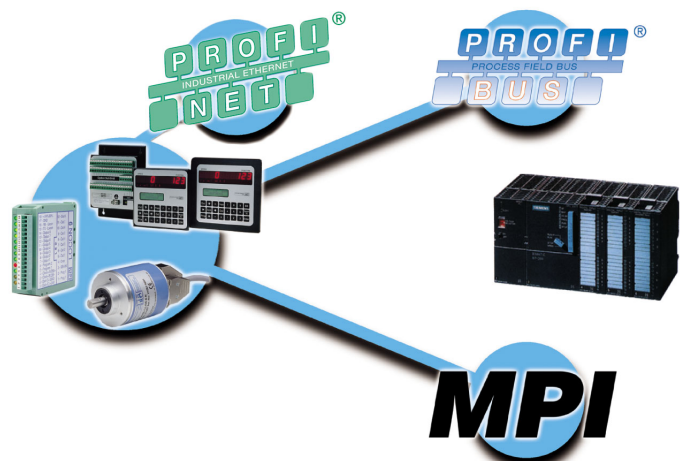


**Deuschmann**

*your ticket to all buses*

## Nockensteuerungen mit Feldbusanbindung



Deuschmann Automation GmbH & Co. KG  
[www.deuschmann.de](http://www.deuschmann.de) | [wiki.deuschmann.de](http://wiki.deuschmann.de)



<b>1</b>	<b>Einführung</b>	<b>5</b>
1.1	Über dieses Handbuch	5
1.1.1	Symbole	5
1.1.2	Begriffliches	5
1.1.3	Anregungen	5
1.2	Produktprogramm der Deutschmann Automation	5
<b>2</b>	<b>Einleitung</b>	<b>6</b>
<b>3</b>	<b>Ablaufschema</b>	<b>7</b>
<b>4</b>	<b>Synchronisation (Startphase)</b>	<b>8</b>
4.1	Datenaustausch	8
4.2	Datenaufbau einer Anforderung (von der SPS zur NS)	9
4.3	Prozessdaten im Singleturn-Format	10
4.4	Prozessdaten im Multiturn-Format	10
<b>5</b>	<b>Tabellen-Typen der Parameter-Daten-Tabelle</b>	<b>11</b>
<b>6</b>	<b>Offset-Tabelle</b>	<b>12</b>
6.1	Nocken in die Nockensteuerung schreiben	12
6.2	Nocken: CAM_ST Nocken	12
6.3	Offset-Tabelle für Totzeit	12
6.4	Totzeit: IDLE	12
6.5	Offset-Tabelle für Steuerungstabelle	13
6.6	Geräte Steuerungstabelle: CONTROL_TYPE	13
6.7	Offset-Tabelle für Multiturn Nocken	13
6.8	Nocken für Multiturn: CAM_MT	13
6.9	Offset-Tabelle für Richtungsnocken	13
6.10	Richtungsnocken: DIRECTION_CAM	14
6.11	Referenz: Reference	14
6.12	Offset-Tabelle für Winkel-Zeit Nocken	15
6.13	Winkel-Zeit Nocken: AT_CAMS_ST	15
6.14	LOGIK-Funktion: LOGIC	15
6.15	Prozeßdaten für Singleturn: PROCESSDATA_ST (read only)	15
6.16	Prozessdaten für Long: PROCESSDATA_LONG (read only)	16
6.17	Prozeßdaten für ROTARNOCK 4: Processdata_80 (read only)	16
6.18	GSD-Module für PROFIBUS NS	16
6.19	Beispiel: Parameter Tabelle	17
6.20	Datenbaustein-Generator	17
<b>7</b>	<b>Anbindung Nockensteuerung über MPI UNIGATE an S7 300 - 400</b>	<b>18</b>
7.1	Zubehör	18
7.2	Inbetriebnahme	18
7.2.1	Gateway konfigurieren	18

7.2.1.1	Konfigmode	18
7.2.2	Beispiel für die Einstellungen	19
7.3	Einstellung RefDB	22
<b>8</b>	<b>Anbindung LOCON 24 Nockensteuerungen über MPI an SPS</b>	<b>24</b>
8.1	Protokoll PLC	24
<b>9</b>	<b>Anbindung LOCON 24 Nockensteuerungen über PROFIBUS an SPS</b>	<b>25</b>
9.1	Protokoll: DICNET	25
9.2	Protokoll: PLC-SPS	25
9.3	Inbetriebnahme	25
<b>10</b>	<b>Anbindung ROTARNOCK Nockensteuerungen über PROFIBUS an eine SPS</b>	<b>26</b>
10.1	Protokoll: PLC-SPS	26
10.2	Inbetriebnahme	26
10.3	Anbindung ROTARNOCK 4 an den PROFIBUS	27
<b>11</b>	<b>Anbindung ROTARNOCK Nockensteuerungen über MPI an S7 300 - 400</b>	<b>29</b>
11.1	Datenaustausch	29
11.2	Inbetriebnahme	29
11.3	Historie	30
11.4	Programmierskabel für ROTARNOCK	30
<b>12</b>	<b>PROFIBUS- und MPI-ID bei LOCON und ROTARNOCK einstellen</b>	<b>31</b>
12.1	PROFIBUS-ID einstellen	31
12.1.1	Beispiel für S7 mit PC-Adapter	31
12.1.2	Beispiel über RS232-Schnittstelle	31
12.2	MPI-ID einstellen	31
<b>13</b>	<b>Anbindung ROTARNOCK Nockensteuerungen über PROFINET an eine S7</b>	<b>32</b>
13.1	Inbetriebnahme	32
13.2	Anbindung ROTARNOCK 100 an PROFINET	33
13.3	PROFINET Gerätenamen	35
13.4	LED-PN	39
13.5	Programmierung der Nocken	40
<b>14</b>	<b>Reaktionszeit der NS bei Hardware-Programmschaltung</b>	<b>43</b>
14.1	Reaktionszeit LOCON 24 MPI	43
14.2	Reaktionszeit ROTARNOCK 2 - Profibus	43
14.3	Schematische Darstellung des Testaufbaus	44
<b>15</b>	<b>Fehlermeldungen</b>	<b>45</b>
15.1	Status LED am ROTARNOCK	45
<b>16</b>	<b>Service</b>	<b>46</b>
16.1	Einsendung eines Gerätes	46
16.2	Internet	47

## 1 Einführung

### 1.1 Über dieses Handbuch

In diesem Handbuch werden die Installation, Funktionen und die Bedienung des jeweils auf dem Deckblatt und in der Kopfzeile genannten Deutschmann-Gerätes dokumentiert.

#### 1.1.1 Symbole



Besonders wichtige Textpassagen erkennen Sie an nebenstehendem Piktogramm.

Diese Hinweise sollten Sie unbedingt beachten, da ansonsten Fehlfunktionen oder Fehlbedienung die Folge sind.

#### 1.1.2 Begriffliches

Im weiteren Verlauf dieses Handbuchs werden häufig die Ausdrücke „ROTARNOCK“ und „LOCON“ ohne weitere Modellangabe benutzt. In diesen Fällen gilt die Information für die gesamte Modellreihe.

#### 1.1.3 Anregungen

Für Anregungen, Wünsche etc. sind wir stets dankbar und bemühen uns, diese zu berücksichtigen. Hilfreich ist es ebenfalls, wenn Sie uns auf Fehler aufmerksam machen.

### 1.2 Produktprogramm der Deutschmann Automation

Eine ausführliche und aktuelle Übersicht über unser Produktspektrum finden Sie auf unserer Homepage <http://www.deutschmann.de>.

## 2 Einleitung

DEUTSCHMANN-Nockensteuerungen (NS) mit PROFIBUS- oder MPI-Schnittstelle<sup>1</sup> können einfach an einer SPS betrieben werden. Hierfür ist es erforderlich, dass entweder das Protokoll DICNET oder das Protokoll PLC-SPS in der Nockensteuerung eingestellt ist.

Das DICNET oder Deutschmann Protokoll kann alle Parameter von einer Nockensteuerung bearbeiten.

Das PLC-SPS Protokoll erlaubt die Bearbeitung der wichtigsten Parameter in der NS (siehe Kapitel 5, "Tabellen-Typen der Parameter-Daten-Tabelle" auf Seite 11). Durch die Tabellenform ist ein vereinfachtes Parametrieren möglich.

Dieses Handbuch beschreibt das PLC-SPS-Protokoll, zum DICNET-Protokoll, siehe Anmerkungen im Kapitel "Protokoll: DICNET" auf Seite 25.

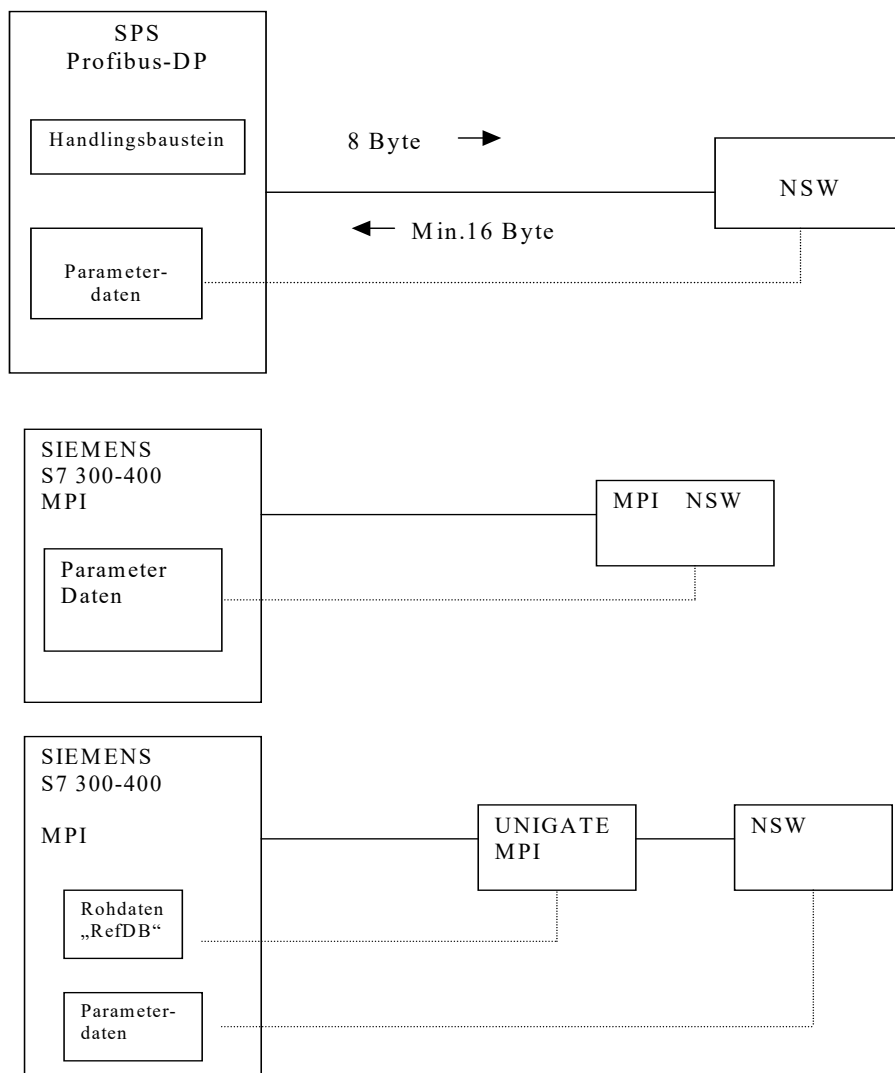
---

**1. MPI-Schnittstelle: nur für eine Siemens SPS mit MPI-Schnittstelle**

### 3 Ablaufschema

Ein SPS Programm (Handlingsbaustein) übernimmt bei einer SPS mit PROFIBUS die Übermittlung der Parameterdaten zur Nockensteuerung (NS).

Die NS liefert in jedem PROFIBUS-Zyklus die Prozessdaten zurück. Die Länge der Prozessdaten ist abhängig von der NS und dem gewählten Modul der GSD-Datei (siehe Kapitel 6.18, "GSD-Module für PROFIBUS NS" auf Seite 16).

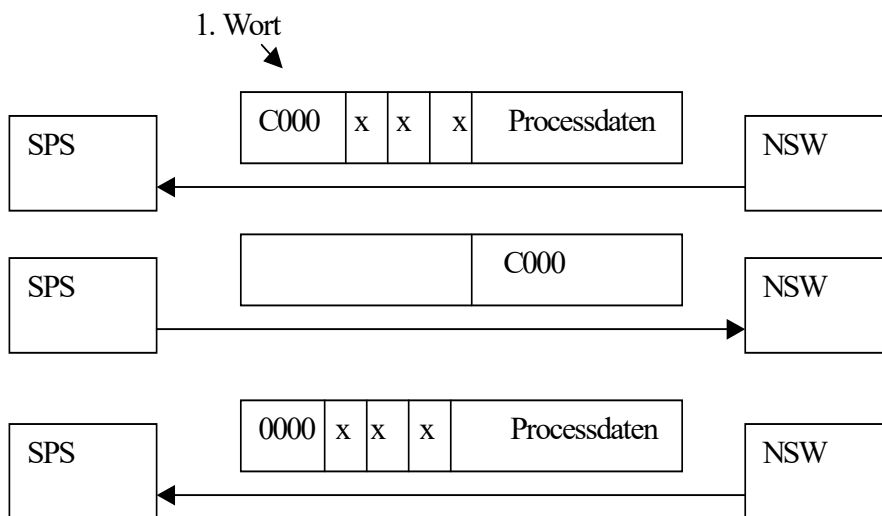


## 4 Synchronisation (Startphase) <sup>2</sup>

Nach dem Einschalten der NS synchronisieren sich die SPS und die NS, danach werden die Parameterdaten zyklisch zur NS kopiert.

Die NS sendet nach dem Einschalten ein 0xC000 (Bit 15 und 14 im 1. Wort gesetzt) solange, bis es dieses Wort als Echo von der SPS zurück bekommt. Daraufhin sendet die NS zur SPS ein 0x0000 (Bit 15 und 14 im 1. Wort gelöscht) und beendet damit die Synchronisation.

Ab jetzt kopiert der Handlingbaustein aus der Parameter Daten Tabelle der SPS immer 3 aufeinander folgende Worte, geführt vom Adresswort über den Bus zur NS (siehe Kapitel 4.2, "Datenaufbau einer Anforderung (von der SPS zur NS)" auf Seite 9).



Werte in Hex

x = not used

### 4.1 Datenaustausch

Die SPS sendet jetzt 0x8000 (Bit 15 write-req gesetzt) mit Start-Adresse des ersten Wortes aus der Tabelle und den ersten drei Worten aus der Parameter Tabelle.

Eine Parameter Tabelle fängt immer mit Adresse 0 an. Es müssen immer alle Daten im Abstand von 6 Byte zur NS kopiert werden. Als Bestätigung wartet die SPS auf das erste Wort der Anfrage (0x8000 Hex).

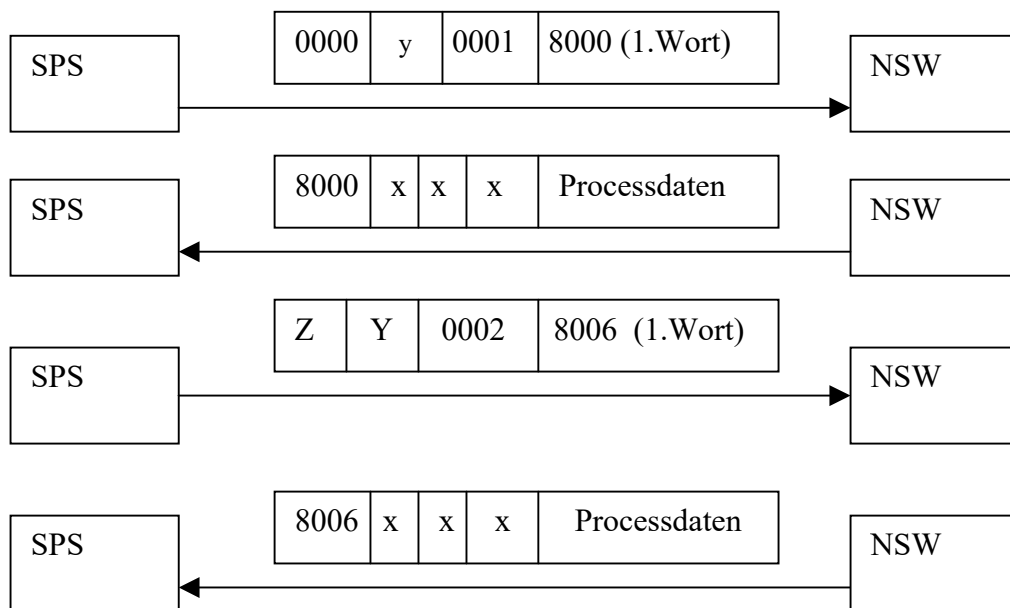
Der zweite Datensatz, den die SPS an die NS sendet ist 0x8006 Hex im ersten Wort und den nächsten 3 Worten aus der Parameter Tabelle.

Somit muß die Startadresse immer ein Vielfaches von 6 sein. Am Ende der Tabelle angelangt beginnt der Kopiervorgang wieder ab der Adresse 0.

Als Bestätigung wartet die SPS auf das erste Wort der Anfrage (0x8006 Hex).

Zusätzlich bekommt die SPS immer die Prozessdaten zurück (siehe Kapitel 4.3, "Prozessdaten im Singleturn-Format" auf Seite 10).





X = not used

Y = Anzahl Bytes der Offset Tabelle

Z = Startadresse der Parameterdaten z.B. vom Typ 2

#### 4.2 Datenaufbau einer Anforderung (von der SPS zur NS)

1. Wort			2. Wort	3. Wort	4. Wort	5. Wort
Bit 15 Befehl	Bit 14 Befehl	13 ..... 0 Address Pointer	15 ... 0 Data	15 ... 0 Data	15 ... 0 Data	nur bei R 4 mit Logikfunktion
0	0	not used	6 Byte Daten ab Address Pointer			16 Bit Logik für R4
0	1	not used				
1	0	write-request				
1	1	sync (Startphase)				

**Hinweis:** In Bezug auf die Logikfunktion muss bei Verwendung der 16 Eingänge folgendes beachtet werden:

\*ROTARNOCK 4-PROFIBUS: Der 1. Eingang ist im 10 Byte im 5. Wort im PROFIBUS.

\*ROTARNOCK 100-PROFIBUS: Der 1. Eingang ist im 9 Byte im 5. Wort im PROFIBUS.

### 4.3 Prozessdaten im Singleturn-Format

Die Prozessdaten werden in jedem PROFIBUS-Zyklus ab dem 5. Wort mitgeliefert.

**Datenaufbau einer Antwort von der NS an die SPS:**

Wort	1. Wort	2. Wort	3. Wort	4. Wort	5. Wort	6. Wort	7. Wort	8. Wort
Bit	15 .. 0	15 .. 0	15 .. 0	15 .. 0	15 .. 0	15 .. 0	15 .. 0	15 .. 8    7 .. 0
	Kopie des 1. Wortes von der SPS als Empfangsbestätigung	—	—	—	Prozessdaten			
					Position	Speed	Output	Act-Progr    Error No
					H   L	H   L		

### 4.4 Prozessdaten im Multiturn-Format

Die Prozessdaten werden permanent ab dem 5. Wort mitgeliefert.

**Datenaufbau einer Antwort von der NS an die SPS:**

Wort	1. Wort	2. Wort	3. Wort	4. Wort	5. Wort	6. Wort	7. Wort	
Byte Nr.	Byte 0, 1	Byte 2, 3	Byte 4, 5	Byte 6, 7	Byte 8, 9	Byte 10, 11	Byte 12, 13	.....
Bit	15 .. 0	15 .. 0	15 .. 0	15 .. 0	31 .. 16	15 .. 0	15 .. 0	.....
	Kopie des 1. Wortes von der SPS als Empfangsbestätigung	—	—	—	Position		Speed	.....

.....	Byte 14	Byte 15	Byte 16	Byte 17	Byte 18	Byte 19
.....	7 .. 0	7 .. 0	7 .. 0	7..0	7..0	7..0
.....	ActProgr	ErrorNo	Output 1 to 8	Output 9 to 16	Output 17 to 24	Output 25 to 32

.....	Byte 20	Byte 21	Byte 22	Byte 23	Byte 24	Byte 25
.....	7 .. 0	7 .. 0	7 .. 0	7..0	7..0	7..0
.....	Output 33 to 40	Output 41 to 48	Output 49 to 56	Output 57 to 64	(Reserve)	Reserve

### Zuordnung der Ausgänge zu Bits

**Beispiel:**

MSB	...	LSB
Bit 7		Bit 0
Output 8	...	Output 1

## 5 Tabellen-Typen der Parameter-Daten-Tabelle

Bezeichnung	Fester Wert	Tabellen-Typ	Länge
OFFSET_TYPE	0x0001	Offset fängt immer bei Tabellen-Adresse 0 an	WORD
CAM_ST_TYPE	0x0003	Nocken Tabelle	WORD
IDLE_TYPE	0x0004	Totzeiten Tabelle	WORD
CONTROL_TYPE	0x0005	Config-Parameter Tabelle	WORD
CAM_MT_TYPE	0x0007	Nocken für Multiturn Geräte	WORD
DIRECTIONCAM_TYPE	0x0008	Richtungsnocken	WORD
REF_TYPE	0x000A	nur für MPI-Gateway mit Protokoll PLC-SPS	WORD
AT_CAM_ST_TYPE	0x000B	Winkel-Zeit Nocken	WORD
LOGIC_TYPE	0x000C	Logikfunktion	WORD

## 6 Offset-Tabelle

In der Offset-Tabelle sind jeweils drei Worte zur Beschreibung einer Parameter-Tabelle reserviert.

Die ersten drei Worte müssen zwingend am Anfang dieser Parameter-Tabelle stehen, sie deklarieren die Offset-Tabelle selbst.

Das erste Wort enthält das Schlüsselzeichen für diese Offset-Tabelle (0x0001), im zweiten Wort steht die Anzahl der benötigten Bytes der Tabelle und im dritten Wort steht die Startadresse der Tabelle.

Die Startadresse der Offset-Adresse ist immer Null.

Die Einträge in der Offset-Tabelle ermöglichen bestehende Tabellen zu erweitern bzw. neue hinzuzufügen. Die Parameter Tabelle fängt immer mit der Offsetdeklaration an (Beispiel)

Bezeichnung	Wert in der SPS	Funktion	Länge
Offset_Type	1		WORD
Offset_Length	muß berechnet werden		WORD
Offset_Address	0		WORD

Danach folgt die Deklaration der einzelnen Tabellen durch die Offset Tabelle.

Als Beispiel sind hier nur die Offset-Tabellen für Nocken und Totzeit aufgeführt.

### 6.1 Nocken in die Nockensteuerung schreiben

#### Offset-Tabelle für Nocken

Bezeichnung	Wert in der SPS	Funktion	Länge
Cam_ST_Type	3		WORD
Cam_ST_Length	Anzahl benötigter Nocken mal 6		WORD
Cam_ST_Address	Adresse des ersten Nockeneintrags		WORD

### 6.2 Nocken: CAM\_ST Nocken

Struktur	Wert	Funktion	Länge
ProgNo	0-15		BYTE
Output		0=Nocke wird gelöscht	BYTE
On			WORD
Off			WORD

Sind die Schaltepunkte ON und OFF beide 0, wird die Nocke auch gelöscht.

### 6.3 Offset-Tabelle für Totzeit

Bezeichnung	Wert in der SPS	Funktion	Länge
Idle_Type	4		WORD
Idle_Length	Anzahl benötigter Totzeiten mal 6		WORD
Idle_Address	Adresse des ersten Totzeiteintrags		WORD

### 6.4 Totzeit: IDLE

Struktur	Wert	Funktion	Länge
ProgNo			BYTE
Output		0 <sub>n</sub> und OFF = 0 => diese Totzeit wird gelöscht	BYTE
IdleT_On			WORD
IdleT_Off			WORD

Jede weitere benötigte Totzeit wird lückenlos angereicht.

Der SPS Programmierer braucht nur die entsprechenden Werte in der Struktur-Tabelle ändern, um automatisch die NS zu programmieren.

## 6.5 Offset-Tabelle für Steuerungstabelle

Bezeichnung	Wert in der SPS	Länge
Control_Type	5	WORD
Control_Length	hier fest auf 6	WORD
Control_Address	Adresse des ersten Controleintrages (New_Prog)	WORD

## 6.6 Geräte Steuerungstabelle: CONTROL\_TYPE

In dieser Tabelle werden Bytes oder Flags vom SPS Programmierer gesetzt, die eine spezielle Konfiguration in der NS ausführen. Die Tabelle besteht aus sechs Byte:

Beschreibung	Wert	Funktion	Länge
New_Prog	0.. 15	Neues Programm wählen	BYTE
ConfigFlags: Teach_In_Zero	Bit 0	Teach-In Zero Point (High aktiv)	BOOL
Invert_Encoder	Bit 1	Invert-Encoder-Countdir (0=not invert, 1=invert)	BOOL
Error_Quit	Bit 2	Error-Quit	BOOL
Res_03...Res_07	Bit 3 - 7	Res_03_Res_07	BOOL
Res_0			BYTE
Res_1			BYTE
Res_2			BYTE
Res_3			BYTE

## 6.7 Offset-Tabelle für Multiturn Nocken

Bezeichnung	Wert in der SPS	Länge
CAM_MT_Type	7	WORD
CAM_MT_Length	Anzahl der benötigten Nocken mal 12	WORD
CAM_MT_Address	Adresse des ersten MT -Eintrages	WORD

## 6.8 Nocken für Multiturn: CAM\_MT

Struktur	Wert	Funktion	Länge
ProgNo			WORD
Output		0 = löscht die Nocke im Gerät	WORD
On			DWORD
Off			DWORD

## 6.9 Offset-Tabelle für Richtungsnocken

Wenn für einen Ausgang korrespondierende Bits 0 sind, ist dieser Ausgang deaktiv.

Wenn für einen Ausgang das korrespondierende Bit „pos“ = 1 und Bit „neg“ = 0 ist schaltet die Nocke in positive Drehrichtung

Bezeichnung	Wert in der SPS	Länge
Direction_Cam_Type	8	WORD
Direction_Cam_Length	6 Byte	WORD
Direction_Cam_Address	Adresse des ersten Richtungs-Eintrages	WORD

## 6.10 Richtungsnocken: DIRECTION\_CAM

Struktur	Wert	Funktion	Länge
Output16pos	Bit 7	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output15pos	Bit 6	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output14 pos	Bit 5	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output13pos	Bit 4	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output12pos	Bit 3	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output11pos	Bit 2	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output10pos	Bit 1	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output9pos	Bit 0	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output8pos	Bit 7	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output7pos	Bit 6	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output6pos	Bit 5	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output5pos	Bit 4	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output4pos	Bit 3	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output3pos	Bit 2	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output2pos	Bit 1	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output1pos	Bit 0	bei pos. Drehrichtung schaltender Ausgang	BOOL
Output16neg	Bit 7	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output15neg	Bit 6	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output14neg	Bit 5	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output13neg	Bit 4	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output12neg	Bit 3	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output11neg	Bit 2	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output10neg	Bit 1	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output9neg	Bit 0	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output8neg	Bit 7	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output7neg	Bit 6	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output6neg	Bit 5	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output5neg	Bit 4	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output4neg	Bit 3	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output3neg	Bit 2	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output2neg	Bit 1	bei neg. Drehrichtung schaltender Ausgang	BOOL
Output1neg	Bit 0	bei neg. Drehrichtung schaltender Ausgang	BOOL
reserved	not used		WORD

### Wirkungsweise der Richtungsnocken

Zustand	Output 1 negativ	Output 1 positiv
Kein Ausgangsupdate	0	0
Nur negative Richtung	1	0
Nur positive Richtung	0	1
Beide Richtungen	1	1

Im Defaultzustand erfolgt ein Ausgangsupdate in beide Richtungen.

## 6.11 Referenz: Reference<sup>3</sup>

Struktur	Wert	Funktion	Länge
Offset_Type	1		WORD
Offset_Length	0x000C		WORD
Offset_Address	0		WORD
ID_DB_Table_Type	0x000A		WORD
ID_DB_Table_Length	6		WORD
ID_DB_Table_Address	0x000 C		WORD
Ref_Table.Device (1).ID			WORD
Ref_Table.Device (1).FlagReg			WORD
Ref_Table.Device (1).DB_No			WORD

### 3. nur für UNIGATE MPI

## 6.12 Offset-Tabelle für Winkel-Zeit Nocken

Bezeichnung	Wert in der SPS	Länge
AT_CAM_ST_Type	0x0B	WORD
AT_CAM_ST_Length	Anzahl der benötigten Nocken mal 6	WORD
AT_CAM_ST_Address	Adresse des ersten Winkel-Zeit-Nocken-Eintrages	WORD

## 6.13 Winkel-Zeit Nocken: AT\_CAMS\_ST

Struktur	Wert	Funktion	Länge
ProgNo			BYTE
Output			BYTE
On			WORD
Duration	0x0001 - 0x7EF4	ms	WORD

## 6.14 LOGIK-Funktion: LOGIC

Struktur	Wert	Funktion	Länge
ProgNo	Von 0 bis MAX_PROG		BYTE
DestNo	Von 1 bis 16	0 löscht gesamte Logik-Funktion	BYTE
DestType	0 = Hardware Ausgang 1 = Merker 2 = Hardware Ausgang invertiert 3 = Merker invertiert		BYTE
OpNo1	1-32		BYTE
OpType1	0 = interner NS Ausgang 1 = Eingang: Hard-/Software 2 = Merker 3 = SR (Schieberegister) 4 = PB-Input (only LOCON 200)		BYTE
LogicFct1_2	0 = keine 1 = Oder 2 = Und 3 = Oder Nicht 4 = Und Nicht		BYTE
OpNo2	1-32		BYTE
OpType2	siehe OpType1		BYTE
LogicFct2-3	siehe LogicFct 1-2		BYTE
OpNo3	1-32		BYTE
OpType3	siehe OpType1		BYTE
LogicFct3-4	siehe LogicFct 1-2		BYTE
OpNo4	1-32		BYTE
OpType4	siehe OpType1		BYTE
OutputDelay	ms	Zur Zeit max 255	WORD
OutputTrigger	0 = steigende Flanke 1 = fallende Flanke		BYTE
Module number (only LOCON 200)	0 Basis X I/O-module number		BYTE

## 6.15 Prozeßdaten für Singleturn: PROCESSDATA\_ST (read only)

Struktur	Wert	Länge
Position	1. Byte Position H, 2. Byte Position L	WORD
Speed	3. Byte Speed H, 4. Byte Speed L	WORD
Output 16_1	5. Byte Output (15..8), 6. Byte Output (7..0)	WORD
Act_Prog	7. Byte zeigt aktuelle Programm Nr. an	BYTE
ErrorNo	8. Byte zeigt aktuelle Fehler Nr. an	BYTE
ResWord_1		WORD
ResWord_2		WORD

## 6.16 Prozessdaten für Long: PROCESSDATA\_LONG (read only)

Struktur	Wert	Länge
Position	1. Byte Position H, 4. Byte Position L	DWORD
Speed	5. Byte Speed H, 6. Byte Speed L	WORD
Act_Prog	7. Byte zeigt aktuelle Programm Nr. an	BYTE
ErrorNo	8. Byte zeigt aktuelle Fehler Nr. an	BYTE
Output 1to8	9. Byte Output (7..0)	BYTE
Output 9to16	10. Byte Output (15..8)	BYTE
Output 17to24	11. Byte Output (23..16)	BYTE
Output 25to32	12. Byte Output (31..24)	BYTE

## 6.17 Prozeßdaten für ROTARNOCK 4: Processdata\_80 (read only)

Struktur	Wert	Länge
Position	1. Byte Position H, 4. Byte Position L	DWORD
Speed	5. Byte Speed H, 6. Byte Speed L	WORD
Act_Prog	7. Byte zeigt aktuelle Programm Nr. an	BYTE
ErrorNo	8. Byte zeigt aktuelle Fehler Nr. an	BYTE
Output 1 to 8	9. Byte Output (7..0)	BYTE
Output 9 to 16	10. Byte Output (15..8)	BYTE
Output 17 to 24	11. Byte Output (23..16)	BYTE
Output 25 to 32	12. Byte Output (31..24)	BYTE
Output 33 to 40	13. Byte Output (39..32)	BYTE
Output 41 to 48	14. Byte Output (47..40)	BYTE
Output 49 to 56	15. Byte Output (55..48)	BYTE
Output 57 to 64	16. Byte Output (63..56)	BYTE
Output 65 to 72	17. Byte Output (71..64)	BYTE
Output 73 to 80	18. Byte Output (79..72)	BYTE

Die Prozessdaten liefert die NS. Ein weiterer Handlingsbaustein muß für die Verarbeitung der Prozessdaten sorgen.

Die Werte der Prozessdaten Tabelle werden vom Handlingbaustein auch wieder zur NS zurück-gesendet, es hat aber keinen Einfluß auf die Nockensteuerung. Die Prozessdaten liefert die Nockensteuerung. Ein weiterer Handlingbaustein kopiert die Daten in diese Tabelle.

## 6.18 GSD-Module für PROFIBUS NS

GSD-Datei	Modul	NS
dagw2079	PLC-CSU-ST	LOCON
dagw2079	PLC-CSU Long	Multiturn/LOCON
R2pb2935	S7 DB + Proc.Data	ROTARNOCK 1, 2, 3
R2pb2935	S7 DB + Proc.DataLong	MTROTARNOCK
R4pb3231	S7 DB + Proc.Data	ROTARNOCK 4
R4pb3231	S7 DB + Proc.DataLogic	ROTARNOCK 4 mit Logik
R100	S7 DB, Proc.Data, No Logic	ROTARNOCK 100
R100	S7 DB, Proc.Data, Logic 16	ROTARNOCK 100 mit Logik
R100	S7 DB, Proc.Data (CPU318)	ROTARNOCK 100 an S7 318
L100	S7 DB, Proc.Data, No Logic	LOCON 100
L100	S7 DB, Proc.Data, Logic 16	LOCON 100 mit 16 Logikeingängen
L100	S7 DB, Proc.Data, Logic 8	LOCON 100 mit 8 Logikeingängen
L100	S7 DB, Proc.Data (CPU318)	LOCON 100 an S7 318

### Erklärung:

Type	Länge
BOOL	1 Bit
BYTE	8 Bit
WORD	2 Byte



DWORD	4 Byte
-------	--------

## 6.19 Beispiel: Parameter Tabelle

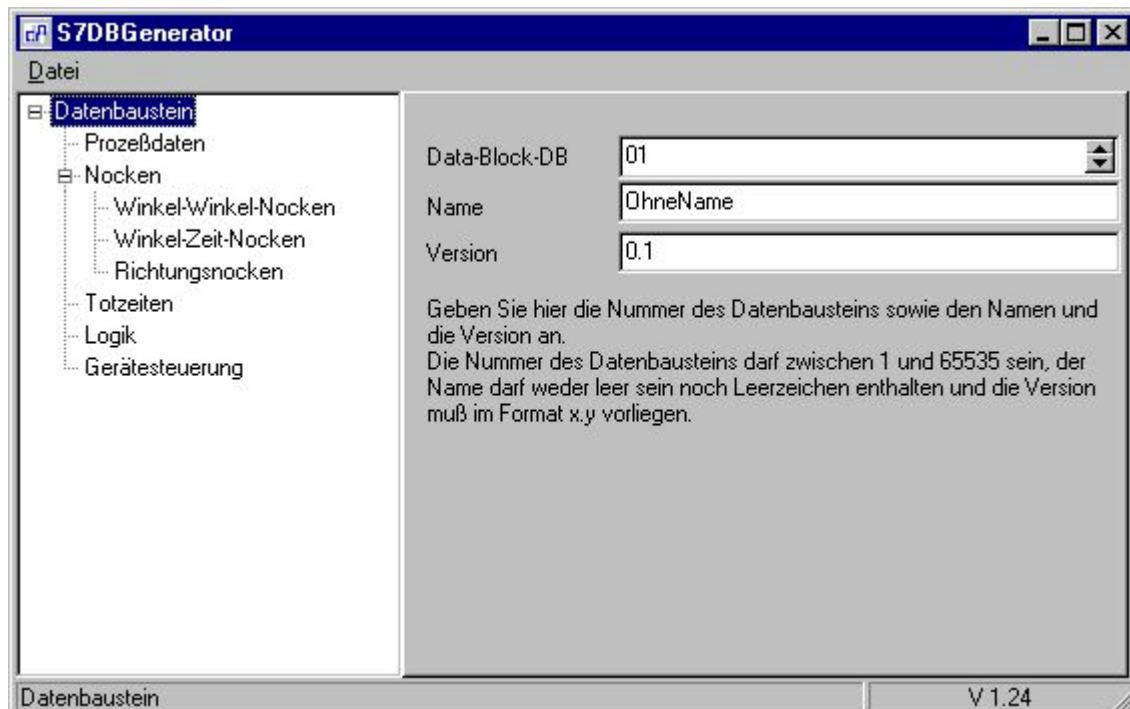
Adresse in Dez	Bezeichnung	Länge	Wert in Dez
0	Offset_Type	WORD	1
2	Offset_Length	WORD	12
4	Offset_Address	WORD	0
6	Cam_ST_Type	WORD	3
8	Cam_ST_Length	WORD	6
10	Cam_ST_Address	WORD	12
12	Cam_ST_Cam (1). ProgNo	BYTE	0
14	Cam_ST_Cam (1).Output	BYTE	4
16	Cam_ST_Cam (1).On	WORD	20
18	Cam_ST_Cam (1).Off	WORD	40

In diesem Beispiel wird der Ausgang 4, Cam\_ST\_Cam(1).Output im Programm 0, Cam\_ST\_Cam(1).ProgNo., zwischen Position 20, Cam\_ST\_Cam(1).On und 40, Cam\_ST\_Cam(1).Off gesetzt. Der Wert Cam\_ST\_Length ist die Anzahl der Bytes und berechnet sich aus der Anzahl der Nocken mal sechs.

Offset\_Length ist die Anzahl Bytes der Offset Parameter, fängt immer bei Null an und endet in diesem Beispiel bei Adresse 10.

## 6.20 Datenbaustein-Generator

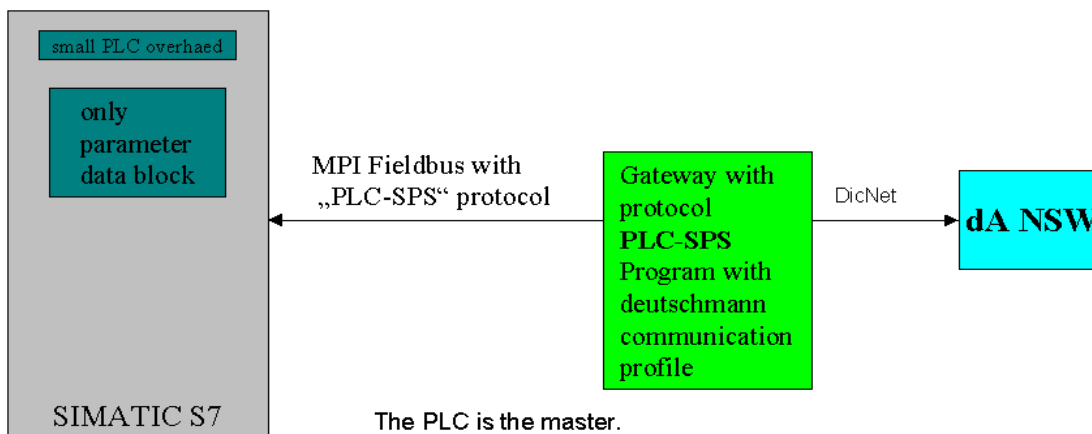
Der Generator erzeugt eine gewünschte Parameter-Tabelle automatisch im AWL-Format. Die Berechnung der Length-Werte und Start-Adressen wird dem Anwender dadurch abgenommen. Das Programm kann aus dem Download-Bereich unserer Homepage <http://www.deutschmann.de> geladen werden.



## 7 Anbindung Nockensteuerung über MPI UNIGATE an S7 300 - 400

### 7.1 Zubehör

- S7 300 - 400
- MPI Gateway (ab V1.3)
- 1 MPI-Buskabel (RS485) mit Stecker
- aktuelle WINGATE Software (zum Konfigurieren des Gateways)
- aktuelle WINLOC Software (erstellt Datenbaustein) oder Datenbausteingenerator
- Verbindungskabel zwischen PC und Gateway (RS232 D-SUB auf 3 pol. Phönix)
- S7 Projekt oder Referenz-Datenbaustein und Nockensteuerung-Datenbaustein als Datei



Anschluß-Schema

### 7.2 Inbetriebnahme

#### 7.2.1 Gateway konfigurieren

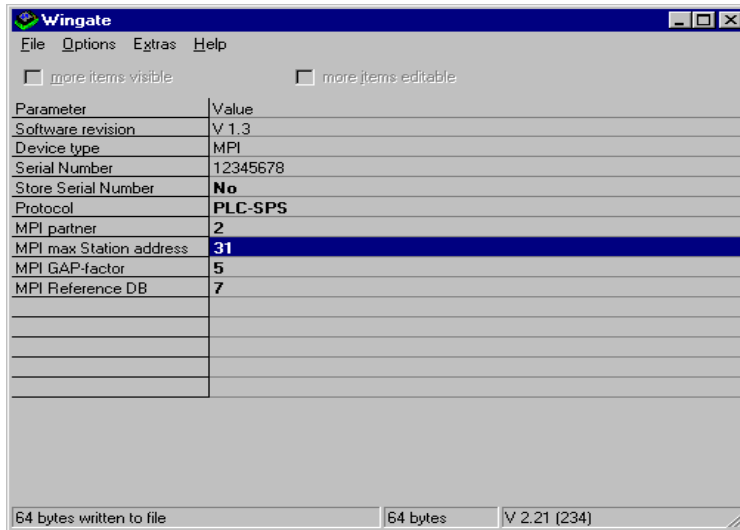
- Gateway mit dem PC über serielle Schnittstelle verbinden: Schnittstelle des Gateways auf RS232 einstellen - (Pinbelegung für dieses Kabel siehe Handbuch)

##### 7.2.1.1 Konfigmode

- Am Gateway zur Konfiguration die Drehschalter S4 und S5 auf der RS Seite auf "F" einstellen
- Gateway durch Spannung AUS/EIN neustarten
- "State" LED blinkt jetzt rot, "Power" LED leuchtet grün
- WINGATE Software auf dem PC starten. Ist das Gateway an den PC angeschlossen worden und liegt an Spannung, wird mit dem Starten der WINGATE Software ein automatischer Upload durchgeführt. D. h. die aktuelle Konfiguration des ausgelieferten Gateways wird geladen - in der Regel ist das Gateway auf das Protokoll Transparent eingestellt.

Sollte es Probleme mit dem automatischen Upload geben, kann dieser auch manuell durchgeführt werden (siehe Beschreibung zum Gateway bzw. zur WINGATE Software).

- In der WINGATE Software das Protokoll Transparent abändern auf Protokoll "PLC - SPS"
- Bei MPI Partner (SPS) die MPI Adresse der SPS eintragen. In der Regel ist die MPI Adresse der SPS auf >> 2 << eingestellt.

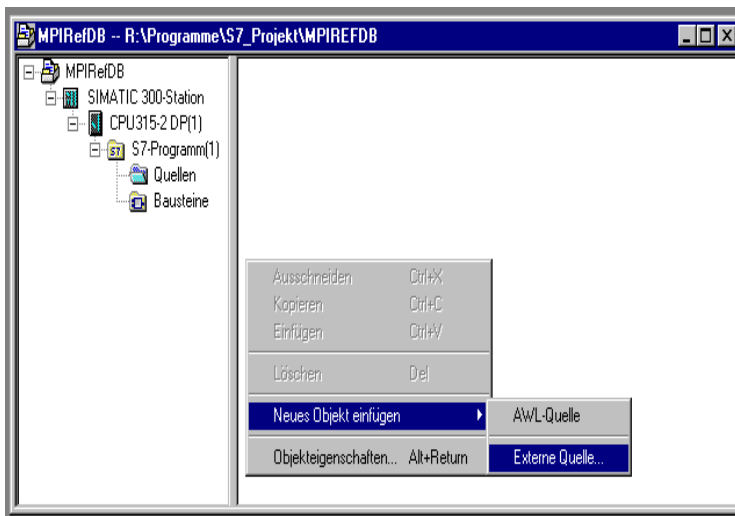


## 7.2.2 Beispiel für die Einstellungen

- Maximale Stationsadresse sollte defaultmäßig auf >> 31 << eingestellt sein. Diese bitte beibehalten.
- GAP Factor sollte defaultmäßig auf 5 eingestellt sein.
- MPI Referenz DB (DB = Datenbaustein) angeben ( im Beispiel >> DB7 <<).
- Achtung: RefDB muß mit dem S7 Projekt übereinstimmen.
- Konfiguration abspeichern. File -> Save As, nicht zwingend.
- Konfiguration zum Gateway downloaden. File -> Download.
- WINGATE beenden.
- Drehschalter S4 und S5 des Gateways auf der RS Seite auf >> 00 << einstellen.
- Am Gateway die MPI Adresse auf der MPI Schnittstellen Seite durch die Drehschalter einstellen, z. B. Adresse >> 3 << Drehschalter Low auf >> 3 << und Drehschalter High auf >> 0 << . Maximal zulässige Adresse ist >> 31 <<.
- RS485 oder RS232, abhängig von der Schnittstelle der Nockensteuerung.
- S7 Projekt in Step7 PC-Software laden. Es sind zwei Bausteine enthalten - NSDB.awl und RefDB.awl.
- Das Gateway mit der SPS S7 über die MPI Schnittstelle mit dem MPI-Kabel verbinden.
- Gateway durch Spannung AUS/EIN neustarten.
- MPI State LED leuchtet grün. Kommunikation mit S7.
- MPI Power leuchtet grün.
- MPI Control leuchtet grün.

oder

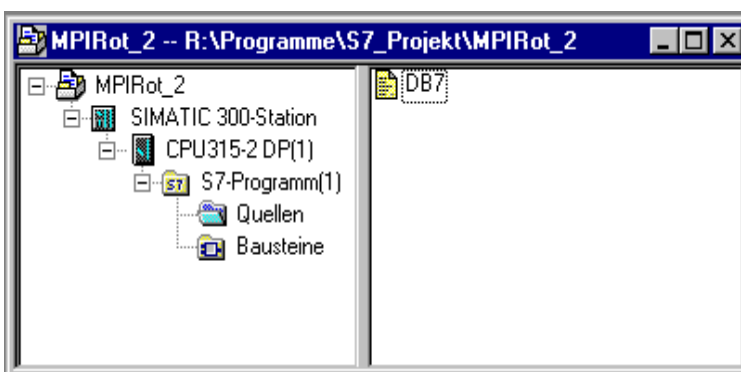
- Die Bausteine über die STEP7 Funktion externe Quelle einfügen.
- RefDB einfügen und übersetzen.



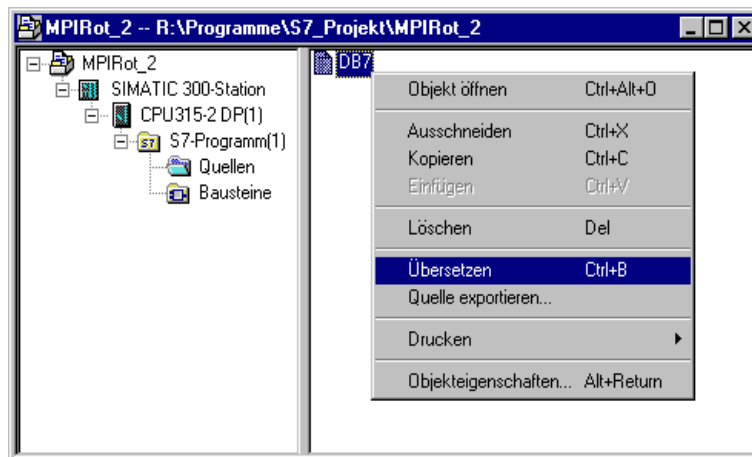
AWL Datei in STEP7 importieren (Ordner "Quelle", rechte Maustaste in rechtes Fenster)



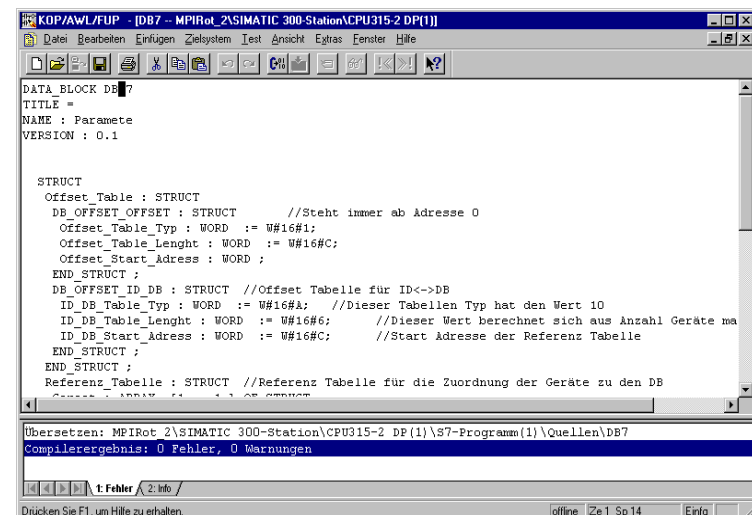
Generierte Quelle aus Ordner wählen



## Importierte Quelle

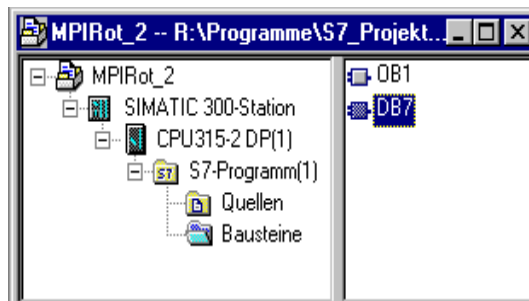


## Importierte Quelle übersetzen (rechte Maustaste)



## Übersetzter Datenbaustein

Nach dem Übersetzen befindet sich im Ordner "Bausteine" der DB Nr 7.



- Ggfs. muß der Ref DB umbenannt werden (in WINGATE wurde >> DB 7 << angegeben, ggfs. muß die RefDB.awl in der STEP7 umbenannt werden).

Adresse	Name	Typ	Anfang	Kommentar
0.0		STRUCT		
+0.0	Offset_Table	STRUCT		
+0.0	DB_OFFSET_OFFSET	STRUCT		Steht immer ab Adresse 0
+0.0	Offset_Table_Typ	WORD	W#16#1	
+2.0	Offset_Table_Lenght	WORD	W#16#C	
+4.0	Offset_Start_Adress	WORD	W#16#0	
=6.0		END_STRUCT		
+6.0	DB_OFFSET_ID_DB	STRUCT		Offset Tabelle für ID->DB
+0.0	ID_DB_Table_Typ	WORD	W#16#A	Dieser Tabellen Typ hat den Wert 10
+2.0	ID_DB_Table_Lenght	WORD	W#16#6	Dieser Wert berechnet sich aus Anzahl Geräte mal 6 Byte
+4.0	ID_DB_Start_Adress	WORD	W#16#C	Start Adresse der Referenz Tabelle
=6.0		END_STRUCT		
=12.0		END_STRUCT		
+12.0	Referenz_Tabelle	STRUCT		Referenz Tabelle für die Zuordnung der Geräte zu den DB
+0.0	Geraet	ARRAY[1..1]		
+0.0		STRUCT		
+0.0	ID	WORD	W#16#1	ID des Nockenschaltwerkes
+2.0	FlagRegister	WORD	W#16#1	Wird LSB 1 gesetzt liest das Gateway den in DB_No definierten DB
+4.0	DB_No	WORD	W#16#1	zugehöriger DB zur ID
=6.0		END_STRUCT		
=6.0		END_STRUCT		
=18.0		END_STRUCT		

#### Deklarationssicht

- NSDB.awl einfügen und übersetzen (obige Schritte wiederholen).
- Ggfs. muß die DBNr. des NSDB.awl umbenannt werden.

### 7.3 Einstellung RefDB

- Im RefDB >> DB 7 << in DB\_No muß jetzt die DB Nummer des NSDB eingetragen werden >> W#16#02<< .
- Je nachdem wieviele Nockensteuerungen angeschlossen werden, muß im Array der Eintrag geändert werden. Bei einer NS muß der Eintrag geändert werden: für eine NS >> Array [1...1] <<, für zwei NS >> Array [1...2] << usw. ..
- Im DB 7 in ID muß jetzt die ID der angeschlossenen Nockensteuerung eingetragen werden >> W#16#01 <<.
- Table Length muß angepaßt werden. Je nach Anzahl der angeschlossenen Nockensteuerungen muß hier die Anzahl der Geräte mit 6 multipliziert werden und dementsprechend der Eintrag vorgenommen werden >> W#16#06 << .
- Bei zwei Nockensteuerungen hat der Eintrag folgendermaßen auszusehen >> W#16#12 <<. In der Nockensteuerung muß jetzt noch die ID >> 1 <<, wie vorher im DB 7 eingetragen, eingestellt werden. Dies geschieht über die Brücken im 9poligen D-Sub Stecker über einen Drehschalter oder über die WINLOC Software.
- Flag Register -> LSB gesetzt >> W#16#01<< dann werden alle Änderungen in den NSDB übertragen.
- Ref DB Nummer über Schalter S5 einstellen.

Adresse	Name	Typ	Anfangswert	Aktualwert	Kommentar
0.0	Offset_Table.DB_OFFSET_OFFSET.Offset_Table_Typ	WORD	W#16#1	W#16#1	
2.0	Offset_Table.DB_OFFSET_OFFSET.Offset_Table_Lenght	WORD	W#16#C	W#16#C	
4.0	Offset_Table.DB_OFFSET_OFFSET.Offset_Start_Adress	WORD	W#16#0	W#16#0	
6.0	Offset_Table.DB_OFFSET_ID_DB.ID_DB_Table_Typ	WORD	W#16#A	W#16#A	Dieser Tabellen Typ hat den Wert
8.0	Offset_Table.DB_OFFSET_ID_DB.ID_DB_Table_Lenght	WORD	W#16#6	W#16#6	Dieser Wert berechnet sich aus A
10.0	Offset_Table.DB_OFFSET_ID_DB.ID_DB_Start_Adress	WORD	W#16#C	W#16#C	Start Adresse der Referenz Tabel
12.0	Referenz_Tabelle.Geraet[1].ID	WORD	W#16#1	W#16#1	ID des Nockenschaltwerkes
14.0	Referenz_Tabelle.Geraet[1].FlagRegister	WORD	W#16#1	W#16#1	Wird LSB 1 gesetzt liest das Gat
16.0	Referenz_Tabelle.Geraet[1].DB_No	WORD	W#16#1	W#16#2	zugehöriger DB zur ID

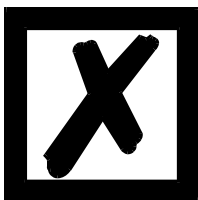
#### Datenansicht des RefDB

- Das Protokoll PLC-SPS wird unterstützt in der LOCON 24-Familie ab V4.03 und in LOCON 1, 2, 16, 17 und ROTARNOCK 1, 2 ab V5.10.
- Das Gateway liest nun zyklisch (Flagregister auf 1) die Daten aus dem Datenbaustein Nr2 (DB\_No auf 2) und vergleicht sie mit dem Inhalt der Nockensteuerung mit der ID 1 (ID auf 1).
- Haben sich die Daten im DB2 geändert, verändert das Gateway die Parameter in der Nockensteuerung automatisch.

Folgende Funktionen werden z. Zt. vom Protokoll PLC-SPS im NSDB unterstützt:

[Process_Table] Position Speed Output15_0 Act_Prog ErrorNo	[Cam_Table] ProgNo Output On Off	[IDL_Table] ProgNo OutputIdleT_On IdleT_Off	[CMD_Table] New_Prog TeachIn Invert_Encoder Quit_Error
---	--	--	--

Die NSDB Datei kann mit dem Datenbausteingenerator automatisch erzeugt werden und wie oben beschrieben importiert werden.



#### Achtung:

Es ist möglich, dass das Gerät nach einem Download in die S7 nicht mehr reagiert.

Die Ausgänge werden dann nicht mehr abgearbeitet und die Position nicht mehr übertragen.

Das Gerät muß dann neugestartet werden.

## **8 Anbindung LOCON 24 Nockensteuerungen über MPI an SPS**

### **8.1 Protokoll PLC**

Default: eigene MPI Adresse: 3

MPI-Adresse der S7: 1

Verwendeter Datenbaustein (RefDB): Nr. 7

LOCON Geräte ID: 0

Geräte DB No: 12 (0x0C)

Die Termination am LOCON 24, R + mit Dic + und R - mit Dic- muß unbedingt aktiviert werden, wenn kein weiteres Gerät am DICNET angeschlossen ist.

Prog Enable mit + 24 V verbinden (vgl. auch im Handbuch zum LOCON 24 das Kap.: „Elektrische Anschlüsse LOCON 24, 48, 64“).



## 9 Anbindung LOCON 24 Nockensteuerungen über PROFIBUS an SPS

### 9.1 Protokoll: DICNET

Siehe Bedienerhandbuch "Kommunikationsprofil für Nockensteuerungen der Deutschmann Automation"

- GSD Datei: Dagw2079.gsd
- Module: "DEUTSCHMANN (Parameter only)" 0xBC
- Module: "DEUTSCHMANN (Para., Pos, Speed)" 0xBC, 0x51

Im Auslieferungszustand ist die DICNET-ID auf 0 eingestellt.

### 9.2 Protokoll: PLC-SPS

siehe dieses Bedienerhandbuch

- GSD Datei: Dagw2079.gsd
- Module: "DEUTSCHMANN (PLC-CSU-ST)" 0xB7,0x97
- Module: "DEUTSCHMANN (PLC-CSU Long)" 0xB7,0x9B



**Achtung:**

Nach einem Kaltstart werden alle Werte in der Nockensteuerung gelöscht.

Die SPS ist Master, es kann somit nur über die SPS die Programmierung vorgenommen werden.

### 9.3 Inbetriebnahme

Profibus Slave ID: 126 Dez

Das Protokoll PLC-SPS wird unterstützt in der LOCON 24-Familie ab V4.03 und in LOCON 1, 2, 16, 17 und ROTARNOCK 1, 2 ab V5.02. Ein Beispielprojekt der S7-Anwendung kann aus dem Download-Bereich unserer Internetseite <http://www.deutschmann.de> heruntergeladen werden.

## 10 Anbindung ROTARNOCK Nockensteuerungen über PROFIBUS an eine SPS

### 10.1 Protokoll: PLC-SPS

Wie in diesem Bedienerhandbuch beschrieben

- GSD Datei für ROTARNOCK 1, 2, 3: R2pb2935.gsd
- Modul: "S7DB+Proc.Data" 0xA7, 0x97, 0x97
- Modul: „S7DB+Proc.DataLong“ 0xB7, 0x9B
- GSD Datei für ROTARNOCK 4: R4pb3231.gsd
- Modul: „S7DB+Proc.Data“ 0xB7, 0xD1, 0x50, 0x9B
- Modul: „S7DB+Proc.Data+Logic“ 0xB7, 0xD1, 0x50, 0x9B, 0xA1



#### Achtung:

Nach einem Kaltstart werden alle Werte in der Nockensteuerung gelöscht.

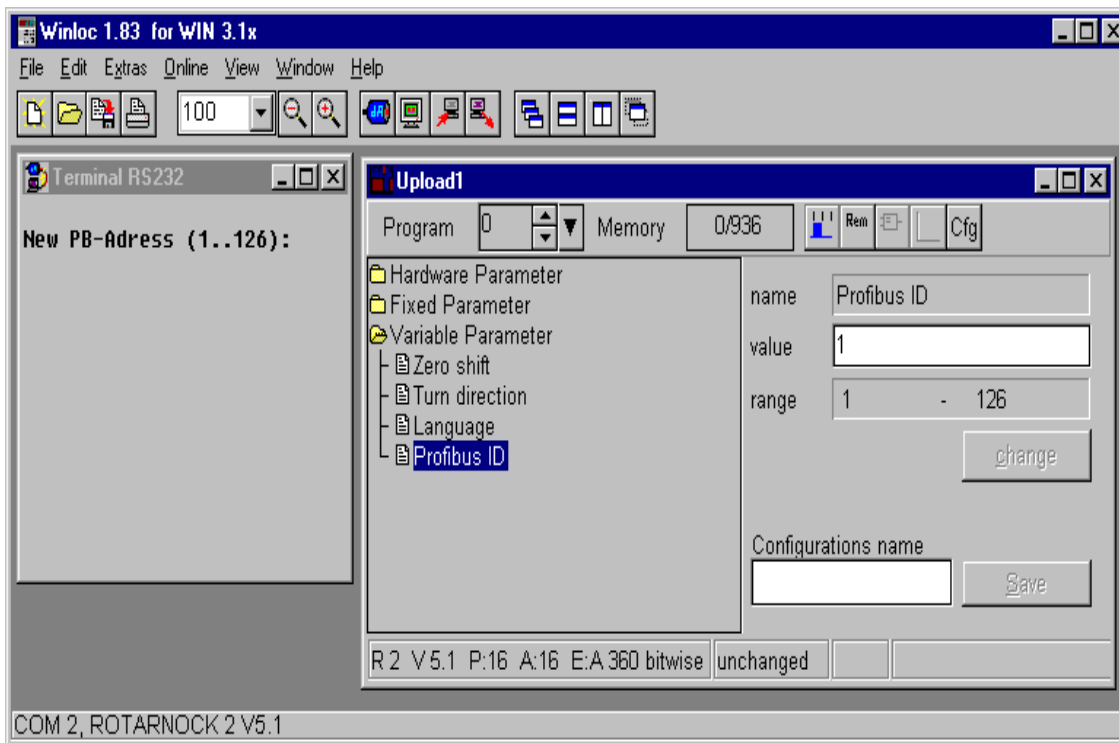
Die SPS ist Master, es kann somit nur über die SPS die Programmierung vorgenommen werden.

### 10.2 Inbetriebnahme

Profibus Slave ID: 126 Dez default

Zum Parametrieren kann das Gerät über eine RS232 an einen PC mit Terminal Programm umkonfiguriert werden (STRG + N).

Komfortabler lässt es sich mit der aktuellen WINLOC Software konfigurieren.



Wird die maximale Parameter Tabellengröße von 2048 Byte überschritten wird der Error 5 ausgegeben.

Ein Beispielprojekt der S7-Anwendung kann von unserer Internetseite <http://www.deutschmann.de> im Downloadbereich heruntergeladen werden.

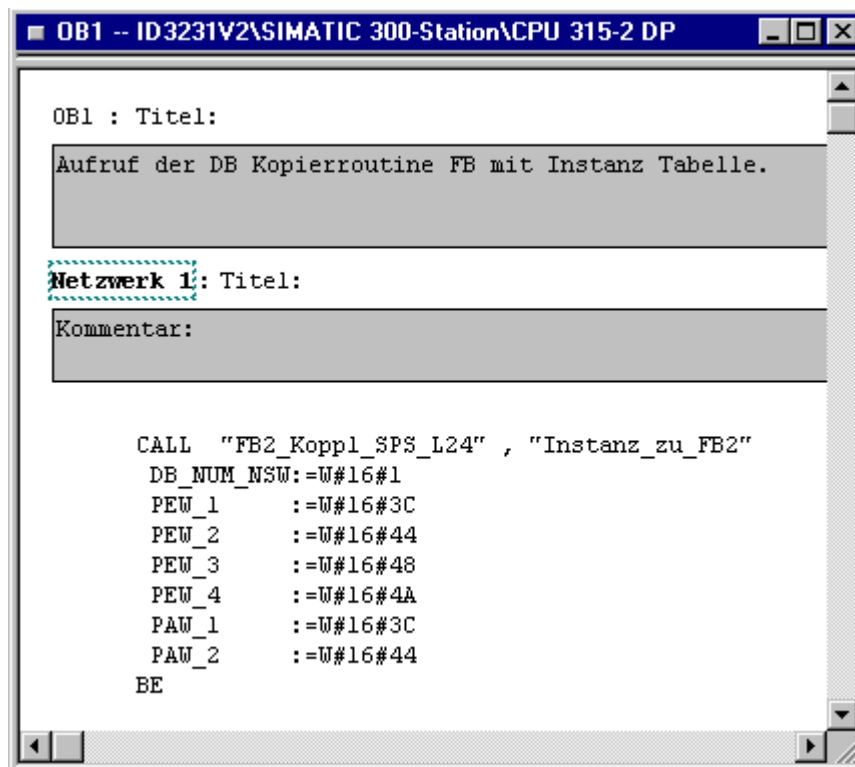
### 10.3 Anbindung ROTARNOCK 4 an den PROFIBUS

Für dieses Gerät gilt die GSD-Datei: r4pb3231.gsd.

Das Gerät hat immer 64 Ausgänge, von denen die ersten 16 hardwaremäßig am Stecker anliegen. Mit dem DB-Typ "PROCESSDATA\_80\_TYPE" können alle 64 Ausgänge im Profibus dargestellt werden.

Die Logikeingänge werden ebenfalls über den PROFIBUS von der SPS zur Nockensteuerung übertragen. Es handelt sich dabei um das 5. Wort im PROFIBUS.

S7 Beispiel für die Zuweisung der E/A Adressen im OB1



...und im Hardware Konfigurator

HW Konfig - [SIMATIC 300-Station (Konfiguration) -- ID3231V2]

Station Bearbeiten Einfügen Zielsystem Ansicht Extras Fenster Hilfe

(0) LR

1					
2	CPU 315-2 DP				
x2	DP-Master				
3					

NSW-DP: DP-Mastersystem (1)

(8) ROTAR

Profil: Standard

- SIPOS
- Weitere FELDGERÄTE
  - I/O
  - Gateway
  - Allgemein
  - SPS
    - ROTARNOCK Profibus-DP
      - ROTARNOCK 4 Profibus-DP**
        - Universalmodul
      - VS710 Variable
    - Encoder
    - Antriebe
    - Kompatible Profibus-DP-Slaves

(8) ROTARNOCK 4 Profibus-DP

Ste...	Bau...	Best...	E-Adr...	A-Adr...	Kommentar
0	183	Univers	60...67	60...67	
1	209	Univers	68...71		
2	1AE	Univers	72...73		
3	155	Univers	74...85		
4	161	Univers		68...69	

V3231

## 11 Anbindung ROTARNOCK Nockensteuerungen über MPI an S7 300 - 400

### 11.1 Datenaustausch

Der Datenaustausch erfolgt automatisch über den konfigurierten Datenbaustein (DB), in diesem Handbuch beschrieben als Parameter Tabelle.

Der DB kann automatisch mit dem DB-Generator erzeugt werden.

Der Aufbau des DB ist wie in diesem Handbuch beschrieben.



**Achtung:**

Nach einem Kaltstart werden alle Werte in der Nockensteuerung gelöscht.

Die SPS ist Master, es kann somit nur über die SPS die Programmierung vorgenommen werden.

### 11.2 Inbetriebnahme

MPI ID: 3 Dez default

MPI Partner Address: 1

MPI GAP Factor: 5 default

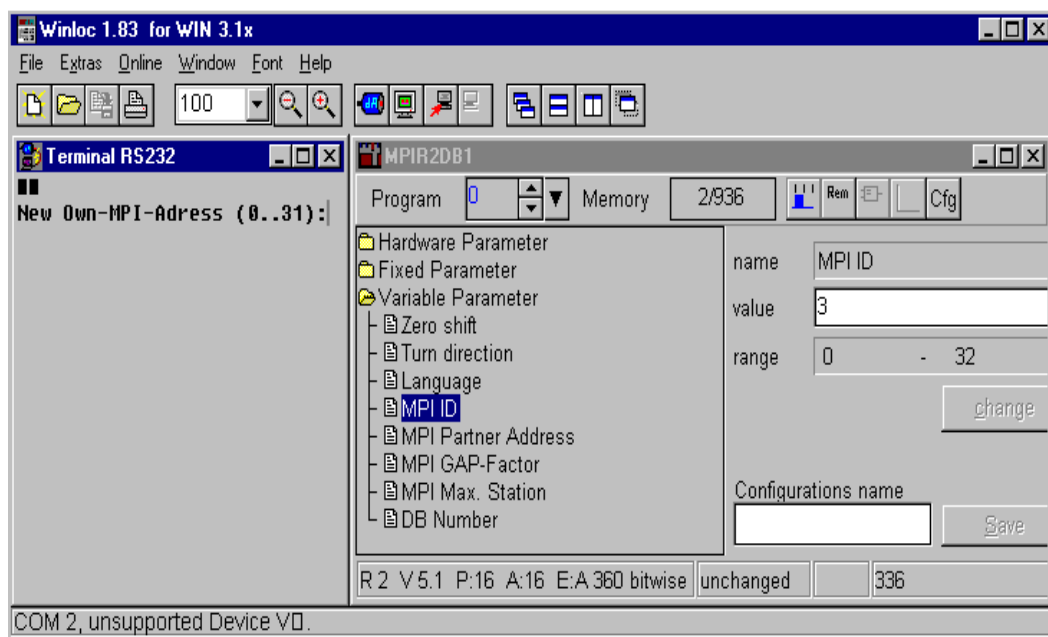
MPI Max Station: 31 default

DB Number: 13 default

Zum Parametrieren kann das Gerät über eine RS232 an einen PC mit Terminal Programm umkonfiguriert werden (STRG + N).

Komfortabler last es sich mit der aktuellen WINLOC Software konfigurieren. Bitte beachten Sie: WINLOC enthält ein Terminalprogramm (s. Bild). Das Gerät kann aber auch über eine RS232 an einen PC mit einem x-beliebigen Terminalprogramm umkonfiguriert werden.

Wird die maximale Parameter-Tabellengröße von 2048 Byte überschritten, wird der Error 5 ausgegeben.



### 11.3 Historie

Gerät	LOCON 24-DP mit Profibus Software				ROTARNOCK-DP				UNIGATE MPI				LOCON 24-MPI				MT-ROTARNOCK-MPI				R100		L100	
					1, 2, 3 4																			
unterstützte Tabellen Typen	V3.3		V3.4		V5.1		V5.02		V1.4								V5.21							
1	x	x			x		x		x				x				x				x		x	
2	x	x			x				x				x											
3	x	x			x		x		x				x								x		x	
4	x	x			x		x		x				x				x				x		x	
5	x	x			x		x		x				x				x				x		x	
6		x															x							
7		x															x				[x]		x	
8		x					x										x				x		x	
9 Reserve																								
A									X															
B		x					x										x				x		x	
C		x					x														x		x	
D																								

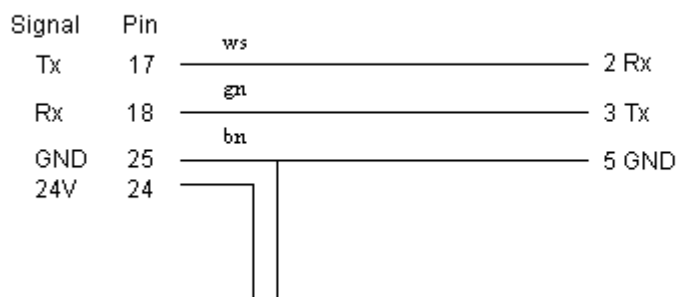
### 11.4 Programmierkabel für ROTARNOCK

Das Programmierkabel für ROTARNOCK PROFIBUS/MPI ist erhältlich unter der Artikelnummer V3467-n\* , wobei n\* = Kabellänge in m.

Dieses Kabel ist zur Konfiguration der Feldbusparameter (ID, DB-Nummer usw.) über die RS232 Schnittstelle erforderlich.

ROTARNOCK (25pol. D-SUB)

PC (9pol. D-SUB)



Extern muss eine Spannungsversorgung auf die Pins 24 (24V) und 25 (Gnd) hergestellt werden.

## **12 PROFIBUS- und MPI-ID bei LOCON und ROTARNOCK einstellen**

### **12.1 PROFIBUS-ID einstellen**

Die PROFIBUS Adresse ist auf 126 Dez default voreingestellt (wie bereits in Kapitel 9.3 "Inbetriebnahme" und Kapitel 10.2 "Inbetriebnahme" beschrieben).

Zum Ändern der ID gibt es folgende Möglichkeiten:

#### **12.1.1 Beispiel für S7 mit PC-Adapter**

Der PROFIBUS-Slave wird direkt mit dem Programmieradapter an den PC angeschlossen. Über ein Projektierungstool z. B. Step 7 Software kann dann im Simatic „Manager-Zielsystem-Profi-bus-Adresse ändern“, die ID geändert werden.

#### **12.1.2 Beispiel über RS232-Schnittstelle**

Die nachfolgend beschriebene Vorgehensweise gilt nur für ROTARNOCK. Das Gerät wird über den 25 pol. Stecker mit der RS232-Schnittstelle eines PC's verbunden. (vgl. auch Kapitel „Grundgerät ROTARNOCK“, Unterkapitel „25-pol. D-SUB“ im Bedienerhandbuch „Elektronische Nockensteuerung ROTARNOCK 1, 2, 3...“). In einem einfachen Terminalfenster z. B. in WINLOC kann über die Tastensteuerung „Crlg + N“ bzw. „Strg + N“ die ID geändert werden oder über das „Config“ Fenster in WINLOC. Dies gilt nicht bei der ROTARNOCK Version 5.3.

### **12.2 MPI-ID einstellen**

Die MPI Adresse ist auf 3 Dez default voreingestellt.

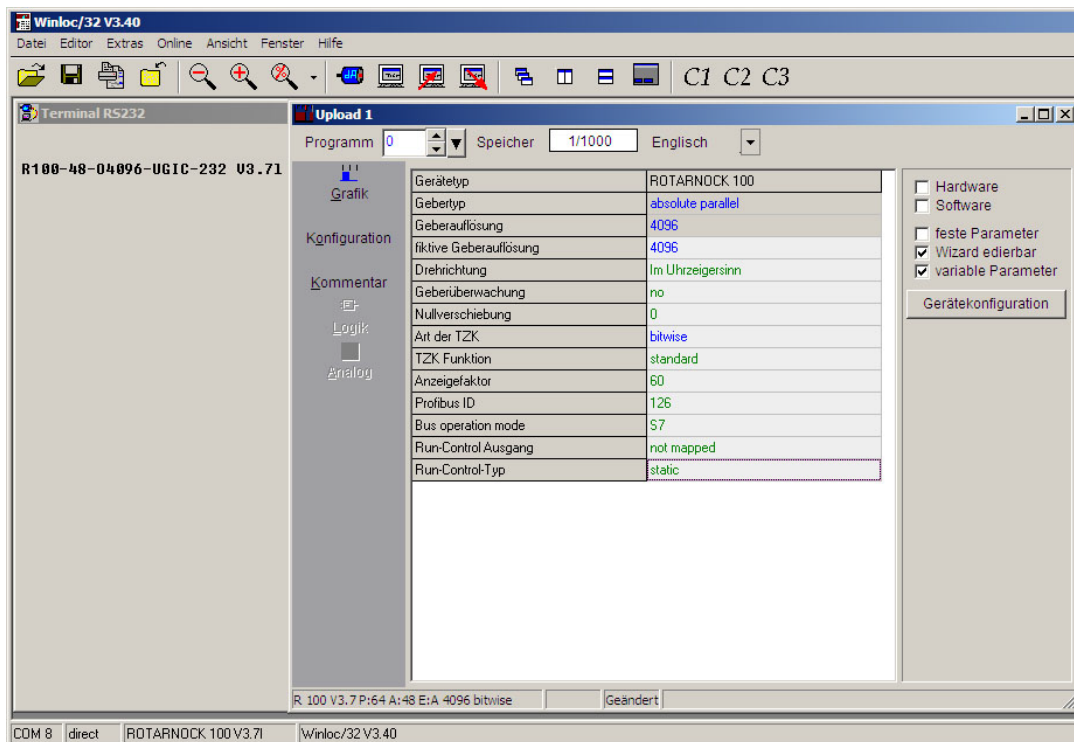
Zum Ändern der ID siehe Kapitel 12.1.2 "Beispiel über RS232-Schnittstelle".

## 13 Anbindung ROTARNOCK Nockensteuerungen über PROFINET an eine S7

### 13.1 Inbetriebnahme

Zum Parametrieren kann das Gerät über eine RS 232-Schnittstelle an ein TERM oder an einen PC mit WINLOC32 angeschlossen werden.

Mit der aktuellen Version von WINLOC32 lässt sich das Gerät komfortabel konfigurieren.



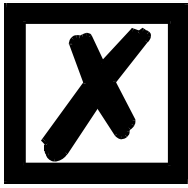
#### Achtung:



Die Anschlussbelegung für das ROTARNOCK 100-PN kann dem Bedienerhandbuch ROTARNOCK 100 unter Kapitel „Anschlussbelegung ROTARNOCK“ entnommen werden.



## 13.2 Anbindung ROTARNOCK 100 an PROFINET



### Achtung:

Die Beschreibungen in den beiden folgenden Kapiteln beziehen sich auf das Beispielprojekt, das auf der Support DVD und auf unserer Homepage zur Verfügung steht.

Zunächst muss die GSDML-Datei (**Download Link: [GSDML-ROTARNOCK100.zip](#)**) importiert werden. Diese kann über die Webseite des ROTARNOCK 100-PN (PROFINET), die beiliegende Support DVD oder über die Homepage [www.deutschmann.de](http://www.deutschmann.de) geladen werden. Danach kann das Gerät ausgewählt und eingefügt werden. Der Adressbereich, der von der SPS automatisch zugewiesen wird, muss dann mit den im DB3 hinterlegten PEW und PAW, die im OB1 verwendet werden, übereinstimmen. (Bild 13.1.1 und Bild 13.1.2)

Gerätename: V3795

DEUTSCHMANN Integrierte elektronische Nockensteuerung

[GSDML-V2.3-Deutschmann-ROTARNOCK100-PN-20150317-16500.xml](#)

(GSDML-Datei-Stand: 03/15)

**Bild 13.1.1:**

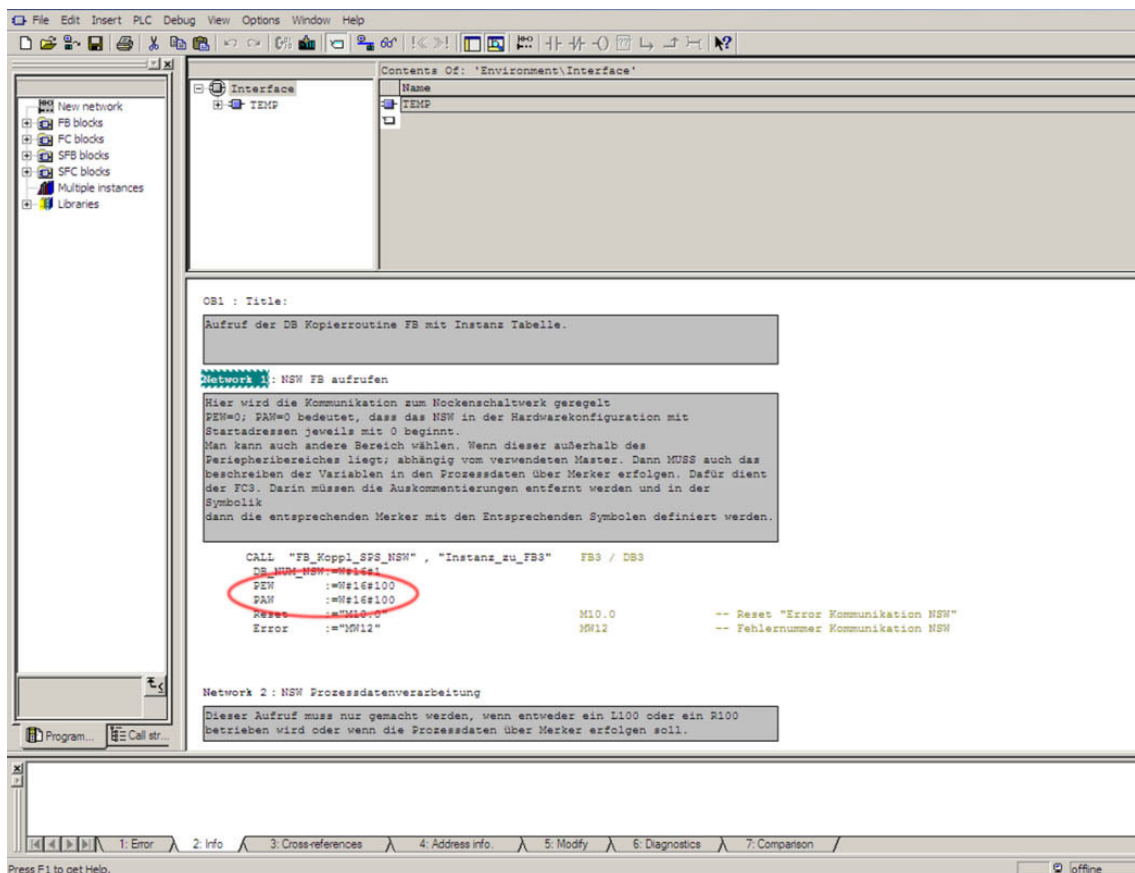
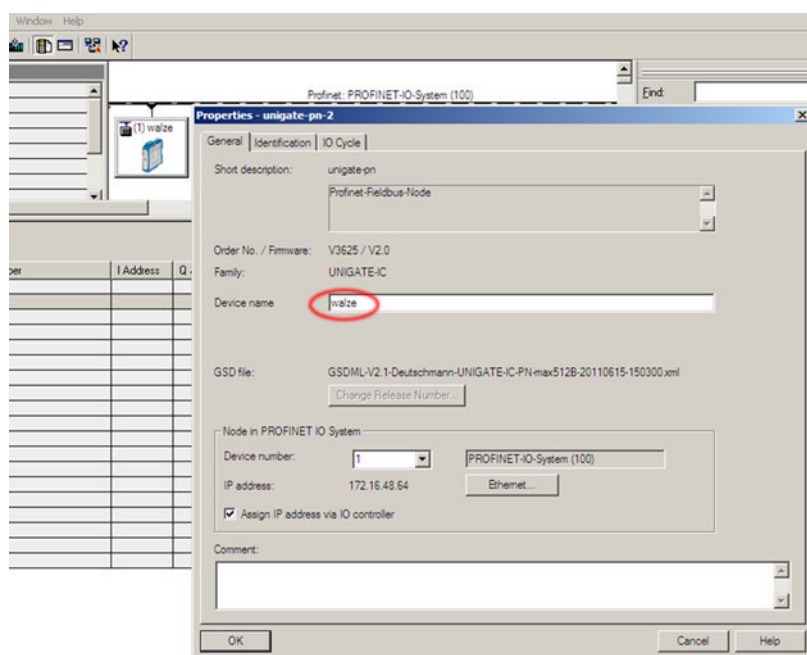


Bild 13.1.2:

■ DB3 -- "Instanz_zu_FB3" -- R100_FB_Profinet_1port\SIMATIC 315-2 DP/PN\CPU 315-2 PN\DP\...\DB3					
Address	Declaration	Name	Type	Initial value	Comment
0.0	in	DB_NUM_NSW	WORD	W#16#0	DB-Nummer des NSW Datenbausteins
2.0	in	PEW	WORD	W#16#0	Prozessdaten Eingang 1 - Anfangsadresse
4.0	in	PAW	WORD	W#16#0	Prozessdaten Ausgang 1 - Anfangsadresse
6.0	in	Reset	BOOL	FALSE	Baustein Reset, Datenübertragung neu initialisieren
8.0	out	Error	WORD	W#16#0	Fehlerrückmeldung des Bausteins
10.0	stat	Zaehler	WORD	W#16#0	Datenwortzähler
12.0	stat	PDATA_IN_1	ARRAY[0..3]		
*2.0	stat		WORD		
20.0	stat	PDATA_OUT_1	ARRAY[0..3]		
*2.0	stat		WORD		
28.0	stat	ErwarteterSlaveEmpfang	WORD	W#16#0	Erwartete Empfangsbestätigung des Slaves (1.Wort)
30.0	stat	answer_Count	WORD	W#16#0	wird hochgezählt, solange keine Antwort vom NSW da ist
32.0	stat	DB_Checked	BOOL	FALSE	Datenbaustein geprüft, nicht nochmal prüfen

In den Eigenschaften kann unter anderem, der Gerätename geändert werden.



### 13.3 PROFINET GeräteName

Hinweis: Im Auslieferungszustand besitzt das Gateway noch keinen Gerätenamen!  
Der GeräteName wird über die Projektierungssoftware dem Gateway zugewiesen. Alternativ dazu kann der GeräteName auch über FTP (Datei „devname.txt“) geändert werden.

Für den Gerätenamen gelten, der PROFINET Spezifikation entsprechend, folgende Regeln:

- \* Er besteht aus einem oder mehreren durch einen Punkt getrennten Namensteilen.
- \* Die Gesamtlänge beträgt 1 bis 240 Zeichen.
- \* Die Länge eines Namensteils beträgt 1 bis 63 Zeichen.
- \* Ein Namensteil besteht ausschließlich aus Kleinbuchstaben, Zahlen und dem Bindestrich.
- \* Weder das erste noch das letzte Zeichen eines Namensteils ist ein Bindestrich.
- \* Der erste Namensteil beginnt nicht mit "port-xyz" oder "port-xyz-abcde", wobei a, b, c, d, e, x, y und z Ziffern sind.
- \* Er hat nicht die Form "k.l.m.n", wobei k, l, m und n Zahlen zwischen 0 und 999 sind.

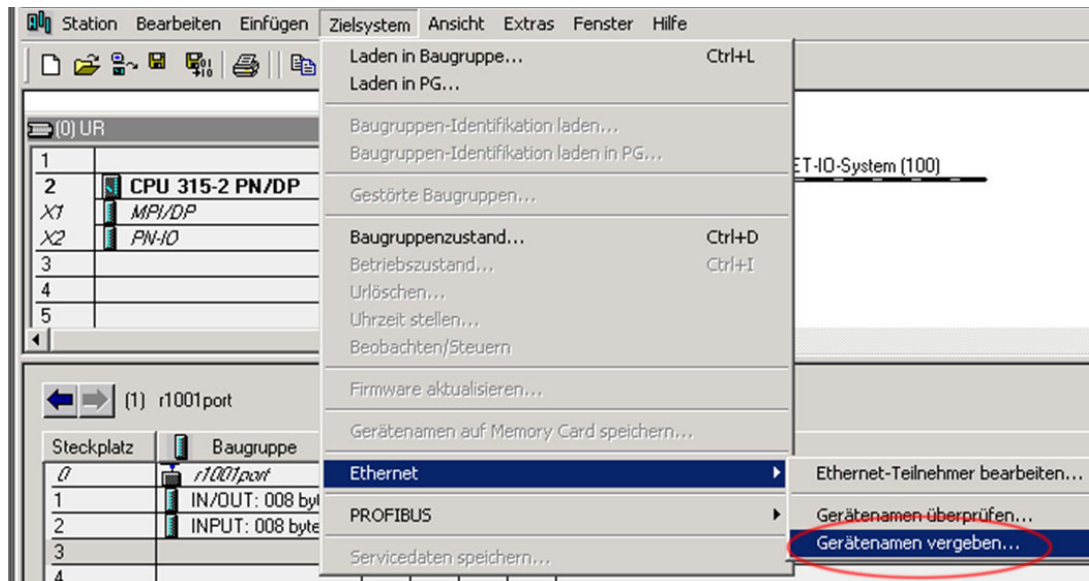
Nachdem nun das Gerät in das Projekt eingefügt wurde, muss der GeräteName noch im jeweiligen Gerät hinterlegt werden. Dieser muss mit dem derzeitigen Gerätenamen, der im Eigenschaftfenster des Gerätes eingetragen wurde, übereinstimmen, damit das Gerät anschließend vom Master erkannt werden kann.

Es gibt zwei Möglichkeiten den Gerätenamen dem entsprechenden Gerät zuzuweisen.

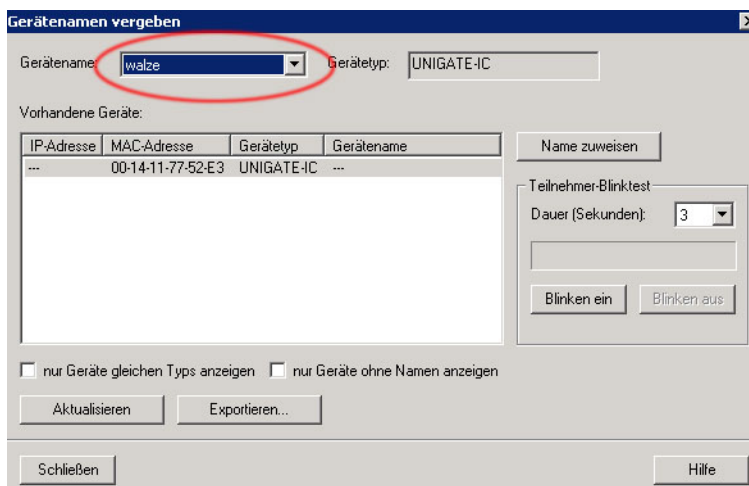
### 1. Möglichkeit: In der Hardwarekonfiguration.

Zunächst muss das Gerät in der Hardwarekonfiguration markiert werden.

Danach über den Menüpunkt „Zielsystem“ unter Ethernet den Punkt „Gerätenamen vergeben“ auswählen.

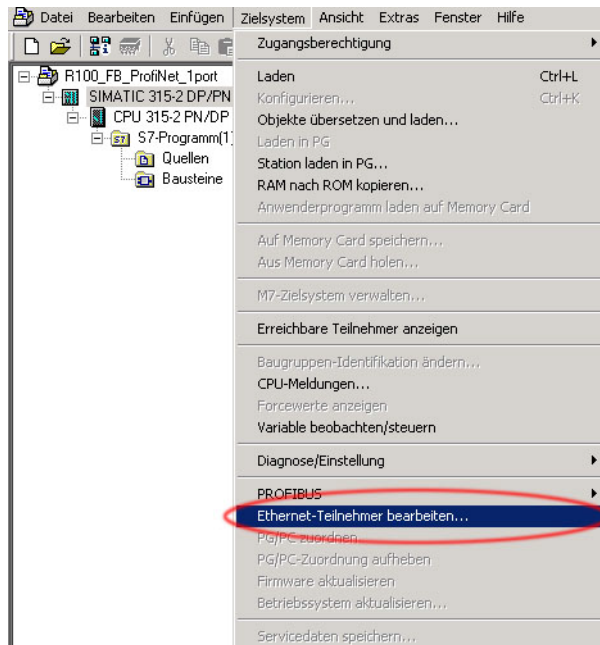


Im darauffolgenden Fenster kann dann der gewünschte Gerätename über das Dropdown-Menü ausgewählt und anschließend über den Button „Name zuweisen“ zugewiesen werden.

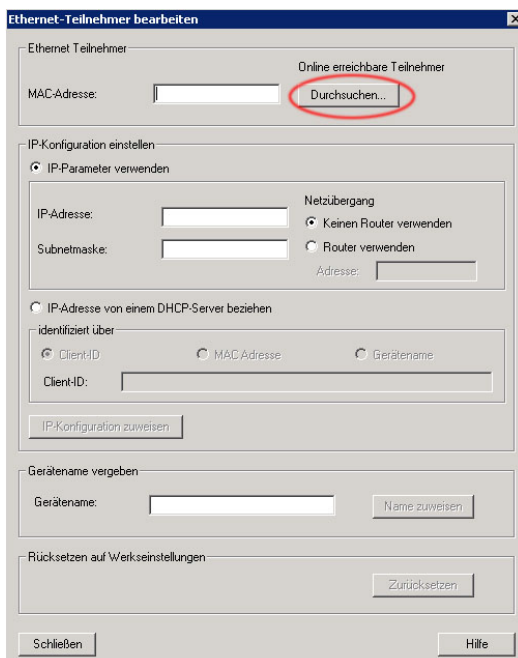


## 2. Möglichkeit: In der Projektansicht.

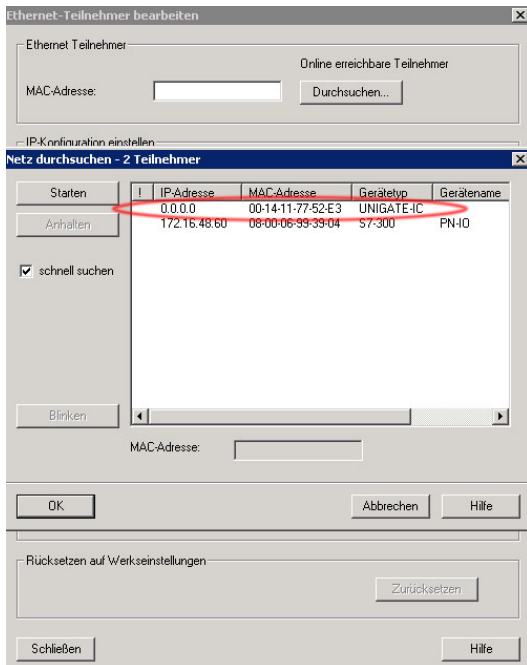
Im Projekt über den Menüpunkt „Zielsystem“ den Punkt „Ethernet-Teilnehmer bearbeiten...“ auswählen.



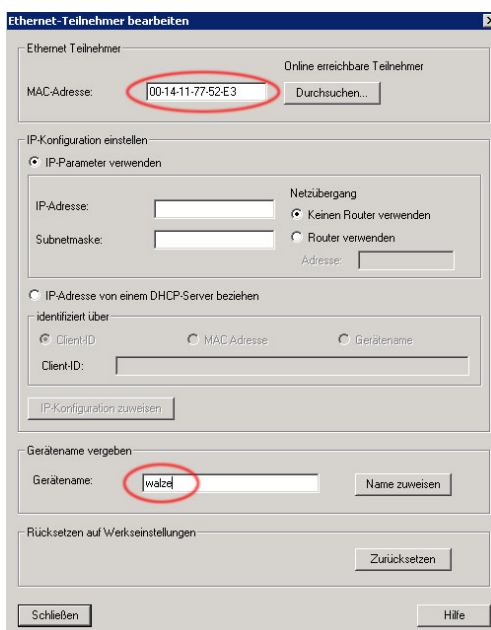
Im darauffolgenden Fenster werden, durch einen Klick auf den Button „Durchsuchen“, alle PROFINET Teilnehmer des Busses angezeigt.



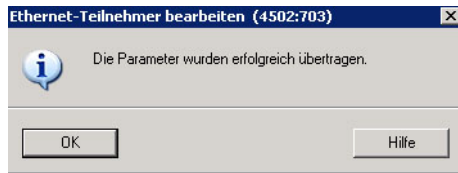
Aus der Liste der gefundenen Teilnehmer kann nun über die angezeigten MAC-Adressen, der gewünschte Teilnehmer ausgewählt, markiert und mit dem Button „OK“ bestätigt werden. Die MAC-Adresse des ROTARNOCK 100 kann dem Typenschild entnommen werden.



In dem darauffolgenden Fenster wird nun die MAC-Adresse des ausgewählten ROTARNOCK 100 angezeigt. Im unteren Bereich dieses Fensters, unter „Gerätenamen vergeben“, kann nun der Gerätenamen hinterlegt werden, der zuvor in den Eigenschaften vergeben worden ist. Durch Klicken des Buttons „Name zuweisen“ wird dieser dann übertragen.



Abschließend sollte dann ein Info-Fenster erscheinen, mit dem Inhalt, dass die Parameter erfolgreich übertragen wurden.



Wenn nun das gesamte Projekt geladen und übertragen wurde, sollte, nach dem Neustart der CPU, die grün blinkende LED am ROTARNOCK 100 statisch grün leuchten.

### 13.4 LED-PN

Die grüne LED zeigt den Bus-Status an und kann folgende Zustände haben:

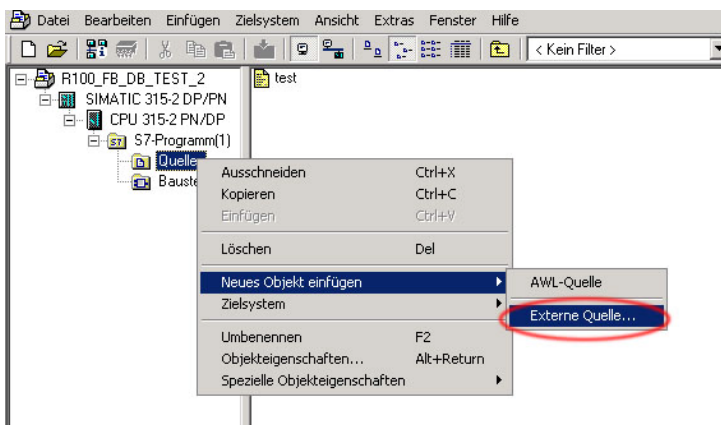
Zustand	LED	Status
Off	-	Bus nicht gestartet
Flash	grün/gelb/rot	10 Hz Error
Flash	grün	2 Hz Bus gestartet, warten auf Verbindung / Konfigurationsmodus
Flash	grün/gelb/rot	1 Hz PROFINET Teilnehmer-Blinktest
On	grün	Verbindung hergestellt

### 13.5 Programmierung der Nocken

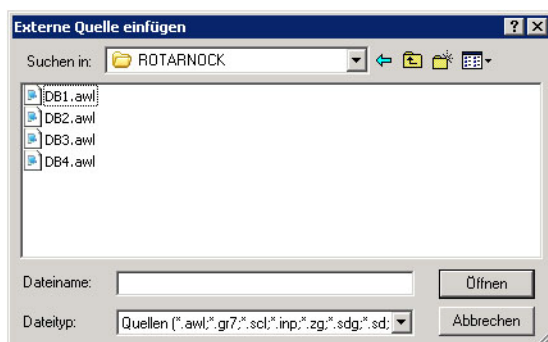
Im DB1 sind alle benötigten Parameter hinterlegt, sodass lediglich die Parametertabelle im AWL-Format, die über WINLOC32 mittels Datenbausteingenerator erzeugt wird, eingebunden werden muss.

Es besteht die Möglichkeit die komplette Programmierung der benötigten Nocken sowie Programme über WINLOC32 durchzuführen. Diese Parameter müssen dann lediglich als Parametertabelle im AWL-Format umgewandelt werden. Dazu wird die gleiche Vorgehensweise verwendet, mit der auch ein jungfräulicher DB erzeugt wird, mit der Ausnahme, dass direkt nach dem Aufruf der Menüpunkt „Datei“ -> „Erzeugen“ angewählt werden muss. Im Anschluss erhält man eine AWL-Datei, die direkt in das Projekt eingebunden werden kann.

Zunächst muss im Projektbaum „Quellen“ markiert sein. Mittels rechter Maustaste gelangt man über den Punkt „Neues Objekt einfügen“ zu „Externe Quelle...“.

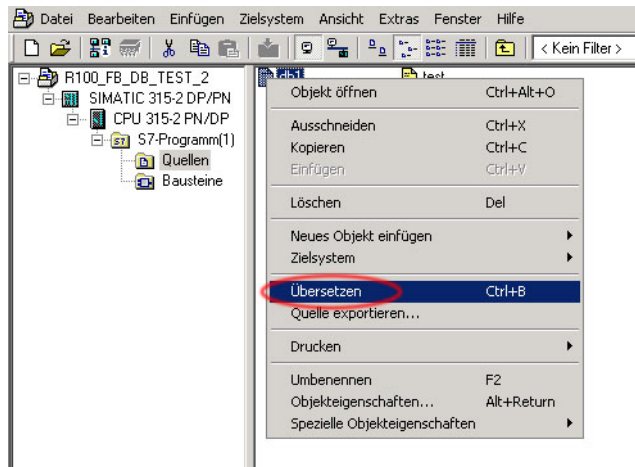


Im darauffolgenden Fenster kann nun die gewünschte AWL-Datei ausgewählt werden.

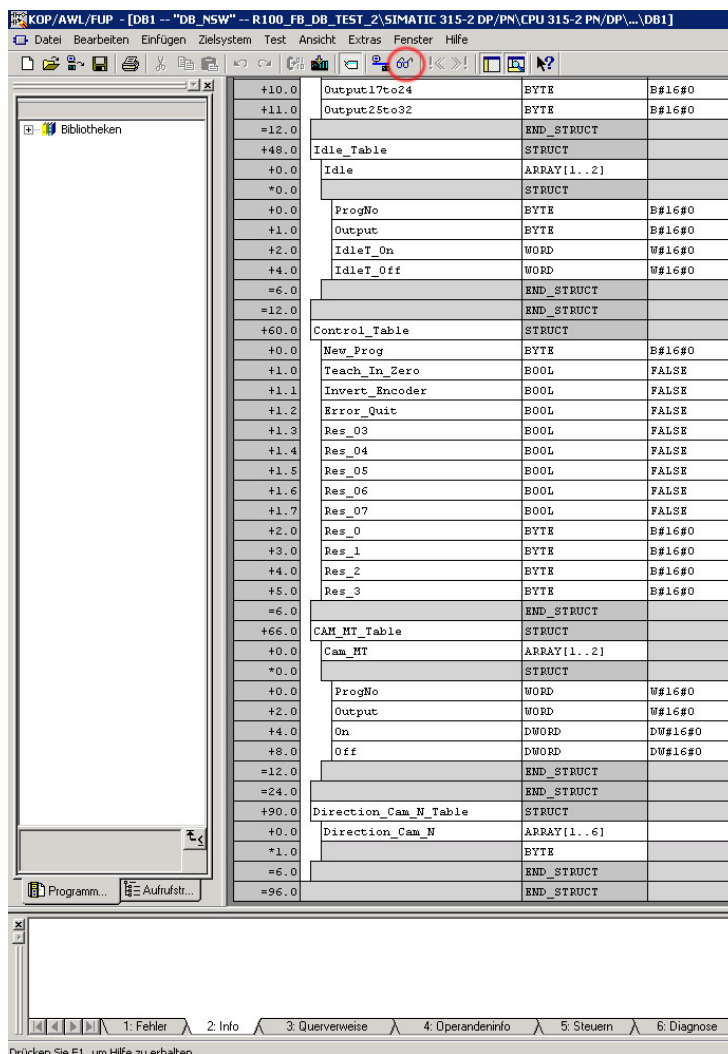


Diese AWL-Datei muss nun markiert werden. Mittels rechter Maustaste kann nun die AWL-Datei in den Datenbaustein übersetzt werden.





In der Online-Ansicht sind alle Parameter zu sehen, die in der Parameter der AWL-Datei hinterlegt wurden. Dazu muss im DB1 die „Brille“ ausgewählt werden.



Online-Ansicht des DB1.

42.0	Processdata_Long_Table.Act_Prog	BYTE	B#16#0	B#16#00
43.0	Processdata_Long_Table.ErrorNo	BYTE	B#16#0	B#16#00
44.0	Processdata_Long_Table.Output1to8	BYTE	B#16#0	B#16#00
45.0	Processdata_Long_Table.Output9to16	BYTE	B#16#0	B#16#00
46.0	Processdata_Long_Table.Output17to24	BYTE	B#16#0	B#16#00
47.0	Processdata_Long_Table.Output25to32	BYTE	B#16#0	B#16#00
48.0	Idle_Table.Idle[1].ProgNo	BYTE	B#16#0	B#16#00
49.0	Idle_Table.Idle[1].Output	BYTE	B#16#0	B#16#01
50.0	Idle_Table.Idle[1].IdleT_On	WORD	W#16#0	W#16#01F4
52.0	Idle_Table.Idle[1].IdleT_Off	WORD	W#16#0	W#16#01F4
54.0	Idle_Table.Idle[2].ProgNo	BYTE	B#16#0	B#16#00
55.0	Idle_Table.Idle[2].Output	BYTE	B#16#0	B#16#02
56.0	Idle_Table.Idle[2].IdleT_On	WORD	W#16#0	W#16#01F4
58.0	Idle_Table.Idle[2].IdleT_Off	WORD	W#16#0	W#16#01F4
60.0	Control_Table.New_Prog	BYTE	B#16#0	B#16#00
61.0	Control_Table.Teach_In_Zero	BOOL	FALSE	FALSE
61.1	Control_Table.Invert_Encoder	BOOL	FALSE	FALSE
61.2	Control_Table.Error_Quit	BOOL	FALSE	FALSE
61.3	Control_Table.Res_03	BOOL	FALSE	FALSE
61.4	Control_Table.Res_04	BOOL	FALSE	FALSE
61.5	Control_Table.Res_05	BOOL	FALSE	FALSE
61.6	Control_Table.Res_06	BOOL	FALSE	FALSE
61.7	Control_Table.Res_07	BOOL	FALSE	FALSE
62.0	Control_Table.Res_0	BYTE	B#16#0	B#16#00
63.0	Control_Table.Res_1	BYTE	B#16#0	B#16#00
64.0	Control_Table.Res_2	BYTE	B#16#0	B#16#00
65.0	Control_Table.Res_3	BYTE	B#16#0	B#16#00
66.0	CAM_MT_Table.Cam_MT[1].ProgNo	WORD	W#16#0	W#16#0000
68.0	CAM_MT_Table.Cam_MT[1].Output	WORD	W#16#0	W#16#0001
70.0	CAM_MT_Table.Cam_MT[1].On	DWORD	DW#16#0	DW#16#00000000
74.0	CAM_MT_Table.Cam_MT[1].Off	DWORD	DW#16#0	DW#16#000000064
78.0	CAM_MT_Table.Cam_MT[2].ProgNo	WORD	W#16#0	W#16#0000
80.0	CAM_MT_Table.Cam_MT[2].Output	WORD	W#16#0	W#16#0002
82.0	CAM_MT_Table.Cam_MT[2].On	DWORD	DW#16#0	DW#16#00000000
86.0	CAM_MT_Table.Cam_MT[2].Off	DWORD	DW#16#0	DW#16#000000064
90.0	Direction_Cam_N_Table.Direction_Cam_N[1]	BYTE	B#16#0	B#16#01
91.0	Direction_Cam_N_Table.Direction_Cam_N[2]	BYTE	B#16#0	B#16#02
92.0	Direction_Cam_N_Table.Direction_Cam_N[3]	BYTE	B#16#0	B#16#00
93.0	Direction_Cam_N_Table.Direction_Cam_N[4]	BYTE	B#16#0	B#16#00
94.0	Direction_Cam_N_Table.Direction_Cam_N[5]	BYTE	B#16#0	B#16#00
95.0	Direction_Cam_N_Table.Direction_Cam_N[6]	BYTE	B#16#0	B#16#00

## 14 Reaktionszeit der NS bei Hardware-Programmumschaltung

Durchgeführt wurde die Messung der Reaktionszeit beispielhaft an den Geräten LOCON 24 MPI Version 4.03 und am Gerät ROTARNOCK 2 Profibus Version 5.3.

Verwendet wurde der Handlingsbaustein für ROTARNOCK-DP PLC-SPS, der auf unserer Internetseite <http://www.deutschmann.de/download> zur Verfügung steht.

Für LOCON 24 mit MPI wurde ein Ref DB und ein Datenbaustein benutzt.

Es ist zu beachten, daß die Reaktionszeit von der Datenbausteingröße abhängig ist.

Folgendes Testprogramm wurde zusätzlich in den OB 1 eingefügt:

L EB 0	- Lade Hardwaretaste für P1
T DB1.DBB234	- Schreibe Wert in New-ProgNo.

### 14.1 Reaktionszeit LOCON 24 MPI

Programmiert wurde eine Vollnocke in Programm 1. Die MPI-Datenrate ist fix 187,5 Kbit/s.

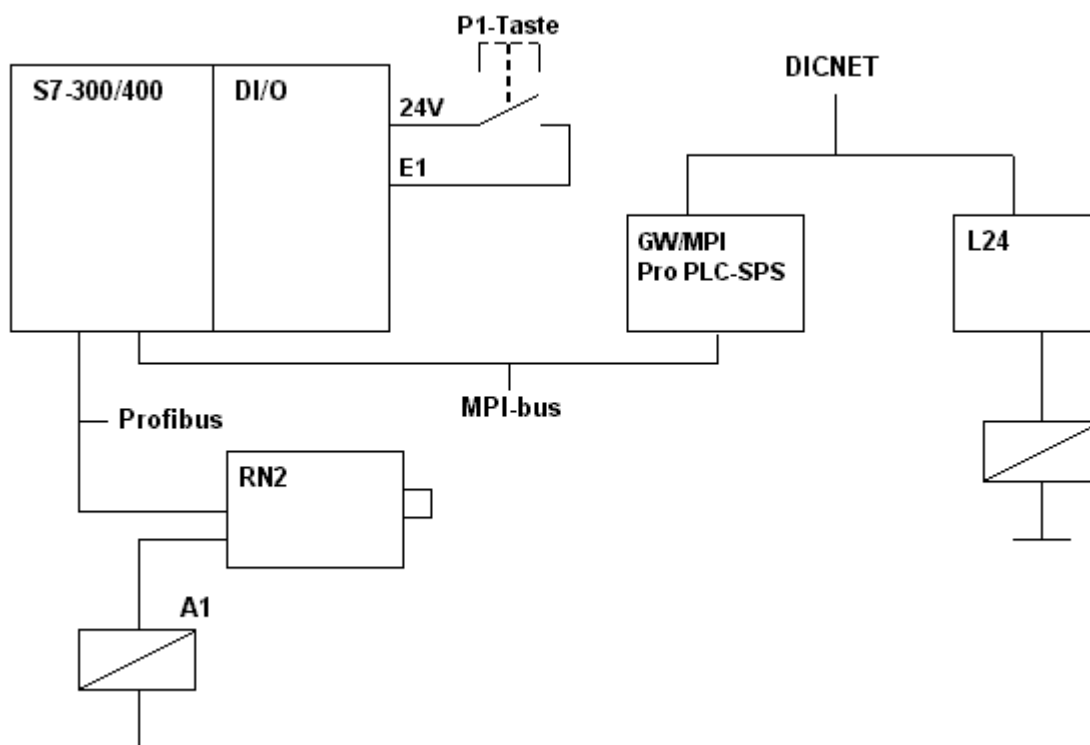
Schalter	Ausgang	Vollnocke	Reaktionszeit
P1	A1	0-360	< 1 Sekunde

### 14.2 Reaktionszeit ROTARNOCK 2 - Profibus

Programmiert wurde eine Vollnocke in Programm 1

Schalter	Ausgang	Vollnocke	Reaktionszeit	ProfibusDP Datenrate
P1	A1	0-360	< 500 ms	12 Mbit/s
P1	A1	0-360	< 500 ms	500 Kbit/s

### 14.3 Schematische Darstellung des Testaufbaus



DI/O =	Digital In/Out (Siemens Anschaltbaugruppe)
A1 =	Ausgang 1
E1 =	Eingang 1
S7- 300/400 =	Simatic S7 mit PROFIBUS DP
RN2 =	ROTARNOCK 2 -PROFIBUS DP
L 24 =	LOCON 24
GW/MPI Pro PLC-SPS =	Gateway MPI Protokoll PLC-SPS

## 15 Fehlermeldungen

Ergänzend zu den Fehlermeldungen (vgl. Kap. „Fehlermeldungen“ im Handbuch zum ROTARNOCK bzw. LOCON 24, 48, 64) sind bei Nockensteuerungen mit Feldbusanbindung folgende Fehlermeldungen möglich:

Fehler-nummer	Bedeutung	Anmerkung
6	Fehler in der MPI-Konfiguration	MPI-ID > 32
9	Fehler in der internen Kommunikation zwischen Prozessor und Feldbus-Chip	Gerät neustarten oder einschicken
36	Angesprochener Datenbaustein in der S7 nicht vorhanden	Z. B. DB1 nicht vorhanden bei PROFIBUS
43	Keine Verbindung zwischen ROTARNOCK und S7	Z. B. ID falsch eingestellt Verbindungskabel defekt
80	Fehler im Aufbau des S7-Datenbausteins	Eventuell falsche DB-Nummer eingestellt Datenbaustein mit Generator neu erzeugen.
22	Fehler beim Speichern eines Nockenwertes	Falscher Wert z. B. zu groß
82	Logik Konfig Fehler	Logik nicht konfiguriert

### 15.1 Status LED am ROTARNOCK

Im Betriebszustand leuchtet die LED rot auf den „Nullpunkt“. Blinkt die LED rot (4 x schneller als bei einem „normalen“ Fehler), liegt ein Fehler vor, der mit oben angeführter Tabelle analysiert werden kann. Die Nummer kann entweder über die Diagnosedaten aus dem Profibus gelesen werden oder über die RS232-Schnittstelle des Onlinefensters von WINLOC.

## 16 Service

Sollten Fragen auftreten, die in diesem Handbuch nicht beantwortet werden, sollte zunächst im

- FAQ-Bereich unserer Homepage [www.deutschmann.de](http://www.deutschmann.de)

und dem

- Deutschmann-WiKi [www.wiki.deutschmann.de](http://www.wiki.deutschmann.de)

sowie dem

- •jeweiligen Handbuch der verwendeten Nockensteuerung nachgesehen werden.

Falls dennoch Fragen unbeantwortet bleiben sollten, wenden Sie sich an den für Sie zuständigen Vertriebspartner (s. im Internet: [www.deutschmann.de](http://www.deutschmann.de)) oder direkt an uns.

Bitte halten Sie für Ihren Anruf folgende Angaben bereit:

- Gerätebezeichnung
- Seriennummer (S/N)
- Art.-Nr.
- Fehlernummer und Fehlerbeschreibung

Sie erreichen uns während der Hotlinezeiten von

Montag bis Donnerstag von 8.00 bis 12.00 und von 13.00 bis 16.00, Freitag von 8.00 bis 12.00.

Zentrale und Verkauf 06434-9433-0

Technische Hotline 06434-9433-33

Fax Verkauf 06434-9433-40

Fax Technische Hotline 06434-9433-49

E-mail Technische Hotline: [hotline@deutschmann.de](mailto:hotline@deutschmann.de)

### 16.1 Einsendung eines Gerätes

Bei der Einsendung eines Gerätes an uns, benötigen wir eine möglichst umfassende Fehlerbeschreibung. Insbesondere benötigen wir die nachfolgenden Angaben:

- Welche Fehlernummer wurde angezeigt
- Wie ist das Gerät extern beschaltet (Geber, Ausgänge, ...), wobei **sämtliche** Anschlüsse des Gerätes aufgeführt sein müssen
- Wie groß ist die 24V-Versorgungsspannung ( $\pm 0,5V$ ) mit angeschlossenem LOCON
- Was waren die letzten Aktivitäten am Gerät (Programmierung, Fehler beim Einschalten, ...)

Je genauer Ihre Angaben und Fehlerbeschreibung, je exakter können wir die möglichen Ursachen prüfen.

Geräte, die ohne Fehlerbeschreibung eingeschickt werden, durchlaufen einen Standardtest, der auch im Fall, dass kein Fehler festgestellt wird, berechnet wird.

## 16.2 Internet

Über unsere Internet-Homepage [www.deutschmann.de](http://www.deutschmann.de) können Sie die Software WINLOC32 laden. Dort erhalten Sie auch aktuelle Produktinformationen, Handbücher und einen Händlernachweis. Internet

Über unsere Internet-Homepage (URL) können Sie diverse Software laden. Dort erhalten Sie auch aktuelle Produktinformationen, Handbücher und einen Händlernachweis.

**URL:** [www.deutschmann.de](http://www.deutschmann.de).

S7 Beispiel-Projekte für:

		PROFIBUS ID
ROTARNOCK 4:	ID3231V1.zip	8
ROTARNOCK 4 Logik:	ID3231V2.zip	8
ROTARNOCK 2:	ID2935V1.zip	8
ROTARNOCK 2 MT:	ID2935V2.zip	8
LOCON 24:	ID2079V1.zip	126
LOCON 24 MT:	ID2079V2.zip	126
R100 / L 100:	R100V1Pa.zip	8

