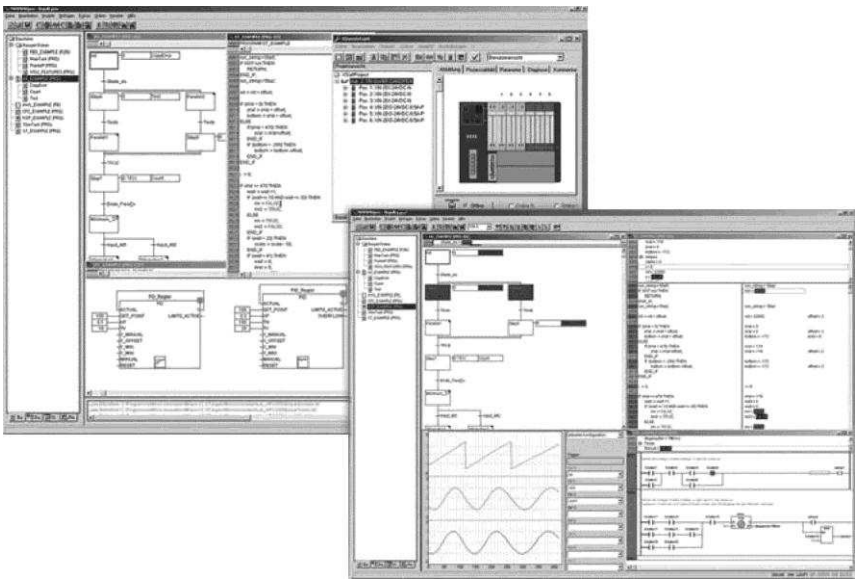
05/2012 MN04802094Z-DE
M003945-03

SPS-Programmierung XVS400



Powering Business Worldwide

Hersteller

Eaton Automation AG
Spinnereistrasse 8-14
CH-9008 St. Gallen
Schweiz
www.eaton-automation.com
www.eaton.com

Support

Region North America
Eaton Corporation
Electrical Sector
1111 Superior Ave.
Cleveland, OH 44114
United States
877-ETN-CARE (877-386-2273)
www.eaton.com

Andere Regionen
Bitte kontaktieren Sie Ihren lokalen Lieferanten oder senden Sie eine E-Mail an:
automation@eaton.com

Originalsprache

Deutsch

Redaktion

Daniel Lenherr

Marken- und Produktnamen

Alle in diesem Dokument erwähnten Marken- und Produktnamen sind Warenzeichen oder eingetragene Warenzeichen der jeweiligen Titelinhaber.

Copyright

© Eaton Automation AG, CH-9008 St. Gallen

Alle Rechte, auch die der Übersetzung, vorbehalten.

Kein Teil dieses Dokuments darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung der Firma Eaton Automation AG, St. Gallen reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Änderungen vorbehalten

Inhaltsverzeichnis

1 Allgemein 5
1.1 Zweck dieses Dokuments 5
1.2 Rückmeldungen zu diesem Dokuments 5
1.3 Ergänzende Dokumentation 5

2 Installation 6
2.1 Lieferumfang 6
2.2 Systemvoraussetzung 6
2.3 Installieren der SPS-Programmierumgebung 7
2.4 Installieren der SPS-Zielsysteme 8

3 Deinstallation 9
3.1 Deinstallieren der SPS-Programmierumgebung 9
3.2 Deinstallieren der SPS-Zielsysteme 9

4 Zielsystemeinstellungen 10
4.1 Zielplattform 10
4.2 Speicheraufteilung 11
4.2.1 Übersicht Speicherlayout 12
4.3 Allgemein 13
4.4 Netzfunktionen 14
4.5 Visualisierung 15

5 Steuerungskonfiguration 16
5.1 Arbeiten in der Steuerungskonfiguration 16
5.1.1 Projektspezifische Konfigurationsdatei(en) und Gerätedatei(en) 18
5.2 Allgemeine Einstellungen 19
5.3 Konfiguration als Profibus-Master 20
5.3.1 Konfiguration des Profibus-Masters 21
5.3.2 Konfiguration der Profibus-Slaves 26
5.3.3 Busdiagnose 32

6 Betrieb 33
6.1 Einschaltverhalten 33
6.2 Ausschaltverhalten 33
6.3 Betriebszustände der Steuerung 34
6.4 Betriebsartenumschaltung 35
6.5 Start-Verhalten 36
6.6 Stop-Verhalten 37
6.7 Reset-Verhalten 37
6.8 Test und Inbetriebnahme 38
6.9 Programmtransfer 39
6.9.1 Bootprojekt erzeugen 40

7 Programmabarbeitung und Systemzeiten 41
7.1 Programmabarbeitung 41
7.2 Taskkonfiguration 42

7.3 Multitasking 44
7.4 Tasküberwachung / Zykluszeitüberwachung 44
7.5 Datenremanenz 47
7.6 Direkter Peripheriezugriff 47
7.7 Interruptverarbeitung 47
7.8 Systembibliotheken, Funktionsbausteine und Funktionen 48
7.9 Prozessabbildverarbeitung / IO-Update 48
7.9.1 Profibus 49

8 Verbindungsaufbau Programmier-PC – Steuerung 50
8.1 Verbindungsaufbau über Ethernet 50

9 Parameter-Manager / Objektverzeichnis 53

10 SPS-Browser 54

11 Alarmkonfiguration 55

12 Anbindung an die Visualisierung / Erstellen der Symboldatei 56
12.1 Symboldatei konfigurieren 56
12.2 Symboldatei senden 59

13 Zielsysteminstallation und Firmware Update 60

14 Lizenzierung 62
14.1 SPS-Programmierumgebung 62
14.2 SPS-Laufzeitsystem 62
14.3 Target-Visualisierung 62
14.4 Web-Visualisierung 62

15 Änderungsindex 63

1 Allgemein

1.1 Zweck dieses Dokuments

Dieses Dokument beschreibt den Umgang mit der SPS-Programmierumgebung "XSoft-CoDeSys-2" und dem SPS-Laufzeitsystem für den XVS400-Gerätetyp mit Windows CE. Dieses Dokument dient als Ergänzung zum Handbuch der SPS-Programmierung CoDeSys V2.3 der Fa. 3S-Smart Software Solutions GmbH.

Dialoge und Beispiele in diesem Dokument sind standardisiert. Je nach Auswahl des Steuerungstyps können sich deshalb vereinzelte Dialoge unterscheiden.

1.2 Rückmeldungen zu diesem Dokuments

Bitte senden Sie Kommentare, Empfehlungen oder Anregungen zu diesem Dokument an automation@eaton.com.

1.3 Ergänzende Dokumentation

Ergänzend zu diesem Dokument können weitere Dokumente hilfreich sein.

Folgende Dokumentation finden Sie auf unserer Homepage (www.eaton-automation.com):

[1] MN05010007Z-DE
Systembeschreibung Windows CE

[2] Diverse
Dokumentationen bezüglich SPS-Funktionsbibliotheken

[3] -
Systembeschreibung CiA Draft Standard DSP301

[4] -
Handbuch der SPS-Programmierung CoDeSys V2.3

2 Installation

Das Produkt "XSoft-CoDeSys-2" ist ein eigenständiges Software Paket. Es besteht aus einer PC-seitigen Komponente (SPS-Programmierumgebung mit entsprechenden SPS-Zielsystemen), welche auf einem beliebigen PC installiert werden kann und einer zielsystem-seitigen Komponente (SPS-Laufzeitsystem), welche unter der Berücksichtigung von Lizenzpunkten auf SPS-Zielsystemen mit Windows CE installiert werden kann (➔ Kap. 14).

Bei Fragen zu Lizenzprodukten wenden Sie sich bitte an Ihren lokalen Eaton Vertriebspartner

2.1 Lieferumfang

Bezeichnung	Version
Software "XSoft-CoDeSys-2" inkl. Benutzerhandbuch	2.3.9 SP3

Bestehend aus:

Bezeichnung	Version
CoDeSys	2.3.9.35
CoDeSys Gateway	2.3.9.25
CoDeSys OPC-Server [1]	2.3.13.8

SPS-Zielsystem	SPS-Laufzeitsystem	Betriebssystem (OS)
XVS-4xx-V2.3.9 SP3	PLCWinCE V 2.4.10 (xxx)	Windows CE 3.0 Image Release 2.20.5 (xxx) Windows CE 5.0 Image Release 2.26.0 (xxx)

2.2 Systemvoraussetzung

Betriebssystem
Windows XP, Windows Vista, Windows 7

[1] Optionale Installation, Komponente ist kein Bestandteil dieser Dokumentation

2.3 Installieren der SPS-Programmierumgebung

Legen Sie die Installations-CD in Ihren PC ein und starten Sie die Installation über 'Setup.exe'.

→ Ist bei der Installation der SPS-Programmierumgebung keine Serienummer bzw. Lizenz vorhanden, werden die SPS-Zielsysteme im Demomode installiert.

Standardmässig werden mit der Installation folgende Verzeichnisse angelegt:

Verzeichnis	Beschreibung
Programmiersystem-Verzeichnis C:\Programme\Eaton\XSoft-CoDeSys V2.3.9 SP3	SPS-Programmierumgebung
Zielsystem-Verzeichnis C:\Programme\Gemeinsame Dateien\CAA-Targets\Eaton Automation\V2.3.9 SP3	Zielsystemabhängige Komponenten wie Bibliotheken und Konfigurationsdaten inkl. Firmware für die SPS-Zielsysteme

Verhalten gegenüber bisher installierter SPS-Programmierumgebungen

Bereits installierte Versionen der SPS-Programmierumgebung "XSoft-CoDeSys-2" und entsprechende SPS-Zielsysteme werden nicht tangiert.

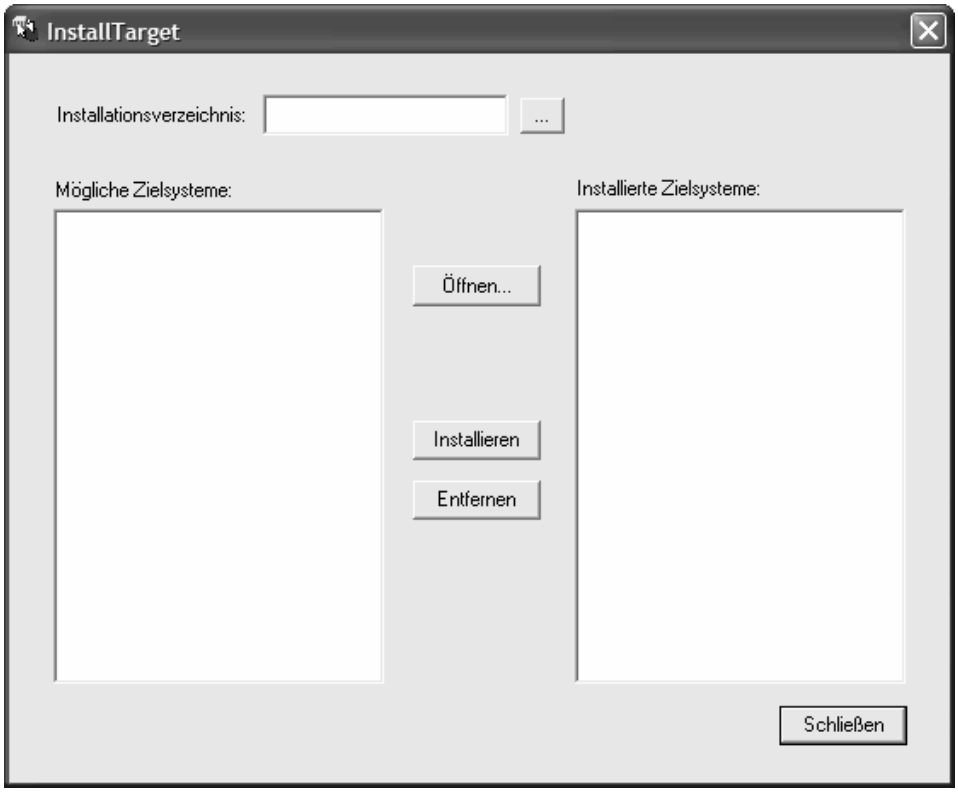
→ Sämtliche zur Installation dazugehörende SPS-Zielsysteme erhalten eine neue Bezeichnung bzw. Identifikation.

→ Möchte ein mit älterer Version der SPS-Programmierumgebung erstelltes SPS-Projekt aktualisiert werden, muss das SPS-Zielsystem umgestellt werden. Nach Schliessen und dem erneuten Öffnen des SPS-Projektes wird das SPS-Projekt aktualisiert und es werden die aktuellen Bibliotheken eingebunden.

2.4 Installieren der SPS-Zielsysteme

Standardmässig werden die zur SPS-Programmierumgebung entsprechenden SPS-Zielsysteme über den Setup installiert.

Über die Anwendung 'InstallTarget' können allerdings auch nachträglich SPS-Zielsysteme installiert bzw. deinstalliert werden.



3 Deinstallation

3.1 Deinstallieren der SPS-Programmierumgebung

Bei der Deinstallation werden lediglich installierte Dateien und Komponenten entfernt. Wenn nötig müssen nach der Deinstallation Dateien und Verzeichnisse aus dem Programmiersystem-Verzeichnis manuell entfernt werden.

3.2 Deinstallieren der SPS-Zielsysteme

Bei der Deinstallation der SPS-Programmierumgebung werden die zielsystemabhängigen Komponenten wie Bibliotheken und Konfigurationsdaten im Zielsystem-Verzeichnis nicht entfernt.

Die installierten SPS-Zielsysteme müssen über die entsprechende Deinstallationsroutine entfernt werden.

Alternativ können installierte SPS-Zielsysteme auch über die Anwendung 'InstallTarget' aus der SPS-Programmierumgebung entfernt werden.

Anschliessend müssen die zielsystemabhängigen Komponenten aus dem Zielsystem-Verzeichnis manuell entfernt werden.

4 Zielsystemeinstellungen

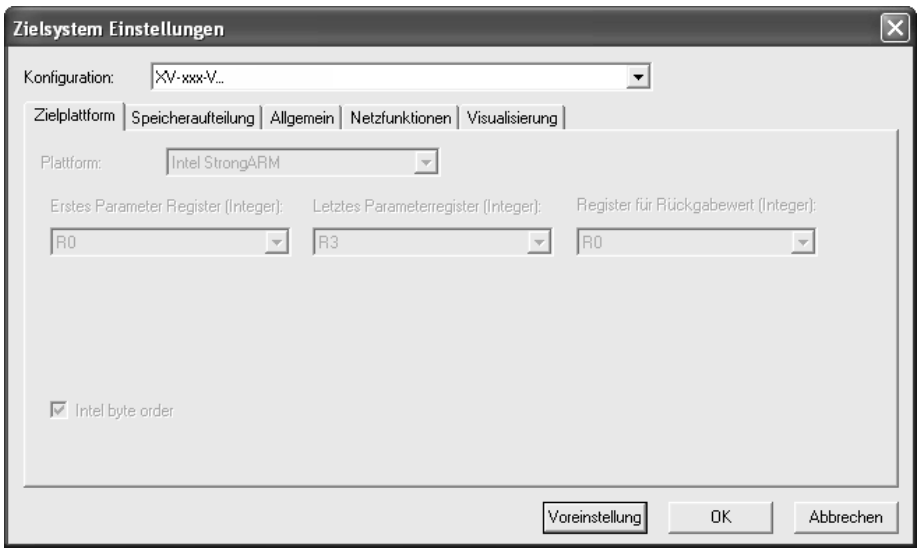
Dieser Dialog öffnet automatisch, wenn ein neues Projekt angelegt wird. Ansonsten wird er über die Funktionalität <Zielsystemeinstellungen> im Registerblatt 'Ressourcen' erreicht.

In den Zielsystemeinstellungen wählen Sie den entsprechenden Steuerungstyp. Anhand dieser Konfiguration werden Prozessortyp und Speichergrössen optimal eingestellt. Der Eintrag '**None**' führt automatisch in den Simulationsmodus.

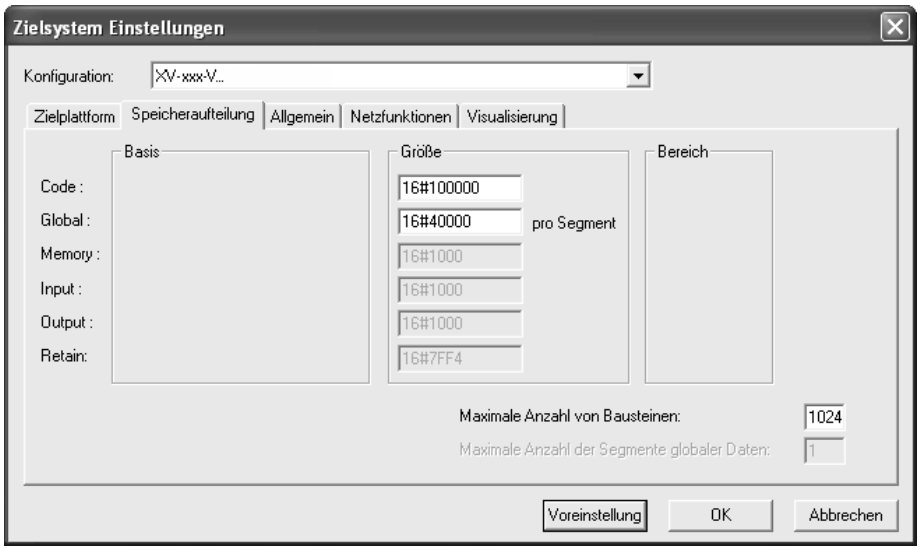
→ Veränderungen der voreingestellten Zielsystemeinstellungen können Auswirkungen auf das Verhalten des SPS-Zielsystems haben!

Über die Schaltfläche **Voreinstellung** kann die allenfalls veränderte Konfiguration wieder auf die Standardkonfiguration zurückgesetzt werden.

4.1 Zielplattform



4.2 Speicheraufteilung



Code Grösse

Standardmässig 1024kB (16#100000) Speicher für den Programmcode. Diese entsprechende Speichermenge wird automatisch alloziert.

Global Grösse

Standardmässig 256kB (16#40000) Speicher für die globalen Daten. Diese entsprechende Speichermenge wird automatisch alloziert.

Max. Anzahl von Programmbausteinen

Pro Programmbaustein werden automatisch 12Byte Speicher benötigt. Insgesamt werden somit 12kB Speicher für die Funktionspointertabelle benötigt (1024 Bausteine -> 12kB). Dieser Speicher wird an den SPS-Datenspeicher angefügt.

→ In die Berechnung der im Programm verwendeten Anzahl der Bausteine werden sämtliche Funktionen und Funktionsbausteine eingefügter Bibliotheken miteinbezogen.

4.2.1 Übersicht Speicherlayout

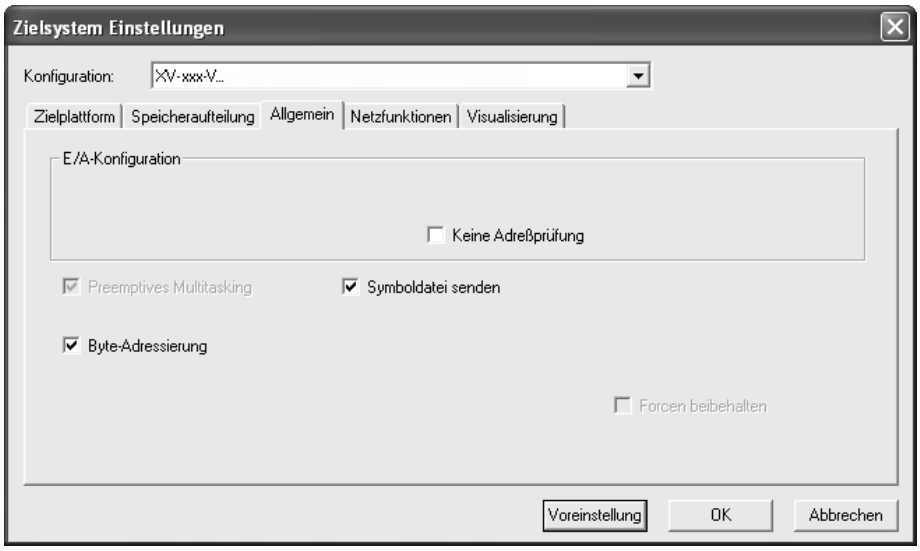
Default Einstellung mit Online-Change
Anzahl Datensegmente = 1
Grösse des Datensegments = 256kB

32kB Remanenter Datenbereich
12kB Funktionspointertabelle
12kB Ein-/Ausgangs-/Merkerdaten
1 * 256kB Globaler Datenbereich
1024kB Programmcode (Online Change)
1024kB Programmcode

→ Beim standardmässigen Übersetzen des SPS-Programms teilt die SPS-Programmierumgebung dem Anwender die Datenbereichs-Grössen mit.

Mit dem SPS-Browser kann Online mit dem Kommando 'sysinfo' die SPS-Programmgrösse ermittelt werden (➔ Kap. 10)

4.3 Allgemein



Keine Adressprüfung

Bei aktivierter Option wird beim Kompilieren keine Adressprüfung vorgenommen.

Symboldatei senden

Bei aktivierter Option wird beim Kompilieren eine Symboldatei erzeugt und beim Programmdownload auf die Steuerung geladen.

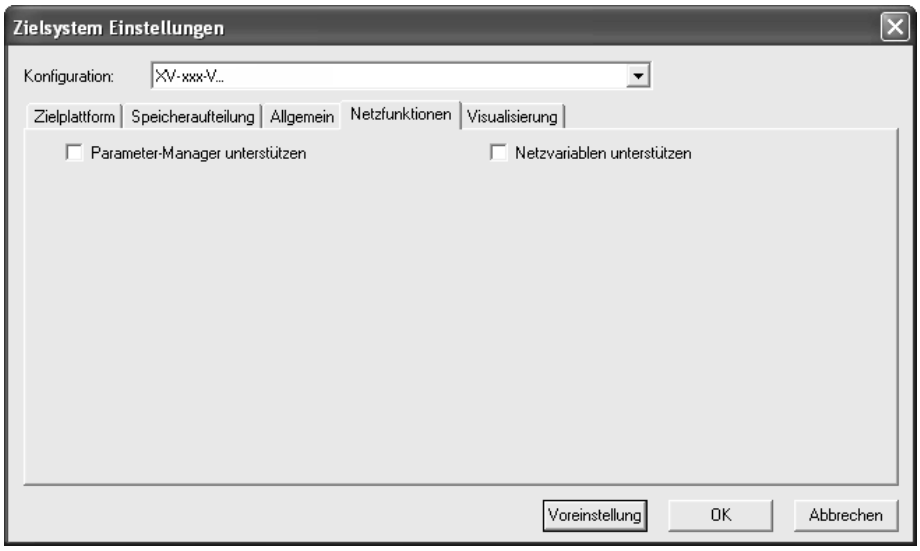
Preemptives Multitasking

Multitasking wird unterstützt (kann vom Anwender nicht verändert werden).

Byte-Adressierung

Bei aktivierter Option erfolgt die Adressierung byteweise (z.B.: %QD4 entspricht %QB4).

4.4 Netzfunktionen



Parameter-Manager unterstützen

Bei aktivierter Option erscheint im Registerblatt 'Ressourcen' der Eintrag **Parameter-Manager**. Dieser ermöglicht das Erstellen eines Objektverzeichnisses für Variablen und Parameter, die dem gezielten, aktiven Datenaustausch mit anderen Steuerungen dienen.

→ Die Objektverzeichnis-Funktionalität wird in Verbindung mit einem in der Steuerungskonfiguration eingefügten CAN-Device unterstützt.

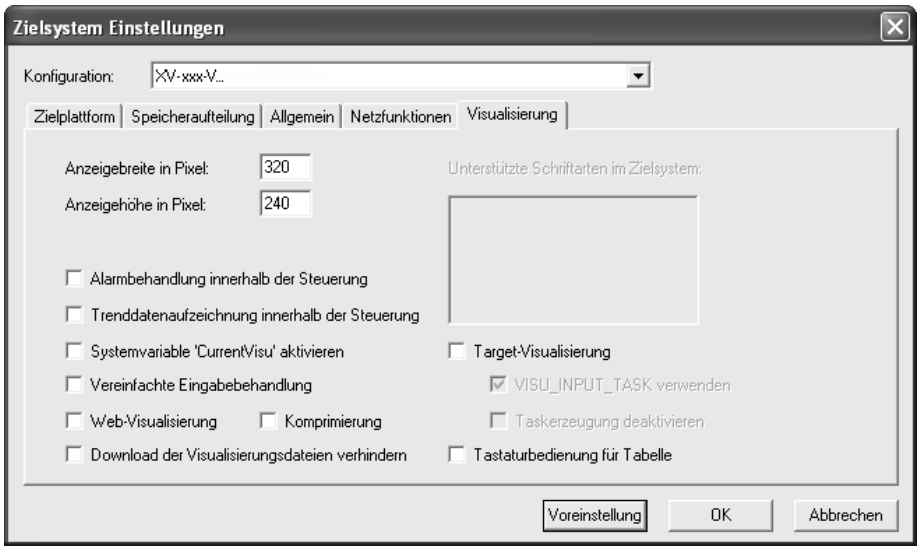
→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3 oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

Netzvariablen unterstützen

Bei aktivierter Option können Netzwerkvariablen verwendet werden. Sie dienen dem automatischen Datenaustausch mit anderen Steuerungen.

→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3 oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

4.5 Visualisierung



In Abhängigkeit vom verwendeten SPS-Zielsystem werden Target-Visualisierung bzw. Web-Visualisierung unterstützt. Entsprechende Information ist nachfolgender Tabelle zu entnehmen.

SPS-Zielsystem	SPS-Laufzeitsystem	Target-Visu	Web-Visu
≥ XVS-4xx-V2.3.9 SP1	≥ PLCWinCE V 2.4.7 (xxx)	✓	✓

→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3, dem Handbuch zur CoDeSys Visualisierung oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

5 Steuerungskonfiguration

Die Steuerungskonfiguration wird über die Funktionalität <Steuerungskonfiguration> im Registerblatt 'Ressourcen' erreicht.

Die Steuerungskonfiguration erlaubt das Konfigurieren von Ein- und Ausgängen bzw. das Anbinden von busfähigen I/O-Modulen. Grundlage für das Arbeiten in der Steuerungskonfiguration sind die Konfigurationsdateien (*.cfg) und Gerätedateien (z.B. *.gsd, *.eds).

Die Konfiguration wird im Editor in Baumstruktur dargestellt und kann über Menüpunkte und Dialoge bearbeitet werden. Es gibt Elemente, die entweder als Ein- oder Ausgänge dienen oder Verwaltungselemente, die ihrerseits wieder Unterelemente haben (z.B. CAN, PROFIBUS).

Ein- und Ausgänge erscheinen im Editor mit der IEC-Adresse, über die auf sie zugegriffen werden kann. Zur Kennzeichnung kann jedem Ein- und Ausgang ein symbolischer Name vergeben werden, der dann vor der IEC-Adresse steht.

5.1 Arbeiten in der Steuerungskonfiguration

Die Steuerungskonfiguration besteht aus zwei Fensterhälften: Links wird der 'Konfigurationsbaum' dargestellt, rechts erscheinen die jeweils zum selektierten Element passenden Konfigurationsdialoge auf einem oder mehreren Registerblättern.

Am Kopf des Konfigurationsbaumes steht das so genannte Root-Modul. Darunter folgen hierarchisch eingerückt die weiteren Elemente der Konfiguration Module verschiedener Typen (CAN, PROFIBUS, I/O), Kanäle oder Bitkanäle.

Selektieren von Elementen

Zum Selektieren von Elementen führen Sie einen Mausklick auf das entsprechende Element aus bzw. bewegen Sie das gepunktete Rechteck mit den Pfeiltasten auf das gewünschte Element.

Elemente, die von einem Pluszeichen angeführt werden, sind Organisationselemente und enthalten Unterelemente. Zum Aufklappen selektieren Sie das Element und führen Sie einen Doppelklick auf dieses Pluszeichen aus oder drücken Sie die <Eingabetaste>. Auf die gleiche Weise werden aufgeklappte Elemente (Minuszeichen vor dem Element) zugeklappt.

Einfügen von Elementen, <Einfügen> <Element einfügen>, <Einfügen> <Unterelement anhängen>

Abhängig von den vor dem Öffnen der SPS-Programmierumgebung zur Verfügung stehenden Konfigurationsdatei(en) und Gerätedatei(en) können Sie, wenn Sie ein Element im Konfigurationsbaum selektiert haben, weitere Elemente einfügen. Dazu stehen verschiedene Befehle zur Verfügung:

Menüpunkt <Einfügen> <Element einfügen>: Ein Element kann ausgewählt und vor dem selektierten Element eingefügt werden.

Menüpunkt <Einfügen> <Unterelement anhängen>**:** Ein Element kann ausgewählt und als letztes Unterelement an das selektierte Element angefügt werden.

Die jeweils wichtigsten Befehle finden Sie im Kontextmenü (rechte Maustaste).

Ersetzen/Umschalten von Elementen, <Extras> <Element ersetzen>

Erlaubt es die Definition in der Konfigurationsdatei, können Sie ein im Konfigurationsbaum selektiertes Element durch ein anderes ersetzen. Dies umfasst auch das Umschalten von Kanälen, die so konfiguriert sind, dass sie als Ein- oder als Ausgang eingesetzt werden können. Verwenden Sie den Menüpunkt <Extras> <Element ersetzen>.

Neuberechnen der Modul-Adressen, <Extras> <Adressen berechnen>

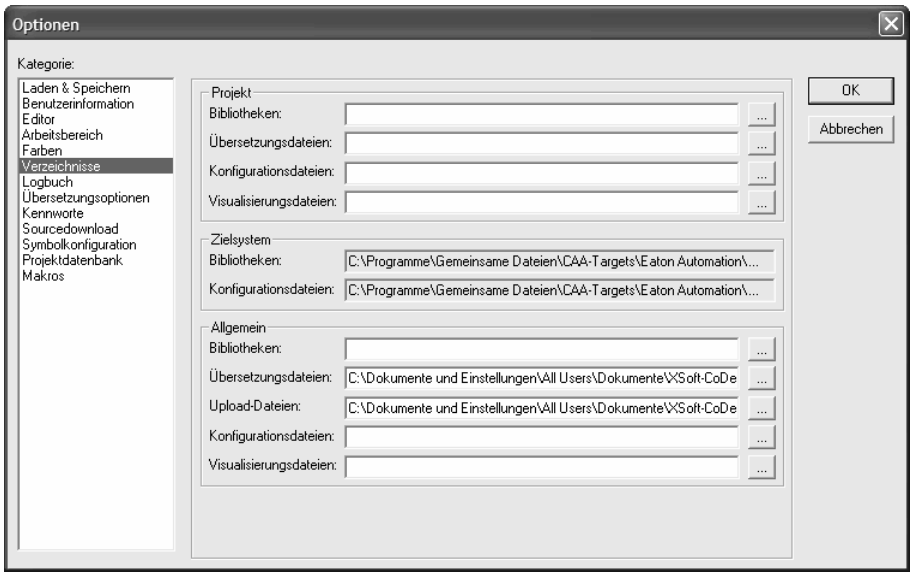
Ist in den allgemeinen Einstellungen der Steuerungskonfiguration die Option **Adressen automatisch** aktiviert, kann über den Menüpunkt <Extras> <Adressen berechnen> eine neue Adressberechnung durchgeführt werden, die ab dem selektierten Modul alle folgenden Elemente erfasst.

Zurück zur Standardkonfiguration, <Extras> <Standardkonfiguration>

Über den Menüpunkt <Extras> <Standardkonfiguration> kann nach Änderungen im Konfigurationseditor die ursprüngliche Steuerungskonfiguration wiederhergestellt werden. Diese basiert auf der Konfigurationsdatei des entsprechenden SPS-Zielsystems.

5.1.1 Projektspezifische Konfigurationsdatei(en) und Gerätedatei(en)

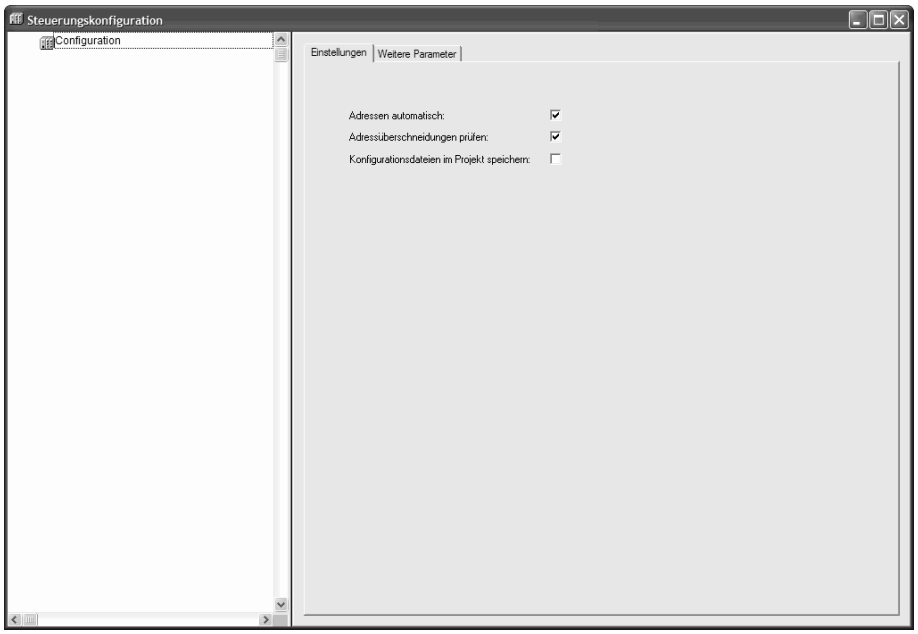
Kundenspezifische Verzeichnisse für Konfigurationsdatei(en) und Gerätedatei(en) können über den Menüpunkt <Projekt> <Optionen> projektspezifisch parametriert werden. Im projektspezifischen Bibliotheks- und Konfigurationspfad können zum Projekt relative Verzeichnispfade parametriert werden (z.B. *.\Libraries, .\PLCconf*).



→ Nach dem Eintragen von projektspezifischen Verzeichnissen für die Konfigurations- und Gerätedatei(en) muss das Projekt geschlossen werden.

Nach dem anschliessenden Öffnen des Projekts sind die zusätzlichen Konfigurations- und Gerätedatei(en) in der Steuerungskonfiguration sichtbar und können eingefügt werden.

5.2 Allgemeine Einstellungen



Adressen automatisch

Bei aktivierter Option erhält jedes neu hinzugefügte Modul automatisch eine Adresse, die sich aus der des zuvor eingefügten Moduls plus dessen Grösse ergibt. Wird ein Modul aus der Konfiguration entfernt, werden die Adressen der nachfolgenden Module automatisch angepasst. Über den Menüpunkt <Extras> <Adressen berechnen> werden die Adressen ab dem aktuell ausgewählten Knoten (Modul) neu ermittelt.

Adressüberschneidungen prüfen

Bei aktivierter Option werden beim Übersetzen des Projekts Adressüberschneidungen überprüft und gemeldet.

Konfigurationsdateien im Projekt speichern

Bei aktivierter Option werden über die Steuerungskonfiguration in das Projekt eingefügte Konfigurationsdatei(en) und Gerätedatei(en) im Projekt abgespeichert.

5.3 Konfiguration als Profibus-Master

Die SPS-Programmierumgebung unterstützt eine Hardwarekonfiguration gemäss Profibus-DP Standard. Voraussetzung ist eine Konfigurationsdatei, die das Einfügen von Profibus-Modulen zulässt.

Die im definierten Verzeichnis für die Konfigurationsdateien abgelegten GSD-Dateien (**G**eräte **S**tamm **D**atei) stehen in der Konfiguration zur Verfügung. In einer GSD-Datei sind die Einstellungsmöglichkeiten eines Moduls beschrieben.

Die Module erhalten eine Konfiguration, die Ihr zeitliches und fehlerbezogenes Verhalten im Nachrichtenaustausch beschreibt.

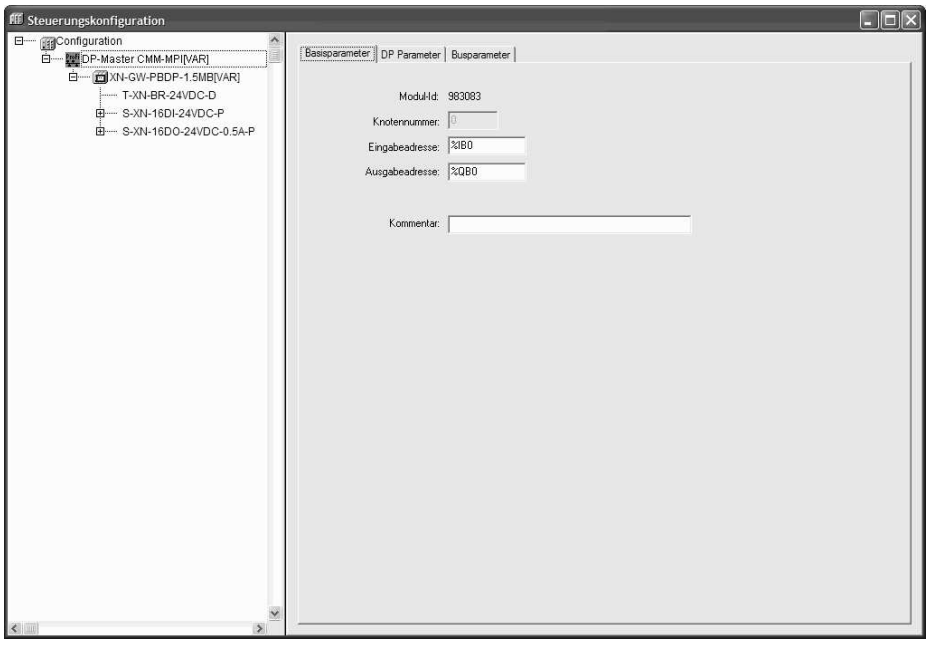
→ In der Steuerungskonfiguration definierte Nachrichten, dessen Ein- bzw. Ausgänge im SPS-Programm nicht verwendet werden, werden standardmässig im Prozessabbild nicht aktualisiert.

5.3.1 Konfiguration des Profibus-Masters

In der Steuerungskonfiguration ist folgender Profibus-Master implementiert:

Bezeichnung	Beschreibung
DP-Master CMM-MPI (MPDP4D09.GSD)	Profibus-Master DP-V1 für Profibus DP / FMS und MPI

5.3.1.1 Basisparameter



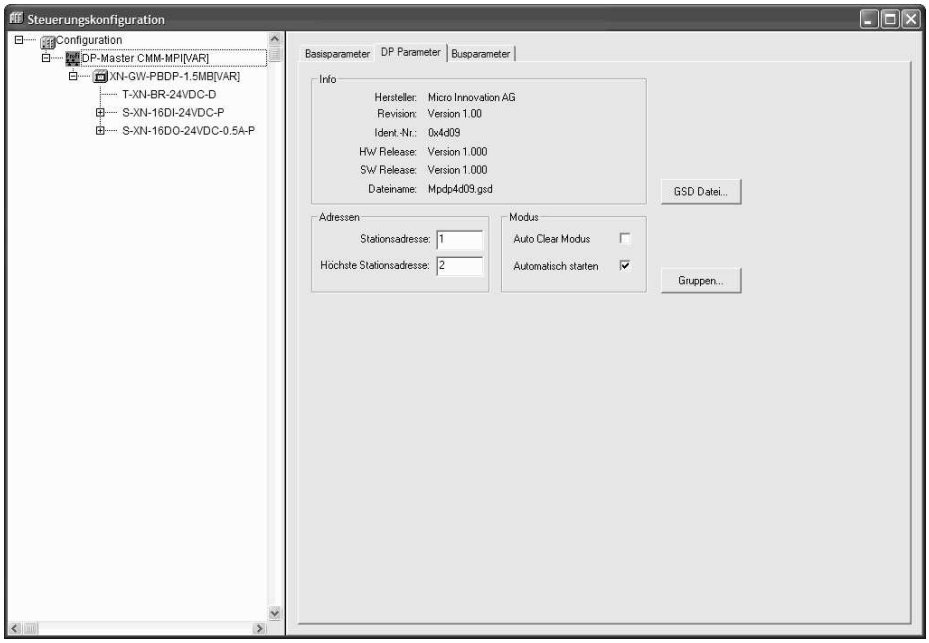
Knotennummer

Die Knotennummer ergibt sich aus einem Eintrag in der Konfigurationsdatei bzw. aus der Position im Konfigurationsbaum und kann vom Anwender nicht geändert werden (Nicht zu verwechseln mit der Node-Id, die vom Anwender parametriert werden kann).

Eingabeadresse, Ausgabeadresse

Hier werden die IEC-Adressen eingegeben, ab welchen die Eingänge bzw. Ausgänge im Projekt angesprochen werden können. Welche Adressen bereits vorgegeben, welcher Adressmodus in der Konfiguration verwendet wird und ob an dieser Stelle noch editiert werden kann, hängt von den allgemeinen Einstellungen und den Definitionen in der Konfigurationsdatei ab.

5.3.1.2 DP Parameter



Stationsadresse

Die **Stationsadresse** ist die eindeutige Kennung des Profibus-Masters. Sie ist mit dem Wert 1 vordefiniert und muss vom Anwender nicht editiert werden.

Höchste Stationsadresse

Sie entspricht der höchsten auf dem Profibus vorhandenen Stationsadresse (vom Anwender einstellbar, oder berechnet durch die SPS-Programmierumgebung).

→ Vergewissern Sie sich, dass eine korrekte HSA eingetragen ist!
Folgende HSAs werden unterstützt: 15, 31, 63 und 126.

Auto Clear Modus

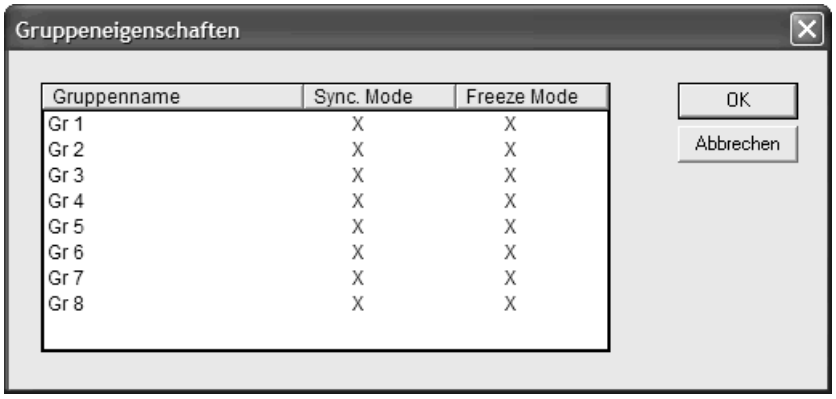
Jeder Slave hat einen Watchdog, der überwacht, ob der Slave regelmäßig vom Master angesprochen wird. Ist dies nicht der Fall, geht der Slave eigenständig in den sicheren Zustand. Bedeutet der Ausfall von Slaves ein hohes Sicherheitsrisiko in einer Anlage, wird mit der Option **Auto Clear Modus** bewirkt, dass der Master bei Ausfall eines Slaves automatisch vom Zustand Operate in den Zustand Clear wechselt. Dies führt wiederum dazu, dass alle anderen in den sicheren Zustand gehen, d.h. digitale Ausgänge werden auf 0 gesetzt und analoge geben vorher projektierte Ersatzwerte aus (FailSafe).

Automatisch Starten

Bei aktivierter Option **Automatisch starten** wird beim Start des SPS-Programms der Profibus automatisch initialisiert und gestartet.

→ Muss immer angewählt sein (ein manueller Start wird nicht unterstützt)

Gruppen



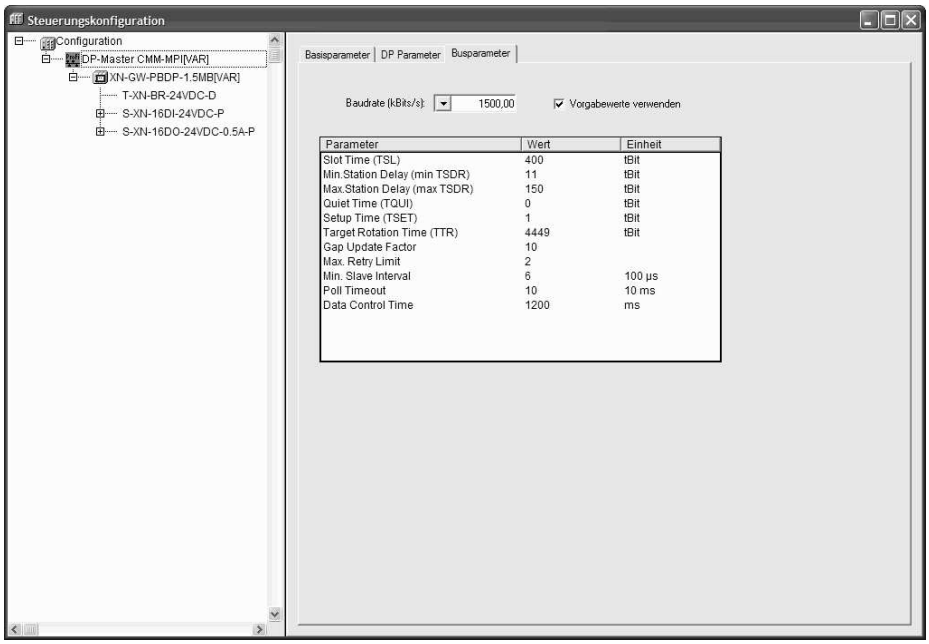
→ Freeze- und Sync-Mode werden nicht unterstützt.

Die Schaltfläche **Gruppen** führt zum Dialog 'Gruppeneigenschaften'. Die Gruppeneigenschaften beziehen sich auf die dem Master zugeordneten Slaves. Bis zu acht Gruppen können eingerichtet werden. Stellen Sie für jede Gruppe ein, ob sie im **Freeze-Mode** und/oder **Sync-Mode** betrieben werden soll. Durch die Zuordnung der Slaves zu verschiedenen Gruppen kann der Datenaustausch vom Master über ein Global-Control-Kommando synchronisiert werden. Mit einem Freeze-Kommando veranlasst ein Master einen Slave oder eine Gruppe, die Eingänge im momentanen Zustand ‚einzufrieren' und diese Daten beim darauf folgenden Datenaustausch zu übertragen. Mit einem Sync-Kommando werden die Slaves veranlasst, die im folgenden Datenaustausch vom Master empfangenen Daten mit dem nächsten Sync-Kommando zeitlich synchron an die Ausgänge durchzuschalten.

Im Dialog ’Gruppeneigenschaften’ lassen sich die Gruppennamen editieren sowie die entsprechenden Übertragungsarten aktivieren bzw. deaktivieren.

5.3.1.3 Busparameter

Die Busparameter beschreiben das Zeitverhalten der Kommunikation.



Baudrate

Baudrate mit der zwischen dem Profibus-Master und den Profibus-Slaves kommuniziert wird.

Vorgabewerte verwenden

Ist die Option **Vorgabewerte verwenden** aktiviert, wird ein Default-Parametersatz für diese Baudrate verwendet. Wird diese Option nicht verwendet, können die Parameter manuell verändert werden.

→ Diese Option nimmt keine Rücksicht auf die konfigurierten I/Os. Wenn optimierte Parameter unter Berücksichtigung der Bustopologie benötigt werden, so müssen diese Parameter mit externen Profibus-Konfiguratoren (z.B. Sycon von Hilscher) ermittelt werden.

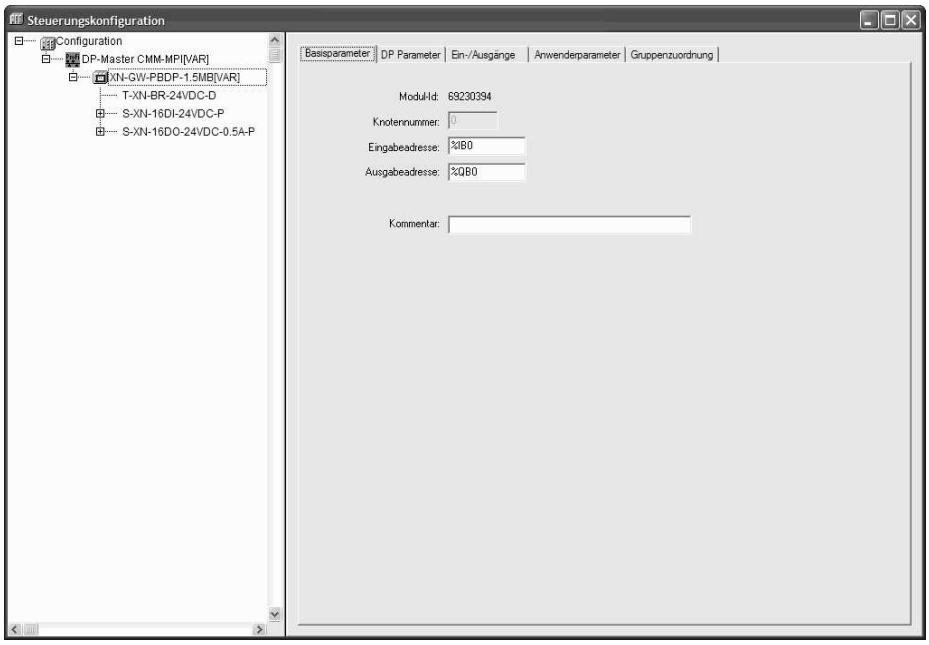
Parameter

Zu jedem Objekt werden **Name, Wert** und **Einheit** angegeben. Der Wert kann verändert werden. Markieren Sie den Wert und drücken Sie die <Leertaste>. Nach Änderung können Sie den Wert über die <Eingabetaste> bestätigen oder mit der <Escape-Taste> verwerfen.

Parameter	Beschreibung
Slot Time (TSL)	Zeit, die der Master nach Aussendung eines Aufruf-Telegramms maximal auf den Empfang des ersten Zeichens des Antwort-Telegramms eines Slaves wartet.
Min. Station Delay (Min. TSDR)	Minimale Reaktionszeit, nach der ein Slave antworten darf.
Max. Station Delay (Max. TSDR)	Maximale Reaktionszeit, innerhalb der ein Slave antworten muss.
Quiet Time (TQUI)	Ruhezeit, die beim Umsetzen von NRZ-Signalen (Non Return to Zero) auf andere Kodierungen zu berücksichtigen ist (Umschaltzeit für Repeater).
Setup Time (TSET)	Zeit, die zwischen dem Empfang eines Telegramms und der darauf folgenden Reaktion im Slave vergehen darf.
Target Rotation Time (TTR)	Token-Soll-Umlaufzeit: Projektiertes Zeitintervall, in dem ein Master den Token erhalten soll. Ergibt sich aus der Summe der Token-Halte-Zeiten aller Master am Bus.
Gap Update Factor (GAP)	GAP-Aktualisierungsfaktor: Anzahl der Busumläufe, nach denen im GAP des Masters (Adressbereich von der eigenen Busadresse bis zur Adresse des nächsten aktiven Teilnehmers) nach einer weiteren, neu hinzugekommenen, aktiven Station gesucht wird.
Max. Retry Limit	Maximale Anzahl der erneuten Aufrufversuche des Masters, wenn er vom Slave keine gültige Antwort empfangen hat.
Min. Slave Interval	Zeit zwischen zwei Buszyklen, in denen ein Slave eine Anforderung des Masters bearbeiten kann (Zeitbasis 100µs). Der hier eingetragene Wert muss mit den jeweiligen Vorgaben in den GSD-Dateien der Slaves abgestimmt sein.
Poll Timeout	Maximale Zeit, nach der die Antwort des Masters bei einer Master-Master-Kommunikation vom Requester (DP-Master Klasse 2) abgeholt sein muss (Zeitbasis 1ms).
Data Control Time	Zeit, in der der Master den zugeordneten Slaves seinen Betriebszustand mitteilt. Gleichzeitig überwacht der Master, ob mit den Slaves innerhalb dieser Zeit jeweils mindestens ein Nutzdatenaustausch stattgefunden hat und aktualisiert die Data_Transfer_List.

5.3.2 Konfiguration der Profibus-Slaves

5.3.2.1 Basisparameter



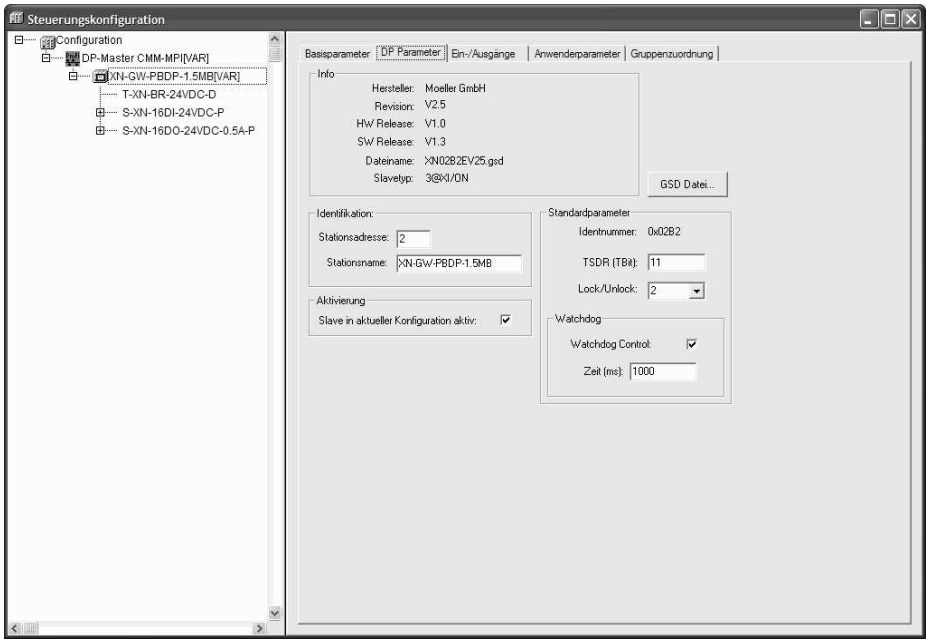
Knotennummer

Die Knotennummer ergibt sich aus einem Eintrag in der Konfigurationsdatei bzw. aus der Position im Konfigurationsbaum und kann vom Anwender nicht geändert werden (Nicht zu verwechseln mit der Stationsadresse, die vom Anwender parametriert werden kann).

Eingabeadresse, Ausgabeadresse

Hier werden die IEC-Adressen eingegeben, ab welchen die Eingänge bzw. Ausgänge im Projekt angesprochen werden können. Welche Adressen bereits vorgegeben, welcher Adressmodus in der Konfiguration verwendet wird und ob an dieser Stelle noch editiert werden kann, hängt von den allgemeinen Einstellungen und den Definitionen in der Konfigurationsdatei ab.

5.3.2.2 DP Parameter



Stationsadresse

Die **Stationsadresse** ist die eindeutige Kennung des Profibus-Moduls. Sie entspricht der am Modul selbst eingestellten Nummer zwischen 0 und 126 und muss dezimal eingegeben werden.

Aktivierung

Bei deaktivierter Option **Aktivierung**, werden die Konfigurationsdaten des Slaves beim Download an den Koppler übertragen, ein Datenaustausch über den Bus findet jedoch nicht statt.

Standardparameter

Die **Identnummer** wird von der PNO vergeben und ist die eindeutige Identifikationsnummer des entsprechenden Gerätetyps. Sie stellt die eindeutige Referenz zwischen Slave und der zugehörigen GSD-Datei her. Der Parameter **TDSR** (Time Station Delay Responder) entspricht der Reaktionszeit nach der der Slave frühestens dem Master antworten darf. Zeiteinheit für die Übertragung eines Bits über PROFIBUS: Kehrwert der Übertragungsrate; z.B. 1 TBit bei 12MBaud=1/12.000.000 Bit/sek=83ns. Über den Parameter **Lock/Unlock** wird der Slave für andere Master gesperrt oder freigegeben:

0: min. TSDR und slave-spezifische Parameter dürfen überschrieben werden

1: Slave für andere Master freigegeben

2: Slave für andere Master gesperrt, alle Parameter werden übernommen

3: Slave für andere Master erneut freigegeben

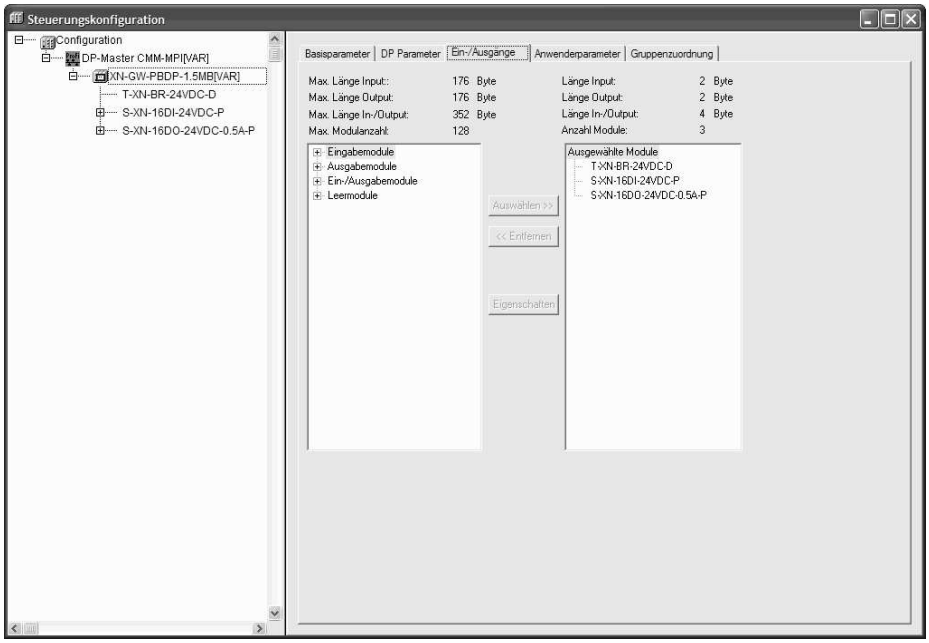
Watchdog

Bei aktivierter Option **Watchdog-Contol** gilt die parametrierte Watchdogzeit. Wird ein Slave nicht innerhalb dieser Zeit vom Master angesprochen, wechselt der Slave in den Initialisierungszustand.

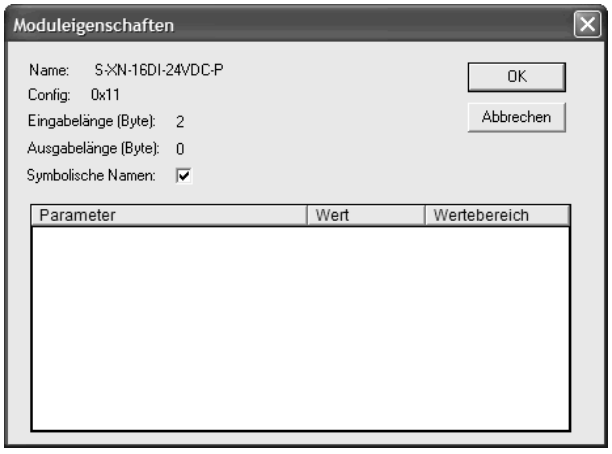
Die Watchdog Control Zeit muss bei allen Slaves gleich eingestellt sein.

5.3.2.3 Modulauswahl

Ist der eingefügte Profibus-Knoten modular aufgebaut und unterstützt er die entsprechenden Standards, so erscheint die entsprechende Registerkarte**.** Die Konfiguration des entsprechenden Aufbaus kann durch **Auswählen** oder **Entfernen** der Module vorgenommen werden.



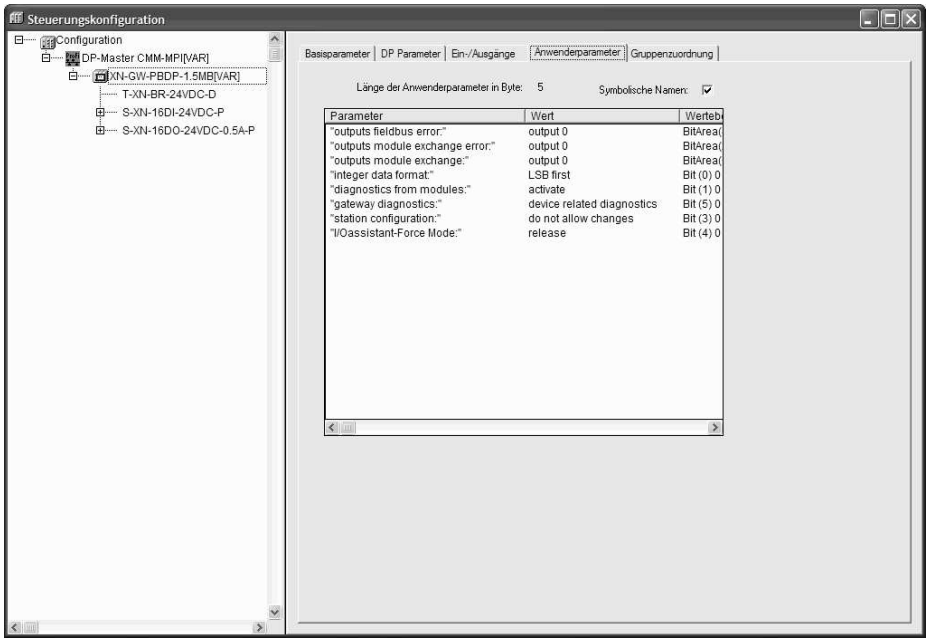
Die Schaltfläche **Eigenschaften** führt zum Dialog 'Moduleigenschaften' des aktuell selektierten Ein- oder Ausgangsmodul. Er zeigt den Namen, die Config (Kodierung der Modulbeschreibung nach PROFIBUS-Norm) und die Ein- und Ausgabelänge des Moduls in Byte. Enthält die Modulbeschreibung in der GSD-Datei neben dem Standardsatz zusätzliche spezifische Parameter, werden diese hier mit Wert und Wertebereich aufgelistet. Ist die Option **Symbolische Namen** aktiviert, werden dabei die symbolischen Namen verwendet.



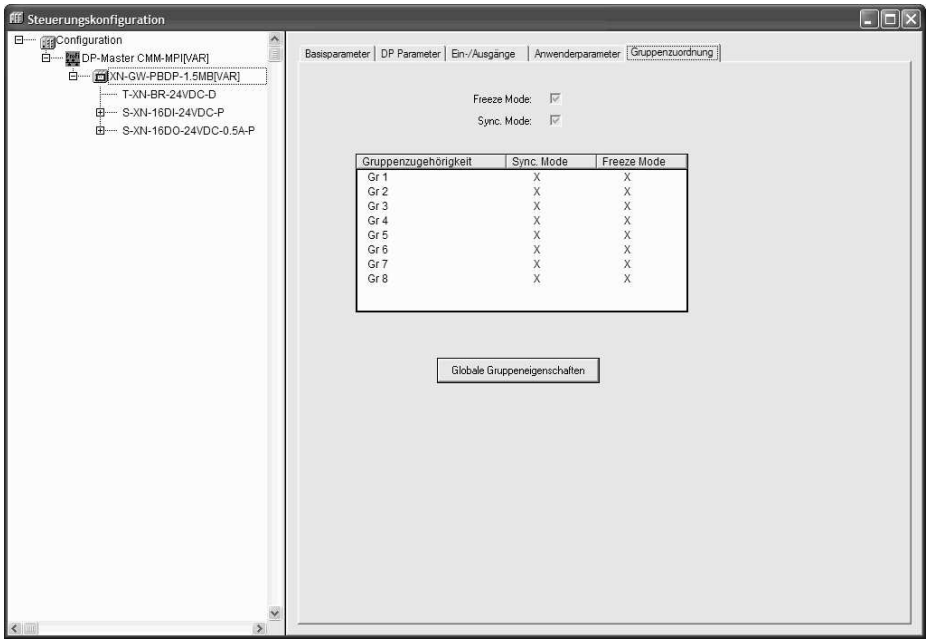
5.3.2.4 Anwenderparameter

Hier sind verschiedene in der GSD-Datei definierte erweiterte Parameter eines Slaves aufgelistet. Die Spalte **Parameter** zeigt den Namen des Parameters. Die in der Spalte **Wert** eingetragenen Parameterwerte können verändert werden. Ausserdem ist der **Wertebereich** angegeben.

Sind in der GSD-Datei für die Parameter auch symbolische Namen vergeben, kann die Option **Symbolische Namen** aktiviert werden, so dass die Werte mit diesen dargestellt werden. Zur Information ist ausserdem über der Tabelle die Länge der Anwenderparameter angegeben.



5.3.2.5 Gruppenzuordnung



→ Freeze- und Sync-Mode werden nicht unterstützt.

Dieser Dialog dient der Zuordnung des Slaves zu einer oder mehrerer der möglichen acht Gruppen. Die allgemeingültigen Gruppeneigenschaften (Sync-Mode und/oder Freeze-Mode) hingegen werden bei der Konfiguration der Master-Eigenschaften definiert. Über die Schaltfläche **Globale Gruppeneigenschaften** gelangt man ebenfalls zu diesem Dialog.

Die Gruppe(n), denen der Slave zugeordnet wurde, werden mit einem Pluszeichen markiert. Das Zuordnen bzw. Entfernen des Slaves zu/aus einer Gruppe erreicht man, indem man den Gruppennamen in der Spalte Gruppenzugehörigkeit selektiert und mit der rechten Maustaste 'Slave zu Gruppe hinzufügen' bzw. 'Slave aus Gruppe entfernen' wählt oder nochmals mit der Maus links neben den Gruppennamen klickt.

Ein Slave-Gerät kann nur solchen Gruppen zugeordnet werden, deren Eigenschaften es unterstützt. Die diesbezüglichen Eigenschaften des jeweiligen Slaves (Sync-Mode / Freeze-Mode) werden oberhalb der Tabelle angezeigt. Vom dem Gerät unterstützte Übertragungsarten sind mit einem Haken versehen.

5.3.3 Busdiagnose

Für die Projektierung der Busdiagnose im SPS-Programm stehen Ihnen diverse Funktionsbibliotheken zur Verfügung.

Die detaillierten Funktionsbeschreibungen entnehmen Sie bitte den entsprechenden Dokumentationen zu den Funktionsbibliotheken.

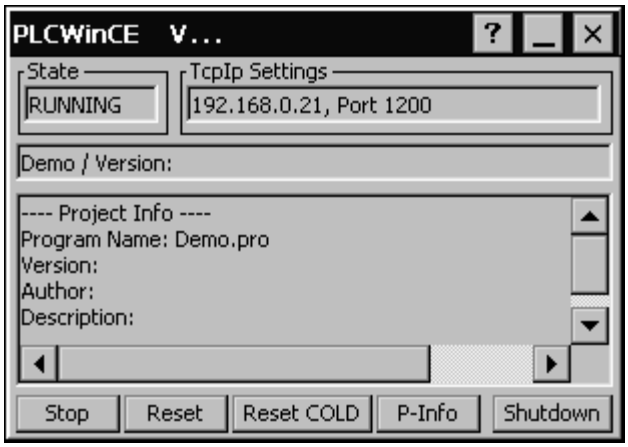
6 Betrieb

6.1 Einschaltverhalten

Nach dem Einschalten führt die Steuerung einen Systemtest durch. Die Steuerung nimmt den Zustand "Stop" oder "Run" erst an, wenn sie keinen Hardwarefehler festgestellt hat. Der Systemtest beinhaltet folgende Tests:

- Speichertest
- SPS-Programmtest

Das Ergebnis des Tests wird im PLCWinCE-Ausgabefenster visualisiert.



Im Weiteren hängt der Zustand der Steuerung vom parametrierten Startverhalten ab (➔ Kap. 6.4)

6.2 Ausschaltverhalten

Bei einem Spannungseinbruch wird die Programmabarbeitung sofort beendet und alle für den Wiederanlauf erforderlichen Informationen werden gespeichert. Nach einem erneuten Einschalten der Spannung führt die Steuerung einen Neustart aus.

6.3 Betriebszustände der Steuerung

Stop

Der Betriebszustand "Stop" ist durch folgende Merkmale charakterisiert:

- Es befindet sich ein SPS-Programm in der Steuerung
- Das SPS-Programm wird nicht durchlaufen

Der Betriebszustand "Stop" wird angenommen:

- Nach Einschalten der Spannung mit parametriertem Startverhalten "Stop" (➔ Kap. 6.4)
- Über die SPS-Programmierumgebung im PC
- Nach einer Zykluszeitüberschreitung

Run

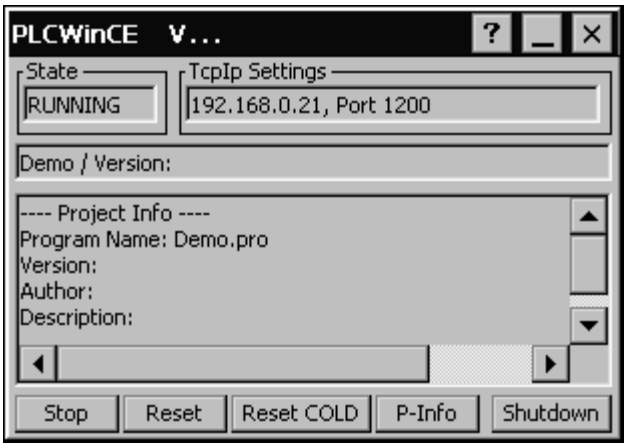
Im Zustand RUN wird das SPS-Programm zyklisch abgearbeitet.

Der Betriebszustand RUN wird eingenommen:

- Nach Einschalten der Spannung mit parametriertem Startverhalten "Warm"
- Über die SPS-Programmierumgebung im PC

6.4 Betriebsartenumschaltung

Die Betriebsartenumschaltung erfolgt über das PLCWinCE-Ausgabefenster oder über die SPS-Programmierumgebung.



Im Weiteren kann das Start-Verhalten nach Einschalten der Spannung über die Datei **PlcRts.reg** im Verzeichnis **PlcRts** parametriert werden.

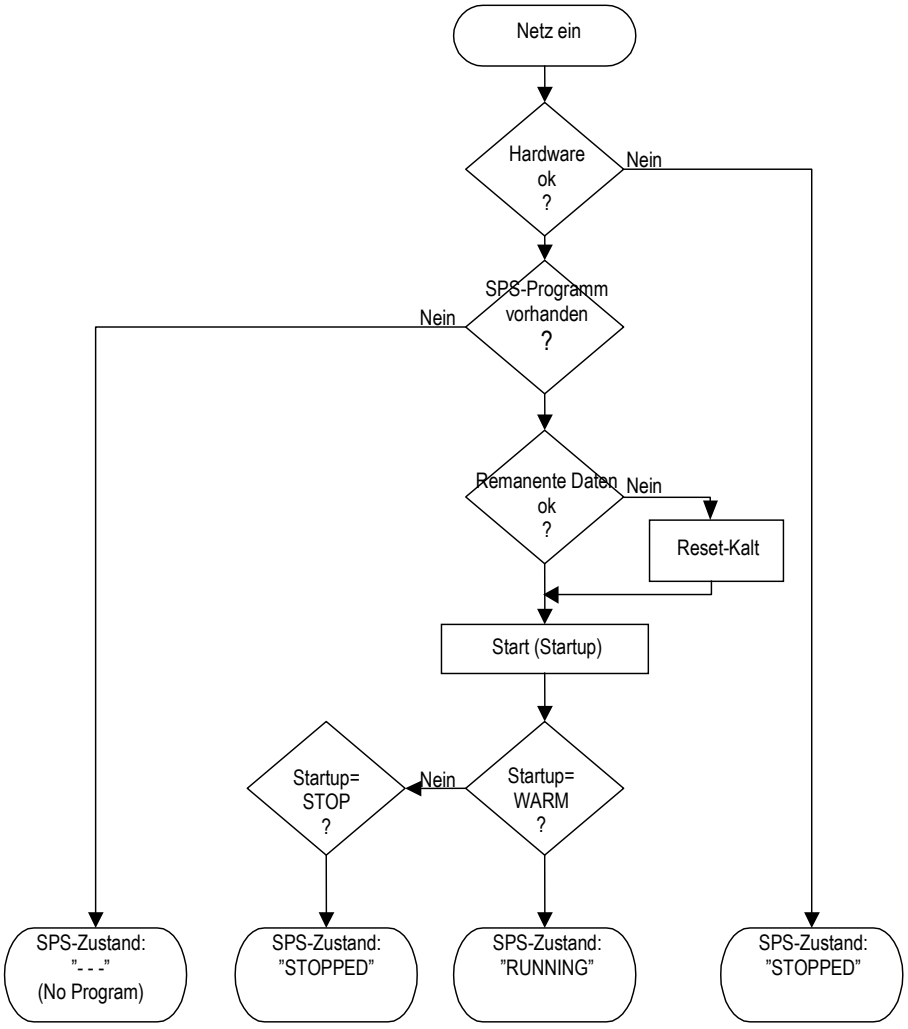
PlcRts.reg
[HKEY_LOCAL_MACHINE\SOFTWARE\Eaton\PLC Runtimesystem]
...
…
"Startup"="WARM"
…
…

Es sind folgende Einstellungen möglich:

- Stop "Startup"="STOP”
- Warmstart (Standardeinstellung) "Startup"="WARM”

6.5 Start-Verhalten

Ablauf



Kaltstart

Ein Kaltstart erfolgt beim ersten Start nach dem Laden des SPS-Programms auf die Steuerung bzw. nach jedem Reset-Kalt. Alle Variablen des SPS-Programms, werden mit ihren Initialisierungswerten initialisiert und das SPS-Programm wird gestartet.

Warmstart

Bei sämtlichen weiteren Starts des geladenen SPS-Programms bzw. nach jedem Reset / Reset-Warm erfolgt ein Warmstart. Die remanenten (RETAIN, RETAIN PERSISITENT) Variablen behalten

ihre Werte, die übrigen Variablen werden mit ihren Initialisierungswerten initialisiert. Variablen ohne explizite Initialisierungswerte werden mit ihren Standardinitialisierungswerten initialisiert.

6.6 Stop-Verhalten

Die Abarbeitung des SPS-Programms wird am Ende des Programmzyklus gestoppt. Anschliessend erfolgt ein Reset / Reset-Warm.

6.7 Reset-Verhalten

Reset / Reset-Warm

Entspricht der Initialisierung eines Warmstarts.

Reset-Kalt

Entspricht der Initialisierung eines Kaltstarts. Dieser Befehl setzt alle Variablen, auch die remanenten (RETAIN, RETAIN PERSISTENT) Variablen auf den Initialisierungswert zurück.

Reset-Ursprung

Dieser Befehl setzt alle Variablen, auch die remanenten (RETAIN, RETAIN PERSISTENT) Variablen auf den Initialisierungswert zurück und löscht das SPS-Programm auf der Steuerung. Die Steuerung wird in den Urzustand zurückversetzt.

6.8 Test und Inbetriebnahme

Die Steuerung unterstützt folgende Tests- und Inbetriebnahmemöglichkeiten:

- Breakpoint/Einzelschritt-Betrieb
- Einzelzyklus-Betrieb
- Forcen
- Online-Änderung

→ Die Ablaufkontrolle (Power Flow) ist nicht implementiert und wird nicht unterstützt.

Breakpoint / Einzelschritt-Betrieb

Innerhalb eines SPS-Programms können Breakpoints gesetzt werden. Bei der Ausführung einer mit einem Breakpoint versehenen Anweisung wird das SPS-Programm an dieser Stelle taskspezifisch angehalten. Die folgenden Anweisungen können im Einzelschritt-Betrieb ausgeführt werden. Die Zykluszeitüberwachung ist in diesem Falle deaktiviert.

→ Das SPS-Programm läuft jeweils nur auf Breakpoints der in der Taskkonfiguration als Debug-Task festgelegten Task auf.

Einzelzyklus-Betrieb

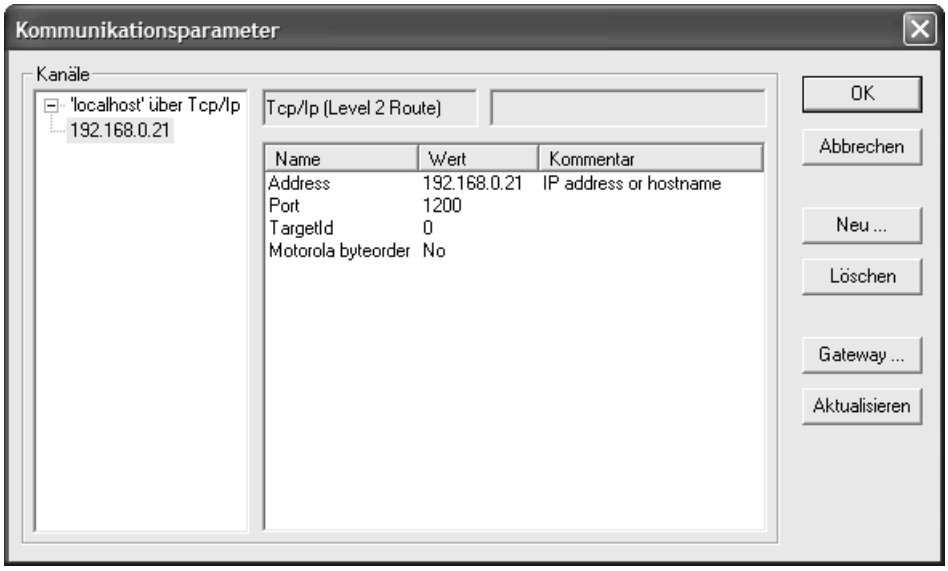
Im Einzelzyklus-Betrieb wird ein einzelner Programmzyklus taskspezifisch ausgeführt. Die Ausgänge sind während dem Programmzyklus freigegeben. Am Ende des Programmzyklus wird das Ausgangabbild gelöscht und die Ausgänge abgeschaltet. Die Zykluszeitüberwachung ist in diesem Falle aktiviert.

Forcen

Alle Variablen eines SPS-Programms können zwangsgesetzt werden.

6.9 Programmtransfer

Wenn das SPS-Programm in der SPS-Programmierumgebung (PC) fehlerfrei übersetzt wurde, kann es in den Arbeitsspeicher der Steuerung geladen und anschliessend gestartet werden. (➔ Kap. 8)



Bei einem Programmtransfer vom PC in die Steuerung wird das Programm in der Steuerung mit dem Programm auf dem PC verglichen. Bei Ungleichheit erfolgt eine Abfrage ob das Programm überschrieben werden soll. Wird diese Abfrage bestätigt, so geht die Steuerung in den Zustand "Stop" und das neue Programm wird in den Arbeitsspeicher geladen.

Kommunikationsfehler

Kann zwischen der SPS-Programmierumgebung und der Steuerung keine Verbindung aufgebaut werden sind folgende Punkte zu überprüfen:

- Physikalische Verbindung
- Kommunikationsparameter in der SPS-Programmierumgebung
- TCP/IP Einstellungen in den Systemeinstellungen PC-seitig
- TCP/IP Einstellungen in den Systemeinstellungen steuerungsseitig

6.9.1 Bootprojekt erzeugen

Ein SPS-Programm wird nur dann netzausfallsicher gespeichert, wenn vor dem Spannungsausfall ein Bootprojekt "online" erzeugt und anschliessend auf die Steuerung übertragen wurde.

→ Das Bootprojekt kann nur "online" erzeugt werden.

7 Programmabarbeitung und Systemzeiten

7.1 Programmabarbeitung

Die zeitliche Ablaufeinheit eines SPS-Programms wird mit "Task" bezeichnet. Zusätzlich zum eigentlichen SPS-Programm werden alle anfallenden Systemaktivitäten bearbeitet.

Hierunter fallen z.B. folgende Systemaktivitäten:

- Kommunikation mit der SPS-Programmierumgebung
- Online-Änderungen
- Bearbeitung des CANopen-Protokollstacks
- Berabeitung des Prozessabbilds (IO-Update)

Programmabarbeitung ohne Taskkonfiguration

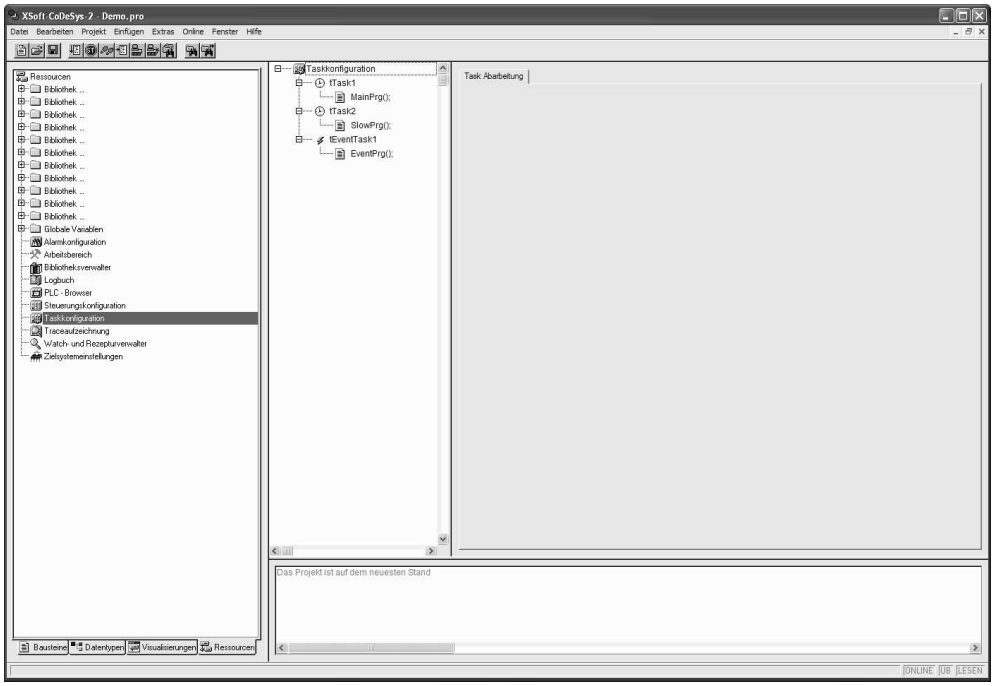
Der erste Baustein, der in einem neuen SPS-Programm angelegt wird, trägt automatisch den Namen **PLC_PRG.** Dort startet die Ausführung. Somit arbeitet die Steuerung das PLC_PRG zyklisch freilaufend mit einem Minimalintervall von 10 ms ab.

Programmabarbeitung mit Taskkonfiguration

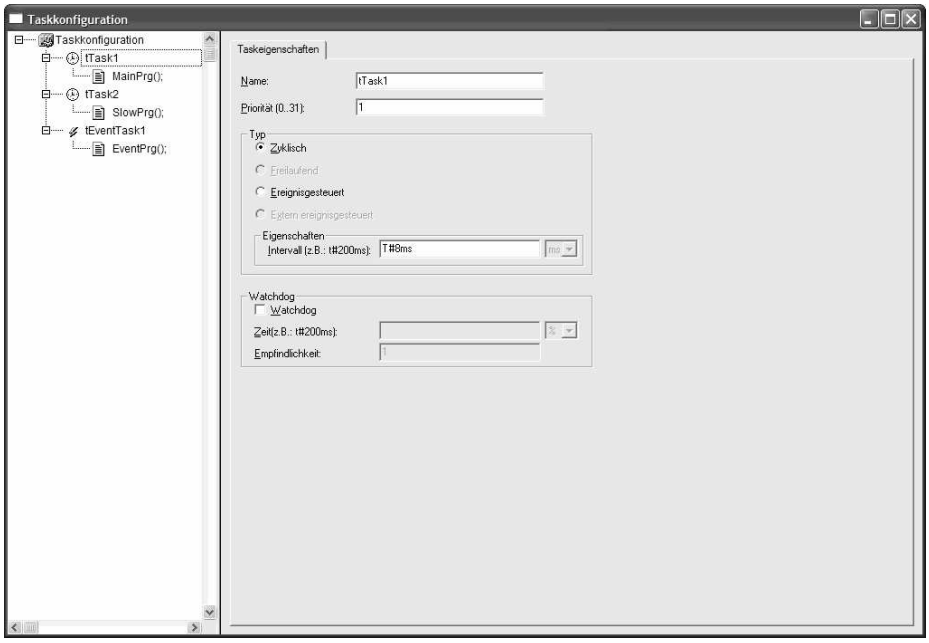
Neben dem Deklarieren des speziellen Programmbausteins PLC_PRG kann die Abarbeitung des SPS-Programms auch über die Taskkonfiguration gesteuert werden. Wenn eine Taskkonfiguration definiert wurde, muss kein Programmbaustein mit dem Namen PLC_PRG angelegt werden.

7.2 Taskkonfiguration

Die Taskkonfiguration wird über die Funktionalität <Taskkonfiguration> im Registerblatt 'Ressourcen' erreicht.



Jeder Task kann eine Folge von Programmen zugeordnet werden, die beim Aufruf der Task ausgeführt werden sollen. Die Taskpriorität und Taskbedingung legen fest, in welcher zeitlichen Abfolge die Tasks abgearbeitet werden. Ein SPS-Programm kann aus mehreren Tasks gleicher oder verschiedener Priorität bestehen, die zyklisch in parametrierbaren Zeitintervallen oder beim Eintreffen von Ereignissen abgearbeitet werden.



Die korrekte Definition einer Task besteht aus folgenden Parametern:

- Taskname
- Taskpriorität
- Tasktyp bzw. Taskbedingung
- Tasküberwachung / Watchdog

Taskname

Eindeutiger Taskname

Taskpriorität

Um die Tasks zu priorisieren, können unterschiedliche Taskprioritäten vergeben werden. Die Taskprioritäten können zwischen 0 und 31 parametriert werden. 0 entspricht der höchsten und 31 der niedrigsten Priorität.

→ Tasks mit gleicher Priorität unterbrechen sich nicht gegenseitig.

Tasktyp bzw. Taskbedingung

Die Taskbedingung kann entweder ein Zeitintervall sein, nach dem die Task ausgeführt werden soll (zyklisch), oder eine globale Variable, die im Falle einer steigenden Flanke deren Ausführung bewirkt (ereignisgesteuert).

Eine ereignisgesteuerte Task setzt eine zyklisch ablaufende Task voraus. Im Programm der zyklischen Task ist das Ereignis zu programmieren. Jede steigende Flanke des Ereignisses führt anschliessend zur Ausführung des Programms der ereignisgesteuerten Task.

Tasküberwachung / Watchdog

Jede Task kann mit Hilfe des Watchdogs überwacht werden (➔ Kap. 7.4)

7.3 Multitasking

Das SPS-Laufzeitsystem ist ein Multitaskingsystem. Das bedeutet, mehrere Tasks werden quasi zeitgleich abgearbeitet. In einem Multitaskingsystem können sich einzelne Task mit unterschiedlicher Priorität unterbrechen.

Die Folge daraus ist, dass die Konsistenz des Ein-/Ausgangsabbildes nur innerhalb der Task gewährleistet ist, die als einzige auf das Ein-/Ausgangsabbilds zugreift. Greifen mehrere Tasks unterschiedlicher Priorität auf die gleichen Ein-/Ausgänge zu, ist die Konsistenz nur bei der Task mit der höchsten Priorität gegeben. (➔ Kap. 7.9)

→ Vermeiden Sie grundsächlich einen Zugriff auf physikalische Ein-/Ausgänge aus mehreren Tasks heraus, um so einen eindeutigen Steuerungsablauf sicher zu stellen.

7.4 Tasküberwachung / Zykluszeitüberwachung

Die Zyklusüberwachung überwacht die zyklischen Tasks des SPS-Programms. Die Zyklusüberwachung unterbricht die Programmbearbeitung, wenn die Task eine definierte Zeit in einer definierten Häufigkeit überschreitet. Standardmässig werden die Ausgänge der Steuerung beim Auslösen der Zykluszeitüberwachung abgeschaltet und die Steuerung wird in den Zustand "Stop" gesetzt. Danach muss das SPS-Programm mit "Reset" zurückgesetzt werden.

Die korrekte Definition der Tasküberwachung besteht aus folgenden Parametern:

- Watchdog ein/aus
- Watchdogzeit
- Watchdog-Empfindlichkeit

→ Ist in der Taskkonfiguration der Watchdog der entsprechenden Task deaktiviert, erfolgt keine Zykluszeitüberwachung!

→ In einem SPS-Programm ohne Taskkonfiguration ist die Zykluszeitüberwachung standardmässig mit einer Zykluszeitüberwachung von 10ms parametriert und aktiviert.

Funktionsweise

Der Watchdog wird zu Beginn eines jeden Abarbeitungszyklus gestartet und nach erfolgreichem Abarbeiten beendet. Dauert die Tasklaufzeit länger als das definierte Taskintervall wird die Zykluszeitüberwachung ausgelöst.

Das Auslösen der Zyklusüberwachung ist weiterhin abhängig von der Watchdog-Empfindlichkeit. Die Watchdog-Empfindlichkeit bestimmt, nach wie vielen unmittelbar aufeinander folgenden Tasklaufzeit-Überschreitungen die Zyklusüberwachung ausgelöst wird.

Der Watchdog wird ausgelöst:

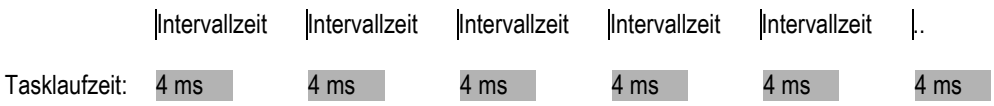
- bei einer Watchdog-Empfindlichkeit von 1 gleich bei der ersten Überschreitung der Watchdogzeit.
- bei einer Watchdog-Empfindlichkeit von x erst beim x-ten unmittelbar aufeinander folgendem Überschreiten der Watchdogzeit.
- Darüber hinaus wird der Watchdog bei einer Endlosschleife ausgelöst, wenn die Ausführungszeit der Task länger dauert als das Ergebnis aus Watchdogzeit * Watchdog-Empfindlichkeit. Dieses Kriterium dient dem Erkennen und Reagieren auf Endlosschleifen bei vorgewählter Watchdog-Funktionalität.

Wenn Sie z. B. bei einer Task die Watchdogzeit von 10ms und die Watchdog-Empfindlichkeit von 5 eintragen, wird die Task spätestens nach 10ms * 5 = 50ms beendet.

Folgende Beispiele sollten das Zusammenspiel von Taskintervall, Tasklaufzeit und Watchdogzeit verdeutlichen.

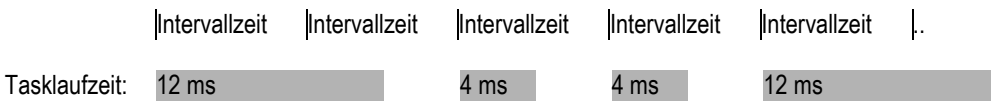
- parametriertes Taskintervall = 8ms
- parametrierte Watchdogzeit = 8ms
- parametrierte Empfindlichkeit = 2

Beispiel 1



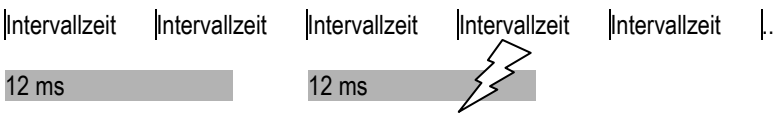
Die Zykluszeitüberwachung wird nicht ausgelöst, da die Tasklaufzeit immer unter dem parametrierten Taskintervall bleibt.

Beispiel 2



Die Zykluszeitüberwachung wird nicht ausgelöst, da nicht 2 aufeinander folgende Tasklaufzeit-Überschreitungen auftreten.

Beispiel 3



Die Zykluszeitüberwachung wird 8ms nach Beginn der zweiten Taskabarbeitung ausgelöst, da beide Tasklaufzeiten länger als die angegebene Watchdogzeit sind und unmittelbar aufeinander folgen.

Beispiel 4 (Endlos-Schlaufe)

Die Zykluszeitüberwachung wird ausgelöst, da die Tasklaufzeit länger dauert als die Multiplikation von Wachdogzeit und Empfindlichkeit.¨

7.5 Datenremanenz

Die Steuerung besitzt einen Speicherbereich für remanente Daten (RETAIN, RETAIN PERSISTENT). Diese Daten werden bei einem Abschalten der Steuerung gespeichert und bleichen somit erhalten.

Da unter Umständen bei einem Spannungsaus-/abfall die Programmabarbeitung inmitten des Programmzyklus abgebrochen wird, sind die Daten des aktuellen Programmzyklus nicht konsistent.

Beim nächstfolgenden Start wird die Steuerung mit den unmittelbar bei Spannungsaus-/abfall noch korrekt geschriebenen Daten initialisiert.

Variablenverhalten

Aktion	VAR	VAR RETAIN	VAR PERSISTENT	VAR RETAIN PERSISTENT
Stop → Start	Initialisierungswert	Wert bleibt erhalten	Initialisierungswert	Wert bleibt erhalten
Warmstart bzw. Reset / Reset-Warm	Initialisierungswert	Wert bleibt erhalten	Initialisierungswert	Wert bleibt erhalten
Kaltstart bzw. Reset-Kalt	Initialisierungswert	Initialisierungswert	Initialisierungswert	Initialisierungswert
Reset-Ursprung	Initialisierungswert	Initialisierungswert	Initialisierungswert	Initialisierungswert
'Alles Bereinigen' und anschliessendem Programmdownload	Initialisierungswert	Initialisierungswert	Wert bleibt erhalten	Wert bleibt erhalten

→ Das Variablenverhalten globaler Variablen entspricht auch dem Variablenverhalten lokaler Variablen.

7.6 Direkter Peripheriezugriff

→ Direkte Peripheriezugiffe (z.B. Prozessabbild) sind zielsystemspezifisch implementiert.

7.7 Interruptverarbeitung

→ Eine direkte Interruptverarbeitung wird nicht unterstützt.

7.8 Systembibliotheken, Funktionsbausteine und Funktionen

Für die Projektierung des SPS-Programms stehen Ihnen diverse Funktionsbibliotheken zur Verfügung.

→ Die detaillierten Funktionsbeschreibungen entnehmen Sie bitte den entsprechenden Dokumentationen zu den Funktionsbibliotheken.

7.9 Prozessabbildverarbeitung / IO-Update

Beim Übersetzen des SPS-Programms wird jeder Task die Konfiguration der in der Task verwendeten Ein- bzw. Ausgänge angehängt.

Eingangsabbild

Eine Anweisung "myVar := %IX0.0" führt in der Konfigurationsdatei zu einem Eintrag, welcher beschreibt, dass der Wert des Eingangs %IX0.0 verarbeitet werden muss. Wird gleichzeitig das Byte %IB0 verwendet, wird anstelle des bitweisen Zugriffs das ganze Byte verarbeitet.

Diese Konfiguration wird taskspezifisch erstellt und der Task angehängt. Anhand dieser Konfiguration werden nun am Anfang der Task die Eingänge gelesen.

Ausgangsabbild

Eine Anweisung "%QX0.0 := myVar" führt in der Konfigurationsdatei zu einem Eintrag, welcher beschreibt, dass der Wert des Ausgangs %QX0.0 verarbeitet werden muss. Wird gleichzeitig das Byte %QB0 verwendet, wird anstelle des bitweisen Zugriffs das ganze Byte verarbeitet.

Diese Konfiguration wird taskspezifisch erstellt und der Task angehängt. Anhand dieser Konfiguration werden nun am Ende der Task die Ausgänge geschrieben.

→ Bei der Projektierung soll konkurrierender Zugriff vermieden werden (d.h. Zugriff auf die gleiche Ein-/Ausgangsadresse aus mehrerer Tasks). Dies kann zu einem inkonsistenten Ein-/Ausgangsabbild führen!

Bei konkurrierendem Zugriff werden beim Kompilieren in der SPS-Programmierumgebung Warnungen ausgegeben.

Infolge unterschiedlichen Bus-Systemen und den entsprechenden Übertragungsmechanismen unterscheidet sich die Verarbeitung des Ein-/Ausgangsabbilds.

7.9.1 Profibus

Profibus-Master CMM-MPI

Die Eingänge und Ausgänge werden in einem zusätzlichen Systemtask (Treiber-Task) zyklisch aktualisiert. Dies erfolgt asynchron zu den Tasks des SPS-Programms.

Die jeweils verwendeten Eingänge werden am Anfang der Task von der Treiber-Task gelesen und in das Eingangsabbild übernommen.

Die jeweils verwendeten Ausgänge werden am Ende der Task vom Ausgangsabbild übernommen, an die Treiber-Task übergeben und anschliessend geschrieben.

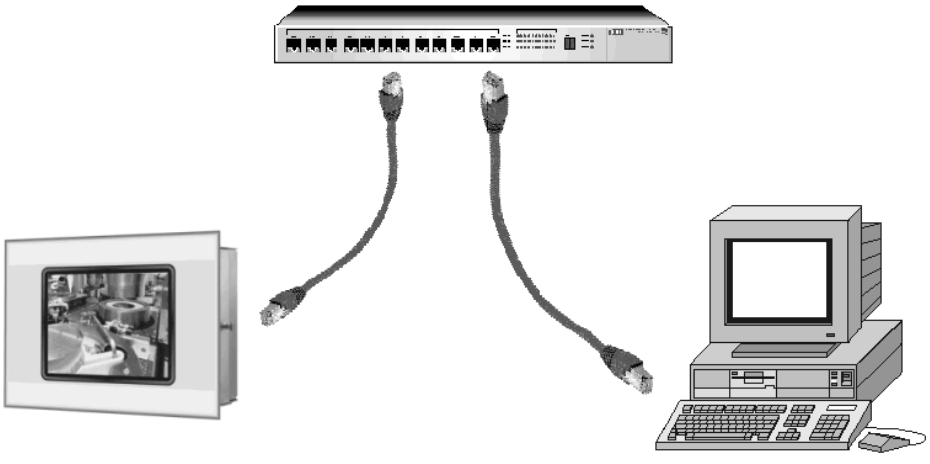
8 Verbindungsaufbau Programmier-PC – Steuerung

8.1 Verbindungsaufbau über Ethernet

Die Kommunikation vom Programmier-PC auf die Steuerung erfolgt mit Ethernet und dem TCP/IP Protokoll. Bei einer direkten Verbindung, ohne Ethernet-Hub oder Switch, ist ein gekreuztes "Crossover" Kabel zu verwenden. Steuerungsseitig ist eine standardisierte RJ45 Buchse angebracht.

gekreuztes RJ45 TwistedPair-Kabel

Erfolgt die Verbindung über einen Ethernet-Hub bzw. Switch, sind von Hub zur Steuerung wie zum Programmier-PC nicht gekreuzte Kabel einzusetzen.

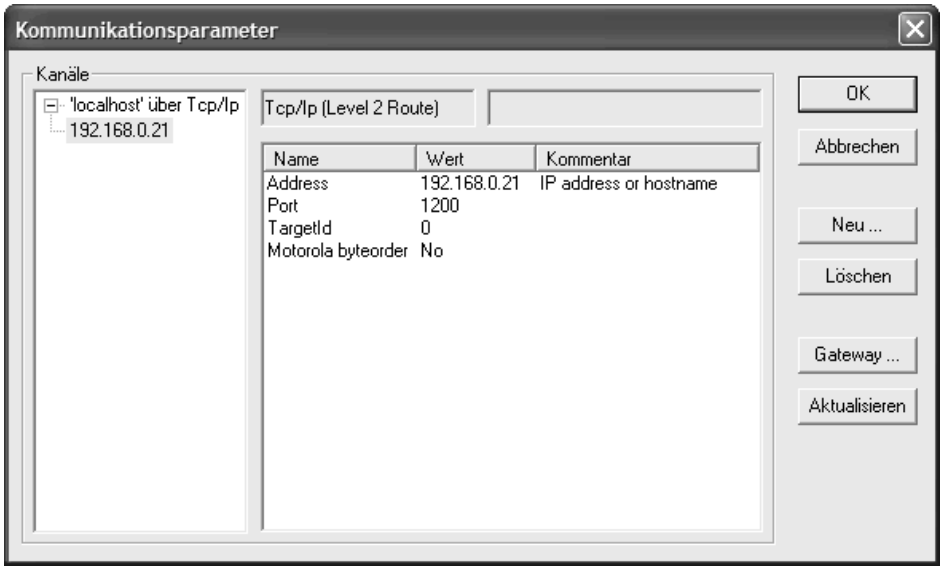


nicht gekreuzte RJ45 TwistedPair-Kabel

Kommunikationsparameter

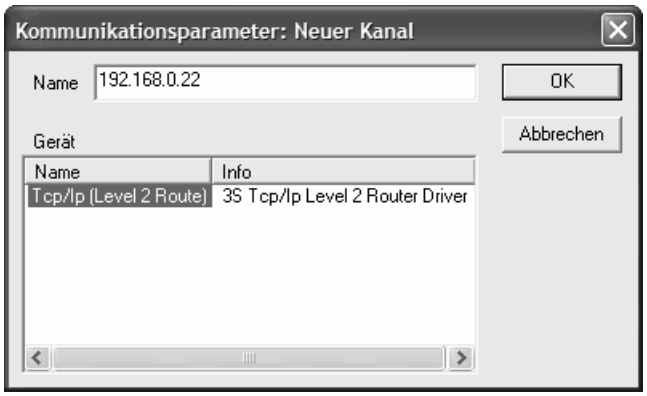
In der SPS-Programmierumgebung wird über den Menüpunkt <Online> <Kommunikationsparameter> der Dialog 'Kommunikationsparameter' erreicht. In diesem Dialog werden die Parameter eingestellt, welche für die Kommunikation zwischen dem Programmier-PC und der Steuerung gelten.

Im Dialog 'Kommunikationsparameter' sind unter "Kanäle" alle bereits erstellten Kommunikationswege aufgelistet. Wählen Sie nun einen der Kanäle aus, indem Sie mit der Maus auf einen Eintrag klicken. Die entsprechenden Parameter werden dann in der Tabelle angezeigt. Für eine Verbindung über Ethernet ist ein Kanal vom Typ Tcp/Ip (Level 2 Route) zu verwenden. Unter Address ist die IP-Adresse der Steuerung aufgeführt mit der durch diesen Kanal kommuniziert wird.



Einrichten eines neuen Kanals

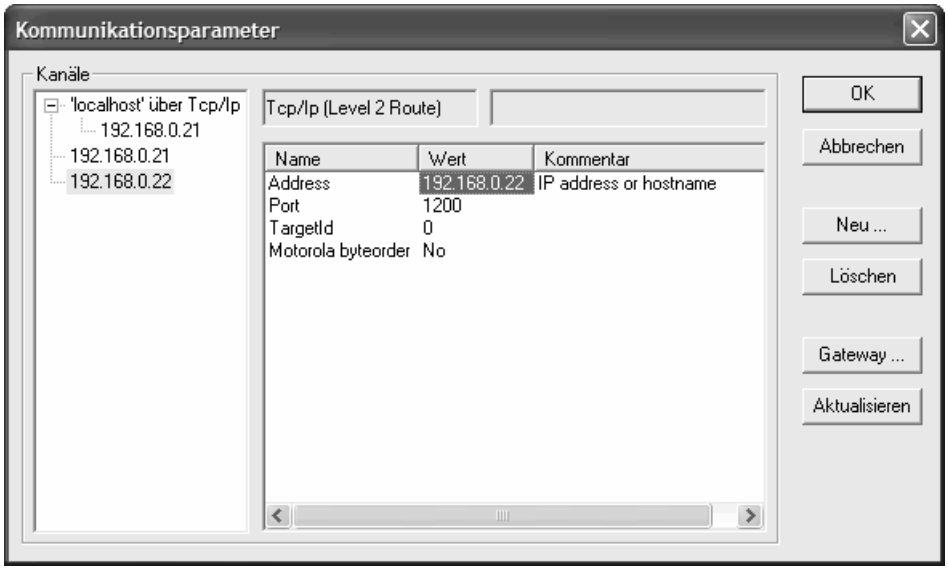
Drücken Sie die Schaltfläche **Neu**. Sie erhalten den Dialog 'Kommunikationsparameter: Neuer Kanal':



Im Eingabefeld **Name** wird automatisch der für den zuletzt eingetragenen Kanal verwendete Name angeboten. Sie können den Kanalnamen hier editieren. Die Kanalnamen sind rein informativ, Eindeutigkeit ist nicht zwingend, aber empfehlenswert.

In der Tabelle unter **Gerät** sind die verfügbaren Gerätetreiber aufgelistet. Aus der Spalte **Name** wählen Sie per Mausklick den Treiber **Tcp/Ip (Level 2 Route)** aus, in der Spalte **Info** steht der dazu vorhandene Kommentar.

Wenn Sie den Dialog 'Neuer Kanal' mit **OK** geschlossen haben, erscheint der neu definierte Kanal im Dialog 'Kommunikationsparameter' als weiterer Eintrag bei **Kanäle**. Er ist hiermit zunächst nur lokal im Projekt gespeichert. In diesem Stadium können Sie die Spalte **Wert** editieren. Geben Sie unter Wert zur Zeile Address die IP-Adresse der Steuerung ein, indem Sie auf die Adresse Doppelklicken, die Adresse editieren und wichtig, die Eingabe mit **Return** abschliessen. Bestätigen Sie dann die eingestellten Parameter mit **OK** und verlassen damit den Dialog 'Kommunikationsparameter'.



Die Parameter eines bereits verwendeten Kanals können im Konfigurationsdialog nicht mehr editiert werden. Die Parameterfelder erscheinen grau. Sie können die Verbindung allerdings löschen, solange sie nicht aktiv ist.

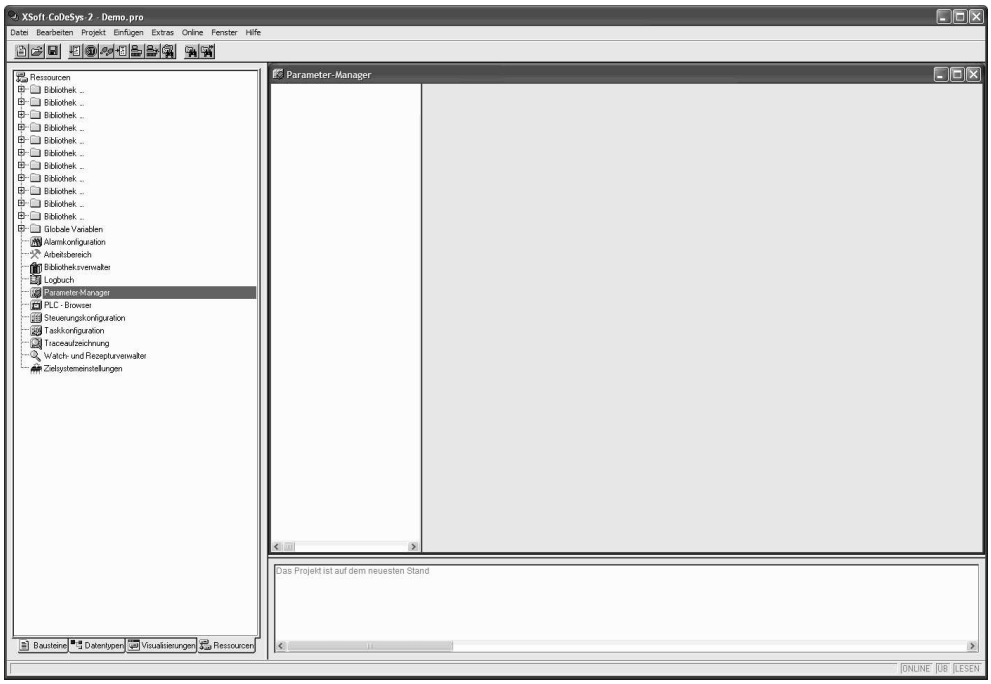
Beachten Sie, dass der Löschvorgang für einen Kanal nicht reversibel ist. Er erfolgt in dem Moment, in dem Sie auf die Schaltfläche **Löschen** drücken!

→ Speichern Sie Ihr SPS-Programm mit den neuen Kommunikationsparametern, übersetzen Sie dies und loggen Sie sich anschliessend auf die Steuerung ein.

→ Achten Sie darauf, dass die IP-Adressen des Programmier-PCs und der Steuerung der gleichen Adressenfamilie angehören. Detaillierte Information entnehmen Sie bitte der Dokumentation "Netzwerk in Kürze".

9 Parameter-Manager / Objektverzeichnis

Bei aktivierter Option erscheint im Registerblatt 'Ressourcen' der Eintrag **Parameter-Manager**. Dieser ermöglicht das Erstellen eines Objektverzeichnisses für Variablen und Parameter, die dem gezielten, aktiven Datenaustausch mit anderen Steuerungen dienen.



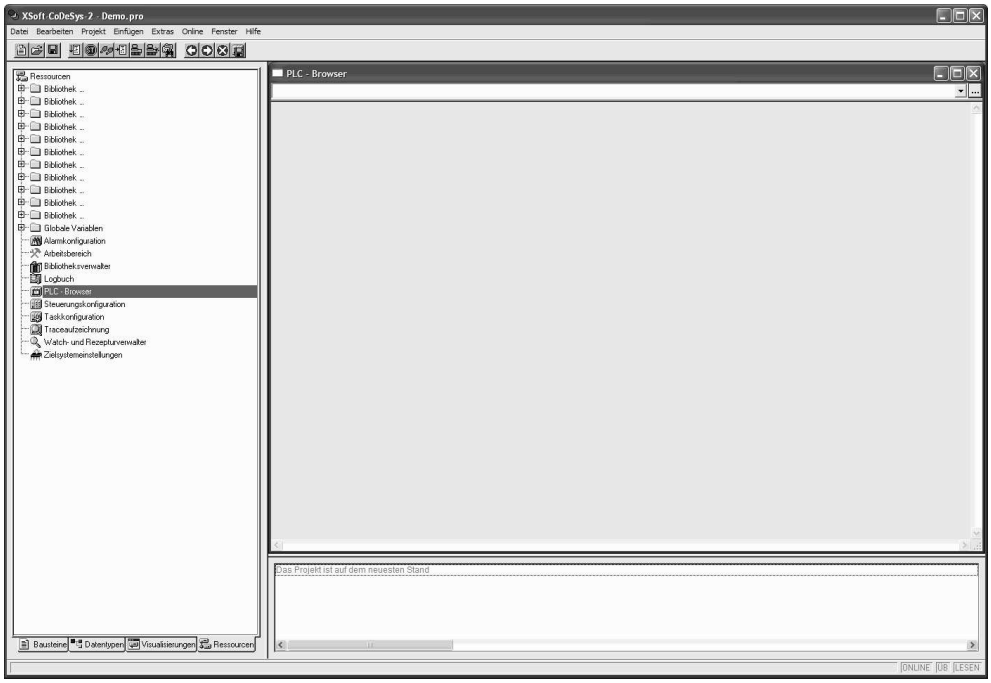
→ Die Objektverzeichnis-Funktionalität wird in Verbindung mit einem in der Steuerungskonfiguration eingefügten CAN-Device unterstützt.

→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3 oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

10 SPS-Browser

Der SPS-Browser ist ein textbasierter Steuerungs-(Terminal) Monitor. Kommandos zur Abfrage bestimmter Informationen aus der Steuerung werden in einer Eingabezeile eingegeben und als String an die Steuerung gesandt. Der Antwortstring wird in einem Ergebnisfenster des SPS-Browsers dargestellt. Diese Funktionalität kann zu Diagnose- und Debugging-Zwecken genutzt werden.

Der SPS-Browser wird über die Funktionalität <PLC-Browser> im Registerblatt 'Ressourcen' erreicht.



Die zur Verfügung stehenden Kommandos gliedern sich in zwei Gruppen:

- Standard Browser-Befehle
- Zielsystemspezifische Browser-Befehle

Diese Befehle werden in einer Datei verwaltet und sind im SPS-Laufzeitsystem implementiert.

Befehl	Beschreibung
?	Liste der implementierten und zur Verfügung stehenden Kommandos holen
...	...

Um die SPS-Browser-Funktionalität zu nutzen, müssen Sie mit dem SPS-Zielsystem online verbunden sein.

11 Alarmkonfiguration

Sämtliche SPS-Zielsysteme mit aktivierter Target-Visualisierung unterstützen die CoDeSys Alarmkonfiguration (➔ Kap. 4.5).

→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3, dem Handbuch zur CoDeSys Visualisierung oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

12 Anbindung an die Visualisierung / Erstellen der Symboldatei

Als Basis für die Kommunikation zwischen Steuerung und einer allfällig verwendeten Visualisierung dient die Symboldatei. Der Inhalt dieser Symboldatei wird in der SPS-Programmierumgebung konfiguriert. Beim Kompilieren wird diese Symboldatei erstellt und beim anschliessenden Programmdownload auf die Steuerung geladen.

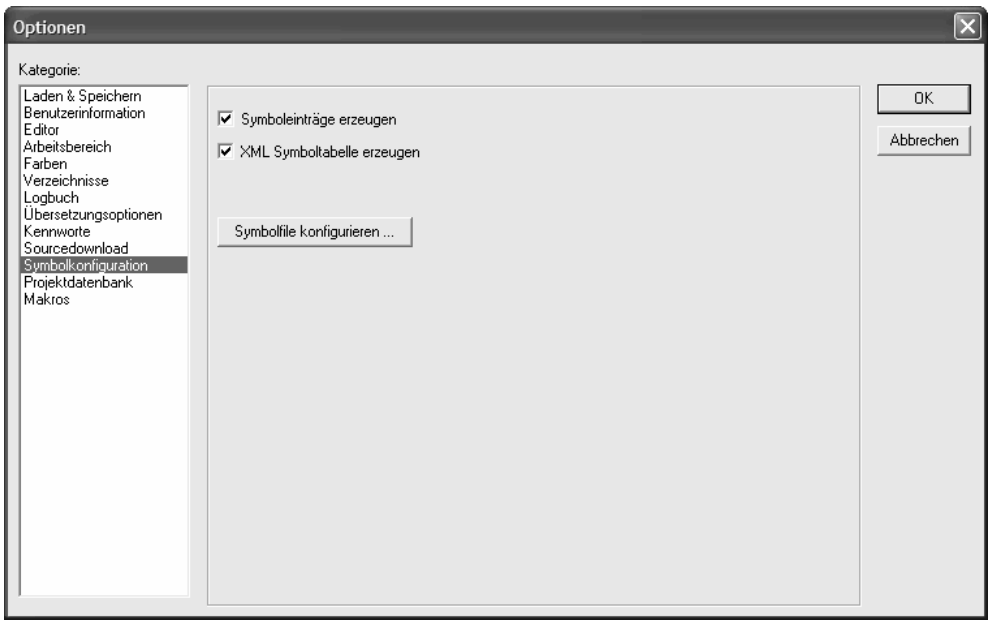
12.1 Symboldatei konfigurieren

Der Inhalt der Symboldatei wird in der Symbolkonfiguration konfiguriert. Die Symbolkonfiguration wird über den Menüpunkt <Projekt> <Optionen> erreicht.

→ Ist im Menüpunkt 'Online' die Simulation aktiviert, erscheint im Dialog Option der Eintrag Symbolkonfiguration nicht.

Damit die Symboldatei beim Kompilieren erzeugt wird, muss die Option **Symboleinträge erzeugen** aktiviert werden.

Teilweise unterstützen Visualisierungen den Import der Symboldatei im XML-Format. Damit diese Symboldatei beim Kompilieren erzeugt wird, muss die Option **XML Symboltabelle erzeugen** aktiviert werden.

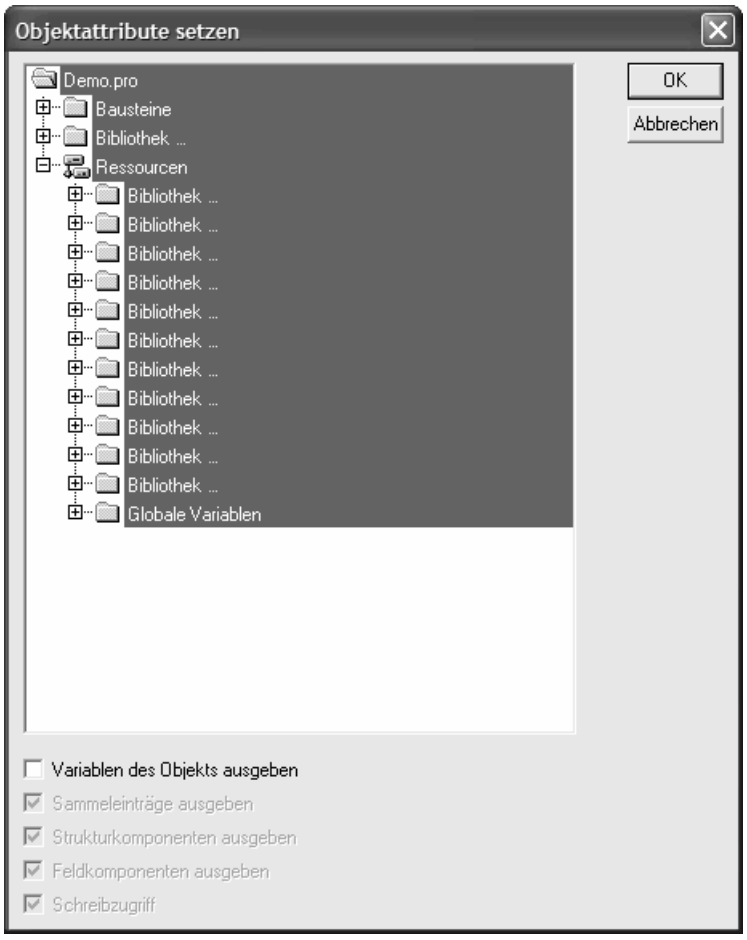


Über die Schaltfläche **Symbolfile konfigurieren** wird definiert, aus welchen Objekten die Variablen in die Symboldatei exportiert werden.

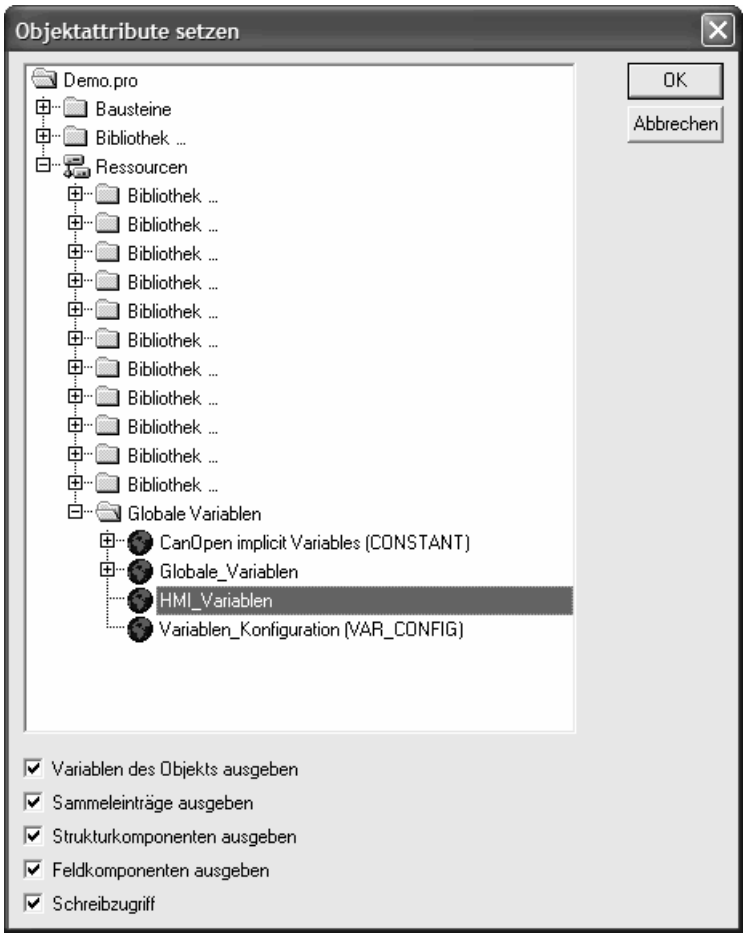
→ Es wird empfohlen, nur die für den Variablenaustausch mit der Visualisierung benötigten Variablen in die Symboldatei zu exportieren. Daher sollte der Bereich dieser Variablen strukturiert werden. Dies erfolgt über das Definieren von globalen Variablenarbeitsblätter oder über das Verwenden von Pragma-Anweisungen.

→ Detaillierte Informationen sind dem Handbuch CoDeSys V2.3 oder der Online-Hilfe der SPS-Programmierumgebung zu entnehmen.

Standardmässig sind sämtliche Objekte selektiert. Deaktivieren Sie deshalb beim erstmaligen Öffnen dieses Dialoges bei sämtlichen Objekten die Option 'Variablen des Objekts ausgeben'.

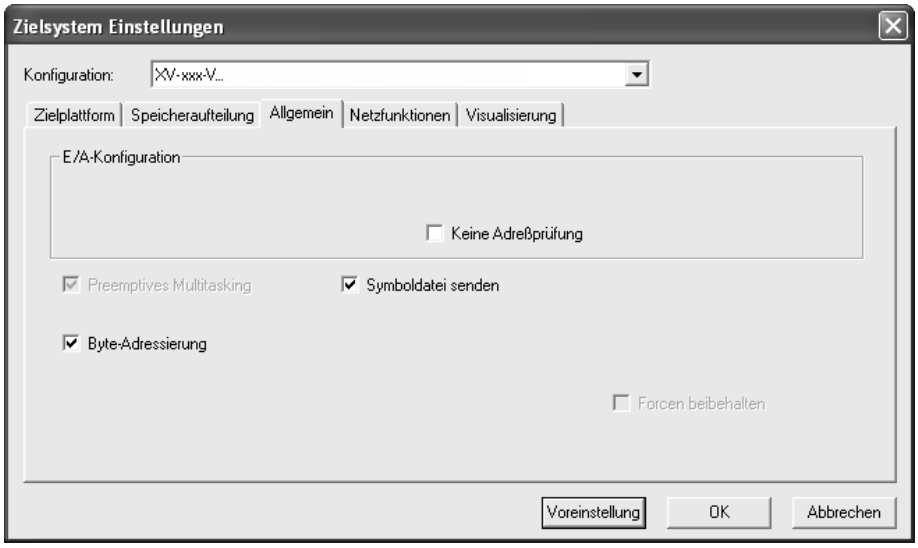


Aktivieren Sie nun selektiv die Option 'Variablen des Objekts ausgeben' zu den gewünschten Objekten.



12.2 Symboldatei senden

Damit die Symboldatei beim Projektdownload auf die Steuerung geladen wird, muss in der Zielsystemeinstellung im Registerblatt 'Allgemein' die Option **Symboldatei senden** aktiviert werden.



13 Zielsysteminstallation und Firmware Update

Die Zielsysteminstallation bzw. ein Firmware Update wird mit Hilfe des Programmaufrufs "TargetFirmwareWinCE" ausgeführt und erfolgt in der SPS-Programmierumgebung über die Funktionalität <Steuerungskonfiguration> im Registerblatt 'Weitere Parameter'.

Betätigen Sie hierfür im Feld "Aktualisieren des Betriebssystems" die Schaltfläche **Start** und wählen Sie anschliessend entsprechende Firmware.

Es stehen folgende Installationsvarianten zur Verfügung:

FTP Installation: Die Installation der Verzeichnisse **PlcRts** und **PlcPrg** erfolgt über FTP.

Installation auf einen Wechseldatenträger: Der Wechseldatenträger (z.B. CompactFlash™) muss am Programmier-PC über einen entsprechenden Adapter (z.B. PC-Card-Adapter) verfügbar sein. Die Installation der Verzeichnisse **PlcRts** und **PlcPrg** erfolgt nun direkt auf den Wechseldatenträger

Installation in ein lokales Verzeichnis: Die Installation der Verzeichnisse **PlcRts** und **PlcPrg** erfolgt auf ein lokales Verzeichnis des Programmier-PCs. Anschliessend müssen die gesamten Verzeichnisse **PlcRts** und **PlcPrg** manuell ins Hauptverzeichnis des Wechseldatenträgers (z.B. CompactFlashs™) kopiert werden.

→ Das gespeicherte SPS-Programm bleibt bei einem Firmware Update erhalten.

→ Die Verzeichnisse PlcRts und PlcPrg werden bei einer Zielsysteminstallation bzw. einem Firmware Update nicht gelöscht. Vorhandene Dateien werden lediglich überschrieben.

14 Lizenzierung

14.1 SPS-Programmierumgebung

Die SPS-Programmierumgebung ist lizenzpflichtig.

→ Ist bei der Installation der SPS-Programmierumgebung keine Serienummer bzw. Lizenz vorhanden, werden die SPS-Zielsysteme im Demomode installiert.

14.2 SPS-Laufzeitsystem

Das SPS-Runtime-System ist lizenzpflichtig und benötigt auf dem SPS-Zielsystem **100 Lizenzpunkte**.

→ Sind beim Start des SPS-Laufzeitsystems nicht genügend Lizenzpunkte vorhanden, beginnt keine Abarbeitung des SPS-Programms. Das SPS-Programm startet im Betriebszustand "Stop".

Die Abarbeitung des SPS-Programms kann anschliessend in der SPS-Programmierumgebung über den Menüpunkt <Online> <Start> erfolgen.

14.3 Target-Visualisierung

Die Target-Visualisierung ist lizenzpflichtig und benötigt auf dem SPS-Zielsystem jeweils **100 Lizenzpunkte**.

→ Sind beim Start des SPS-Laufzeitsystems nicht genügend Lizenzpunkte vorhanden, beginnt keine Abarbeitung des SPS-Programms. Das SPS-Programm startet im Betriebszustand "Stop".

Die Abarbeitung des SPS-Programms kann anschliessend in der SPS-Programmierumgebung über den Menüpunkt <Online> <Start> erfolgen.

14.4 Web-Visualisierung

Die Web-Visualisierung ist nicht lizenzpflichtig und benötigt auf dem SPS-Zielsystem **keine Lizenzpunkte**.

15 Änderungsindex

Index	Datum / Visum	Änderungen
01	21.12.10 / DL	Initialversion
02	23.05.11 / DL	Anpassungen und Korrekturen
03	18.05.12 / DL	Anpassungen und Korrekturen infolge XSoft-CoDeSys V2.3.9 SP3, Hinweis zum korrekten Wert der HSA bei der Profibus Master Konfiguration hinzugefügt, Hinweis zum Symbolimport im XML-Format hinzugefügt