

# Zusatzprojekt Automatisierungstechnik

## Einfache Positionierung

### Mit B&R



**Projektteam:**

Maurice Baczynski  
Fabian Kamp  
Rouven Kolze  
Andreas Leer

FSEA14

## **Kurzdarstellung**

Dieses Unterrichtsprojekt befasst sich mit dem Erstellen einer Motion Anwendung welche eine Einfach Positionierung darstellt. Mit jeweiligem Betätigen des Taster 1 dreht der Motor um 90 Grad nach links, mit dem Betätigen des Taster 2 fährt der Motor rechts herum auf die Position 270 Grad und mit dem Taster 3 bewegt sich der Motor auf die Ausgangsposition.

Lernziele des Unterrichtsprojektes:

- Erstellen eines neuen Projektes
- Einfügen der Hardware manuell
- Parametrieren des Servomotors (Acopos) und Motor sowie Kurzinbetriebnahme des Reglers
- Konfigurieren der Kommunikationsparameter und Herstellung der Verbindung zwischen Programmiergerät und PLC
- Erstellen eines Programms in Strukturiertem Text das eine Einfachpositionierung ausführt

Voraussetzung:

Dieses Projekt wird mit der Automation Studio 4.2.2.134 Version erstellt.

Es wird ein Kartenlesegerät benötigt mit Flash-Kartenslot.

B&R Übungsrecks. ( Bilder im Anhang unter Kapitel 14)

## Inhaltsverzeichnis

<b>Kurzdarstellung</b> .....	0
<b>1. Erstellen eines neuen Projekts in Automation Studio</b> .....	2
<b>2. Hardwarekonfiguration</b> .....	4
2.1 CPU.....	4
2.2 Interface und der Baugruppen .....	6
2.3 Acopos Servoumrichter und Motor.....	8
<b>3. Motor manuell hinzufügen</b> .....	19
3.1 Download Motordaten .....	19
3.2 Motoren in Motion einfügen .....	20
3.3 Motoren mit dem Acopos verbinden .....	24
<b>4. Motor Variable und Endat Verbindung</b> .....	25
<b>5. Einrichten der Ethernet Verbindung</b> .....	26
5.1 Ethernet Konfiguration.....	26
5.2 Knotennummer des Acopos .....	28
<b>6. Compact-Flash erstellen</b> .....	29
<b>7. Online verbinden</b> .....	31
<b>8. Parametrieren und Testen des Motors</b> .....	33
8.1 Auswahl der Achse für die Parametrierung .....	33
8.2 Geber Einheiten pro Motorumdrehungen.....	35
8.3 Motorparameter abspeichern .....	36
8.4 Projekt via Netzwerk übertragen.....	37
8.5 Referenzieren des Motors .....	39
8.6 Tuning des Antriebes.....	42
8.7 Basis Bewegung.....	45
<b>9. Automatische Variablen Deklaration</b> .....	47
<b>10. Programmierung</b> .....	48
10.1 Programm Anlegen.....	48
10.2 Programm Initialisierung .....	51
10.3 Programm Ablauf .....	52
10.4 Programmtest über die Watch Funktion.....	57
<b>11. Externe Schalter dem Programm zuweisen</b> .....	59
<b>12. Abbildungsverzeichnis</b> .....	63
<b>13. Komponenten</b> .....	66
<b>14. Aufbau der Racks</b> .....	67

## 1. Erstellen eines neuen Projekts in Automation Studio

Um ein neues Projekt zu erstellen starten Sie Automation Studio. Sobald die Oberfläche des Automation Studios geöffnet wurde klicken Sie auf „Neues Projekt“.



Abbildung 1 Automation Studio Oberfläche

Nun öffnet sich ein Fenster in dem Sie „Ein leeres Projekt“ auswählen um in die Grundkonfigurationen zu gelangen.



Abbildung 2 Leeres Projekt

Der Projektassistent wird geöffnet und wie folgt konfiguriert.

Neues Projekt

**Automation Studio - Projektassistent**

Auf dieser Seite können die Basisparameter für das neue Projekt eingegeben werden.

Name des Projektes:

Pfad des Projektes:

C:\projects

Hinweis: Ein Unterordner mit dem Projektnamen wird automatisch erstellt.

Automation Runtime Dateien ins Projekt kopieren

Automation Runtime Simulation verwenden

Beschreibung des Projektes:

Weiter > Abbrechen Hilfe

Name des Projektes.

Speicherort des

Auswahl, wenn alle Runtime- und Updatedateien ins Projekt kopiert werden sollen.

Beschreibung des Projektes mit Auftrag oder Kenndaten des Kunden.

Abbildung 3 Projektassistent

In unserem Projekt wird die Simulation nicht benötigt, da wir direkt mit der Hardware arbeiten und daher werden nur die Runtime Dateien ins Projekt kopiert. Um dies zu bestätigen auf „Weiter“ klicken.

## 2. Hardwarekonfiguration

### 2.1 CPU

Die Grundkonfiguration der Hardware kann beibehalten werden, da Sie die Hardwarekonfiguration manuell einbinden. Optional gibt es die Möglichkeit die Hardware Online zu identifizieren oder eine bestehende Hardware zu referenzieren. Alternativ kann der Name der Konfiguration geändert werden.

Neues Projekt

**Automation Studio - Projektassistent**

Auf dieser Seite können die Parameter für die neue Konfiguration eingegeben werden.

Name der Konfiguration:  
Config1

Hardwarekonfiguration

- Eine neue Hardwarekonfiguration manuell definieren
- Die Hardwarekonfiguration Online identifizieren
- Eine bestehende Hardwarekonfiguration ( \*.hw ) referenzieren.

Beschreibung der Konfiguration:

< Zurück   Weiter >   Abbrechen   Hilfe

Abbildung 4 Projektassistent Hardware Auswahl

Um die Eingaben zu bestätigen auf „Weiter“ klicken.

Nun muss die aktuelle CPU manuell in das „Suchen...“ Fenster eingegeben werden. Wenn die Produktnummer in das „Suche..“ Fenster eingegeben wurde erscheint im unteren Teil des Fensters die CPU die verwendet wird. Diese CPU muss mit einem Mausklick ausgewählt werden und um diese zu übernehmen muss abschließend auf „ Fertig stellen“ bestätigt werden.

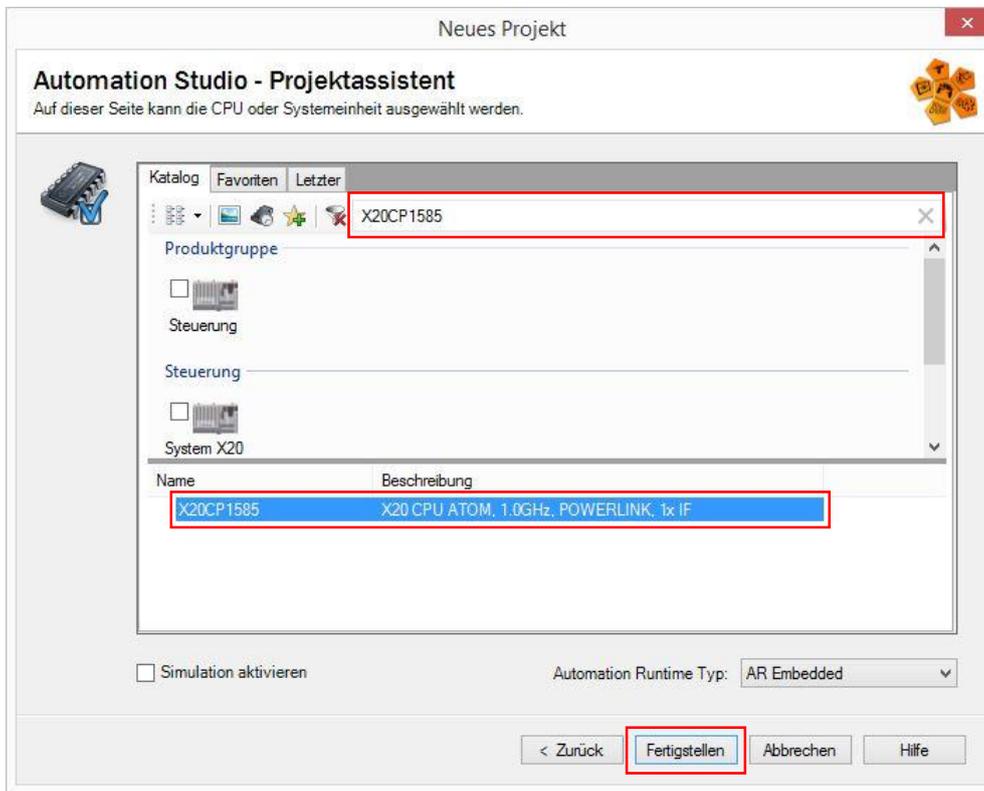


Abbildung 5 Einbinden der CPU

Nun öffnet sich der System Designer mit der ausgewählten CPU.

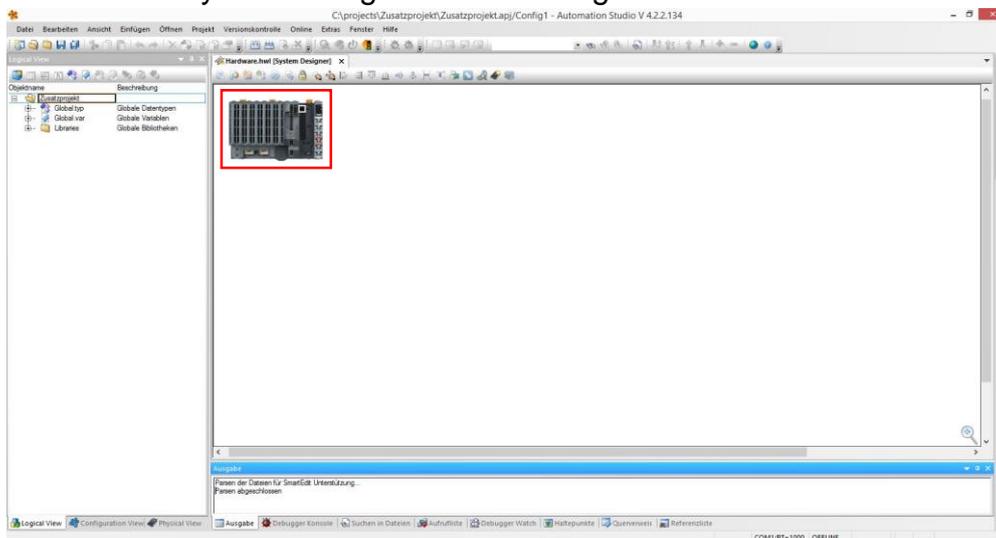


Abbildung 6 System Designer

## 2.2 Interface und der Baugruppen

Wählen Sie den „Toolbox – Hardware Katalog“ in der rechten oberen Ecke aus. Es öffnet sich das zugehörige Fenster in dem alle B&R Hardware Komponenten zu finden sind.

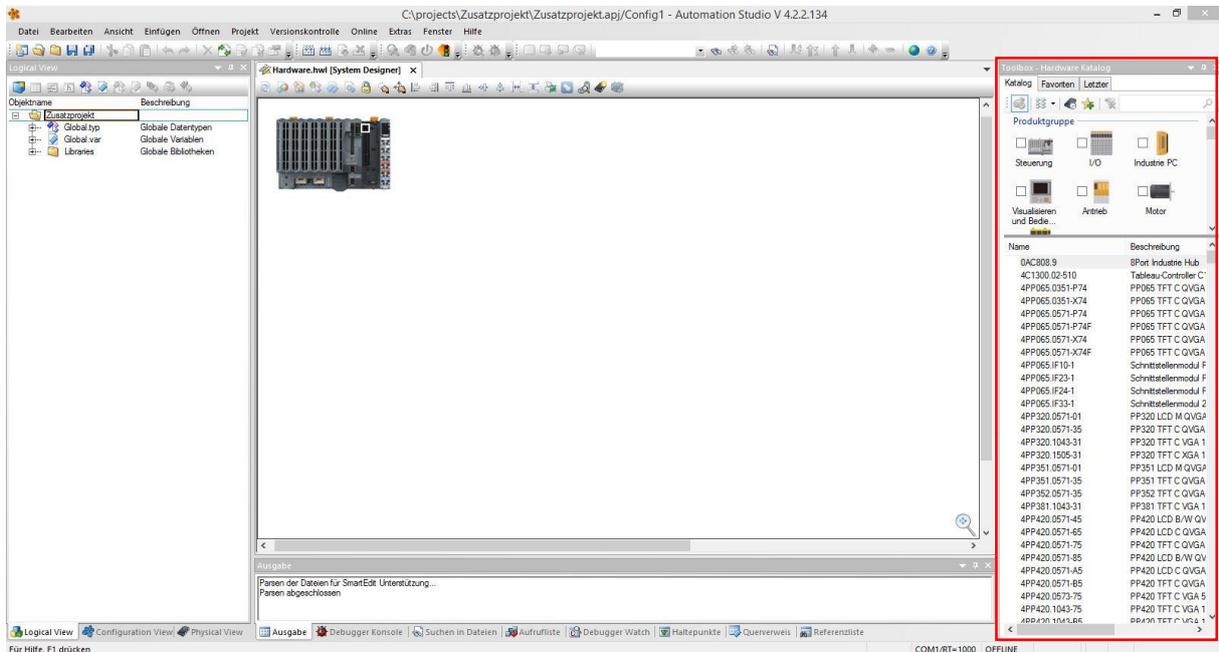


Abbildung 7 Toolbox-Hardware Katalog

Wählen Sie die markierte Fläche auf der CPU aus um in dem „Suche“ Fenster der Toolbox den Profinet RT Controller zu suchen. Nach Eingabe der Bestellnummer (X20IF10E1-1) finden Sie das Interface des RT Controllers. Das Interface wird über Doppelklick auf die CPU übertragen.

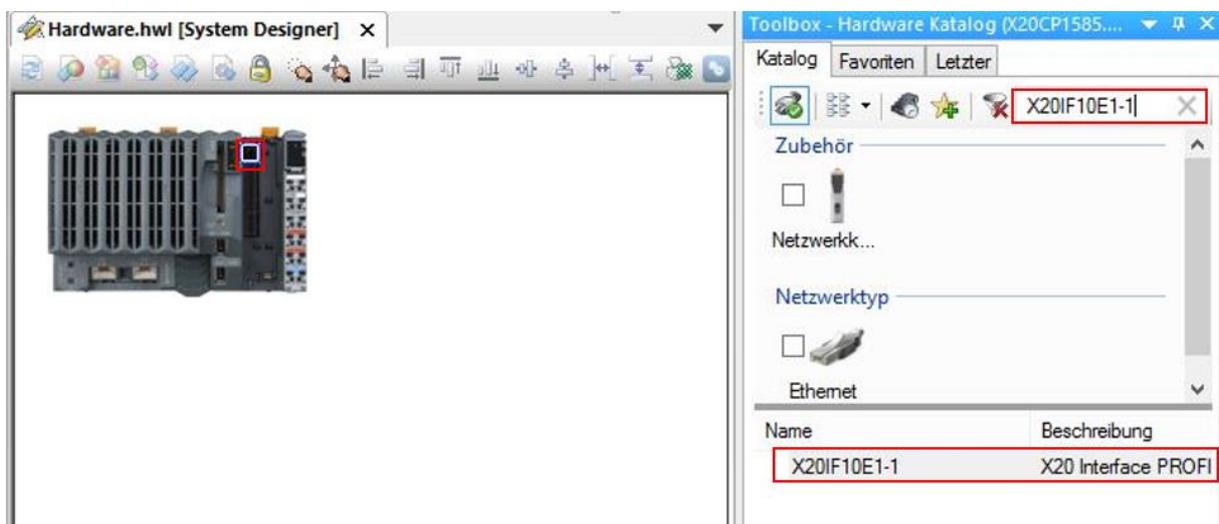


Abbildung 8 Einbindung Interface RT Controller

Durch einfaches Klicken auf die CPU, wird diese markiert um weiter Baugruppen hinzuzufügen. Über das „Suche“ Fenster werden folgende Baugruppen gesucht und über Doppelklick an die CPU hinzugefügt. *Wichtig ist das die folgende Reihenfolge der Baugruppen eingehalten wird.*

### Hardware:

Komponenten der SPS:

2x X20DI8371

2x X20DO8322

1x X20DS1319

1x X20AI4622

1x X20AO2632 ( $\pm 10$  V / 0 bis 20 mA)

1x X20AT2222

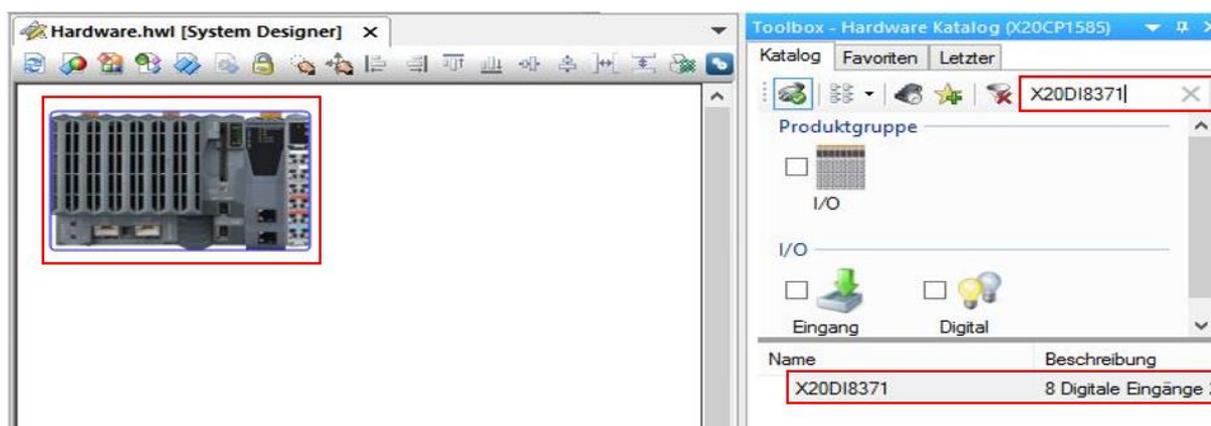


Abbildung 9 Einbinden der Baugruppen

Die SPS mit ihren E/A Baugruppen sieht wie folgt aus. Wenn dies nicht übereinstimmt, überprüfen Sie die richtige Anzahl und Reihenfolge der Baugruppen.

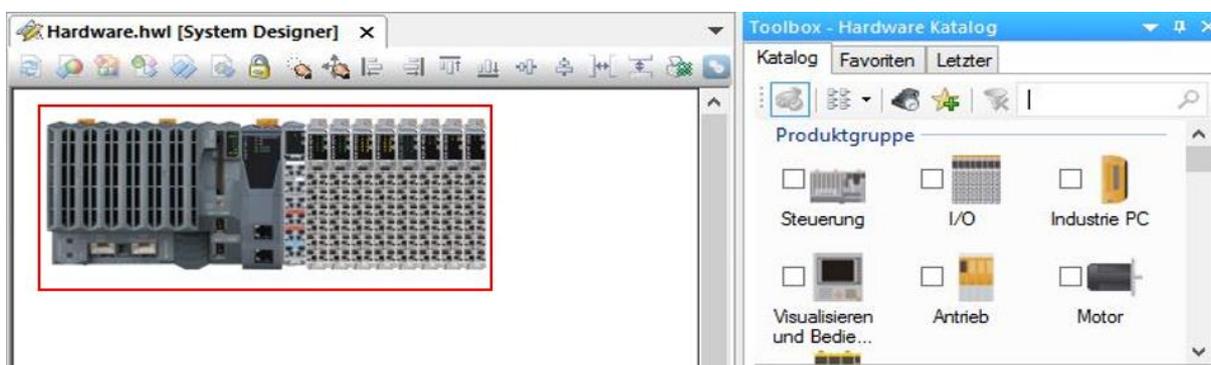


Abbildung 10 Fertige Hardware

### 2.3 Acopos Servoumrichter und Motor

Nachdem die Hardware für die Steuerung eingebunden wurde folgt als nächster Schritt die Einbindung des Acopos Servoumrichter.

Um den Acopos 1010 einzubinden markiert man die IF3/POWERLINK Schnittstelle und gibt die Bestellnummer (8v1010.50-2) in das „Suche“ Fenster der Toolbox ein. Durch Doppelklick auf den Namen wird der Acopos ausgewählt.

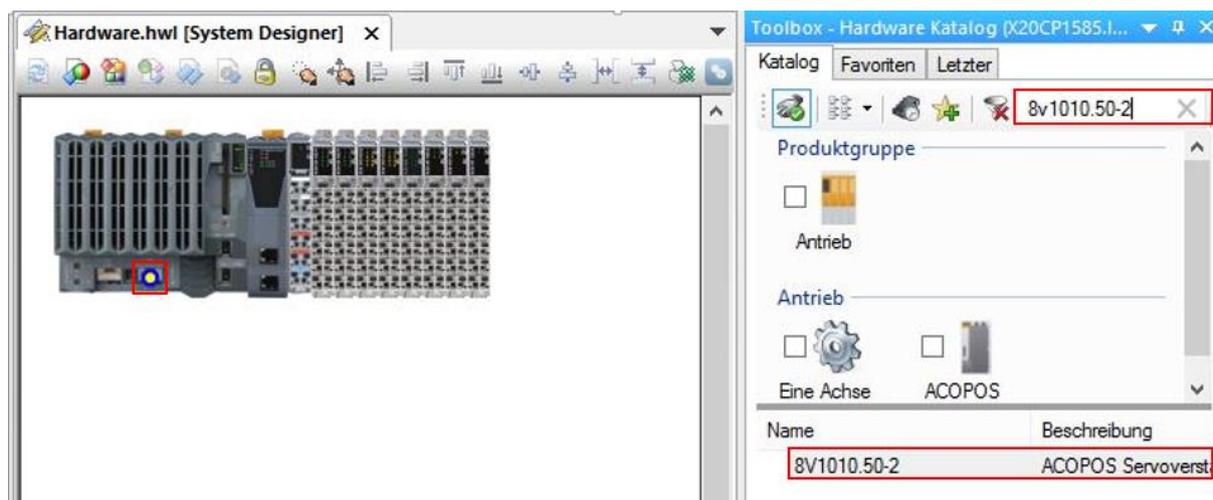


Abbildung 11 Acopos Auswahl

Nun öffnet sich das Fenster zur Antriebskonfiguration. Die SS1 (Schnittstelle 1) wurde automatisch als 8AC114.60-1 Schnittstelle erkannt und vergeben. Nun wird die SS2 konfiguriert.

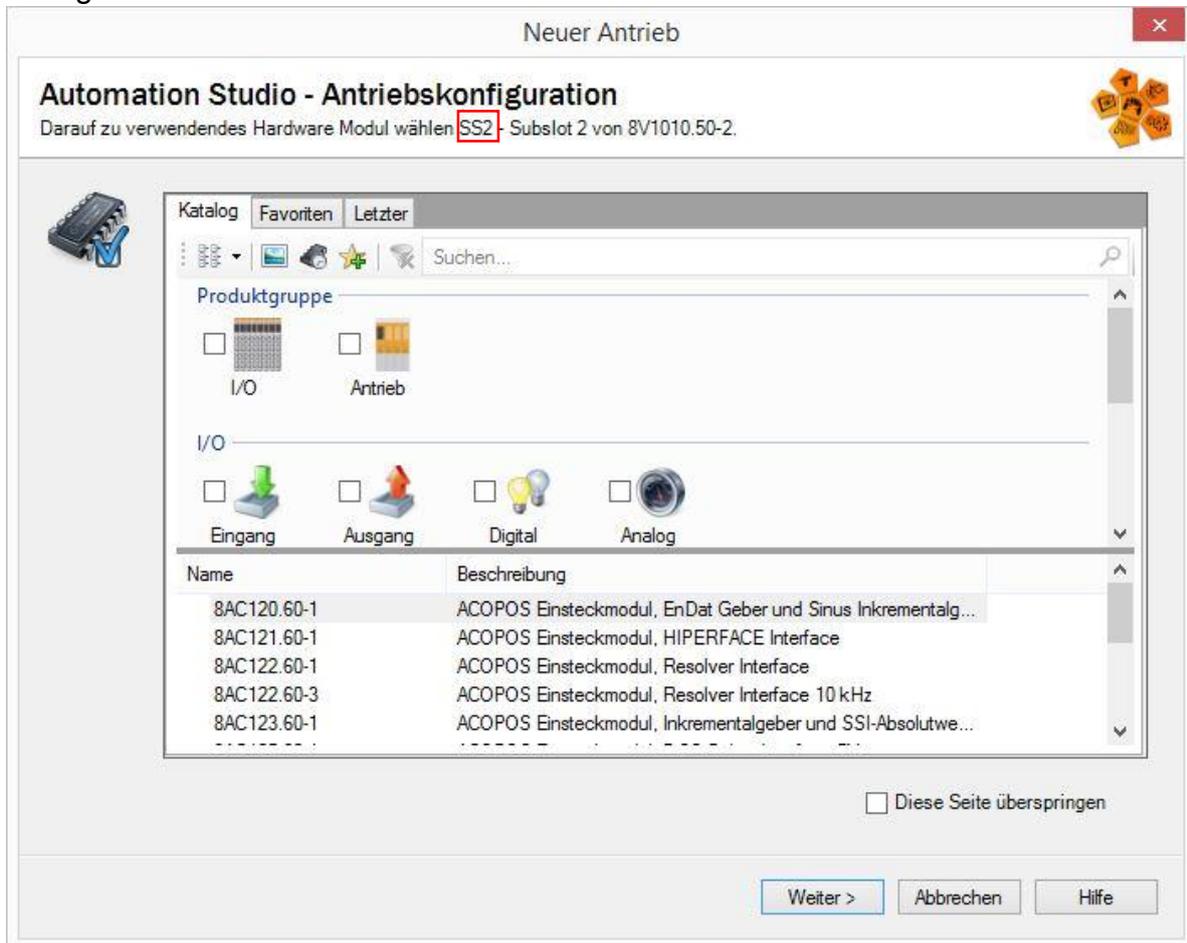


Abbildung 12 Acopos Konfiguration

Jetzt wird die SS2 (Schnittstelle 2) Konfiguriert. In unserem Fall ist es ein Einsteckmodul, das als EnDat 2.2 Geber Interface mit Geberüberwachung fungiert. Geben Sie in das „Suche“ Fenster die Bestellnummer AC126 ein und wählen Sie das Modul im unteren Bereich aus. Um dies zu bestätigen klicken Sie auf Weiter.

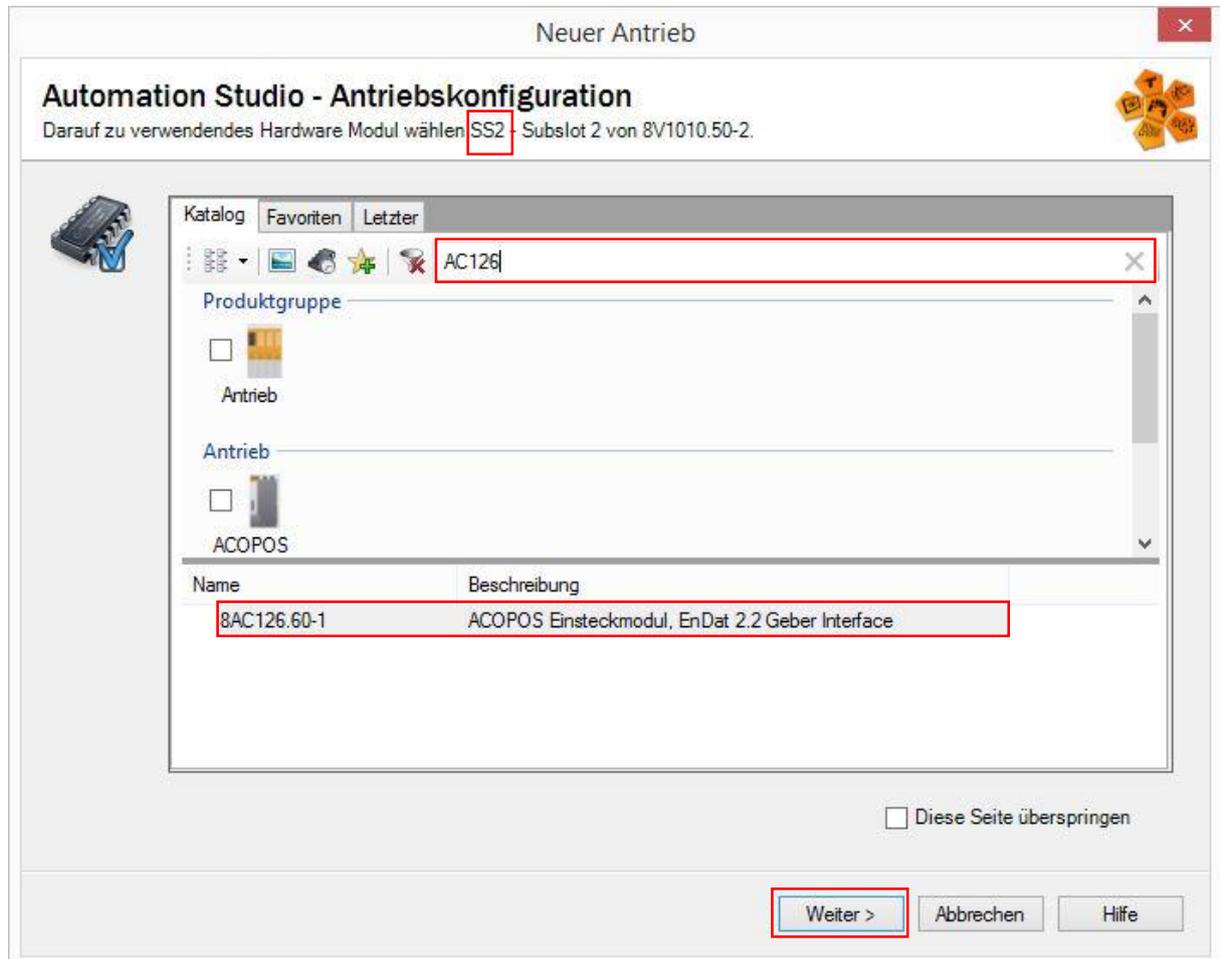


Abbildung 13 Auswahl des Einsteckmoduls

Als nächstes wird gefragt welches Modul sich im SS3 befindet. In diesem Fall ist kein weiteres Modul verwendet, weshalb „Diese Seite überspringen“ markiert und mit Weiter bestätigt wird.

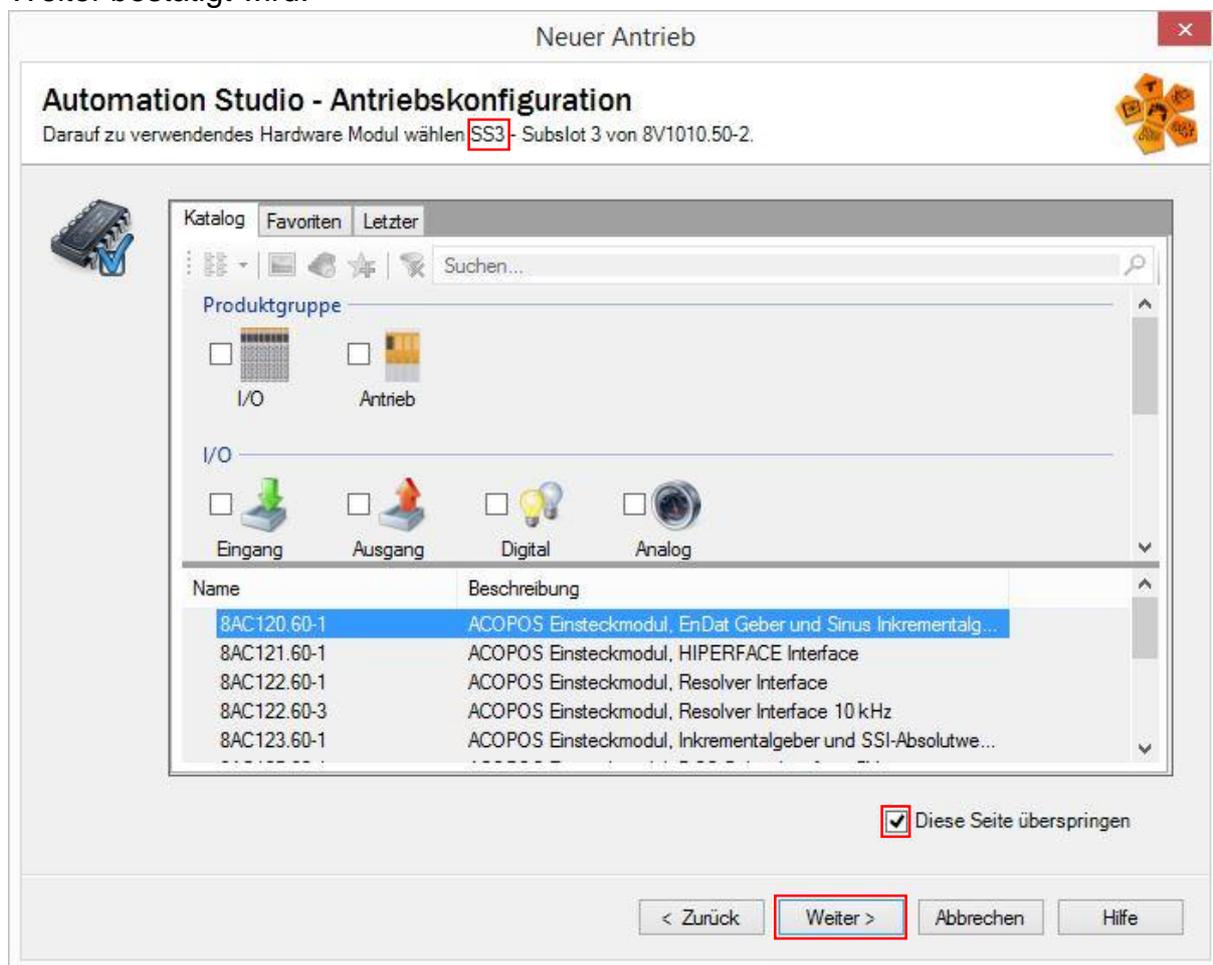


Abbildung 14 SS3

Nachdem der Acopos ausgewählt und konfiguriert wurde, wird als nächstes der Motor eingebunden.

Nun soll das zu verwendende Hardware Modul für den MT1 (Motorkonnektor 1) ausgewählt werden. An dem Reck ist ein Synchron-Motor von B&R mit 4 Polpaarzahlen, 1500 1/min und 0,68 Nm Stillstandsmoment an den Acopos angeschlossen. In das „Suche“ Fenster wird die Bestellnummer des Motors (8LVA22.B1015D000-0) eingegeben und mit weiter bestätigt. Wurde die Bestellnummer des Motors in der Antriebskonfiguration nicht gefunden, kann dieser nach der Antriebskonfiguration nachträglich manuell hinzugefügt werden. Ist der Motor in der Konfiguration auffindbar wird mit „Weiter“ bestätigt. Falls der Motor nicht auffindbar ist wird das Kästchen „Diese Seite überspringen“ markiert und mit „Weiter“ bestätigen.

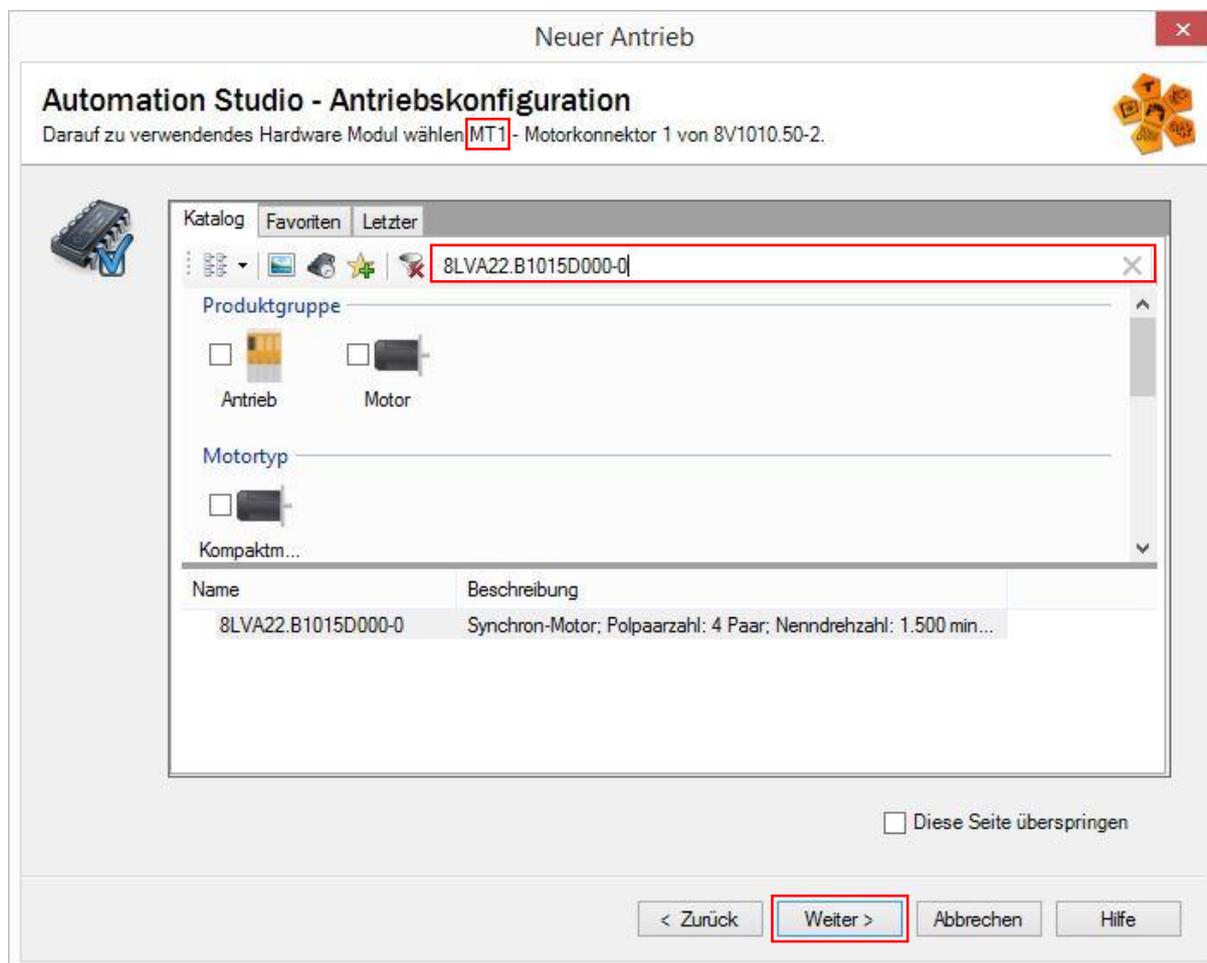


Abbildung 15 Motor Konfiguration

Hier wird die Motion System Wahl vorgegeben. In diesem Fall ist es die „Acp10/Arnc0“ ausgewählt und muss mit Weiter bestätigt werden.

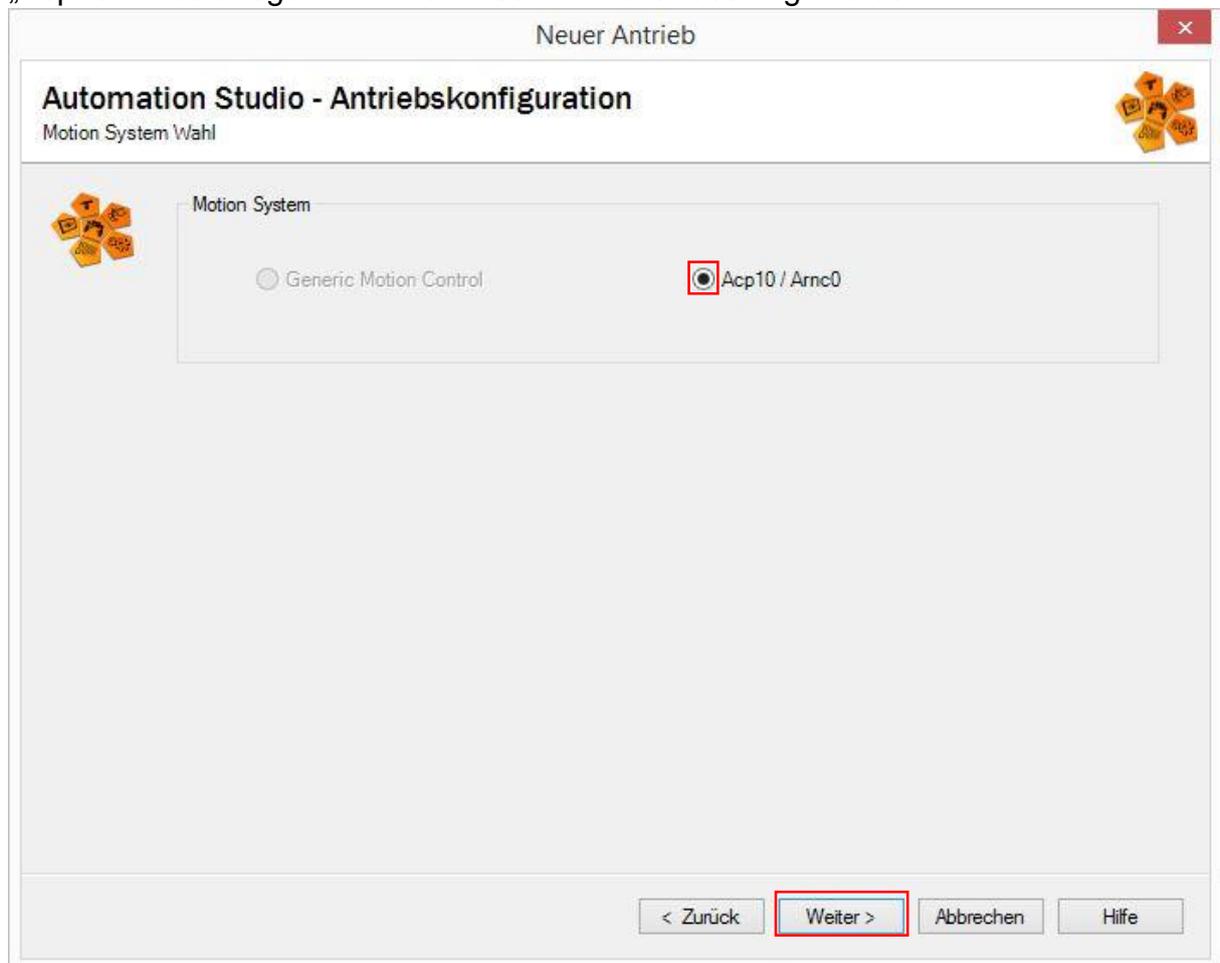


Abbildung 16 Motor System Wahl

Im nächsten Fenster wird die Sprache der Fehlermeldungen ausgewählt. Ist die gewünschte Sprache eingestellt wird mit Weiter bestätigen.

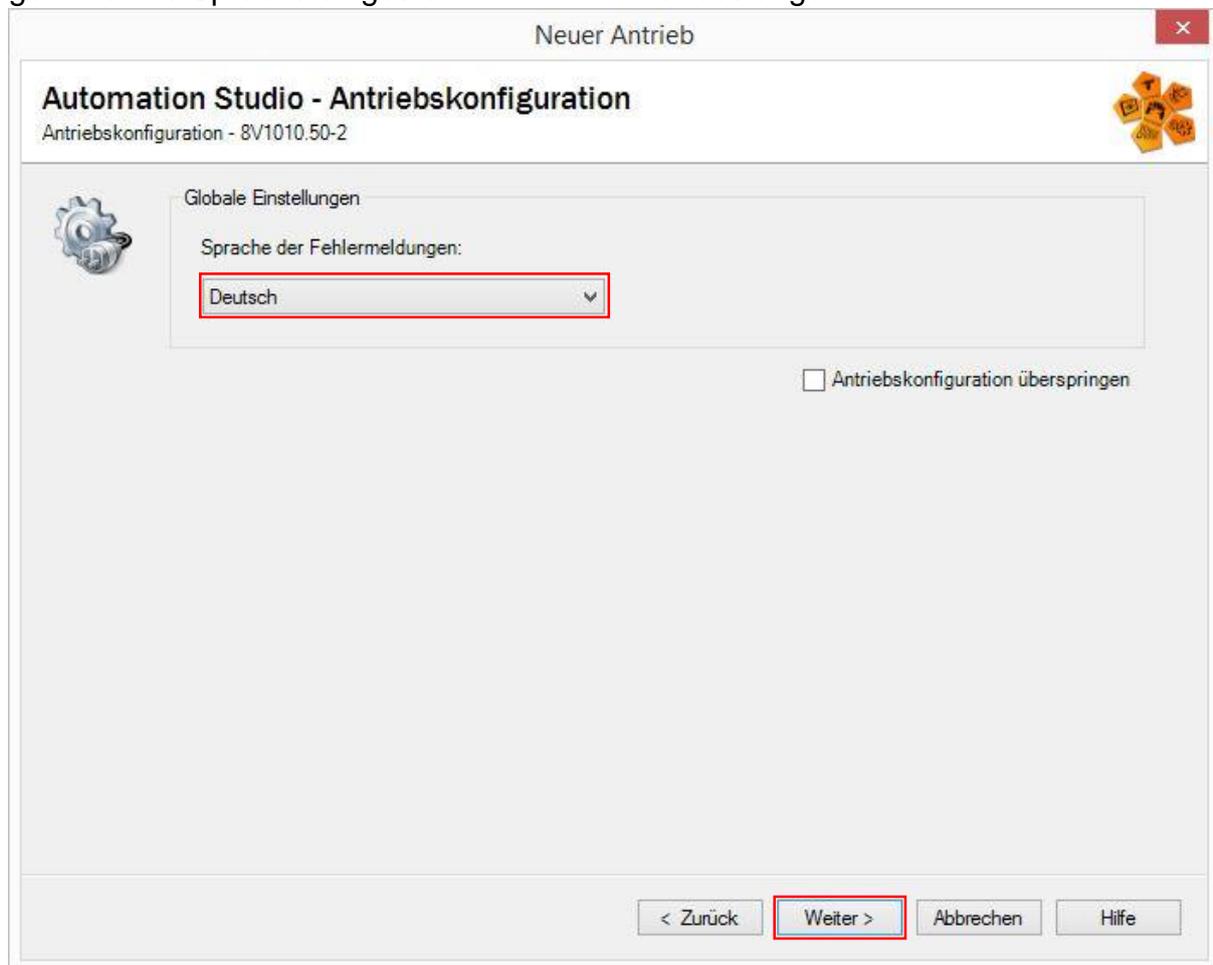


Abbildung 17 Sprache Fehlermeldungen

Beim nächsten Fenster wird die Achse wie folgt konfiguriert und mit Weiter bestätigt.

The screenshot shows a software window titled 'Neuer Antrieb' with a sub-header 'Automation Studio - Antriebskonfiguration' and 'Reelle Achse1'. The 'Allgemein' section contains a text field for 'Name' with the value 'gAxis01', a 'Verwendung' section with a checked checkbox for 'PLCopen verwenden' and an unchecked checkbox for 'CNC', and an 'Einstellung' section with an unchecked checkbox for 'Achse ist periodisch'. Below this is a 'Standard Einstellungen überschreiben' section with three checkboxes: 'Motor Simulation aktivieren' (unchecked), 'Beide Endschalter sind normalerweise geöffnet' (checked), and 'Quick stop ist normalerweise offen' (checked). At the bottom right of this section is a checkbox for 'Diese Seite überspringen' (unchecked). At the bottom of the window are four buttons: '< Zurück', 'Weiter >', 'Abbrechen', and 'Hilfe'. The 'Weiter >' button is highlighted with a red rectangular box.

Abbildung 18 Achse Konfigurieren

PLCopen verwenden:

Wenn diese Checkbox angehakt ist, dann wird, falls noch nicht im Projekt vorhanden, die Bibliothek "acp10\_mc" ins Projekt eingefügt damit die Bedienung dieser Achse mit PLCopen-Funktionsblöcken möglich ist.

Beide Endschalter sind normalerweise geöffnet:

Wenn angewählt: Beide Endschalter werden auf "normal open" konfiguriert. D.h. erst wenn an den Eingängen Spannung angelegt wird, wird der Endschalter ausgelöst. Diese Einstellung ist beispielsweise notwendig wenn keine physikalischen Schalter vorhanden sind.

Durch diese Einstellung werden die Endschalter nicht mehr auf Drahtbruch überwacht und lösen erst mit Spannung aus!

Quick stop ist normalerweise offen:

Wenn angewählt: Notstop Eingang wird auf "normal open" konfiguriert. D.h. erst wenn am Eingang Spannung angelegt wird, wird der Notstop ausgelöst. Diese Einstellung ist beispielsweise notwendig wenn kein physikalischer Schalter vorhanden ist. Durch diese Einstellung wird der Notstop nicht mehr auf Drahtbruch überwacht und löst erst mit Spannung aus!

In dem darauf folgenden Fenster wird nach weiteren Achsen gefragt. Da wir keine weiteren Achsen verwenden, wird „Diese Seite überspringen“ markiert und mit Weiter bestätigt.

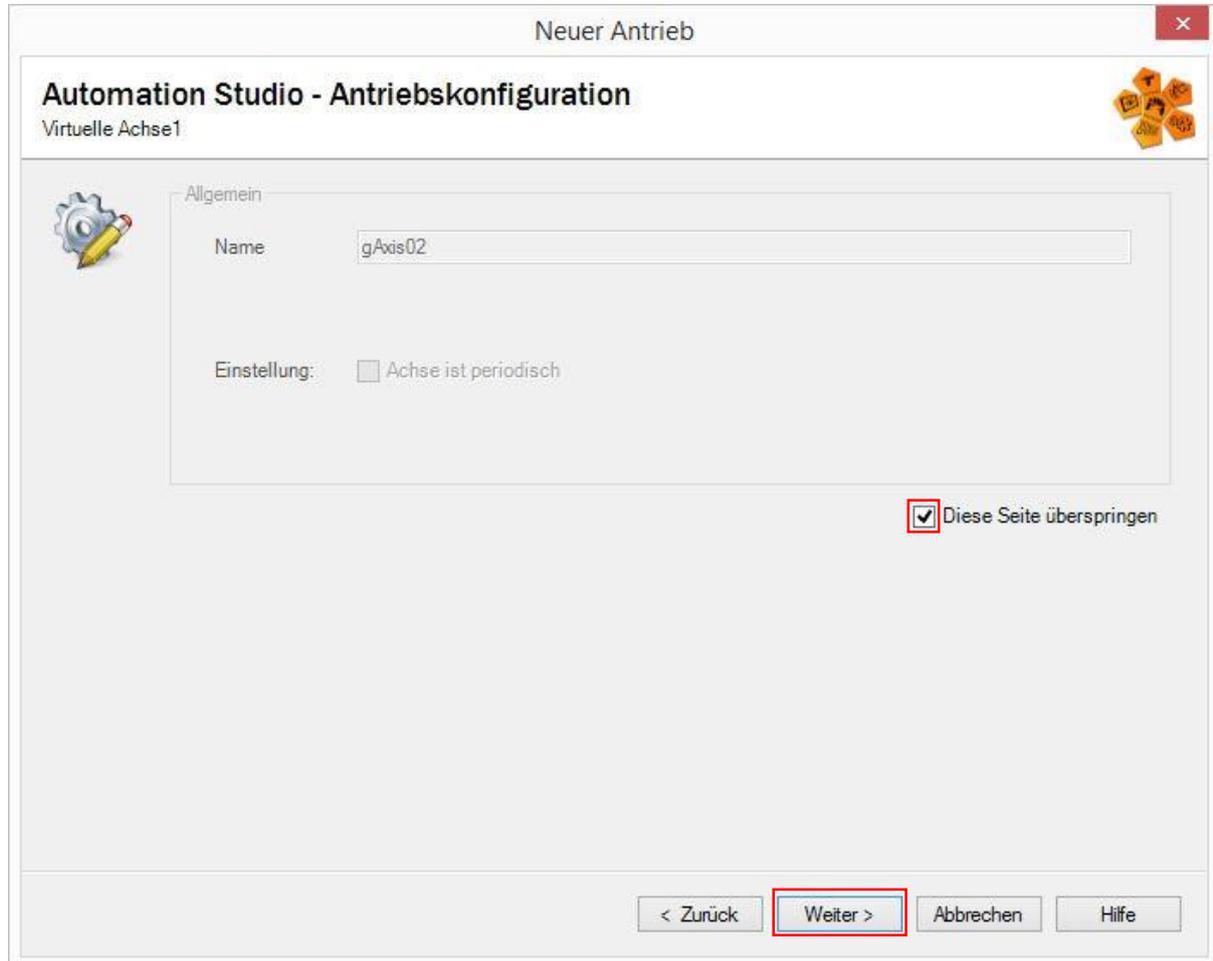


Abbildung 19 Achsen Konfiguration 2

Nun wird die Zusammenfassung der Antriebskonfiguration aufgeführt. Bevor „Fertigstellen“ betätigt werden darf, muss der „Betrieb an einer Phase“ von Aus auf Ein gestellt werden da wir nur einen Einphasen Betrieb haben.

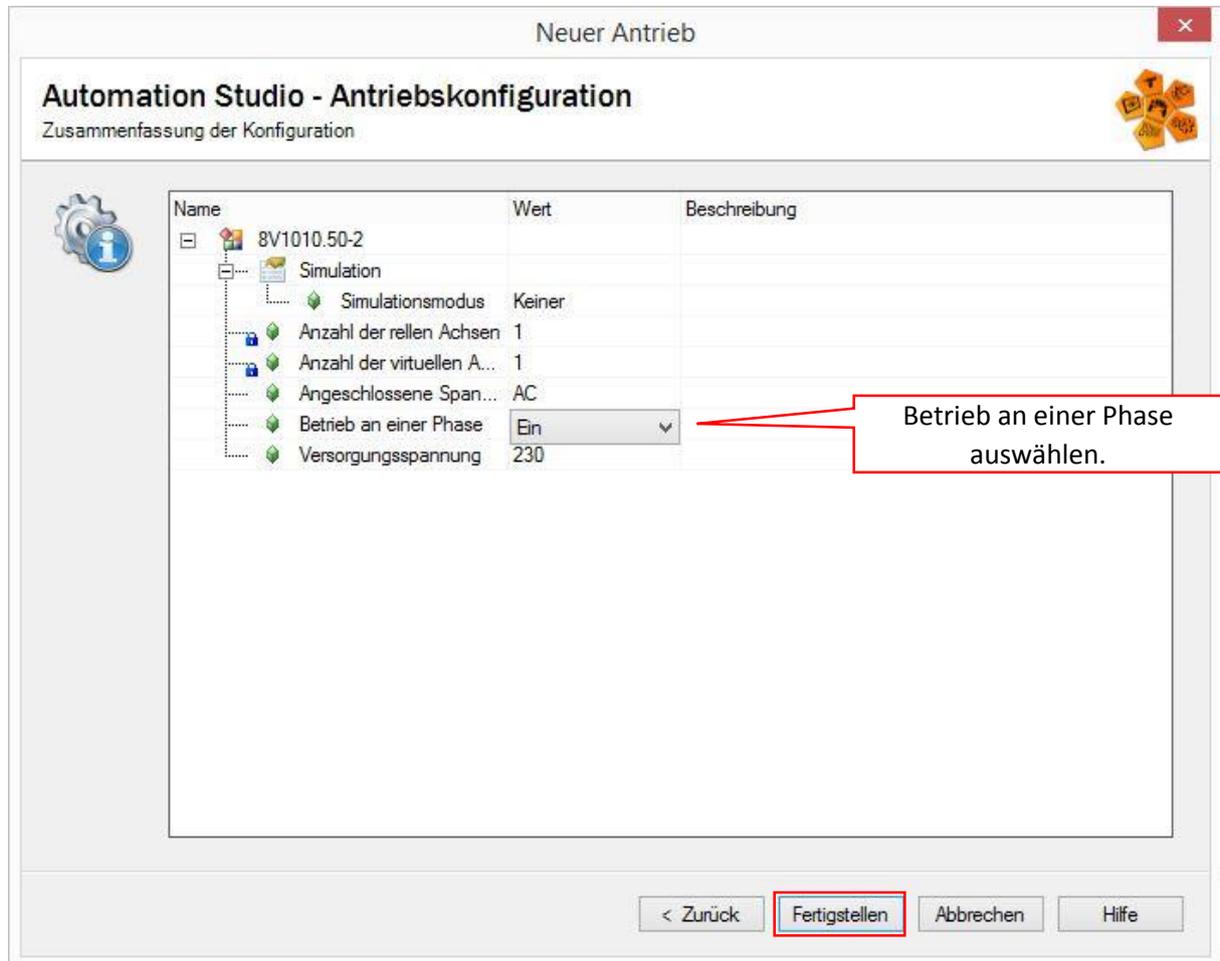


Abbildung 20 Antriebskonfiguration Fertigstellen

Im nächsten Bild ist die Hardware ohne Motor abgebildet.

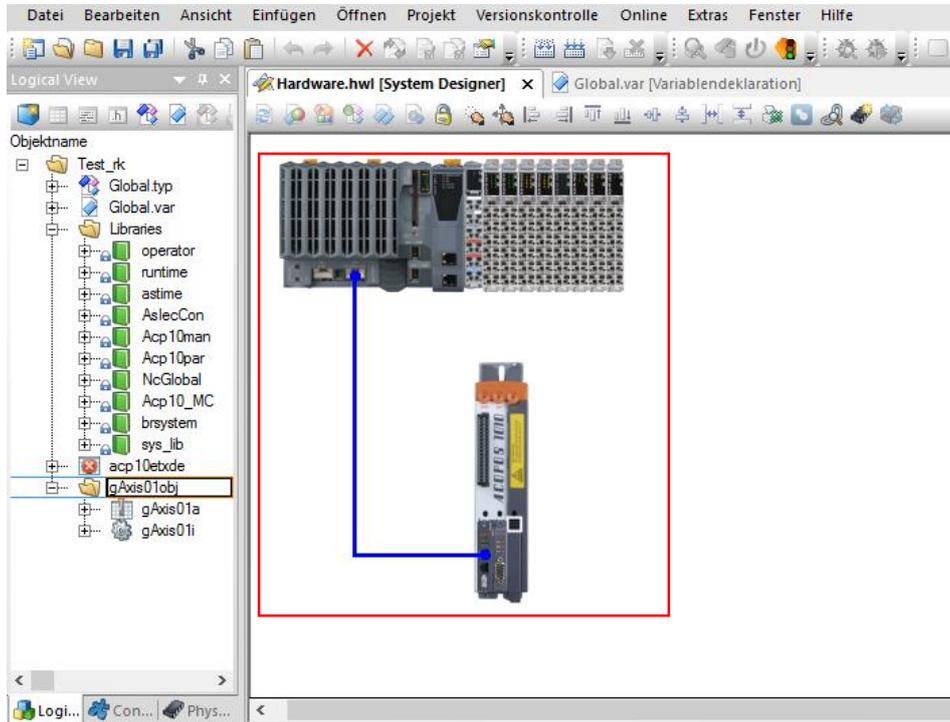


Abbildung 21 Hardware ohne Motor

Falls der Motor in der Hardware Konfiguration gefunden wurde, sieht der Hardware System Designer wie im Bild drunter aus.

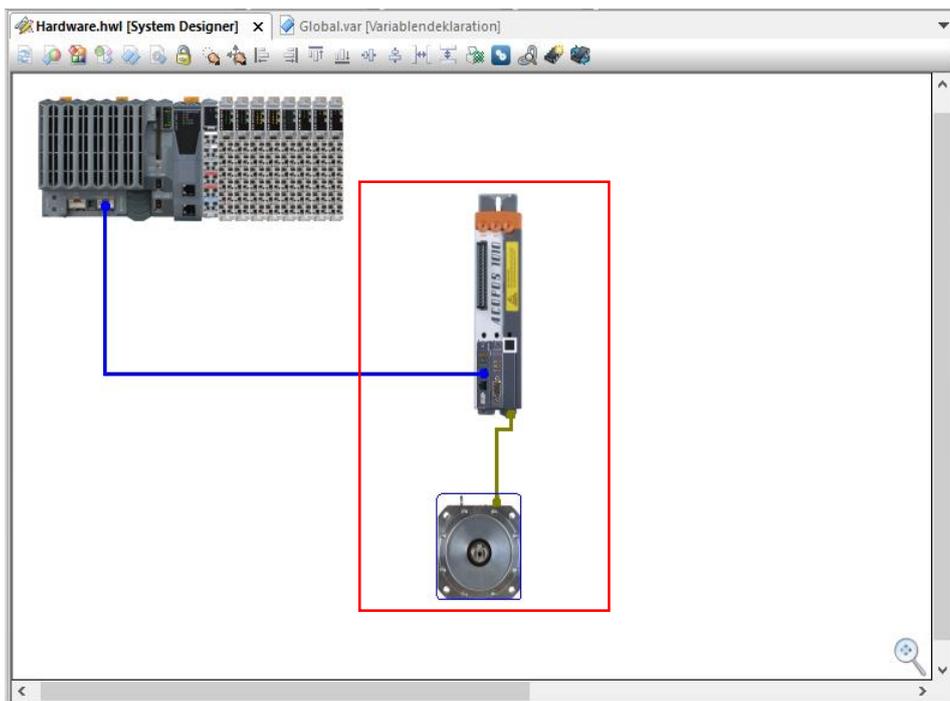


Abbildung 22 Hardware mit Motor

### 3. Motor manuell hinzufügen

#### 3.1 Download Motordaten

Konnte der Motor in den Antriebskonfigurationen hinzugefügt werden, kann zu Kapitel 4 gesprungen werden.

Um den gesuchten Motor in die Motion Anwendung einbinden zu können, muss dieser auf der Internetseite ( [www.br-automation.com/de-de/perfection-in-automation/](http://www.br-automation.com/de-de/perfection-in-automation/) ) von B&R herunter geladen werden.

Zuerst müssen Sie die Internet Seite aufrufen. Nach dem die Seite geöffnet wurde, muss in das Fenster „Suche“ die Produktnummer (8lva22.b1015d000-0) des Motors eingegeben werden und dies mit der Entertaste bestätigt werden.

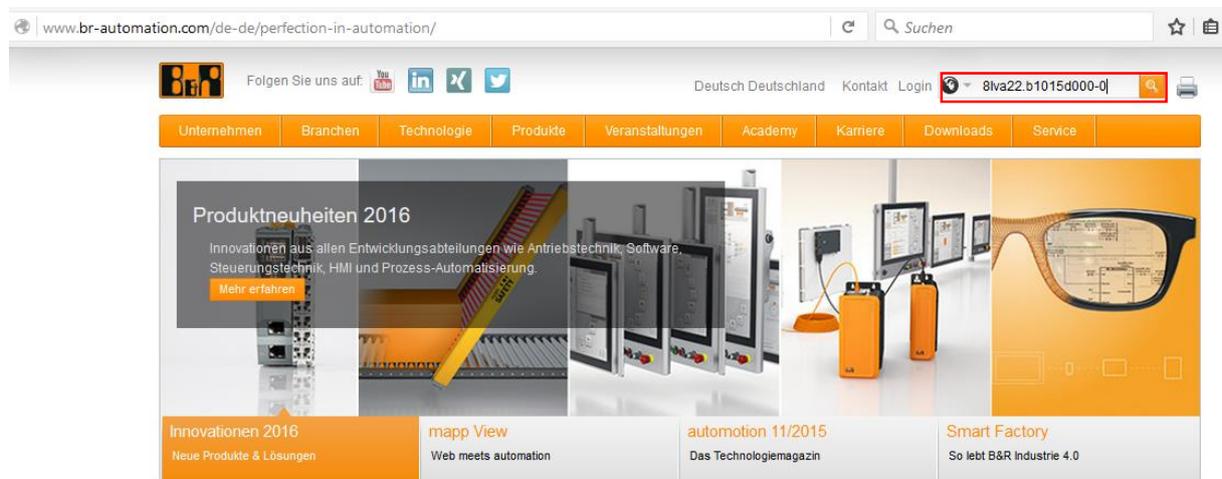


Abbildung 23 B&R Internetseite

Nun öffnet sich die Seite in dem der Link für den benötigten Download angezeigt wird. Dieser Link muss angeklickt werden.

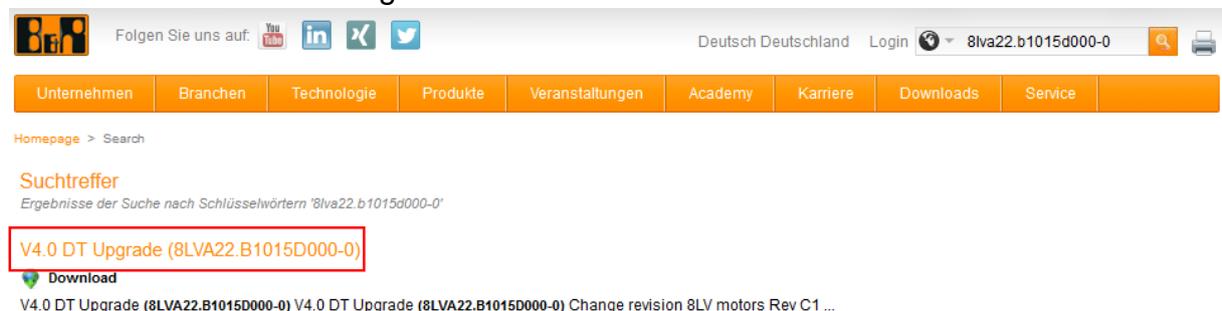


Abbildung 24 Link Download

Nun muss unter „Download“ die Datei für den Motor herunter geladen werden.

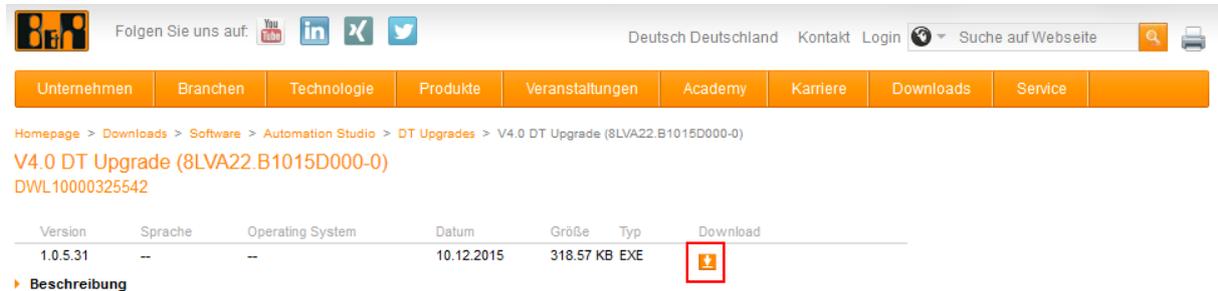


Abbildung 25 Download Datei

### 3.2 Motoren in Motion einfügen

Als nächstes gehen Sie wieder in das Automation Studio und öffnen Sie im Reiter „Extras“ den Unterpunkt „Upgrades“.

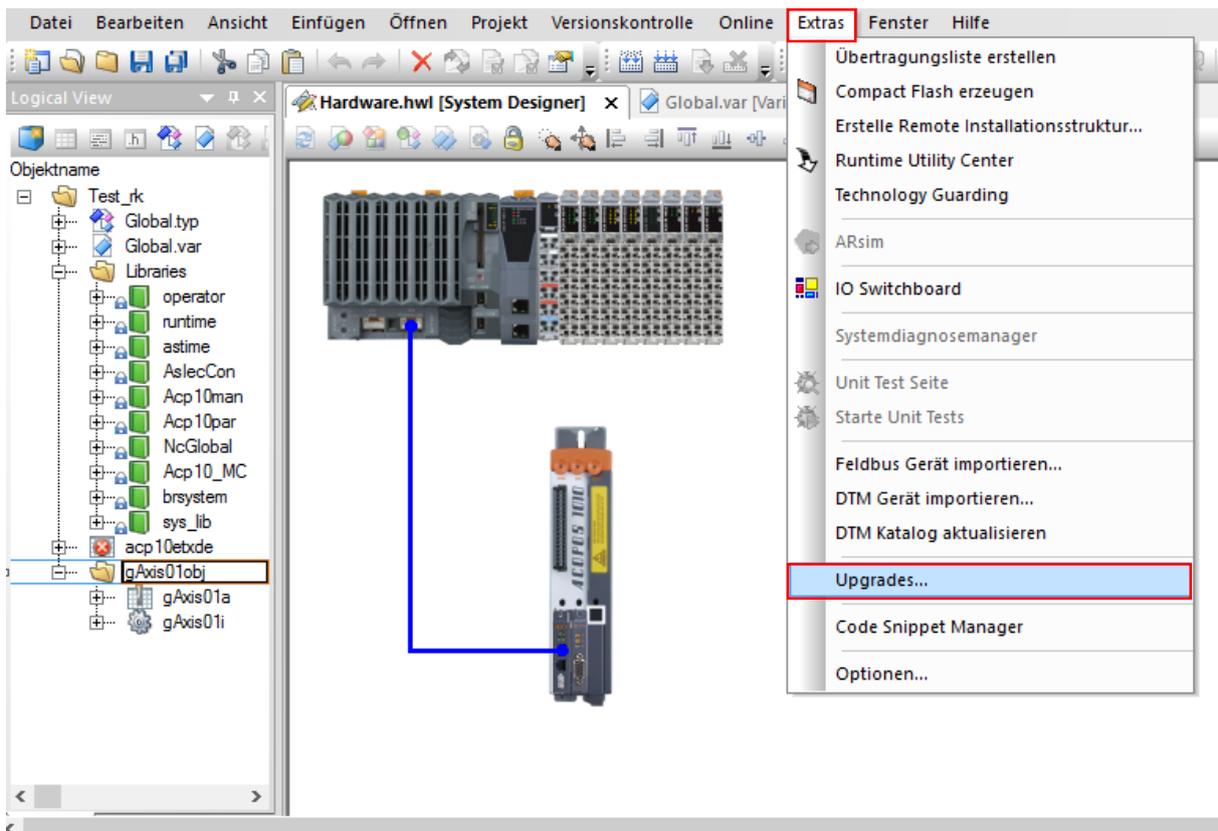


Abbildung 26 Upgrades

Jetzt öffnet sich das Upgrade Fenster. In diesem Fenster können sämtliche Upgrades herunter geladen werden, die automatisch vom Upgrade Manager gefunden worden sind.

Um den Motor manuell hinzuzufügen, klicken Sie auf „Lokal“.

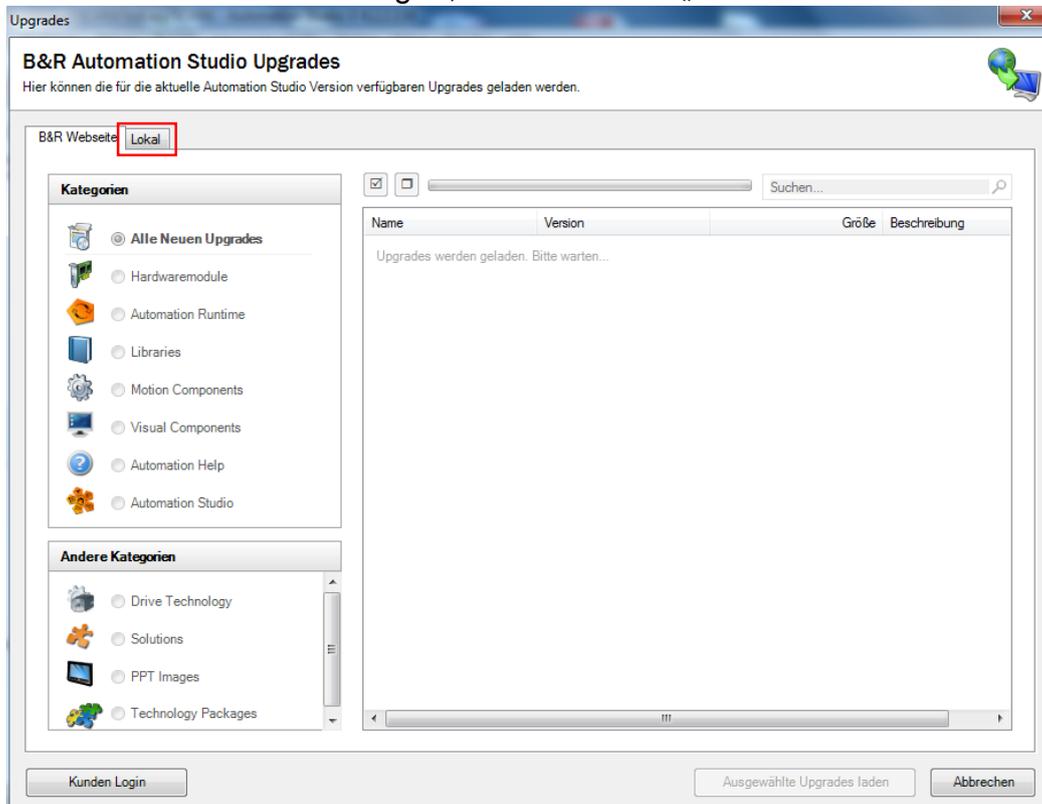


Abbildung 27 Lokal

Öffnen Sie nun den Pfad, indem Sie die Datei des Motors abgespeichert haben und wählen den Ordner indem die Datei ist aus. Als nächstes wählen Sie unter „Installierbare Upgrades“ den verwendeten Motor aus und klicken auf „Ausgewählte Upgrades installieren“.

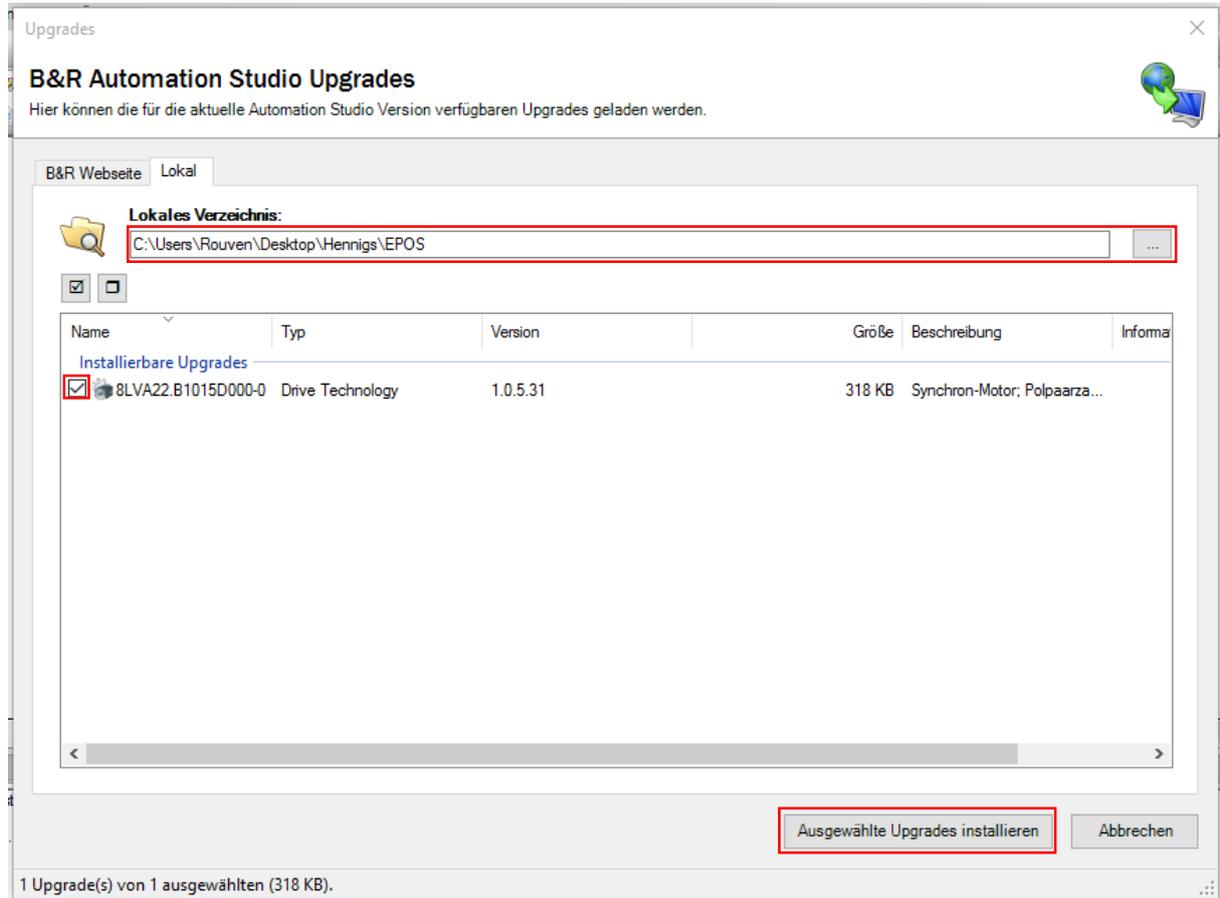


Abbildung 28 Upgrade Installieren

Jetzt „Installation starten“ drücken.

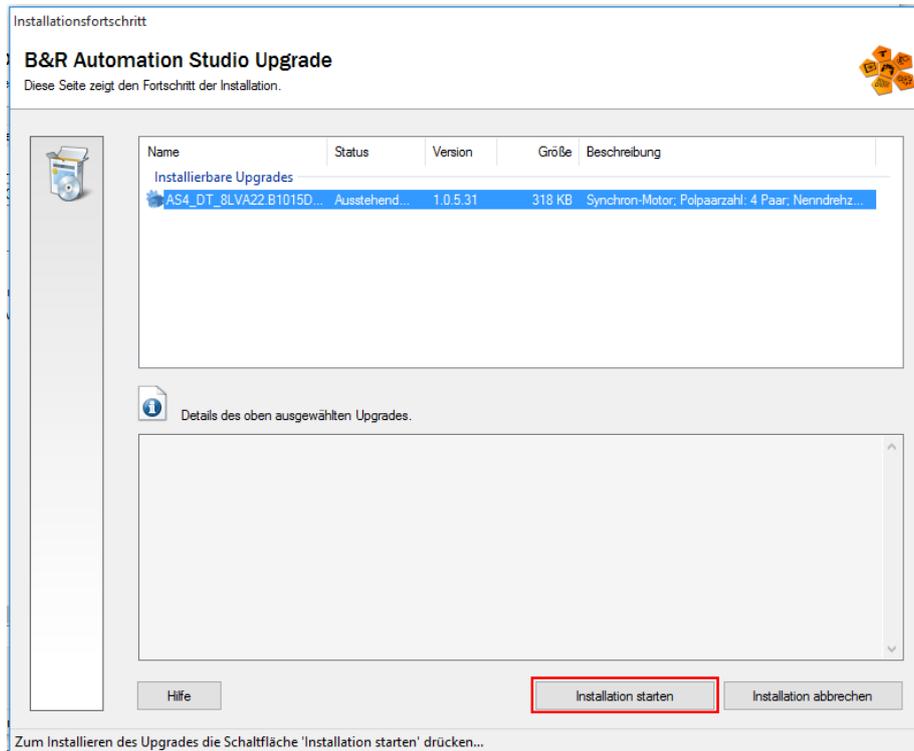


Abbildung 29 Installationen starten

Wenn die Installation fertig ist, steht unter „Status“ „Erledigt“ und schließen Sie das Fenster.

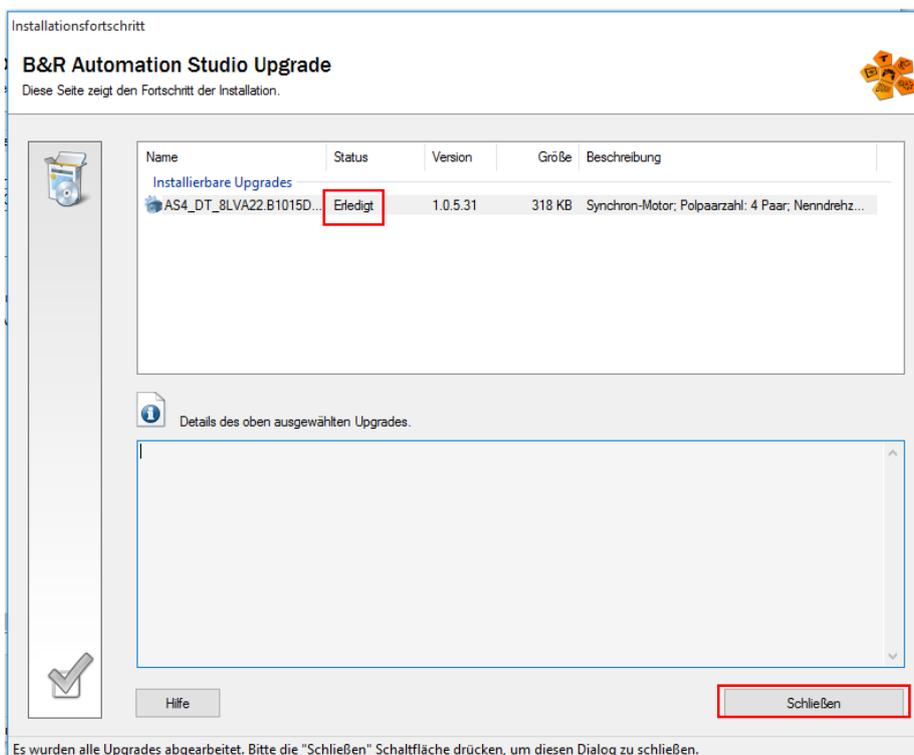


Abbildung 30 Installation Fertig

### 3.3 Motor mit dem Acopos verbinden

Nun gehen Sie wieder in ihre Hardware Konfiguration und klicken auf die umrandete Fläche am Acopos, um den Motor an dieser Stelle einzubinden. Es öffnet sich die Toolbox und im Fenster „Suche“ wird die Produktnummer ( 8LVA22.B1015D000-0 ) des Motors eingeben. Nun muss der Motor mit Doppelklick ausgewählt werden.

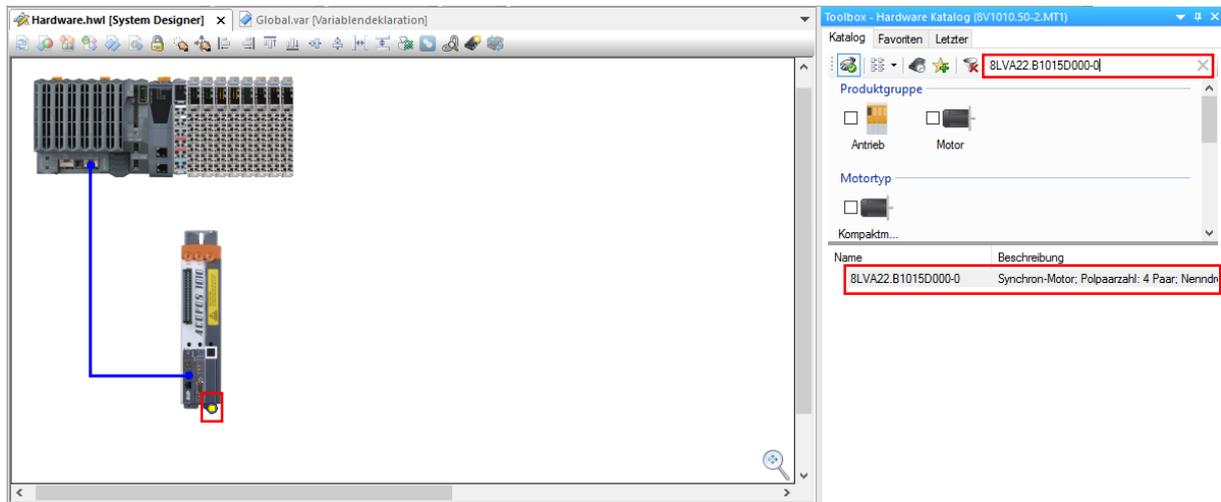


Abbildung 31 Motor manuell einbinden

So sieht nun die Hardwarekonfiguration aus.

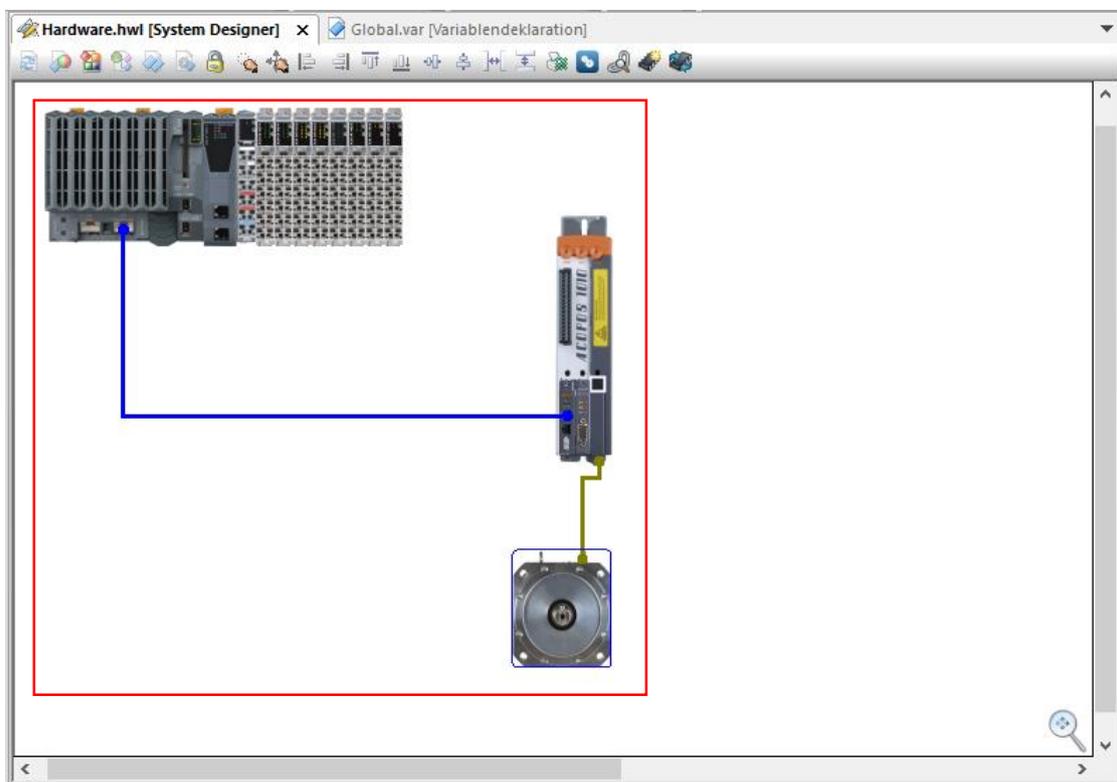


Abbildung 32 Hardware komplett

## 4. Motor Variable und Endat Verbindung

Mit eingefügtem Motor wurde die Achse als Globale Variable angelegt. Dies ist unter Logical View unter dem Punkt „Global.var“ zu finden. Mit Doppelklick auf „Global.var“ öffnet sich das Fenster indem die Variable angelegt wurde.

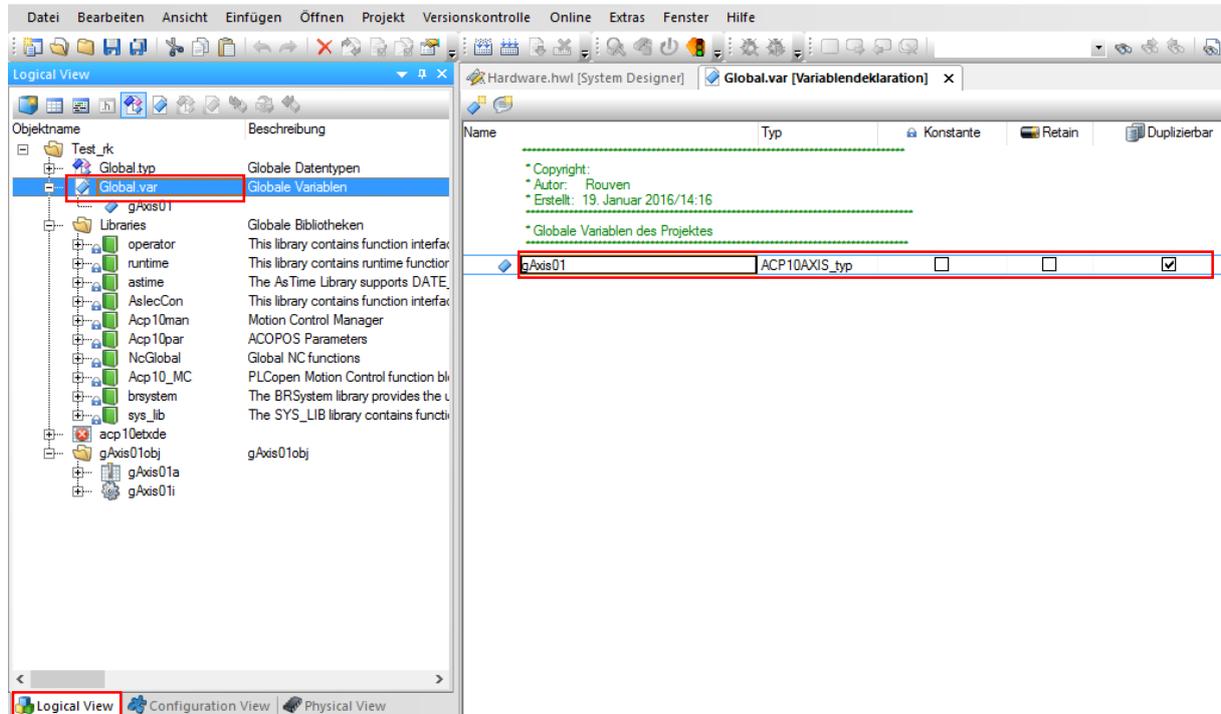


Abbildung 33 Motor Variable

Nun wird der Drehlagegeber des Motors mit dem Servoumrichter verbunden. Per Drag and Drop wird eine Verbindung vom EC Anschluss des Motors zum EC Anschluss des Acopos gezogen. (Hellgrüne Verbindung)

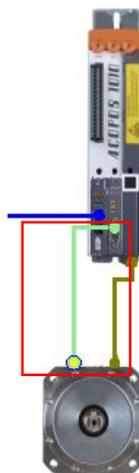


Abbildung 34 Endat Verbindung

## 5. Einrichten der Ethernet Verbindung

### 5.1 Ethernet Konfiguration

Wechseln Sie auf den Physical View um eine Ethernet Verbindung herzustellen. Mit Rechtsklick auf „ETH“ öffnet sich ein Fenster in dem „Konfiguration“ ausgewählt werden muss.

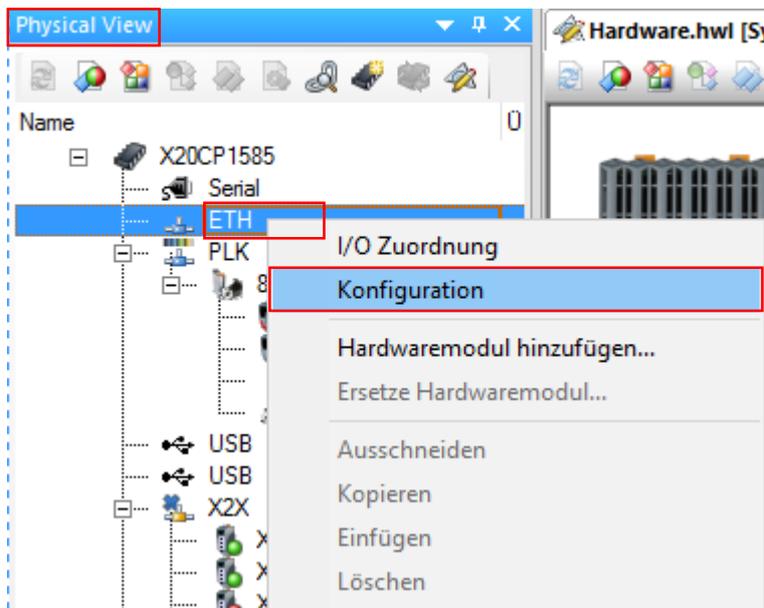


Abbildung 35 ETH Konfiguration

Nun öffnet sich die Konfiguration der CPU. Hier muss der „Modus“ von „IP Adresse von DHCP Server beziehen“ auf „IP Adresse manuell festlegen“ geändert werden.

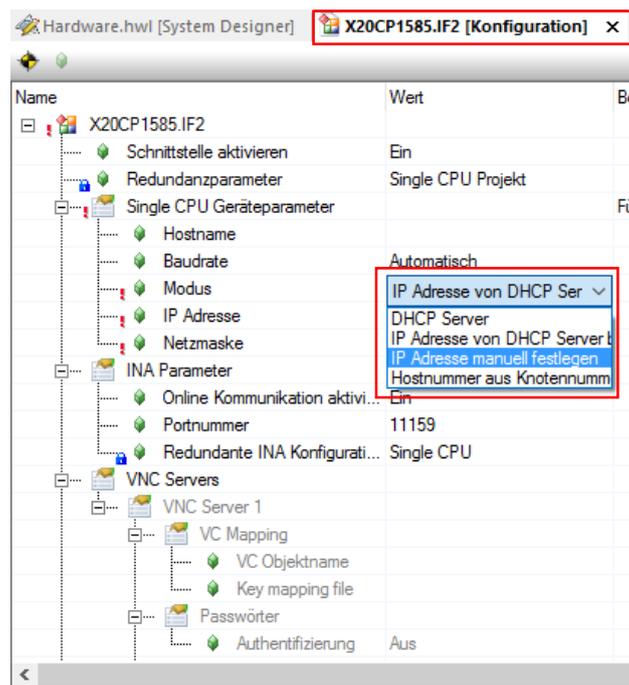


Abbildung 36 ETH Modus

Wie folgt werden IP Adresse und die Netzmaske der CPU definiert. Wichtig ist das zuerst die Netzmaske und danach die IP Adresse eingetragen wird.

Netzmaske: 255.255.255.0

IP Adresse: 192.168.0.160

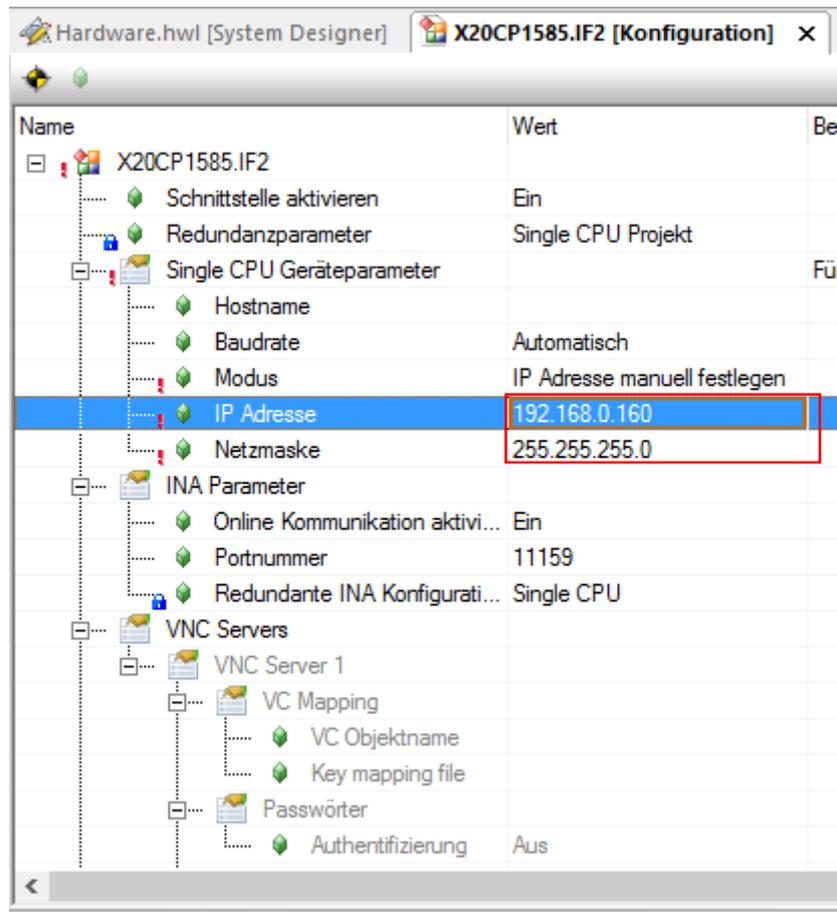


Abbildung 37 IP Adresse

Nachdem die Daten geändert wurden, müssen diese gespeichert werden.

## 5.2 Knotennummer des Acopos

Die POWERLINK Knotennummer wird mit zwei HEX Codierschaltern eingestellt:

1. 16-er Stelle (Hi)
2. 1-er Stelle (Lo)



Die am Acopos hardwareseitig eingestellte Knotennummer muss mit der Softwareseitig eingesetzte Knotennummer übereinstimmen.

Hierzu wird der Physical View geöffnet und der Acopos markieren.

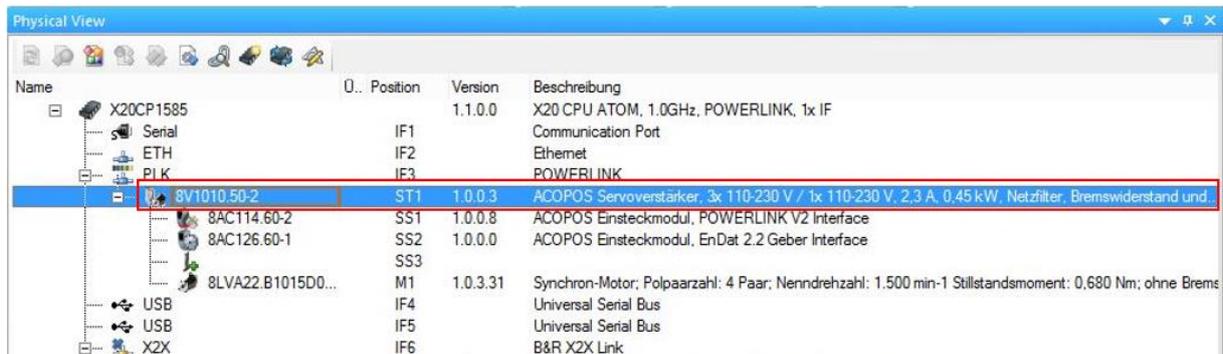


Abbildung 38 Physical View Acopos

Nachdem der Acopos markiert ist, muss die Position ST1 zu ST2 geändert werden. Dies kann über einen einfacklick auf die Position des Acopos erfolgen.

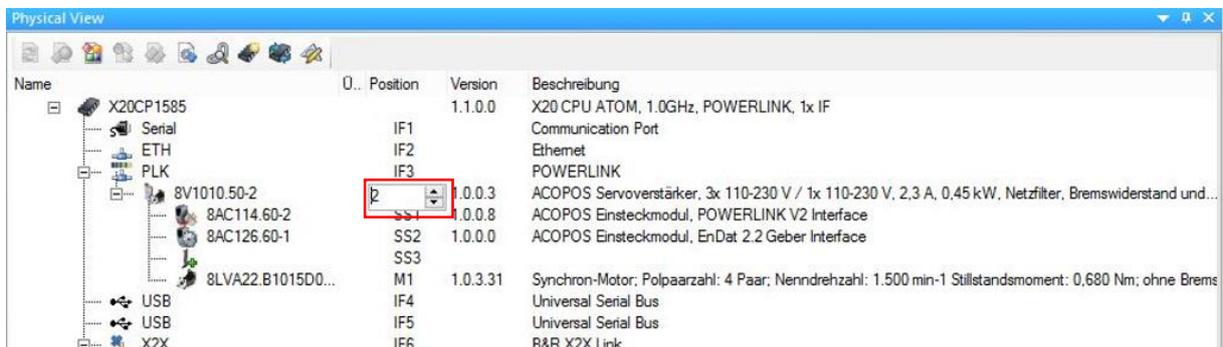


Abbildung 39 Knotennummer

Damit ist die Hardwarekonfiguration abgeschlossen ist muss die Einstellungen speichert werden.

## 6. Compact-Flash erstellen

Bei einer Erstinbetriebnahme des Projektes muss eine Compact-Flash erstellt werden, da eine neue Hardware Konfiguration vorliegt und der Speicher neu formatiert wird.

Hierfür wird ein Compact-Flash-Kartenleser benötigt.

Wenn Sie die Compact-Flash aus der SPS entfernen muss darauf geachtet werden das diese vorher ausgeschaltet wird.

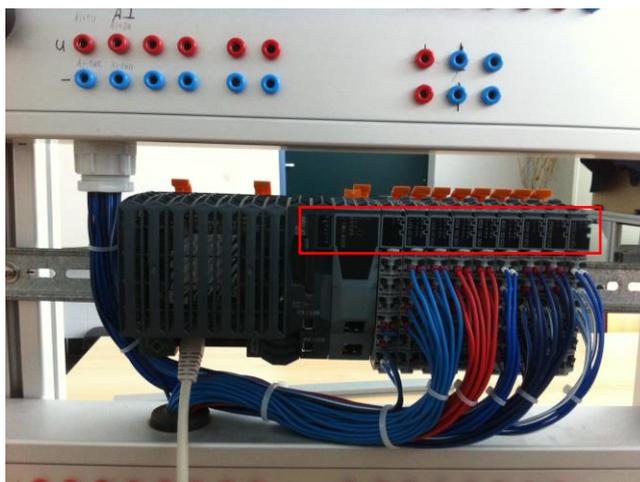


Abbildung 40 Ausgeschaltete SPS

Ist die Speicherkarte im Kartenlesegerät und mit dem PC verbunden kann über die Motion Oberfläche unter dem Punkt „Extras“ eine „Compact-Flash erzeugen“ ausgeführt werden.

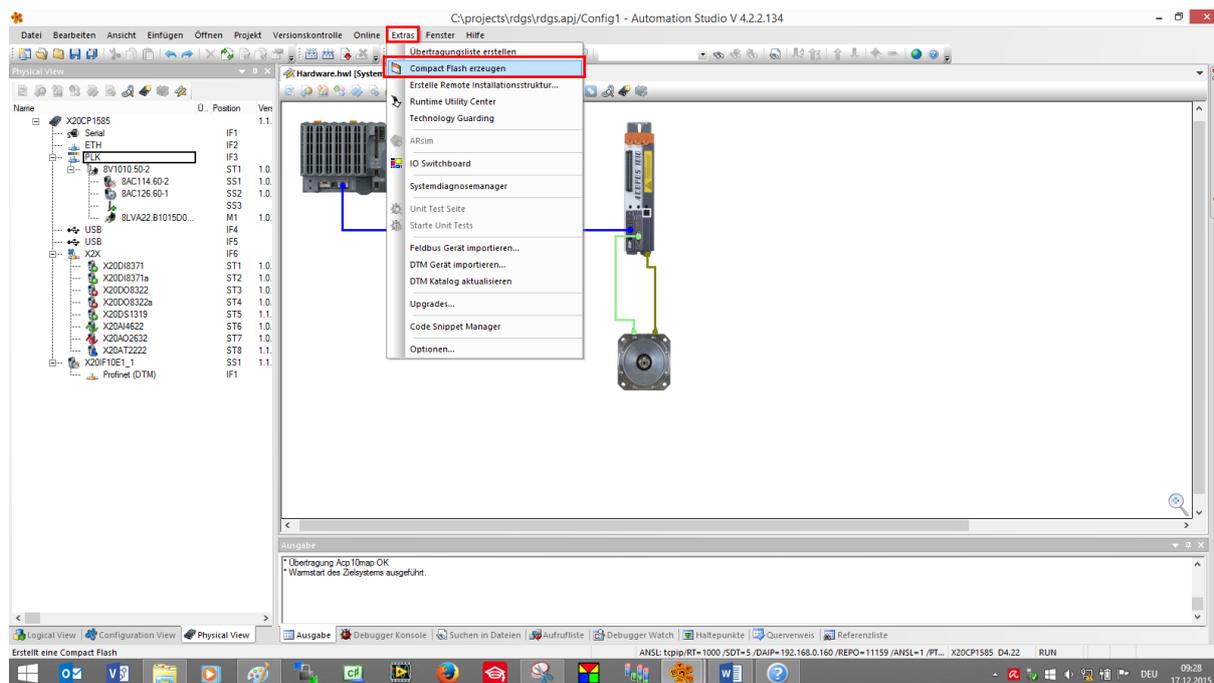


Abbildung 41 Compact-Flash

Es öffnet sich das Fenster „Compact Flash erstellen“. Hier wird das entsprechende Laufwerk ausgewählt an dem das Kartenlesegerät angeschlossen ist. Um die Speicherkarte zu beschreiben drücken Sie auf „Compact Flash erstellen“.

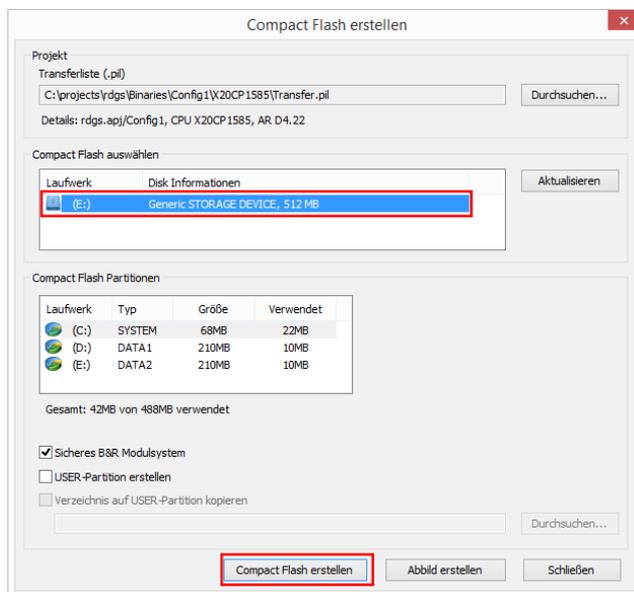


Abbildung 42 Compact-Flash erstellen

Es erscheint das Fenster „Runtime Utility Center“ dieses wird mit „Ja“ bestätigt.

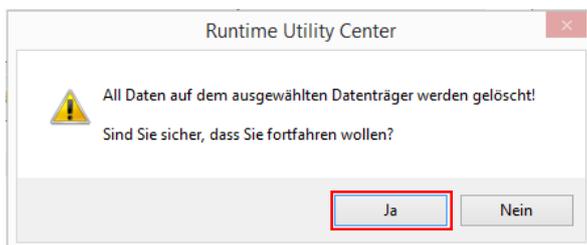


Abbildung 43 Runtime Utility Center

Wenn der Vorgang erfolgreich muss dieses mit „OK“ bestätigt werden.

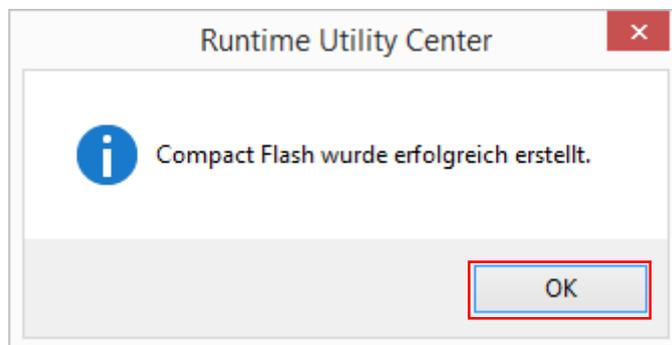


Abbildung 44 Compact Flash Erstellt

Jetzt ist das Projekt auf der Compact Flash und kann in die ausgeschaltete SPS wieder eingesteckt werden. Anschließend kann die SPS wieder eingeschaltet werden.

## 7. Online verbinden

Um eine Online Verbindung aufzubauen müssen Sie die SPS mit ihrem Rechner über ein Ethernet Kabel verbinden. Es ist darauf zu achten das der Linke Steckplatz (ETH) der Steuerung verwendet wird.

Die IP Adressen des PG und der Steuerung müssen im gleichen Netzwerk sein.

IP Adresse Steuerung : 192.168.0.160

IP Adresse PG : 192.160.0.XXX

Über den Punkt „Online“ muss der Unterpunkt „Einstellungen“ ausgewählt werden.

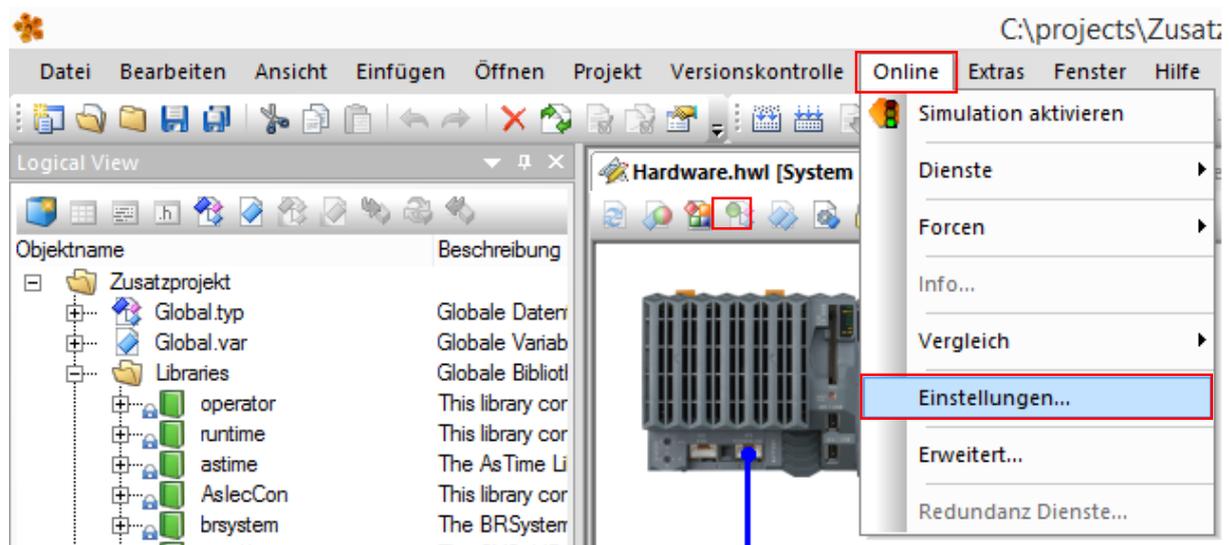


Abbildung 45 Online

Es öffnet sich das „Online Einstellungen“ Fenster. Um die angeschlossene Steuerung zu finden, muss auf „Durchsuchen“ geklickt werden.

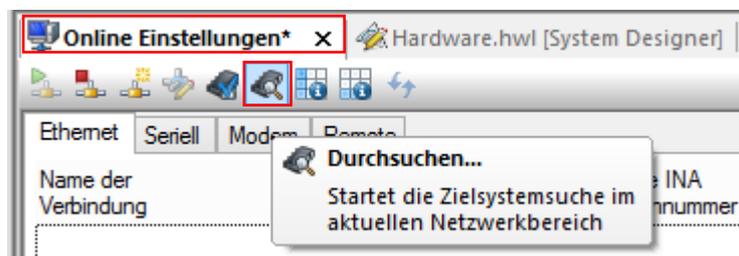


Abbildung 46 Online Einstellungen

Auf der rechten Seite der „Online Einstellungen“ wird die angeschlossene SPS angezeigt.

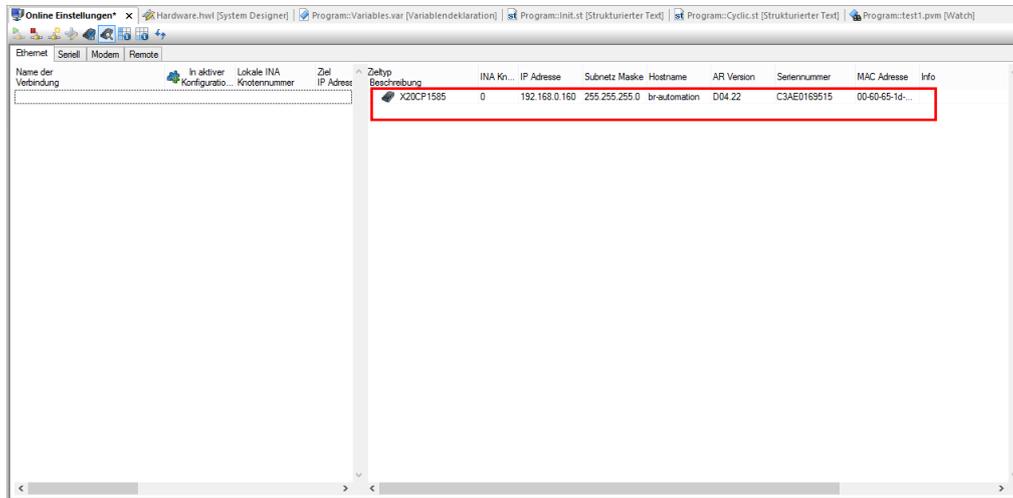


Abbildung 47 Angeschlossene SPS

Wählen Sie mit Rechtsklick die angezeigte Steuerung aus und über „Verbinden“ kommunizieren das PG und die Steuerung miteinander.

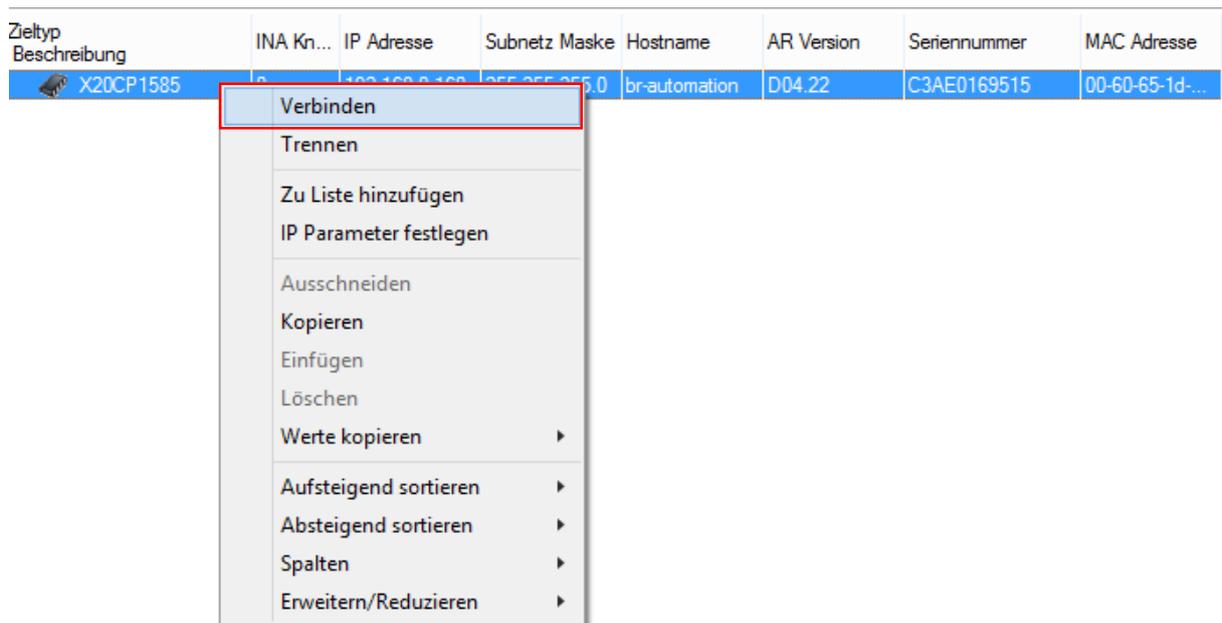


Abbildung 48 Verbinden mit der SPS

Ist die Verbindung hergestellt, wird dieses im unteren Tab über „RUN“ angezeigt.



Abbildung 49 Statusanzeige

## 8. Parametrieren und Testen des Motors

### 8.1 Auswahl der Achse für die Parametrierung

Um die Achse erstmalig anzusteuern wird über die Registerkarte „Hardware.hwl [System Designer]“ der Acopos mit Rechtsklick markiert und der Unterpunkt „Test“ ausgewählt.

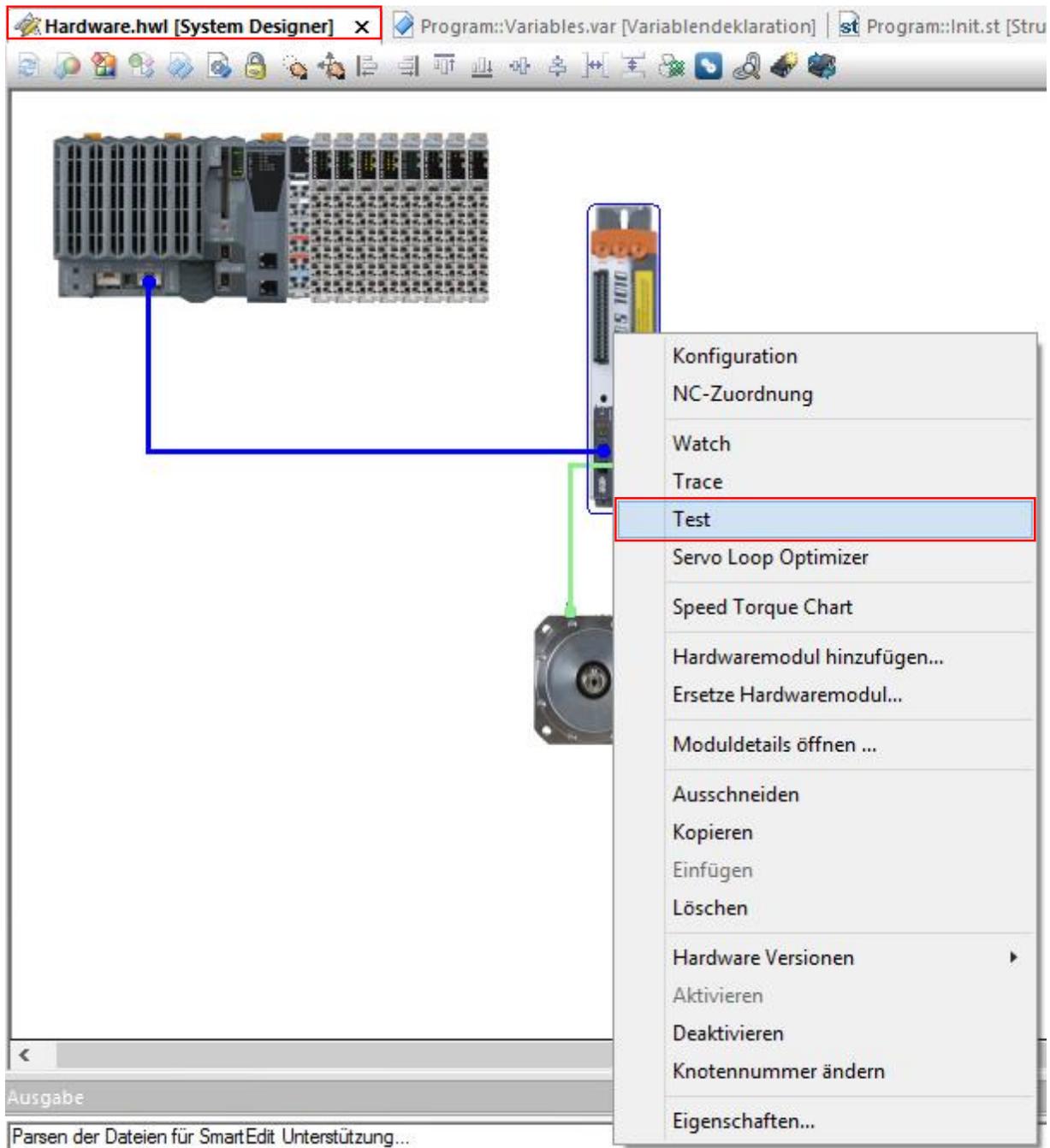


Abbildung 50 Test öffnen

Es öffnet sich das Fenster „Objekt auswählen“, darin wählt man die Achse „gAxis01“ aus und bestätigt diese mit „OK“.

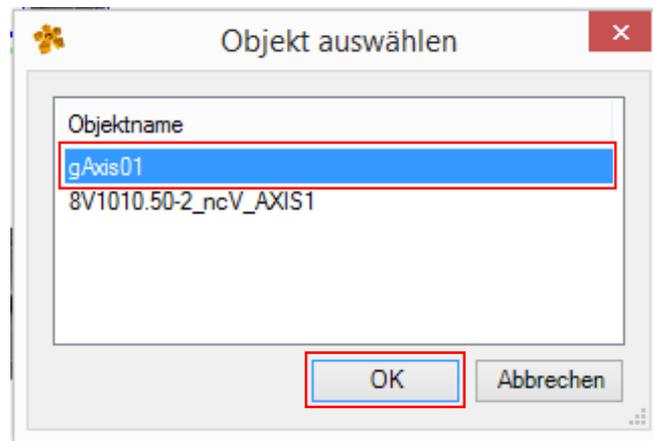


Abbildung 51 Achsen Auswahl

Es öffnet sich das „NC-Test Fenster“ (gAxis01.mdc[Test]). Es öffnet sich ein Fenster indem der „Exklusiver Modus“ bestätigt werden muss, da nicht im laufenden Programm gearbeitet wird.

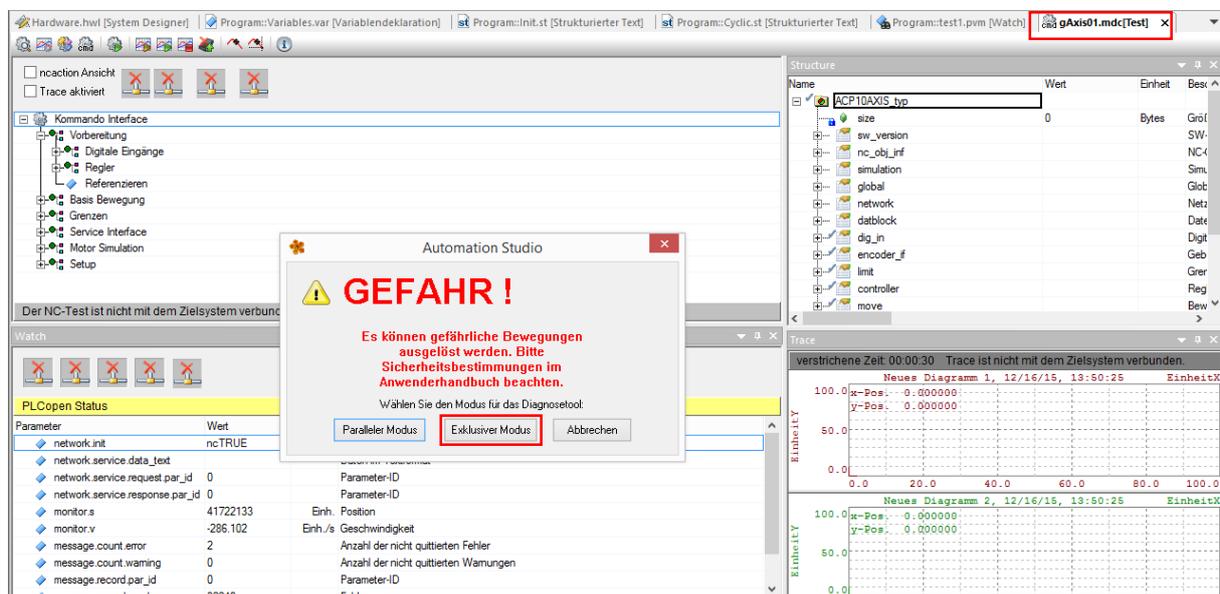
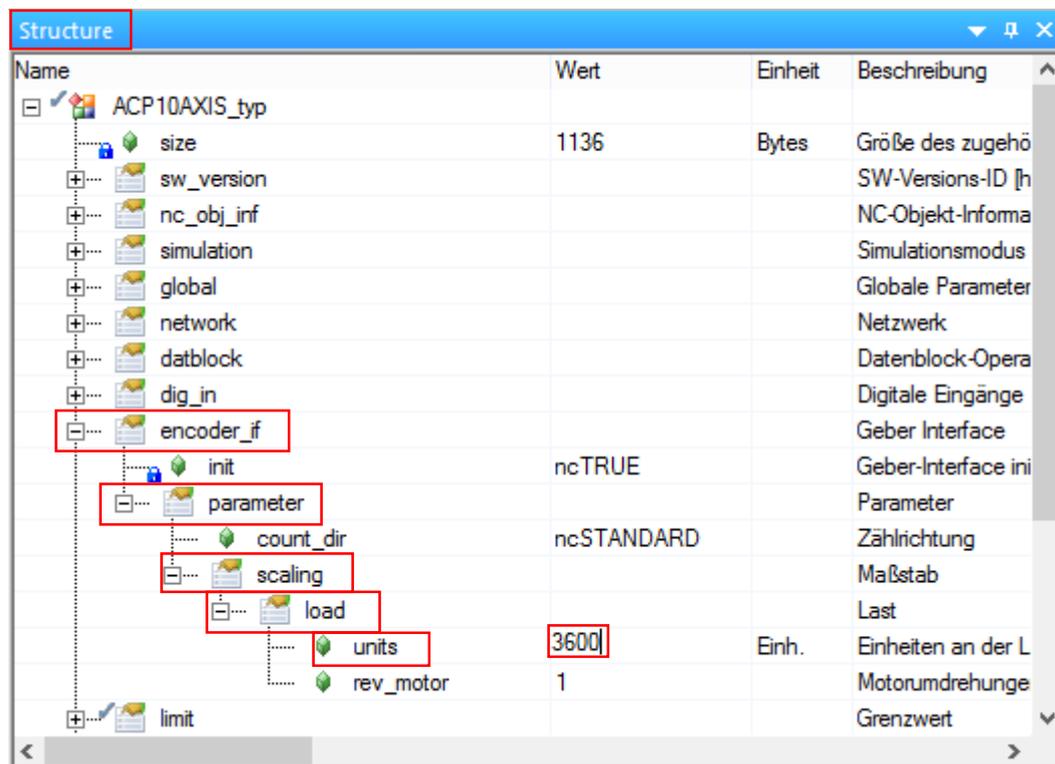


Abbildung 52 Exklusiver Modus

Parallel Modus steht für „Bedienung mit dem geschriebenen Programm“.

## 8.2 Geber Einheiten pro Motorumdrehungen

Im Fenster „Structure“ unter „encoder\_if“ → „parameter“ → „scaling“ → „load“ → „units“ werden die Einheiten pro Motorumdrehung festgelegt. Wir haben 3600 Einheiten pro Motorumdrehung eingestellt. Der eingestellte Wert muss mit der „Enter“ Taste des PG`s bestätigt werden. Dies bedeutet dass pro Grad 10 units eingestellt sind. Je größer der Wert, umso genauer lässt sich die Welle positionieren.



Name	Wert	Einheit	Beschreibung
ACP10AXIS_typ			
size	1136	Bytes	Größe des zugehö
sw_version			SW-Versions-ID [h
nc_obj_inf			NC-Objekt-Infoma
simulation			Simulationsmodus
global			Globale Parameter
network			Netzwerk
datblock			Datenblock-Opera
dig_in			Digitale Eingänge
encoder_if			Geber Interface
init	ncTRUE		Geber-Interface ini
parameter			Parameter
count_dir	ncSTANDARD		Zählrichtung
scaling			Maßstab
load			Last
units	3600	Einh.	Einheiten an der L
rev_motor	1		Motorumdrehunge
limit			Grenzwert

Abbildung 53 Einheiten pro Motorumdrehung

Hinweis:

Nach dem die Geberskalierung umgestellt und auf die Steuerung hochgeladen wurde, verändert sich nach dem Neustart der Anlage der angezeigte Geberwert im Fenster „Watch“. Aufgrund der neuen Geberskalierung werden die Einheiten neu berechnet.

### 8.3 Motorparameter abspeichern

Um die Motorparameter abzuspeichern wird in der Taskleiste „Test“ der Unterpunkt „Markierte Parameter speichern...“ ausgewählt.

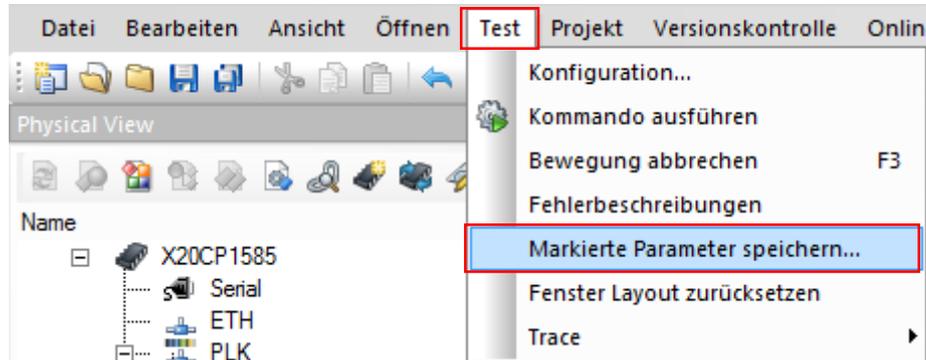


Abbildung 54 Parameter Speichern

Es öffnet sich ein Fenster „Speichern unter“, hier kann die Datei „gAxis01i.ax“ überspeichert werden. Um den geänderten Parameter zu sichern, mit „Speichern“ bestätigen.

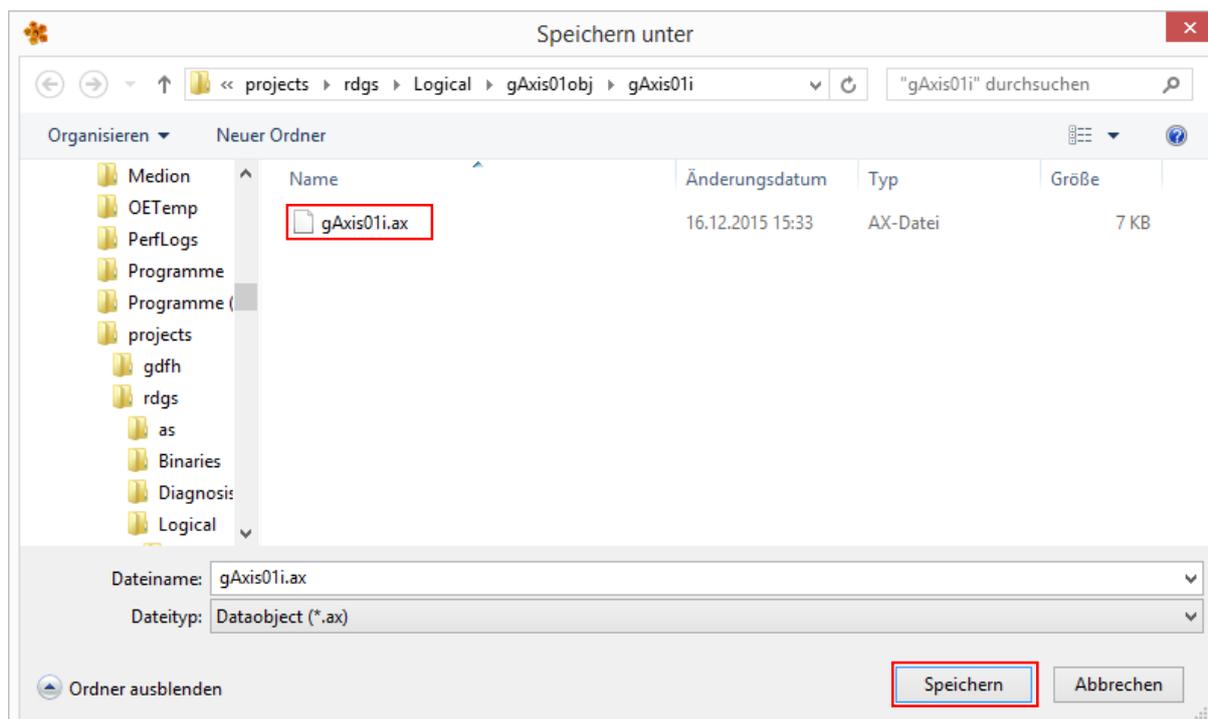


Abbildung 55 Parameter Speicherort

#### 8.4 Projekt via Netzwerk übertragen

Damit die Parameter wirksam werden, müssen sie auf SPS via Netzwerk übertragen werden.

Zum Übertragen muss das „NC-Test Fenster“ geschlossen werden.

Um die Parameter zu übertragen muss das rot umrandete Symbol „In das Zielsystem übertragen“ angewählt werden.



Abbildung 56 Zielsystem übertragen

Nun ist im „Ausgabe“ Fenster zu erkennen, dass das Programm überprüft und kompiliert wird.



Abbildung 57 Kompilieren

Nach dem kompilieren öffnet sich das Fenster „Übertragung des Projektes“. Darin wird angezeigt ob Hardwarekonflikte, Lizenzprobleme oder Betriebssystemkonflikte vorhanden sind. Des Weiteren wie viele Kalt- und Warmstarts benötigt werden und ob Softwarekonflikte vorhanden sind. Dies mit „OK“ bestätigen um zu Übertragung.



Abbildung 58 Infos zur Übertragung des Projektes

Nachdem das Projekt erfolgreich Übertragen wurde und die Verbindung zur SPS wieder hergestellt ist, erscheint das Fenster „Projekt übertragen“. Dies muss mit „OK“ bestätigt werden.

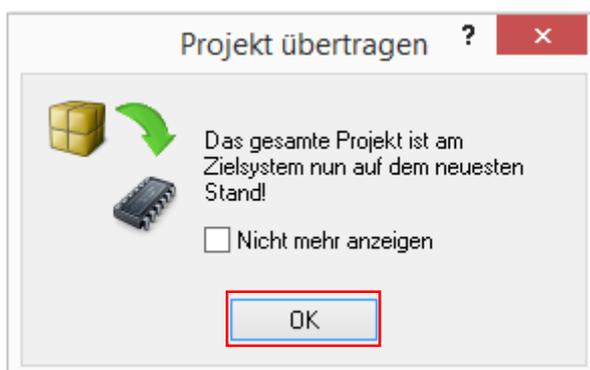


Abbildung 59 Projekt übertragen

### 8.5 Referenzieren des Motors

Um die Achse zu Referenzieren wird über die Registerkarte „Hardware.hwl [System Designer]“ der Acopos mit Rechtsklick markiert und der Unterpunkt „Test“ ausgewählt.

Es öffnet sich das Fenster „Objekt auswählen“, darin wählt man die Achse „gAxis01“ aus und bestätigt diese mit „OK“.

Es öffnet sich das „NC-Test Fenster“ (gAxis01.mdc[Test]). (Dies ist in Kapitel 8 unter dem Punkt 8.1 nachzulesen)

Jetzt muss der Motor in die gewünschte Ausgangsstellung von Hand gedreht werden und durch einmaliges anklicken auf „Referenzieren“ öffnet sich das „Structure“ Fenster.

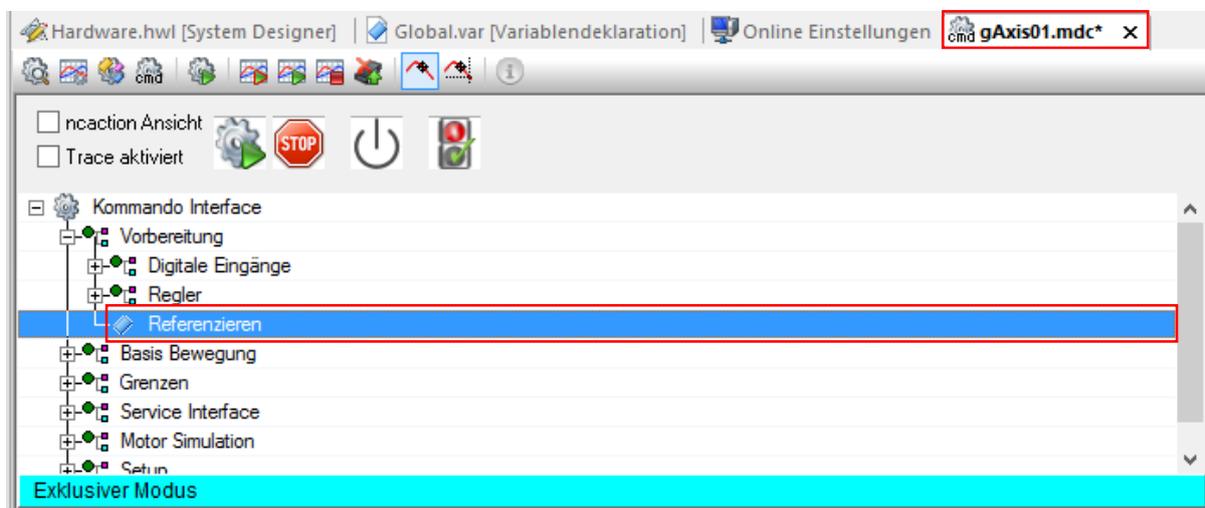


Abbildung 60 Structure Fenster

Nachdem man die Welle des Motors in die gewünschte „null“ Stellung gebracht hat, wird im Fenster „Structure“ der Parameter „mode“ von „ncDIRECT“ auf „ncHOME\_OFFSET“ geändert. Dieses wird durchgeführt um diese Position als „null“ Stellung zu speichern.

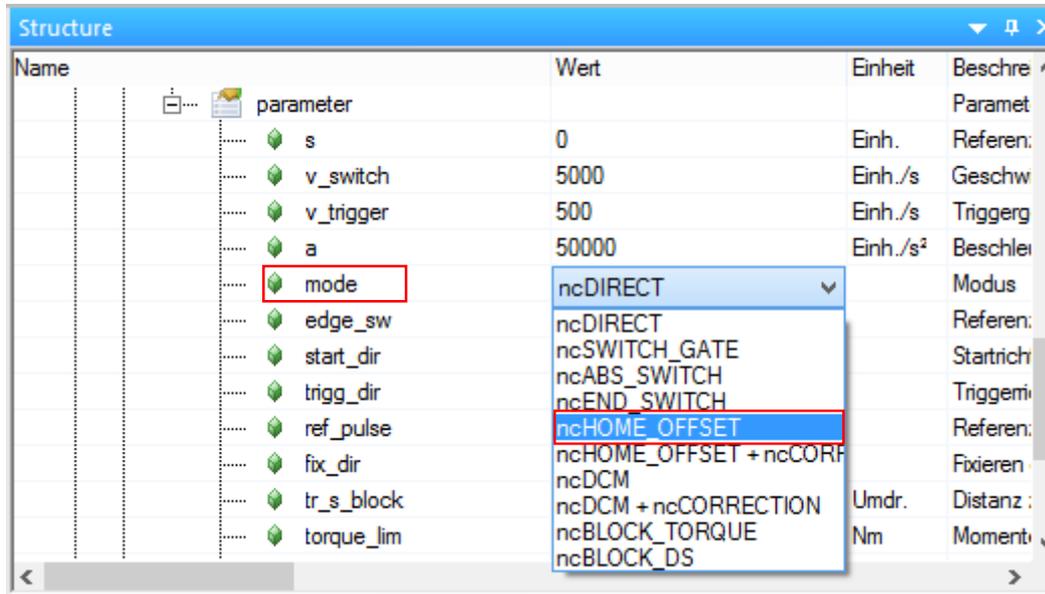


Abbildung 61 Grundposition

Im Fenster „Watch“ muss der Wert des Parameters „monitor.s“ ablesen werden und sich auf einem Zettel notiert werden.

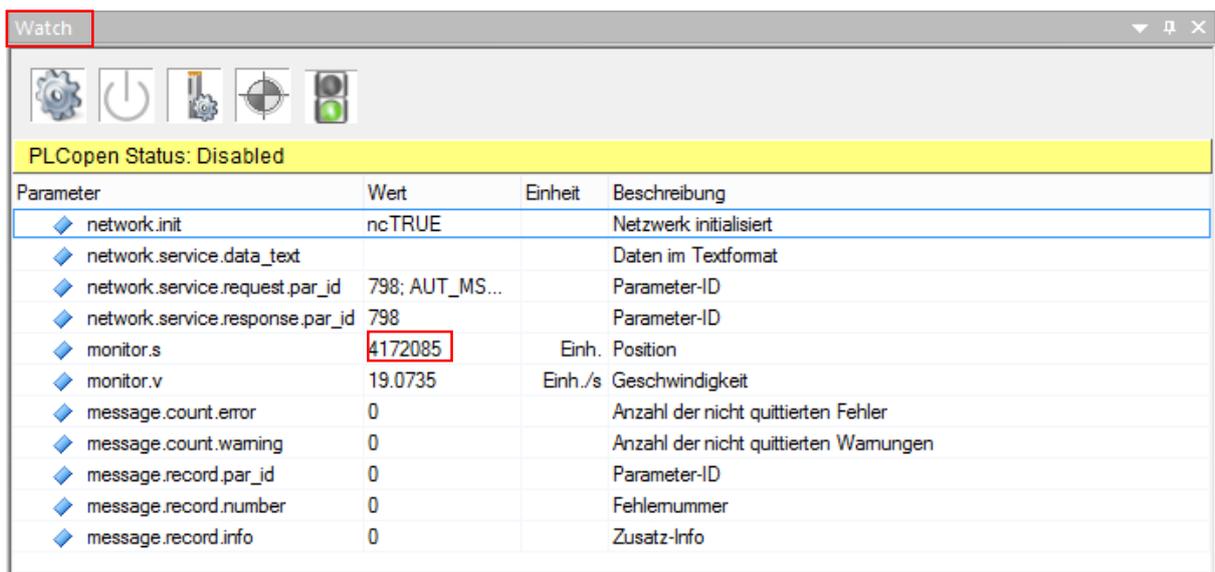
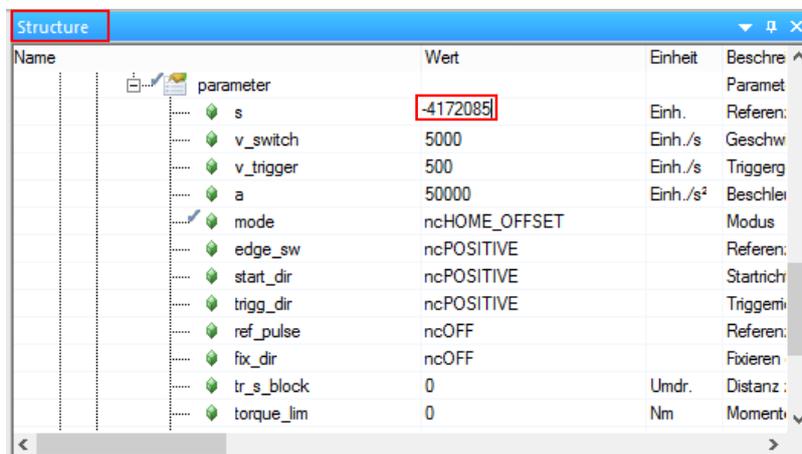


Abbildung 62 monitor.s

In diesem Beispiel ist es der Wert „4 172 085“, bei anderen Anwendungen kann der Wert abweichen.

Nun muss der abgelesene Wert des Parameters „monitor.s“ mit geändertem Vorzeichen (-4 172 085) im Fenster „Structure“ beim Parameter „s“ eingegeben werden. Dies wird getan, weil der eingegeben Wert auf den Wert des Absolutwertgebers addiert wird und somit der Antrieb auf die gewünschte Position referenziert ist.



Name	Wert	Einheit	Beschre
parameter			Paramet
s	-4172085	Einh.	Referen:
v_switch	5000	Einh./s	Geschw
v_trigger	500	Einh./s	Triggere
a	50000	Einh./s <sup>2</sup>	Beschlei
mode	ncHOME_OFFSET		Modus
edge_sw	ncPOSITIVE		Referen:
start_dir	ncPOSITIVE		Starttrich
trigg_dir	ncPOSITIVE		Triggere
ref_pulse	ncOFF		Referen:
fix_dir	ncOFF		Fixieren
tr_s_block	0	Umdr.	Distanz :
torque_lim	0	Nm	Moment

Abbildung 63 Parameter „s“

Durch Doppelklick auf „Referenzieren“ wird die gewünschte „null“ Position übernommen.

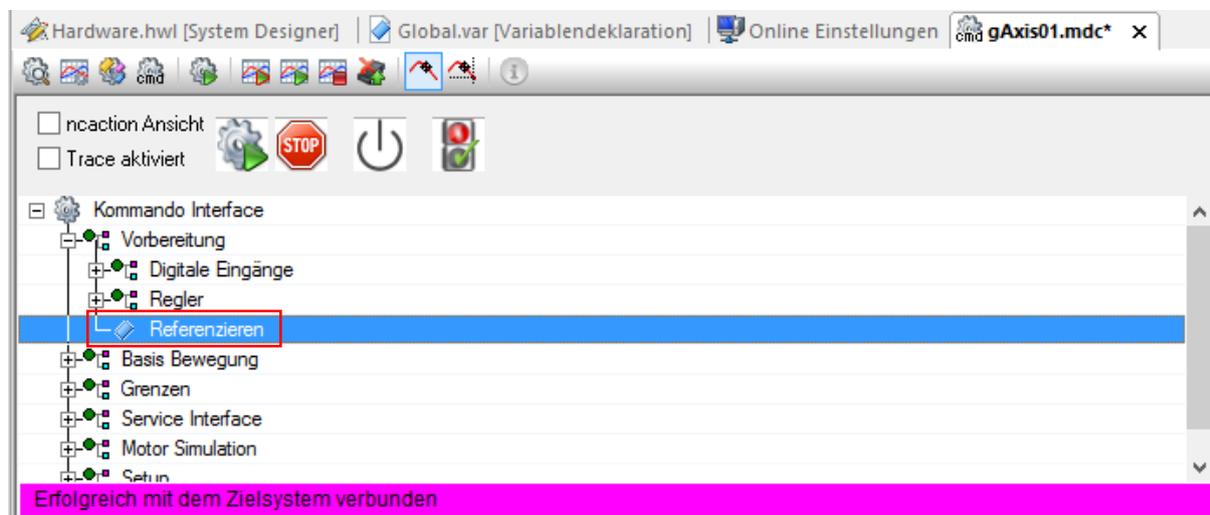


Abbildung 64 Referenzieren der „null“ Position

Im „Watch“ Fenster hat sich der Wert des Parameters „monitor.s“ auf 0 geändert. Das Symbol für referenziert ist nun farbig hinterlegt.

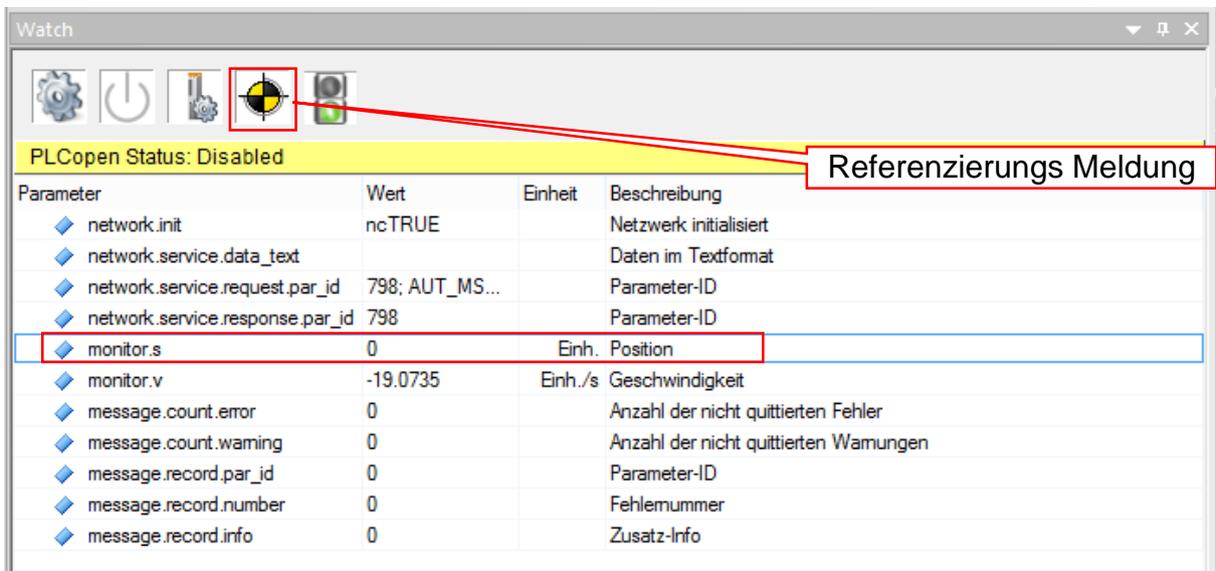


Abbildung 65 Watch Fenster

### 8.6 Tuning des Antriebes

Jetzt im Hauptfenster unter „Setup“ → “Regler Autotuning“ auf „Tuning starten“ einfach klicken.

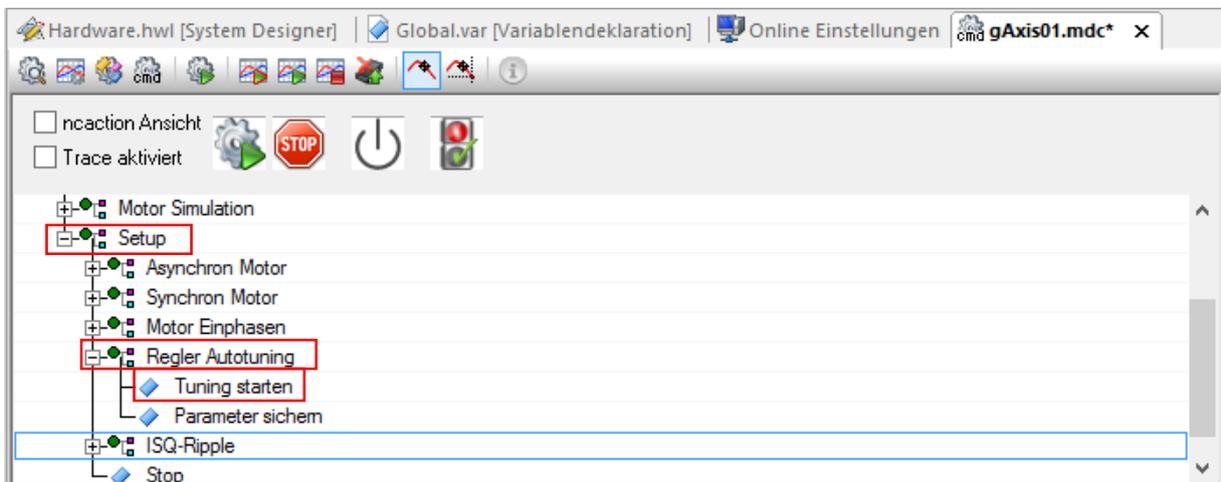


Abbildung 66 Tuning Starten

Im Fenster „Structure“ den Parameter „mode“ Doppelklicken und „ncSPEED“ auswählen. Danach wieder im Hauptfenster auf „Tuning starten“ doppelklicken.

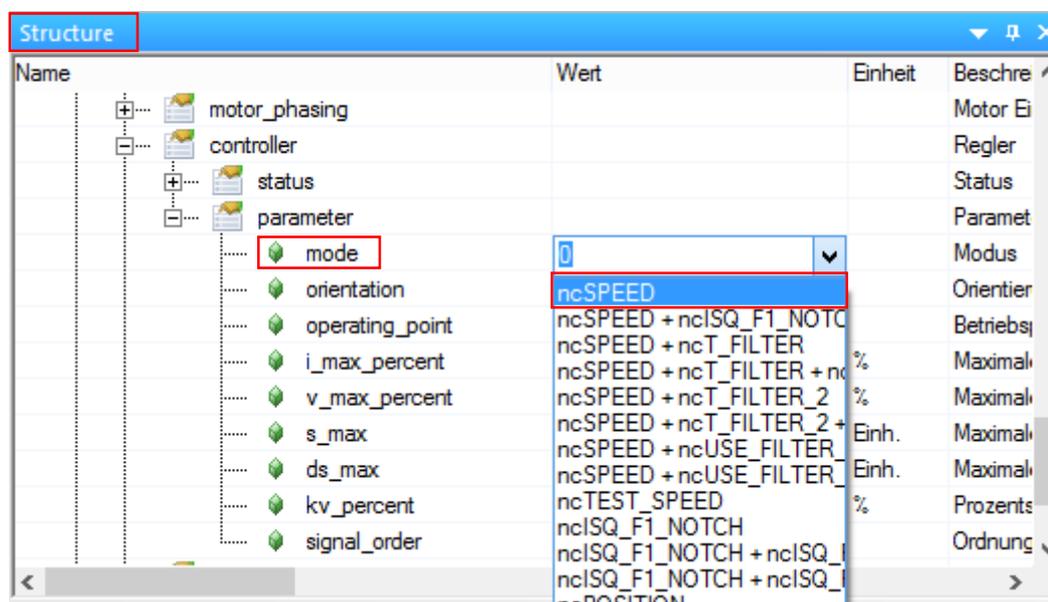


Abbildung 67 ncSPEED

Den Vorgang mit „ncPOSITION“ wiederholen.

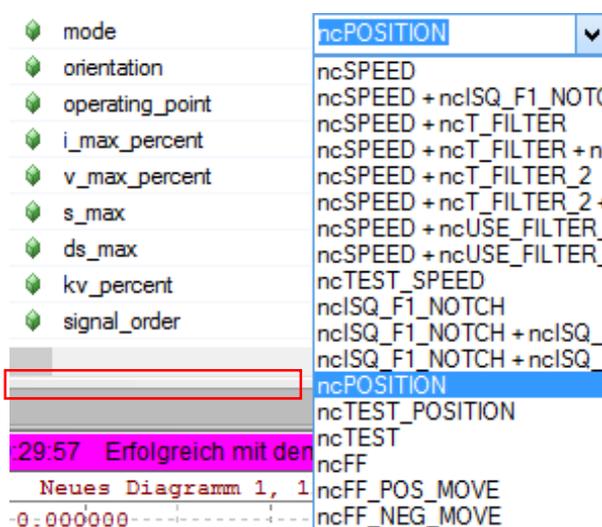


Abbildung 68 ncPOSITION

Nun wird im Hauptfenster unter „Vorbereitung“ → „Regler“ mit Doppelklick auf „Einschalten“ der Regler freigeben.

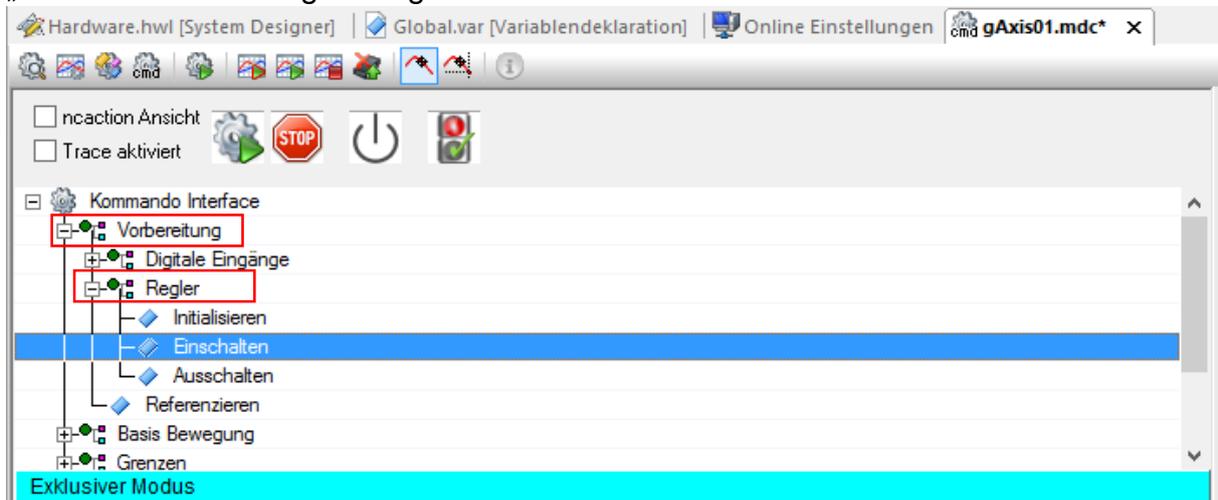


Abbildung 69 Regler Einschalten

Nun sind im Fenster „Watch“ die Symbole für Reglerfreigabe und für Tuning farbig hervorgehoben.



Abbildung 70 Statusmeldungen

Der Antrieb ist nun einsatzbereit.

### 8.7 Basis Bewegung

Jetzt kann der Motor bei Bedarf getestet werden. Im Hauptfenster unter „Basis Bewegung“ können „Absolut, Additiv, Positiv, Negativ, Stop“ ausgewählt werden.

Absolut: Der Motor fährt den eingestellten Wert an.

Additiv: Addiert den eingestellten Wert auf den Istwert

Positiv: Rechtslauf des Motors

Negativ: Linkslauf des Motors

Stop: Stoppt den Motor

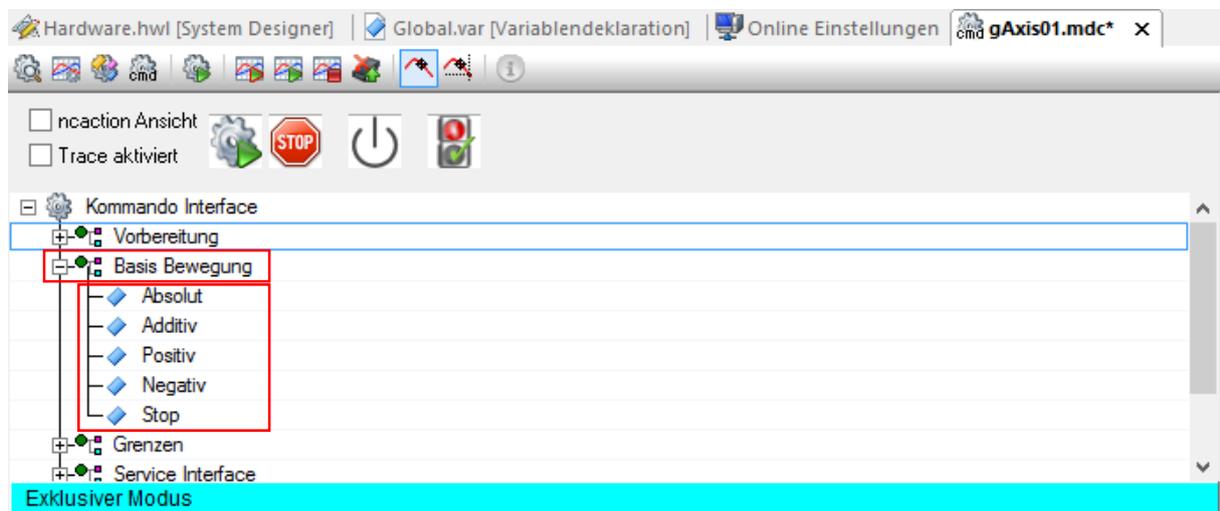


Abbildung 71 Motor Grundbewegung

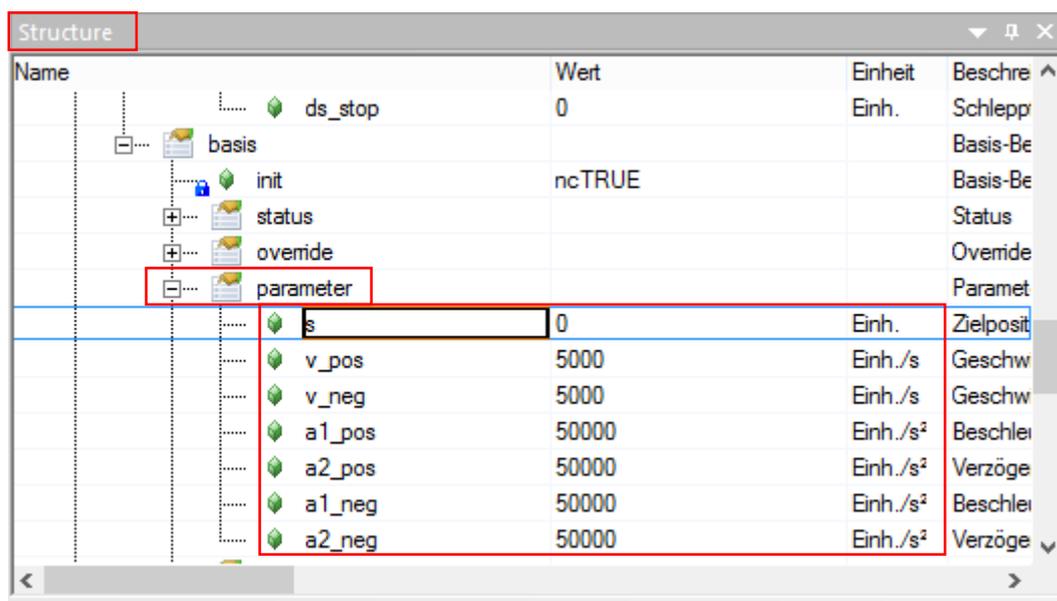
Im „Structure“ Fenster können die gewünschten Werte unter Parameter eingetragen werden und mit Doppelklick auf den jeweiligen Modus im Hauptfenster ausgeführt werden.

s = Strecke gewünschte Zielposition

v\_pos / v\_neg = Geschwindigkeit

a1\_pos / a1\_neg = Hochlauframpe in jeweilige Richtungen

a2\_pos / a2\_neg = Tieflauframpe in jeweilige Richtungen



Name	Wert	Einheit	Beschre
ds_stop	0	Ein.	Schlepp
basis			Basis-Be
init	ncTRUE		Basis-Be
status			Status
override			Override
parameter			Paramet
s	0	Ein.	Zielposit
v_pos	5000	Ein./s	Geschw
v_neg	5000	Ein./s	Geschw
a1_pos	50000	Ein./s <sup>2</sup>	Beschlei
a2_pos	50000	Ein./s <sup>2</sup>	Verzüge
a1_neg	50000	Ein./s <sup>2</sup>	Beschlei
a2_neg	50000	Ein./s <sup>2</sup>	Verzüge

Abbildung 72 Parameter

Nun müssen die Motorparameter gespeichert werden (Dies ist unter dem Kapitel 8 unter dem Punkt 8.3 zu finden). Ist die Speicherung abgeschlossen wird das Testfenster geschlossen und auf das Zielsystem neu übertragen (Dies ist unter dem Kapitel 8 unter dem Punkt 8.4 zu finden).

## 9. Automatische Variablen Deklaration

Damit die Variablen automatisch deklariert werden, müssen folgende Einstellungen vorgenommen werden.

In der Taskleiste auf „Extras“ und den Unterpunkt „Optionen...“ aufrufen.

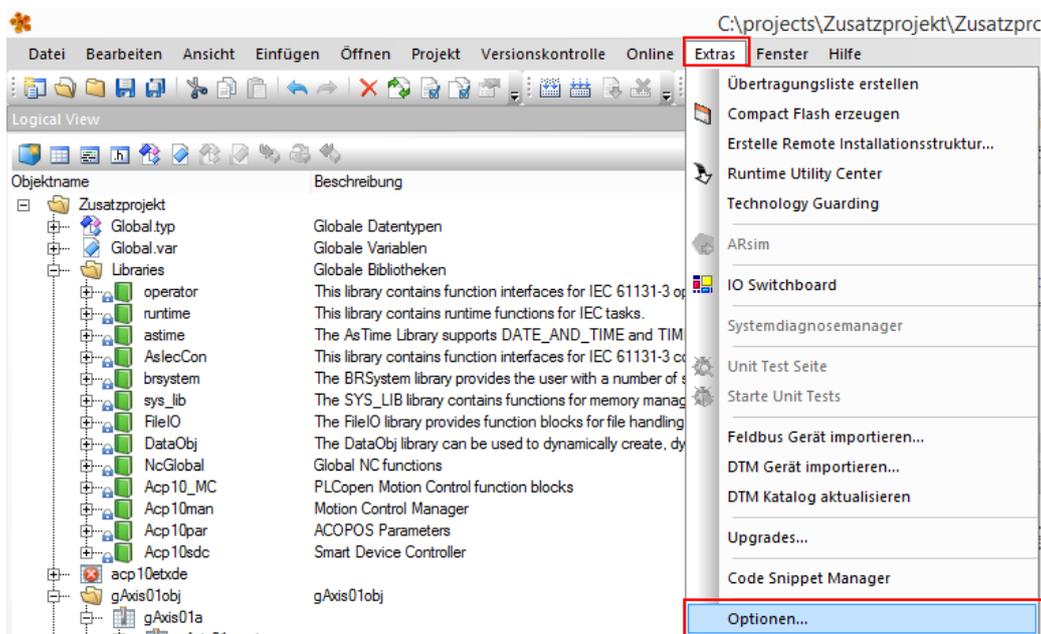


Abbildung 73 Variablen Deklaration

Es öffnet sich das Fenster „Optionen“, hier den Reiter „SmartEdit“ auswählen und einen Hacken bei „Automatische Deklaration neuer Variablen“ setzen und mit „OK“ bestätigen.

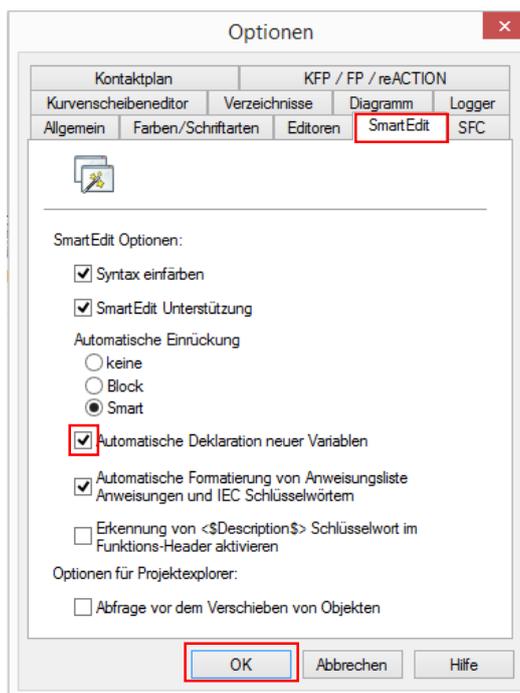


Abbildung 74 Automatische Deklaration

## 10. Programmierung

### 10.1 Programm Anlegen

Um ein Programm anzulegen wird im „Logical View“ der Projektordner markiert. In diesem Fall ist der Name des Projektordners „Zusatzprojekt“.

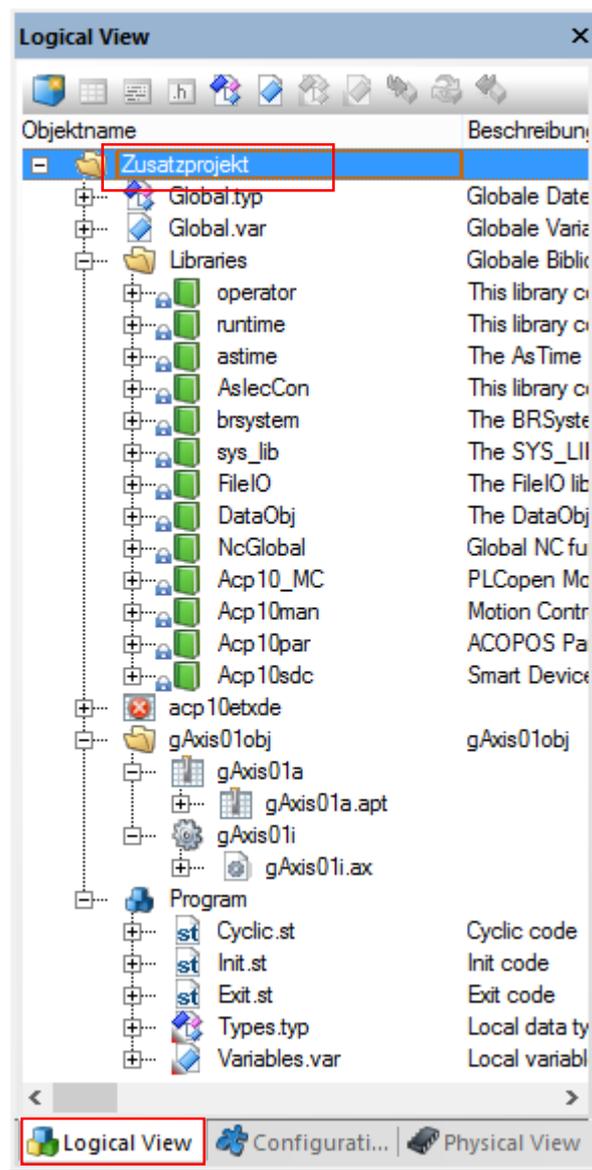


Abbildung 75 Projekt Ordner

Nun im „Toolbox – Objektkatalog“ den Punkt „Programm“ anwählen. Hier können verschiedene Programmiersprachen ausgewählt werden. Wir wählen mit Doppelklick „ST Programm“ (Strukturierter Text).

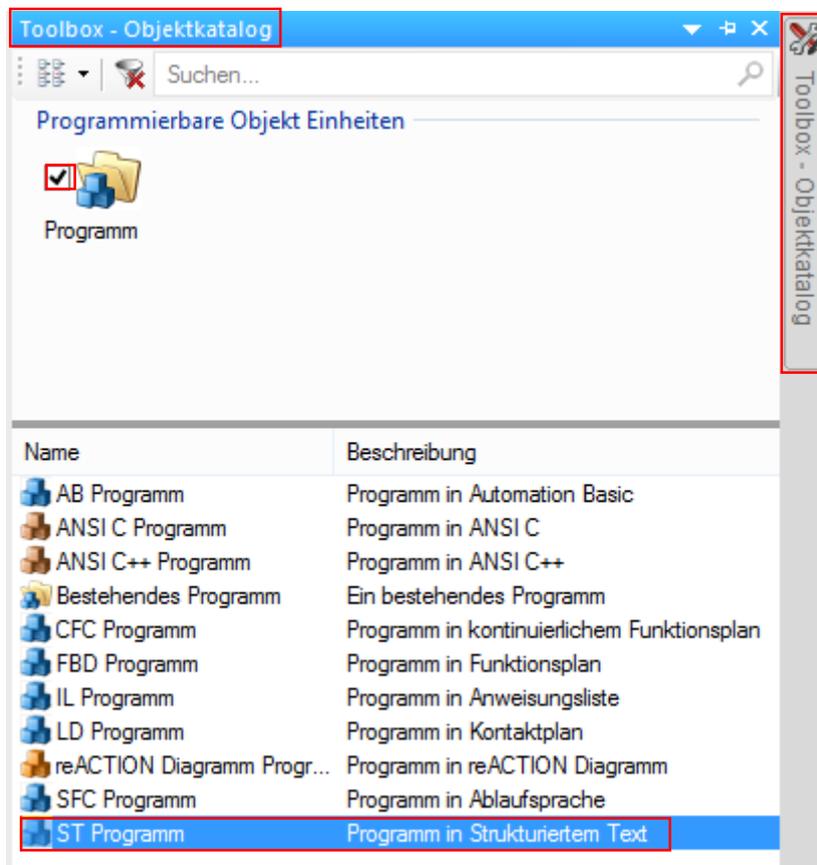


Abbildung 76 ST Programm

Jetzt ist im „Logical View“ ein neuer Ordner Namens „Program“ angelegt worden, darin wird das ganze Programm geschrieben.

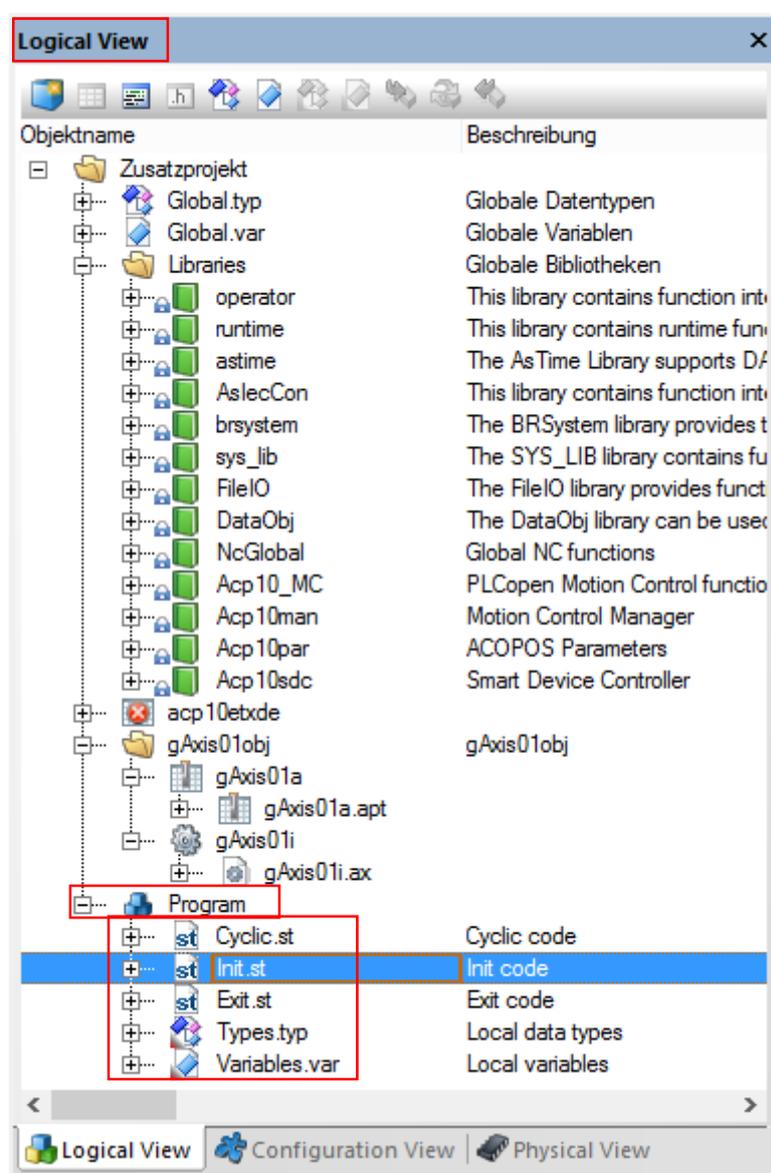


Abbildung 77 Aufteilung im Program

Cyclic.st = Kontinuierlich ablaufende Programm.

Init.st = Wird nur beim ersten Start des Programms ausgeführt.

Exit.st = Wird beim Beenden des Programms ausgeführt.

Types.type = Strukturierung der Variablen.

Variables.var = Für Lokale Variablen.

## 10.2 Programm Initialisierung

Durch Doppelklick auf „Init.st“ wird das Fenster geöffnet, wo die Variablen angelegt werden.

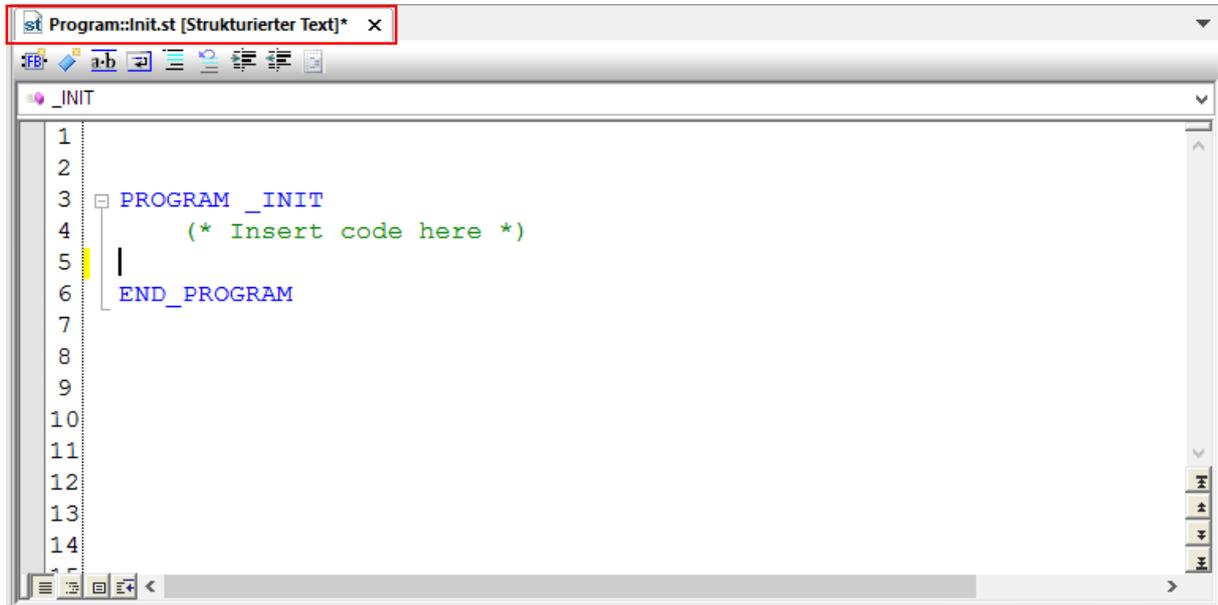


Abbildung 78 Init.st

Bei jeder neuen Variable erscheint bei drücken auf die Entertaste das Fenster „Variablendeklaration“.

Hier muss der „Typ“ und eventuell der „Wert“ von jeder Variablen angepasst werden.

Hier ein Beispiel:

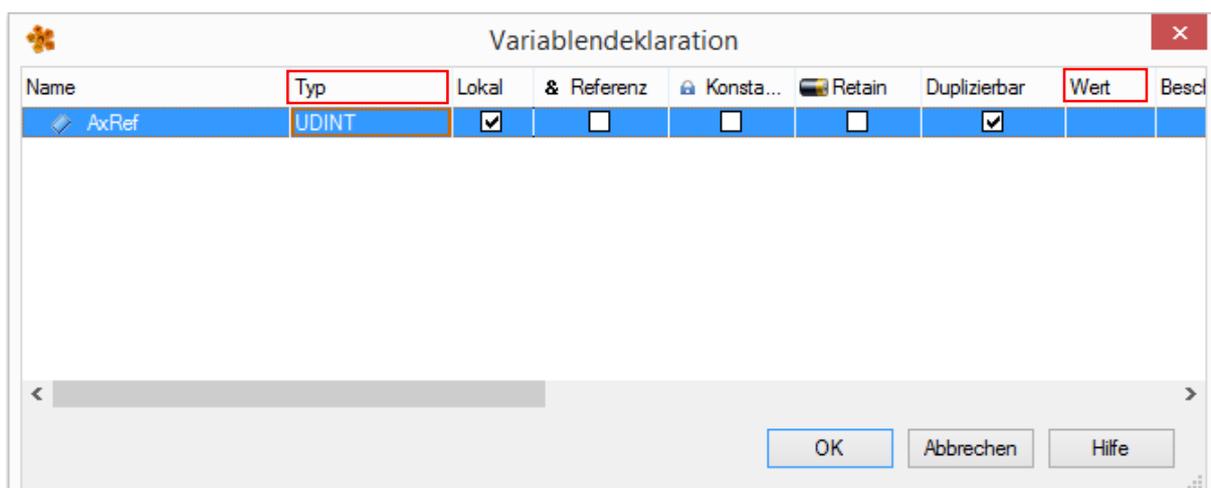


Abbildung 79 Variablendeklaration

Das ist der Programmcode für das Init:

```
AxRef:= ADR(gAxis01);//TYP:UDINT  
//Verknüpfung der Achse mit einer lokalen Variable  
setPower:= FALSE; //TYP:BOOL  
//Variable zum einschalten des Reglers (kann auch in "Variables.var" eingestellt  
werden)  
sethoming:=TRUE; //TYP:BOOL  
//Variable zum Setzen der Referenz (kann auch in "Variables.var" eingestellt werden)
```

Ist die Variablen Deklaration für die Initialisierung abgeschlossen muss dies gespeichert werden.

### 10.3 Programm Ablauf

Durch Doppelklick auf „Cyclic.st“ wird das Fenster, in dem das Ablaufprogramm geschrieben wird, geöffnet.

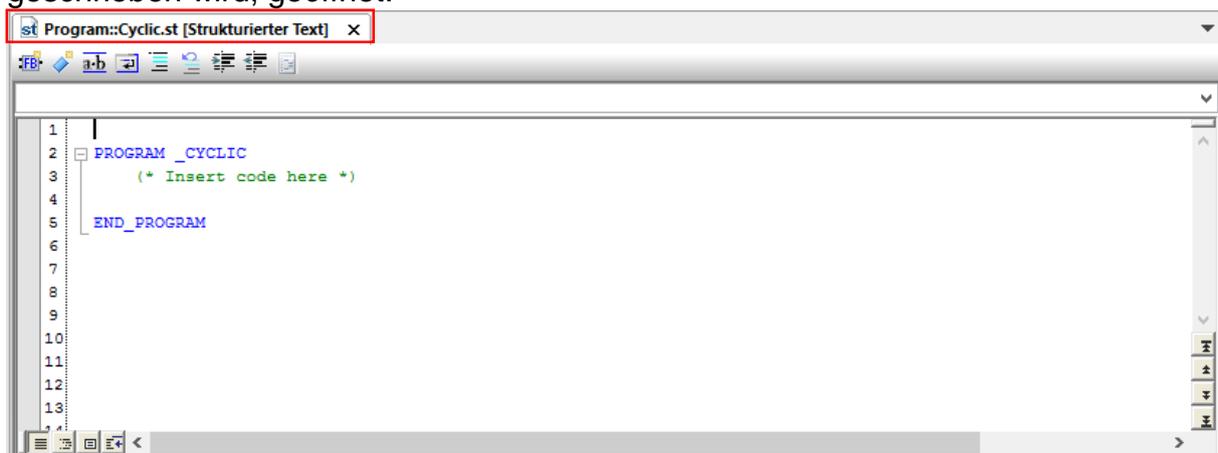


Abbildung 80 Ablaufprogramm

Um das Programmieren zu erleichtern, können fertige Funktionsblöcke ausgewählt und ins Programm eingefügt werden.

Dazu muss „Funktion/Funktionsblock einfügen...“ angeklickt werden.

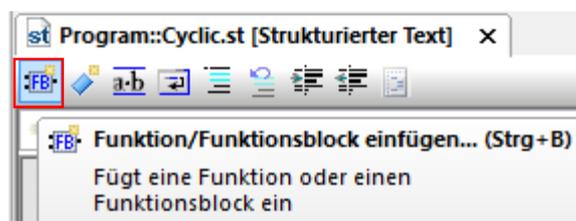


Abbildung 81 Funktionsblock

Es öffnet sich das Fenster „Datentyp auswählen“, im Filterfeld wird der gewünschte Bausteinname eingegeben und mit Doppelklick auf den Baustein übernommen.

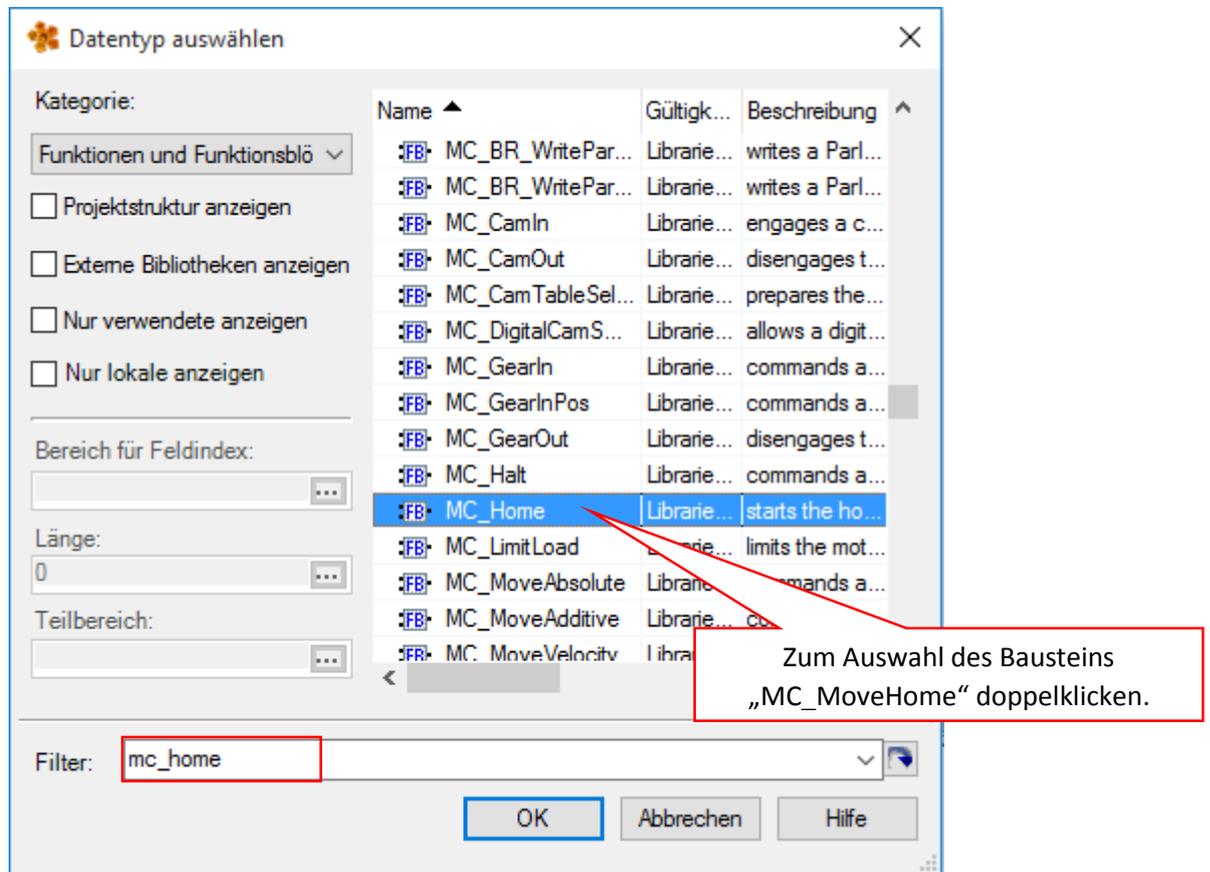


Abbildung 82 Auswahl Funktionsblock

mc\_Home = Referenzieren der Achse

Nach Auswahl von „MC\_Home“ erscheint der Strukturierte Text im Programm und es kann der jeweilige Variable oder Wert nach „:=“ eingegeben werden.

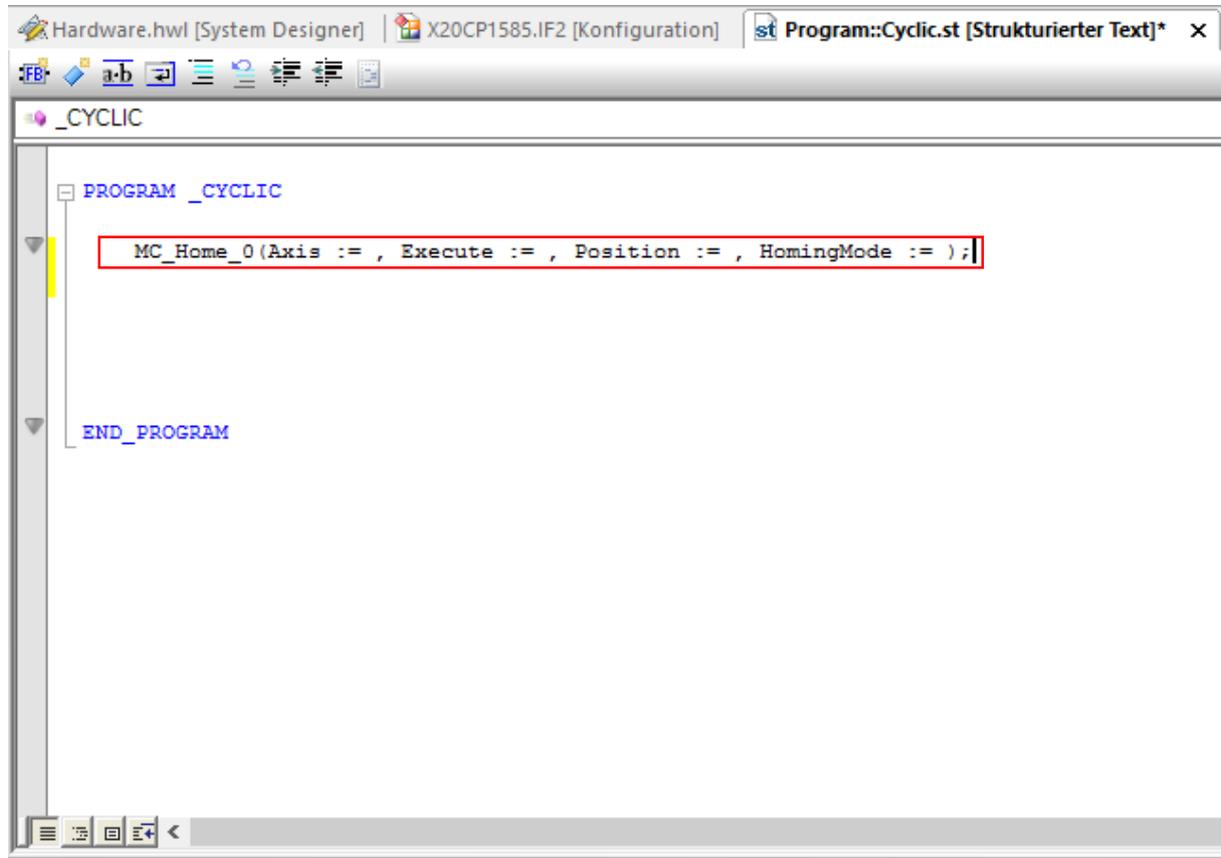


Abbildung 83 MC\_MoveHome Baustein

Hinweis: Zum schnelleren Programmieren, kann die Autofillfunktion mit der Tastenkombination „Strg+Leertaste“ genutzt werden.

Hier der Strukturierte Text für das Programm:

**1** = In dem Feld mit der Gelb Hinterlegten Fläche muss der wert rein geschrieben werden, den man sich beim Referenzieren auf einem Zettel notiert hat. Dieser Wert muss wieder mit einem negativen Vorzeichen versehen werden. Da dieser Befehl die Grundposition anfährt.

```
MC_Home_0(Axis := AxRef, Execute := sethoming , Position := -4 172 085 ,  
HomingMode := mcHOME_ABSOLUTE );  
//Achse referenzieren  
//der Wert "Position", der vorher aus dem NC Test Fenster ermittelt wurde, muss hier  
eingetragen werden.
```

```
IF MC_Home_0.Done THEN sethoming:=FALSE;  
// Variable sethoming auf FALSE setzen.  
END_IF  
IF MC_Home_0.Done THEN setPower:=TRUE;  
END_IF
```

```
MC_Power_0(Axis :=AxRef , Enable := setPower);  
//Regler einschalten mit Variable "setPower"  
MC_MoveAbsolute_0(Axis :=AxRef , Execute :=Move0 , Position :=0 , Velocity  
:=5000 , Acceleration :=50000 , Deceleration :=50000 , Direction :=  
mcSHORTEST_WAY );  
//Bewegungsbefehl auf Absolutposition 0 (Grundstellung).  
IF MC_MoveAbsolute_0.Done THEN Move0:= FALSE;  
END_IF
```

```
MC_MoveAdditive_0(Axis :=AxRef , Execute := Add90 , Distance :=-900 , Velocity  
:=500 , Acceleration :=50000 , Deceleration :=50000 );  
//Bewegungsbefehl, drehe um 90° nach links. Befehl wird durch Variable Add90  
gesetzt.  
IF MC_MoveAdditive_0.Done THEN Add90:= FALSE;  
END_IF
```

```
MC_MoveAbsolute_1(Axis :=AxRef , Execute := Absolut270 , Position :=900 ,  
Velocity :=1000 , Acceleration := 50000 , Deceleration := 50000 , Direction  
:=mcPOSITIVE_DIR );  
//Bewegungsbefehl, fahre rechts auf 270°, Ausführung durch Variable Absolut270  
IF MC_MoveAbsolute_1.Done THEN Absolut270:=FALSE;  
END_I
```

Während des schreiben des Programmes wird jede neue Variable in der Variablen Tabelle deklariert.

Nach der Variablendeklaration sollte die Tabelle bei „Variables.var“ diese Variablen beinhalten.

Name	Typ	& Referenz	Konstante	Retain	Duplizierbar	Wert
Absolut270	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Add90	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
AxRef	UDINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_Home_0	FUB_TYPE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAbsolute_0	type	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAbsolute_1	FUB_TYPE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAdditive_0	FUB_TYPE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_Power_0	FUB_TYPE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Move0	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
sethoming	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
setPower	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Abbildung 84 Variablen

Jetzt müssen die Typen „FUB\_TYPE“ und „type“ umbenannt werden. Die Typen werden nach dem jeweiligen Funktionsbaustein benannt. Dieses ist in Abbildung 85 zu sehen.

Name	Typ	& Referenz	Konstante	Retain	Duplizierbar	Wert
Absolut270	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Add90	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
AxRef	UDINT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_Home_0	MC_Home	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAbsolute_0	MC_MoveAbsolute	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAbsolute_1	MC_MoveAbsolute	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_MoveAdditive_0	MC_MoveAdditive	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
MC_Power_0	MC_Power	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
Move0	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
sethoming	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
setPower	BOOL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

Abbildung 85 Variable Typ ändern

Das Projekt muss nun über „Alles Speichern“ (doppelte Diskette) gespeichert werden.

Nun muss das Projekt In das Zielsystem übertragen werden. (Dies ist in Kapitel 8 unter dem Punkt 8.4 beschrieben)

### 10.4 Programmtest über die Watch Funktion

Dazu wird im „Logical View“ mit Rechtsklick das „Programm“ → „Öffnen“ und „Watch“ ausgewählt.

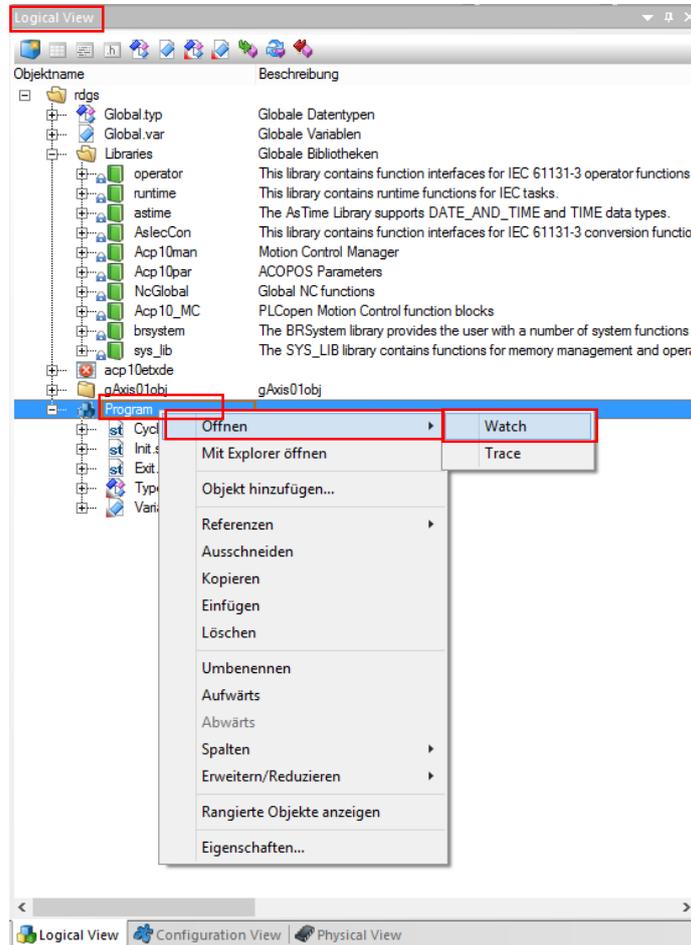


Abbildung 86 Programm Test

Es öffnet sich die Registerkarte „Program::pvm[Watch]“ hier können die zu beobachten Variablen hinzugefügt werden.

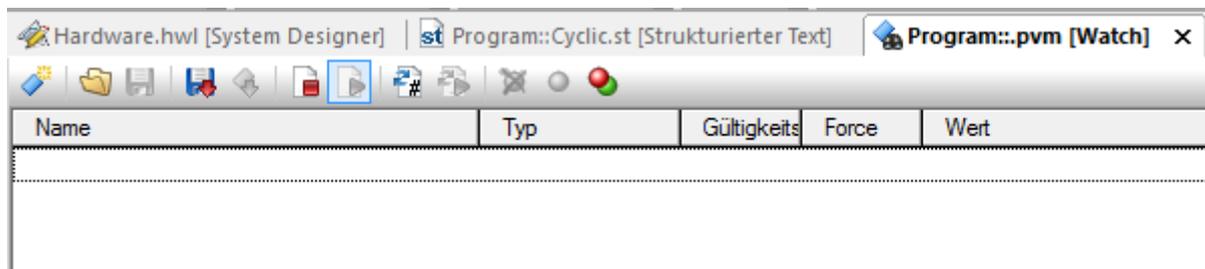


Abbildung 87 Variablen Test

Es öffnet sich das Fenster „Variable einfügen“, hier müssen die drei Variablen „Absolut270, Add90, Move0“ markiert werden. Um diese Variablen zu testen müssen diese noch mit „Hinzufügen“ eingebunden.

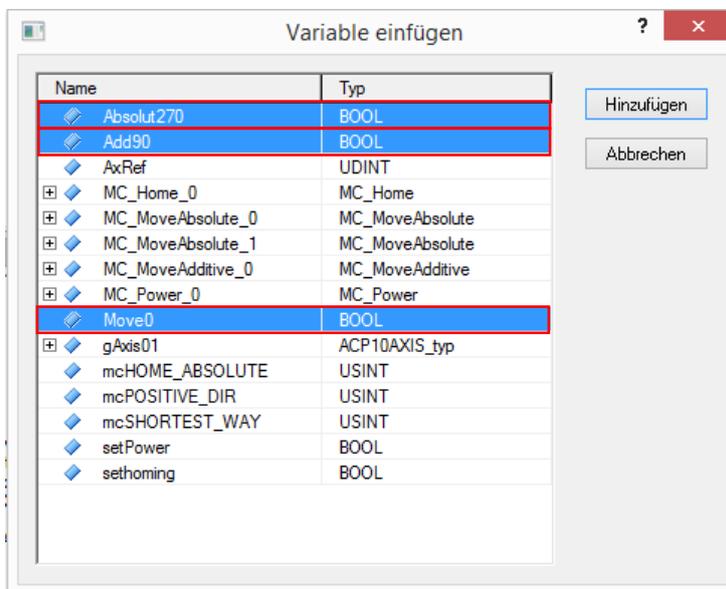


Abbildung 88 Variablen Hinzufügen

Die ausgewählten Variablen erscheinen im Fenster „Program::pvm [Watch]“. Über die Spalte „Wert“ wird der aktuelle Status der Variable angezeigt, dieser kann auch überschrieben werden um manuell einen Befehl zu senden. Die Variablen können durch ein verändern des Wertes von „FALSE“ auf „TRUE“ gesetzt werden.

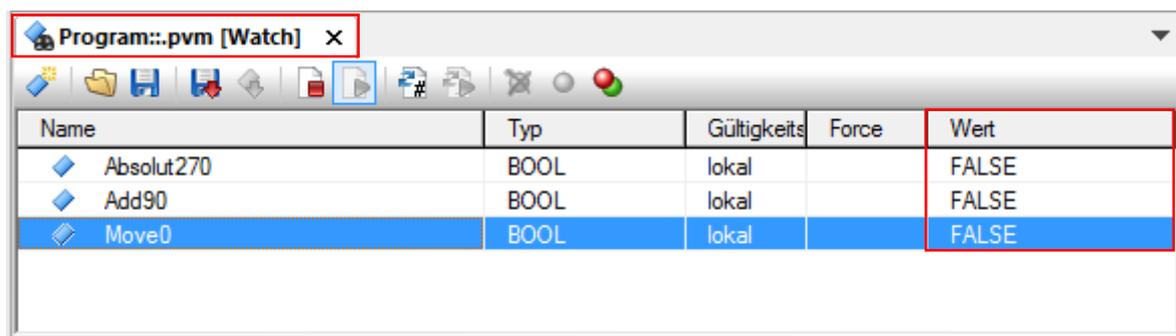


Abbildung 89 Variablen manuell Setzen

Als erstes muss der „Wert“ von „Move0“ auf „True“ gesetzt werden, um auf die Grundposition zu fahren. Danach kann man mit dem Befehl „Absolut270“, den Motor auf die Position 270° rechtsrum fahren lassen und mit dem Befehl „Add90“ dreht der Motor um 90° nach links.

## 11. Externe Schalter dem Programm zuweisen

Hierfür im „Hardware.hwl“ die erste Digitale Input Baugruppe auswählen.

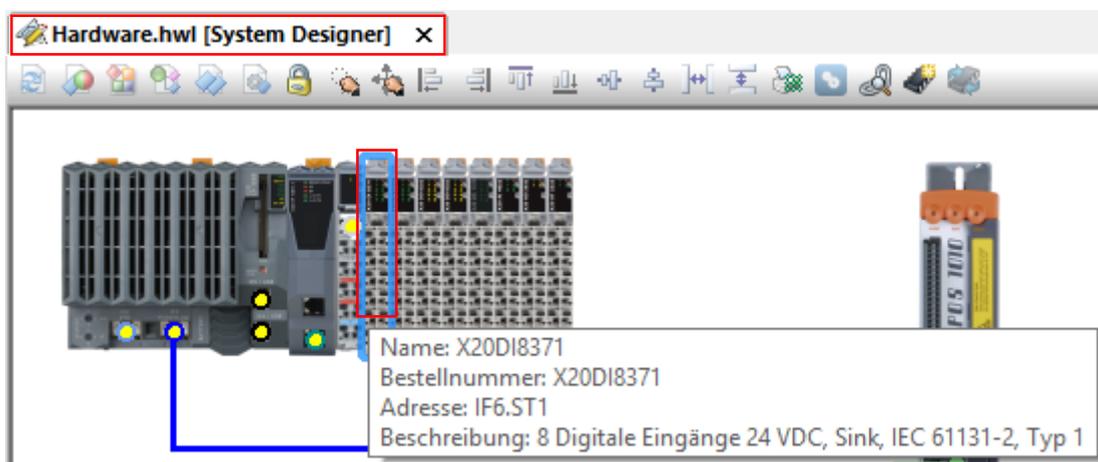


Abbildung 90 Baugruppe Markieren

Danach mit rechtsklick auf die entsprechende Baugruppe klicken und den Unterpunkt „I/O Zuordnung“ anklicken.

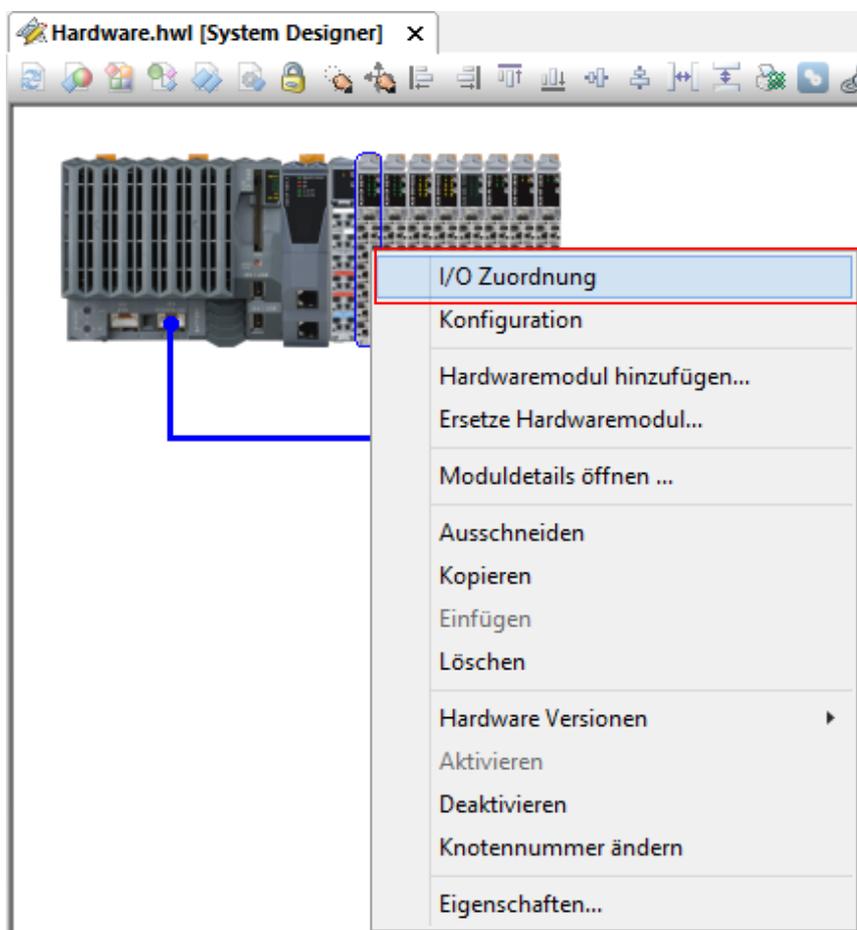


Abbildung 91 I/O Zuordnung

Nun erscheint der Reiter „X20DI8371“ mit den 8 Eingängen.

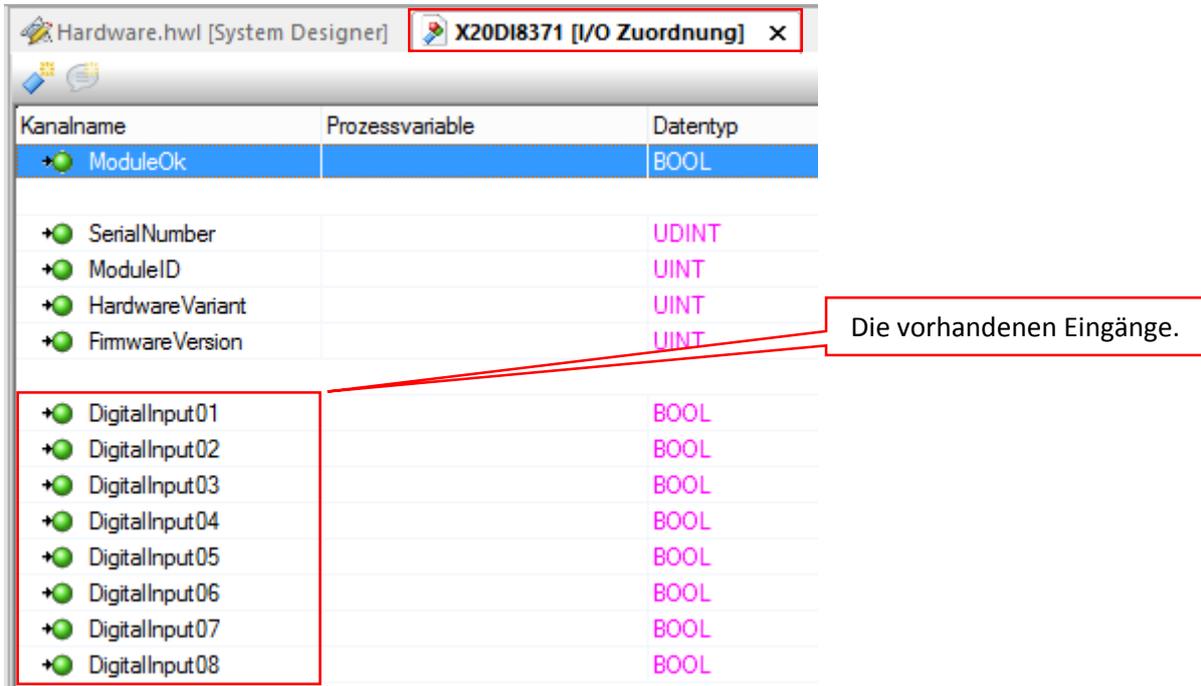


Abbildung 92 Digitale Eingänge einer Baugruppe

Es kann der gewünschte Eingang ausgewählt werden und durch Doppelklick auf „Prozessvariable“ und dann durch Klick auf „...“ eine Variable vergeben werden.

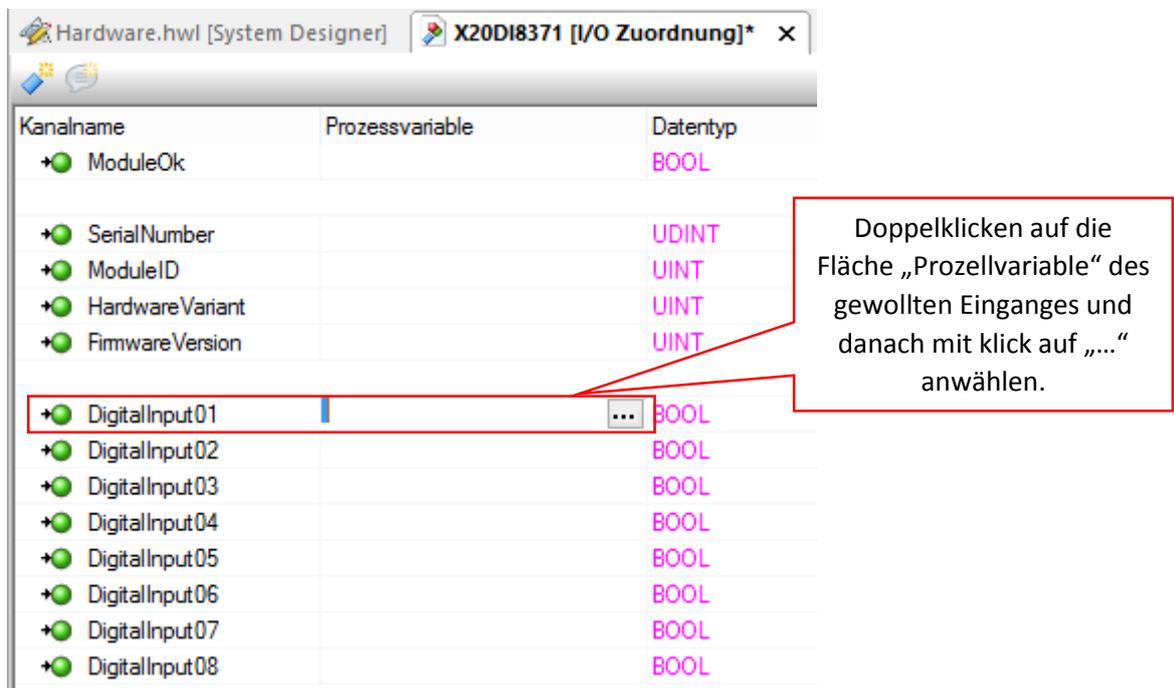


Abbildung 93 Variable Zuweisen

Nun öffnet sich das Fenster „Auswahl“ hier werden die erstellten Variablen unter „Program“ aufgelistet. Diese Variablen können durch einen Doppelklick auf den Gewünschten Eingang vergeben werden.

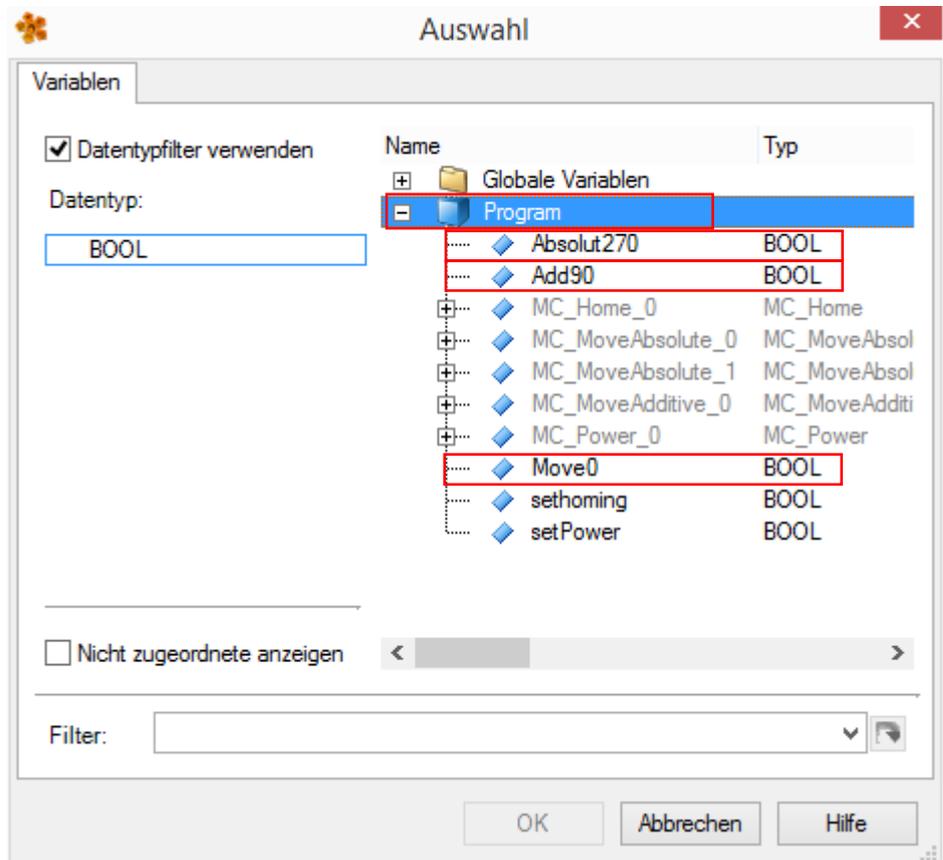
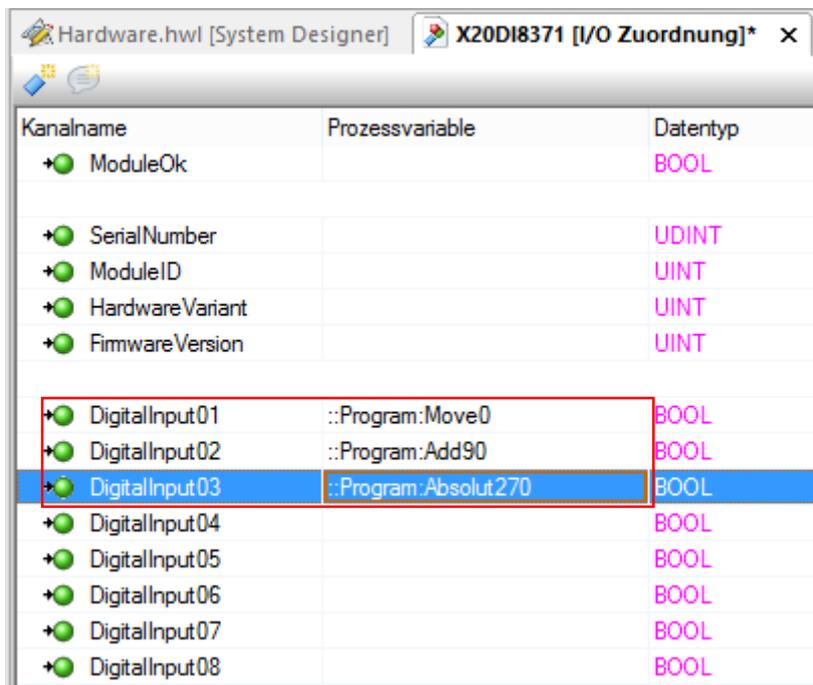


Abbildung 94 Variablen Vergabe

So kann die Zuweisung der Eingänge aussehen.



Kanalname	Prozessvariable	Datentyp
+● ModuleOk		BOOL
+● SerialNumber		UDINT
+● ModuleID		UINT
+● HardwareVariant		UINT
+● FirmwareVersion		UINT
+● DigitalInput01	::Program:Move0	BOOL
+● DigitalInput02	::Program:Add90	BOOL
+● DigitalInput03	::Program:Absolut.270	BOOL
+● DigitalInput04		BOOL
+● DigitalInput05		BOOL
+● DigitalInput06		BOOL
+● DigitalInput07		BOOL
+● DigitalInput08		BOOL

Abbildung 95 Zuweisung Eingänge

Jetzt alle Daten speichern (doppelte Diskette) und auf das Zielsystem übertragen.  
(Dies ist im Kapitel 8 unter dem Punkt 8.3 beschrieben)

## 12. Abbildungsverzeichnis

Abbildung 1 Automation Studio Oberfläche.....	2
Abbildung 2 Leeres Projekt.....	2
Abbildung 3 Projektassistent.....	3
Abbildung 4 Projektassistent Hardware Auswahl.....	4
Abbildung 5 Einbinden der CPU.....	5
Abbildung 6 System Designer.....	5
Abbildung 7 Toolbox-Hardware Katalog.....	6
Abbildung 8 Einbindung Interface RT Controller.....	6
Abbildung 9 Einbinden der Baugruppen.....	7
Abbildung 10 Fertige Hardware.....	7
Abbildung 11 Acopos Auswahl.....	8
Abbildung 12 Acopos Konfiguration.....	9
Abbildung 13 Auswahl des Einsteckmoduls.....	10
Abbildung 14 SS3.....	11
Abbildung 15 Motor Konfiguration.....	12
Abbildung 16 Motor System Wahl.....	13
Abbildung 17 Sprache Fehlermeldungen.....	14
Abbildung 18 Achse Konfigurieren.....	15
Abbildung 19 Achsen Konfiguration 2.....	16
Abbildung 20 Antriebskonfiguration Fertigstellen.....	17
Abbildung 21 Hardware ohne Motor.....	18
Abbildung 22 Hardware mit Motor.....	18
Abbildung 23 B&R Internetseite.....	19
Abbildung 24 Link Download.....	19
Abbildung 25 Download Datei.....	20
Abbildung 26 Upgrades.....	20
Abbildung 27 Lokal.....	21
Abbildung 28 Upgrade Installieren.....	22
Abbildung 29 Installationen starten.....	23
Abbildung 30 Installation Fertig.....	23
Abbildung 31 Motor manuell einbinden.....	24
Abbildung 32 Hardware komplett.....	24
Abbildung 33 Motor Variable.....	25
Abbildung 34 Endat Verbindung.....	25
Abbildung 35 ETH Konfiguration.....	26
Abbildung 36 ETH Modus.....	26
Abbildung 37 IP Adresse.....	27
Abbildung 38 Physical View Acopos.....	28
Abbildung 39 Knotennummer.....	28
Abbildung 40 Ausgeschaltete SPS.....	29
Abbildung 41 Compact-Flash.....	29
Abbildung 42 Compact-Flash erstellen.....	30
Abbildung 43 Runtime Utility Center.....	30
Abbildung 44 Compact Flash Erstellt.....	30
Abbildung 45 Online.....	31

Abbildung 46 Online Einstellungen .....	31
Abbildung 47 Angeschlossene SPS .....	32
Abbildung 48 Verbinden mit der SPS .....	32
Abbildung 49 Statusanzeige .....	32
Abbildung 50 Test öffnen.....	33
Abbildung 51 Achsen Auswahl .....	34
Abbildung 52 Exklusiver Modus .....	34
Abbildung 53 Einheiten pro Motorumdrehung.....	35
Abbildung 54 Parameter Speichern .....	36
Abbildung 55 Parameter Speicherort .....	36
Abbildung 56 Zielsystem übertragen .....	37
Abbildung 57 Kompilieren .....	37
Abbildung 58 Infos zur Übertragung des Projektes.....	38
Abbildung 59 Projekt übertragen.....	38
Abbildung 60 Structure Fenster.....	39
Abbildung 61 Grundposition.....	40
Abbildung 62 monitor.s .....	40
Abbildung 63 Parameter „s“ .....	41
Abbildung 64 Referenzieren der „null“ Position .....	41
Abbildung 65 Watch Fenster.....	42
Abbildung 66 Tuning Starten .....	42
Abbildung 67 ncSPEEd .....	43
Abbildung 68 ncPOSITION .....	43
Abbildung 69 Regler Einschalten .....	44
Abbildung 70 Statusmeldungen .....	44
Abbildung 71 Motor Grundbewegung.....	45
Abbildung 72 Parameter .....	46
Abbildung 73 Variablen Deklaration .....	47
Abbildung 74 Automatische Deklaration .....	47
Abbildung 75 Projekt Ordner.....	48
Abbildung 76 ST Programm.....	49
Abbildung 77 Aufteilung im Program .....	50
Abbildung 78 Init.st.....	51
Abbildung 79 Variablendeklaration .....	51
Abbildung 80 Ablaufprogramm .....	52
Abbildung 81 Funktionsblock.....	52
Abbildung 82 Auswahl Funktionsblock.....	53
Abbildung 83 MC_MoveHome Baustein .....	54
Abbildung 84 Variablen.....	56
Abbildung 85 Variable Typ ändern.....	56
Abbildung 86 Programm Test .....	57
Abbildung 87 Variablen Test.....	57
Abbildung 88 Variablen Hinzufügen .....	58
Abbildung 89 Variablen manuell Setzen .....	58
Abbildung 90 Baugruppe Markieren .....	59
Abbildung 91 I/O Zuordnung.....	59
Abbildung 92 Digitale Eingänge einer Baugruppe.....	60

Abbildung 93 Variable Zuweisen.....	60
Abbildung 94 Variablen Vergabe .....	61
Abbildung 95 Zuweisung Eingänge .....	62

## 13. Komponenten

### Software:

Automation Studio 4.2.2.134

### Hardware:

Komponenten der SPS:

X20CP1585:

X20 CPU ATOM, 1.0GHz, POWERLINK, 1x IF

X20IF10E1-1:

X20 Interface PROFINET RT Master (DTM)

X20DI8371:

8 Digitale Eingänge 24 VDC, Sink, IEC 61131-2, Typ 1

X20DO8322:

8 Ausgänge 24 VDC / 0,5 A, Source

X20DS1319:

Digital Signal, 24V, SSI, ABR/PR Emulation

X20AI4622:

4 Eingänge  $\pm 10$  V / 0 bis 20 mA 12 Bit

X20AO2632:

2 Ausgänge  $\pm 10$  V / 0 bis 20 mA

X20AT2222:

2 Widerstands Temperatur Eingänge

8V1010.50-2:

ACOPOS Servoverstärker, 3x 110-230 V / 1x 110-230 V, 2,3 A, 0,45 kW, Netzfilter, Bremswiderstand und...

- 8AC114.60-2:

ACOPOS Einsteckmodul, POWERLINK V2 Interface

- 8AC126.60-1:

ACOPOS Einsteckmodul, EnDat 2.2 Geber Interface

8LVA22.B1015D000-0:

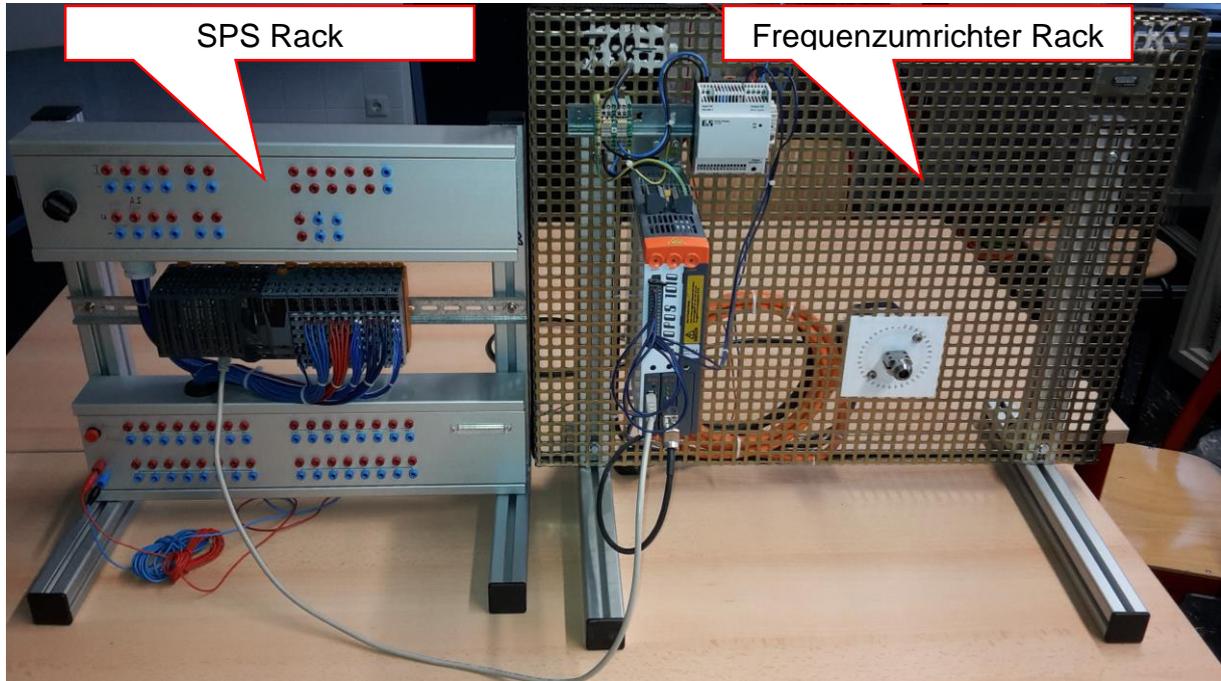
Synchron-Motor; Polpaarzahl: 4 Paar; Nenndrehzahl: 1.500 min<sup>-1</sup> Stillstandsmoment: 0,680 Nm; ohne Bremse; glatte Welle; Stecker gewinkelt (drehbar);

Zentrierdurchmesser 50 mm; Passung h7 selbstgekühlt, Bauform A; Schutzklasse

IP54; ohne Wellendichtring 80 VDC; Haltemoment der Bremse: 0,00 Nm; B1 2.2 Bat

ind. 16Str. multiti.

## 14. Aufbau der Racks



Das Rack mit der SPS, welches auf der linken Seite zu sehen ist, muss mit 24V versorgt werden.

Das Rack mit dem Frequenzumrichter, welches auf der rechten Seite zu sehen ist, wird mit 230V versorgt.

Auf dem Rack mit dem Frequenzumrichter ist eine 24V Versorgungsquelle für die SPS vorhanden.