksam	Wenn MR 004 >= 0

MR 008**IP-Adresse des FTP-Clients**

Der Wert in diesem Register zeigt die IP-Adresse des FTP-Clients an.

Modulregistereigenschaften

Zugriff	Lesen
Wird wirksam	Wenn MR 004 >= 0

MR 009

Port-Nummer des FTP-Clients

Der Wert in diesem Register zeigt die Port-Nummer des FTP-Clients an.

Modulregistereigenschaften

Zugriff	Lesen
Wird wirksam	Wenn MR 004 >= 0

MR 100

Status der FTP-Client-Bearbeitung

Die Ausführung des FTP-Clients Zugriffs kann hier mitverfolgt werden.

Modulregistereigenschaften

Werte	0	Es wird kein Zugriff ausgeführt
	1	Übergabe der Parameter an den FTP-Client der Steuerung
	2	Der FTP-Client kommuniziert mit dem FTP-Server
	3	Der Zugriff ist abgeschlossen
Zugriff	Lesen	

MR 101

Task-ID

Der Wert in diesem Register zeigt die ID des Tasks an, der gerade eine Funktion des FTP-Clients ausführt.

Modulregistereigenschaften

Werte	0 ... 99	Task-ID
	255	Kein Task führt gerade eine FTP-Funktion aus
Wert nach Reset	255	
Zugriff	Lesen	

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Einleitung	Dieses Kapitel beschreibt die AutoCopy-Funktion, die es ermöglicht, Dateien innerhalb der Steuerung und/oder zwischen der Steuerung und einem FTP-Server, den angeschlossenen Erweiterungsmodulen und einer Steuerung im Netzwerk zu kopieren. Hierzu erstellen Sie eine Kommandodatei, die zusammen mit den Daten, wahlweise auf der SD-Karte oder dem USB-Stick, abgelegt ist. Die Kommandodatei wird beim Booten von der Steuerung automatisch abgearbeitet.
Funktionen im lokalen Dateisystem	<p>Folgende Funktionen führt die AutoCopy-Funktion aus:</p> <ul style="list-style-type: none">▪ Register und Merker in eine Datei abspeichern▪ Register und Merker aus einer Datei einspielen▪ Verzeichnisse anlegen▪ Verzeichnisse löschen▪ Dateien kopieren▪ Dateien löschen
Funktionen im Dateisystem des FTP-Servers	<p>Folgende Funktionen führt die AutoCopy-Funktion aus:</p> <ul style="list-style-type: none">▪ Dateien vom FTP-Server kopieren▪ Dateien zum FTP-Server kopieren▪ Dateien löschen▪ Verzeichnis wechseln▪ Verzeichnis anlegen▪ Verzeichnis löschen
Anwendungsbereiche	<p>Für die AutoCopy-Funktion gibt es hauptsächlich folgende Anwendungsbereiche:</p> <ul style="list-style-type: none">▪ Wo Fernwartung nicht möglich ist.▪ Kein PC vor Ort vorhanden ist.▪ Der Anlagenbediener nicht in der Lage ist oder nicht in die Lage versetzt werden soll, Änderungen an der Anlage vorzunehmen. <p>Folgendes kann mit der AutoCopy-Funktion ausgeführt werden:</p> <ul style="list-style-type: none">▪ Änderung des Anwendungsprogramms▪ Änderung der Anwendungsdaten▪ Änderung der Steuerungskonfiguration▪ Betriebssystemupdate (Steuerung, Module am Systembus, Netzwerkteilnehmer)▪ Duplizieren eines Steuerungssystems

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Voraussetzungen

Für das automatische Kopieren von Steuerungsdaten müssen folgende Voraussetzungen erfüllt sein:

- Der Programmierer verfügt über Kenntnisse des Dateisystems.
- Der Programmierer verfügt über Grundkenntnisse der Anwendung von FTP.

config.ini - Beispiel

Das ist ein Beispiel für eine Konfigurationsdatei **config.ini** mit einem Eintrag *AutoCopyIni*.

```
;Copyright (c) 2009 by Jetter AG, Ludwigsburg, Germany
```

```
[IP]
Address      = 192.168.  1.  1
SubnetMask   = 255.255.255.  0
DefGateway   =  0.  0.  0.  0
DNSServer    =  0.  0.  0.  0
```

```
[HOSTNAME]
SuffixType   = 0
Name         = JetControl350
```

```
[PORTS]
JetIPBase    = 50000
JVMDebug     = 52000
```

```
[FILES]
AutoCopyIni  = /SD/project_name/autocopy.ini
```

AutoCopyIni - Hinweise

- Die AutoCopy-Funktion macht nur Sinn, wenn die zu kopierenden Daten auf der SD-Karte oder dem USB-Stick abgespeichert sind. Das Stammverzeichnis ist dann zwangsläufig */SD/* oder */USB/*.
- Die Datei **autocopy.ini** kann in einem beliebigen Verzeichnis sein.
- Statt **autocopy.ini** können Sie die Datei beliebig benennen.

Bezeichnung

Vollständiger Name bezeichnet in dieser Beschreibung den Namen der Datei oder des Verzeichnisses mit komplettem Pfad.

Inhalt

Thema	Seite
Funktionsweise	77
autocopy.ini - Aufbau	83
Protokolldatei	96
Datendateien	98

5.1 Funktionsweise

Einleitung

Dieses Kapitel beschreibt das Starten und Ausführen der AutoCopy-Funktion.

Inhalt

Thema	Seite
AutoCopy-Funktion starten.....	78
AutoCopy-Funktion ausführen.....	79
AutoCopy-Funktion beenden.....	82

AutoCopy-Funktion starten

Einleitung

Die AutoCopy-Funktion kann nur während der Boot-Phase (nach dem Einschalten) der Steuerung ausgeführt werden.

Voraussetzungen

Sie haben die Kommandodatei erstellt und im entsprechenden Verzeichnis abgelegt.

Wenn der Eintrag *AutoCopyIni* in der Konfigurationsdatei **config.ini** nicht vorhanden ist, ist der Name der Kommandodatei und das Verzeichnis wie folgt vorgeschrieben:

	Wert	Bemerkung
Dateiname	autocopy.ini	Alles Kleinbuchstaben
Verzeichnis	/SD/	Stammverzeichnis auf der SD-Karte
Verzeichnis	/USB/	Stammverzeichnis auf dem USB-Stick

- Die Datei **autocopy.ini** kann in einem beliebigen Verzeichnis sein.
- Statt **autocopy.ini** können Sie die Datei beliebig benennen.

In diesem Fall ist die Bedingung, dass die Konfigurationsdatei **config.ini** den Eintrag *AutoCopyIni* enthält. Dieser Eintrag definiert dann den Verzeichnis- und Dateinamen der Kommandodatei.

AutoCopy-Funktion starten

Um die AutoCopy-Funktion zu starten, führen Sie folgende Schritte aus:

Schritt	Vorgehen	
1	Schalten Sie die Steuerung aus.	
2	Wenn dann ...
	... Sie eine SD-Karte verwenden,	... stecken Sie die SD-Karte vollständig in den Steckplatz der SD-Karte.
	... Sie einen USB-Stick verwenden,	... stecken Sie den USB-Stick in die USB-Buchse der Steuerung.
3	Stellen Sie den Schalter in Stellung <i>LOAD</i> .	
4	Schalten Sie die Steuerung ein.	
5	Warten Sie bis die rote LED D1 an ist, sowie die grüne LED R und die gelbe LED SD und  langsam mit ca. 1 Hz blinken.	
⇒	Ergebnis: Die Steuerung führt die AutoCopy-Funktion aus.	
6	Warten Sie bis die rote LED D1 und die grüne LED R langsam mit ca. 1 Hz blinken.	
⇒	Ergebnis: Die AutoCopy-Funktion ist beendet.	

AutoCopy-Funktion ausführen

Einleitung

Während der Boot-Phase in der Betriebsart AutoCopy führt die Steuerung die Kommandos in der Kommandodatei aus.

Einschränkungen

In der Betriebsart AutoCopy gelten folgende Einschränkungen der Steuerungsfunktion:

- Die Steuerung führt das Anwendungsprogramm nicht aus.
- Die Kommunikation mit der Steuerung ist nicht möglich.
- Nach Beendigung der AutoCopy-Funktion ist ein Neustart der Steuerung erforderlich.

AutoCopy-Funktion ausführen

Das Betriebssystem der Steuerung bearbeitet die AutoCopy-Funktion in folgenden Stufen:

Stufe	Beschreibung
1	Die Steuerung öffnet die Kommandodatei, die im Eintrag <i>AutoCopyIni</i> in der Konfigurationsdatei <i>/System/config.ini</i> angegeben ist.
2	Die Steuerung liest die Werte aus der Sektion [OPTIONS].
3	Die Steuerung liest das Kommando und die Kommandoparameter aus der Sektion [COMMAND_1], bearbeitet es und schreibt das Ergebnis ggf. in die Protokolldatei.
4 ... n	Die Steuerung bearbeitet die weiteren Kommandos in aufsteigender Reihenfolge, bis zur Anzahl aus der Sektion [OPTIONS].
n+1	Die Steuerung ermittelt die Ergebnisstatistik aller Kommandos und schreibt sie in die Protokolldatei.

LEDs in der Betriebsart AutoCopy

Die Status-LEDs des Betriebssystems haben in der Boot-Phase der Steuerung folgende Zustände:

Stufe	Beschreibung					
1						
	R	E	D1	D2	SD/ 	Zustand
	 4Hz	 4Hz	 4Hz	 4Hz	 OFF	Reset

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

2	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input type="radio"/> OFF	<input type="radio"/> OFF	<input checked="" type="radio"/> ON	<input type="radio"/> OFF	Der Bootloader lädt und prüft das Betriebssystem. Bei Steuerungen ohne Bootloader: Die Steuerung initialisiert das Betriebssystem.
3	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input type="radio"/> OFF	<input type="radio"/> OFF	<input type="radio"/> OFF	<input type="radio"/> OFF	Das Betriebssystem liest den DIP-Schalter auf dem Backplane-Modul und prüft das Vorhandensein des Ethernet-Switches.
4	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input checked="" type="radio"/> ON	<input type="radio"/> OFF	<input type="radio"/> OFF	<input type="radio"/> OFF	Das Betriebssystem initialisiert die Echtzeituhr, Ethernet-Schnittstelle und das Dateisystem
5	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input checked="" type="radio"/> ON	<input checked="" type="radio"/> ON	<input type="radio"/> OFF	 / <input type="radio"/> OFF	Das Betriebssystem initialisiert die Module am JX3- und JX2-Systembus und die SD-Karte
6	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input type="radio"/> OFF	<input checked="" type="radio"/> ON	<input type="radio"/> OFF		Die Kommandodatei der AutoCopy-Funktion wird bearbeitet
7a	R	E	D1	D2	SD/ 	Zustand
	 1Hz	<input type="radio"/> OFF	 1Hz	<input type="radio"/> OFF	<input type="radio"/> OFF	Die AutoCopy-Funktion ist beendet; Fehler sind keine aufgetreten.

7b						
	R	E	D1	D2	SD/ 	Zustand
	 1Hz	 ON	 1Hz	<input type="radio"/> OFF	<input type="radio"/> OFF	Die AutoCopy-Funktion ist beendet; Fehler sind aufgetreten.

AutoCopy-Funktion beenden

Einleitung

Nur ein Neustart der Steuerung beendet die AutoCopy-Funktion.

Voraussetzung

Die Bearbeitung der AutoCopy-Kommandos ist beendet.

AutoCopy-Funktion beenden

Um die AutoCopy-Funktion zu beenden, führen Sie folgende Schritte aus:

Schritt	Vorgehen
1	Schalten Sie die Steuerung aus.
2	Sie können jetzt die SD-Karte oder den USB-Stick entfernen (nicht erforderlich).
3	Stellen Sie den Schalter in Stellung <i>RUN</i> oder <i>STOP</i> .
4	Schalten Sie die Steuerung ein.

Ergebnis: Die Steuerung bootet neu.

5.2 autocopy.ini - Aufbau

Einleitung	Dieses Kapitel beschreibt den Aufbau der Datei autocopy.ini und die verfügbaren Kommandos.								
Aufbau der Datei	Die Kommandodatei der AutoCopy-Funktion ist eine Textdatei, deren Einträge in Sektionen gruppiert sind: <ul style="list-style-type: none">▪ In diesen Sektionen setzen Sie Werte, mit denen die AutoCopy-Funktion arbeitet.▪ Leerzeilen können Sie beliebig einfügen.▪ Leiten Sie Kommentarzeilen mit "!", "#" oder ";" ein.								
Sektionen	Die Kommandodatei enthält zwei Sektionstypen: <ul style="list-style-type: none">▪ In der Sektion <i>[OPTIONS]</i> legen Sie grundsätzliche Einstellungen fest. Diese Sektion ist nur einmal vorhanden.▪ In den Sektionen <i>[COMMAND_#]</i> geben Sie die auszuführenden Kommandos an. Die Anzahl der Kommandosektionen ist auf 128 beschränkt.								
Inhalt	<table><thead><tr><th>Thema</th><th>Seite</th></tr></thead><tbody><tr><td>Die Sektion [OPTIONS]</td><td>84</td></tr><tr><td>Die Kommandosektionen</td><td>85</td></tr><tr><td>Beispiel für eine Kommandodatei</td><td>93</td></tr></tbody></table>	Thema	Seite	Die Sektion [OPTIONS]	84	Die Kommandosektionen	85	Beispiel für eine Kommandodatei	93
Thema	Seite								
Die Sektion [OPTIONS]	84								
Die Kommandosektionen	85								
Beispiel für eine Kommandodatei	93								

Die Sektion [OPTIONS]

Einleitung

In der Sektion [OPTIONS] legen Sie grundsätzliche Einstellungen fest. Diese Sektion ist nur einmal, vorzugsweise am Beginn der Datei, vorhanden.

Beispiel

```
[OPTIONS]
CommandCount = 14
LogFile      = /SD/autocopy.log
LogAppend   = 1
```

Elemente der Sektion

Die Sektion besteht aus den folgenden Elementen:

CommandCount

Im Beispiel	14
Funktion	Anzahl der folgenden Kommando-sektionen
Gültige Werte	> = 0
Nicht gültige Werte	< 0
Bei ungültigem Wert oder nicht vorhandenem Eintrag	0

LogFile

Im Beispiel	/SD/autocopy.log
Funktion	Vollständiger Name der Protokolldatei
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Dateinamen▪ Verzeichnis ist vorhanden
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Dateiname▪ Nicht vorhandenes Verzeichnis
Bei ungültigem Wert oder nicht vorhandenem Eintrag	Die Steuerung erstellt keine Protokolldatei.

LogAppend

Im Beispiel	1
Funktion	Legt fest, ob eine neue Protokolldatei erstellt oder an eine vorhandene angefügt wird
Gültige Werte	<ul style="list-style-type: none">▪ 0 = ggf. bestehende Datei löschen und neue Datei erstellen▪ 1 = an bestehende Datei anfügen; wenn keine Datei vorhanden ist, erstellt die Steuerung eine neue Protokolldatei
Nicht gültige Werte	<ul style="list-style-type: none">▪ < 0▪ > 1
Bei ungültigem Wert oder nicht vorhandenem Eintrag	Die Steuerung erstellt die Protokolldatei neu.

Die Kommandosektionen

Einleitung

In diesen Sektionen geben Sie die Befehle an, die die Funktion AutoCopy ausführt.

Beispiel

```
[COMMAND_1]
Command      = DirCreate
Path         = /Homepage
ErrorAsWarning = 1

[COMMAND_2]
Command      = FileCopy
Source       = /SD/Index.htm
Destination  = /Homepage/index.htm

[COMMAND_3]
Command      = FtpConnect
ServerAddr   = 192.168.123.45
UserName     = admin
Password     = admin
```

Name der Sektionen

Die Namen der Sektionen bilden sich aus der Zeichenkette `COMMAND_` gefolgt von einer Zahl. Die Zahl liegt zwischen eins und dem Wert des Eintrags `CommandCount` aus der Sektion `[OPTIONS]`.

Bearbeitung der Kommandos

Die AutoCopy-Funktion bearbeitet die Kommandos in der Reihenfolge der Sektionsnamen:

- Beginnend mit dem Kommando unter Sektion `[COMMAND_1]`.
- Endend mit dem Kommando unter der Sektion mit dem Wert des Eintrags `CommandCount` aus der Sektion `[OPTIONS]`.
- Jede Kommandosektion darf nur ein Kommando enthalten. Somit müssen Sie für jedes Kommando eine eigene Sektion anlegen.

Fehlerbehandlung

Wenn bei der Bearbeitung eines Kommandos ein Fehler auftritt, legt die Steuerung einen entsprechenden Eintrag in der Protokolldatei an. Für jedes Kommando können Sie einstellen, ob die Steuerung den Fehler als `Error` oder als `Warning` in die Protokolldatei einträgt. Nehmen Sie die Einstellung über den optionalen Parameter `ErrorAsWarning` vor.

ErrorAsWarning	Eintrag in die Protokolldatei
Parameter nicht vorhanden	Error
ErrorAsWarning = 0	Error
ErrorAsWarning = 1	Warning

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Dateinamen

- Der Funktionsparameter für die lokale Datei kann auch den Pfad zu dieser Datei enthalten, z. B. `'/Data/TestFiles/LocalTestFile.txt'`.
- Wenn das Dateisystem dies unterstützt, kann der Funktionsparameter für die Datei auf dem FTP-Server auch den Pfad zu dieser Datei enthalten. Sonst ist zuvor mit dem Befehl `FtpDirChange()` das Verzeichnis einzustellen.
- Das Dateisystem unterstützt beide Varianten.

Verfügbare Kommandos im lokalen Dateisystem

Die folgenden Kommandos für Zugriffe auf das lokale Dateisystem sind verfügbar:

Command = DirCreate

Funktion	Legt Unterverzeichnis an
Parametername	Path
Parameterwert	Vollständiger Verzeichnisname
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Verzeichnisnamen▪ Übergeordnete Verzeichnisse sind vorhanden
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Verzeichnisname▪ Nicht vorhandenes übergeordnetes Verzeichnis▪ Name eines vorhandenen Verzeichnisses
Bei ungültigem Wert	Die Steuerung legt das Verzeichnis nicht an. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_1] Command = DirCreate Path = /sub1 [COMMAND_2] Command = DirCreate Path = /sub1/sub2</pre>

Command = DirRemove

Funktion	Löscht ein Unterverzeichnis
Parametername	Path
Parameterwert	Vollständiger Verzeichnisname
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Verzeichnisnamen▪ Das Verzeichnis ist leer
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Verzeichnisname▪ Verzeichnis ist nicht leer
Bei ungültigem Wert	Die Steuerung löscht das Verzeichnis nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_8] Command = DirRemove Path = /sub1/sub2</pre>

Command = FileCopy

Funktion	Kopieren einer Datei
Parametername 1	Source

Parameterwert 1	Vollständiger Name der Quelldatei
Parametername 2	Destination
Parameterwert 2	Vollständiger Name der Zieldatei
Gültige Werte	<ul style="list-style-type: none"> ▪ Alle gültigen Dateinamen ▪ Das Zielverzeichnis ist vorhanden
Nicht gültige Werte	<ul style="list-style-type: none"> ▪ Ungültiger Dateiname ▪ Nicht vorhandene Quelldatei ▪ Nicht vorhandenes Zielverzeichnis
Bei ungültigem Wert	Die Steuerung kopiert die Datei nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_1] Command = FileCopy Source = /SD/OS/JC-340_1.04.0.03.os Destination = /System/OS/op_system.os [COMMAND_2] Command = FileCopy Source = /SD/Manual.pdf Destination = /sub1/Manual.pdf</pre>

Command = FileRemove

Funktion	Datei löschen
Parametername	Path
Parameterwert	Vollständiger Name der Datei
Gültige Werte	Alle gültigen Dateinamen
Nicht gültige Werte	Ungültiger Dateiname
Bei ungültigem Wert	Die Steuerung löscht die Datei nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_5] Command = FileRemove Path = /sub1/Manual.pdf</pre>

Command = DaFileRead

Funktion	Übertragen von Registerwerten und Merkerzuständen von einer Datendatei zur Steuerung
Parametername	DaFile
Parameterwert	Vollständiger Name der Datendatei
Gültige Werte	Alle gültigen Dateinamen für Datendateien
Nicht gültige Werte	<ul style="list-style-type: none"> ▪ Ungültiger Dateiname ▪ Nicht vorhandene Datendatei
Bei ungültigem Wert	Die Daten werden nicht an die Steuerung übertragen. Die Steuerung trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_12] Command = DaFileRead DaFile = /SD/Data/MyTestData.da</pre>

Command = DaFileWrite

Funktion	Abspeichern von Registerwerten und Merkerzuständen in einer Datendatei
Parametername 1	DaFile
Parameterwert 1	Vollständiger Name der Datendatei
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Dateinamen für Datendateien▪ Das Zielverzeichnis ist vorhanden
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Dateiname▪ Nicht vorhandenes Zielverzeichnis
Bei ungültigem Wert	Die Steuerung legt die Datendatei nicht an. Sie trägt den Fehler in die Protokolldatei ein.
Parametername 2	Append
Parameterwert 2	Legt fest, ob eine neue Datendatei erstellt oder an eine vorhandene angefügt wird
Gültige Werte	<ul style="list-style-type: none">▪ 0 = ggf. bestehende Datendatei löschen und neue Datendatei erstellen▪ 1 = an bestehende Datendatei anfügen; wenn keine Datendatei vorhanden ist, erstellt die Steuerung eine neue Datendatei
Nicht gültige Werte	<ul style="list-style-type: none">▪ < 0▪ > 1
Bei ungültigem Wert	Die Datendatei wird neu erstellt.
Parametername 3	Type
Parameterwert 3	Legt fest, ob Register oder Merker abgespeichert werden sollen
Gültige Werte	<ul style="list-style-type: none">▪ Register▪ Flag
Nicht gültige Werte	Werte ungleich <i>Register</i> oder <i>Flag</i>
Bei ungültigem Wert	Die Steuerung legt die Datendatei nicht an. Sie trägt den Fehler in die Protokolldatei ein.
Parametername 4	First
Parameterwert 4	Nummer des ersten Registers/Merkers
Gültige Werte	Alle gültigen Nummern aus dem Speicherbereich der jeweiligen Steuerung
Nicht gültige Werte	Ungültige Nummern
Bei ungültigem Wert	Die Steuerung legt die Datendatei nicht an. Sie trägt den Fehler in die Protokolldatei ein.
Parametername 5	Last

Parameterwert 5	Nummer des letzten Registers/Merkers
Gültige Werte	Alle gültigen Nummern aus dem Speicherbereich der jeweiligen Steuerung, die größer oder gleich sind als der Wert bei <i>First</i>
Nicht gültige Werte	<ul style="list-style-type: none"> ▪ Ungültige Nummern ▪ Nummern kleiner als <i>First</i>
Bei ungültigem Wert	Die Steuerung speichert nur einen Wert (<i>First</i>) ab.
Beispiel	<pre>[COMMAND_11] Command = DaFileWrite DaFile = /SD/MyTestData2.da Append = 0 Type = Register First = 1000000 Last = 1000000 [COMMAND_12] Command = DaFileWrite DaFile = /SD/MyTestData2.da Append = 1 Type = Flag First = 10 Last = 20 [COMMAND_13] Command = DaFileWrite DaFile = /SD/MyTestData2.da Append = 1 Type = Register First = 1000001 Last = 1000999</pre>

Verfügbare Kommandos für FTP-Zugriffe

Die folgenden Kommandos für Zugriffe über das Netzwerk mit FTP sind verfügbar:

Command = FtpConnect

Funktion	Verbindung zu einem FTP-Server herstellen
Parametername 1	ServerAddr
Parameterwert 1	IP-Adresse oder Name des FTP-Servers
Gültige Werte	<ul style="list-style-type: none"> ▪ Erreichbare IP-Adresse ▪ Name, der über DNS auflösbar ist
Nicht gültige Werte	<ul style="list-style-type: none"> ▪ Nicht erreichbare IP-Adresse ▪ Nicht auflösbarer Name
Parametername 2	UserName
Parameterwert 2	User-Name für die Anmeldung am FTP-Server
Parametername 3	Password
Parameterwert 3	Passwort für die Anmeldung am FTP-Server
Bei ungültigen Werten	Die Steuerung stellt die Verbindung nicht her. Sie trägt den Fehler in die Protokolldatei ein.

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Beispiel	<pre>[COMMAND_1] Command = FtpConnect ServerAddr = 192.168.123.45 UserName = admin Password = admin</pre>
Einschränkung	Nur eine Verbindung zu einem FTP-Server ist möglich. Wenn eine Verbindung zu einem anderen FTP-Server hergestellt wird, schließt die Steuerung zuvor die bestehende Verbindung.

Command = FtpFileRead

Funktion	Datei vom FTP-Server in das lokale Dateisystem kopieren
Parametername 1	ServerFile
Parameterwert 1	Vollständiger Name der Quelldatei im FTP-Server
Parametername 2	ClientFile
Parameterwert 2	Vollständiger Name der Zieldatei im lokalen Dateisystem
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Dateinamen▪ Das Zielverzeichnis ist vorhanden
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Dateiname▪ Nicht vorhandene Quelldatei▪ Nicht vorhandenes Zielverzeichnis
Bei ungültigem Wert	Die Steuerung kopiert die Datei nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_8] Command = FtpFileRead ServerFile = /app/cantest/cantest.es3 ClientFile = /SD/cantest3.es</pre>

Command = FtpFileWrite

Funktion	Datei vom lokalen Dateisystem in das des FTP-Servers kopieren
Parametername 1	ServerFile
Parameterwert 1	Vollständiger Name der Zieldatei im FTP-Server
Parametername 2	ClientFile
Parameterwert 2	Vollständiger Name der Quelldatei im lokalen Dateisystem
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Dateinamen▪ Das Zielverzeichnis ist vorhanden
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Dateiname▪ Nicht vorhandene Quelldatei▪ Nicht vorhandenes Zielverzeichnis
Bei ungültigem Wert	Die Steuerung kopiert die Datei nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	<pre>[COMMAND_5] Command = FtpFileWrite ServerFile = /System/OS/op_system.os ClientFile = /SD/OS/JC-340_1.09.0.00.os</pre>

Command = FtpFileRemove

Funktion	Datei im FTP-Server löschen
Parametername	ServerFile
Parameterwert	Vollständiger Dateiname
Gültige Werte	Alle gültigen Dateinamen
Nicht gültige Werte	Ungültiger Dateiname
Bei ungültigem Wert	Die Steuerung löscht die Datei nicht. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	[COMMAND_9] Command = FtpFileRemove ServerFile = /sub1/Manual.pdf

Command = FtpDirChange

Funktion	Arbeitsverzeichnis im FTP-Server wechseln
Parametername	ServerDir
Parameterwert	Vollständiger Verzeichnisname
Gültige Werte	Alle gültigen Verzeichnisnamen
Nicht gültige Werte	Ungültiger Verzeichnisname
Bei ungültigem Wert	Die Steuerung wechselt nicht das Verzeichnis. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	[COMMAND_12] Command = FtpDirChange ServerDir = /Data/MyTestData

Command = FtpDirCreate

Funktion	Unterverzeichnis anlegen im FTP-Server
Parametername	ServerDir
Parameterwert	Vollständiger Verzeichnisname
Gültige Werte	<ul style="list-style-type: none"> ▪ Alle gültigen Verzeichnisnamen ▪ Übergeordnete Verzeichnisse sind vorhanden
Nicht gültige Werte	<ul style="list-style-type: none"> ▪ Ungültiger Verzeichnisname ▪ Nicht vorhandenes übergeordnetes Verzeichnis ▪ Name eines vorhandenen Verzeichnisses
Bei ungültigem Wert	Die Steuerung legt das Verzeichnis nicht an. Sie trägt den Fehler in die Protokolldatei ein.
Beispiel	[COMMAND_6] Command = FtpDirCreate ServerDir = /Data/MyTestData
Einschränkung	Wenn als Funktionsparameter ein Verzeichnis mit einem Pfad angegeben wird, müssen alle Verzeichnisse bis auf das zu Erstellende vorhanden sein. Ein rekursives Anlegen von mehreren Verzeichnissen wird nicht unterstützt.

Command = FtpDirRemove

Funktion	Unterverzeichnis löschen im FTP-Server
Parametername	ServerDir

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Parameterwert	Vollständiger Verzeichnisname
Gültige Werte	<ul style="list-style-type: none">▪ Alle gültigen Verzeichnisnamen▪ Das Verzeichnis ist leer
Nicht gültige Werte	<ul style="list-style-type: none">▪ Ungültiger Verzeichnisname▪ Verzeichnis ist nicht leer
Bei ungültigem Wert	Die Steuerung löscht das Verzeichnis nicht. Sie trägt den Fehler in die Protokolldatei ein.

Beispiel	[COMMAND_8] Command = FtpDirRemove ServerDir = /Data/MyTestData
----------	---

Beispiel für eine Kommandodatei

Aufgabe

Die Steuerung JetControl 340 soll hier als Beispiel dienen. Sie steuert mit verschiedenen JX3-Modulen eine bestehende Anlage. In dieser Anlage wollen Sie die Funktionen erweitern.

Hierfür sind folgende Änderungen erforderlich:

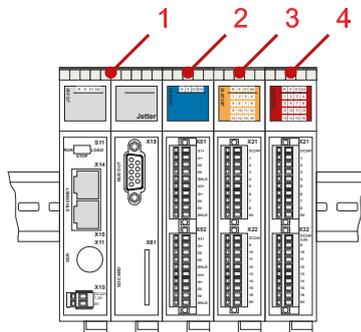
- Betriebssystemupdate für die Steuerung
- Betriebssystemupdate für ein Analogausgangsmodul
- Neues Anwendungsprogramm
- Neue Werte für einige Register

Lösung

Sie kopieren die benötigten Dateien auf eine SD-Karte und erstellen eine Kommandodatei für die AutoCopy-Funktion. Anschließend schicken Sie diese SD-Karte mit einer kurzen Bedienungsanleitung zum Anlagenbediener, der sie nach dem Update wieder zurückschickt.

Beispielkonfiguration

Dieses Beispiel basiert auf folgender Konfiguration:



Nummer	Teil	Beschreibung
1	JC-340	Steuerung
2	JX3-AO4	Analogausgangsmodul Modulnummer: 2
3	JX3-DI16	Digitales Eingangsmodul
4	JX3-DIO16	Digitales Ausgangsmodul

5 AutoCopy - Automatisches Kopieren von Steuerungsdaten

Inhalt der SD-Karte

Die folgende Abbildung zeigt die Verzeichnisstruktur und die Dateien auf der SD-Karte aus der Sicht der Steuerung vor der Ausführung der AutoCopy-Funktion:



Nach der Ausführung ist zusätzlich die Protokolldatei **autocopy.log** vorhanden.

Kommandodatei

```
[OPTIONS]
CommandCount = 7
LogFile      = /SD/autocopy.log
LogAppend   = 0

# update operating system of controller
[COMMAND_1]
Command     = FileCopy
Source      = /SD/OS/JC-340_1.04.0.00.os
Destination = /System/OS/op_system.os

# update operating system of JX3-A04 module
[COMMAND_2]
Command     = FileCopy
Source      = /SD/OS/JX3-A04_1.01.0.00.os
Destination = /System/JX3-Module02/OS/system.os

# create user program directories
# probably already present - but to be sure ...
[COMMAND_3]
Command     = DirCreate
Path        = /app
ErrorAsWarning = 1

[COMMAND_4]
Command     = DirCreate
Path        = /app/userprogtest
```

```
# copy user program start file
[COMMAND_5]
Command      = FileCopy
Source       = /SD/UserProgs/start.ini
Destination  = /app/start.ini

# copy user program
[COMMAND_6]
Command      = FileCopy
Source       = /SD/UserProgs/userprogtest.es3
Destination  = /app/userprogtest/userprogtest.es3

# set registers and flags
[COMMAND_7]
Command      = DaFileRead
DaFile       = /SD/UserData/MyTestData.da
```

5.3 Protokolldatei

Einleitung

Dieses Kapitel beschreibt Aufbau und Inhalt der Protokolldatei, in der das Gerät die Ergebnisse der jeweiligen Kommandos einträgt.

Inhalt

Thema	Seite
Dateiinhalt	97

Dateiinhalt

Einleitung

Die Protokolldatei ist eine Textdatei. Durch einen Eintrag in die Kommando-datei bestimmen Sie, ob eine Protokolldatei neu erstellt oder das Gerät die Einträge an eine bestehende Protokolldatei anfügt.

Beispiel

```
JetControl AutoCopy log file 07.11.2008 09:14:09

1: Ok      - FileCopy  /SD/OS/JC-340_1.04.0.00.os
              /System/OS/op_system.os (345740 byte)
2: Ok      - FileCopy  /SD/OS/JX3-A04_1.01.0.00.os
              /System/JX3-Module02/OS/system.os
              (16832 byte)
3: Warning - DirCreate  /app
4: Ok      - DirCreate  /app/userprogtest
5: Ok      - FileCopy  /SD/UserProgs/start.ini
              /app/start.ini (63 byte)
6: Ok      - FileCopy  /SD/UserProgs/userprogtest.es3
              /app/userprogtest/userprogtest.es3
              (169 byte)
7: Error   - DaFileRead /SD/UserData/MyTestData.da

Command statistics:
  Total   : 7
  Ok      : 5
  Warning : 1
  Error   : 1
```

Beschreibung

Die Protokolldatei besteht - beim Anfügen an eine bestehende Protokolldatei für jede Ausführung der AutoCopy-Funktion - aus drei Elementen:

- Die Kopfzeile enthält Datum und Uhrzeit.
- Der anschließende Block gibt Auskunft über die ausgeführten Kommandos.
- Abschließend erscheint eine kurze Statistik über die Kommandobearbeitung.

Im obigen Beispiel wird beim Versuch, das Verzeichnis */app* anzulegen, ein Fehler erzeugt, da das Verzeichnis bereits vorhanden ist. Diesen Fehler trägt das Gerät als Warnung ein. Wenn das Gerät die DA-Datei liest, tritt ebenfalls ein Fehler auf. Diesen Fehler trägt das Gerät in die Protokolldatei ein.

5.4 Datendateien

Einleitung

Dieses Kapitel beschreibt die Datendatei, in die das Gerät die Werte von Registern und Merkern abspeichert.

Inhalt

Thema	Seite
Dateiformat	99

Dateiformat

Format

Die Datendatei ist wie folgt aufgebaut:

- Reine Textdatei
- Jeder Eintrag muss in einer eigenen Zeile stehen
- Jede Zeile muss mit Wagenrücklauf / Zeilenvorschub (carriage return/line feed) abgeschlossen sein
- Kommentarzeilen werden mit einem Semikolon (;) eingeleitet
- Als erster Eintrag in der Datei muss als Kennung *SD1001* stehen

Datenzeilen

Eine Datenzeile besteht aus den folgenden Teilen:

- Variablenkennung am Beginn der Zeile
- Durch Leerzeichen oder Tabulator getrennt folgt die Variablennummer
- Durch Leerzeichen oder Tabulator getrennt folgt der Variablenwert

Variablenkennung	Variablentyp
FS	Merker
RS	Ganzzahlregister
QS	Fließkommazahlregister

Beispiel

```
SD1001
; Data File - Jetter AG
;
; Register 1000000 ... 1000005
RS 1000000 12345
RS 1000001 2
RS 1000002 -1062729008
RS 1000003 502
RS 1000004 50
RS 1000005 3
QS 1009000 3.14
;
; Flag10 ... 13
FS 10 0
FS 11 1
FS 12 1
FS 13 0
```

6 Anwendungsprogramm

Einleitung Dieses Kapitel beschreibt, wie das Anwendungsprogramm im Gerät abgelegt wird. Der Anwender legt fest, welches Programm ausgeführt werden soll.

Voraussetzungen an den Programmierer Dieses Kapitel setzt Kenntnisse über das Erzeugen und Übertragen von Anwendungsprogrammen in JetSym und über das Dateisystem des Gerätes voraus.

Inhalt

Thema	Seite
Standardablage des Anwendungsprogramms.....	102
Anwendungsprogramm auf der SD-Karte oder dem USB-Stick ablegen...	103
Anwendungsprogramm laden.....	105

Standardablage des Anwendungsprogramms

Einleitung

Beim Download des Anwendungsprogramms von JetSym auf die Steuerung wird dieses Programm als Datei auf der internen Flash-Disk abgespeichert. Das Gerät trägt den Pfad- und Dateinamen in der Datei **start.ini** im Ordner **app** ein.

Pfad- und Dateiname

Standardmäßig legt JetSym im Ordner **app** ein Unterverzeichnis mit dem Projektnamen an und speichert dort das Anwendungsprogramm mit der Dateierweiterung ***.esX** ab. Die Pfad- und Dateinamen werden dabei immer in Kleinbuchstaben konvertiert. **X** ist ein hardwareabhängiger Platzhalter, z. B. **es4** für JetControl 400 und **es9** für JetControl 900.

start.ini - Aufbau

Die Datei ist eine reine Textdatei mit einer Sektion mit zwei Einträgen:

Element	Beschreibung
[Startup]	Sektionsname
Project	Pfad zur Anwendungsprogrammdatei relativ zum Ordner app
Program	Name der Anwendungsprogrammdatei

Beispiel:

```
[Startup]
Project = test_program
Program = test_program.es9
```

Das Anwendungsprogramm wird aus der Datei **test_programm.es9** im Unterverzeichnis **test_program** im Ordner **app** geladen.

Verwandte Themen

- **Ablage des Anwendungsprogramms** (siehe Seite 103)
-

Anwendungsprogramm auf der SD-Karte oder dem USB-Stick ablegen

Einleitung

Beim Download des Anwendungsprogramms von JetSym auf die Steuerung wird die Standardablage des Anwendungsprogramms verwendet.

Wollen Sie, dass das Gerät das Anwendungsprogramm von der SD-Karte oder dem USB-Stick liest, müssen Sie den Ablageort konfigurieren.

Wenn Sie das Anwendungsprogramm in einem anderen Verzeichnis der internen Flash-Disk ablegen möchten, gehen Sie analog hierzu vor.

Voraussetzungen

Ausschließlich Kleinschreibung für Verzeichnis- und Dateinamen verwenden.

Anwendungsprogramms auf der SD-Karte oder dem USB-Stick ablegen

Wenn Sie das Anwendungsprogramm auf der SD-Karte oder dem USB-Stick ablegen wollen, dann konfigurieren Sie das Gerät wie folgt:

Schritt	Vorgehen
1	Erstellen Sie eine Anwendungsprogrammdatei mit JetSym.
2	Legen Sie das gewünschte Verzeichnis auf der SD-Karte oder dem USB-Stick an.
3	Legen Sie die Anwendungsprogrammdatei im gewünschten Verzeichnis ab.
4	Tragen Sie den Pfad zur Anwendungsprogrammdatei und den Programmnamen in die Datei start.ini im Ordner <i>app</i> auf der internen Flash-Disk des Geräts ein.

Ergebnis: Das Gerät lädt das Anwendungsprogramm von der SD-Karte oder dem USB-Stick beim Neustart.

start.ini - Aufbau

Die Datei ist eine reine Textdatei mit einer Sektion mit zwei Einträgen:

Element	Beschreibung
[Startup]	Sektionsname
Project	Pfad zur Anwendungsprogrammdatei
Program	Name der Anwendungsprogrammdatei

Steuerung:

Beispiel für SD-Karte:

```
[Startup]
Project = /sd/testprogram
Program = test1.esx
```

Beispiel für USB-Stick:

```
[Startup]
Project = /usb/testprogram
Program = test1.esx
```

Bediengerät:

Beispiel für SD-Karte:

```
[Startup]
Project = \sd\testprogram
Program = test1.esx
```

Beispiel für USB-Stick:

```
[Startup]
Project = \usb\testprogram
Program = test1.esx
```

Bediengerät JetView:

Beispiel für SD-Karte:

```
[Startup]
Project = \..\..\..\Storage Card
Program = test1.esx
```

Beispiel für USB-Stick:

```
[Startup]
Project = \..\..\..\USBMemory
Program = test1.esx
```

Ergebnis:

Das Anwendungsprogramm wird aus der Datei **test1.esx** im Unterverzeichnis **testprogram** im Ordner *sd* auf der SD-Karte oder dem Ordner *usb* auf dem USB-Stick geladen.

Verwandte Themen

- **Standardablage des Anwendungsprogramms** (siehe Seite 102)
-

Anwendungsprogramm laden

Einleitung

Beim Neustart des Anwendungsprogramms über JetSym oder nach dem Booten des Geräts wird das Anwendungsprogramm über das Dateisystem geladen und ausgeführt.

Der Ladevorgang bei der Steuerung

Dazu muss der Schalter S11 in Stellung *RUN* stehen. So wird das Anwendungsprogramm vom Betriebssystem der Steuerung geladen.

Stufe	Beschreibung
1	Das Betriebssystem liest die Datei <code>/app/start.ini</code> von der internen Flash-Disk.
2	Das Betriebssystem wertet den Eintrag Project aus. Er enthält den Pfad zur Anwendungsprogrammdatei.
3	Das Betriebssystem wertet den Eintrag Program aus. Er enthält den Programmnamen.
4	Das Betriebssystem lädt das Anwendungsprogramm aus der Datei <code><Project>/<Program></code> .

Der Ladevorgang beim Bediengerät

So wird das Anwendungsprogramm vom Betriebssystem des Bediengeräts geladen.

Stufe	Beschreibung
1	Das Betriebssystem liest die Datei <code>\app\start.ini</code> von der internen Flash-Disk.
2	Das Betriebssystem wertet den Eintrag Project aus. Er enthält den Pfad zur Anwendungsprogrammdatei.
3	Das Betriebssystem wertet den Eintrag Program aus. Er enthält den Programmnamen.
4	Das Betriebssystem lädt das Anwendungsprogramm aus der Datei <code><Project>\<Program></code> .

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.