PSSu CANopen

Product
Type: PSSu
Name: PSSu, CAN, PDO, SDO, PDO-mapping, Heartbeat
Manufacturer: Pilz GmbH & Co. KG, Safe Automation

Document
Release Number: 03
Release Date: 21 March 2011

Application Note - No. 1001996_EN_03

the spirit of safety

Document Revision History

Release	Date	Changes	Chapter
01	2010-02-12	Creation	all
02	2010-07-28	Make ready for Internet-Release	only logistic data
03	2011-03-21	Adjustments for publication	only editorial

Exclusion of liability

We have taken great care in compiling our application note. It contains information about our company and our products. All statements are made in accordance with the current status of technology and to the best of our know-ledge and belief.
However, we cannot accept liability for the accuracy and entirety of the information provided, except in the case of gross negligence. In particular it should be noted that statements do not have the legal quality of assurances or assured properties.
We are grateful for any feedback on the contents.

March 2011

Support

Technical help round the clock!
Technical support is available from Pilz round the clock.
This service is provided free of charge beyond standard business hours.

Americas
- Brazil
  +55 11 8245-8267
- Mexico
  +52 55 5572 1300
- USA (toll free)
  +1 877-PILZUSA (745-9872)

Asia
- China
  +86 21 62494658-216
- Japan
  +81 45 471-2281
- Korea
  +82 2 2263 9540

Australia
- Australia
  +61 3 95446300

Europe
- Austria
  +43 1 7986263-0
- Belgium, Luxembourg
  +32 9 3217575
- England
  +44 1536 462203
- France
  +33 3 88104000
- Germany
  +49 711 3409-444
- Ireland
  +353 21 4804983
- Italy
  +39 031 789511
- Scandinavia
  +45 74436332
- Spain
  +34 938497433
- Switzerland
  +41 62 88979-30
- The Netherlands
  +31 347 320477
- Turkey
  +90 216 5775552

You can reach our international hotline on:

+49 711 3409-444 or **mailto:support@pilz.com**

Pilz GmbH & Co. KG
Safe Automation
Felix-Wankel-Straße 2
73760 Ostfildern, Germany

Telephone: +49 711 3409-0
Telefax: +49 711 3409-133
E-Mail: pilz.gmbh@pilz.de
Internet: www.pilz.com

Contents

1. Useful documentation

Reading the documentation listed below is necessary for understanding this application note. The availability of the indicated tools and safe handling are also presupposed with the user.

1.1. Documentation from Pilz GmbH & Co. KG

No.	Description	Item No.
1	Pilz international homepage, download section	www.pilz.com
2	Software "PSSuniversal Assistant", Online Help Pilz's international homepage > Products section > Safety control technology > PSSuniversal > Selection guides (download link)	www.pilz.com
3	Operating manual of module "PSSu H CAN" (english)	21 560-EN-xx
4	Operating manual of module "PSSu H SB CAN" (english)	21 459-EN-xx
5	Technical Catalogue of PSSuniversal (english)	1001264-EN-xx
6	System Description of PSSuniversal (english)	21 256-EN-xx
7	Installation Manual of PSSuniversal (english)	21 262-EN-xx
8	Operating manual of Software “ST SB CANopen” (english)	19 886-xx
9	Operating manual of Software “ST SB CANopen Master” (english)	21 877-xx

1.2. Documentation from other sources of information

No.	Description	Item No.
1	CAN in Automation (CiA)	www.can-cia.org
2	Vector Informatik GmbH (Tool ‘CANalyzer’) CiA 301 DS V4.0.2 - CANopen application layer and communication profile Controller Area Network from Konrad Etschberger IXXAT Press	www.vector.com

2. Hardware configuration

- This is a PSSu system with Digital Failsafe an Standard-I/Os:

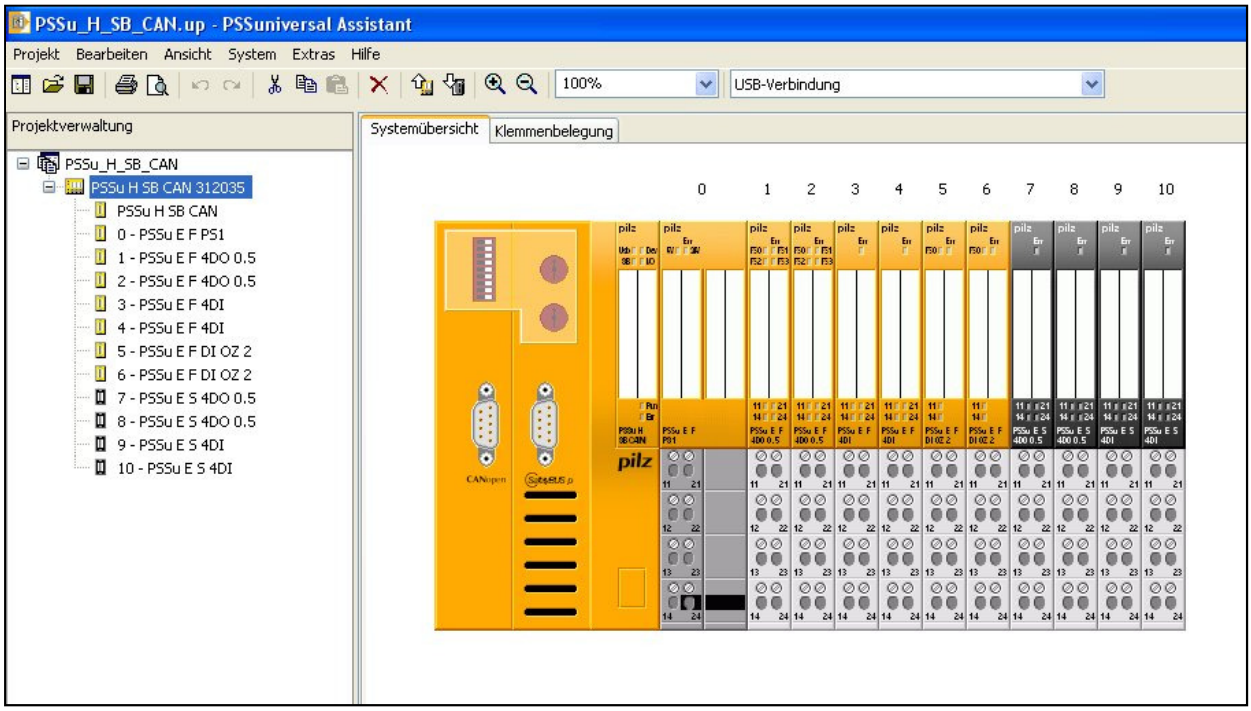


Fig. 1: PSSu System manually configured OR upload via menu System > upload system data

- Bug Fixes and new features are based on the used Hardware and Firmware versions:

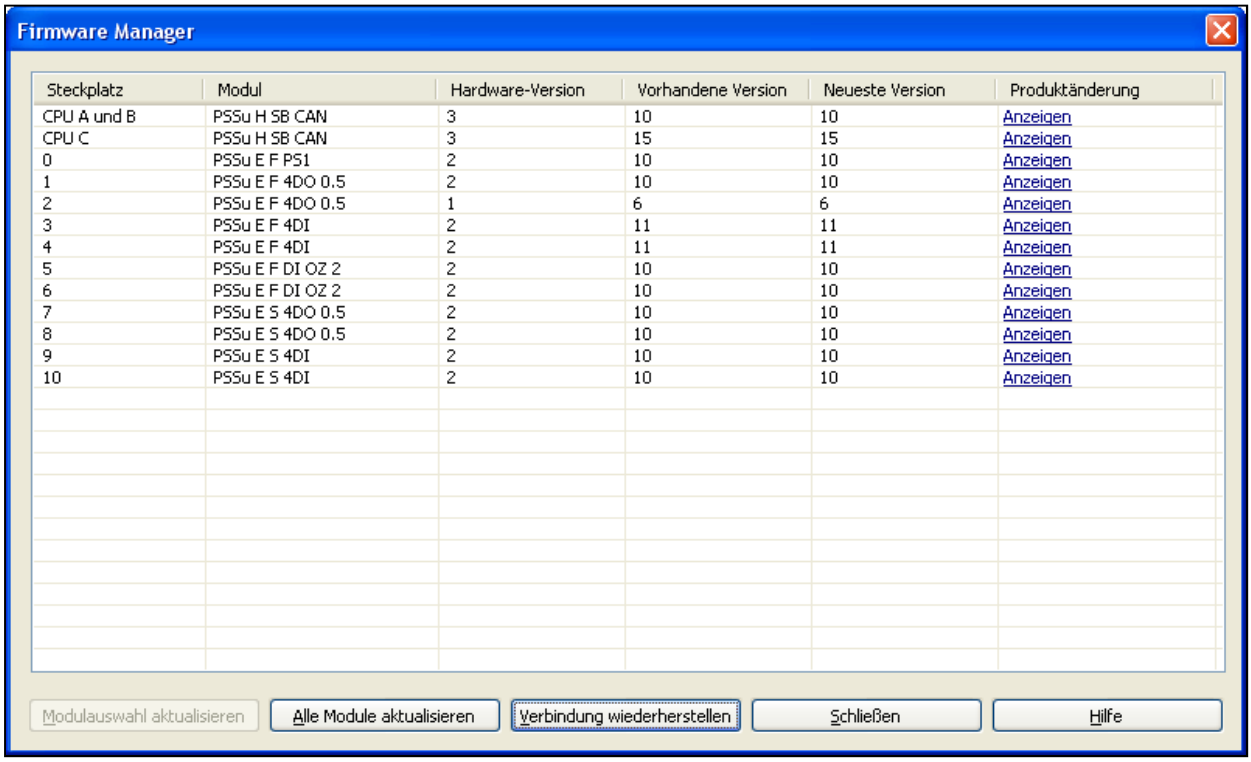


Fig. 2: PSSu Firmware Manager

- Sequence:
  - > Start of PSSu Assistant menu Tools
  - > Firmware Manager OR Start of PSSu Startup menu System
  - > View Online System Information

3. Pilz Application Software

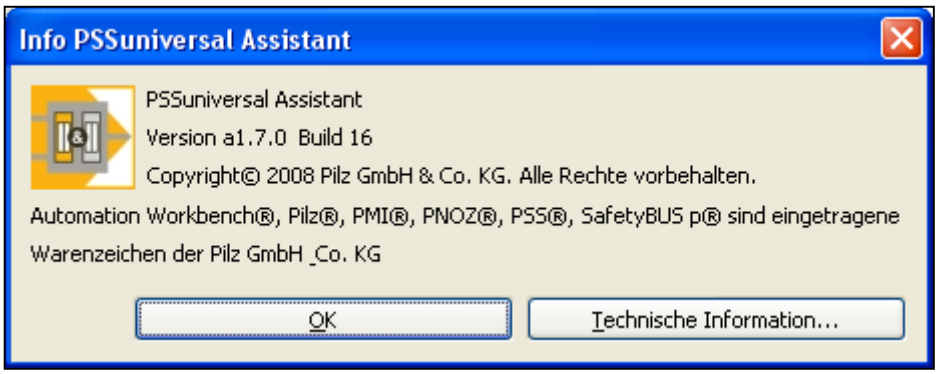


Fig. 3: PSSu Assistant / Configurator Version

4. Third party Application Software

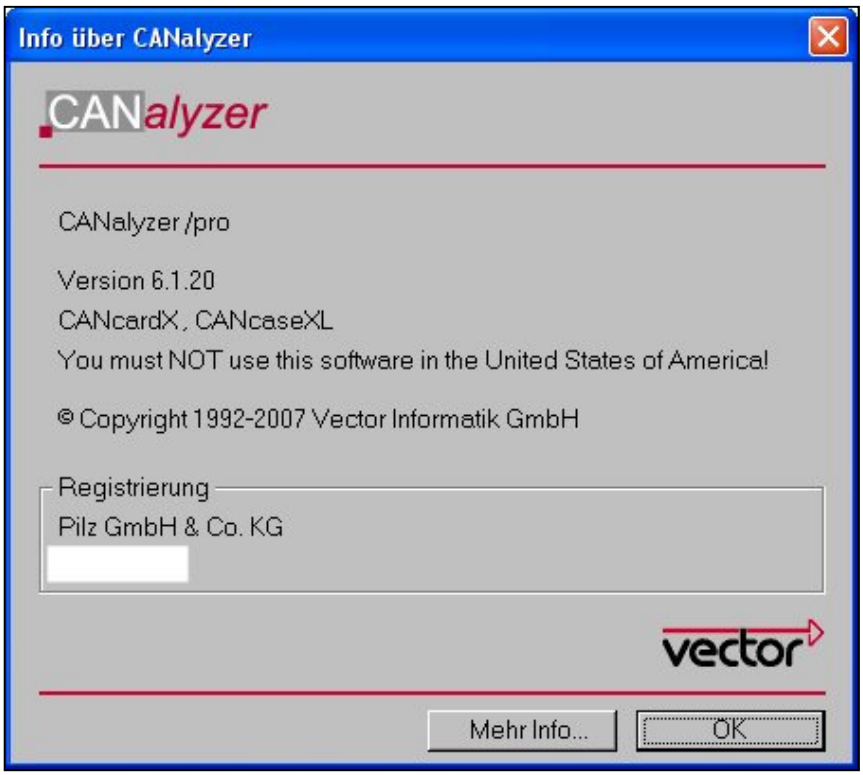


Fig. 4: CANalyzer Version

5. Preface

This is only a very short introduction to the CANopen world to get some know how basics.
In chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** (page **Fehler! Textmarke nicht definiert.**) one can further CANopen details:

5.1. Network Management State (NMT) state machine

The NMT Master is responsible for network management on CANopen. It controls the NMT Slaves. A minimum functionality is required to enable a Master to communicate with Slaves that do not have full NMT Slave functionality. Such NMT Slaves are known as minimum capability devices. They have the following functions:

5.1.1. NMT services:

- Reset_Node
- Enter_Pre_Operational_State
- Start_Remote_Node
- Stop_Remote_Node
- Reset_Communication

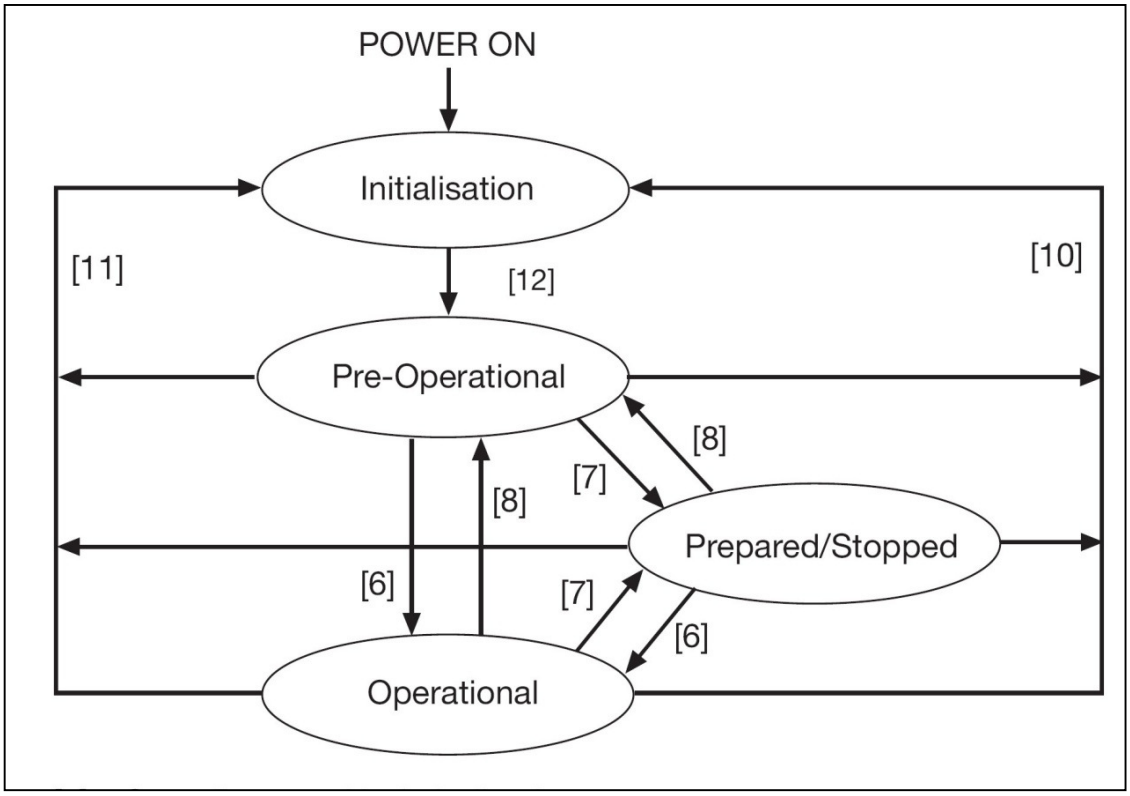


Fig. 5: CANopen Network Management State Machine

5.2. Heartbeat monitoring

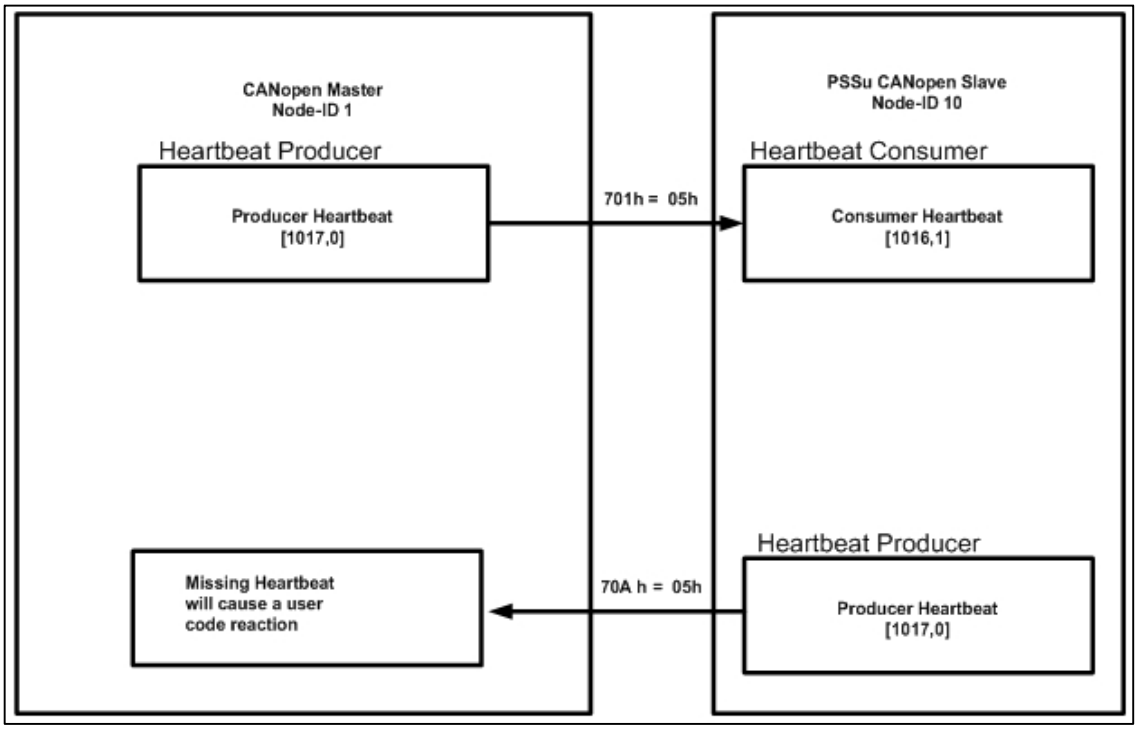


Fig. 6: CANopen node monitoring via Heartbeat

5.2.1. Heartbeat Producer

The CANopen nodes send their respective status message independently. In contrast to the node guarding procedure, remote frames are not required. Less bus load is generated as a result.
As part of each cycle the heartbeat producer sends a heartbeat telegram with its status (e.g. “05h” for “Operational” status), at intervals defined in Object 1017, Subindex 0. This way, other nodes are able to detect a failure of the heartbeat producer.

5.2.2. Heartbeat Consumer

In a Subindex of the Object 1016, the heartbeat consumer of the respective CANopen node is assigned to a heartbeat producer node with a monitoring time constant. Monitoring of the heartbeat producer becomes active once the first heartbeat telegram has been received from the assigned CANopen node.
If a sub-telegram is not triggered within the configured time, heartbeat monitoring is activated. As a rule, the heartbeat producer switches to “Preoperational” status (definition in Object 1029).

- Note:
  In the figure above the CANopen Master doesn’t have a Heartbeat consumer detection. A user code reaction has to be implemented, because the breakdown of a single node shouldn’t switch the I/O-communication (PDO) to the other CANopen nodes.

5.3. Process Data Object (PDO)

PDOs are used for PSSu Input/output transmissions from 1 up to 8 Byte. For a PDO-Transmission you need the Operation-Mode. There are different PDO-Transmission types. The main Transmission types are event driven, triggered by Sync-telegrams or on a request (RTR) The PDOs of all connected CANopen nodes uses unique identifiers (COB-IDs). With the factory settings (predefined connection setup) the Node-ID has an effect on the COB-ID. The COB-ID is displayed normally in hexadecimal numeral.

5.3.1. PDO-Mapping

Properties of a CANopen device are mapped in CANopen objects. There are two different ways to access these objects via SDO or PDO. The PDO-transfer has a higher priority than the SDO-transfer. Map able objects can be mapped to a PDO to use this fast I/O transfer. There are Default- , Static- and Dynamic PDO-Mappings.

5.4. Service Data Object (SDO)

SDO are normally used for the PSSu configuration or non time critical transmissions. For a SDO-Transmission you need the Operation- or Preoperational-Mode. You need SDOs to have read or write access to the PSSu object directory. The SDO read access is called SDO-Upload and the SDO write access is called SDO-Download. The SDOs of all connected CANopen nodes uses unique identifiers. The identifier on a SDO is displayed normally in hexadecimal numeral.

5.5. Object directory/EDS-file

The EDS-file of the PSSu CAN device includes the object directory information’s and the predefined PSSu CANopen node parameters. A CANopen object has a object number and a Subindex. Both are displayed normally in hexadecimal numeral.

5.6. PSSu CANopen I/O properties Part 1

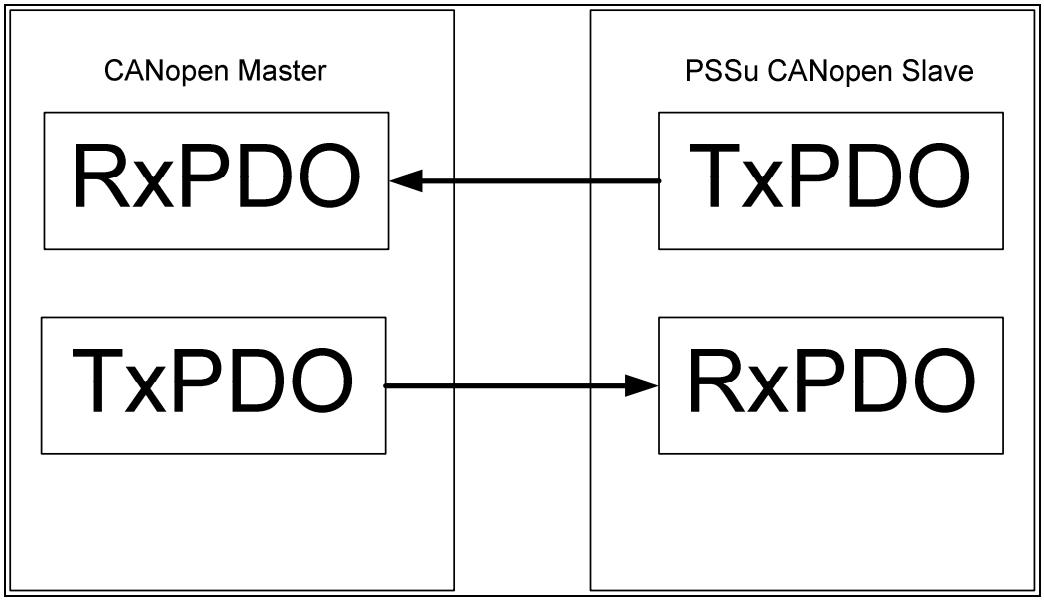


Fig. 7: The PDO-view is based from the device to the CANopen bus.

- The **physical Inputs** of a PSSu-system are represented by **TPDO**s on the CANopen bus.
- The **physical Outputs** of a PSSu-system are represented by **RPDO**s on the CANopen bus.

5.7. Generic Pre-defined Connection Set of PDOs and SDOs

COB	Resultig CAN-ID (Identifier)
TPDO1	0x 181 – 0x 1FF
TPDO2	0x 281 – 0x 2FF
TPDO3	0x 381 – 0x 3FF
TPDO4	0x 481 – 0x 4FF
RPDO1	0x 201 – 0x 27F
RPDO2	0x 301 – 0x 37F
RPDO3	0x 401 – 0x 47F
RDDO4	0x 501 – 0x 57F
SDO(tx)	0x 581 – 0x 5FF
SDO(rx)	0x601 – 0x 67F

Fig. 8: CANopen Identifier

The identifier of the sheet above are the factory setting (default) of all CANopen nodes with CiA-specification. This means a device with factory settings and possible node address from 1 to 127 will cause one of the identifier in the resulting CAN-ID range.

- Example 1:
  A PSSu CANopen node with default settings uses the address 10 (0xA)
  24 Volt at the first digital Input will cause a PDO-telegram with the identifier 0x18A (line 5):

	Time	Chn	Dir	ID	Name	NodeTransfer data		ErrorInterpretation	Data
1									
2	7.497489	1	Rx	70A		Boot-up			00
3	7.497557	1	Tx	0				NMT: Start Node	01 0A
4	7.499703	1	Rx	8A		80 00 00 00 00 00 00 00	E	Error Reset or No Error	80 00 00 00 00 00 00 00
5	56.797423	1	Rx	18A		01 00 00 00 00 00 00 00		PSSu TPDO1	01 00 00 00 00 00 00 00

Fig. 9: PSSu Input creates TPDO

- Example 2:
  A PSSu CANopen node with default settings uses the address 10 (0xA)
  A PDO-telegram with the identifier 0x20A will switch on the first digital Output of the PSSu-System:

1	Time	Chn	Dir	ID	NameNodeTransfer data	ErrorInterpretationData
2	4.031096	1	Tx	20A	01 00 00 00 00 00 00 00	PSSu RPDO1 Bit 0 = RLO-1

Fig. 10: RPDO switch PSSu Output

5.8. PSSu CANopen I/O properties Part 2

5.8.1. PSSu factory settings

- With default setting a defined PSSu node address will have these event driven PDO-Identifiers:

PSSu node: 10

Object	Subindex		default/ predefined	default + node	valid		Object	Subindex		default/ predefined	HEX	valid	
1400	1	RxPDO 01	200	20A	yes	digital	1800	1	TxPDO 01	180	18A	yes	digital
1401	1	RxPDO 02	300	30A	yes	analog	1801	1	TxPDO 02	280	28A	yes	analog
1402	1	RxPDO 03	400	40A	yes	analog	1802	1	TxPDO 03	380	38A	yes	analog
1403	1	RxPDO 04	500	50A	yes	analog	1803	1	TxPDO 04	480	48A	yes	analog
1404	1	RxPDO 05	240	24A	no	digital	1804	1	TxPDO 05	1C0	1CA	no	digital
1405	1	RxPDO 06	340	34A	no	digital	1805	1	TxPDO 06	2C0	2CA	no	digital
1406	1	RxPDO 07	440	44A	no	digital	1806	1	TxPDO 07	3C0	3CA	no	digital
1407	1	RxPDO 08	540	54A	no	digital	1807	1	TxPDO 08	4C0	4CA	no	digital
1408	1	RxPDO 09	260	26A	no	digital	1808	1	TxPDO 09	1E0	1EA	no	digital
1409	1	RxPDO 10	360	36A	no	digital	1809	1	TxPDO 10	2E0	2EA	no	digital
140A	1	RxPDO 11	460	46A	no	digital	180A	1	TxPDO 11	3E0	3EA	no	digital
140B	1	RxPDO 12	560	56A	no	analog	180B	1	TxPDO 12	4E0	4EA	no	analog

Fig. 11: Get default COB-ID from Node ID 2009-01-20_Get_COBID_PDO_PSS_PSSu_mc6p

Only T/RPDOs1 to T/RPDOs 4 are enabled (valid). T/RPDOs5 to T/RPDOs 12 are defined, but disabled in default mode. The factory settings include only TPDO 1 for digital Inputs and RPDO1 for digital Output.

- Note:
  This means with default setting you can use max. 8 digital Input slots and 8 digital Output slots.

5.8.2. CANopen Object 3001: PSSu default hardware registry

After the power on boot up and a change of the hardware registry of the PSSu there will be new entries in the objects 3001.
The first Object 3001 Subindex 0 [3000,0] represents the amount of active PSSu slots and the maximum amount of following Subindexes. The following Subindexes include the Hardware ID and configuration hexadecimal number of each PSSu slot. These objects can be modified by the user. All CANopen 3001 objects are retentive.

- Note:
  You can't download the wrong PSSu configuration, because this is blocked by the PSSu C-Channel firmware.

The possible Hardware ID and configuration is written in the EDS-file.
Pullout of PSSu H SB CAN EDS file V4.0:

```
[3001]
SubNumber=65
ParameterName=SollConfiguration
ObjectType=0x0008

[3001sub0]
ParameterName=Number of Modules
ObjectType=0x0007
DataType=0x05
AccessType=rw
LowLimit=1
HighLimit=64
DefaultValue=64
PDOMapping=0

; Values for 3001sub1 ... 64 (Subindizes of 3001) are following Modul Types:
;  1 -> "PSSu E F PS"
;  2 -> "PSSu E F PS R"
;  3 -> "PSSu E F PS1"
;  4 -> "PSSu E F PS1 R"
;  5 -> "PSSu E F PS-P"
;  6 -> "PSSu E F BSW"
;  7 -> "PSSu E F BSW R"
;  8 -> "PSSu E F BSW R*"
;  9 -> "PSSu E S 4DI"
;  10 -> "PSSu E S 4DI*"
;  11 -> "PSSu E S 4DO 0.5"
;  12 -> "PSSu E S 4DO 0.5*"
;  13 -> "PSSu E S 2DO 2"
;  14 -> "PSSu E S 2DO 2*"
;  15 -> "PSSu E F 4DI"
;  16 -> "PSSu E F 4DI R"
;  17 -> "PSSu E F 4DI R*"
```

```
;  18 -> "PSSu E F 4DO 0.5"
;  19 -> "PSSu E F 4DO 0.5 R"
;  20 -> "PSSu E F 4DO 0.5 R*"
;  21 -> "PSSu E F 4DO 0.5 &"
;  22 -> "PSSu E F 4DO 0.5 &*"
;  23 -> "PSSu E F 2DO 2"
;  24 -> "PSSu E F 2DO 2 R"
;  25 -> "PSSu E F 2DO 2 R*"
;  26 -> "PSSu E F 2DO 2 &"
;  27 -> "PSSu E F 2DO 2 &*"
;  28 -> "PSSu E F 2DOR 8"
;  29 -> "PSSu E F 2DOR 8 R"
;  30 -> "PSSu E F 2DOR 8 R*"
;  31 -> "PSSu E F 2DOR 8 &"
;  32 -> "PSSu E F 2DOR 8 &*"
;  33 -> "PSSu E F DI OZ 2"
;  34 -> "PSSu E F DI OZ 2 R"
;  35 -> "PSSu E F DI OZ 2 R*"
;  36 -> "PSSu E F DI OZ 2 &"
;  37 -> "PSSu E F DI OZ 2 &*"
;  38 -> "PSSu E S 4AI U  "
;  39 -> "PSSu E S 4AI U X "
;  40 -> "PSSu E S 4AI U S "
;  41 -> "PSSu E S 2AI U "
;  42 -> "PSSu E S 2AI U X"
;  43 -> "PSSu E S 2AI U S "
;  44 -> "PSSu E S 2AI I"
;  45 -> "PSSu E S 2AI I X "
;  46 -> "PSSu E S 2AI I S "
;  47 -> "PSSu E S 4AO U "
;  48 -> "PSSu E S 2AO U"
;  49 -> "PSSu E S 2AO I"
;  50 -> "PSSu E S ABS SSI"
;  51 -> "PSSu E S INC"
;  52 -> "PSSu E S 2AI RTD"
;  53 -> "PSSu E S 2AI RTD X"
;  54 -> "PSSu E S 2AI TC"
;  55 -> "PSSu E S 2AI TC X"
;  56 -> "PSSu CB F"
;  57 -> "PSSu CR F"
;  58 -> "PSSu WB S IDN"
;  59 -> "PSSu WR S IDN"
;  60 -> "PSSu E S 2DOR 10"
;  61 -> "PSSu E S 2DOR 10*"
;  62 -> "PSSu E S 4DO 0.5 -D"
;  63 -> "PSSu E S 4DO 0.5 -D*"
;  64 -> "PSSu E S 4DO 0.5 -D S"
;  65 -> "PSSu E S 4DO 0.5 -D S*"
;  66 -> "PSSu E S 2DO 2 -D"
```

```
; 67 -> "PSSu E S 2DO 2 -D*"
; 68 -> "PSSu E S 2DO 2 -D S"
; 69 -> "PSSu E S 2DO 2 -D S*"
; 70 -> "PSSu E S PD -D"
; 71 -> "PSSu E S PD -D S"
; 72 -> "PSSu E S 2DOR 2"
; 73 -> "PSSu E S 2DOR 2*"
```

Fig. 12: Hardware Registry comments from a PSSu EDS-file

5.9. CANopen Process Image of digital Input and Output electronic modules

Without a CANopen 3001-Object modification you can use the PSSu with factory settings This is the PSSu H SB CAN System used for the following examples:

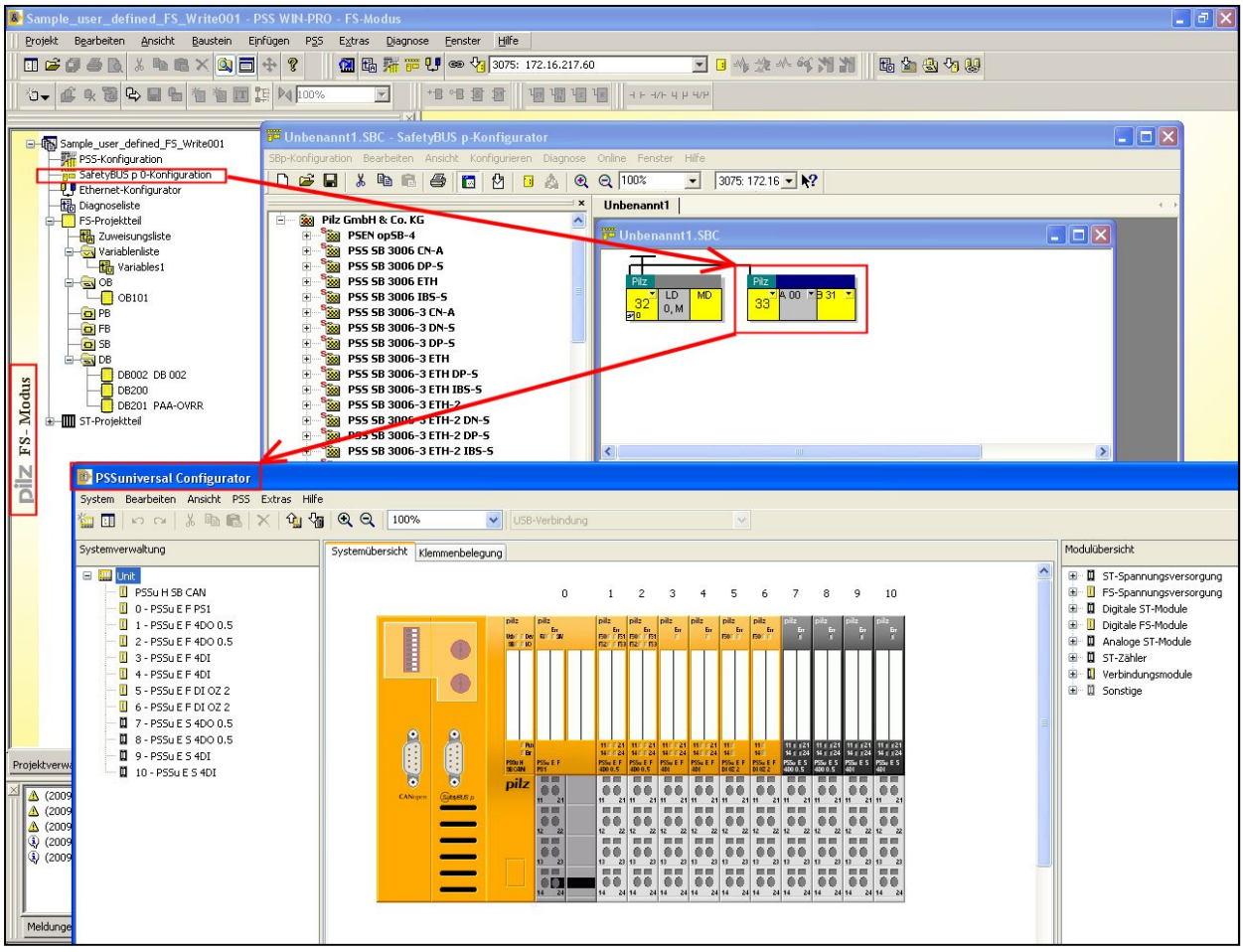


Fig. 13: Start of the PSSu Configurator via WIN-PRO and SafetyBUSp configurator.

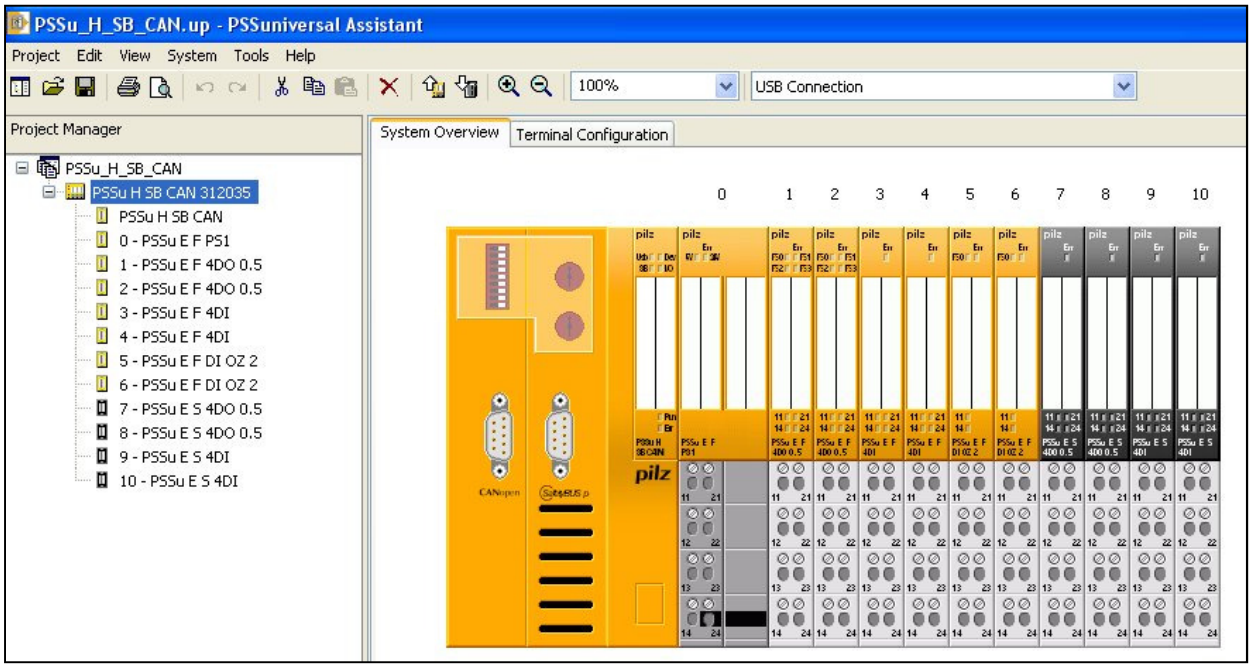


Fig. 14: No changes in PSSu Configurator / Assistant

On factory setting you have to do configure nothing if you use the digital I/Os only .The user has to be sure that the 3001-object are in default state. These settings can be made reset by changing the PSSu sequence and power up the PSSu system later on.

- The hardware registry objects 3001 are untouched in this default state:

PSSu Device Hardware Registry								
			Object	Sub-index		decimal	hexadecimal	
Slot	00	03 : PSSu E F PS1	[3001,	1	]:=	03	3	
Slot	01	18 : PSSu E F 4DO 0.5	[3001,	2	]:=	18	12	
Slot	02	18 : PSSu E F 4DO 0.5	[3001,	3	]:=	18	12	
Slot	03	15 : PSSu E F 4DI	[3001,	4	]:=	15	F	
Slot	04	15 : PSSu E F 4DI	[3001,	5	]:=	15	F	
Slot	05	33 : PSSu E F DI OZ 2	[3001,	6	]:=	33	21	
Slot	06	33 : PSSu E F DI OZ 2	[3001,	7	]:=	33	21	
Slot	07	11 : PSSu E S 4DO 0.5	[3001,	8	]:=	11	B	
Slot	08	11 : PSSu E S 4DO 0.5	[3001,	9	]:=	11	B	
Slot	09	09 : PSSu E S 4DI	[3001,	A	]:=	09	9	
Slot	10	09 : PSSu E S 4DI	[3001,	B	]:=	09	9	

Fig. 15: Each PSSu slot uses a Process Image Byte

Slot 00	03 : PSSu E F PS1		
Slot 01	18 : PSSu E F 4DO 0.5		no ST Input or Output
Slot 02	18 : PSSu E F 4DO 0.5		no ST Input or Output
Slot 03	15 : PSSu E F 4DI		no ST Input
Slot 04	15 : PSSu E F 4DI		no ST Input
Slot 05	33 : PSSu E F DI OZ 2		no ST Input or Output
Slot 06	33 : PSSu E F DI OZ 2		no ST Input or Output
Slot 07	11 : PSSu E S 4DO 0.5		COB-ID 20A Byte 0
Slot 08	11 : PSSu E S 4DO 0.5		COB-ID 20A Byte 1
Slot 09	09 : PSSu E S 4DI		COB-ID 18A Byte 0
Slot 10	09 : PSSu E S 4DI		COB-ID 18A Byte 1

Fig. 16: Resulting PSSu COB-IDs

	Time	ID	NameNodeTransfer data		ErrorInterpretation	Data
1	1.847973	0			Reset	81 0A
2	0.123509	70A	Boot-up			00
3	0.516475	0			Start Node	01 0A
4	0.000952	8A	80 00 00 00 00 00 00 00	E	Error Reset or No Error	80 00 00 00 00 00 00 00
5	2.863176	20A	0f 0f 00 00 00 00 00 00		switch PSSu Output 0..7 on	0F 0F 00 00 00 00 00 00
6	0.920024	20A	00 00 00 00 00 00 00 00		switch PSSu Output 0..7 off	00 00 00 00 00 00 00 00

Fig. 17: CAN-trace switch all Outputs on

- The 2 X 4 =8 Inputs of the two PSSu electronic modules needs 2 PDO-Bytes:

	Time	ID	NameNodeTransfer data	
1	4.768875	18A	01 00 00 00 00 00 00 00	PSSu Slot 09: Input 0 Clamp 11 = 24V
2	0.756716	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 0 Clamp 11 = 00V
3	3.298970	18A	02 00 00 00 00 00 00 00	PSSu Slot 09: Input 1 Clamp 21 = 24V
4	0.686571	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 1 Clamp 21 = 00V
5	2.573039	18A	04 00 00 00 00 00 00 00	PSSu Slot 09: Input 2 Clamp 14 = 24V
6	0.902246	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 2 Clamp 14 = 00V
7	3.212377	18A	08 00 00 00 00 00 00 00	PSSu Slot 09: Input 3 Clamp 24 = 24V
8	0.463982	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 3 Clamp 24 = 00V
9	3.243607	18A	00 01 00 00 00 00 00 00	PSSu Slot 10: Input 4 Clamp 11 = 24V
10	0.577627	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 4 Clamp 11 = 00V
11	1.978989	18A	00 02 00 00 00 00 00 00	PSSu Slot 10: Input 5 Clamp 21 = 24V
12	0.393267	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 5 Clamp 21 = 00V
13	1.791362	18A	00 04 00 00 00 00 00 00	PSSu Slot 10: Input 6 Clamp 14 = 24V
14	0.450010	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 6 Clamp 14 = 00V
15	1.494184	18A	00 08 00 00 00 00 00 00	PSSu Slot 10: Input 7 Clamp 24 = 24V
16	0.497439	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 7 Clamp 24 = 00V

Fig. 18: CAN-trace of digital Inputs.

- Note:
  The Transmission types of the PDOs are “event driven”, so each signal change (0V => 24V or 24V => 0V) will cause one telegram only.

5.10. Optimised ST Process Image, no FS access from field bus side

You can only optimise the Process Image of similar PSSu electronic modules, if there are adjoining. This means no other active PSSu electronic module separate the *-configured from the start module.
The Hardware registry in Object 3000 Subindex Slot 8 and Slot 10 have to be changed by SDO-downloads to use the high nibble of Byte 0 in the Process Image of Inputs and Outputs.

			PSSu Device Hardware Registry					
				Object	Sub-index		decimal	hexadecimal
Slot	00	03 : PSSu E F PS1		[3001,	1	]:=	03	3
Slot	01	18 : PSSu E F 4DO 0.5		[3001,	2	]:=	18	12
Slot	02	18 : PSSu E F 4DO 0.5		[3001,	3	]:=	18	12
Slot	03	15 : PSSu E F 4DI		[3001,	4	]:=	15	F
Slot	04	15 : PSSu E F 4DI		[3001,	5	]:=	15	F
Slot	05	33 : PSSu E F DI OZ 2		[3001,	6	]:=	33	21
Slot	06	33 : PSSu E F DI OZ 2		[3001,	7	]:=	33	21
Slot	07	11 : PSSu E S 4DO 0.5		[3001,	8	]:=	11	B
Slot	08	12 : PSSu E S 4DO 0.5*		[3001,	9	]:=	12	C
Slot	09	09 : PSSu E S 4DI		[3001,	A	]:=	09	9
Slot	10	10 : PSSu E S 4DI*		[3001,	B	]:=	10	A

Fig. 19: Slot 8 and Slot 10 uses the process image high nibble of the previous Process Image

	Time	ID	NameNodeTransfer data	ErrorInterpretation	Data
2	1.231993	0		Reset	81 0A
3	0.123657	70A	Boot-up		00
4	11.088459	60A	0c 00	[3001,09] Initiate Download Rq. expedited	2B 01 30 09 0C 00 00 00
5	0.001112	58A		[3001,09] Initiate Download Rsp	60 01 30 09 00 00 00 00
6	3.096353	60A	0a 00	[3001,0b] Initiate Download Rq. expedited	2B 01 30 0B 0A 00 00 00
7	0.000878	58A		[3001,0b] Initiate Download Rsp	60 01 30 0B 00 00 00 00
8	9.873484	0		Start Node	01 0A
9	0.001242	8A	80 00 00 00 00 00 00 00 E	Error Reset or No Error	80 00 00 00 00 00 00 00
10	5.310868	20A	ff 00 00 00 00 00 00 00	switch PSSu Outputs 0..7 on	FF 00 00 00 00 00 00 00
11	4.240034	20A	00 00 00 00 00 00 00 00	switch PSSu Outputs 0..7 off	00 00 00 00 00 00 00 00

Fig. 20: CAN-trace switch all Outputs on

- The 2 X 4 =8 Inputs of the two PSSu electronic modules needs one PDO-Byte only.

The Process Image of Inputs are optimised too:

	Time	ID	NameNodeTransfer data	
2	4.768875	18A	01 00 00 00 00 00 00 00	PSSu Slot 09: Input 0 Clamp 11 = 24V
3	0.756716	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 0 Clamp 11 = 00V
4	3.298970	18A	02 00 00 00 00 00 00 00	PSSu Slot 09: Input 1 Clamp 21 = 24V
5	0.686571	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 1 Clamp 21 = 00V
6	2.573039	18A	04 00 00 00 00 00 00 00	PSSu Slot 09: Input 2 Clamp 14 = 24V
7	0.902246	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 2 Clamp 14 = 00V
8	3.212377	18A	08 00 00 00 00 00 00 00	PSSu Slot 09: Input 3 Clamp 24 = 24V
9	0.463982	18A	00 00 00 00 00 00 00 00	PSSu Slot 09: Input 3 Clamp 24 = 00V
10	3.243607	18A	10 00 00 00 00 00 00 00	PSSu Slot 10: Input 4 Clamp 11 = 24V
11	0.577627	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 4 Clamp 11 = 00V
12	1.978989	18A	20 00 00 00 00 00 00 00	PSSu Slot 10: Input 5 Clamp 21 = 24V
13	0.393267	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 5 Clamp 21 = 00V
14	1.791362	18A	40 00 00 00 00 00 00 00	PSSu Slot 10: Input 6 Clamp 14 = 24V
15	0.450010	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 6 Clamp 14 = 00V
16	1.494184	18A	80 00 00 00 00 00 00 00	PSSu Slot 10: Input 7 Clamp 24 = 24V
17	0.497439	18A	00 00 00 00 00 00 00 00	PSSu Slot 10: Input 7 Clamp 24 = 00V

Fig. 21: CAN-trace of digital Inputs with optimised PII.

5.11. User defined ST Process Image for full PSSusystem

- This PSSu CAN system application uses the maximum of 240 digital Inputs.

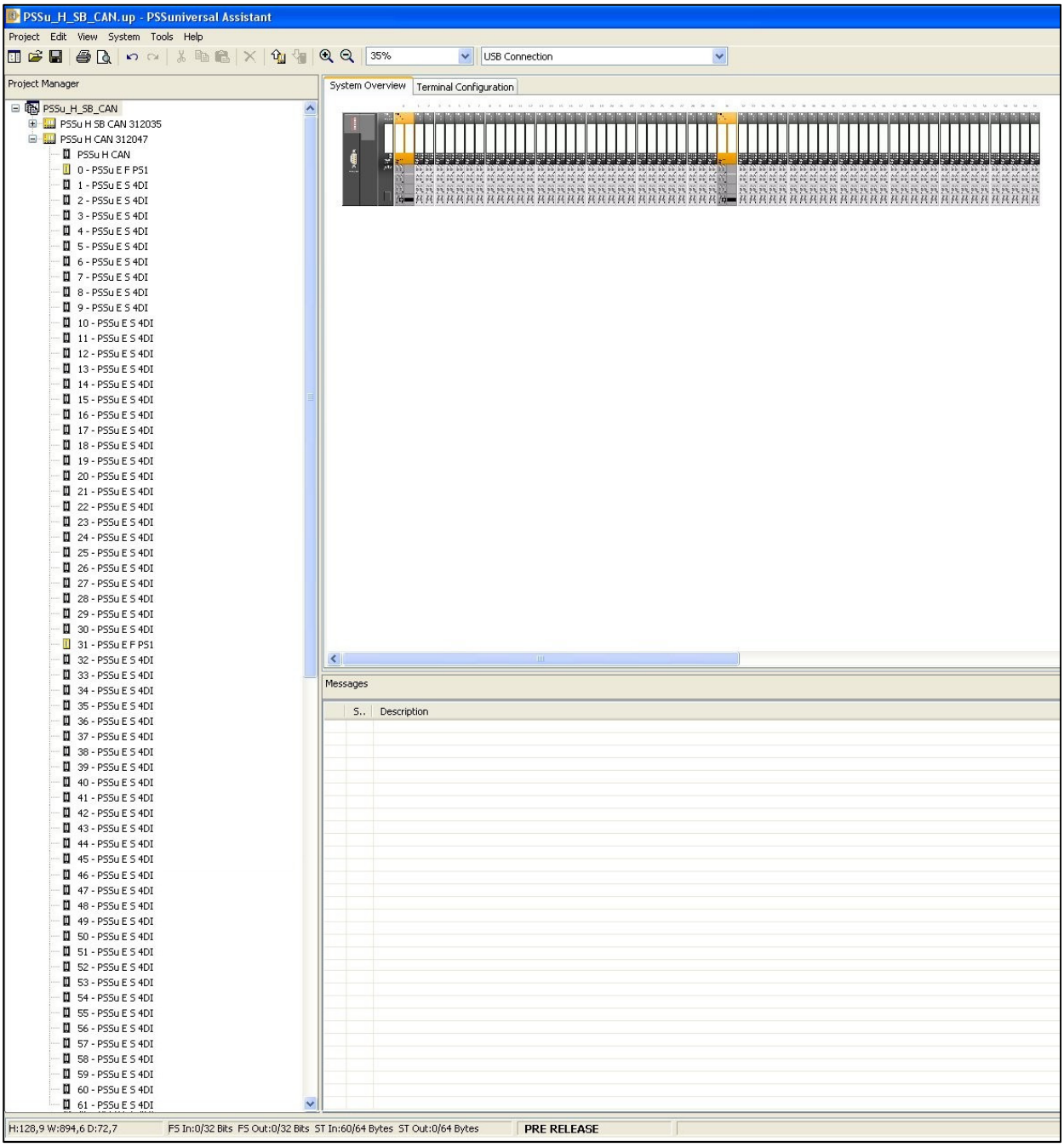


Fig. 22: 60 pieces of PSSu E S 4DI with 240 digital Inputs

There are 60 digital ST-Input slots. This means you need 60 :8 = 7,5 => 8 TPDOs without the Process Image of Inputs (PII) optimisation via Object 3001.

If you optimise the Process Image of Inputs you would need 4 TPDOs only.

- Note: There PS1 is placed on the right slot, because it is placed after a PII-Byte optimisation is finished.

The CANopen Identifier of the TPDOs is new. The Transmission types and inhibit times are unchanged. Each second PSSu E S 4 DI is * configured in the Object 3001 directory.

- Configuration:

PSSu Device Hardware Registry									
			Object	Sub-index		decimal	hexadecimal		
Slot	00	03 : PSSu E F PS1	[3001,	1	]:=	03	3		
Slot	01	09 : PSSu E S 4DI	[3001,	2	]:=	09	9	TPDO1 COB-ID 181 Byte 0 Low Byte	
Slot	02	10 : PSSu E S 4DI*	[3001,	3	]:=	10	A	TPDO1 COB-ID 181 Byte 0 High Byte	Optimised PII Object 3001 with SDO-download1
Slot	03	09 : PSSu E S 4DI	[3001,	4	]:=	09	9	TPDO1 COB-ID 181 Byte 1 Low Byte	
Slot	04	10 : PSSu E S 4DI*	[3001,	5	]:=	10	A	TPDO1 COB-ID 181 Byte 1 High Byte	Optimised PII Object 3001 with SDO-download2
Slot	05	09 : PSSu E S 4DI	[3001,	6	]:=	09	9	TPDO1 COB-ID 181 Byte 2 Low Byte	
Slot	06	10 : PSSu E S 4DI*	[3001,	7	]:=	10	A	TPDO1 COB-ID 181 Byte 2 High Byte	Optimised PII Object 3001 with SDO-download3
Slot	07	09 : PSSu E S 4DI	[3001,	8	]:=	09	9	TPDO1 COB-ID 181 Byte 3 Low Byte	
Slot	08	10 : PSSu E S 4DI*	[3001,	9	]:=	10	A	TPDO1 COB-ID 181 Byte 3 High Byte	Optimised PII Object 3001 with SDO-download4
Slot	09	09 : PSSu E S 4DI	[3001,	A	]:=	09	9	TPDO1 COB-ID 181 Byte 4 Low Byte	
Slot	10	10 : PSSu E S 4DI*	[3001,	B	]:=	10	A	TPDO1 COB-ID 181 Byte 4 High Byte	Optimised PII Object 3001 with SDO-download5
Slot	11	09 : PSSu E S 4DI	[3001,	C	]:=	09	9	TPDO1 COB-ID 181 Byte 5 Low Byte	
Slot	12	10 : PSSu E S 4DI*	[3001,	D	]:=	10	A	TPDO1 COB-ID 181 Byte 5 High Byte	Optimised PII Object 3001 with SDO-download6
Slot	13	09 : PSSu E S 4DI	[3001,	E	]:=	09	9	TPDO1 COB-ID 181 Byte 6 Low Byte	
Slot	14	10 : PSSu E S 4DI*	[3001,	F	]:=	10	A	TPDO1 COB-ID 181 Byte 6 High Byte	Optimised PII Object 3001 with SDO-download7
Slot	15	09 : PSSu E S 4DI	[3001,	10	]:=	09	9	TPDO1 COB-ID 181 Byte 7 Low Byte	
Slot	16	10 : PSSu E S 4DI*	[3001,	11	]:=	10	A	TPDO1 COB-ID 181 Byte 7 High Byte	Optimised PII Object 3001 with SDO-download8
Slot	17	09 : PSSu E S 4DI	[3001,	12	]:=	09	9	TPDO2 COB-ID 182 Byte 0 Low Byte	
Slot	18	10 : PSSu E S 4DI*	[3001,	13	]:=	10	A	TPDO2 COB-ID 182 Byte 0 High Byte	Optimised PII Object 3001 with SDO-download9
Slot	19	09 : PSSu E S 4DI	[3001,	14	]:=	09	9	TPDO2 COB-ID 182 Byte 1 Low Byte	
Slot	20	10 : PSSu E S 4DI*	[3001,	15	]:=	10	A	TPDO2 COB-ID 182 Byte 1 High Byte	Optimised PII Object 3001 with SDO-download10
Slot	21	09 : PSSu E S 4DI	[3001,	16	]:=	09	9	TPDO2 COB-ID 182 Byte 2 Low Byte	
Slot	22	10 : PSSu E S 4DI*	[3001,	17	]:=	10	A	TPDO2 COB-ID 182 Byte 2 High Byte	Optimised PII Object 3001 with SDO-download11
Slot	23	09 : PSSu E S 4DI	[3001,	18	]:=	09	9	TPDO2 COB-ID 182 Byte 3 Low Byte	
Slot	24	10 : PSSu E S 4DI*	[3001,	19	]:=	10	A	TPDO2 COB-ID 182 Byte 3 High Byte	Optimised PII Object 3001 with SDO-download12
Slot	25	09 : PSSu E S 4DI	[3001,	1A	]:=	09	9	TPDO2 COB-ID 182 Byte 4 Low Byte	
Slot	26	10 : PSSu E S 4DI*	[3001,	1B	]:=	10	A	TPDO2 COB-ID 182 Byte 4 High Byte	Optimised PII Object 3001 with SDO-download13
Slot	27	09 : PSSu E S 4DI	[3001,	1C	]:=	09	9	TPDO2 COB-ID 182 Byte 5 Low Byte	
Slot	28	10 : PSSu E S 4DI*	[3001,	1D	]:=	10	A	TPDO2 COB-ID 182 Byte 5 High Byte	Optimised PII Object 3001 with SDO-download14
Slot	29	09 : PSSu E S 4DI	[3001,	1E	]:=	09	9	TPDO2 COB-ID 182 Byte 6 Low Byte	
Slot	30	10 : PSSu E S 4DI*	[3001,	1F	]:=	10	A	TPDO2 COB-ID 182 Byte 6 High Byte	Optimised PII Object 3001 with SDO-download15
Slot	31	03 : PSSu E F PS1	[3001,	20	]:=	03	3		
Slot	32	09 : PSSu E S 4DI	[3001,	21	]:=	09	9	TPDO2 COB-ID 182 Byte 7 Low Byte	
Slot	33	10 : PSSu E S 4DI*	[3001,	22	]:=	10	A	TPDO2 COB-ID 182 Byte 7 High Byte	Optimised PII Object 3001 with SDO-download16
Slot	34	09 : PSSu E S 4DI	[3001,	23	]:=	09	9	TPDO3 COB-ID 183 Byte 0 Low Byte	
Slot	35	10 : PSSu E S 4DI*	[3001,	24	]:=	10	A	TPDO3 COB-ID 183 Byte 0 High Byte	Optimised PII Object 3001 with SDO-download17
Slot	36	09 : PSSu E S 4DI	[3001,	25	]:=	09	9	TPDO3 COB-ID 183 Byte 1 Low Byte	
Slot	37	10 : PSSu E S 4DI*	[3001,	26	]:=	10	A	TPDO3 COB-ID 183 Byte 1 High Byte	Optimised PII Object 3001 with SDO-download18
Slot	38	09 : PSSu E S 4DI	[3001,	27	]:=	09	9	TPDO3 COB-ID 183 Byte 2 Low Byte	
Slot	39	10 : PSSu E S 4DI*	[3001,	28	]:=	10	A	TPDO3 COB-ID 183 Byte 2 High Byte	Optimised PII Object 3001 with SDO-download19
Slot	40	09 : PSSu E S 4DI	[3001,	29	]:=	09	9	TPDO3 COB-ID 183 Byte 3 Low Byte	
Slot	41	10 : PSSu E S 4DI*	[3001,	2A	]:=	10	A	TPDO3 COB-ID 183 Byte 3 High Byte	Optimised PII Object 3001 with SDO-download20
Slot	42	09 : PSSu E S 4DI	[3001,	2B	]:=	09	9	TPDO3 COB-ID 183 Byte 4 Low Byte	
Slot	43	10 : PSSu E S 4DI*	[3001,	2C	]:=	10	A	TPDO3 COB-ID 183 Byte 4 High Byte	Optimised PII Object 3001 with SDO-download21
Slot	44	09 : PSSu E S 4DI	[3001,	2D	]:=	09	9	TPDO3 COB-ID 183 Byte 5 Low Byte	
Slot	45	10 : PSSu E S 4DI*	[3001,	2E	]:=	10	A	TPDO3 COB-ID 183 Byte 5 High Byte	Optimised PII Object 3001 with SDO-download22
Slot	46	09 : PSSu E S 4DI	[3001,	2F	]:=	09	9	TPDO3 COB-ID 183 Byte 6 Low Byte	
Slot	47	10 : PSSu E S 4DI*	[3001,	30	]:=	10	A	TPDO3 COB-ID 183 Byte 6 High Byte	Optimised PII Object 3001 with SDO-download23
Slot	48	09 : PSSu E S 4DI	[3001,	31	]:=	09	9	TPDO3 COB-ID 183 Byte 7 Low Byte	
Slot	49	10 : PSSu E S 4DI*	[3001,	32	]:=	10	A	TPDO3 COB-ID 183 Byte 7 High Byte	Optimised PII Object 3001 with SDO-download24
Slot	50	09 : PSSu E S 4DI	[3001,	33	]:=	09	9	TPDO4 COB-ID 184 Byte 0 Low Byte	
Slot	51	10 : PSSu E S 4DI*	[3001,	34	]:=	10	A	TPDO4 COB-ID 184 Byte 0 High Byte	Optimised PII Object 3001 with SDO-download25
Slot	52	09 : PSSu E S 4DI	[3001,	35	]:=	09	9	TPDO4 COB-ID 184 Byte 1 Low Byte	
Slot	53	10 : PSSu E S 4DI*	[3001,	36	]:=	10	A	TPDO4 COB-ID 184 Byte 1 High Byte	Optimised PII Object 3001 with SDO-download26
Slot	54	09 : PSSu E S 4DI	[3001,	37	]:=	09	9	TPDO4 COB-ID 184 Byte 2 Low Byte	
Slot	55	10 : PSSu E S 4DI*	[3001,	38	]:=	10	A	TPDO4 COB-ID 184 Byte 2 High Byte	Optimised PII Object 3001 with SDO-download27
Slot	56	09 : PSSu E S 4DI	[3001,	39	]:=	09	9	TPDO4 COB-ID 184 Byte 3 Low Byte	
Slot	57	10 : PSSu E S 4DI*	[3001,	3A	]:=	10	A	TPDO4 COB-ID 184 Byte 3 High Byte	Optimised PII Object 3001 with SDO-download28
Slot	58	09 : PSSu E S 4DI	[3001,	3B	]:=	09	9	TPDO4 COB-ID 184 Byte 4 Low Byte	
Slot	59	10 : PSSu E S 4DI*	[3001,	3C	]:=	10	A	TPDO4 COB-ID 184 Byte 4 High Byte	Optimised PII Object 3001 with SDO-download29
Slot	60	09 : PSSu E S 4DI	[3001,	3D	]:=	09	9	TPDO4 COB-ID 184 Byte 5 Low Byte	
Slot	61	10 : PSSu E S 4DI*	[3001,	3E	]:=	10	A	TPDO4 COB-ID 184 Byte 5 High Byte	Optimised PII Object 3001 with SDO-download30

Fig. 23: PSSu configuration; 3001-Object directory and COB-IDs

- Trace:

```
60A    8a 01 00 80                [1800,01] Initiate Download Rq. expedited          23 00 18 01 8A 01 00 80
58A                               [1800,01] Initiate Download Rsp                    60 00 18 01 00 00 00 00
60A    00                         [1a00,00] Initiate Download Rq. expedited          2F 00 1A 00 00 00 00 00
58A                               [1a00,00] Initiate Download Rsp                    60 00 1A 00 00 00 00 00
60A    08 01 00 60 [6000,01]      [1a00,01] Initiate Download Rq. expedited "...`"   23 00 1A 01 08 01 00 60
58A                               [1a00,01] Initiate Download Rsp                    60 00 1A 01 00 00 00 00
60A    08 02 00 60 [6000,02]      [1a00,02] Initiate Download Rq. expedited "...`"   23 00 1A 02 08 02 00 60
58A                               [1a00,02] Initiate Download Rsp                    60 00 1A 02 00 00 00 00
60A    08 03 00 60 [6000,03]      [1a00,03] Initiate Download Rq. expedited "...`"   23 00 1A 03 08 03 00 60
58A                               [1a00,03] Initiate Download Rsp                    60 00 1A 03 00 00 00 00
60A    08 04 00 60 [6000,04]      [1a00,04] Initiate Download Rq. expedited "...`"   23 00 1A 04 08 04 00 60
58A                               [1a00,04] Initiate Download Rsp                    60 00 1A 04 00 00 00 00
60A    08 05 00 60 [6000,05]      [1a00,05] Initiate Download Rq. expedited "...`"   23 00 1A 05 08 05 00 60
58A                               [1a00,05] Initiate Download Rsp                    60 00 1A 05 00 00 00 00
60A    08 06 00 60 [6000,06]      [1a00,06] Initiate Download Rq. expedited "...`"   23 00 1A 06 08 06 00 60
58A                               [1a00,06] Initiate Download Rsp                    60 00 1A 06 00 00 00 00
60A    08 07 00 60 [6000,07]      [1a00,07] Initiate Download Rq. expedited "...`"   23 00 1A 07 08 07 00 60
58A                               [1a00,07] Initiate Download Rsp                    60 00 1A 07 00 00 00 00
60A    08 08 00 60 [6000,08]      [1a00,08] Initiate Download Rq. expedited "...`"   23 00 1A 08 08 08 00 60
58A                               [1a00,08] Initiate Download Rsp                    60 00 1A 08 00 00 00 00
60A    08                         [1a00,00] Initiate Download Rq. expedited          2F 00 1A 00 08 00 00 00
58A                               [1a00,00] Initiate Download Rsp                    60 00 1A 00 00 00 00 00
60A    ff                         [1800,02] Initiate Download Rq. expedited          2F 00 18 02 FF 00 00 00
58A                               [1800,02] Initiate Download Rsp                    60 00 18 02 00 00 00 00
60A    00 00                      [1800,03] Initiate Download Rq. expedited          2B 00 18 03 00 00 00 00
58A                               [1800,03] Initiate Download Rsp                    60 00 18 03 00 00 00 00
60A    81 01 00 00                [1800,01] Initiate Download Rq. expedited          23 00 18 01 81 01 00 00
58A                               [1800,01] Initiate Download Rsp                    60 00 18 01 00 00 00 00
60A    0a 00                      [3001,03] Initiate Download Rq. expedited          2B 01 30 03 0A 00 00 00
58A                               [3001,03] Initiate Download Rsp                    60 01 30 03 00 00 00 00
60A    0a 00                      [3001,05] Initiate Download Rq. expedited          2B 01 30 05 0A 00 00 00
58A                               [3001,05] Initiate Download Rsp                    60 01 30 05 00 00 00 00
60A    0a 00                      [3001,07] Initiate Download Rq. expedited          2B 01 30 07 0A 00 00 00
58A                               [3001,07] Initiate Download Rsp                    60 01 30 07 00 00 00 00
60A    0a 00                      [3001,09] Initiate Download Rq. expedited          2B 01 30 09 0A 00 00 00
58A                               [3001,09] Initiate Download Rsp                    60 01 30 09 00 00 00 00
60A    0a 00                      [3001,0b] Initiate Download Rq. expedited          2B 01 30 0B 0A 00 00 00
58A                               [3001,0b] Initiate Download Rsp                    60 01 30 0B 00 00 00 00
60A    0a 00                      [3001,0d] Initiate Download Rq. expedited          2B 01 30 0D 0A 00 00 00
58A                               [3001,0d] Initiate Download Rsp                    60 01 30 0D 00 00 00 00
60A    0a 00                      [3001,0f] Initiate Download Rq. expedited          2B 01 30 0F 0A 00 00 00
58A                               [3001,0f] Initiate Download Rsp                    60 01 30 0F 00 00 00 00
60A    0a 00                      [3001,11] Initiate Download Rq. expedited          2B 01 30 11 0A 00 00 00
58A                               [3001,11] Initiate Download Rsp                    60 01 30 11 00 00 00 00
60A    ca 01 00 80                [1801,01] Initiate Download Rq. expedited          23 01 18 01 CA 01 00 80
```

Fig. 24: This is the trace of mapping TPDO1

In this case there are 6 steps to PDO mapping implemented by SDO-Downloads from the CAN Master to the PSSu object directory:

- 1. Delete PDO identifier (COB-ID, Bit 31) in the PDO communication parameter (line 92).
- 2. Set mapped objects in the PDO mapping parameter to zero.(line 94)
- 3. Map the required objects to the PDO. (lines 96 to 110)
- 4. Enter the sum total of mapped objects in the PDO mapping parameter (line 112)

- 5. Activate PDO identifier (COB-ID, Bit 31) in the PDO communication (118) parameter.

- 6. Change the hardware registry of the object 3001 configuration of each second to PSSu S T DI4*(line 120 to 134)

  Line 114: Transmission type = 255 = event driven
  Line 116: PDO Inhibit Time = 0 = off

5.12. Optimised ST Process Image, Read FS access from field bus side

- Used PSSu configuration:

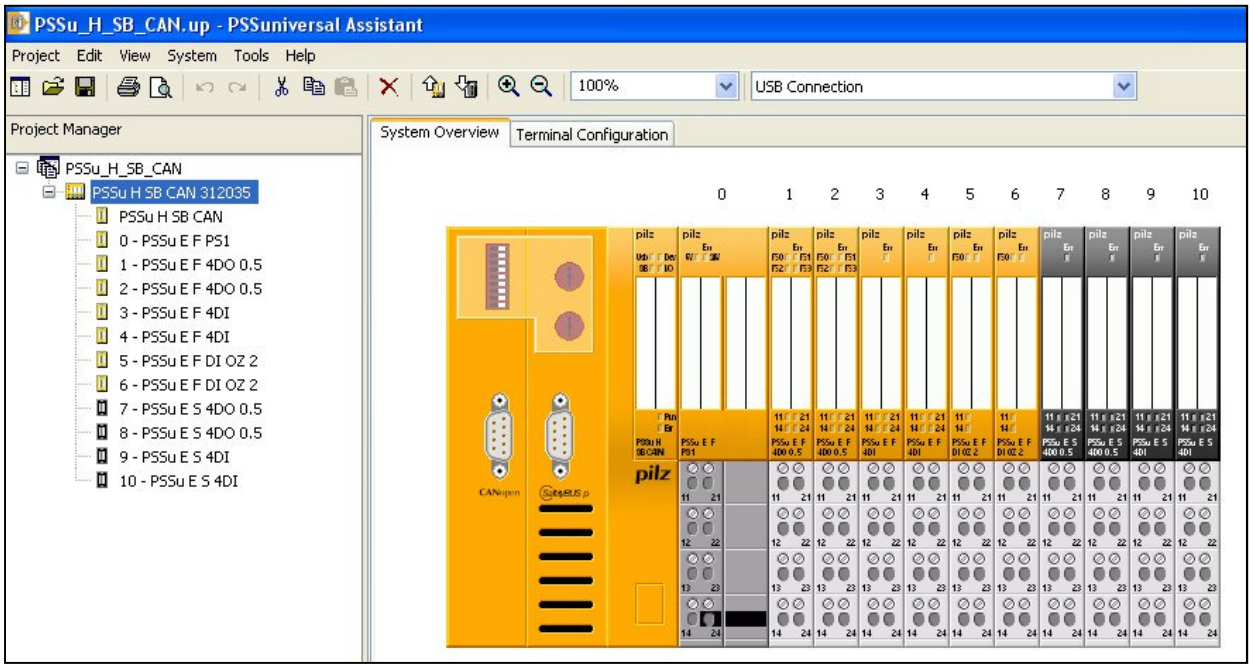


Fig. 25: PSSu Hardware Registry

- Start Configuration of Process image:

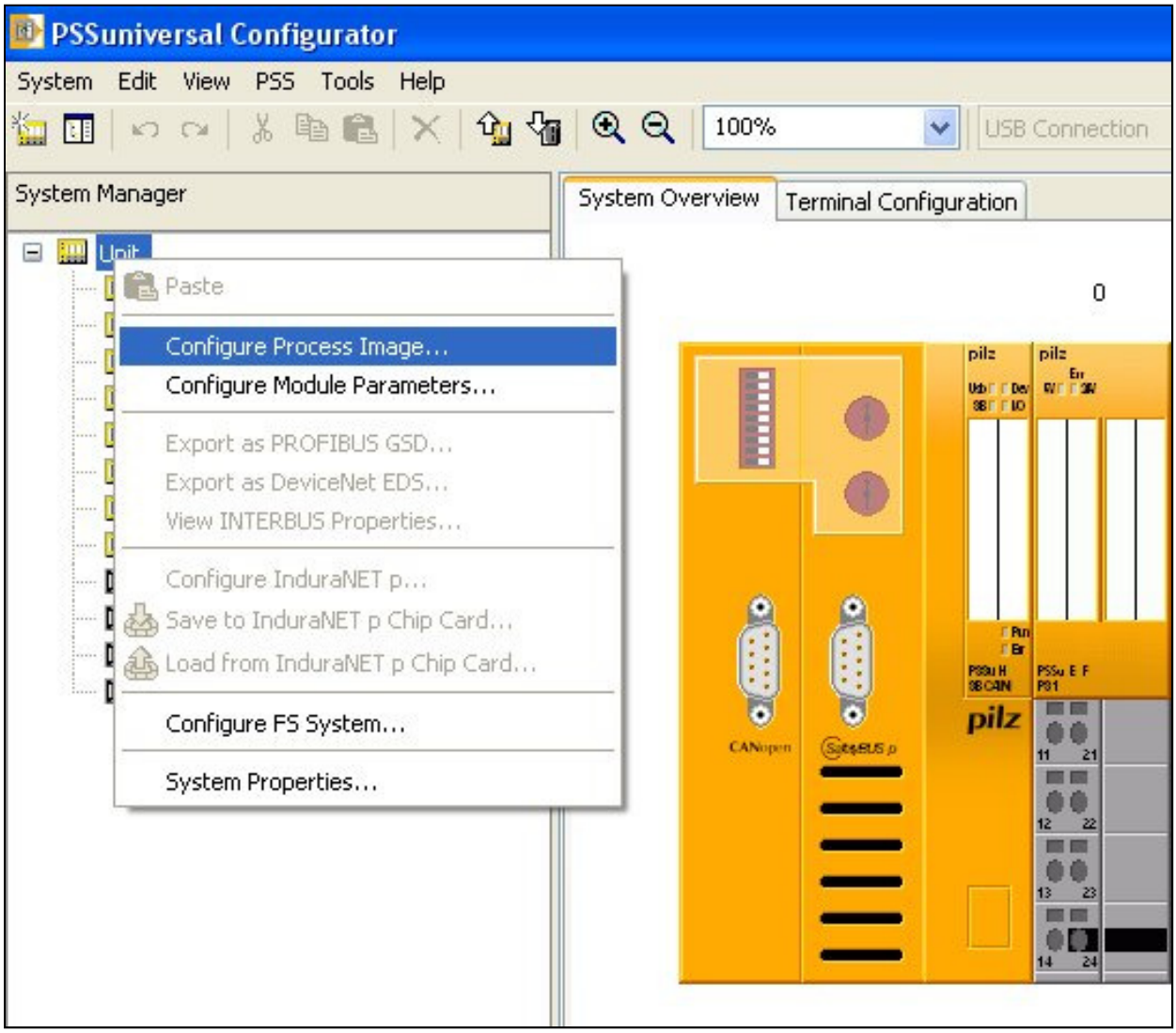


Fig. 26: Start of ST Process Image Configuration to get the Process Image View

- Set parameters of Process image:

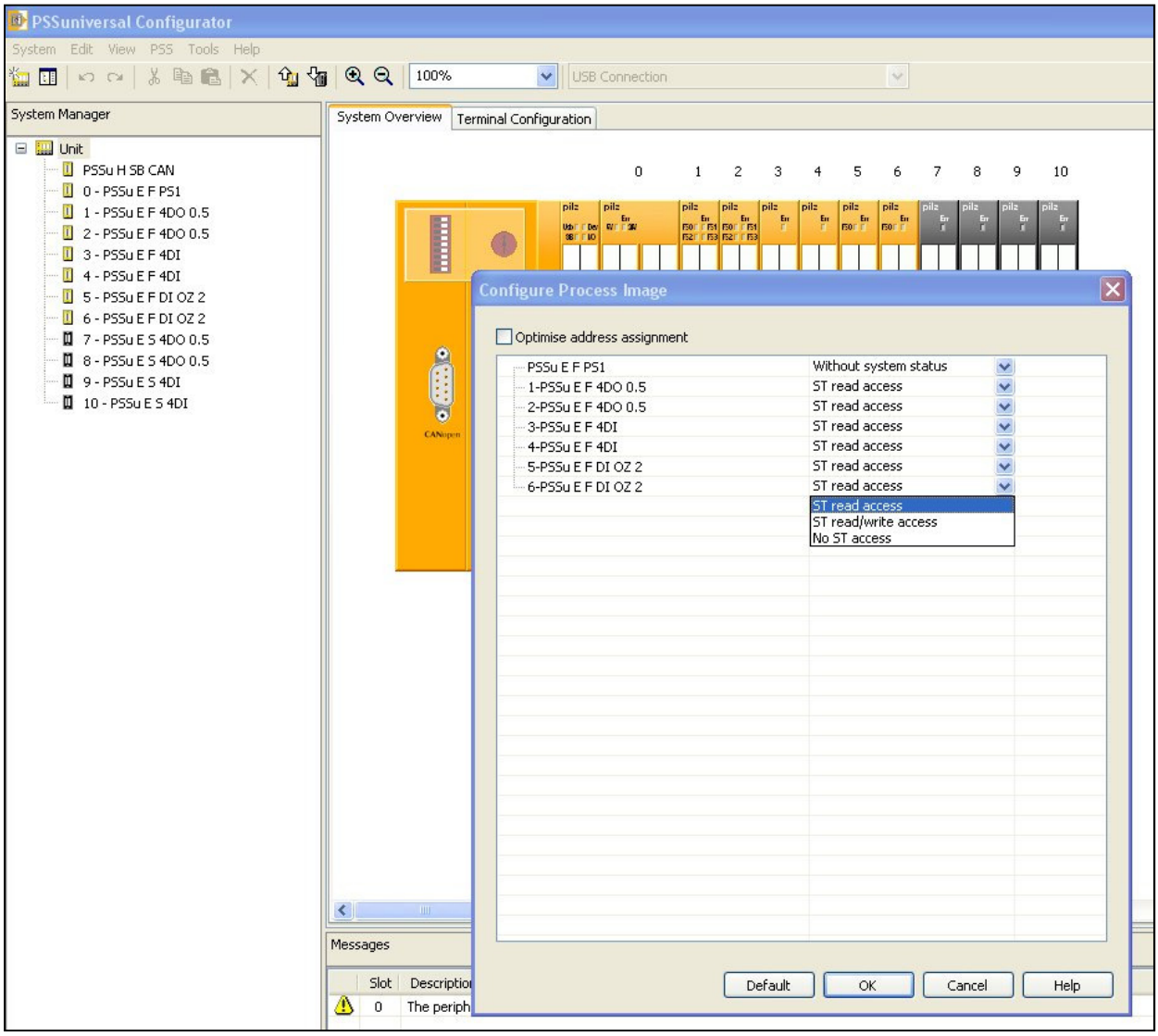


Fig. 27: Each PSSu FS Electronic module I/O status should be read from the CANopen field bus

- Note:
  The Optimise address assignment option in the PSSu Configurator is for generating a Profibus GSD- and DeviceNet ESD-file generation only. There is no CANopen configuration tool implemented.

5.12.1. PDO-Mapping and Hardware Configuration

```
79  |  0.086828   1  Rx  60A   8a 01 00 80                [1800,01] Initiate Download Rq. expedited
80  |  0.000763   1  Rx  58A                             [1800,01] Initiate Download Rsp
81  |  0.001405   1  Rx  60A   00                        [1a00,00] Initiate Download Rq. expedited
82  |  0.005725   1  Rx  58A                             [1a00,00] Initiate Download Rsp
83  |  0.000401   1  Rx  60A   08 01 00 20 [2000,01]     [1a00,01] Initiate Download Rq. expedited "... "
84  |  0.000305   1  Rx  58A                             [1a00,01] Initiate Download Rsp
85  |  0.000411   1  Rx  60A   08 02 00 20 [2000,02]     [1a00,02] Initiate Download Rq. expedited "... "
86  |  0.000647   1  Rx  58A                             [1a00,02] Initiate Download Rsp
87  |  0.000410   1  Rx  60A   08 03 00 20 [2000,03]     [1a00,03] Initiate Download Rq. expedited "... "
88  |  0.000783   1  Rx  58A                             [1a00,03] Initiate Download Rsp
89  |  0.000438   1  Rx  60A   08 01 00 60 [6000,01]     [1a00,04] Initiate Download Rq. expedited "...`"
90  |  0.000776   1  Rx  58A                             [1a00,04] Initiate Download Rsp
91  |  0.000393   1  Rx  60A   04                        [1a00,00] Initiate Download Rq. expedited
92  |  0.002804   1  Rx  58A                             [1a00,00] Initiate Download Rsp
93  |  0.000449   1  Rx  60A   ff                        [1800,02] Initiate Download Rq. expedited
94  |  0.000316   1  Rx  58A                             [1800,02] Initiate Download Rsp
95  |  0.000401   1  Rx  60A   00 00                     [1800,03] Initiate Download Rq. expedited
96  |  0.000596   1  Rx  58A                             [1800,03] Initiate Download Rsp
97  |  0.000402   1  Rx  60A   81 01 00 00               [1800,01] Initiate Download Rq. expedited
98  |  0.000718   1  Rx  58A                             [1800,01] Initiate Download Rsp
99  |  0.000440   1  Rx  600   40 0a 14 01 6a 04 00 80
100 |  0.577969   1  Rx  701   T0 Operational
101 |  0.421182   1  Rx  60A   6a 05 00 80                [140b,01] Initiate Download Rq. expedited "j..€"
102 |  0.000471   1  Rx  58A                             [140b,01] Initiate Download Rsp
103 |  0.000390   1  Rx  600   40 00 00 00 00 00 00 00
104 |  0.577937   1  Rx  701   T0 Operational
105 |  0.421200   1  Rx  600   40 00 00 00 00 00 00 00
```

```
106 |  0.578781   1  Rx  701   T0 Operational
107 |  0.421180   1  Rx  60A   13 00                     [3001,02] Initiate Download Rq. expedited
108 |  0.000763   1  Rx  58A                             [3001,02] Initiate Download Rsp
109 |  0.000391   1  Rx  60A   14 00                     [3001,03] Initiate Download Rq. expedited
110 |  0.000735   1  Rx  58A                             [3001,03] Initiate Download Rsp
111 |  0.000434   1  Rx  60A   10 00                     [3001,04] Initiate Download Rq. expedited
112 |  0.000697   1  Rx  58A                             [3001,04] Initiate Download Rsp
113 |  0.000410   1  Rx  60A   11 00                     [3001,05] Initiate Download Rq. expedited
114 |  0.000722   1  Rx  58A                             [3001,05] Initiate Download Rsp
115 |  0.000389   1  Rx  60A   22 00                     [3001,06] Initiate Download Rq. expedited "."
116 |  0.000757   1  Rx  58A                             [3001,06] Initiate Download Rsp
117 |  0.000409   1  Rx  60A   23 00                     [3001,07] Initiate Download Rq. expedited "#."
118 |  0.000903   1  Rx  58A                             [3001,07] Initiate Download Rsp
119 |  0.000434   1  Rx  60A   0b 00                     [3001,08] Initiate Download Rq. expedited
120 |  0.000705   1  Rx  58A                             [3001,08] Initiate Download Rsp
121 |  0.000403   1  Rx  60A   0c 00                     [3001,09] Initiate Download Rq. expedited
122 |  0.000728   1  Rx  58A                             [3001,09] Initiate Download Rsp
123 |  0.000441   1  Rx  60A   09 00                     [3001,0a] Initiate Download Rq. expedited
124 |  0.000696   1  Rx  58A                             [3001,0a] Initiate Download Rsp
125 |  0.000412   1  Rx  60A   0a 00                     [3001,0b] Initiate Download Rq. expedited
126 |  0.000724   1  Rx  58A                             [3001,0b] Initiate Download Rsp
```

Fig. 28: PDO-mapping

Explanation:

- 1. Delete PDO identifier (COB-ID, Bit 31) in the TPDO1 communication parameter COB-ID 0x18A (line 79).
- 2. Set mapped objects in the PDO mapping parameter to zero.(line 81)
- 3. Map the required objects to the PDO. (lines 83 to 89):
  - [Object 2000 Subindex 1] FS-Input (Byte) 1, read PSSu E F 4DO, slot 1 & 2
  - [Object 2000 Subindex 2] FS-Input (Byte) 2, read PSSu E F 4DI, slot 3 & 4
  - [Object 2000 Subindex 3] FS-Input (Byte) 3, read PSSu E F DIOZ, slot 5 & 6
  - [Object 6000 Subindex 1] ST-Input (Byte) 1, read PSSu E S 4 DI, slot 9 & 10

- 4. Enter the sum total of mapped objects in the TPDO 1 mapping parameter = 4 (line 91)

- 5. Activate PDO identifier (COB-ID, Bit 31) in the TPDO1 communication parameter COB-ID 0x181 (line 97)

- 6. Change the hardware registry of the object 3001 configuration of each second to PSSu S T DI4*(line 107 to 125)

Line 93: Transmission type = 255 = “event driven”
Line 95: PDO Inhibit Time = 0 = off

5.12.2. The resulting Process Image of Inputs (PII)

- Optimize address assignment of Process image:

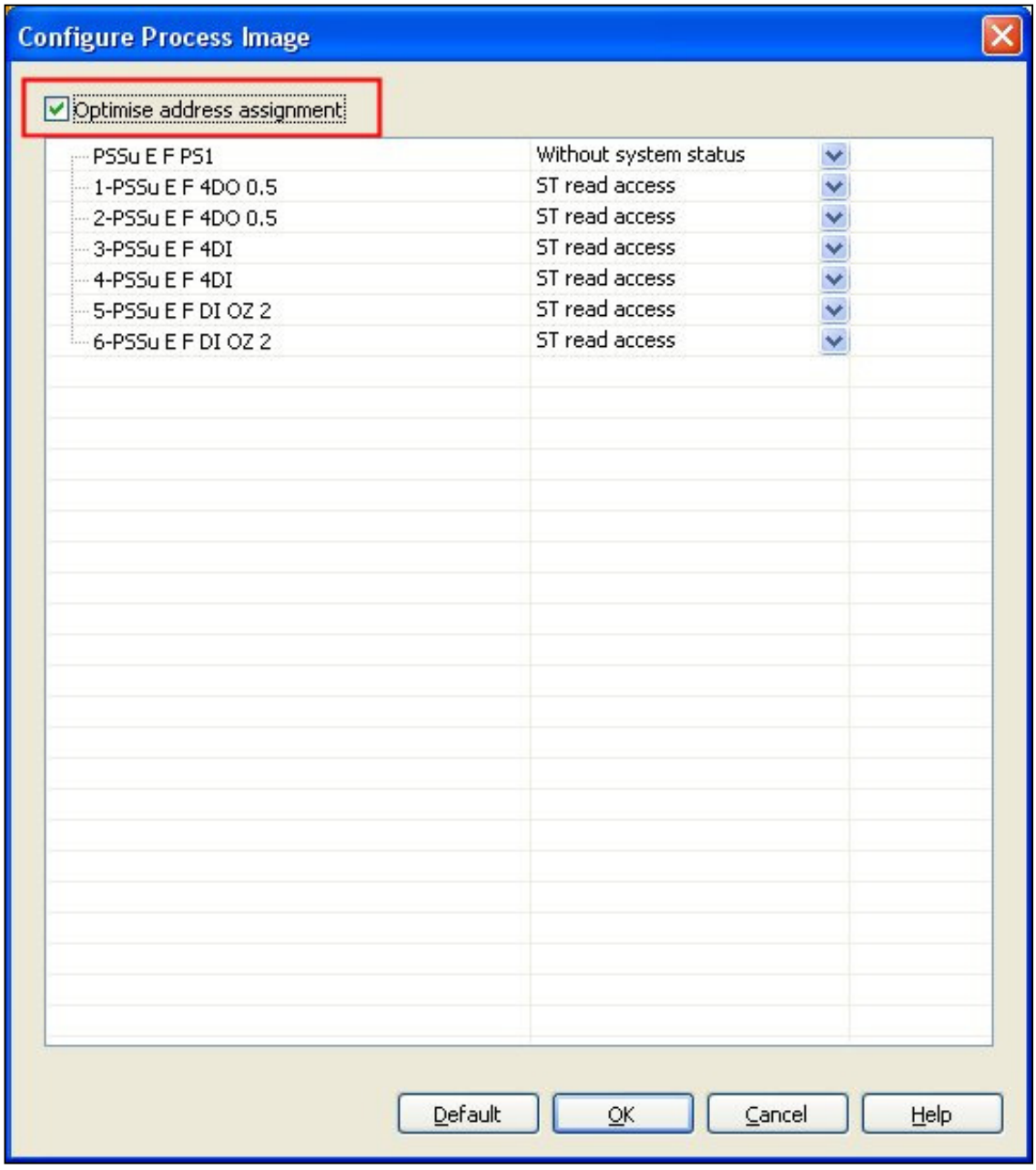


Fig. 29: Enable the option “Optimise address assignment”

- Start view of Process image:

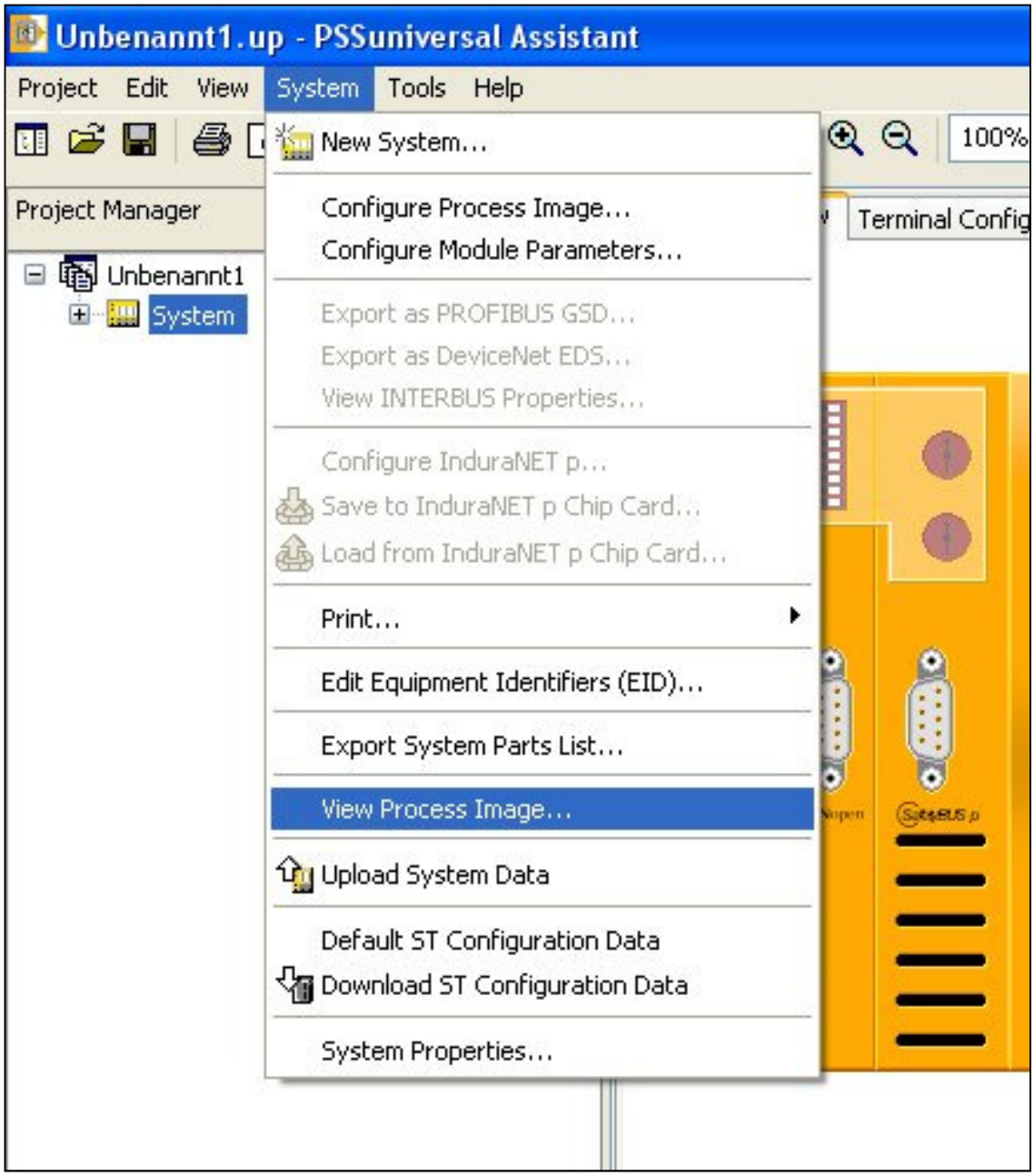


Fig. 30: View resulting Process Image

- Copy Process image:

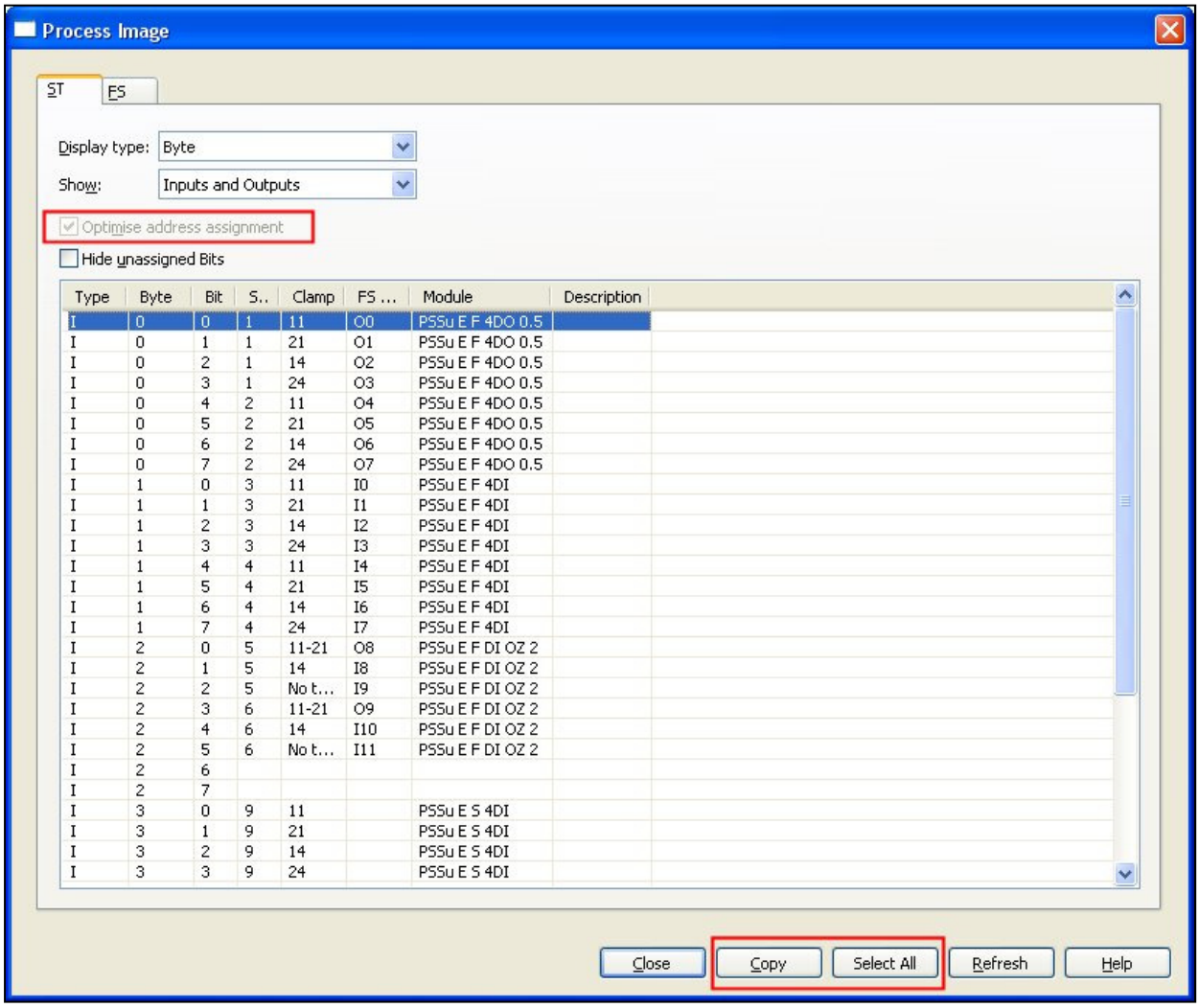


Fig. 31: Copying the resulting to the clipboard

- If you press the buttons “Select All” and “Copy” you list of the Process Image of Inputs and Outputs is copied to the Windows XP clipboard.

- You can use MS Excel to modify the list and add your comments.

Byte	Bit	Slot	Clamp	FS I/O	Electronic Module	CANopen			
0	0	1	11	A33.00	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 0	
0	1	1	21	A33.01	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 1	
0	2	1	14	A33.02	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 2	
0	3	1	24	A33.03	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 3	
0	4	2	11	A33.04	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 4	
0	5	2	21	A33.05	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 5	
0	6	2	14	A33.06	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 6	
0	7	2	24	A33.07	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 0	Bit 7	
1	0	3	11	E33.00	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 0	
1	1	3	21	E33.01	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 1	
1	2	3	14	E33.02	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 2	
1	3	3	24	E33.03	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 3	
1	4	4	11	E33.04	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 4	
1	5	4	21	E33.05	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 5	
1	6	4	14	E33.06	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 6	
1	7	4	24	E33.07	PSSu E F 4DI	COB-ID 0x181	Byte 1	Bit 7	
2	0	5	11-21	A33.08	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 0	
2	1	5	14	E33.08	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 1	
2	2	5	No terminal	E33.09	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 2	no wire break
2	3	6	11-21	A33.09	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 3	
2	4	6	14	E33.10	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 4	
2	5	6	No terminal	E33.11	PSSu E F DI OZ 2	COB-ID 0x181	Byte 2	Bit 5	no wire break
2	6								
2	7								
3	0	9	11		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 0	
3	1	9	21		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 1	
3	2	9	14		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 2	
3	3	9	24		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 3	
3	4	10	11		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 4	
3	5	10	21		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 5	
3	6	10	14		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 6	
3	7	10	24		PSSu E S 4DI	COB-ID 0x181	Byte 3	Bit 7	
0	0	7	11		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 0	
0	1	7	21		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 1	
0	2	7	14		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 2	
0	3	7	24		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 3	
0	4	8	11		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 4	
0	5	8	21		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 5	
0	6	8	14		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 6	
0	7	8	24		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 0	Bit 7	

Fig. 32: PDO-Mapped physical and non physical operands

- Note:
  The PSSu FS Outputs are still switched from the PSS side only.

5.13. Optimised ST Process Image, Read FS & Write access from field bus side

- There are two way to get the PSSu System these information's:
  - Unit_Diag_Bit(0) = "SBp is ready"
  - Unit_Diag_Bit(1) = "SBp I/O-Group of Inputs RUN"
  - Unit_Diag_Bit(2) = "SBp I/O-Group of Outputs RUN"
  - Unit_Diag_Bit(3) = "Device Status FS: Error"
  - Unit_Diag_Bit(4) = "Device Status ST: Error"
  - Unit_Diag_Bit(5) = "FS-Module Error"
  - Unit_Diag_Bit(6) = "ST-Module Error"
  - Unit_Diag_Bit(7) = "Reserved"

5.13.1. PSSu System information by reading the CAN Emergency telegram

This is the first way to get the PSSu-System information's.

The default Emergency CAN Identifier of PSSu node is 0x80 plus Node ID.
Node ID = 10 => Emergency CAN Identifier = 0x80 + 0x0A = 0x8A

- Example
  "8A 00 00 00 **07** 00 00 00 00" means:

  - Unit_Diag_Bit(0) = "SBp is ready"
    = RLO-1 => SBp is in RUN-mode<<<<

  - Unit_Diag_Bit(1) = "SBp I/O-Group of Inputs RUN"
    = RLO-1 => SBp I/O-Group of Inputs in RUN-mode

  - Unit_Diag_Bit(2) = "SBp I/O-Group of Outputs RUN"
    = RLO-1 => SBp I/O-Group of Outputs in RUN-mode

  - Unit_Diag_Bit(3) = "Device Status FS: Error"
    = RLO-0 => no Device Status FS: Error

  - Unit_Diag_Bit(4) = "Device Status ST: Error"
    = RLO-0 => no Device Status ST: Error

  - Unit_Diag_Bit(5) = "FS-Module Error"
    = RLO-0 => no FS-Module Error

  - Unit_Diag_Bit(6) = "ST-Module Error"
    = RLO-1 => no ST-Module Error

  - Unit_Diag_Bit(7) = "Reserved"
    = RLO-1 => default

5.13.2. PSSu System information by configuring the first PSSu E F PS(1) with “R”

- The second way to get the PSSu-System information is to configure the Object 3001 Subindex 1 (slot 0 only!) with:
  - PSSu E F PS1 R value = 2 or
  - PSSu E F PS R value = 4

You will get the PSSu System information via the ST Object 0x2F01 Subindex 1. The Object can be read by SDO upload or mapped TPDO. In the following example the PSSu System information is mapped to TPDO1.

5.13.3. PSS FS-Configuration of local enable:

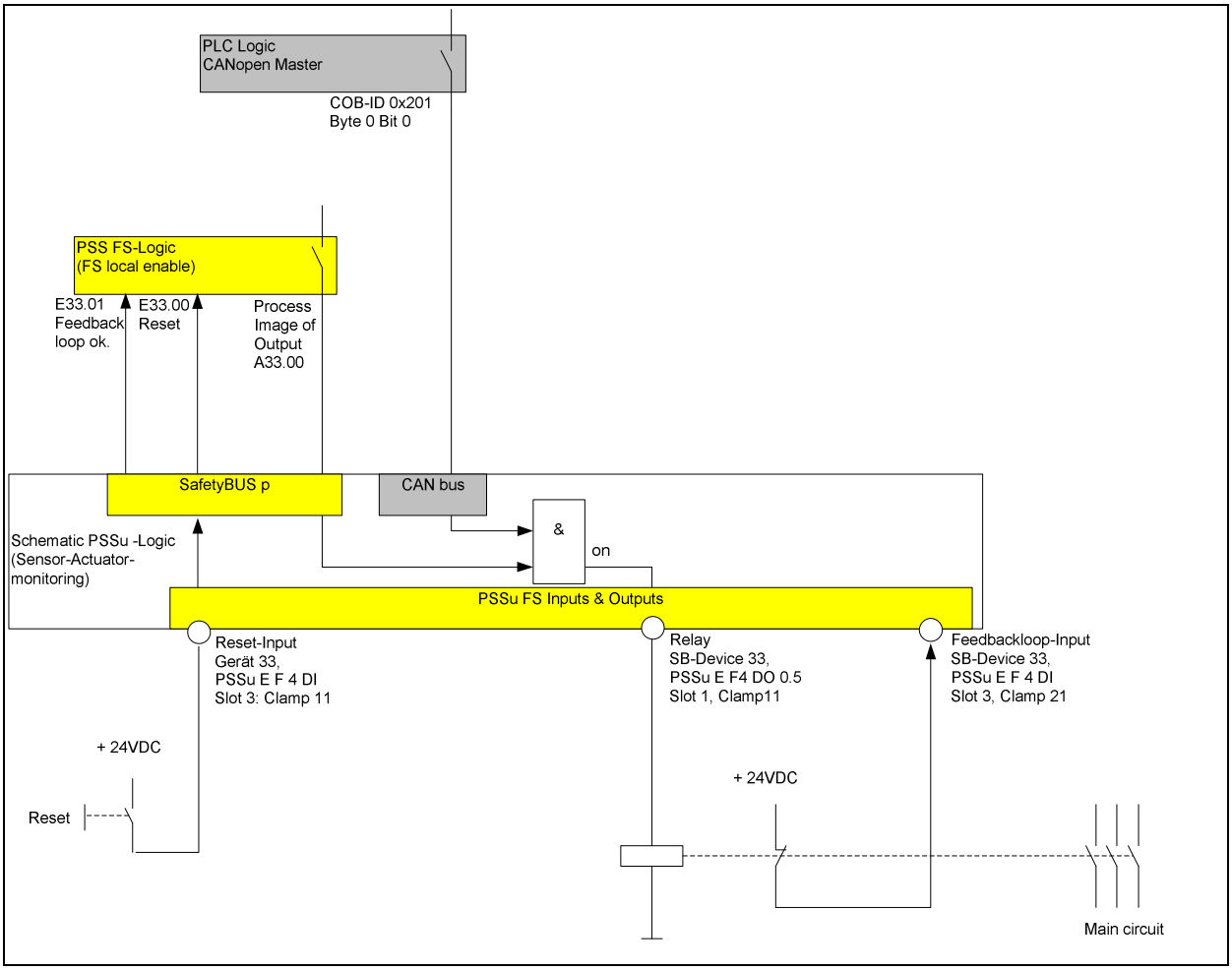


Fig. 33: Schematic view to a PSSu local enable with a singled channel feedback loop monitoring.

- Used PSSu configuration:

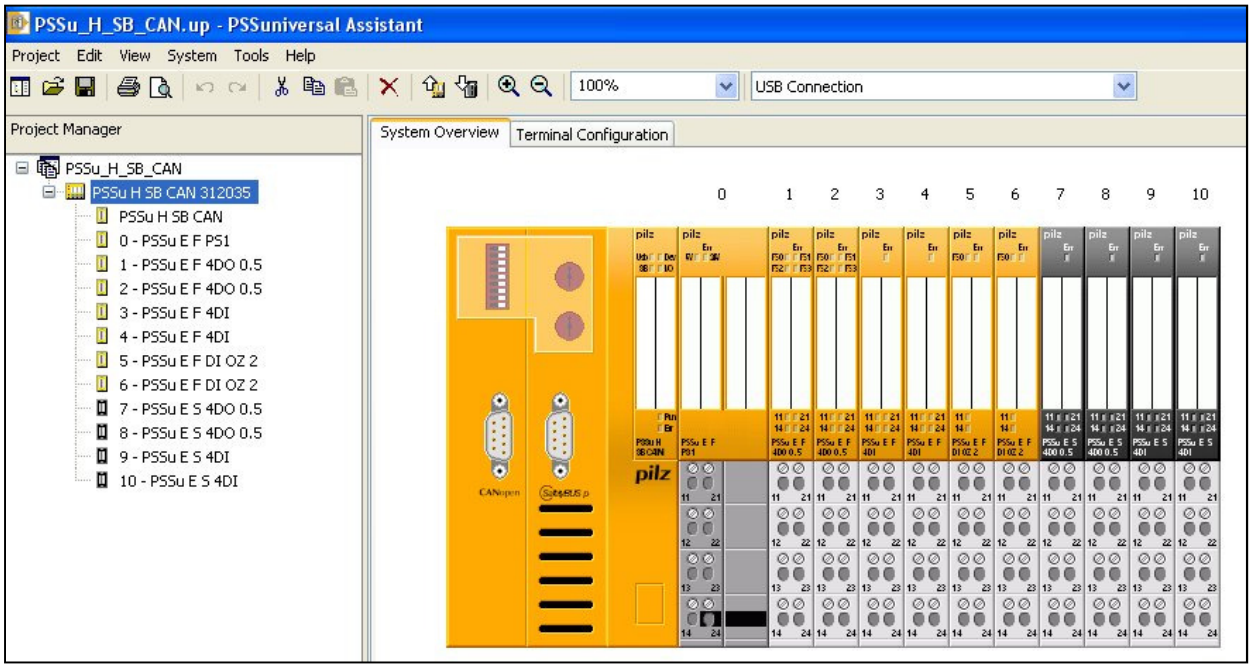


Fig. 34: PSSu Hardware Registry

- Start PSSu configuration:

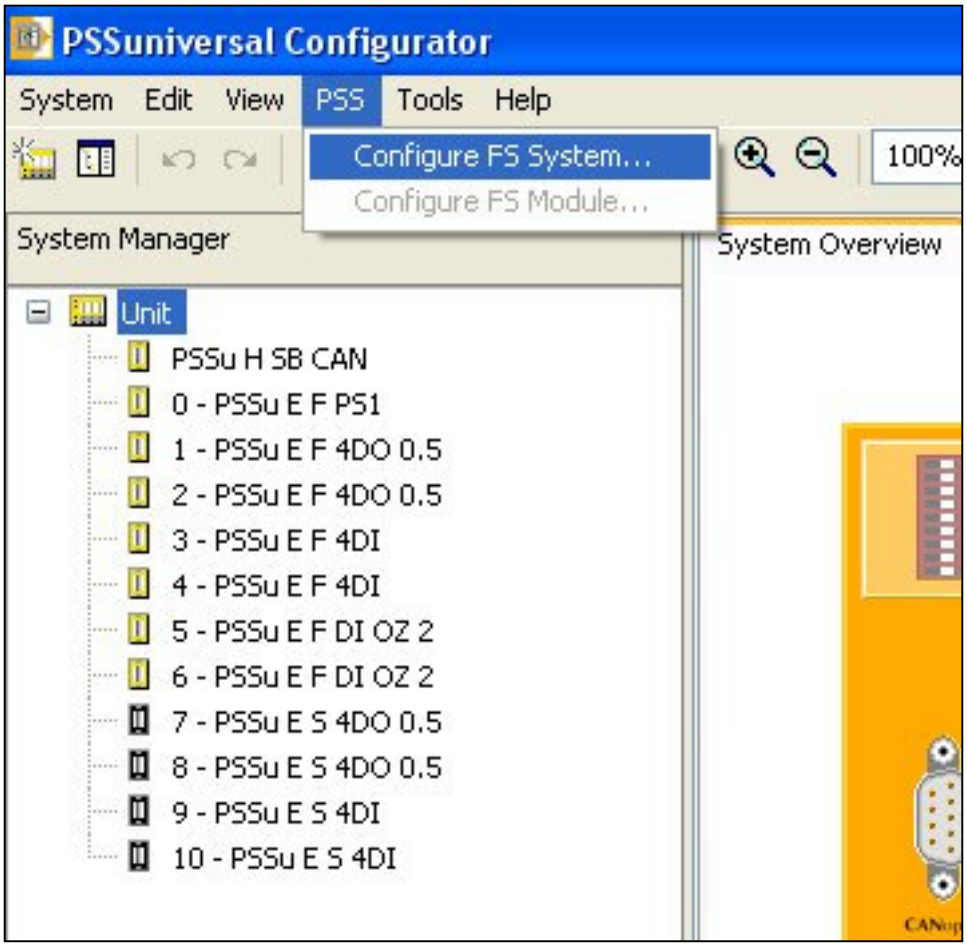


Fig. 35: PSSu FS configuration

- All PSSu FS Outputs uses the local enable:
  - Configuration of Slot 1

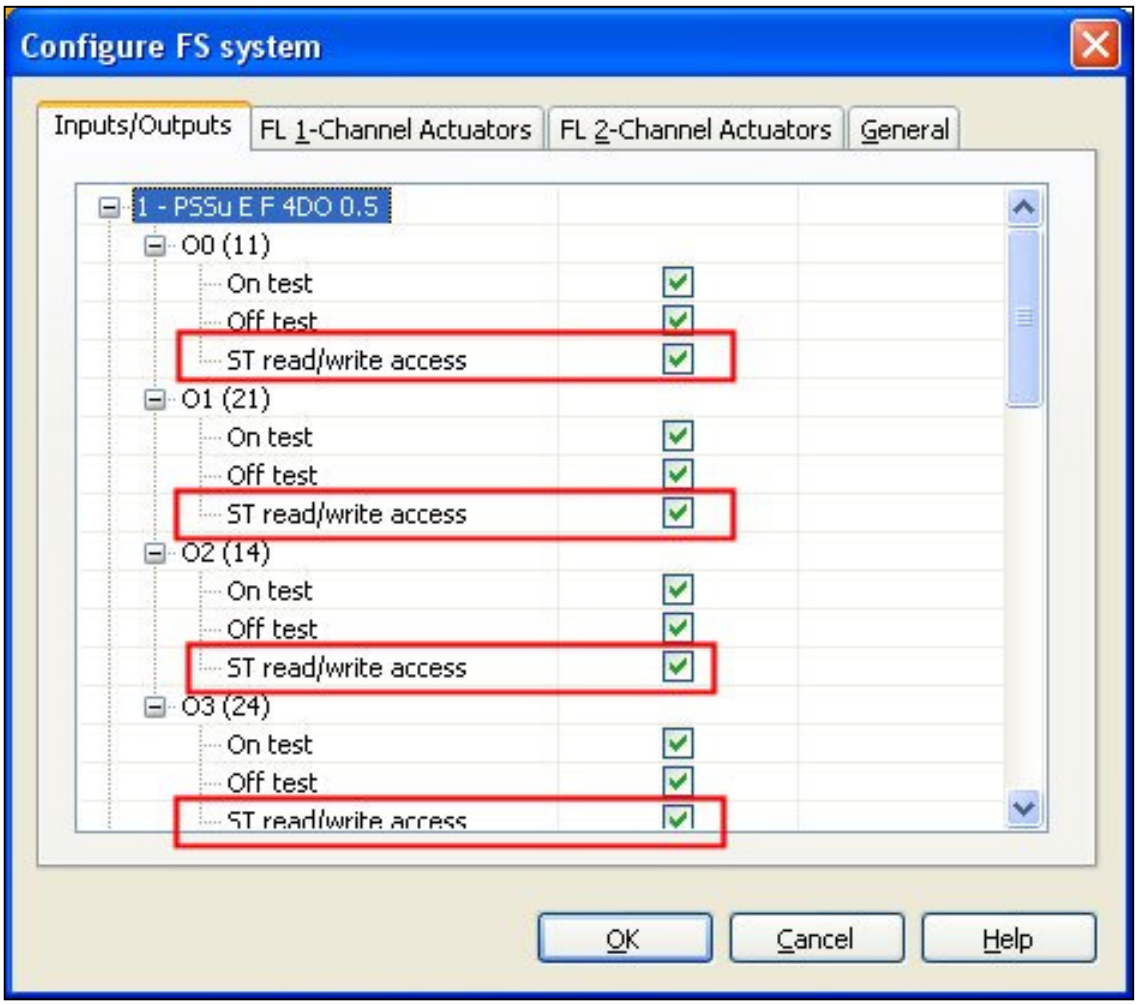


Fig. 36: PSSu local enabling of Slot 1

- Configuration of Slot 2

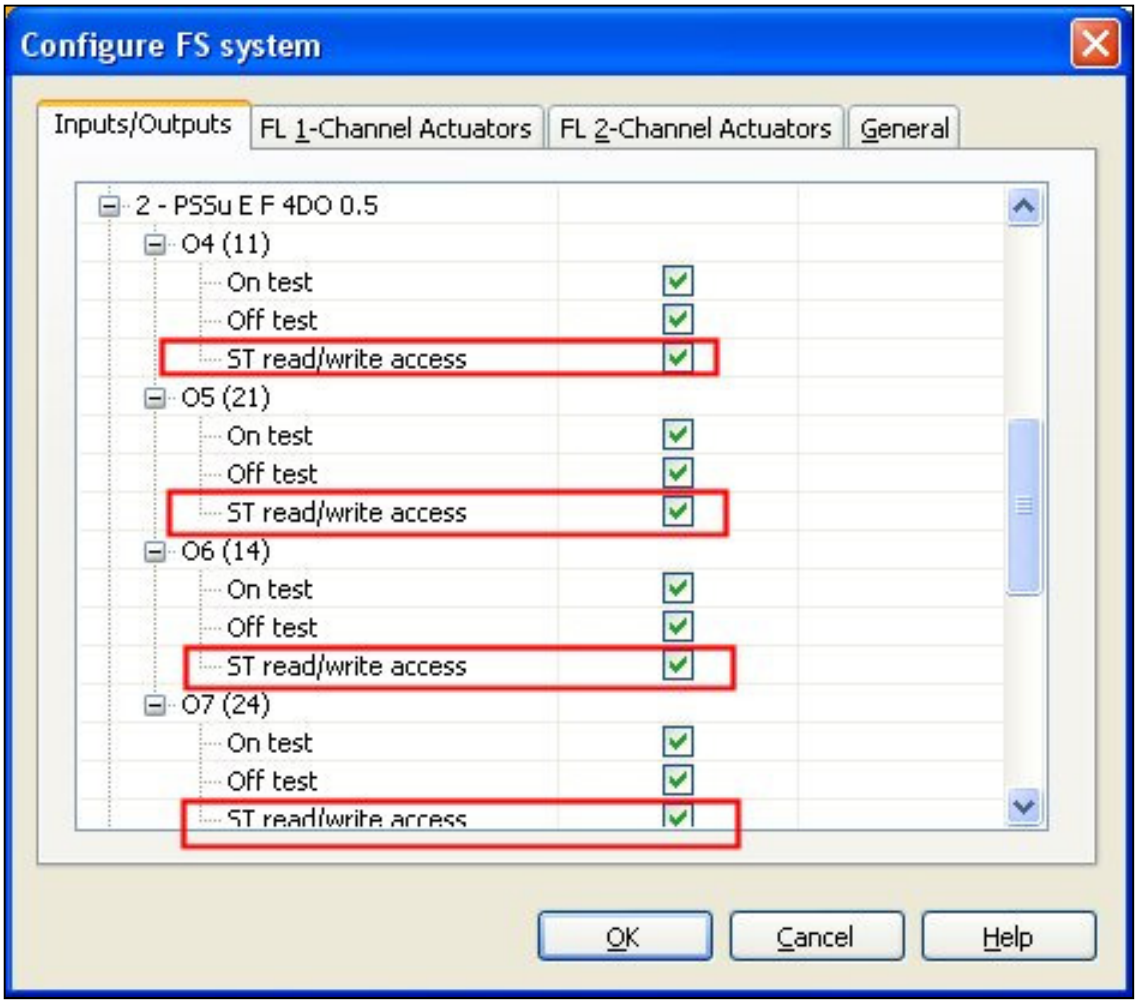


Fig. 37: PSSu local enabling of Slot 2

- Configuration of Slots 5/6

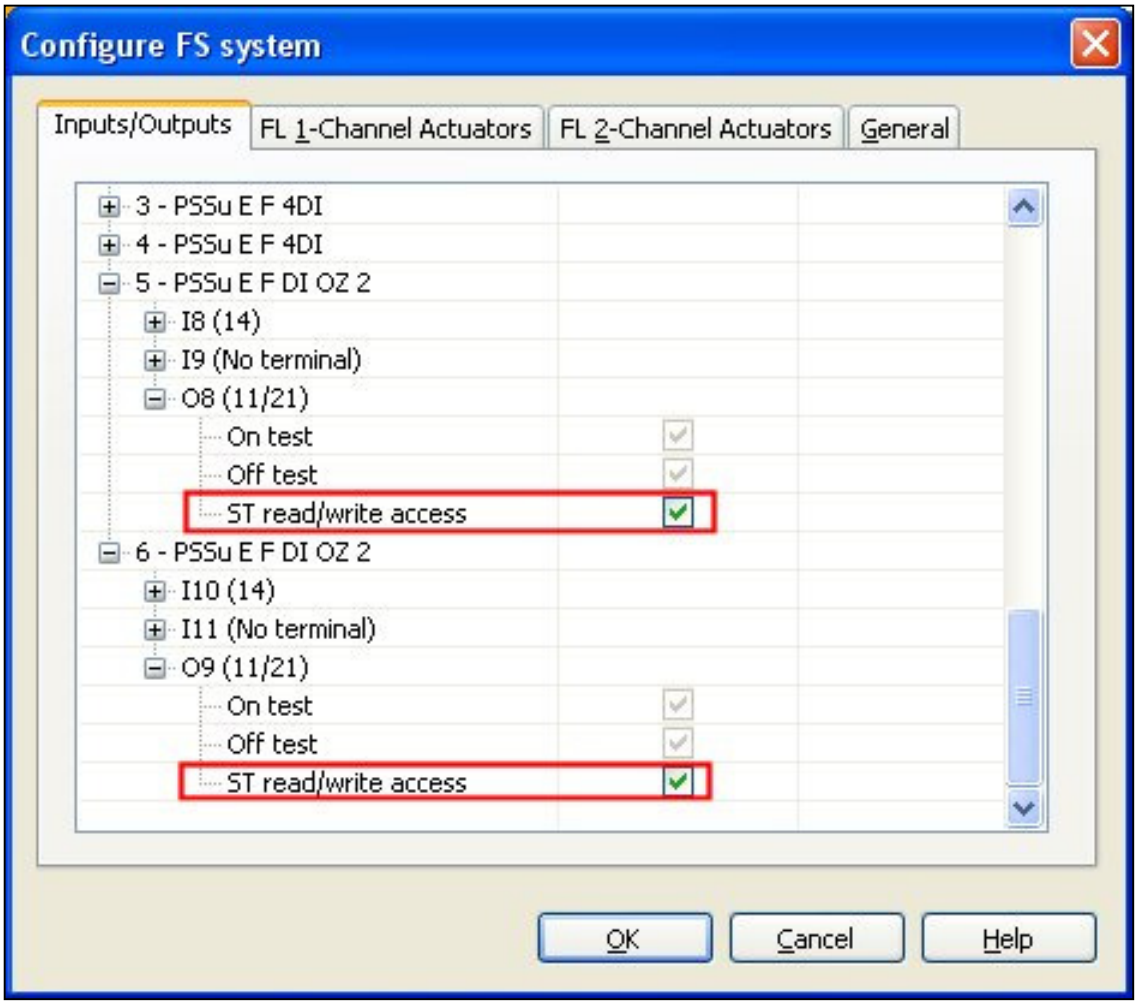


Fig. 38: PSSu local enabling of Slot 5 & Slot 6

5.13.4. PSS FS-Configuration of Feedback loop:

5.13.4.1. General feedback loop monitoring

A PSSu system with SafetyBUS p interface provides various options for monitoring the feedback loop. Basically there are two different procedures:

5.13.4.2. Feedback loop monitoring via the FS user program of the PSS

Pilz's FS SB065 is available for monitoring the feedback loop within a SafetyBUS p network. FS SB065 is a component part of various Pilz software packages (e.g. MBS E-STOP circuits, MBS eccentric presses, MBS hydraulic presses).
We recommend that you use FS SB065 if you require feedback loop monitoring but are working without the local enable principle. A description of FS SB065 can be found in the operating manual for the relevant software package.

5.13.4.3. Feedback loop monitoring via the PSSu system

In the PSSuniversal Configurator it is possible to configure feedback loop monitoring and to load the configuration into the PSSu system along with the other configuration data. In this case, the PSSu operating system performs the feedback loop monitoring function.
This type of feedback loop monitoring should always be used if you are using feedback loop monitoring in conjunction with the local enable principle.

- INFORMATION:
  Please refer also to the information provided on feedback loop monitoring in the PSS and SafetyBUS p Application Manual.

- Start configuration of FS-System:

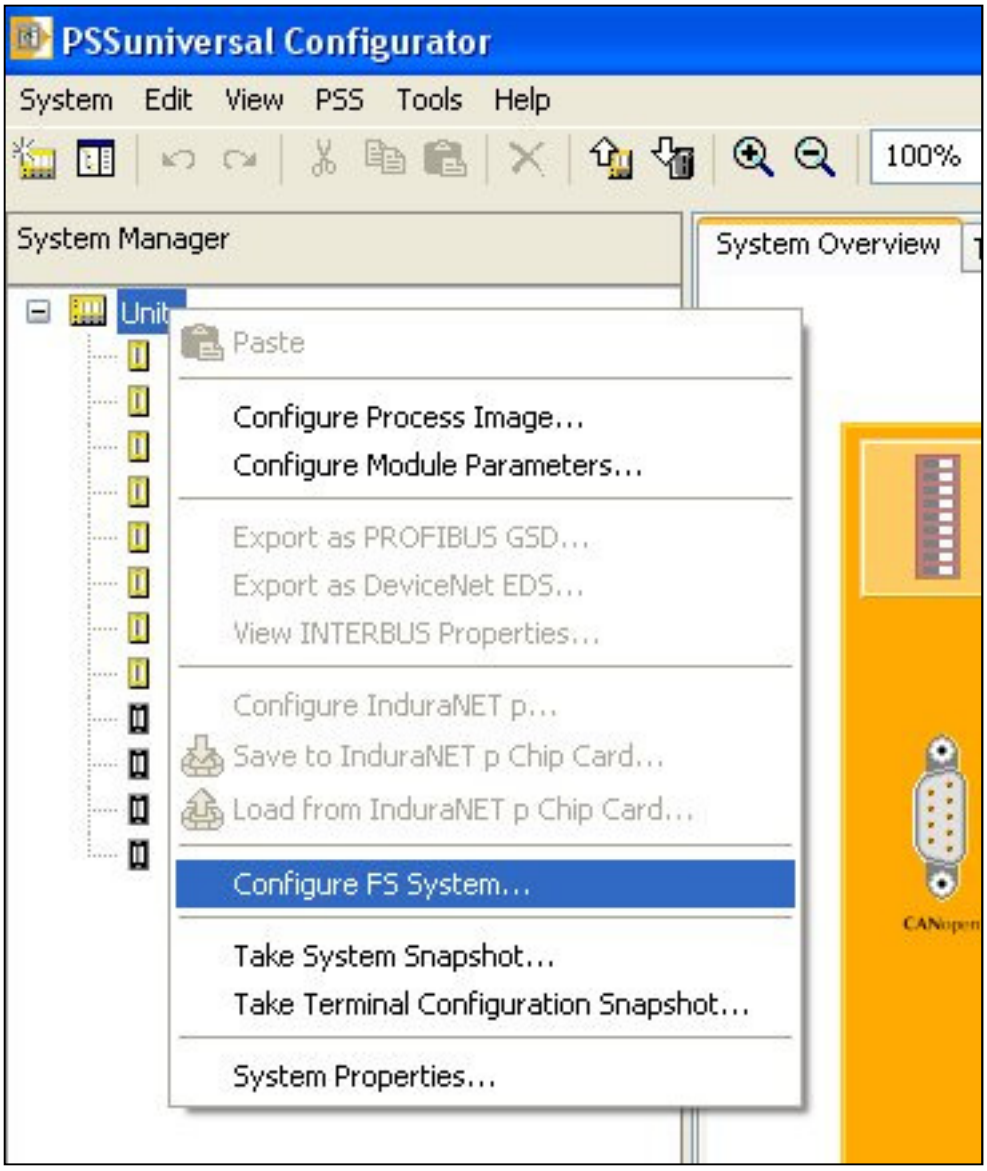


Fig. 39: Start with the menu “Configure FS System”

- Start configuration of feedback loop:

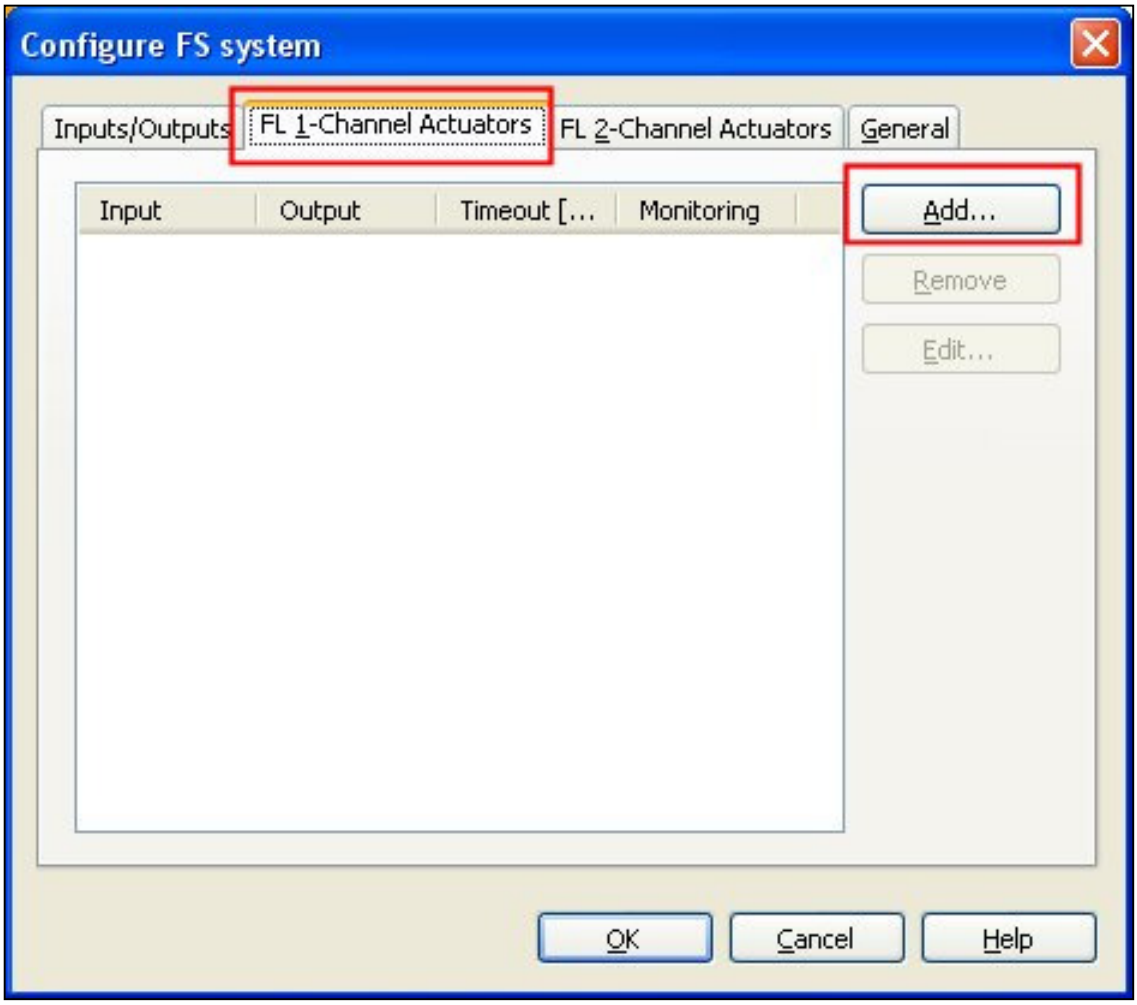


Fig. 40: Select Single Channel Feedback loop Monitoring

- Add feedback loop:

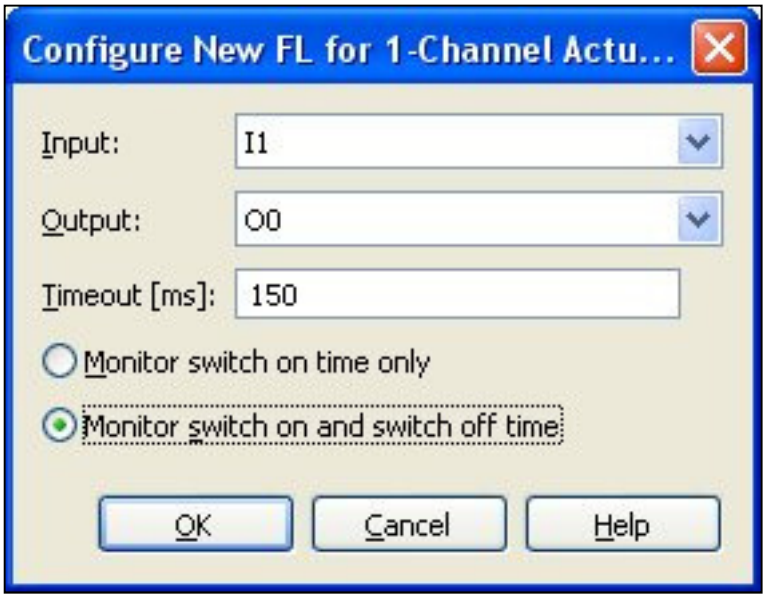


Fig. 41: Select the PSSu FS Relay Output and the Feedback loop Input.

- Note:
  If an Input is configured to a Feedback loop function the status is not the physical input anymore. The status is the “enabled” signal of the Feedback loop monitoring (RLO-1 = Feedback loop is ok.)

- Configuration of feedback loop is complete:

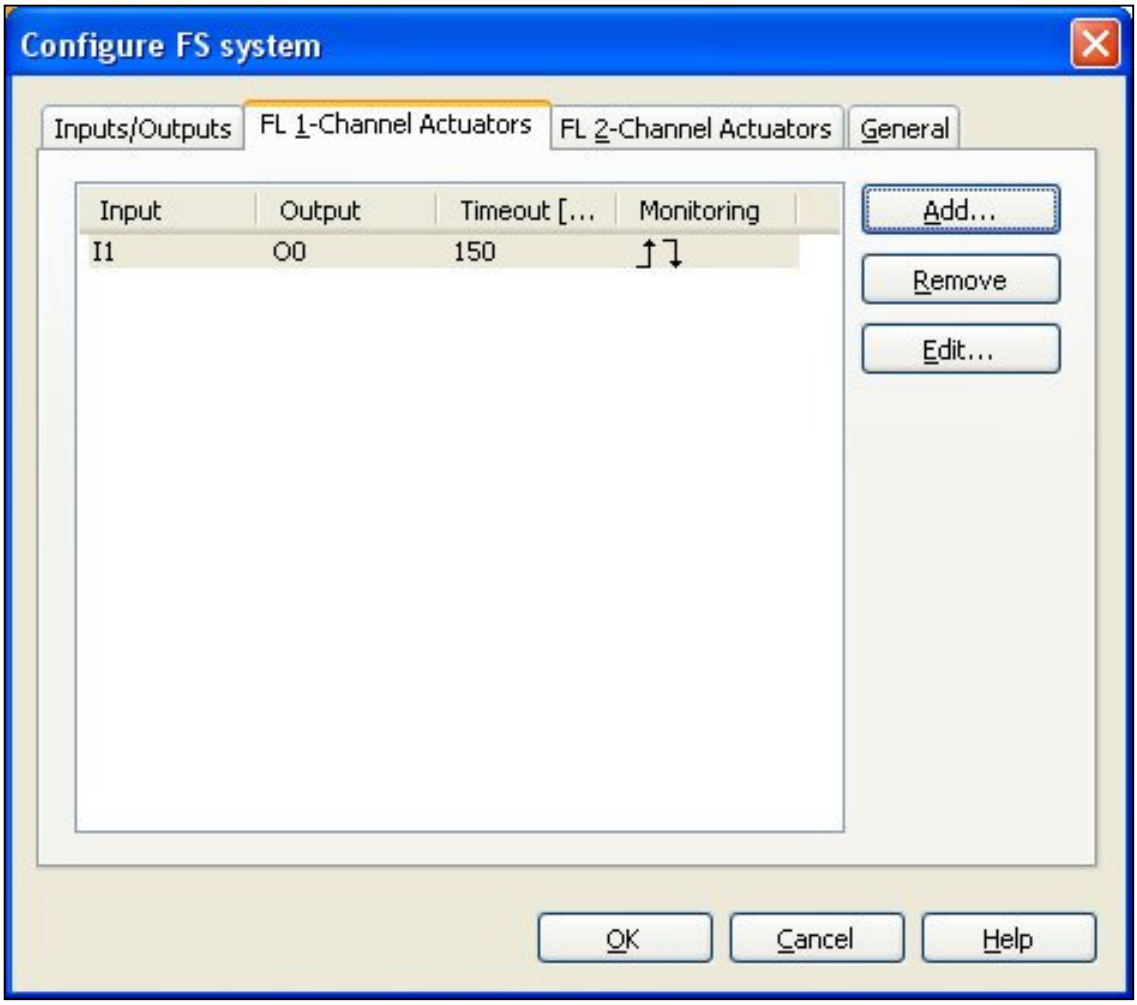


Fig. 42: Overview of the PSSu-System Feedback loop configuration

- Note:
  There are two different diagnostic results on the PSSu Feedback loop detection.
  In any configuration case a “Switch on Error” will cause a diagnostic warning (F83-01 Error caused by Feedback loop) and the “Switch off Error” will cause a SafetyBUS I-Group stop.

5.13.4.4. ST-Configuration to get a view of the Process Image:

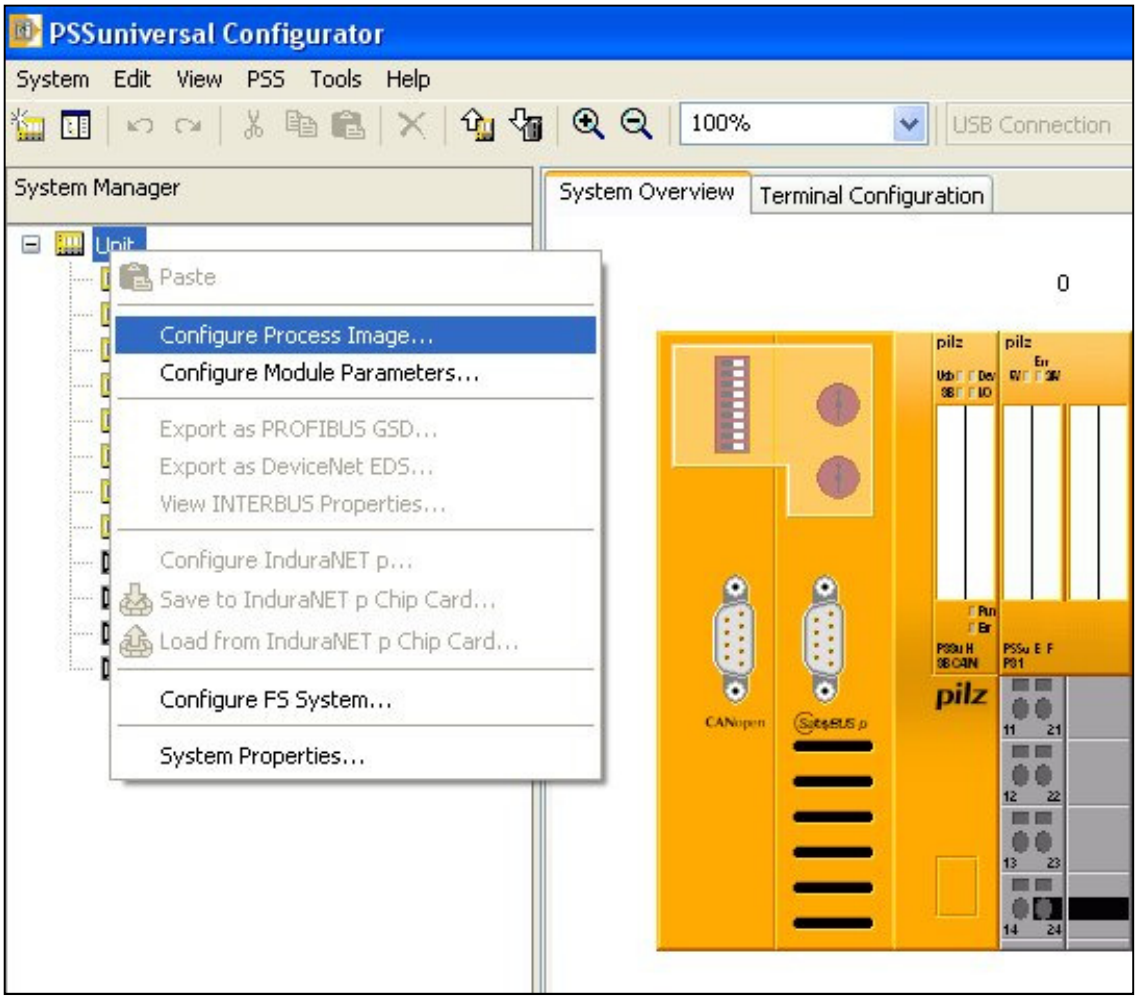


Fig. 43: Configure Process Image (Process Image view only)

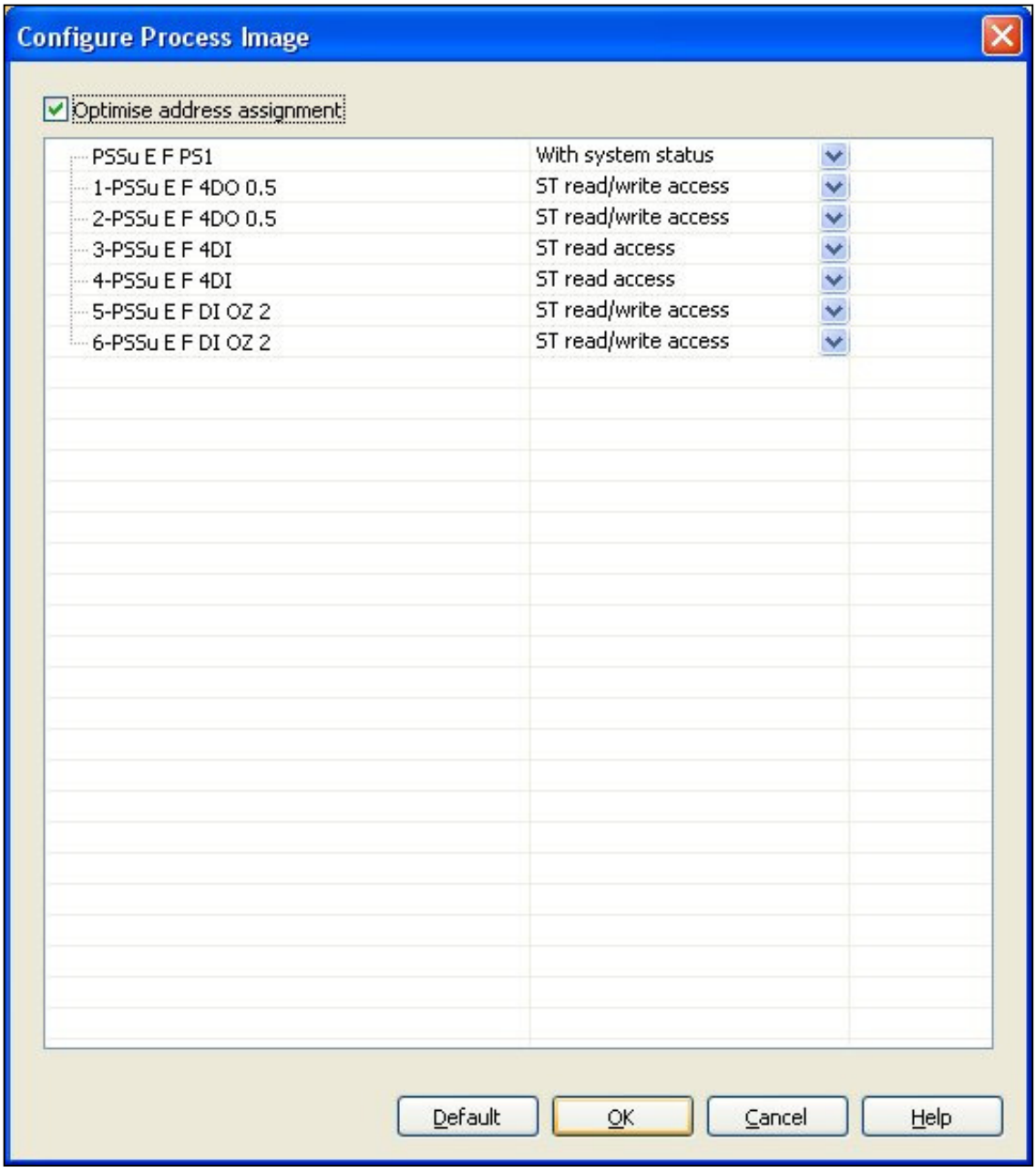


Fig. 44: Get all possible PSSu FS informations and ST write access to the FS Outputs

	Byte	Bit	Slot	Clamp	FS I/O	Electronic Module	CANopen			
CAN Master Input	0	0	0	-	SBp: System status STOP/RUN	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 0	
CAN Master Input	0	1	0	-	SBp: Status FS inputs: STOP/RUN	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 1	
CAN Master Input	0	2	0	-	SBp: Status FS outputs: STOP/RUN	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 2	
CAN Master Input	0	3	0	-	General FS system status: no error/error	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 3	
CAN Master Input	0	4	0	-	General ST system status: no error/error	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 4	
CAN Master Input	0	5	0	-	FS module status: no error/ error (min. one module)	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 5	
CAN Master Input	0	6	0	-	ST module status: no error/ error (min. one module)	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 6	
CAN Master Input	0	7	0	-	Reserved	PSSu E F PS1	COB-ID 0x181	Byte 0	Bit 7	
CAN Master Input	1	0	1	11	PSS PIO A33. 0	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 0	
CAN Master Input	1	1	1	21	PSS PIO A33. 1	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 1	
CAN Master Input	1	2	1	14	PSS PIO A33. 2	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 2	
CAN Master Input	1	3	1	24	PSS PIO A33. 3	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 3	
CAN Master Input	1	4	2	11	PSS PIO A33. 4	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 4	
CAN Master Input	1	5	2	21	PSS PIO A33. 5	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 5	
CAN Master Input	1	6	2	14	PSS PIO A33. 6	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 6	
CAN Master Input	1	7	2	24	PSS PIO A33. 7	PSSu E F 4DO 0.5	COB-ID 0x181	Byte 1	Bit 7	
CAN Master Input	2	0	3	11	E 33.00	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 0	
CAN Master Input	2	1	3	21	E 33.01	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 1	
CAN Master Input	2	2	3	14	E 33.02	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 2	
CAN Master Input	2	3	3	24	E 33.03	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 3	
CAN Master Input	2	4	4	11	E 33.04	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 4	
CAN Master Input	2	5	4	21	E 33.05	PSSu E F 4DI	COB-ID 0x181	Byte 2	Bit 5	
CAN Master Input	2	6	4	14	E 33.06	PSSu E F 4DI	COB-ID 0x181	Byte 3	Bit 6	
CAN Master Input	2	7	4	24	E 33.07	PSSu E F 4DI	COB-ID 0x181	Byte 3	Bit 7	
CAN Master Input	3	0	5	11-21	PSS PIO A33. 8	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 0	
CAN Master Input	3	1	5	14	E 33.08	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 1	
CAN Master Input	3	2	5	No terminal	E 33.09	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 2	no wire break
CAN Master Input	3	3	6	11-21	PSS PIO A33. 9	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 3	
CAN Master Input	3	4	6	14	E 33.010	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 4	
CAN Master Input	3	5	6	No terminal	E 33.011	PSSu E F DI OZ 2	COB-ID 0x181	Byte 3	Bit 5	no wire break
CAN Master Input	4	0	9	11		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 6	ST-Input 00
CAN Master Input	4	1	9	21		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 7	ST-Input 01
CAN Master Input	4	2	9	14		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 0	ST-Input 02
CAN Master Input	4	3	9	24		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 1	ST-Input 03
CAN Master Input	4	4	10	11		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 2	ST-Input 04
CAN Master Input	4	5	10	21		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 3	ST-Input 05
CAN Master Input	4	6	10	14		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 4	ST-Input 06
CAN Master Input	4	7	10	24		PSSu E S 4DI	COB-ID 0x181	Byte 4	Bit 5	ST-Input 07

	Byte	Bit	Slot	Clamp	FS I/O	Electronic Module	CANopen			
CAN Master Output	0	0	1	11	PSS local enable A33. 0	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 0	
CAN Master Output	0	1	1	21	PSS local enable A33. 1	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 1	
CAN Master Output	0	2	1	14	PSS local enable A33. 2	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 2	
CAN Master Output	0	3	1	24	PSS local enable A33. 3	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 3	
CAN Master Output	0	4	2	11	PSS local enable A33. 4	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 4	
CAN Master Output	0	5	2	21	PSS local enable A33. 5	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 5	
CAN Master Output	0	6	2	14	PSS local enable A33. 6	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 6	
CAN Master Output	0	7	2	24	PSS local enable A33. 7	PSSu E F 4DO 0.5	COB-ID 0x201	Byte 0	Bit 7	
CAN Master Output	1	0	5	11-21	PSS local enable A33. 8	PSSu E F DI OZ 2	COB-ID 0x201	Byte 1	Bit 0	
CAN Master Output	1	3	6	11-21	PSS local enable A33. 9	PSSu E F DI OZ 2	COB-ID 0x201	Byte 1	Bit 1	
CAN Master Output	2	0	7	11		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 2	ST-Output 00
CAN Master Output	2	1	7	21		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 3	ST-Output 01
CAN Master Output	2	2	7	14		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 4	ST-Output 02
CAN Master Output	2	3	7	24		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 5	ST-Output 03
CAN Master Output	2	4	8	11		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 6	ST-Output 04
CAN Master Output	2	5	8	21		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 1	Bit 7	ST-Output 05
CAN Master Output	2	6	8	14		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 2	Bit 0	ST-Output 06
CAN Master Output	2	7	8	24		PSSu E S 4DO 0.5	COB-ID 0x201	Byte 2	Bit 1	ST-Output 07

Fig. 45: Physical and non physical I/Os and used COB-IDs

PSSu Device Hardware Registry							
			Object	Sub-index		decimal	hexadecimal
Slot	00	04 : PSSu E F PS1 R	[3001,	1	]:=	04	4
Slot	01	21 : PSSu E F 4DO 0.5 &	[3001,	2	]:=	21	15
Slot	02	22 : PSSu E F 4DO 0.5 &*	[3001,	3	]:=	22	16
Slot	03	16 : PSSu E F 4DI R	[3001,	4	]:=	16	10
Slot	04	17 : PSSu E F 4DI R*	[3001,	5	]:=	17	11
Slot	05	36 : PSSu E F DI OZ 2 &	[3001,	6	]:=	36	24
Slot	06	37 : PSSu E F DI OZ 2 &*	[3001,	7	]:=	37	25
Slot	07	11 : PSSu E S 4DO 0.5	[3001,	8	]:=	11	B
Slot	08	12 : PSSu E S 4DO 0.5*	[3001,	9	]:=	12	C
Slot	09	09 : PSSu E S 4DI	[3001,	A	]:=	09	9
Slot	10	10 : PSSu E S 4DI*	[3001,	B	]:=	10	A

Fig. 46: PSSu Hardware Registry and Object 3001 configuration

5.13.5. PDO-mapping

	Time	Chn	Dir	ID	Name	NodeTransfer data	ErrorInterpretation
1							
2							
3							
4	15.429869	1	Rx	58A			[140b,01] Initiate Download Rsp
5	15.516980	1	Rx	60A		8a 01 00 80	[1800,01] Initiate Download Rq. expedited
6	15.517472	1	Rx	58A			[1800,01] Initiate Download Rsp
7	15.519149	1	Rx	60A		00	[1a00,00] Initiate Download Rq. expedited
8	15.524970	1	Rx	58A			[1a00,00] Initiate Download Rsp
9	15.525395	1	Rx	60A		08 01 01 2f [2f01,01]	[1a00,01] Initiate Download Rq. expedited ".../"
10	15.526244	1	Rx	58A			[1a00,01] Initiate Download Rsp
11	15.526672	1	Rx	60A		08 02 01 2f [2f01,02]	[1a00,02] Initiate Download Rq. expedited ".../"
12	15.527303	1	Rx	58A			[1a00,02] Initiate Download Rsp
13	15.527728	1	Rx	60A		08 03 01 2f [2f01,03]	[1a00,03] Initiate Download Rq. expedited ".../"
14	15.528369	1	Rx	58A			[1a00,03] Initiate Download Rsp
15	15.528781	1	Rx	60A		08 04 01 2f [2f01,04]	[1a00,04] Initiate Download Rq. expedited ".../"
16	15.529430	1	Rx	58A			[1a00,04] Initiate Download Rsp
17	15.529837	1	Rx	60A		08 05 01 2f [2f01,05]	[1a00,05] Initiate Download Rq. expedited ".../"
18	15.530672	1	Rx	58A			[1a00,05] Initiate Download Rsp
19	15.531060	1	Rx	60A		05	[1a00,00] Initiate Download Rq. expedited
20	15.534666	1	Rx	58A			[1a00,00] Initiate Download Rsp
21	15.535109	1	Rx	60A		ff	[1800,02] Initiate Download Rq. expedited
22	15.535455	1	Rx	58A			[1800,02] Initiate Download Rsp
23	15.535876	1	Rx	60A		00 00	[1800,03] Initiate Download Rq. expedited
24	15.536438	1	Rx	58A			[1800,03] Initiate Download Rsp
25	15.536873	1	Rx	60A		81 01 00 00	[1800,01] Initiate Download Rq. expedited
26	15.537582	1	Rx	58A			[1800,01] Initiate Download Rsp
27							
32	19.537643	1	Rx	60A		04 00	[3001,01] Initiate Download Rq. expedited
33	19.538701	1	Rx	58A			[3001,01] Initiate Download Rsp
34	19.539089	1	Rx	60A		15 00	[3001,02] Initiate Download Rq. expedited
35	19.539854	1	Rx	58A			[3001,02] Initiate Download Rsp
36	19.540258	1	Rx	60A		16 00	[3001,03] Initiate Download Rq. expedited
37	19.541046	1	Rx	58A			[3001,03] Initiate Download Rsp
38	19.541478	1	Rx	60A		10 00	[3001,04] Initiate Download Rq. expedited
39	19.542189	1	Rx	58A			[3001,04] Initiate Download Rsp
40	19.542590	1	Rx	60A		11 00	[3001,05] Initiate Download Rq. expedited
41	19.543335	1	Rx	58A			[3001,05] Initiate Download Rsp
42	19.543761	1	Rx	60A		24 00	[3001,06] Initiate Download Rq. expedited "$."
43	19.544499	1	Rx	58A			[3001,06] Initiate Download Rsp
44	19.544926	1	Rx	60A		25 00	[3001,07] Initiate Download Rq. expedited "%."
45	19.545657	1	Rx	58A			[3001,07] Initiate Download Rsp
46	19.546087	1	Rx	60A		0b 00	[3001,08] Initiate Download Rq. expedited
47	19.546995	1	Rx	58A			[3001,08] Initiate Download Rsp
48	19.547425	1	Rx	60A		0c 00	[3001,09] Initiate Download Rq. expedited
49	19.548143	1	Rx	58A			[3001,09] Initiate Download Rsp
50	19.548593	1	Rx	60A		09 00	[3001,0a] Initiate Download Rq. expedited
51	19.549293	1	Rx	58A			[3001,0a] Initiate Download Rsp
52	19.549701	1	Rx	60A		0a 00	[3001,0b] Initiate Download Rq. expedited
53	19.550461	1	Rx	58A			[3001,0b] Initiate Download Rsp
62	19.620393	1	Rx	60A		0a 02 00 80	[1400,01] Initiate Download Rq. expedited
63	19.620768	1	Rx	58A			[1400,01] Initiate Download Rsp
64	19.622506	1	Rx	60A		00	[1600,00] Initiate Download Rq. expedited
65	19.622830	1	Rx	58A			[1600,00] Initiate Download Rsp
66	19.623226	1	Rx	60A		08 01 00 22 [2200,01]	[1600,01] Initiate Download Rq. expedited "...""
67	19.623920	1	Rx	58A			[1600,01] Initiate Download Rsp
68	19.624336	1	Rx	60A		08 02 00 22 [2200,02]	[1600,02] Initiate Download Rq. expedited "...""
69	19.624767	1	Rx	58A			[1600,02] Initiate Download Rsp
70	19.625163	1	Rx	60A		08 01 00 62 [6200,01]	[1600,03] Initiate Download Rq. expedited "...b"
71	19.626038	1	Rx	58A			[1600,03] Initiate Download Rsp
72	19.626447	1	Rx	60A		03	[1600,00] Initiate Download Rq. expedited
73	19.627445	1	Rx	58A			[1600,00] Initiate Download Rsp
74	19.627841	1	Rx	60A		ff	[1400,02] Initiate Download Rq. expedited
75	19.628210	1	Rx	58A			[1400,02] Initiate Download Rsp
76	19.628617	1	Rx	60A		01 02 00 00	[1400,01] Initiate Download Rq. expedited
77	19.629393	1	Rx	58A			[1400,01] Initiate Download Rsp

Fig. 47: CAN trace of PDO mapping

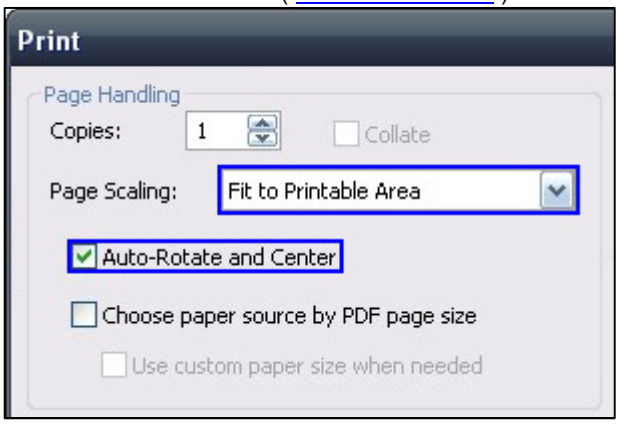
- Explanation:
  - 1. Delete PDO identifier (COB-ID, Bit 31) in the TPDO1 communication parameter (line 5 & line 62).
  - 2. Set mapped objects in the PDO mapping parameter to zero. (line 7 & line 64)
  - 3. Map the required objects to the PDO. (lines 9 to 17 & 66 to 70):
  - 4. Enter the sum total of mapped objects in the TPDO 1 mapping parameter = 4 (line 19 & line 72)
  - 5. Activate PDO identifier (COB-ID, Bit 31) in the TPDO1 communication parameter COB-ID 0x181 (line 25 & line 76)
  -
  - 6. Change the hardware registry of the object 3001 configuration of each second to PSSu S T DI4* (line 32 to 52)
  - 7. Transmission type = 255 = event driven (line 21 & line 74)
  - 8. TPDO-Inhibit-Time (line 23)

- Note:
  For sophisticated trouble shooting you need a CANalyzer.

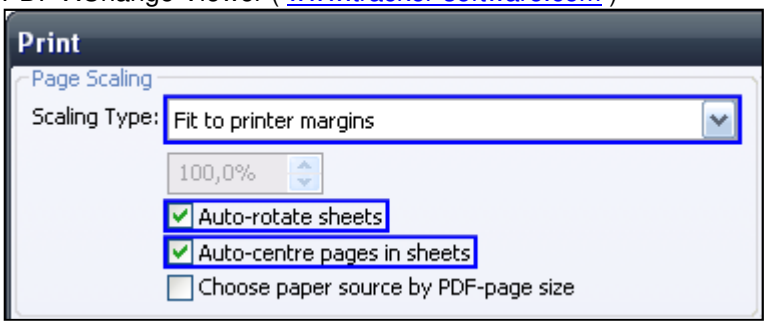
6. Table of figures

Recommended printer settings

Adobe Acrobat Reader (www.adobe.com)



PDF-XChange Viewer (www.tracker-software.com)



Technical support
+49 711 3409-444
support@pilz.com

...
In many countries we are represented by our subsidiaries and sales partners.

Please refer to our homepage for further details or contact our headquarters.

Pilz GmbH & Co. KG
Felix-Wankel-Straße 2
73760 Ostfildern, Germany
Telephone: +49 711 3409-0
Telefax: +49 711 3409-133
E-Mail: pilz.gmbh@pilz.de
Internet: www.pilz.com