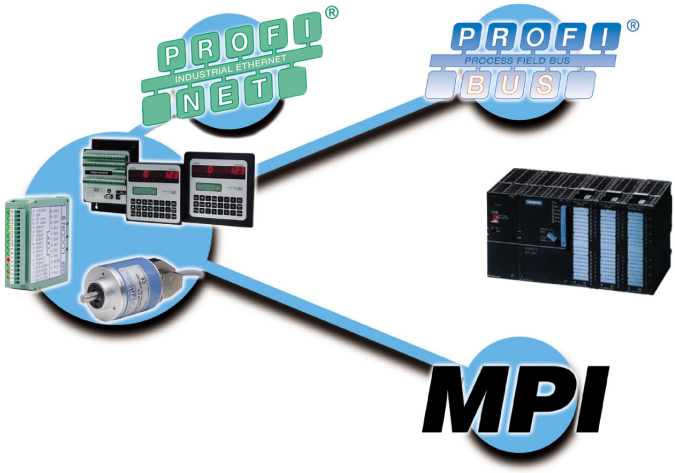
Deutschmann

*your ticket to all buses*

# Instruction Manual
# Cam Controls
# with Fieldbus connection




**Deutschmann Automation GmbH & Co. KG**
**www.deutschmann.com | wiki.deutschmann.de**

Manual Art.-No. V3058E

# 1 Introduction

## 1.1 On this manual

This manual documents installation, functions and operation of the Deutschmann unit specified on the cover sheet and in the header.

### 1.1.1 Symbols

Particularly **important text sections** can be seen from the adjacent pictogram.

You should **always** follow this information since, otherwise, this could result in malfunctions or operating errors.

### 1.1.2 Concepts

The expressions ‘LOCON’ and ’TERM’ are frequently used throughout this manual with no further model specifications. In such cases, the information applies to the entire model series.

### 1.1.3 Suggestions

We are always pleased to receive suggestions and wishes etc. and endeavour to allow for these. It is also helpful if you bring our attention to any errors.

## 1.2 Product program of Deutschmann Automation

A detailed and topical outline of our product range can be found on our homepage at http://www.deutschmann.de

# 2 Introduction

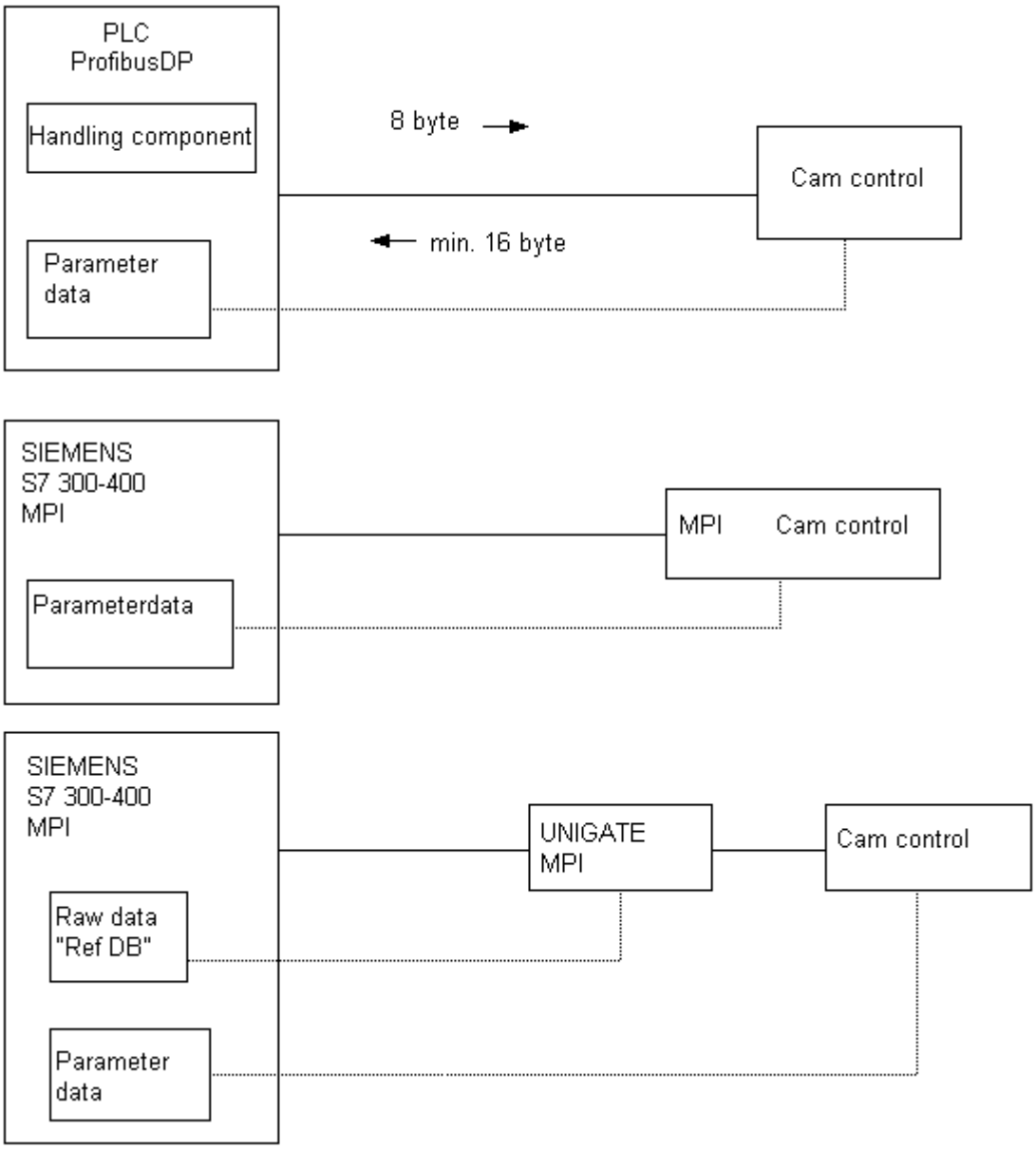
DEUTSCHMANN cam controls with PROFIBUS- or MPI-interface[1] can easily be operated at a PLC. For this it is necessary that either the protocol DICNET or the protocol PLC-SPS is set at the cam control. All parameters from a cam control can be processed by the DICNET or Deutschmann protocol.
The PLC-SPS protocol allows the processing of the most important parameters in the cam control (see chapter 5, "Table-types of the parameter-data-table" on page 11).
This table form makes a parameterization easier.

This instruction manual describes the PLC-SPS protocol to the DICNET protocol, see comment in chapter 9.1, "Protocol: DICNET" on page 25.

*1. MPI-interface: for a Siemens PLC with MPI-interface*

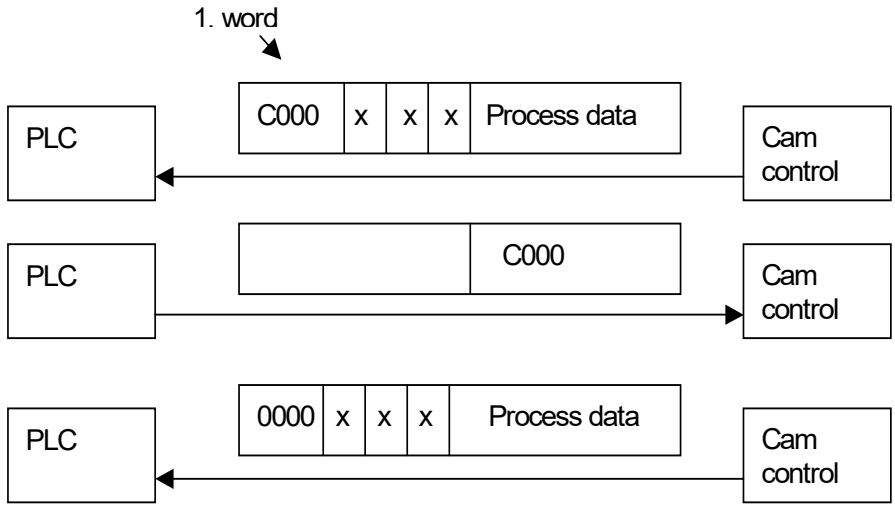
# 3 Program

At a PLC with Profibus the transfer of the parameter data to the cam control is taken over by a PLC program (handling component).
The cam control transfers the process data back in every PROFIBUS cycle. The length of the process data is dependent on the cam control and the selected module of the GSD-file (see chapter 6.18, "GSD-module for Profibus cam control" on page 16).



# 4 Synchronization (starting phase) [2]

After switching on the cam control, the PLC and the cam control synchronize themselves. Afterwards the parameter data are copied to the cam control cyclically.
After switching on, the cam control sends a 0xC000 (bit 15 and 14 set in the 1. word) until it gets this word back from the PLC as an echo. After that the cam control sends a 0x0000 (bit 15 and 14 deleted in the 1. word) and ends the synchronization with it.
From that point on the handling component copies always 3 subsequent words from the parameter data table, from the address word via the bus to the cam control (see chapter 4.2, "Data structure of a request (from PLC to cam control)" on page 9).

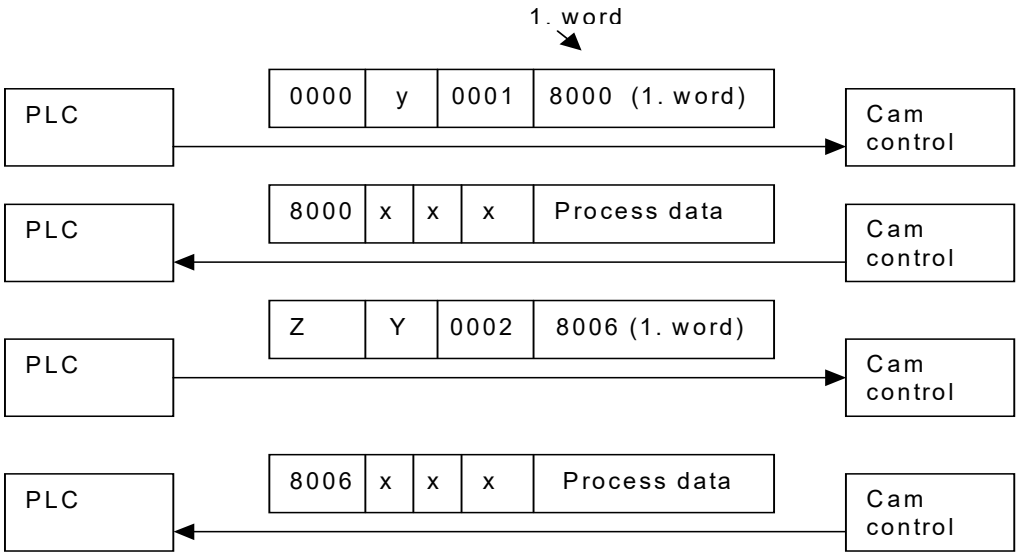



Values in hex
x = not used

## 4.1 Data exchange

Now the PLC sends 0x8000 (bit 15 write-req set) with start address of the first word from the table and the first three words from the parameter table.
A parameter table always starts with the address 0. All data have always to be copied in intervals of 6 byte to the cam control. As a confirmation the PLC waits for the first word of the request (0x8000 hex).
The second data record, that is sent from the PLC to the cam control is 0x8006 hex in the first word and the next three words from the parameter table.
Consequently the start address always has to be a multiple of 6. Reaching the end of the table the copy process starts again from the address 0.
As a confirmation the PLC waits for the first word of the request (0x8006 hex).
Besides the PLC always gets the process data back (see chapter 4.3, "Process data in singleturn-format" on page 10).

*2. not at MPI*




x = Not used
Y = Number of bytes of the offset table
Z = Start address of the parameter data, e. g. from type 2

## 4.2 Data structure of a request (from PLC to cam control)

| 1. word | | | 2. word | 3. word | 4. word | 5. word |
|---|---|---|---|---|---|---|
| Bit 15<br>Command | Bit 14<br>Command | 13 ..... 0<br>Address pointer | 15 ... 0<br>Data | 15 ... 0<br>Data | 15 ... 0<br>Data | only for R4 with logic function |
| 0<br>0<br>1<br>1 | 0<br>1<br>0<br>1 | not used<br>not used<br>write-request<br>sync (starting phase) | 6 byte data from address pointer | | | 16 bit logoc for ROTARNOCK 4 |

**Note:** With regard to the logic function, the following must be observed when using the 16 inputs:

* ROTARNOCK 4-PROFIBUS: The 1st input is in the 10 byte in the 5th word in the PROFIBUS.

* ROTARNOCK 100-PROFIBUS: The 1st input is in the 9 byte in the 5th word in the PROFIBUS.

## 4.3 Process data in singleturn-format

From the 5. word on the process data are also provided in every PROFIBUS-cycle.

**Data structure of a response from the cam control to the PLC:**

| Word | 1. word | 2. word | 3. word | 4. word | 5. word | 6. word | 7. word | 8. word | |
|---|---|---|---|---|---|---|---|---|---|
| Bit | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 8 | 7 .. 0 |
| | Copy of the 1. word from the PLC as confirmation of receipt | — | — | — | Process data | | | | |
| | | | | | Position | Speed | Output | Act-Progr | Error No |
| | | | | | H L | H L | | | |

## 4.4 Process data in Multiturn-format

From the 5. word on the process data are permanently provided.

**Data structure of a response from the cam control to the PLC:**

| Word<br>Byte No. | 1. word<br>Byte 0, 1 | 2. word<br>Byte 2, 3 | 3. word<br>Byte 4, 5 | 4. word<br>Byte 6, 7 | 5. word<br>Byte 8, 9 | 6. word<br>Byte 10, 11 | 7. word<br>Byte 12 , 13 | ..... |
|---|---|---|---|---|---|---|---|---|
| Bit | 15 .. 0 | 15 .. 0 | 15 .. 0 | 15 .. 0 | 31 .. 16 | 15 .. 0 | 15 .. 0 | ..... |
| | Copy of the 1. word from the PLC as confirma-tion of receipt | — | — | — | Position | | Speed | ..... |

| ..... | Byte 14 | Byte 15 | Byte 16 | Byte 17 | Byte 18 | Byte 19 |
|---|---|---|---|---|---|---|
| ..... | 7 .. 0 | 7 .. 0 | 7 .. 0 | 15 .. 8 | 23 .. 16 | 31 .. 24 |
| ..... | ActProgr | Error No. | Output<br>1 to 7 | Output<br>9 to 16 | Output<br>17 to 24 | Output<br>25 to 32 |

| ..... | Byte 20 | Byte 21 | Byte 22 | Byte 23 | Byte 24 | Byte 25 |
|---|---|---|---|---|---|---|
| ..... | 7 .. 0 | 7 .. 0 | 7 .. 0 | 15 .. 8 | 23 .. 16 | 31 .. 24 |
| ..... | Output<br>33 to 40 | Output<br>41 to 48 | Output<br>49 to 56 | Output<br>57 to 64 | (Reserve) | Reserve |

**Assignment of the outputs to bits**

**Example:**

| MSB<br>Bit 7 | ... | LSB<br>Bit 0 |
|---|---|---|
| Output 8 | ... | Output 1 |

# 5 Table-types of the parameter-data-table

| Description | Fixed value | Table-type | Length |
|---|---|---|---|
| OFFSET_TYPE | 0x0001 | Offset always starts at table-address 0 | WORD |
| | | | |
| CAM_ST_TYPE | 0x0003 | Cam table | WORD |
| IDLE_TYPE | 0x0004 | Idle time table | WORD |
| CONTROL_TYPE | 0x0005 | Config-parameter table | WORD |
| | | | |
| CAM_MT_TYPE | 0x0007 | Cams for Multiturn devices | WORD |
| DIRECTIONCAM_TYPE | 0x0008 | Direction cams | WORD |
| REF_TYPE | 0x000A | Only for MPI-Gateway with protocol PLC-SPS | WORD |
| AT_CAM_ST_TYPE | 0x000B | Angle-time cams | WORD |
| LOGIC_TYPE | 0x000C | Logic function | WORD |
| | | | |
| | | | |

# 6 Offset-table

In the offset-table three words are reserved in each case for the description of a parameter table. The first three words urgently have to be located at the beginning of this parameter table, they, themselves declare the offset-table.
The first word contains the key sign for this offset-table (0x0001), the second word contains the number of required bytes of the table and the third word contains the start address of the table.
The start address of the offset-table always equals zero.
The entries in the offset-table enable already existing tables to be extended or new ones to be added. The parameter table always starts with the offset declaration (example).

| Description | Value in the PLC | Function | Length |
|---|---|---|---|
| Offset_Type | 1 | | WORD |
| Offset_Length | To be calculated | | WORD |
| Offset_ Address | 0 | | WORD |

Thereupon the declaration of each table follows by means of the offset-table.
As an example only the offset-tables for cams and idle times are listed here.

## 6.1 Writing cams in the cam control

**Offset-table for cams**

| Dscription | Value in the PLC | Function | Length |
|---|---|---|---|
| Cam_ST_Type | 3 | | WORD |
| Cam_ST_Length | Number of required cams by 6 | | WORD |
| Cam_ST_Address | Address of the first cam entry | | WORD |

## 6.2 CAM_ST Cams

| Structure | Value | Function | Length |
|---|---|---|---|
| ProgNo | 0-15 | | BYTE |
| Output | | 0 = cam will be deleted | BYTE |
| On | | | WORD |
| Off | | | WORD |

If both switching points ON and OFF as well equal 0, then the cam will also be deleted.

## 6.3 Offset-table for idle time

| Description | Value in the PLC | | Length |
|---|---|---|---|
| Idle_Type | 4 | | WORD |
| Idle_Length | Number of required idle times by 6 | | WORD |
| Idle_Address | Address of the first idle time entry | | WORD |

## 6.4 Idle time: IDLE

| Structure | Value | Function | Length |
|---|---|---|---|
| ProgNo | | | BYTE |
| Output | | $0_n$ und OFF = 0 => this idle time will be deleted | BYTE |
| IdleT_On | | | WORD |
| IdleT_Off | | | WORD |

Any further required idle time will be lined up gaplessly.
In order to automatically program the cam control, the programmer of the PLC only has to change the corresponding values in the structure-table.

## 6.5 Offset table for control table

| Description | Value in the PLC | Length |
|---|---|---|
| Control_Type | 5 | WORD |
| Control_Length | here fixed on 6 | WORD |
| Control_Address | Address of the first control entry (New_Prog) | WORD |

## 6.6 Devices control-table: CONTROL_TYPE

In this table those bytes and flags are set by the PLC programmer, that carry out a specific configuration in the cam control. The table consist of six bytes:

| Description | Value | Function | Length |
|---|---|---|---|
| New_Prog | 0..15 | Select new program | BYTE |
| ConfigFlags: Teach _In_Zero | Bit 0 | Teach-In Zero Point (high active) | BOOL |
| Invert_Encoder | Bit 1 | Invert-Encoder-Countdir (0=not inverted, 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 table for Multiturn cams

| Description | Value in the PLC | Length |
|---|---|---|
| Cam_MT_Type | 7 | WORD |
| Cam_MT_Length | Amount of required cams by 12 | WORD |
| Cam_MT_Address | Address of the first MT-entry | WORD |

## 6.8 Cams for Multiturn: CAM_MT

| Structure | Value | Function | Length |
|---|---|---|---|
| ProgNo | | | WORD |
| Output | | 0 = deletes the cams in the device | WORD |
| On | | | DWORD |
| Off | | | DWORD |

## 6.9 Offset table for direction cams

If the number of bits, which are corresponding to an output is 0, then this output is not active.
If the bit corresponding to an output is „pos“ = 1 and bit „neg“ = 0, then the cam turns in positive rotational direction

| Description | Value in the PLC | Length |
|---|---|---|
| Direction_Cam_Type | 8 | WORD |
| Direction_Cam_Length | 6 byte | WORD |
| Direction_Cam_Address | Address of the first direction-cam-entry | WORD |

## 6.10 Direction cams: DIRECTION_CAM

| Structure | Value | Function | Length |
|---|---|---|---|
| Output16pos | Bit 7 | At positive rotational direction switching output | BOOL |
| Output15pos | Bit 6 | At positive rotational direction switching output | BOOL |
| Output14 pos | Bit 5 | At positive rotational direction switching output | BOOL |
| Output13pos | Bit 4 | At positive rotational direction switching output | BOOL |
| Output12pos | Bit 3 | At positive rotational direction switching output | BOOL |
| Output11pos | Bit 2 | At positive rotational direction switching output | BOOL |
| Output10pos | Bit 1 | At positive rotational direction switching output | BOOL |
| Output9pos | Bit 0 | At positive rotational direction switching output | BOOL |
| Output8pos | Bit 7 | At positive rotational direction switching output | BOOL |
| Output7pos | Bit 6 | At positive rotational direction switching output | BOOL |
| Output6pos | Bit 5 | At positive rotational direction switching output | BOOL |
| Output5pos | Bit 4 | At positive rotational direction switching output | BOOL |
| Output4pos | Bit 3 | At positive rotational direction switching output | BOOL |
| Output3pos | Bit 2 | At positive rotational direction switching output | BOOL |
| Output2pos | Bit 1 | At positive rotational direction switching output | BOOL |
| Output1pos | Bit 0 | At positive rotational direction switching output | BOOL |
| Output16neg | Bit 7 | At negative rotational direction switching output | BOOL |
| Output15neg | Bit 6 | At negative rotational direction switching output | BOOL |
| Output14neg | Bit 5 | At negative rotational direction switching output | BOOL |
| Output13neg | Bit 4 | At negative rotational direction switching output | BOOL |
| Output12neg | Bit 3 | At negative rotational direction switching output | BOOL |
| Output11neg | Bit 2 | At negative rotational direction switching output | BOOL |
| Output10neg | Bit 1 | At negative rotational direction switching output | BOOL |
| Output9neg | Bit 0 | At negative rotational direction switching output | BOOL |
| Output8neg | Bit 7 | At negative rotational direction switching output | BOOL |
| Output7neg | Bit 6 | At negative rotational direction switching output | BOOL |
| Output6neg | Bit 5 | At negative rotational direction switching output | BOOL |
| Output5neg | Bit 4 | At negative rotational direction switching output | BOOL |
| Output4neg | Bit 3 | At negative rotational direction switching output | BOOL |
| Output3neg | Bit 2 | At negative rotational direction switching output | BOOL |
| Output2neg | Bit 1 | At negative rotational direction switching output | BOOL |
| Output1neg | Bit 0 | At negative rotational direction switching output | BOOL |
| Reserved | not used | | WORD |

### Direction cams' mode of operation

| State | Output 1 negativ | Output 1 positiv |
|---|---|---|
| No output update | 0 | 0 |
| Only negative direction | 1 | 0 |
| Only positive direction | 0 | 1 |
| Both directions | 1 | 1 |

In the state of default an output update takes place in both directions.

## 6.11 Reference: REFERENCE [3]

| Structure | Value | Function | Length |
|---|---|---|---|
| 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. only for Unigate MPI***

## 6.12 Offset table for angle-time cams

| Description | Value in the PLC | Length |
|---|---|---|
| AT_CAM_ST_Type | 0x0B | WORD |
| AT_CAM_ST_Length | Amount of required cams by 6 | WORD |
| AT_CAM_ST_Address | Address of the first angle-cam-entry | WORD |

## 6.13 Angle-time cams: AT_CAM_ST

| Structure | Value | Function | Length |
|---|---|---|---|
| ProgNo | | | BYTE |
| Output | | | BYTE |
| On | | | WORD |
| Duration | 0x0001 - 0x7EF4 | ms | WORD |

## 6.14 LOGIC-function: LOGIC

| Structure | Value | Function | Length |
|---|---|---|---|
| ProgNo | from 0 to MAX_PROG | | BYTE |
| DestNo | from 1 to 16 | 0 deletes complete logic function | BYTE |
| DestType | 0 = hardware output<br>1 = flag<br>2 = hardware output inverted<br>3 = flag inverted | | Byte |
| OpNo1 | 1 - 32 | | BYTE |
| OpType1 | 0 = internal cam control output<br>1 = Input: hard-/software<br>2 = flag<br>3 = SR (shift register)<br>4 = PB-Input (only LOCON 200) | | BYTE |
| LogicFct1-2 | 0 = none<br>1 = or<br>2 = and<br>3 = or not<br>4 = and not | | BYTE |
| OpNo2 | 1 - 32 | | BYTE |
| OpType2 | see OpType1 | | BYTE |
| LogicFct2-3 | see LogicFct1-2 | | BYTE |
| OpNo3 | 1 - 32 | | BYTE |
| OpType3 | see OpType1 | | BYTE |
| LogicFct3-4 | see LogicFct1-2 | | BYTE |
| OpNo4 | 1 - 32 | | BYTE |
| OpType4 | see OpType1 | | BYTE |
| OutputDelay | ms | at present max. 255 | WORD |
| OutputTrigger | 0 = leading edge<br>1 = trailing edge | | BYTE |
| Module number<br>(only LOCON 200) | 0 basis<br>x I/O-module number | | BYTE |

## 6.15 Process data for Singleturn: Processdata_ST (read only)

| Structure | Value | Length |
|---|---|---|
| Position | 1. byte position H, 2. byte position L | WORD |
| Speed | 3. byte speed H, 4. byte speed L | WORD |
| Output16_1 | 5. byte output (15..8), 6. byte output (7..0) | WORD |
| Act_Prog | 7. byte indicates the current program number | BYTE |
| ErrorNo | 8. byte indicates current error numbers | BYTE |
| ResWord_1 | | WORD |
| ResWord_2 | | WORD |

## 6.16 Process data for Long: PROCESSDATA_LONG (read only)

| Structure | Value | Length |
|---|---|---|
| Position | 1. byte position H, 4. byte position L | DWORD |
| Speed | 5. byte speed H, 6. byte speed L | WORD |
| Act_Prog | 7. byte indicates current program number | BYTE |
| ErrorNo | 8. byte indicates current error number | BYTE |
| Output1to8 | 9. byte output (7..0) | BYTE |
| Output9to16 | 10. byte output (15..8) | BYTE |
| Output17to24 | 11. byte output (23..16) | BYTE |
| Output25to32 | 12. byte output (31..24) | BYTE |

## 6.17 Process data for ROTARNOCK 4: Processdata_80 (read only)

| Structurer | Value | Length |
|---|---|---|
| Position | 1. byte position H, 4. byte position L | DWORD |
| Speed | 5. byte speed H, 6. byte speed L | WORD |
| Act_Prog | 7. byte indicates the current program number | BYTE |
| ErrorNo | 8. byte indicates the current error number | 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 |

The cam control comes up with the process data. Another handling component has to take care of the processing of the process data.

The handling component sends back the values of the process data-table to the cam control, however, it does not have an influence on the cam control. The process data are supplied by the cam control. Another handling component copies the data into this table.

## 6.18 GSD-module for Profibus cam control

| GSD-file | Module | Cam control |
|---|---|---|
| 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 | Multiturrn ROTARNOCK |
| R4pb3231 | S7 DB + Proc.Data | ROTARNOCK 4 |
| R4pb3231 | S7 DB + Proc.DataLogic | ROTARNOCK 4 with logic |
| R100 | S7 DB, Proc.Data, No Logic | ROTARNOCK 100 |
| R100 | S7 DB, Proc.Data, Logic 16 | ROTARNOCK 100 with logic |
| R100 | S7 DB, Proc.Data (CPU318) | ROTARNOCK 100 to S7 318 |
| L100 | S7 DB, Proc.Data, No Logic | LOCON 100 |
| L100 | S7 DB, Proc.Data, Logic 16 | LOCON 100 with 16 logic inputs |
| L100 | S7 DB, Proc.Data, Logic 8 | LOCON 100 with 8 logic inputs |
| L100 | S7 DB, Proc.Data (CPU318) | LOCON 100 to S7 318 |

**Explanation:**

| Type | Length |
|---|---|
| BOOL | 1 bit |
| BYTE | 8 bit |
| WORD | 2 byte |
| DWORD | 4 byte |

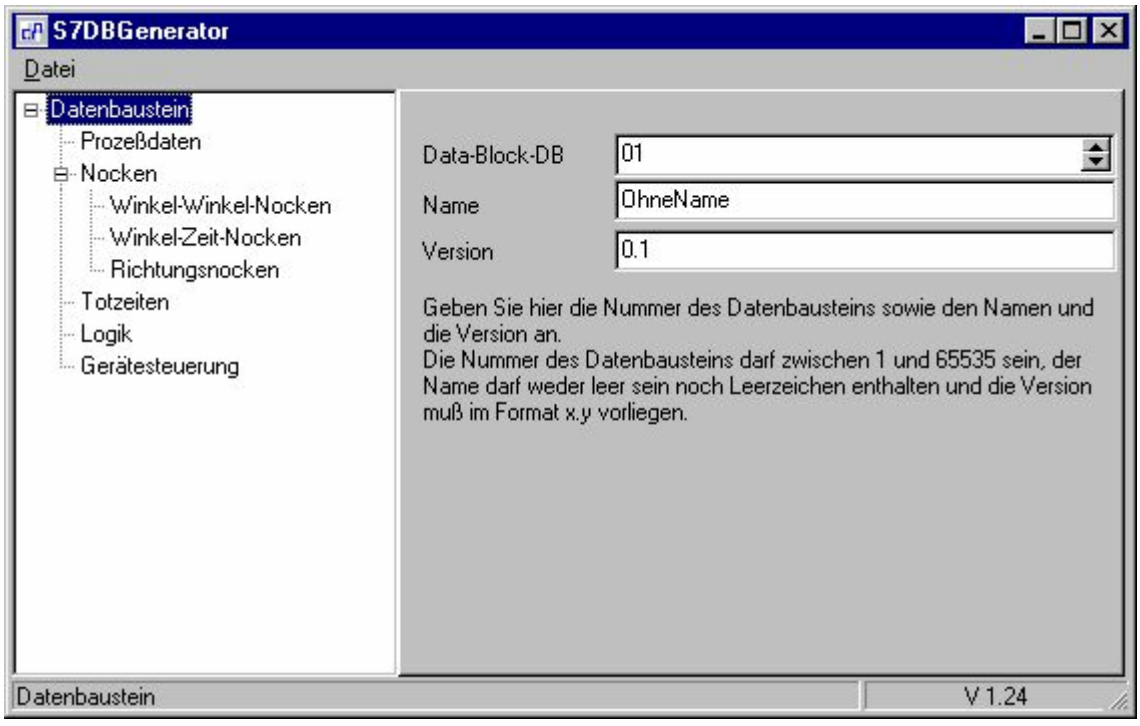
## 6.19 Example: Parameter-table

| Address in dez | Description | Length | Value 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 this example the output 4, Cam_ST_Cam(1).Output is set in the program 0, Cam_ST_-Cam(1).ProgNr., between position 20, Cam_ST_Cam(1).On and 40, Cam_ST_Cam(1).Off. The value Cam_ST_Length is the amount of bytes and is calculated as follows: Amount of cams multiplied by six.
Offset_Length is the amount of bytes of the offset parameters, always starting at zero and ending at the address 10 in this example.

## 6.20 Data component-generator

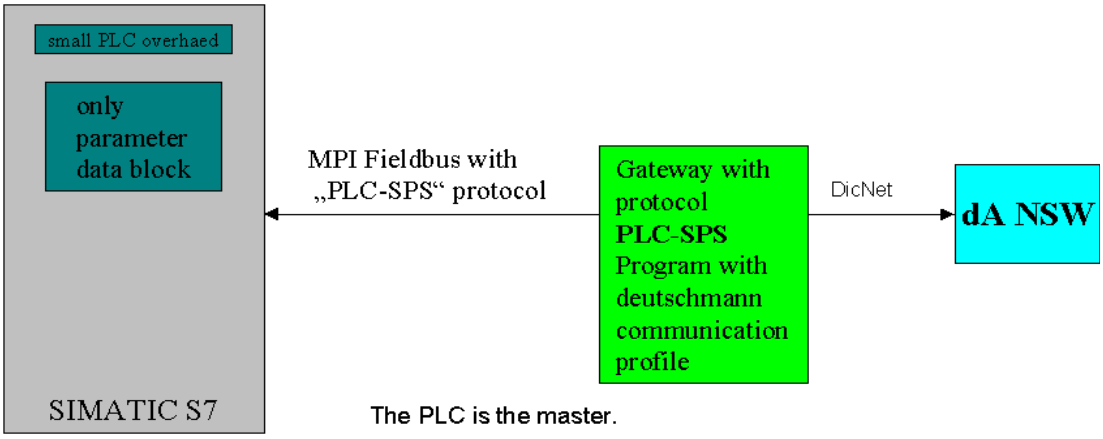
The generator generates a desired parameter-table (empty) automatically in the AWL-Format. With it the user is relieved of the calculation of the length-values and start addresses. The program can be loaded from our homepage at http://www.deutschmann.de.



# 7 Connection cam controls via MPI UNIGATE to S7 300-400

## 7.1 Accessories

- S7 300 - 400
- MPI Gateway (from V1.3 on)
- 1 MPI-bus cable (RS485) with plug
- current WINGATE software (for configuring the Gateway)
- current WINLOC software (generates data component) or data component generator
- connection cable between PC and Gateway (RS232 D-SUB to 3pol. Phoenix)
- S7 project or reference-data component and cam control data component as file




Connection diagram

## 7.2 Initiation

### 7.2.1 Configuring the Gateway

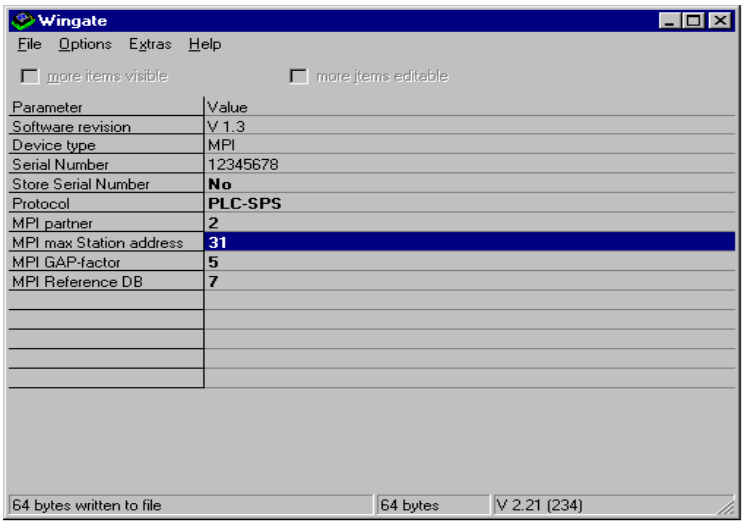
- Connect the Gateway with the PC via Com Port: Set the interface of the Gateway to RS232 - (pin assignment for this cable, see instruction manual).

#### 7.2.1.1 Configmode

- Set the rotary switches on the RS side of the Gateway to “F“ for the configuration.
- Reset the Gateway by means of voltage OFF/ON.
- Now the "State" LED flashes red, "Power" LED flashes green.
- Start the WINGATE software on the PC. When the Gateway is connected to the PC and voltage is applied, an automatic is carried out by starting WINGATE. That means, the current configuration of the Gateway is loaded - normally the Gateway is set to the protocol transparent.

If problems occur with the automatic upload, it can also be carried out manually (see description of the Gateway or of the WINGATE software).

- Change the transparent protocol to PLC - SPS in the WINGATE software.
- Enter the MPI address at the MPI partner (PLC). Normally the MPI address of the PLC is set to >> 2 <<.

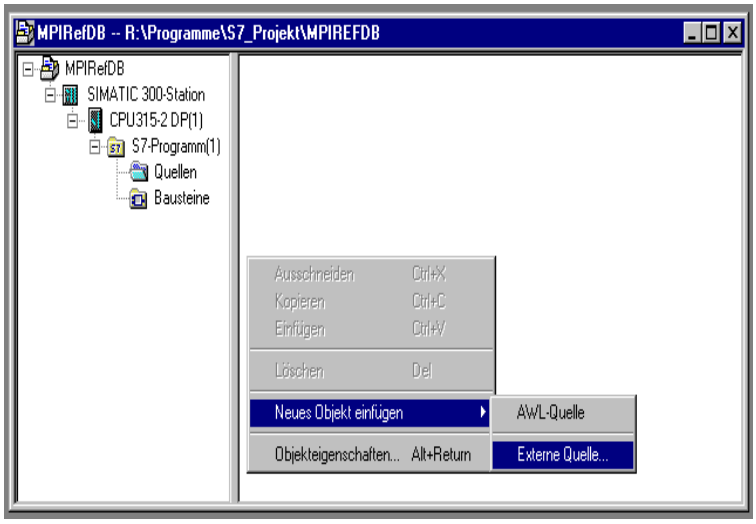



### 7.2.2 Example for the settings

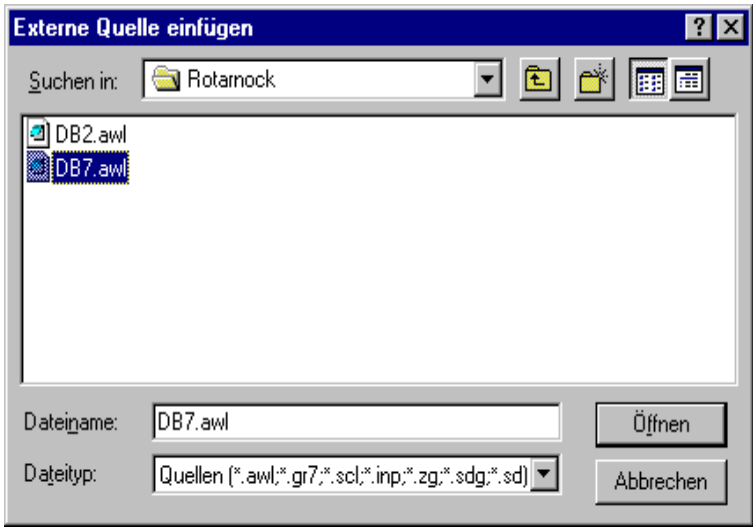
- The maximum station address is supposed to be set to >> 31 << by default. Please keep it.
- The GAP factor is supposed to be set to 5 by default.
- Please set the MPI reference DB (DB = data component) (in the example >> DB7 <<).
- Attention: RefDB has to correspond to the S7 project.
- Save configuration. File -> Save as, not necessarily
- Download the configuration to the Gateway. File -> Download
- Close WINGATE.
- Set the Gateway's rotary switches S4 and S5 on the RS-side to >> 00 <<.
- Set the MPI address at the Gateway on the side of the MPI-interface by means of the rotary switch, e. g. address >> 3 << rotary switch low to >> 3 << and rotary switch high to >> 0 << Maximum allowed address is >> 31 <<.
- RS485 or RS232, dependent on the interface of the cam control.
- Load the S7 project in Step7 PC-software. Two components are included - NSWDB.awl and RefDB.awl).
- Connect the Gateway with the PLC S7 via the MPI-interface by means of the MPI cable.
- Reset the Gateway by means of voltage OFF/ON.
- MPI State LED shines green. Communication with S7
- MPI Power shines green.
- MPI Control shines green.

or

- Add the components through the STEP7 function external source.
- Add and convert RefDB.

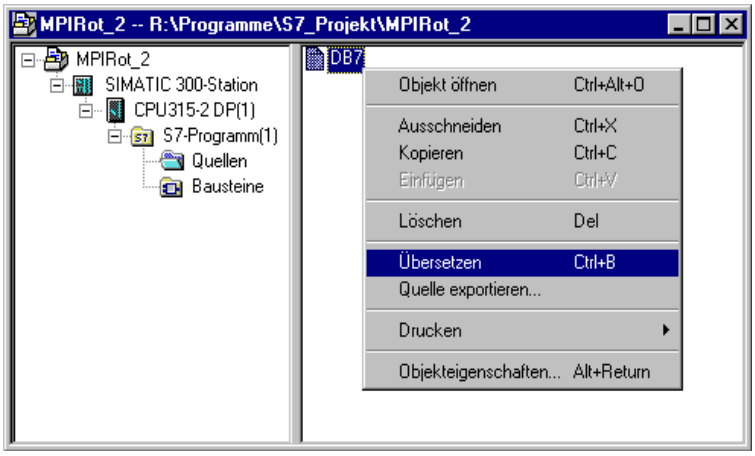



Import AWL-file in STEP7 (File "Source", right mouse button into right window)

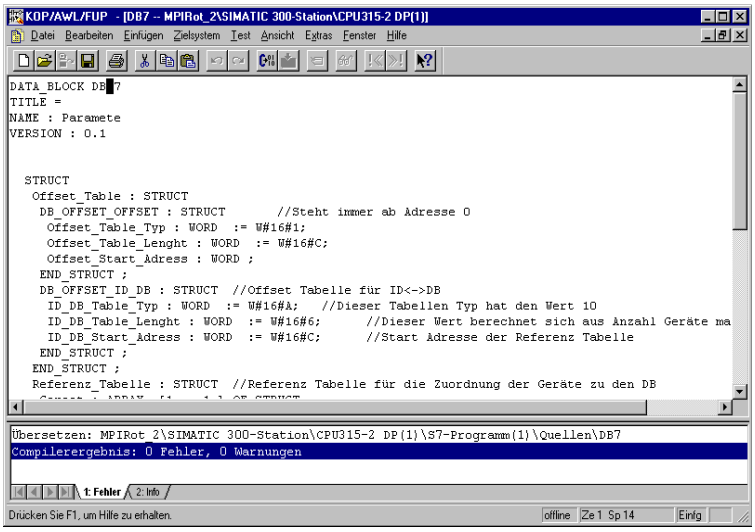



Select generated source from file

Imported source

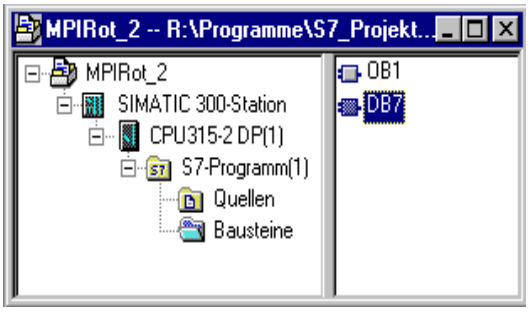



Convert imported source (right mouse button)




Converted data component

After the conversion the DB no. 7 is in the file „data components“




- If necessary the RefDB has to be renamed (in WINGATE >> DB 7 << was entered, if necessary the RefDB.awl has to be renamed in STEP7).

| 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 | | |

Declaration view

- Insert and convert NSWDB.awl (repeat above steps).
- If necessary the DBNo. of the NSWDB.awl has to be renamed.

## 7.3 Setting RefDB

- In the RefDB >> DB 7 << in DB_No now the DB number of the NSWDB has to be entered >> W#16#02<<.
- Depending on how many cam controls are connected, the entry has to be changed in the array. For one cam control the entry has to be changed: for one cam control >> Array [1...1] <<, for two cam controls >> Array [1...2] << and so on...
- In the DB 7 now the ID of the connected cam control has to entered in ID >> W#16#01 <<
- Table Length has to be adjusted. Depending on the number of connected cam controls the amount of devices has to be multiplied by 6 and the entry has to be made correspondingly >> W#16#06 << .
- For two cam controls the entry has to look as follows: >> W#16#12 <<. Now the ID >> 1<< has to be set, as entered in the DB 7 before. This is made through the bridges in the 9pol. D-SUB plug, by means of a rotary switch or with the WINLOC software.
- Flag Register -> LSB set >>W#16#01<<. Then all changes are transferred in the NSWDB.
- Set RefDB number via switch S5.

| Adresse | Name | Typ | Anfangswert | Aktualwer | 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 |

Data view of the RefDB

- In the LOCON 24-series the protocol PLC-SPS is supported from V4.03 on and in the LOCON 1, 2, 16, 17-series from the V5.10 on.

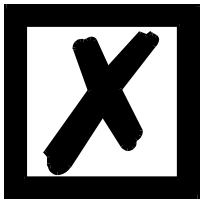
- Now the Gateway cyclically reads (flagregister to 1) the data from the data component No2 (DB_No to 2) and compares them to the content of the cam control with the ID 1 (ID to 1).

- If the data in the DB2 has changed, the Gateway automatically changes the parameter in the cam control.

**At present the following functions supports by the protocol PLC-SPS in the NSWDB:**

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

The NSWDB-file can automatically be generated with the data component generator and it can be imported as described above.



**Attention:**
**It is possible that the device does no longer react after a download into the S7.**
**Then the outputs are not processed and the position is no longer transferred.**
**The device has to be restarted then.**

# 8 Connection LOCON 24 cam controls via MPI to PLC

## 8.1 Protocol PLC

Default: own MPI address: 3
MPI-address PLC: 1
Used data component (RefDB): No. 7
LOCON device ID: 0
Devices DB-no.: 12 (0x0C)

The termination at the LOCON 24, R + with Dic + and R - with Dic - absolutely has to be activated, if no other device is connected at the DICNET.
Connect Prog Enable with + 24 V (see also chapter „Electrical connections LOCON 24, 48, 64“ in the instruction manual for LOCON 24).

# 9 Connection LOCON 24 cam controls via PROFIBUS to PLC

## 9.1 Protocol: DICNET

See instruction manual Communication profile for cam controls of Deutschmann Automation.
- GSD-file: Dagw2079.gsd
- Module: "DEUTSCHMANN (Parameter only)" 0xBC
- Module: "DEUTSCHMANN (Para., Pos, Speed)" 0xBC, 0x51

In the status on delivery the DICNET-ID is set to 0.

## 9.2 Protocol: PLC-SPS

See corresponding instruction manual
- GSD-file: Dagw2079.gsd
- Module: "DEUTSCHMANN (PLC-CSU-ST)" 0xB7, 0x97
- Module: "DEUTSCHMANN (PLC-CSU Long)" 0xB7, 0x9B

**Attention:**
**After a cold start all values in the cam control are deleted.**

**The PLC is master, therefore the programming can only be carried out through the PLC.**

## 9.3 Initiation

Profibus Slave ID: 126 Dez

In the LOCON 24-series the protocol PLC-SPS is supported from V4.03 on and in the LOCON 1, 2, 16, 17-series from the V5.02 on. A project example of the S7-application can be downloaded from our homepage at **http://www.deutschmann.de**.

# 10 Connection ROTARNOCK cam controls via PROFIBUS to a PLC

## 10.1 Protocol: PLC-SPS

As described in this instruction manual
- GSD-file for ROTARNOCK 1, 2, 3: R2pb2935.gsd
- Module: "S7DB+Proc.Data" 0xA7, 0x97, 0x97
- Module: „S7DB+Proc.DataLong“ 0xB7, 0x9B
- GSD-file for ROTARNOCK 4: R4pb3231.gsd
- Module: „S7DB+Proc.Data“ 0xB7, 0xD1, 0x50, 0x9B
- Module: „S7DB+Proc.Data+Logic“ 0xB7, 0xD1, 0x50, 0x9B, 0xA1

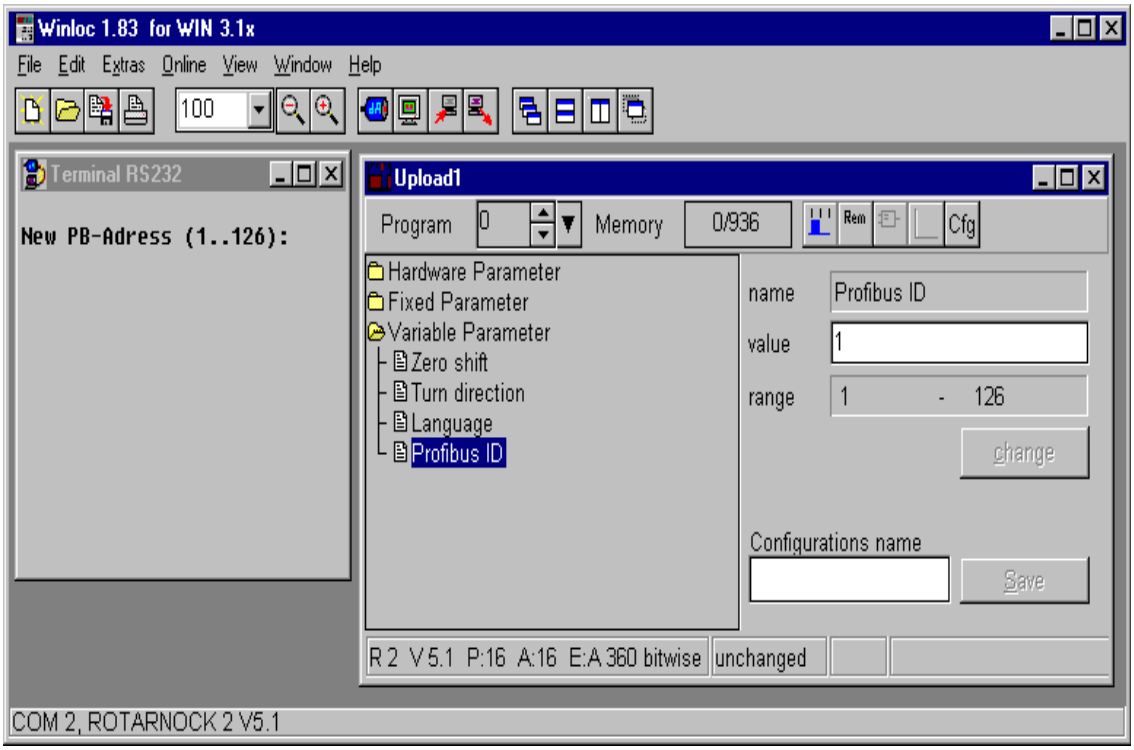
**Attention:**
**After a cold start all values in the cam control are deleted.**

**The PLC is master, therefore the programming can only be carried out through the PLC.**

## 10.2 Initiation

PROFIBUS Slave ID: 126 Dez default

For the parameterization the devices can be reconfigured through a RS232 to a PC with terminal program (STRG + N).
It is more comfortable to configure the device with the current WINLOC software.

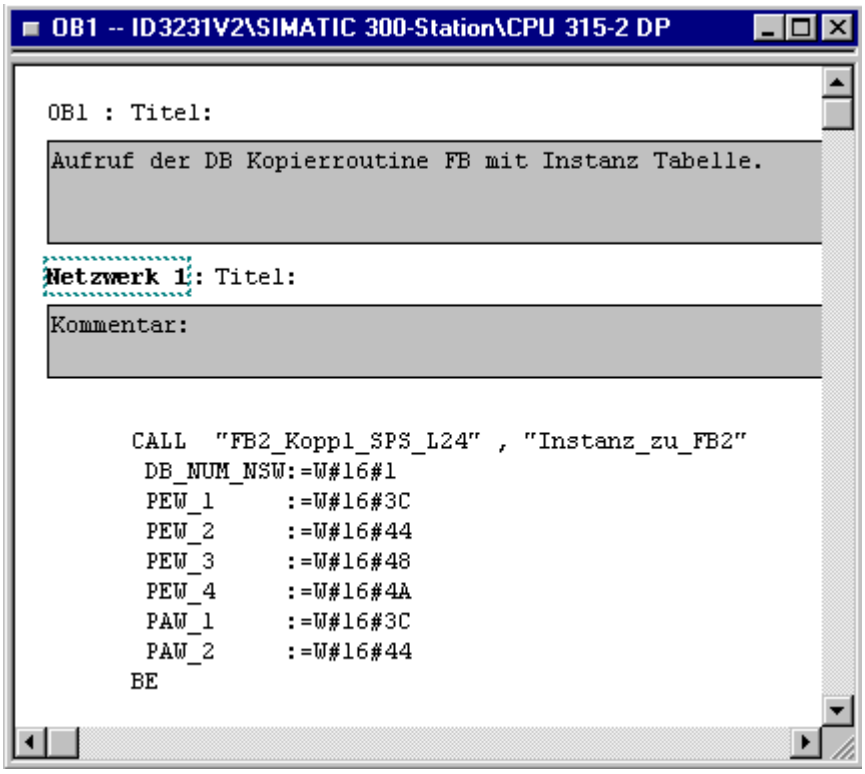



In case the maximum parameter-table size of 2048 byte is exceeded, error 5 is output.
A project example for a S7-application can be downloaded from our homepage at **http://www.deutschmann.de**.

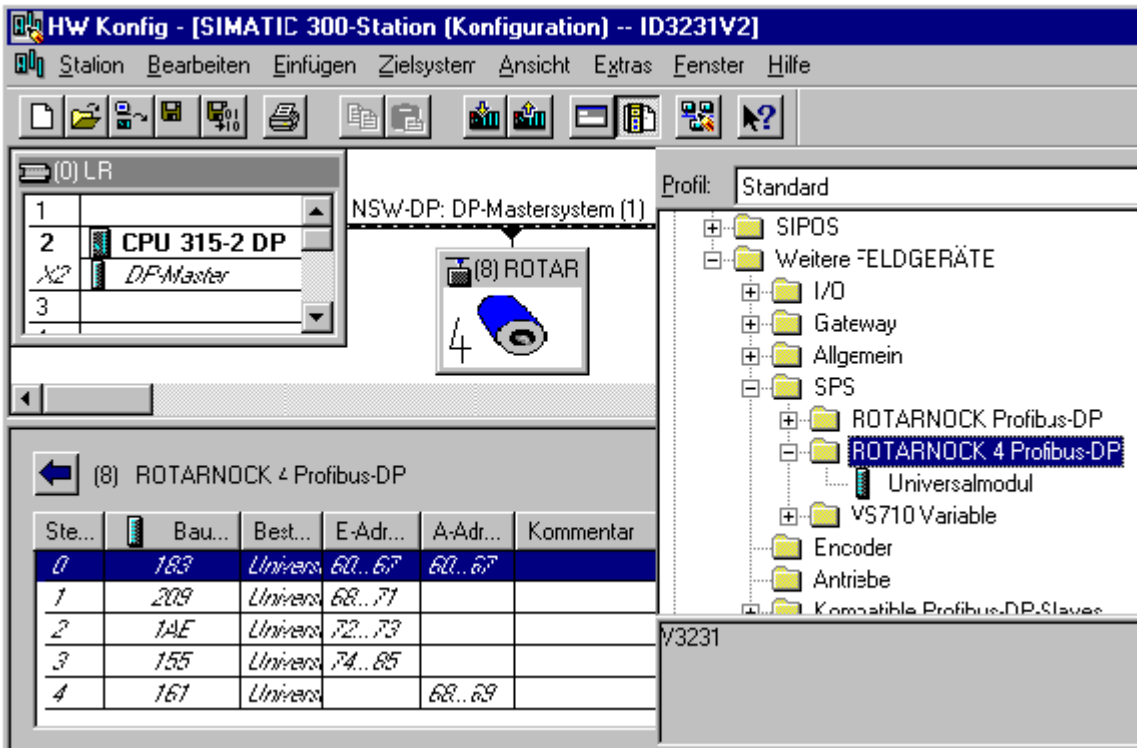
## 10.3 Connecting ROTARNOCK 4 with the PROFIBUS

There is an extra GSD-file for this device r4pb3231.gsd.
The device always features 64 outputs, from which by hardware means the first 16 are located at the plug. All 64 outputs can be presented in the PROFIBUS with the DB-type “PROCESSDATA_80_TYPE“.
The logic inputs are also transferred to the cam control by the PLC. It is the 5th word in the PROFIBUS.

S7-examples for the assignment of I/O-addresses in OB1



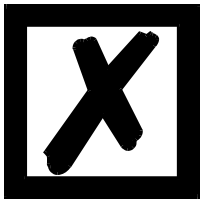
... and in the hardware configurator



# 11 Connection ROTARNOCK cam controls via MPI to S7 300 - 400

## 11.1 Data exchange

The data exchange is carried out automatically through the configured data component (DB), described as parameter-table in this instruction manual.
The DB can automatically be generated by means of the DB-generator (data component generator).
The structure of the DB is described in this instruction manual.



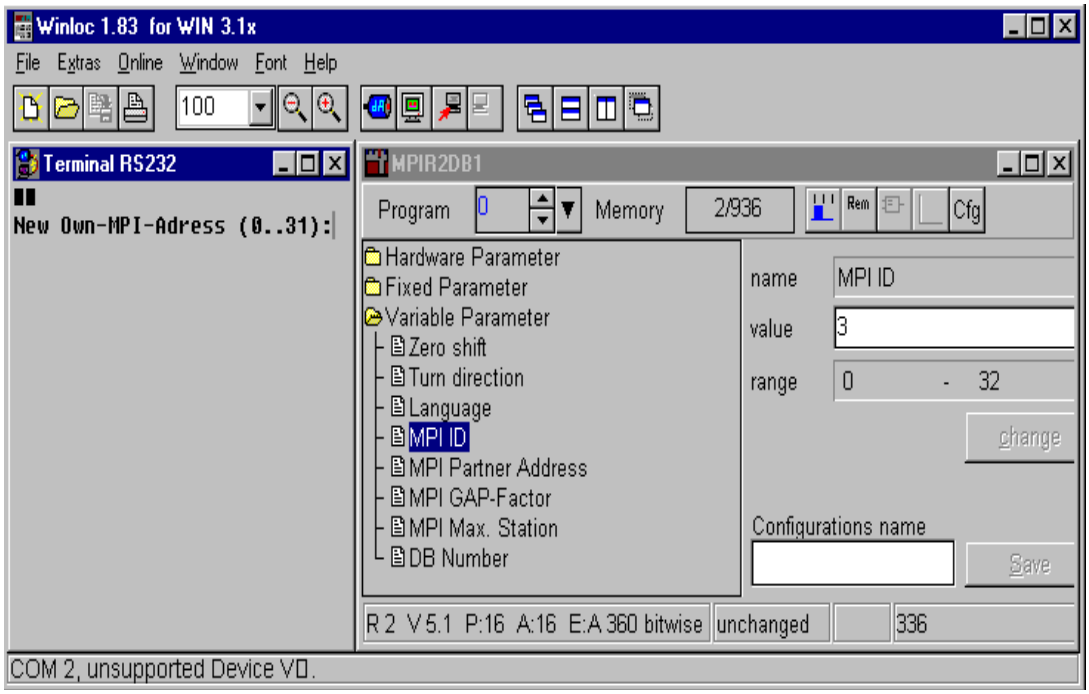
**Attention:**
**After a cold start all values in the cam control are deleted.**

**The PLC is master, therefore the programming can only be carried out through the PLC.**

## 11.2 Initiation

MPI ID: 3 Dez default
MPI Partner Address: 1
MPI GAP Factor: 5 default
MPI Max Station: 31 default
DB Number: 13 default

For the parameterization the devices can be reconfigured through a RS232 to a PC with terminal program (STRG + N).
It is more comfortable to configure the device with the current WINLOC software. Please note that WINLOC contains a teminal program (see picture). However, it is also possible to change the device's configuration via an RS232 at a PC with any terminal program.
In case the maximum parameter-table size of 2048 byte is exceeded, error 5 is output.

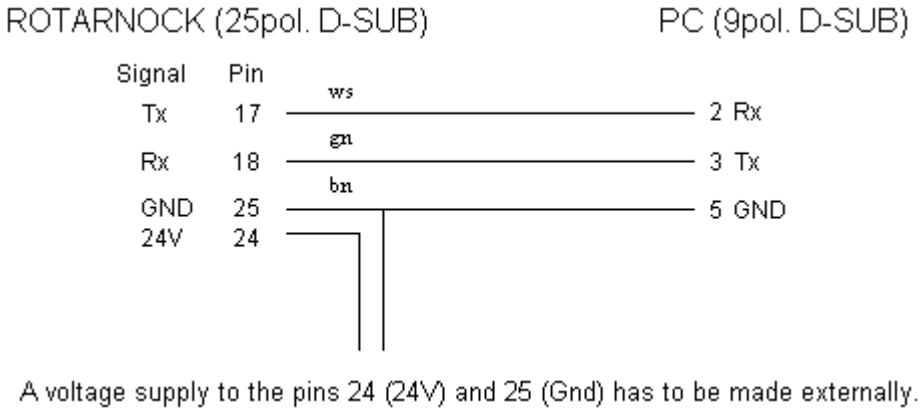


## 11.3 History

| Device | LOCON 24-DP with Profibus soft-ware | | | | ROTARNOCK-DP | | | | UNIGATE MPI | | | | LOCON 24-MPI | | | | MT-ROTARNOCK-MPI | | | | R100 | L100 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | 1, 2, 3 | | 4 | | | | | | | | | | | | | | | |
| Supported table types | 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 | | | | | | | | | | 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 | | | | | | | x | | | | | | | | | | | | | | | |

## 11.4 Programming cable for ROTARNOCK

The programming cable for ROTARNOCK PROFIBUS/MPI is available at the article no. V3467-n*, where n* = cable length in meters.
This cable is required for the configuration of the fieldbus parameters (ID, DB-number etc.) via the RS232-interface.




A voltage supply to the pins 24 (24V) and 25 (Gnd) has to be made externally.

# 12 Adjusting the PROFIBUS- and MPI-ID at LOCON and ROTARNOCK

## 12.1 Adjusting the PROFIBUS-ID

By default the Profibus address is preset to 126 Dez (as already described in chapter 9.3 "Initiation" and chapter 10.2 "Initiation").
To change the ID try one of the following possibilities:

### 12.1.1 Example for S7 with PC-adapter

The PROFIBUS-slave is directly connected to the PC with the programming adapter. By means of a projecting tool, as for instance Step 7 software, the ID can be changed in the Simatic „change Manager-target system-Profibus-address“.

### 12.1.2 Example via the RS232-interface

The proceeding described in the following only applies to ROTARNOCK. The device is connected to the RS232-interface of a PC via the 25-pol. plug (see also chapter “Basic device“, subsection “25-pol. D-SUB“ in the instruction manual “Electronic cam control ROTARNOCK 1, 2, 3...“). In a simple terminal window, e. g. in WINLOC the ID can be changed with the key combination “Clrg + N“ or “Strg + N“ or by means of the “config“ window in WINLOC. This does not apply to the ROTARNOCK version 5.3.

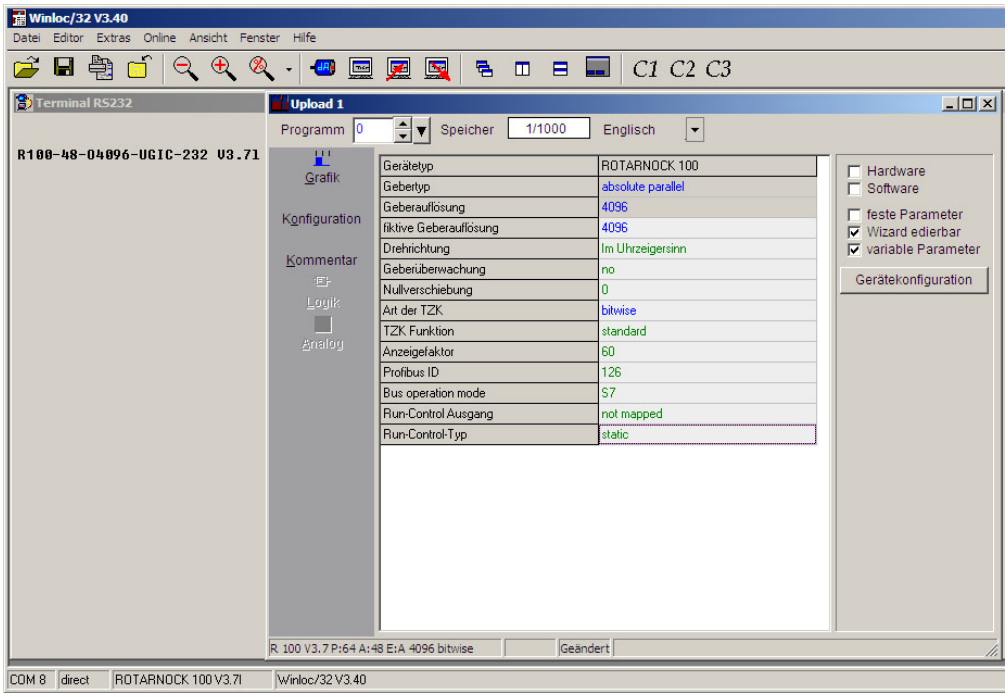
## 12.2 Adjusting the MPI-ID

By default the MPI-address is preset to 3 Dez.
For changing the ID see chapter 12.1.2 "Example via the RS232-interface".

# 13 Connection ROTARNOCK cam controls via PROFINET to S7

## 13.1 Start-up

For parameterizing, the device can be connected to a Terminal or PC with WINLOC32 via an RS 232 interface.
With the current version of WINLOC32 the device can be configured comfortably.




**Attention:**
**The pin assignment for the ROTARNOCK 100-PN can be found in the ROTARNOCK 100 manual in chapter "Pin assignment ROTARNOCK".**

## 13.2 Connection ROTARNOCK 100 to PROFINET

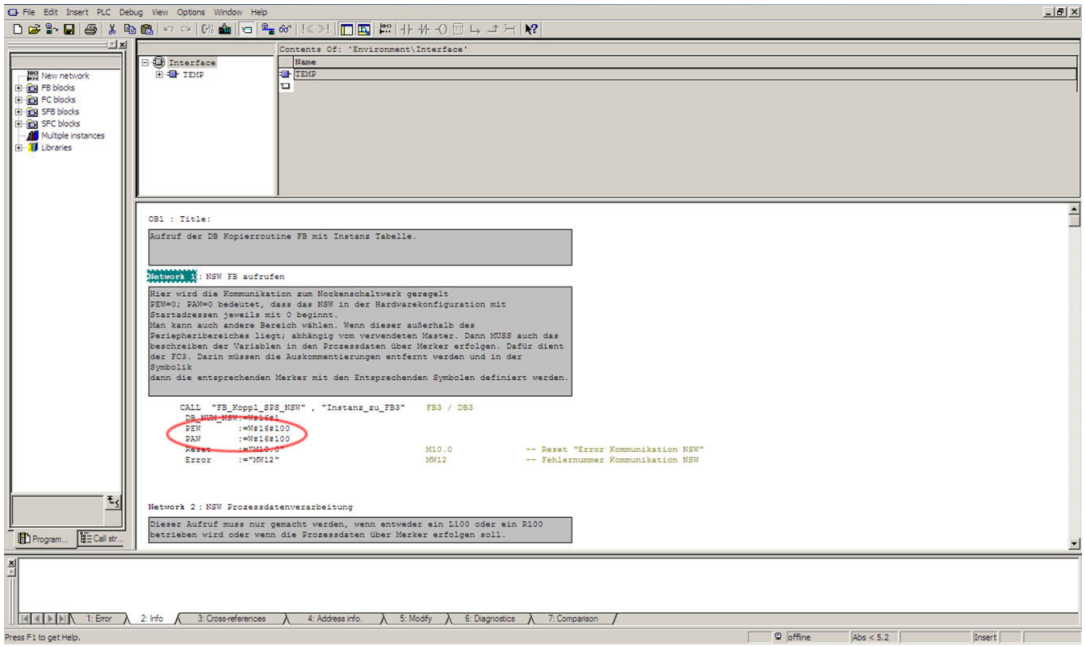
**Attention:**
**The descriptions in the following two chapters refer to the sample project that can be found on the support DVD and on our homepage.**

First, the GSDML file (**Download Link: GSDML-ROTARNOCK100.zip**) needs to be imported. The file can be loaded via the web page of the ROTARNOCK 100-PN (PROFINET) or via the accompanied Support DVD or www.deutschmann.com.
Then the device can be selected and inserted. The address area, which is assigned automatically by the SPS must then be consistent with the in the DB3 deposited PEW and PAW, which are used in OB1. (Picture 13.1.1 and Picture 13.1.2)

Device Name: V3795
DEUTSCHMANN Built-in Cam Control

GSDML-V2.3-Deutschmann-ROTARNOCK 100-PN-20150317-16500.xml
Illustration: GSDML file: 03/15

**Picture 13.1.1:**

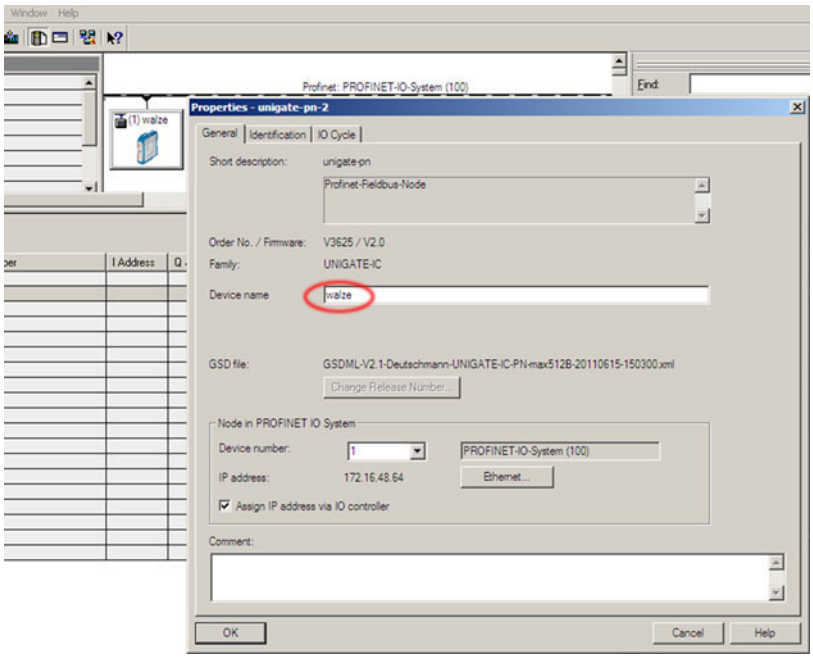


**Picture 13.1.2:**

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, Datenübetragung neu initialisieren |
| 8.0 | out | Error | WORD | W#16#0 | Fehlerausgabe 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 |

**The device name can be changed under properties.**




## 13.3 PROFINET device name

Note: In delivery condition, the gateway does not have a device name!

The device name is assigned to the gateway via the configuration software. Alternatively, the device name can be changed via FTP (file "devname.txt").
For the device name, considering the Profinet specification, the following rules apply:

* It consists of one or more name parts seperated through a full stop
* the total length is from 1 to 240 characters.
* the length of a name part is from 1 up to 63 characters.
* a name consists of only lowercase letters, numbers and the hyphen.
* neither the first nor the last character of a name is a hyphen.
* the first part of the name does not begin with "port-xyz" or "port-xyz-abcde", whereas a, b, c, d, e, x, y and z are seen as numerals.
* it doesn‘t have the form "k,l,m,n", where k, l, m and n are numbers between 0 and 999

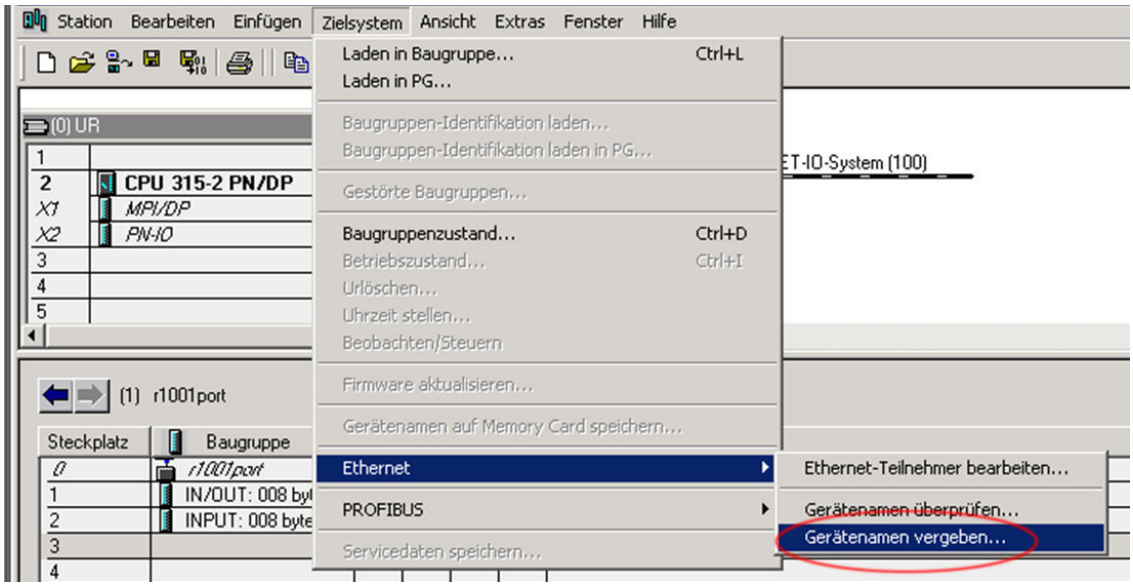
Now that the device has been added to the project, the device name must be stored in the respective device. This must match the current device name, that was entered in the properties window of the device, so that the device can be detected by the master afterwards.

There are two ways to assign the device name to the corresponding device.

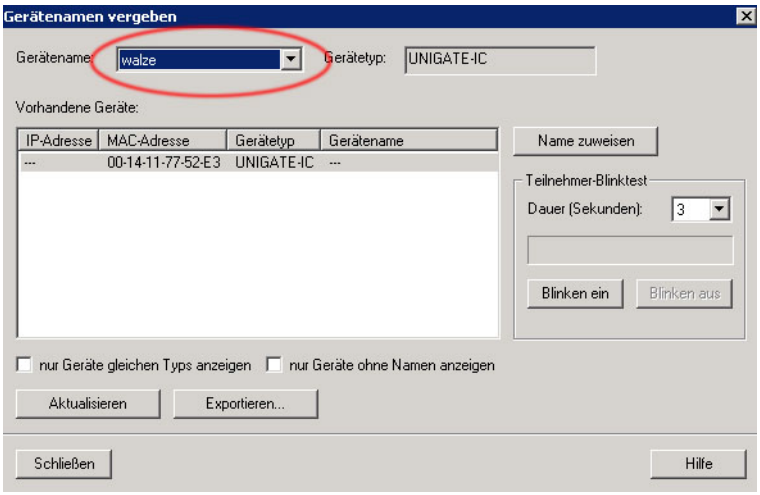
**1. possibility: In the hardware configuration**

First of all the device must be marked in the hardware configuration.

Then go to „Zielsystem“ in the menu and under Ethernet select „Gerätename vergeben“.

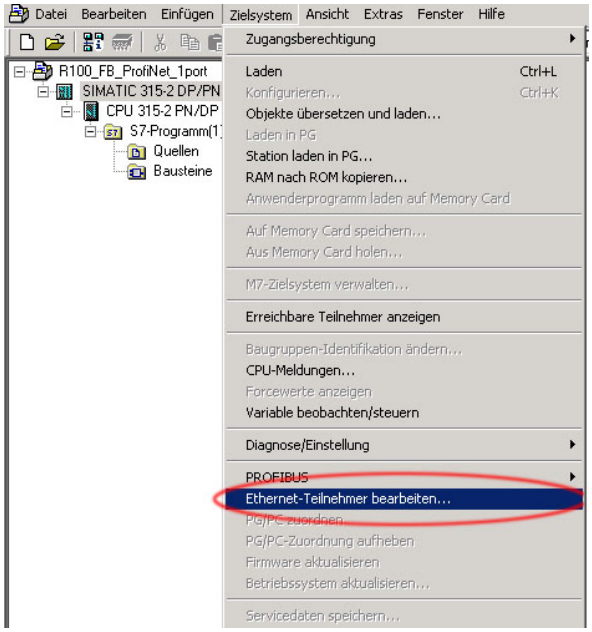



In the next window the desired device name can be chosen via the Dropdown-Menu and can then be assigned via the button „Name zuweisen.“

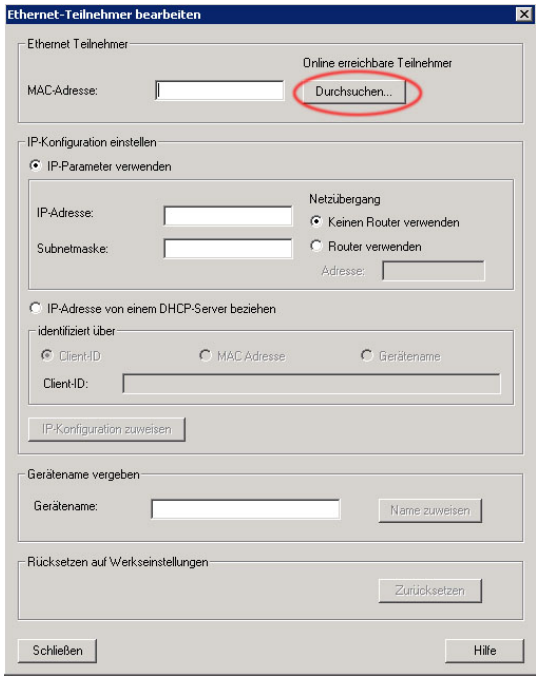


### Second possibility: Project view

In the project via Menu point „Zielsystem“ select „Ethernet-Teilnehmer bearbeiten...)

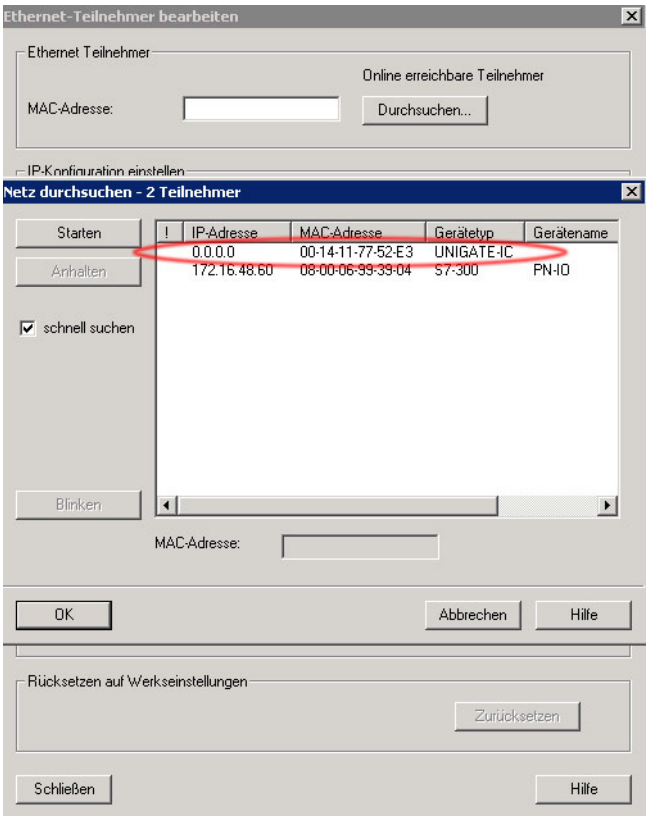



If you click the button „Durchsuchen“ in the following window, all PROFINET participants will be shown.

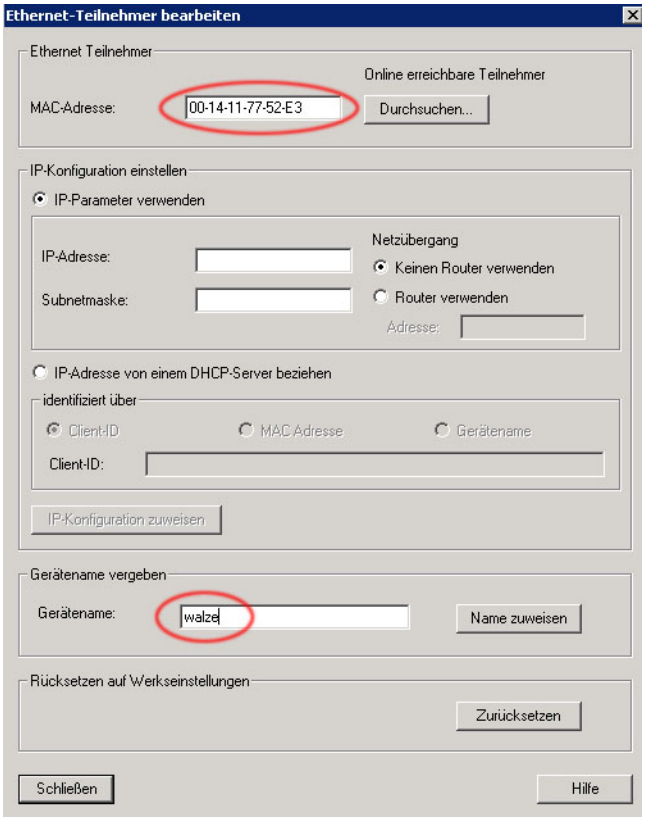


Via the list of the found participants, you can now select the desired participant from the displayed MAC addresses, mark them and confirm it by clicking the „OK“ button.

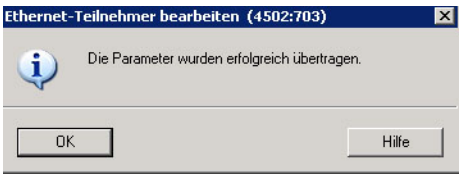
The MAC address of the ROTARNOCK100 can be found on the nameplate.



The subsequent window displays the MAC address of the chosen ROTARNOCK 100. In the lower section of the window, under „Gerätenamen vergeben“, you can now store the device name that you assigned in properties earlier.
With clicking the button „Name zuweisen“ the name will be transmitted.




As a last point an info window will pop up, which tells you that the parameter were transferred successfully.




When the whole project was loaded and transferred, and after you restarted the CPU, the flickering green LED on the ROATARNOCK 100 should be lighting with a static green.

## 13.4 LED-PN

The green LED shows the bus status and can have the following conditions.

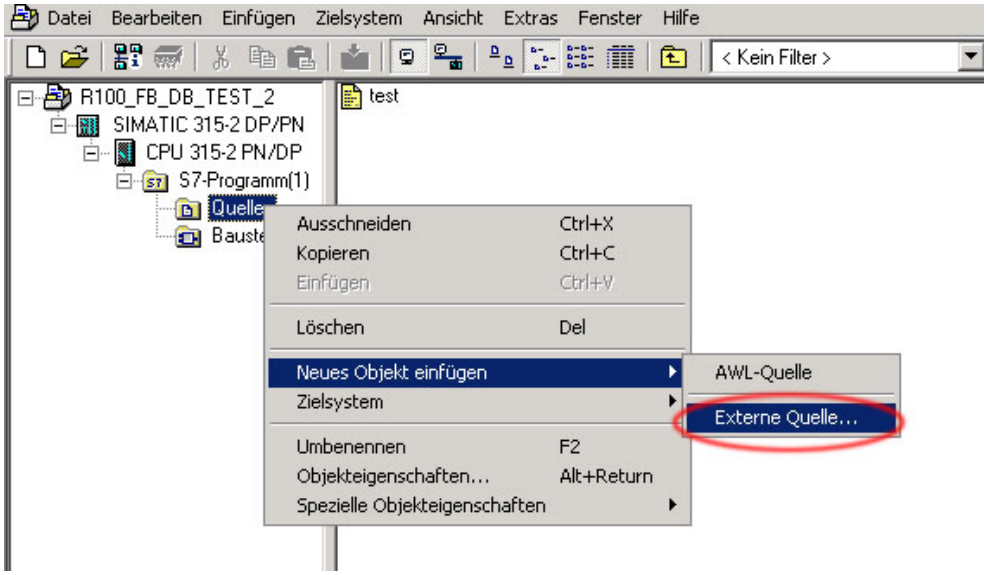
| Condition | LED | State |
|---|---|---|
| Off | - | Bus not started |
| Flash | green/yellow/red | 10 Hz Error |
| Flash | green | 2 Hz Bus started, waiting for connection / configuration mode |
| Flash | green/yellow/red | 1 Hz PROFINET participant flash-test |
| On | green | Connection made |

## 13.5 Programming of the cams

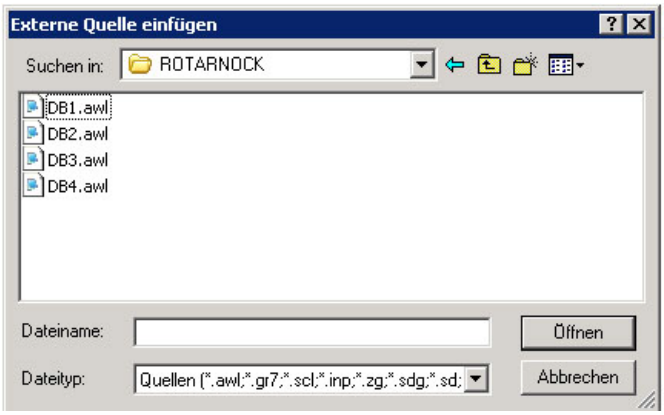
All required parameters are stored in the DB1, so only the parameter table in the AWL format, which is generated by means of WINLOC32 data block generator, must be integrated.

It is possible to carry out the complete programming of the required cams, as well as programs on WINLOC32. These parameters only have to be converted as a parameter table in the AWL format. For this, the same procedure is used when a virgin DB is created, with the exception that immediately after clicking "File" in the menu -> "Create" must be selected. Following you get a AWL file, which can be integrated into the project directly.

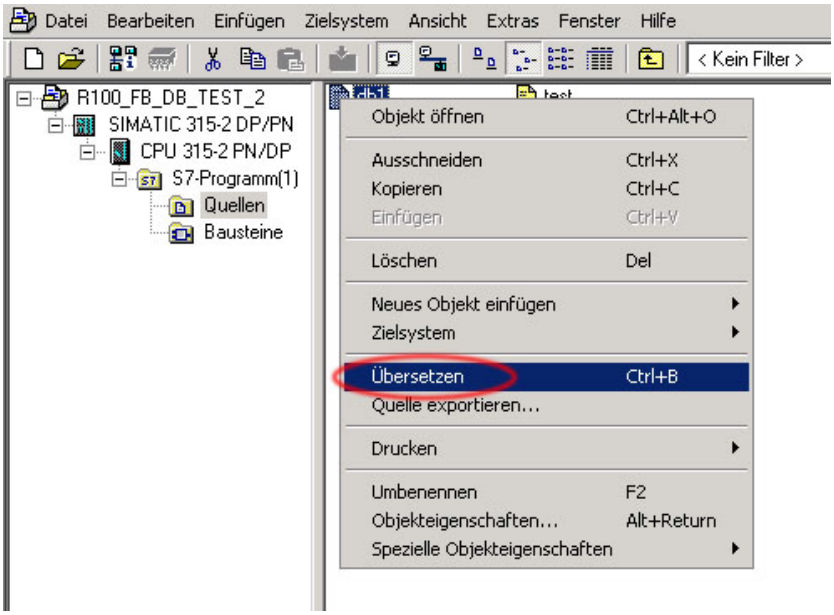
First, highlight "sources" in the project tree. Then, using the right mouse button, choose "Insert New Object". You get to the point „external source“ through „Neues Objekt einfügen“.



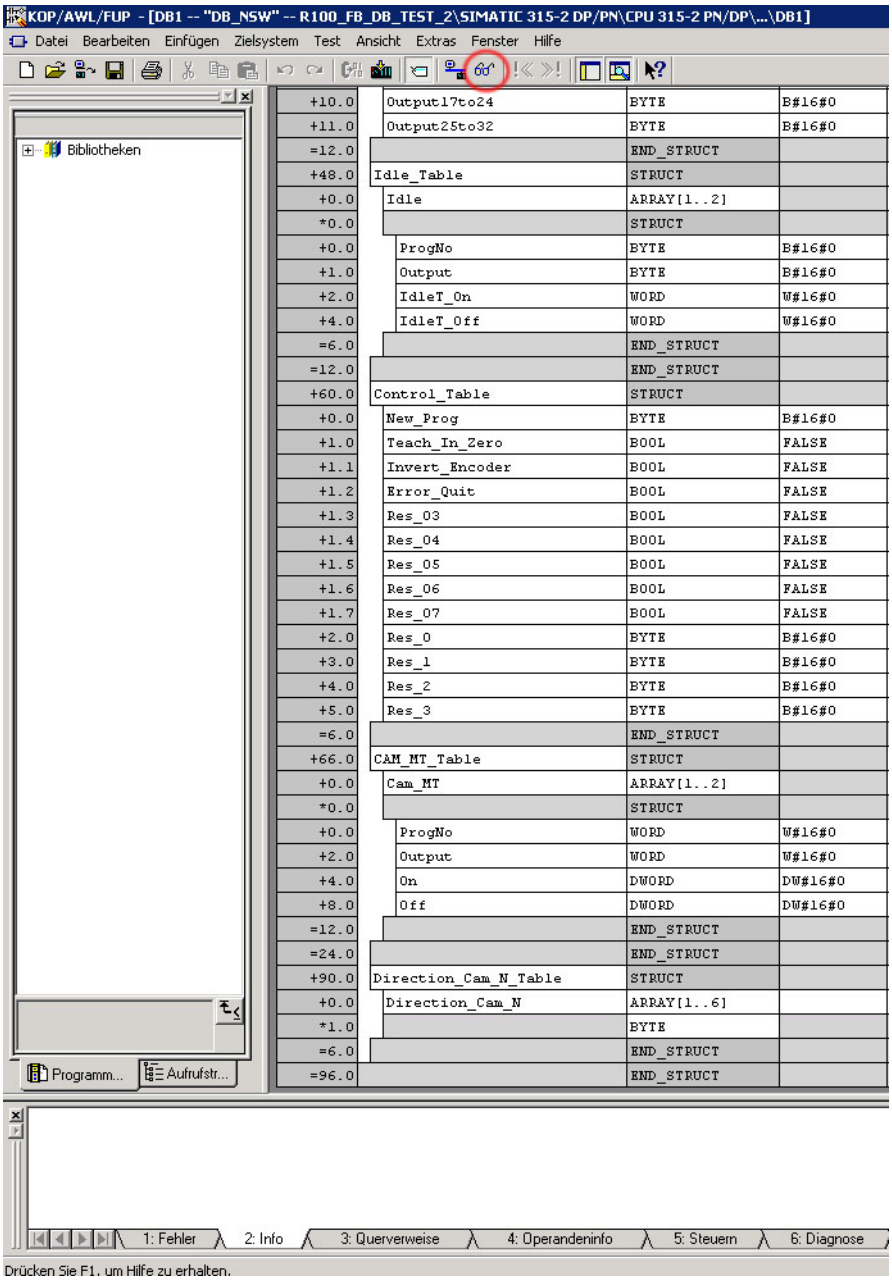
The AWL file can be chosen in the subsequent window.




Highlight the AWL file. The AWL file can now be translated into the data block.

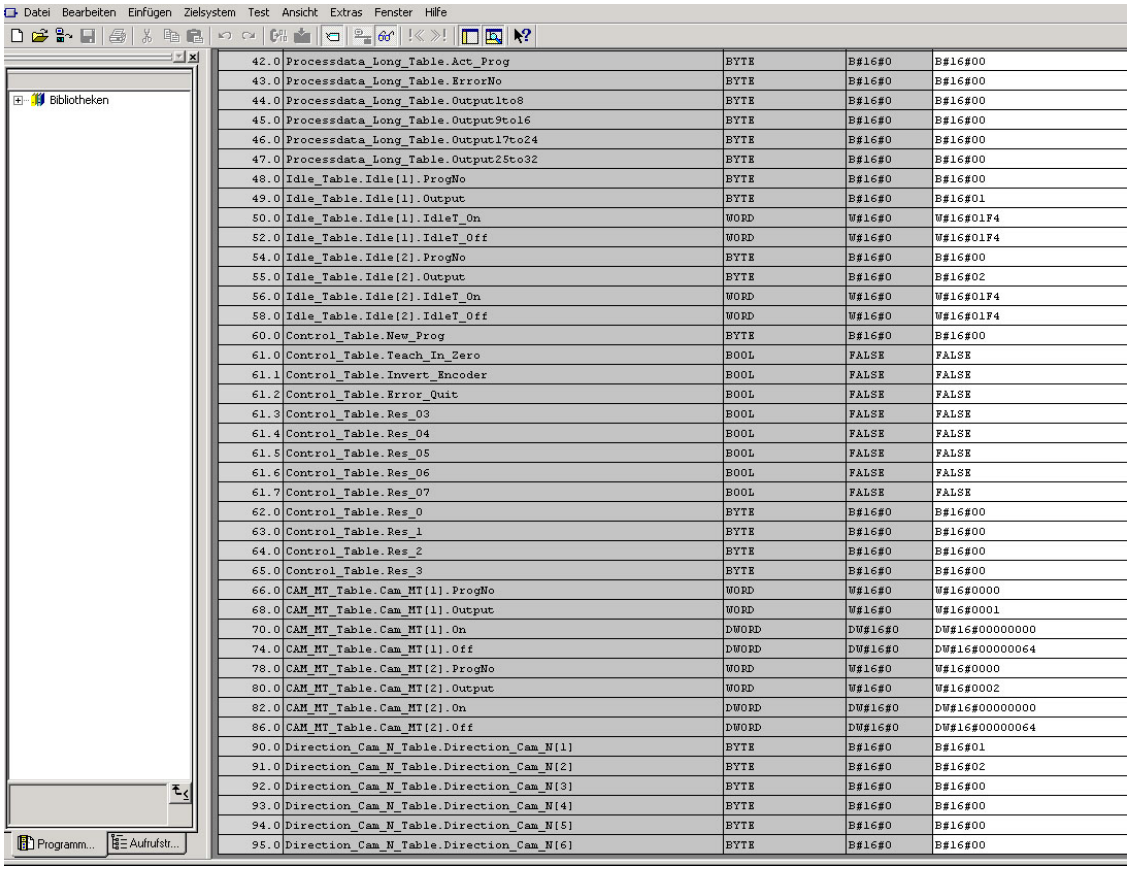


In the online view all parameters which are stored in the parameter table of the AWL file are shown.
For that choose the „glasses“ in DB1.




| | | | |
|---|---|---|---|
| +10.0 | Output17to24 | BYTE | B#16#0 |
| +11.0 | Output25to32 | BYTE | B#16#0 |
| =12.0 | | END_STRUCT | |
| +48.0 | Idle_Table | STRUCT | |
| +0.0 | Idle | ARRAY[1..2] | |
| *0.0 | | STRUCT | |
| +0.0 | ProgNo | BYTE | B#16#0 |
| +1.0 | Output | BYTE | B#16#0 |
| +2.0 | IdleT_On | WORD | W#16#0 |
| +4.0 | IdleT_Off | WORD | W#16#0 |
| =6.0 | | END_STRUCT | |
| =12.0 | | END_STRUCT | |
| +60.0 | Control_Table | STRUCT | |
| +0.0 | New_Prog | BYTE | B#16#0 |
| +1.0 | Teach_In_Zero | BOOL | FALSE |
| +1.1 | Invert_Encoder | BOOL | FALSE |
| +1.2 | Error_Quit | BOOL | FALSE |
| +1.3 | Res_03 | BOOL | FALSE |
| +1.4 | Res_04 | BOOL | FALSE |
| +1.5 | Res_05 | BOOL | FALSE |
| +1.6 | Res_06 | BOOL | FALSE |
| +1.7 | Res_07 | BOOL | FALSE |
| +2.0 | Res_0 | BYTE | B#16#0 |
| +3.0 | Res_1 | BYTE | B#16#0 |
| +4.0 | Res_2 | BYTE | B#16#0 |
| +5.0 | Res_3 | BYTE | B#16#0 |
| =6.0 | | END_STRUCT | |
| +66.0 | CAM_MT_Table | STRUCT | |
| +0.0 | Cam_MT | ARRAY[1..2] | |
| *0.0 | | STRUCT | |
| +0.0 | ProgNo | WORD | W#16#0 |
| +2.0 | Output | WORD | W#16#0 |
| +4.0 | On | DWORD | DW#16#0 |
| +8.0 | Off | DWORD | DW#16#0 |
| =12.0 | | END_STRUCT | |
| =24.0 | | END_STRUCT | |
| +90.0 | Direction_Cam_N_Table | STRUCT | |
| +0.0 | Direction_Cam_N | ARRAY[1..6] | |
| *1.0 | | BYTE | |
| =6.0 | | END_STRUCT | |
| =96.0 | | END_STRUCT | |

## Online view of the 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#00000064 |
| 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#00000064 |
| 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 Reaction time of cam controls at hardware-program shift

The measurement of the reaction time was carried on the example of the devices LOCON 24 MPI, version 4.03 and ROTARNOCK 2 PROFIBUS, version 5.3.
The handling component for ROTARNOCK-DP PLC-SPS was used. This handling component is available for download from our website at http://www.deutschmann.de/download.
Please note that the reaction time depends on the size of the data block.

A Ref DB and a data block were used for LOCON 24 with MPI.

The following check program was additionally inserted into the organization component 1:

```
L EB 0              - Load hardware button for P1
T DB1.DBB234        - Write value in New-ProgNo.
```

## 14.1 Reaction time LOCON 24 MPI

A full cam was programmed in program 1. The MPI-data rate is 187.5 Kbit/s fixed.

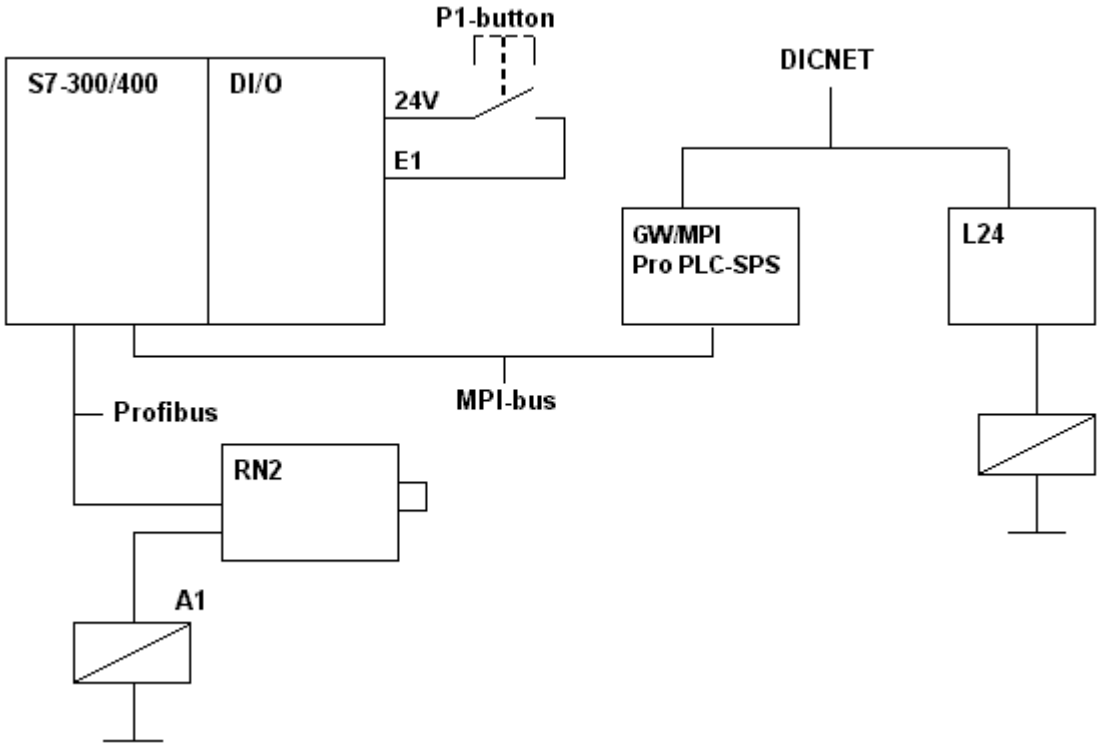
| Switch | Output | Full cam | Reaction time |
|---|---|---|---|
| P1 | A1 | 0-360 | < 1 second |

## 14.2 Reaction time ROTARNOCK 2 - PROFIBUS

A full cam was programmed in program 1

| Switch | Output | Full cam | Reaction time | Profibus-DP data rate |
|---|---|---|---|---|
| P1 | A1 | 0-360 | < 500 ms | 12 Mbit/s |
| P1 | A1 | 0-360 | < 500 ms | 500 Kbit/s |

### Schematical representation of the test setup




| | | |
|---|---|---|
| DI/O | = | Digital In/Out (Siemens connection module) |
| A1 | = | Output 1 |
| E1 | = | Input 1 |
| S7- 300/400 | = | Simatic S7 with PROFIBUS DP |
| RN2 | = | ROTARNOCK 2 -PROFIBUS DP |
| L 24 | = | LOCON 24 |
| GW/MPI Pro PLC-SPS | = | Gateway MPI Protocol PLC-SPS |

# 15 Error messages

Complementary to the error messages (see chapter “Error messages“ in the instruction manual for ROTARNOCK or LOCON 24, 48, 64) the following error messages are possible for cam controls with fieldbus connection:

| Error number | Meaning | Remark |
|---|---|---|
| 6 | Error in the MPI-configuration | MPI-ID > 32 |
| 9 | Error in the internal communication between processor and fieldbus-chip | Restart the device or send it in |
| 36 | Addressed data component not existent in the S7 | E. g. DB1 not existent at Profibus |
| 43 | No connection between ROTARNOCK and S7 | E. g. wrong setting of the ID<br>Defective connection cable |
| 80 | Error in the structure of the S7-data component | DB-number might be set the wrong way.<br>Generate the data component again with the generator |
| 22 | Error when saving a cam value | Wrong value (e. g. too big) |
| 82 | Logic confic error | Logic not configures |

## 15.1 Status LED at the ROTARNOCK

In the operating state the LED lights up red to the „zero point“. A LED that flashes red (4 times as fast as in case of a “regular“ error) indicates an error, which can be analysed by means of the above stated table. The number can either be seen via the diagnosis data from PROFIBUS or via the RS232-interface of the WINLOC-online-window.

# 16 Servicing

Should questions arise that are not covered in this manual you can find further information in our

- FAQ/Wiki area on our homepage www.deutschmann.com or directly in our Wiki on www.wiki.deutschmann.de
- Corresponding Manual of the used Cam Control

If your questions are still unanswered please contact the responsible sales partner (see www.deutschmann.com) or contact us directly.

Please note down the following information before calling:

- Device designation
- Serial number (S/N)
- Article number
- Error number and error description

You can reach us during hotline hours which are as follows:

Monday to Thursday from 8 am to midday and from 1 pm to 4 pm, Friday from 8 am to midday (CET).

Deutschmann Automation GmbH & Co. KG
Carl-Zeiss-Straße 8
D-65520 Bad Camberg
Germany

| | |
|---|---|
| Central office and sales department | +49 6434 9433-0 |
| Technical hotline | +49 6434 9433-33 |
| Fax sales department | +49 6434 9433-40 |
| Fax technical hotline | +49 6434 9433-44 |
| E-mail technical Hotline | hotline@deutschmann.de |


## 16.1 Returning a unit

If you return a unit to us, we require as comprehensive a description of the error as possible. We require the following information in particular:

- What error number was displayed?
- How is the unit externally wired (encoders, outputs, ..)? Please state all connections of the unit.
- What is the magnitude of the 24V supply voltage (± 0.5V) with connected LOCON?
- What were you last doing on the unit (programming, error on power-up, ...)?

The more precise your information and error description, the more precisely we can check the possible causes.

## 16.2 Internet

On our Internet-homepage (URL) various software can be loaded. Beyond that you will also find topical information on Deutschmann products, instruction manuals and a list of our distribution partners.
**URL:** www.deutschmann.de

S7 example projects for:

| | | PROFIBUS ID |
|---|---|---|
| ROTARNOCK 4: | ID3231V1.zip | 8 |
| ROTARNOCK 4 Logic: | 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 / L100: | R100V1Pa.zip | 8 |