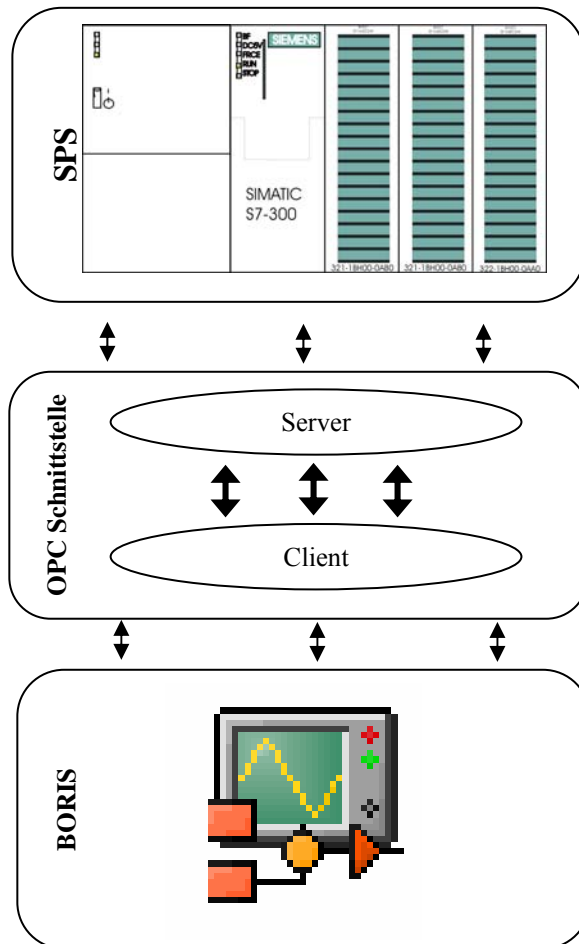


# WinFACT-SPS-Kopplung über OPC



**Kommunikation von BORIS und  
Speicherprogrammierbaren Steuerungen  
via OPC (OLE for Process Control)**

Rev. 1.0

Ingenieurbüro Dr. Kahlert

---

## Inhaltsverzeichnis

---

<b>Inhaltsverzeichnis</b>	<b>2</b>
<b>1. Einleitung</b>	<b>3</b>
1.1 Was steckt hinter OPC?	3
1.2 Was brauche ich konkret?	3
1.3 Wo bekomme ich alles her?	3
<b>2. Einrichtung des OPC-Servers</b>	<b>4</b>
2.1 Vorbemerkungen	4
2.2 SOFTING OPC-Server	5
2.2.1 Konfigurierung des SOFTING OPC-Servers	5
2.2.2 Erstellung der Aliasdatei	6
2.2.3 Starten des Softing OPC- Servers	7
2.3 IBH OPC-Server	8
2.3.1 Konfigurierung des IBH OPC-Servers	8
2.3.2 Starten des IBH OPC-Servers	10
<b>3. Der BORIS-OPC-Client</b>	<b>10</b>
3.1 OPC-Client zum Server verbinden	11
3.2 OPC-Client-Tag-Wahl	11
<b>4. Beispiel</b>	<b>12</b>

---

## 1. Einleitung

---

Die vorliegende Dokumentation beschreibt die Kopplung zwischen dem blockorientierten Simulationssystem BORIS und einer speicherprogrammierbaren Steuerung (SPS) über OPC (OLE für Process Control, siehe [www.opc-foundation.org](http://www.opc-foundation.org)). Diese Kopplung kann z. B. benutzt werden, um Messdaten aus einer SPS in BORIS weiterzuverarbeiten, einen realen Prozess über BORIS zu steuern oder regeln oder aber um beispielsweise für Ausbildungszwecke ein Software-Prozessmodell über eine SPS zu steuern. Für letzteren Anwendungsfall enthält Kapitel 4 ein Beispiel. Die Kopplung wird beispielhaft an einer SPS vom Typ Siemens S7 mit OPC-Servern der Firmen SOFTING bzw. IBHSoftec.

### 1.1 Was steckt hinter OPC?

OPC ist eine herstellerübergreifende Schnittstelle zur Kommunikation zwischen Software-Applikationen, die auf dem DCOM-Standard von Microsoft basiert. Die OPC-Schnittstelle besitzt eine Client/Server-Architektur, bei der Client und Server auch auf unterschiedlichen Rechnern im Netz laufen können. Das blockorientierte Simulationssystem BORIS kann prinzipiell sowohl als OPC-Server als auch als OPC-Client arbeiten. Bei der OPC-Ankopplung an eine SPS stellt die SPS in der Regel den OPC-Server dar, während BORIS als OPC-Client arbeitet.

### 1.2 Was brauche ich konkret?

Um eine OPC-Verbindung zwischen BORIS und einer SPS herzustellen, benötigen Sie folgende „Minimalausstattung“:

- SPS (Speicherprogrammierbare Steuerung)
- OPC-Server zur entsprechenden SPS
- Blockorientiertes Simulationssystem BORIS (Kernmodul des Programmsystems WinFACT)
- OPC-Client-Toolbox zu BORIS

### 1.3 Wo bekomme ich alles her?

Nachfolgend wird beispielhaft die OPC-Ankopplung einer SPS vom Typ Siemens S7 über zwei unterschiedliche OPC-Server aufgezeigt:

- S7/S5 OPC-Server der Firma SOFTING (zeitbegrenzte Demo-Version verfügbar unter <http://www.softing.com>)
- S7 OPC-Server der Firma IBHSoftec (zeitbegrenzte Demo-Version verfügbar unter <http://www.ibhsoftec-sps.de>)

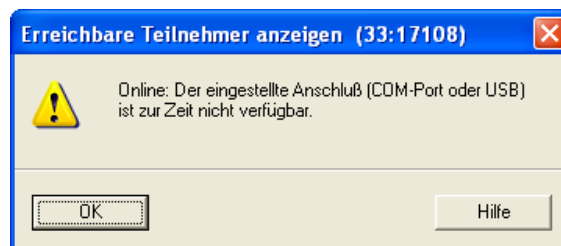
Es gibt eine Vielzahl weiterer Anbieter von OPC-Servern für speicherprogrammierbare Steuerungen; die Handhabung ist jeweils ähnlich wie bei den beiden beispielhaft ausgewählten Servertypen. Auch für speicherprogrammierbare Steuerungen anderer Hersteller werden selbstverständlich OPC-Server angeboten.

Demo-Versionen des Programmsystems WinFACT/BORIS sowie der zugehörigen OPC-Toolbox finden Sie im Downloadbereich unter [www.kahlert.com](http://www.kahlert.com).

## 2. Einrichtung des OPC-Servers

### 2.1 Vorbemerkungen

Die Kommunikation zwischen dem OPC-Server und der SPS findet in der Regel über dieselbe Schnittstelle statt, die auch zur Programmierung der SPS benutzt wird (z. B. die MPI-Schnittstelle). Während des RUN-Zustandes des OPC-Servers kann diese Schnittstelle daher von keiner Fremdsoftware (z.B. STEP7/STEP5) parallel verwendet werden. So ist es seitens STEP7 nicht möglich, währenddessen die Zustände der Operanden auszulesen oder ein überarbeitetes Programm aufzuspielen. Wird eine der folgenden Fehlermeldungen ausgegeben, so liegt dies also an dem laufenden OPC-Server:



Zuerst muss aber ein OPC-Server erst einmal zum Laufen gebracht werden. Die grundsätzliche Vorgehensweise dazu ist bei allen Servern gleich. Die Schnittstelle wird konfiguriert, die genutzte Steuerung ausgewählt und Variablen definiert. Eingänge, Ausgänge und Merker können aber auch absolut - also ohne speziellen Alias - vom OPC-Client angesprochen werden.

Die wichtigsten Adressen im Überblick

Eingangsbit:

*Ex.x*                      *E0.0*

Eingangsbyte:

*EBx*                        *EB1*

Ausgangsbit:

*Ax.x*                      *A0.0*

Eingangsbyte:

*ABx*                        *AB1*

Merkerbit:

*Mx.x*                      *M0.0*

Merkerbyte:

*MBx*                        *MB0*

Die nachfolgenden Abschnitte enthalten eine Konfigurationsanleitung für den SOFTING OPC-Server und den IBH OPC-Server.

## 2.2 SOFTING OPC-Server

### 2.2.1 Konfigurierung des SOFTING OPC-Servers

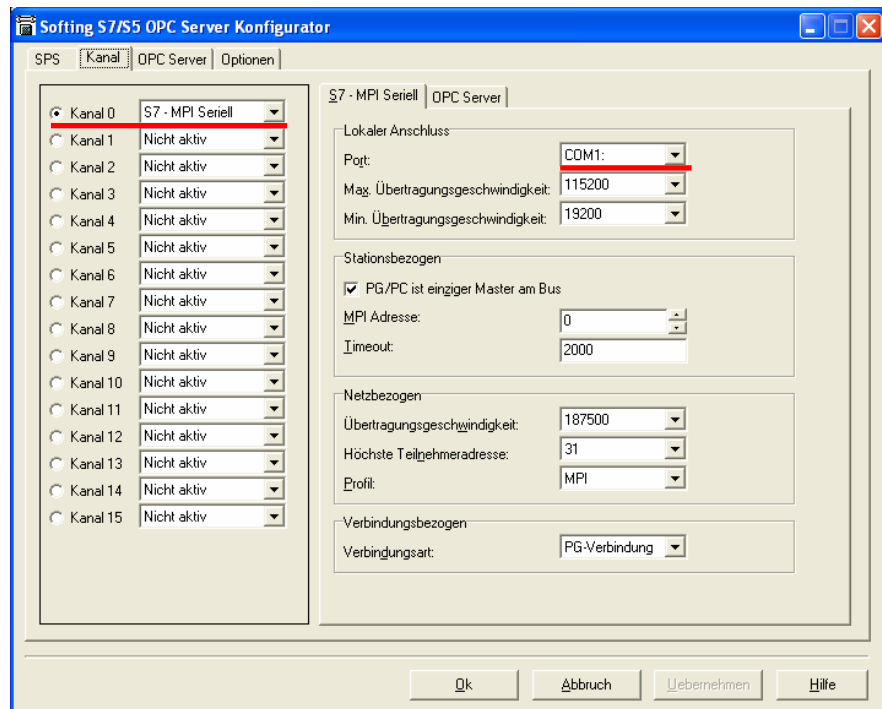
Nach der Installation des Server findet sich unter START ► ALLE PROGRAMME ► SOFTING S7-S5 OPC SERVER der „Konfigurator“.

Alle Reiter des Konfigurations-Tools im Überblick:

*Optionen:* Hier kann die gewünschte Spracheinstellung getätigt werden.

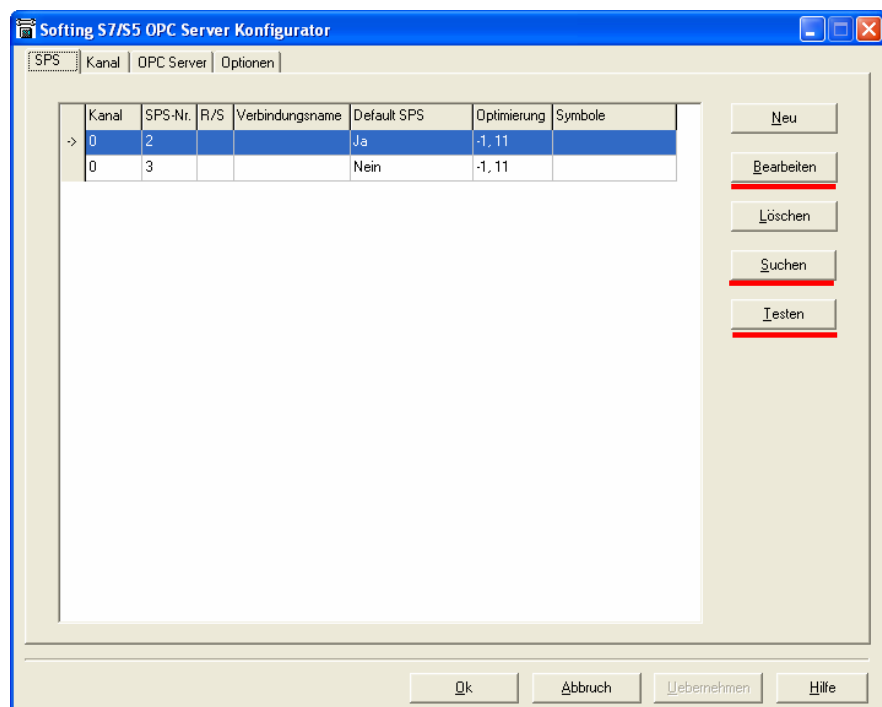
*Kanal:*

Dort wird zunächst die SPS mit passender Schnittstelle unter Kanal 0 ausgesucht, sowie rechts der richtige COM-Port. Alle anderen Kanal-Einstellungen werden normalerweise vom Programm selbst passend gewählt.



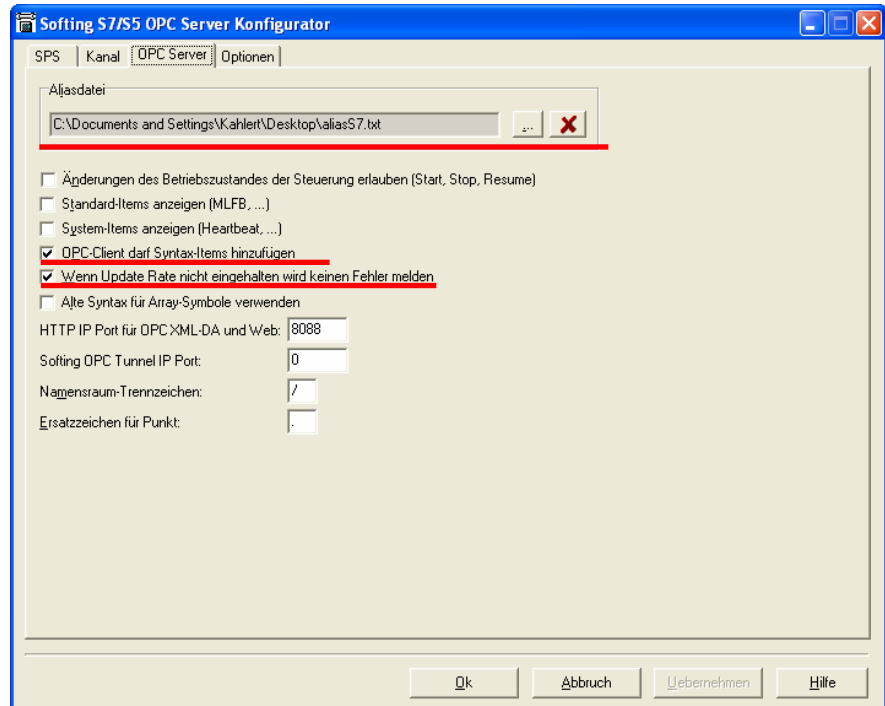
*SPS:*

Mit einem Klick auf *Suchen* und Angabe der Kanalnummer 0 werden alle verfügbaren CPUs und CPs am Bus aufgelistet. Werden, da ein CP in der Anlage verbaut ist, mehrere SPS-Nummern angezeigt, lässt sich mit einem Klick auf *Testen* herausfinden, welche die richtige ist, und mit *Bearbeiten* als Standard/ Default SPS festlegen.



### OPC Server:

Hier sind lediglich die nebenstehend gekennzeichneten Optionen zu aktivieren. Zum Schluss kann noch eine Aliasdatei angelegt werden und unter *Aliasdatei* ausgewählt werden.

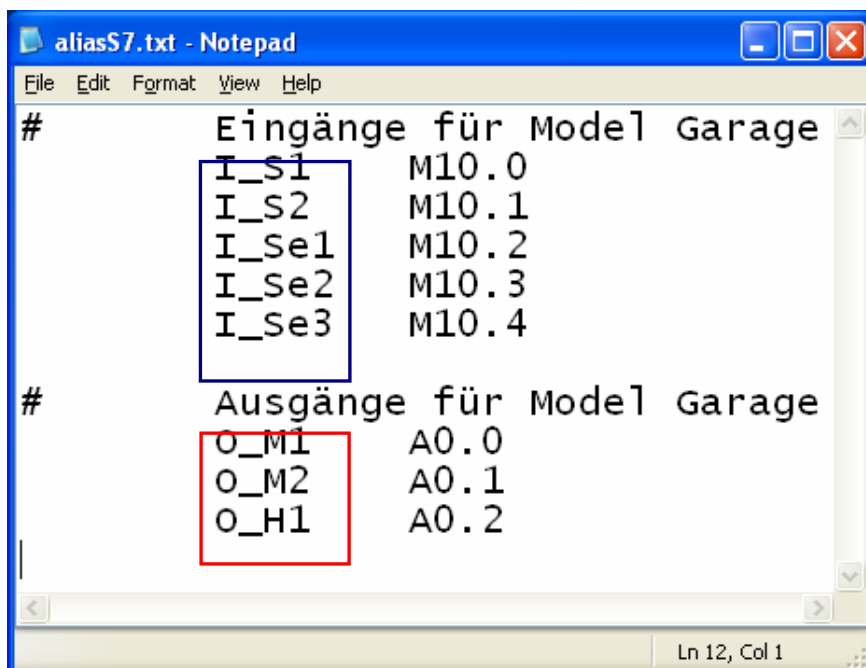


### 2.2.2 Erstellung der Aliasdatei

An einem beliebigen Ort wird eine TXT Datei (z.B. *aliasS7.txt*) angelegt. Mit # wird kommentiert, Adressen werden genauso behandelt wie in STEP 7. Die Syntax sieht wie folgt aus:

<tabulator>Symbol<tabulator>Adresse

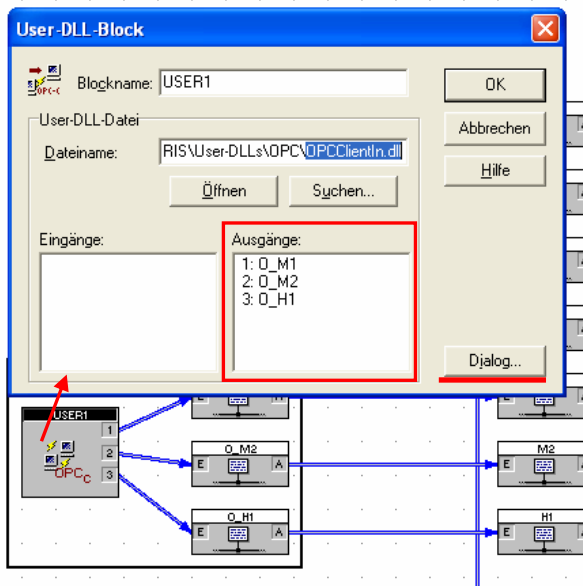
Beispiel:



Es ist notwendig, dass die gleichen Symbole wie in den OPC-Client Bausteinen verwendet werden. Diese lassen sich einfach mit einem Doppelklick auf die zugehörigen User-Blöcke im geöffneten Modell (\*.BSY Datei) in BORIS auslesen.

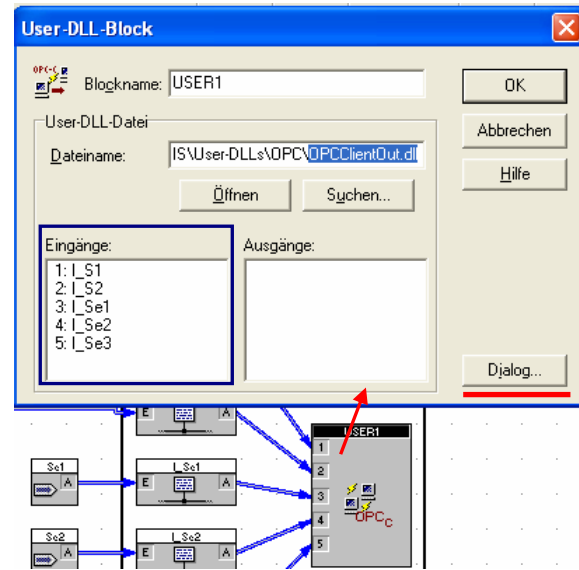
Andererseits können natürlich auch die Symbole in den Blöcken geändert werden. Werden z.B. absolute Adressen benutzt, so wird keine Aliasdatei benötigt. Das heißt: Ausgänge: A0.0, A0.1, ..... Eingänge: M0.0, M0.1, ..... etc.

Ausgänge:




Mit dem OPCClientIn-Userblock lassen sich Daten aus der SPS lesen.

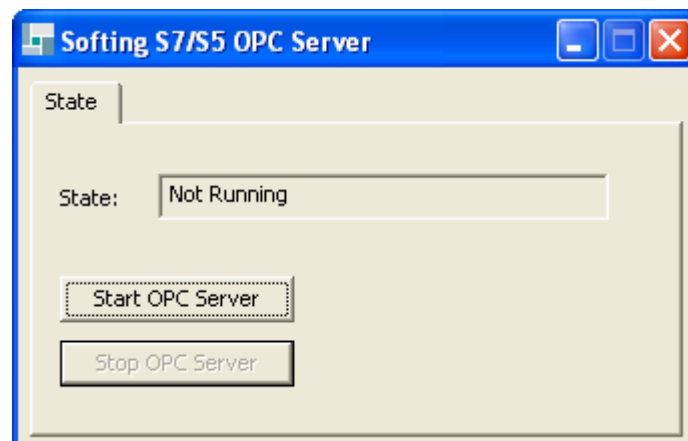
Eingänge:



Mit dem OPCClientOut-Userblock lassen sich Daten auf die SPS schreiben.

### 2.2.3 Starten des Softing OPC-Servers

Ist die Aliasdatei erstellt und eingetragen, kann der OPC-Server gestartet werden. Er wird unten rechts in der Windows-Taskbar mit dem Icon  angezeigt. Mit einem Rechtsklick und der Auswahl von ÖFFNEN kann man im Folgefenster den Server beliebig starten und stoppen:



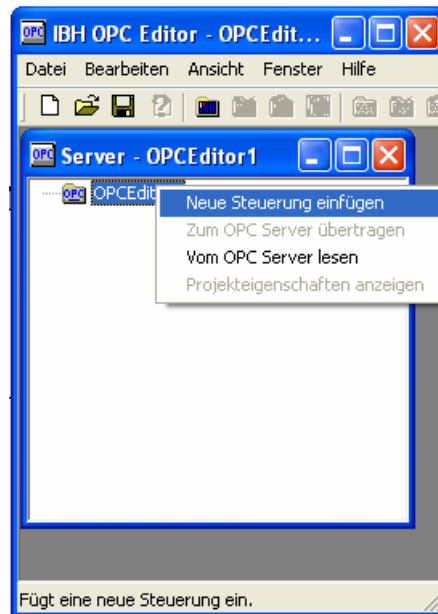
Bei Gebrauch wird der Softing-Server auch automatisch gestartet.

**Achtung:** Das eigentliche Programm muss vorher auf die S7 gespielt worden sein; der OPC-Server belegt den COM-Port und lässt somit keine weitere Verbindung durch Fremdsoftware zu.

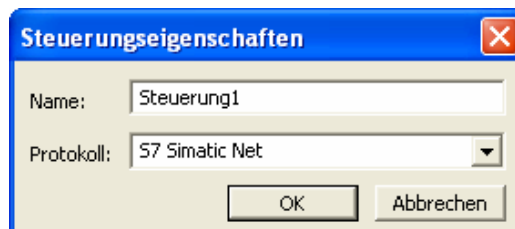
## 2.3 IBH OPC-Server

### 2.3.1 Konfigurierung des IBH OPC-Servers

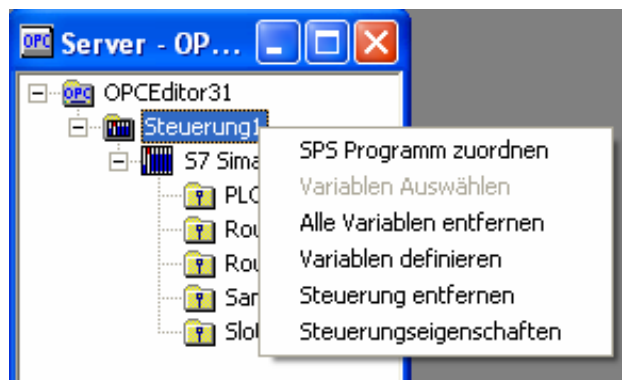
Nach der Installation findet sich unter ALLE PROGRAMME ▶ IBH SOFTEC GMBH ▶ IBH OPC SERVER der *IBH OPC Editor*. Über die Menüoption DATEI ▶ NEU wird eine neue Serverkonfigurationsdatei angelegt. Mit einem Rechtsklick auf *OPCEditor1* wird nun eine neue Steuerung eingefügt:



Im folgenden Fenster wird die individuelle Wahl des Protokolls getätigt:



Durch Rechtsklick auf die neu erschienene Steuerung kann man unter mehreren Optionen wählen:



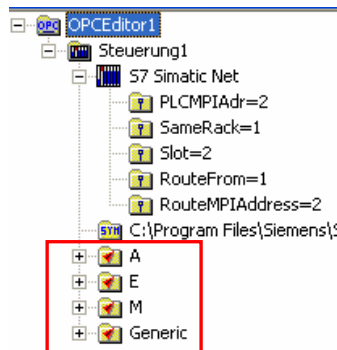
*SPS Programm zuordnen:*

Das geschriebene SPS Programm kann ausgewählt werden, sodass unter „*Variablen Auswählen*“ die Variablen aus der Programm-Symboltabelle aktiviert und für den OPC- Server verfügbar gemacht werden können.



### Variablen definieren:

Hier werden Aliase für Variablen per Hand definiert:



### Alle Variablen entfernen:

Löscht alle Variablen, die unter der SPS aufgetaucht sind.

### Generic:

Alle selbst definierten Variablen

### A-E-M:

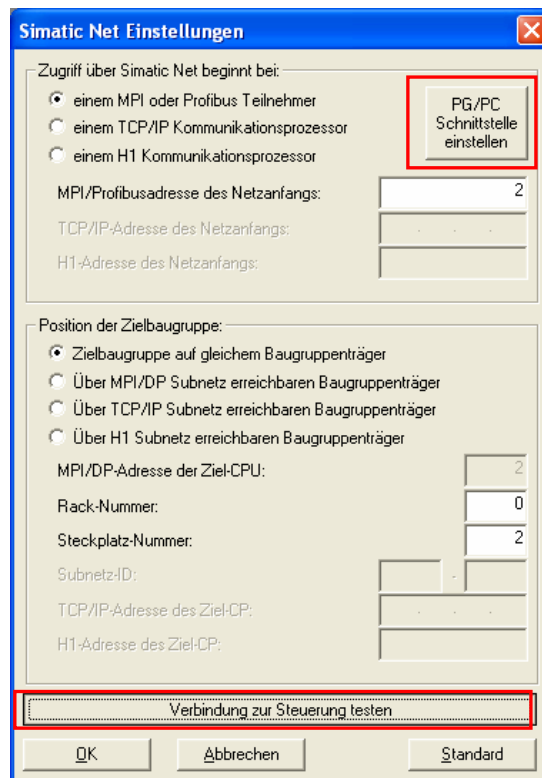
Alle aus dem SPS-Programm übernommenen Variablen

### Steuerung entfernen:

Löscht die Steuerung aus der Konfiguration.

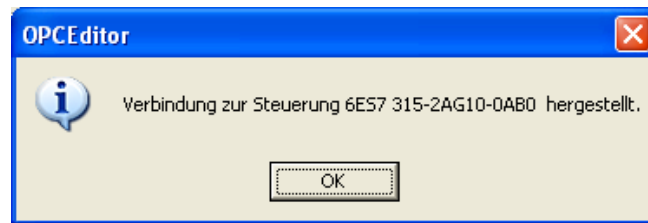
### Steuerungseigenschaften:

Vorheriges Fenster mit Protokoll-Auswahl. Mit einem Rechtsklick auf die SPS (hier: S7 Simatic Net) erscheint unter *Verbindungseinstellungen* folgender Dialog:



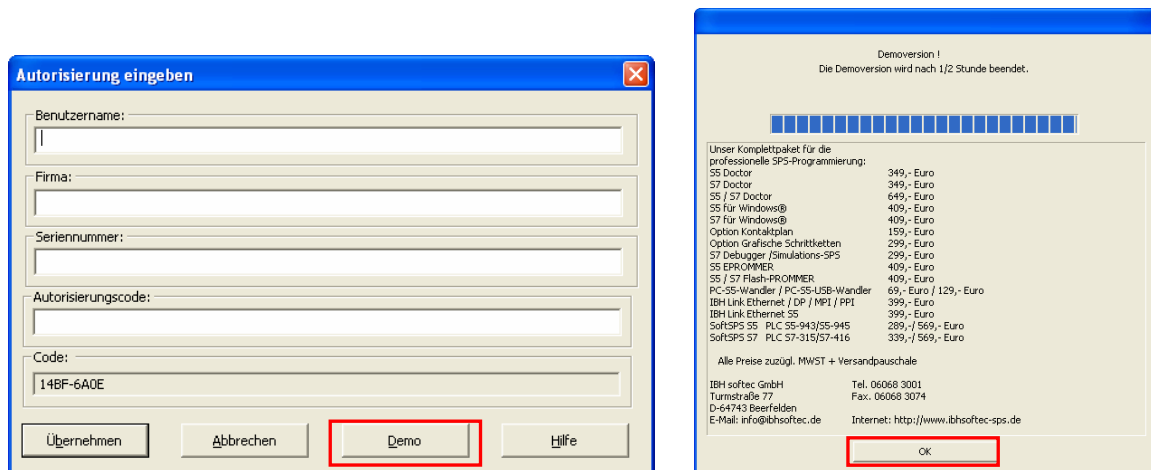
Normalerweise sind alle Einstellungen korrekt, da alle Einstellungen von STEP 7 übernommen werden. Sie können aber über *PG/PC Schnittstelle einstellen* noch verändert werden.

Wird nun auf *Verbindung zur Steuerung testen* geklickt, muss *Verbindung hergestellt* erscheinen mit individuellen Angaben der CPU Seriennummer:



### 2.3.2 Starten des IBH OPC-Servers

Ist alles geschafft, wird der OPC Server bei Gebrauch automatisch aktiviert. Wird mit der Demo-Version gearbeitet, muss jeweils noch auf *Demo* geklickt werden:

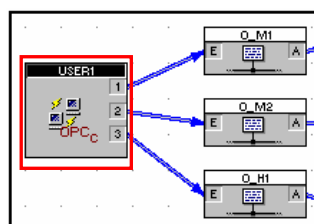


Auch hier gilt wieder: Die COM-Port-Schnittstelle wird nun für Fremdsoftware nicht mehr erreichbar sein.

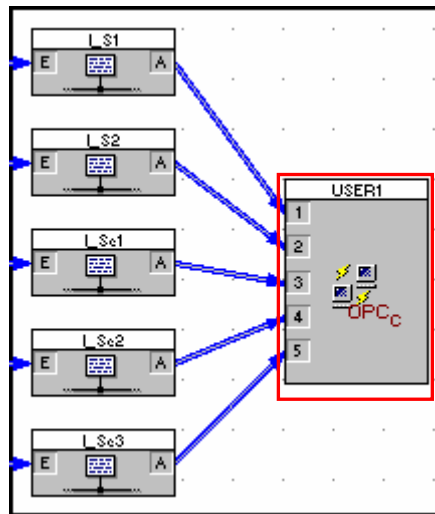
## 3. Der BORIS-OPC-Client

Der OPC-Client wird in BORIS mit entsprechenden Blöcken aus der *OPC-Client-Toolbox* realisiert, welche leicht zu parametrieren sind. Mit einem Doppelklick werden alle Ein und Ausgänge angezeigt und mit einem weiteren Klick auf *Dialog* können Server und Aliase gewählt werden. Auch kann mit *Add Item...* ein Eingang, Ausgang oder Merker absolut angesprochen werden, z.B. A0.0, E0.0. Dies kann sinnvoll sein, wenn man verschiedene Server benutzt, da die IOs nicht mehr für den jeweiligen Server angepasst werden müssen.

Mit dem *OPC Client Input*-Block lassen sich Zustände von Ausgängen und Merkern auslesen:



Mit dem *OPC Client Output*-Block lassen sich Zustände auf die SPS (z. B. auf Merker) schreiben:

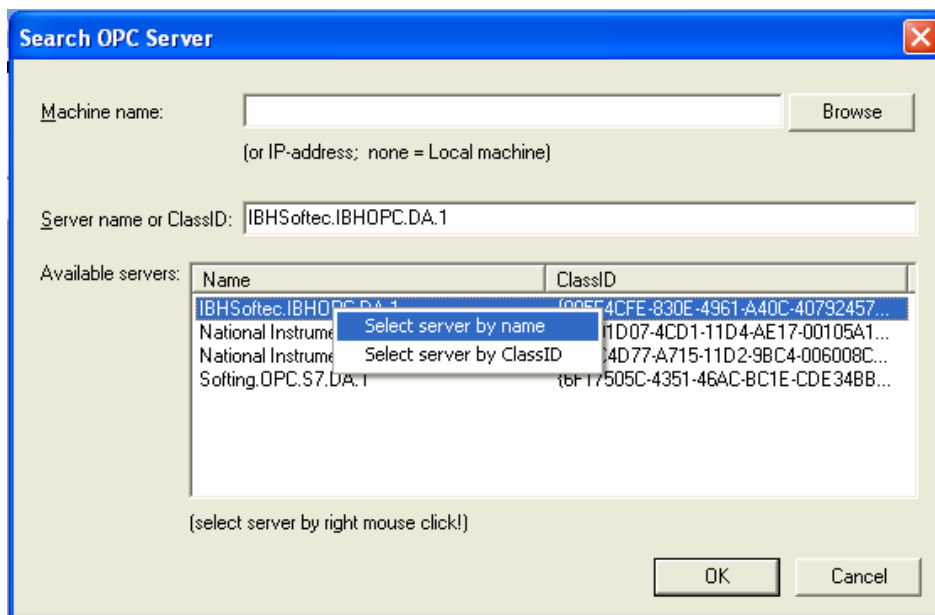


### 3.1 OPC-Client zum Server verbinden

In den Einstellungen unter *Dialog* wird nun zuerst der Server mit *Search...* gesucht und ausgewählt:

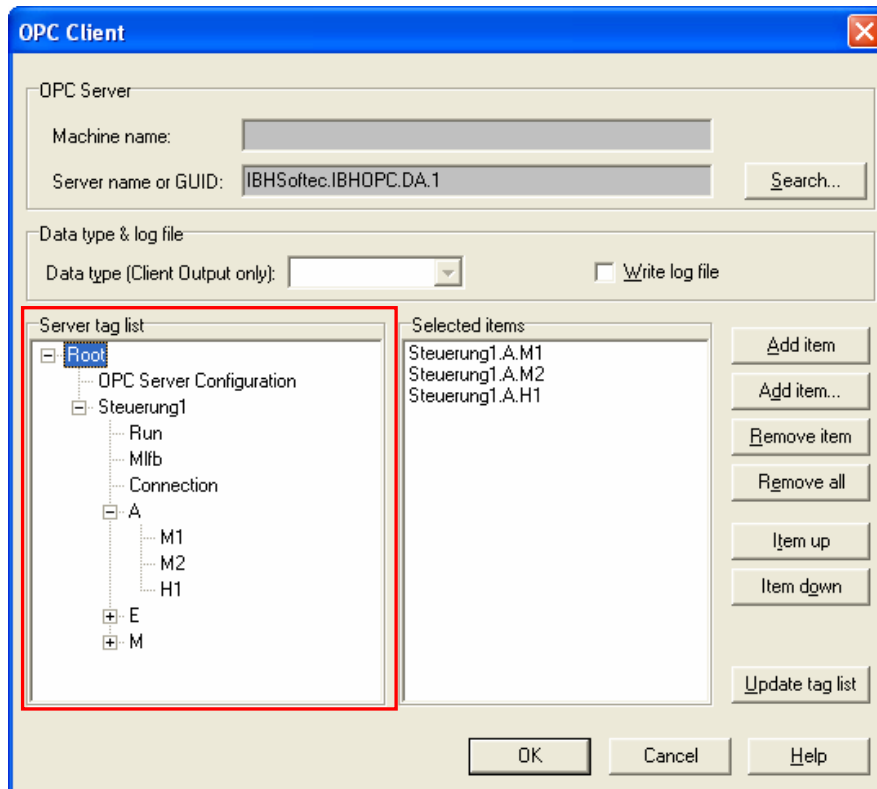
*Search* → Rechtsklick auf gewünschten Server → *Select server by name*

Auf diese Weise wird mit *OPC Client Input*- und *OPC Client Output*-Block verfahren:



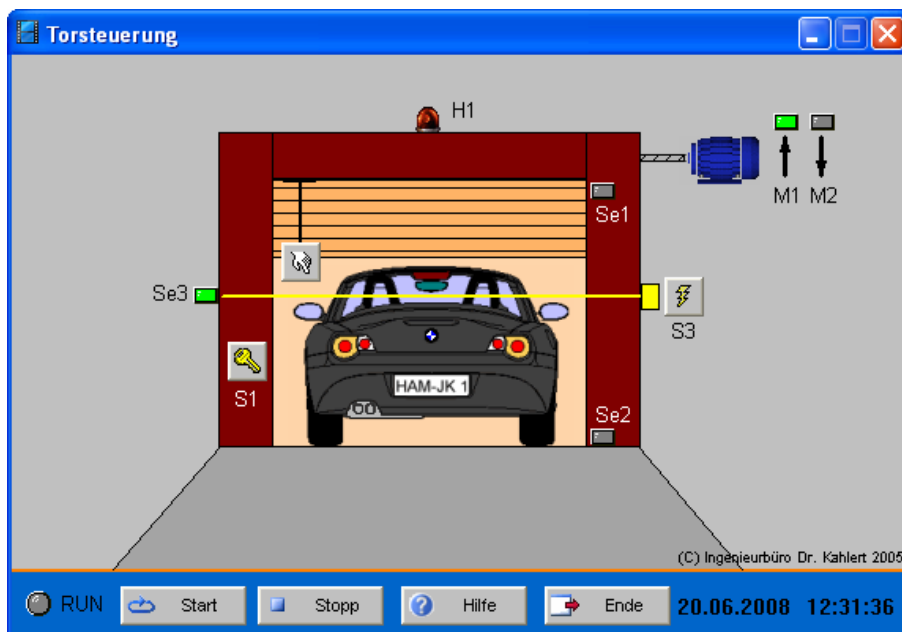
### 3.2 OPC-Client-Tag-Wahl

Ist der Client eine korrekte Verbindung mit dem Server eingegangen, wird unter *Server tag list* ein Rootverzeichnis mit allen Aliasen, die vorher für den Server angelegt/ausgewählt wurden, angezeigt. Mit *Update tag list* wird diese aktualisiert. In *Selected items* werden die Variablen oder direkten (d. h. absoluten) Ein- und Ausgänge der SPS gewählt. Hier wird wieder das gleiche hierarchische Schema angewendet. Oben ist Ein- bzw. Ausgang 1 (hier A0.0), an zweiter Stelle Ausgang 2 (A0.1) usw.



## 4. Beispiel

Nachfolgendes Beispiel demonstriert, wie OPC benutzt werden kann, um das steuerungstechnische Kompaktmodell *Garagentorsteuerung* (siehe nachfolgende Bildschirmgrafik) über eine SPS anzusteuern. Dabei wird hier lediglich die Konfiguration der OPC-Verbindung betrachtet, *nicht* die Erstellung des eigentlichen SPS-Programms.



Die erforderlichen Einstellungen im Überblick:

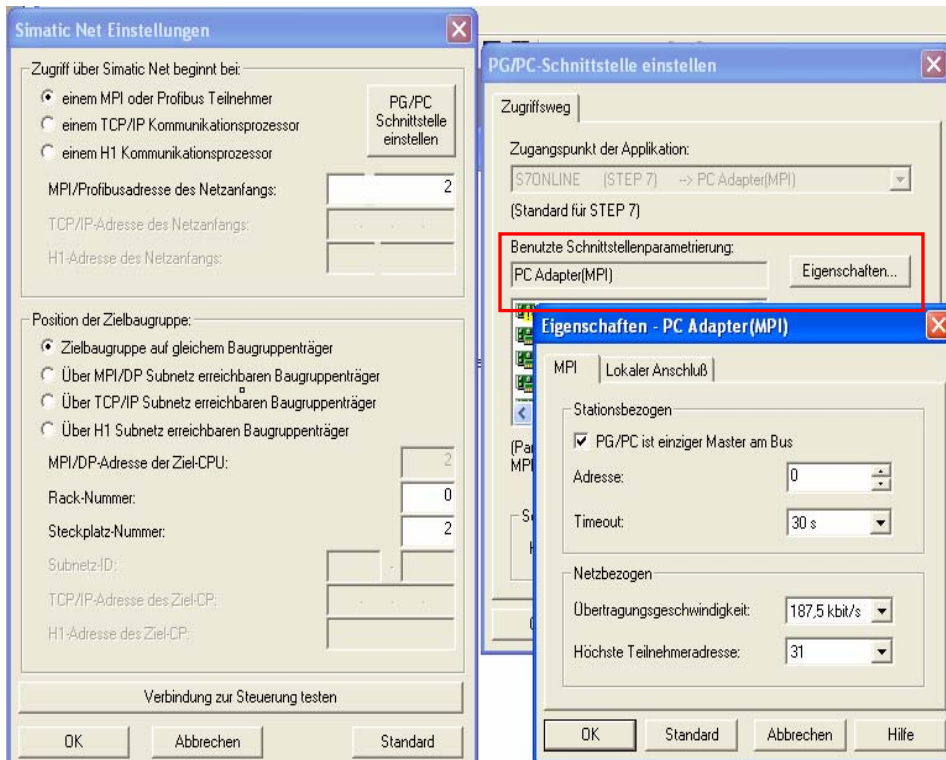
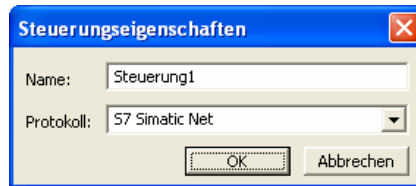
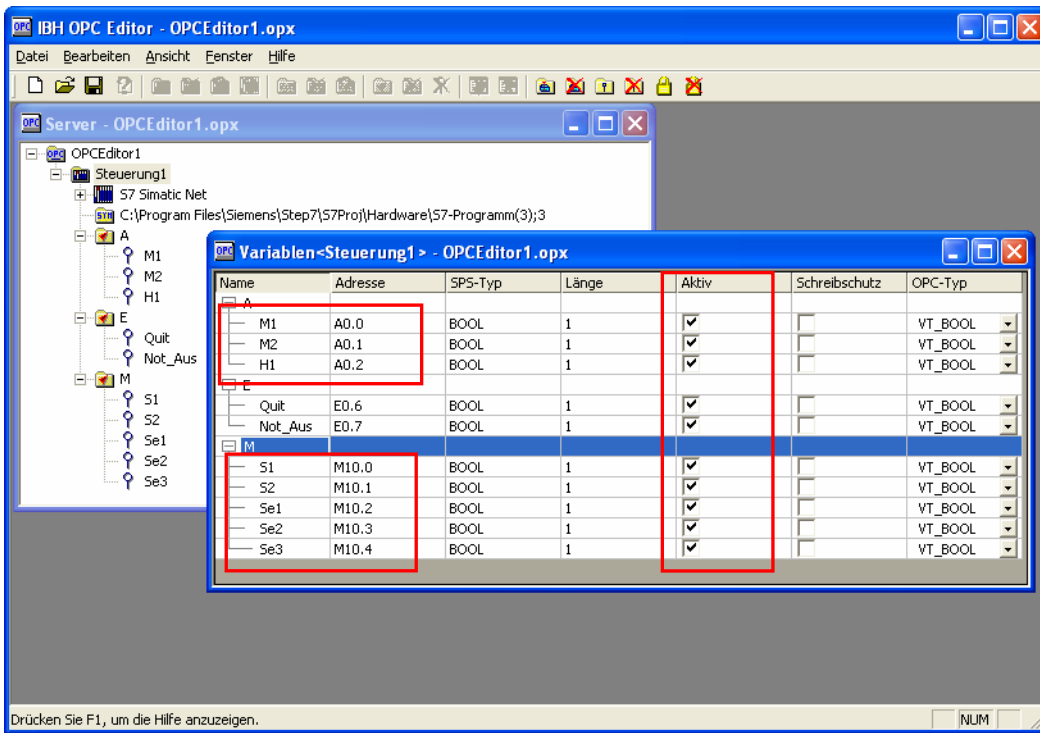
### STEP 7-Symboltabelle

Status	Symbol	Adresse	Datentyp	Kommentar
	Automatikbetrieb	FC 2	FC 2	
	H1	A 0.2	BOOL	Leuchtmelder: Tor wird geschlossen
	M1	A 0.0	BOOL	Motor: Tor öffnen
	M2	A 0.1	BOOL	Motor: Tor schließen
	Not_Aus	E 0.7	BOOL	Notabschalter zum Stilllegen der Anlage
	NotAus	FC 1	FC 1	
	Quit	E 0.6	BOOL	Quittierungstaster zum Zurücksetzen des NotAus
	S1	M 10.0	BOOL	Schlüsselstaster Außen
	S2	M 10.1	BOOL	Zugtaster Innen
	Se1	M 10.2	BOOL	Endlage_Tor geöffnet
	Se2	M 10.3	BOOL	Endlage_Tor geschlossen
	Se3	M 10.4	BOOL	Lichtschränke(öffner)
	Zuweisungen	FC 3	FC 3	

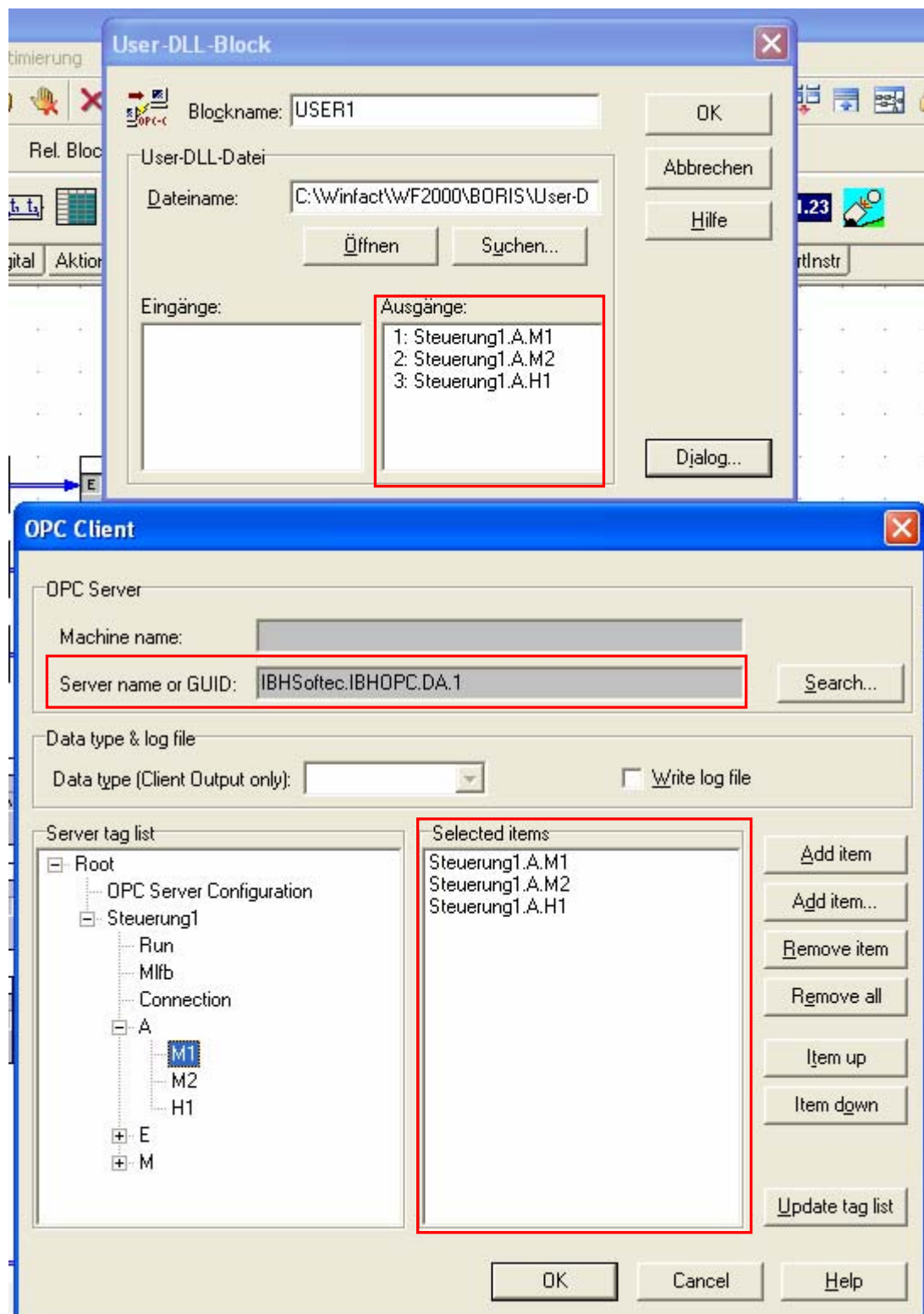
### SPS Hardwarekonfiguration

Steckplatz	Baugruppe	Bestellnummer	Fi...	M...	E...	A...	Kommentar
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU 315-2 DP	6ES7 315-2AG10-0AB0	V2.0	2			
X2	DP				2047		
3							
4	DI16xDC24V	6ES7 321-1BH00-0AA0			0...1		
5	DO16xDC24V/0.5A	6ES7 322-1BH00-0AA0				0...1	

OPC- Server von IBH



## OPC-Client von BORIS (Input Block: OPCClientIn.dll)



## OPC- Client von BORIS (Output Block: OPCClientOut.dll)

